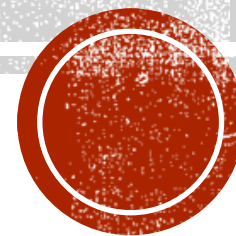


多媒體實習-OpenCV教學2



OUTLINE

- 平滑法 (Smoothing Method)
- 高斯濾波器 (Gaussian Filter)
- 中值濾波器 (Median Filter)
- 二值化 (Binarization)



平滑法

Filter 會掃過範圍內每個位置

10	10	10	10
10	150	200	10
10	10	0	10
10	10	10	10

image

如果是 5*5 的 Filter size，各個位置的權重就會變成 1/25

*

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

filter



其實各個？就是對分別掃到的 Mask 範圍內取平均

10	10	10	10
10	?	?	10
10	?	?	10
10	10	10	10



平滑法

10	10	10	10
10	150	200	10
10	10	0	10
10	10	10	10

image

*

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

filter

$$(10+10+10+10+150+200+10+10+0)/9$$



10	10	10	10
10	45	?	10
10	?	?	10
10	10	10	10



平滑法

10	10	10	10
10	150	200	10
10	10	0	10
10	10	10	10

image

*

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

filter

$$(10+10+10+10+150+200+10+10+0)/9$$



10	10	10	10
10	45	45	10
10	?	?	10
10	10	10	10



平滑法

10	10	10	10
10	150	200	10
10	10	0	10
10	10	10	10

image

*

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

filter

$$(10+10+10+10+150+200+10+10+0)/9$$



10	10	10	10
10	45	45	10
10	45	?	10
10	10	10	10



平滑法

10	10	10	10
10	150	200	10
10	10	0	10
10	10	10	10

image

*

$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$

filter

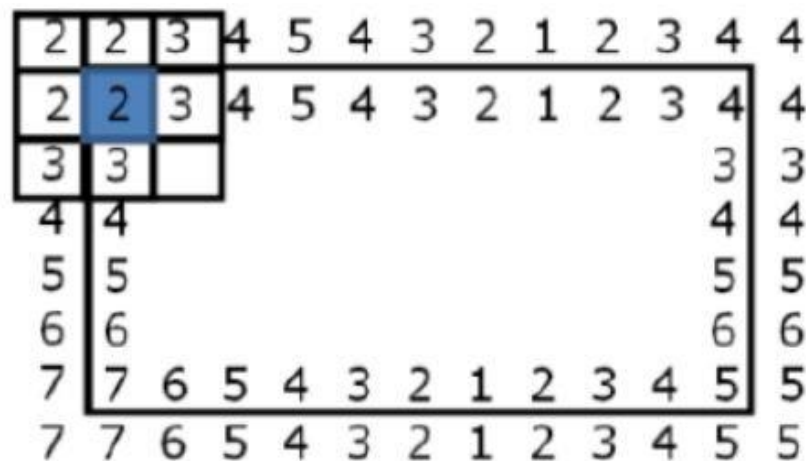
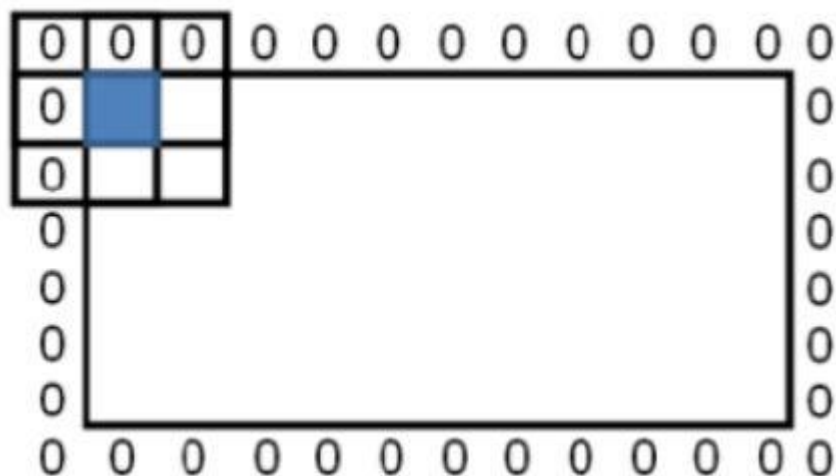


