# Principles and Applications of Digital Image Processing

B07611001 楊禮蔚

- Part 1: (25%)

**3.22** Answer the following:

**(a)*** If $\mathbf{v} = [1 \quad 2 \quad 1]^T$ and $\mathbf{w}^T = [2 \quad 1 \quad 1 \quad 3]$, is the kernel formed by $\mathbf{v}\mathbf{w}^T$ separable?

**(b)** The following kernel is separable. Find $w_1$ and $w_2$ such that $w = w_1 \bigstar w_2$.

$$w = \begin{bmatrix} 1 & 3 & 1 \\ 2 & 6 & 2 \end{bmatrix}$$

(a)$\mathrm{v} \otimes \mathrm{w}^T = \begin{bmatrix} 2 & 1 & 1 & 3 \\ 4 & 2 & 2 & 6 \\ 2 & 1 & 1 & 3 \end{bmatrix}$, it is separable because the 2-D function can be expressed as the outer product of two 1-D functions.

(b)$\mathrm{w} = \begin{bmatrix} 1 & 3 & 1 \\ 2 & 6 & 2 \end{bmatrix}$, $\mathrm{w} = w_1 \otimes w_2$, $w_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$, $w_2 = [1 \quad 3 \quad 1]$.

**3.27** An image is filtered four times using a Gaussian kernel of size $3 \times 3$ with a standard deviation of 1.0. Because of the associative property of convolution, we know that equivalent results can be obtained using a single Gaussian kernel formed by convolving the individual kernels.

**(a)*** What is the size of the single Gaussian kernel?

**(b)** What is its standard deviation?

(a)A Gaussian kernel is separable. $(\mathrm{w} \otimes \mathrm{w} \otimes \mathrm{w} \otimes \mathrm{w}) \otimes f = w' \otimes f$, the size of w' would be a 3*3 kernel convolute for times, so 3+4*(3-1) = 11. The size of w' would be 11*11.

(b)According to the formula, $\sigma_{f \otimes g} = \sqrt{\sigma_f^2 + \sigma_g^2}$, so the final σ of w' would be $\sqrt{4 * 1^2}$ = 2.

**3.38** In a given application, a smoothing kernel is applied to input images to reduce noise, then a Laplacian kernel is applied to enhance fine details. Would the result be the same if the order of these operations is reversed?

No. When the smoothing kernel applied first, the noise would decrease, so only edge would be enhance by Laplacian kernel. On the contrary, if the Laplacian kernel applied first, the noise and the edge would be both enhance by the operation, then the noise may not be reduce enough when the same smoothing kernel is applied.

**4.3** What is the convolution of two, 1-D impulses:

**(a)*** $\delta(t)$ and $\delta(t-t_0)$?

**(b)** $\delta(t-t_0)$ and $\delta(t+t_0)$?

(a) $\delta(t)\bigstar\delta(t-t_0)=\int_{-\infty}^{\infty}\delta(\tau)*\delta(t-t_0-\tau)d\tau$, the integral has value only when $\tau = 0$. So $\delta(t)\bigstar\delta(t-t_0)=\ \delta(t-t_0)$.

(b) $\delta(t+t_0)\bigstar\delta(t-t_0)=\int_{-\infty}^{\infty}\delta(\tau+t_0)*\delta(t-t_0-\tau)d\tau=\delta(t)$.

**4.32** We mentioned in Example 4.10 that embedding a 2-D array of even (odd) dimensions into a larger array of zeros of even (odd) dimensions keeps the symmetry of the original array, provided that the centers coincide. Show that this is true also for the following 1-D arrays (i.e., show that the larger arrays have the same symmetry as the smaller arrays). For arrays of even length, use arrays of 0's ten elements long. For arrays of odd lengths, use arrays of 0's nine elements long.