10	10	10	10
10	45	45	10
10	45	45	10
10	10	10	10



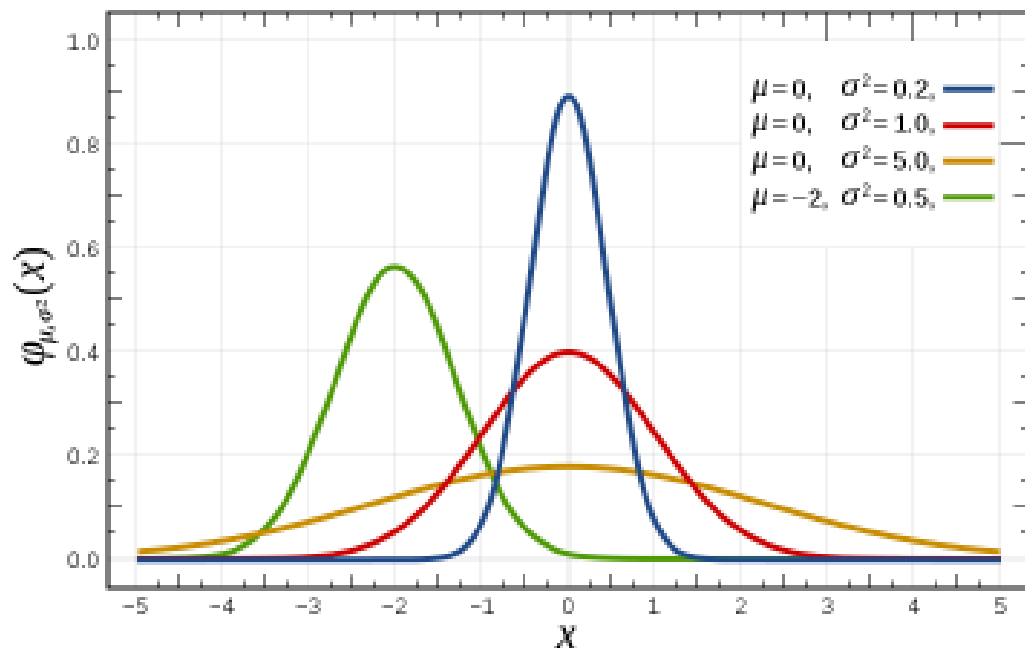
平滑法

- 周圍外框處理
 - 不管、補0、複製



高斯濾波器

- 功能: 濾除雜訊、低通、模糊化



$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

σ (sigma) 是標準差
 μ (mean) 是平均值



高斯濾波器

- 將此矩陣的 x,y 值以及 $\mu = 0, \sigma = 1$ 套入 Gaussian function 並正規化後即可得到 3*3 的 Gaussian filter
- 推導詳見 Excel

$$\begin{bmatrix} (-1, -1) & (0, -1) & (1, -1) \\ (-1, 0) & (0, 0) & (1, 0) \\ (-1, 1) & (0, 1) & (1, 1) \end{bmatrix}$$