**(a)*** $\{a, b, c, c, b\}$

**(b)** $\{0, -b, -c, 0, c, b\}$

**(c)** $\{a, b, c, d, c, b\}$

**(d)** $\{0, -b, -c, c, b\}$

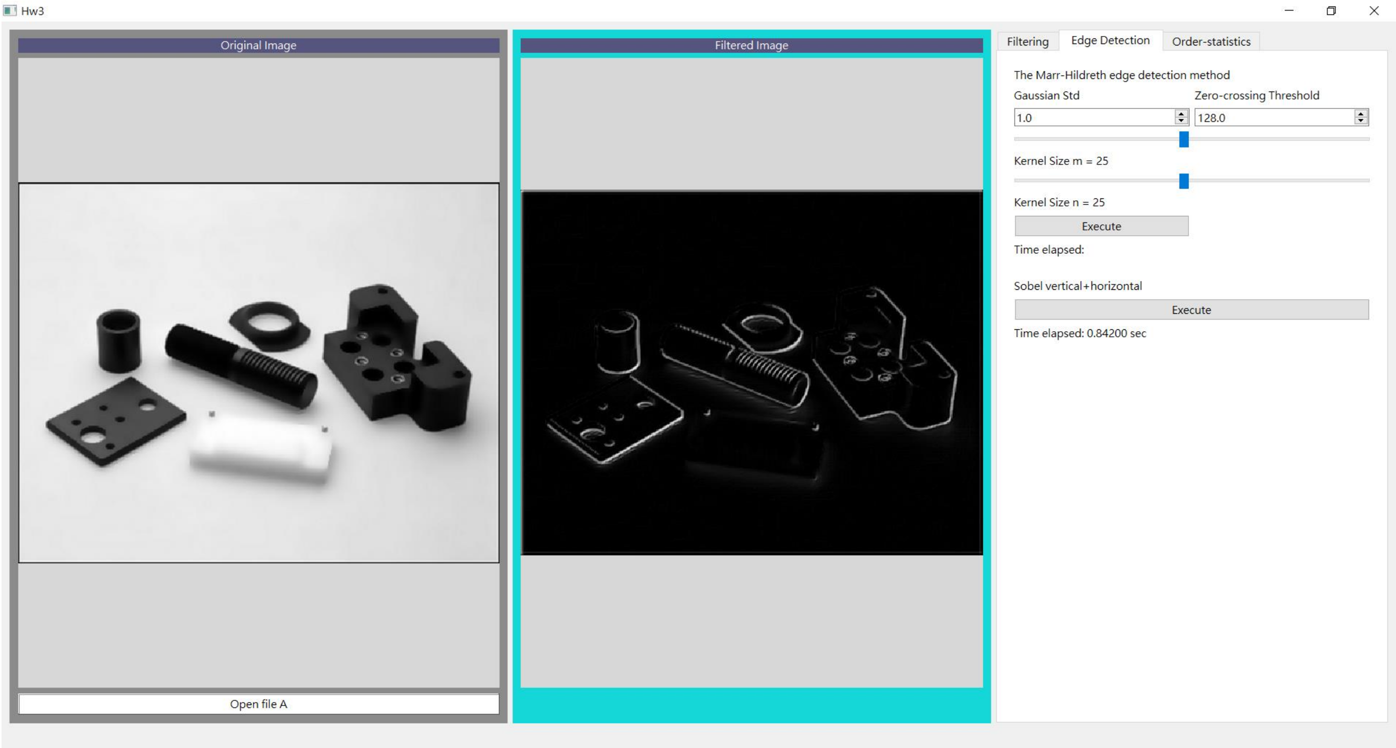
(a){0, 0, a, b, c, c, b, 0, 0}, originally even but now odd.

(b){0, 0, 0, -b, -c, 0, c, b, 0, 0}, still odd.

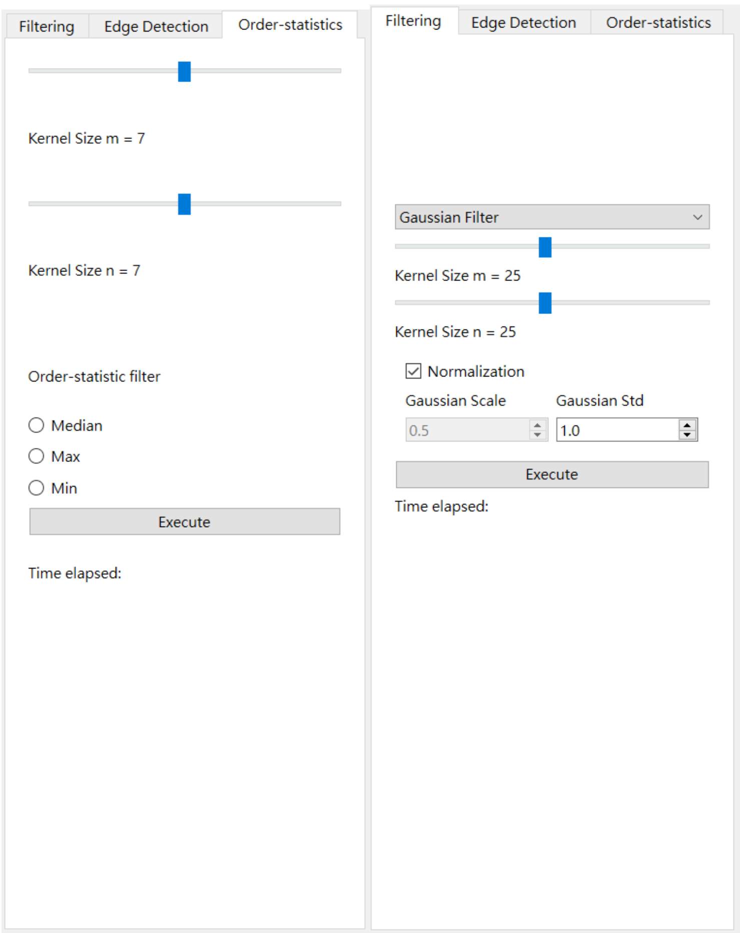
(c){0, 0, a, b, c, d, c, b, 0, 0}, originally even but now odd.

(d){0, 0, 0, -b, -c , c, 0, 0}, still odd.

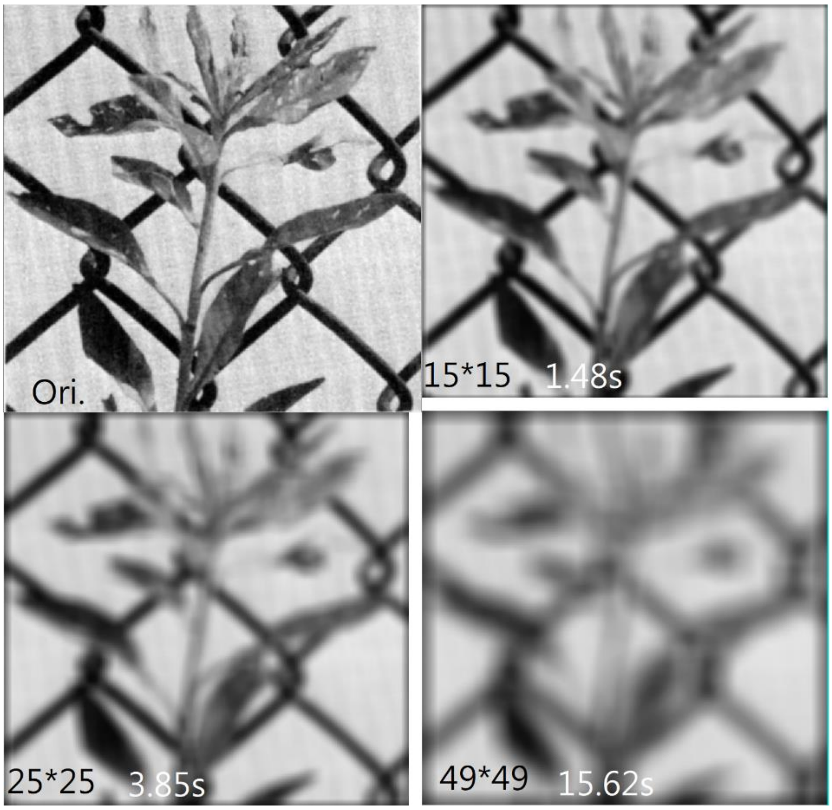
- Part 2: (25%), Part 3: (25%), Part 4: (25%):
  - UI:



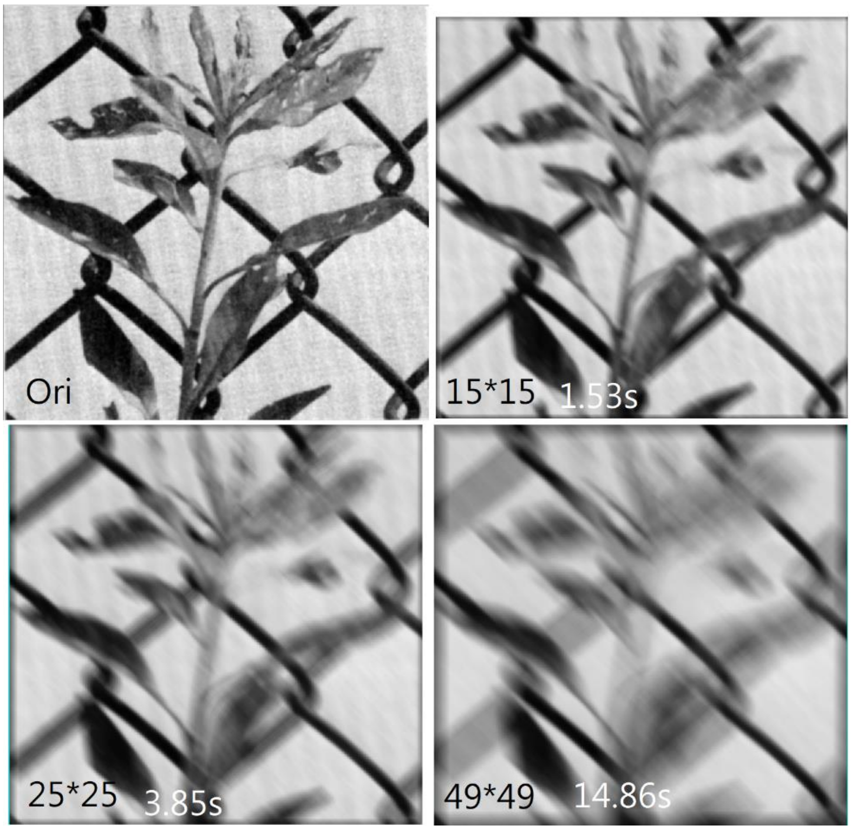

Use tab to switch between parts. When processing image, the cursor would turn into busy. When the operation is done, the application would beep in “La” for 500 ms, at the same time the calculation time would be shown in label.



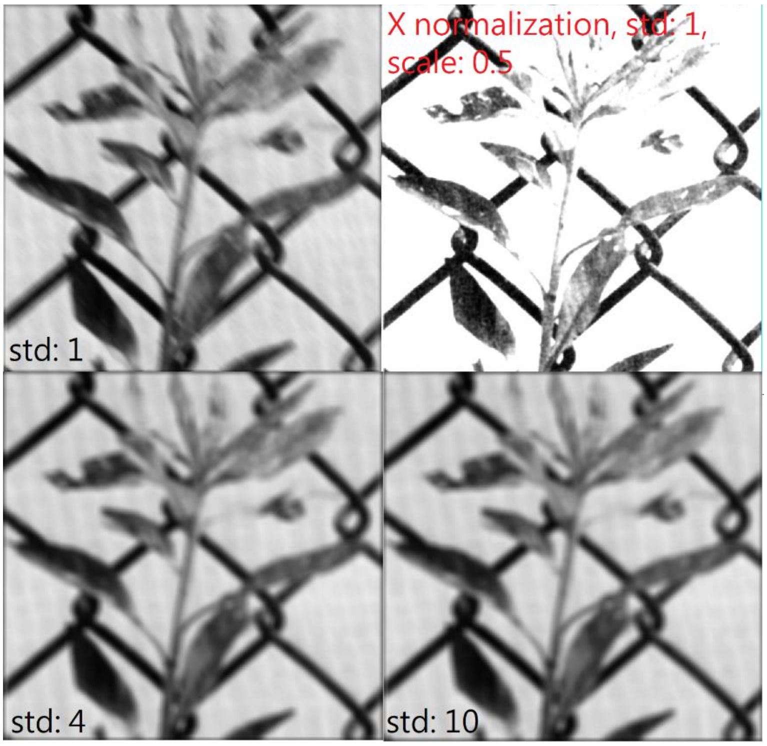
- Discussion
  - Larger kernel size of box filter required more calculation time. As kernel gets larger, the blurring effect becomes more obvious.