矩陣的 x,y 值



$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

function



0.045	0.122	0.045
0.122	0.332	0.122
0.045	0.122	0.045

filter



中值濾波器

- 找遮罩的中位數，
然後取代中間的值，
當有數值差異大的
值出現時效果通常
會比平滑法好

10	10	10	10
10	150	200	10
10	10	0	10
10	10	10	10

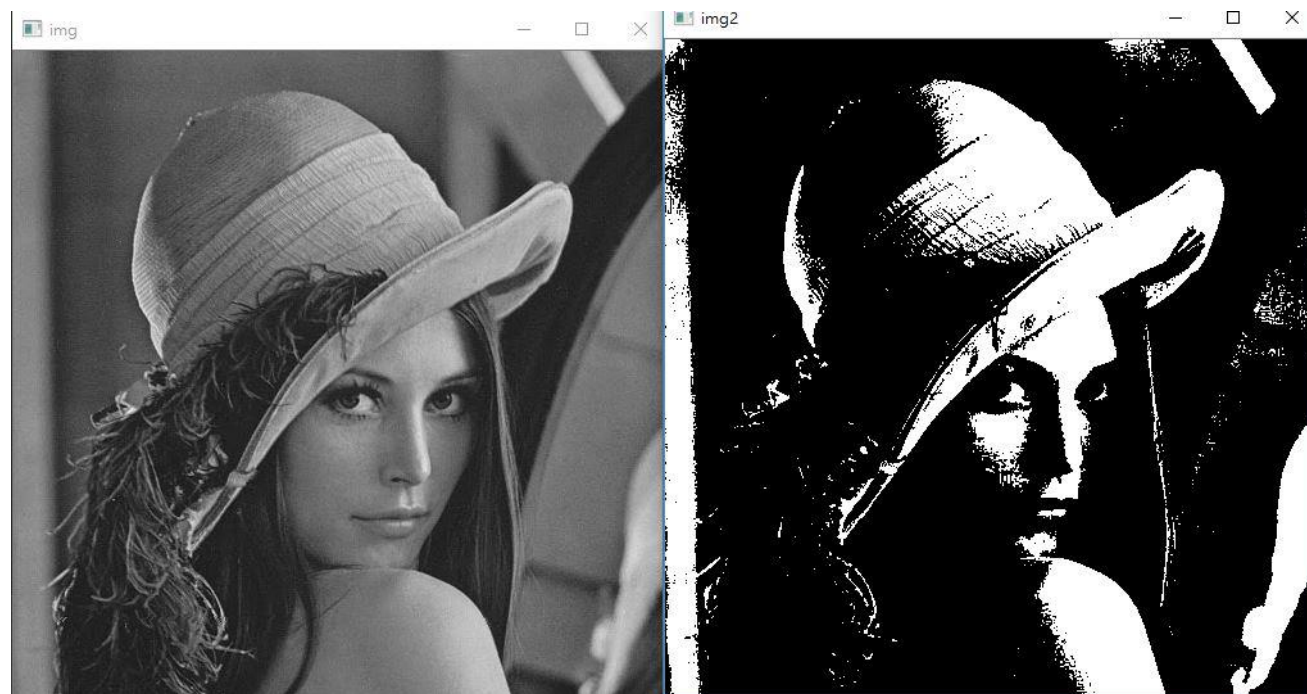


10	10	10	10
10	10	10	10
10	10	10	10
10	10	10	10



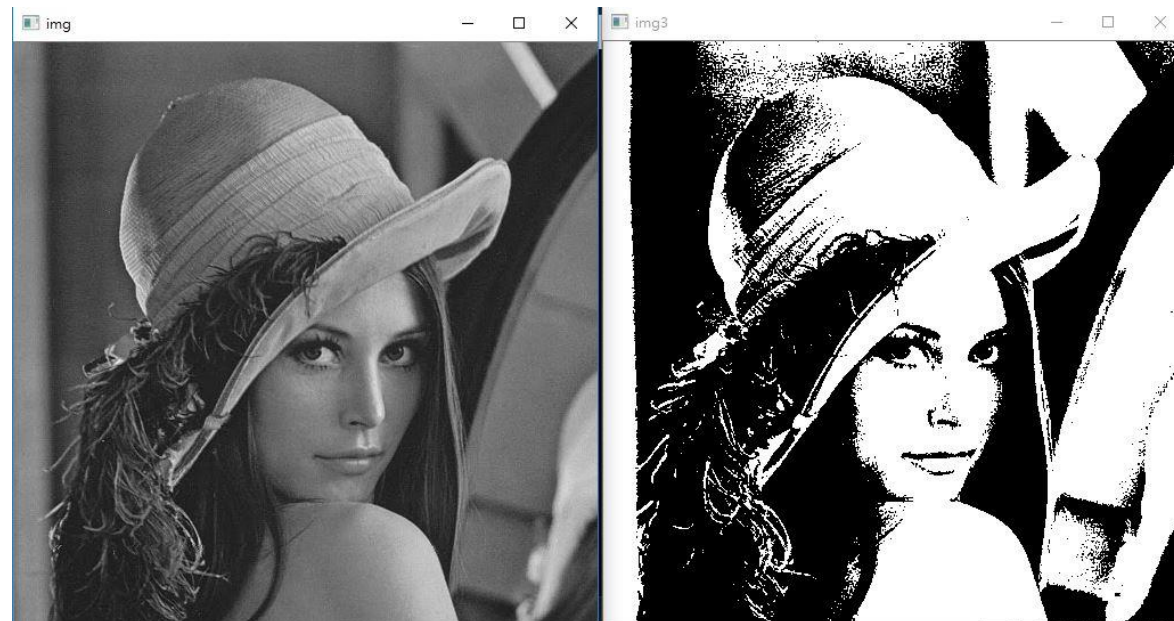
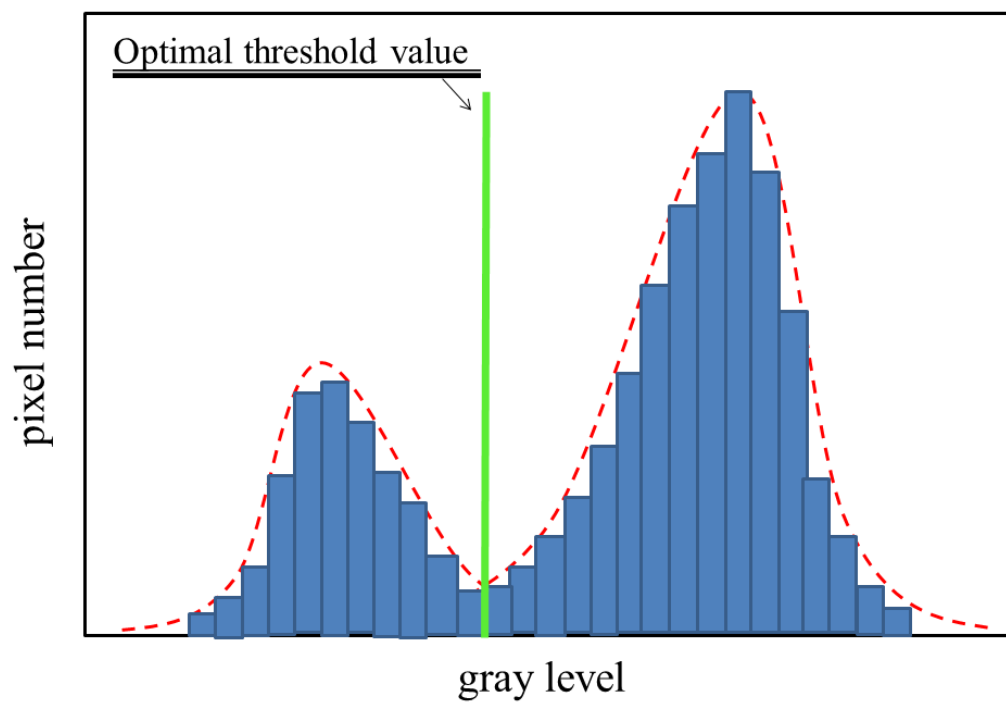
二值化