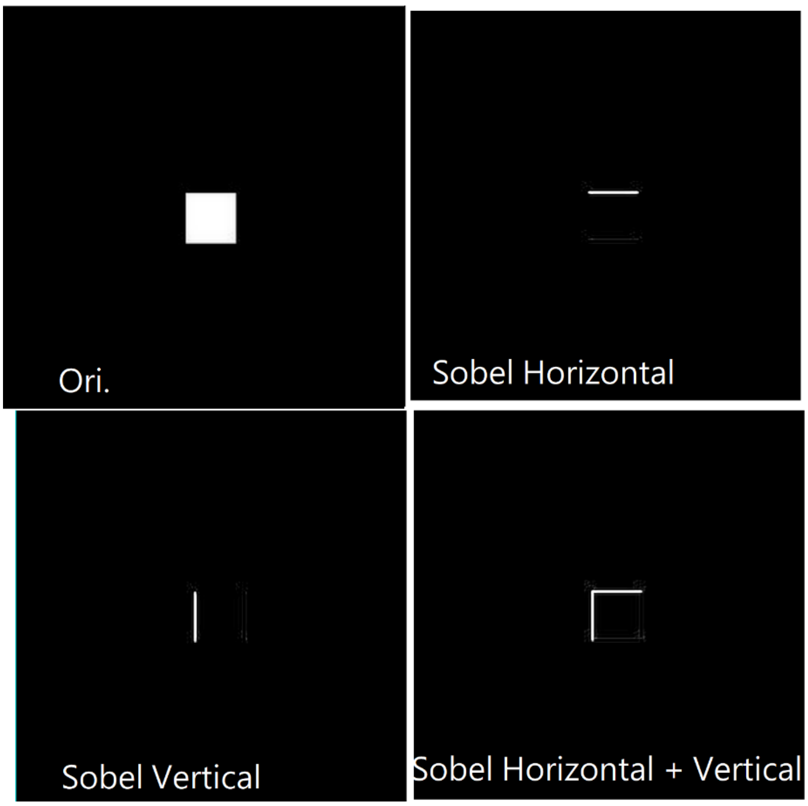

  - For Gaussian filter, under same standard deviation (σ=1), as the kernel size gets larger, the blurring effect becomes more obvious. Meanwhile, there exists a shifting effect compared to box filter; as the kernel size gets larger, the shifting effect becomes more obvious.



- Under same kernel size (15*15), we change the standard deviation of a Gaussian filter from 1 to 10. As σ increase, the shifting effect of Gaussian filter is reduced. For the upper right image, the normalization process is removed, with K=0.5. As we can see the image becomes pale, for most of pixels exceeded the 255 boundary.

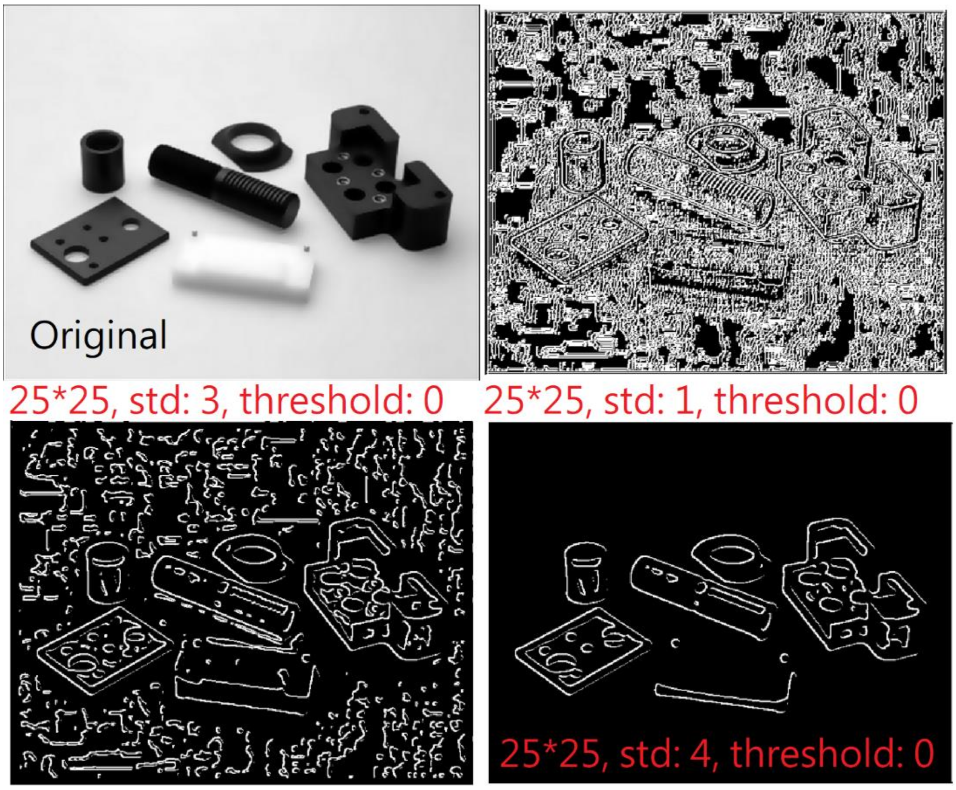



- For Sobel Filter, Sobel horizontal filter can scan where the image turn from black to white, while it cannot detect where the image turn from white to black. The same for Sobel vertical filter. If the negative sign is assigned to the other side of the filter, the effect would be reverse. As a result, in theory, with four Sobel filters, all the edges of an image can be found.

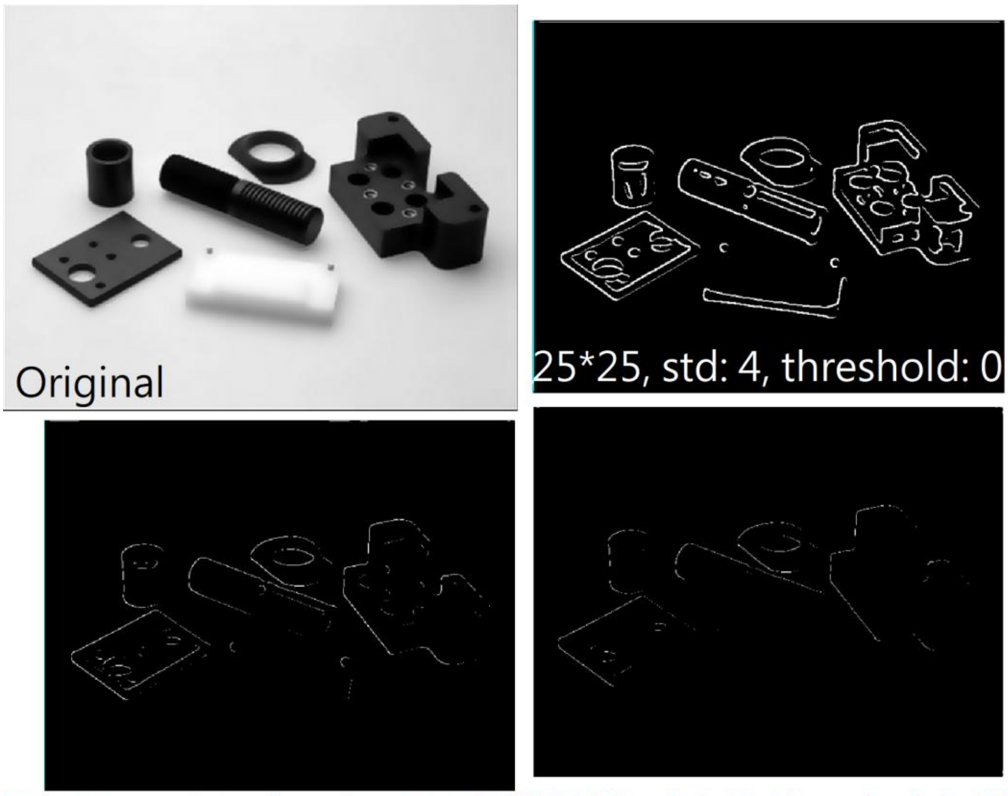


- For Laplacian Filter, the edge of the image is enhanced. Laplacian full not only contain vertical and horizontal, it also contain information about diagonal direction. As a result, the overall enhance effect is stronger than Laplacian half.