- `threshold(src, dst, 127, 255, THRESH_BINARY);`



OTSU自動門檻值

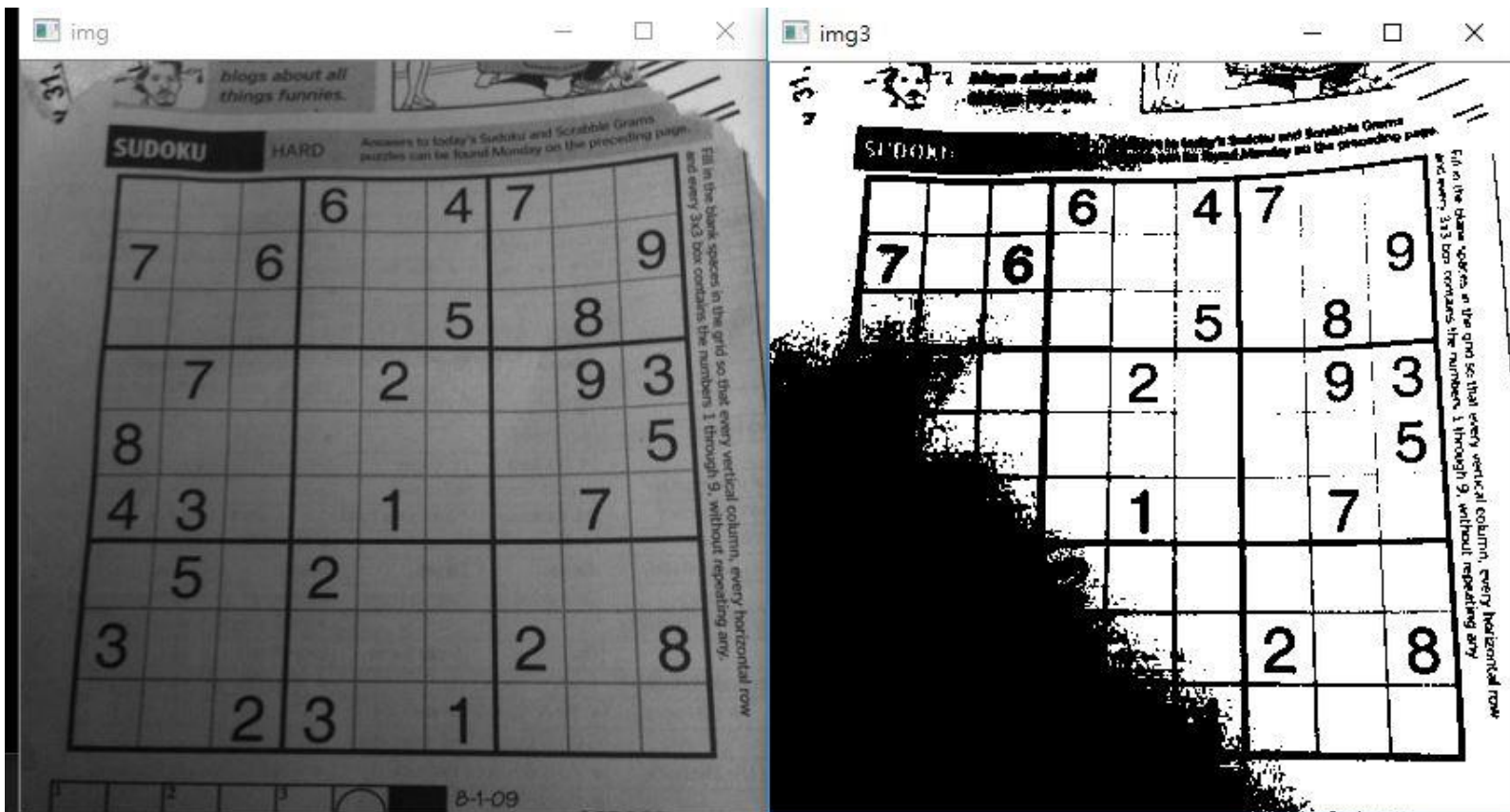
- `threshold(src, dst, 0, 255, THRESH_BINARY | THRESH_OTSU);`



二值化

暗

亮



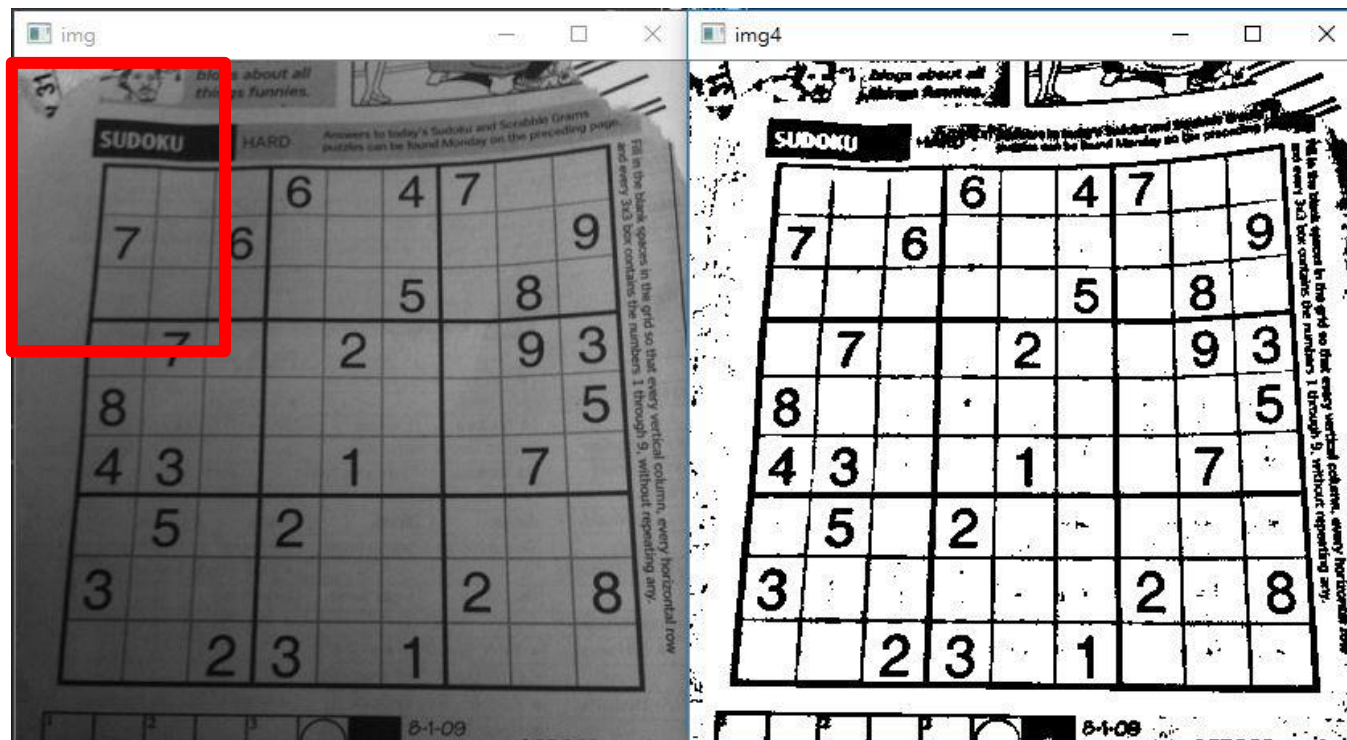
OTSU 二值化結果



二值化

- adaptiveThreshold

每一個點的閾值，
會根據周圍的點
決定



二值化

◆ adaptiveThreshold()