- For part3, the result of Sobel is shown above at bottom right, while the result of the Marr-Hildreth edge detection method is shown below. For Sobel, as mentioned above, we will need four Sobel filters to detect all the edges of the object. On the other hand, the MR method cam detect all edges with single filter. In the below figure, we fix the kernel size and the zero-crossing threshold to observe the effect of standard deviation. When σ increases the background noise is reduced, and the edge of object becomes more obvious.

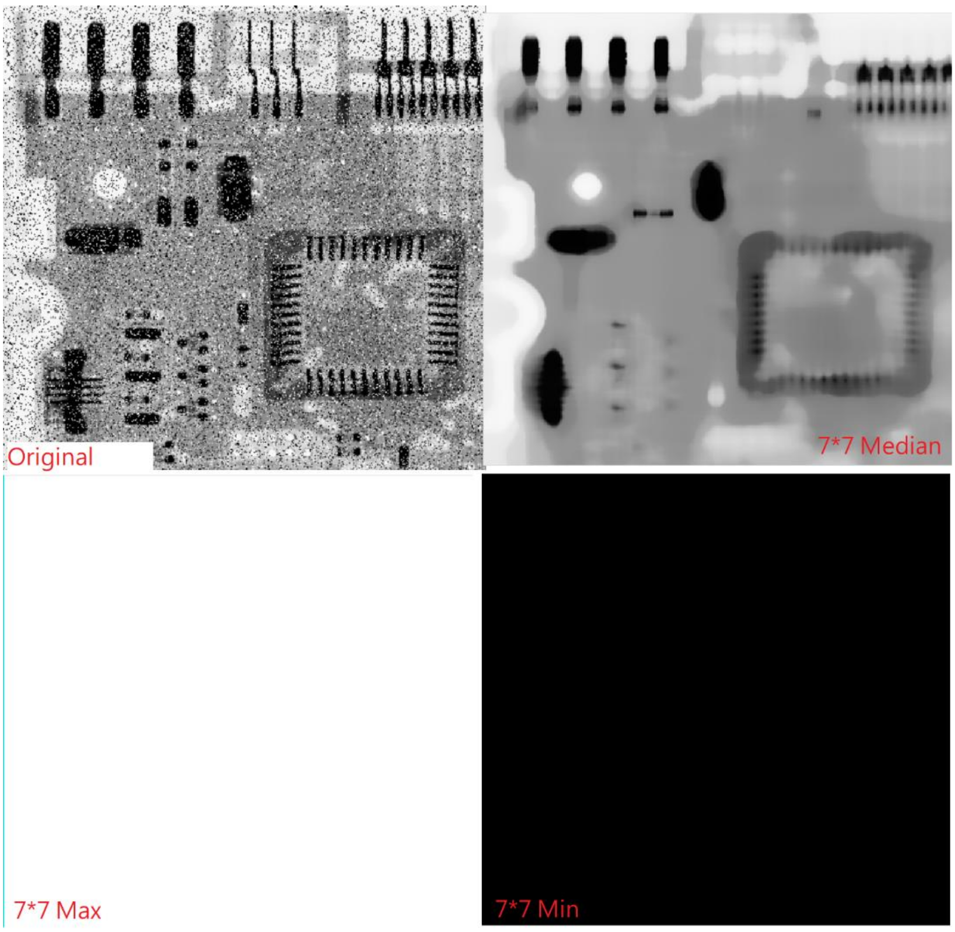


- If we fix the σ and adjust the zero-crossing threshold, the edges are obvious when the threshold is low. As threshold rises, the edges becomes dimmer.




25*25, std: 4, threshold:128 25*25, std: 4, threshold: 255

- With a median filter, the salt-and-pepper noise is filtered. With min and max filter, the noise is picked up by the filter, so the image becomes



◆ The size of the filter seems of no obvious difference.