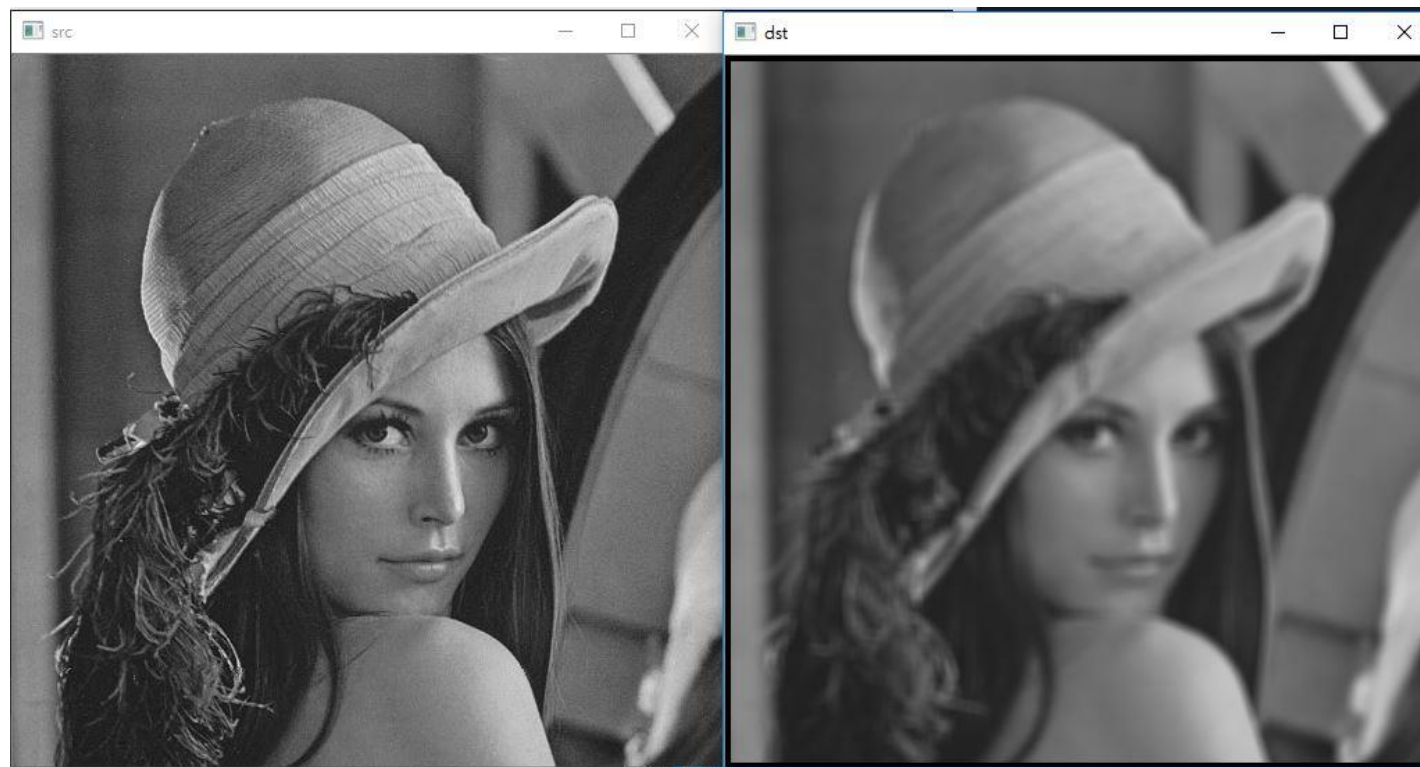
```
void cv::adaptiveThreshold ( InputArray   src,  
                             OutputArray dst,  
                             double        maxValue,  
                             int           adaptiveMethod,  
                             int           thresholdType,  
                             int           blockSize,  
                             double        C  
                             )
```

- **src**：輸入圖。
- **dst**：輸出圖。
- **maxValue**：最大值。
- **adaptiveMethod**：方法：
有 **ADAPTIVE_THRESH_MEAN_C** 和 **ADAPTIVE_THRESH_GAUSSIAN_C**
- **thresholdType**：二值化型態：
有 **THRESH_BINARY** 和 **THRESH_BINARY_INV**
- **blockSize**：遮罩大小。
- **C**：常數，計算閾值時，要從平均或加權平均減去的數。



練習

- 實作平滑法(5x5)，周圍不用處理



加分題

- 實作中值濾波(3x3)，周圍不用處理
- 可使用 algorithm library 中的 `sort()`
<http://www.cplusplus.com/reference/algorithm/sort/>

