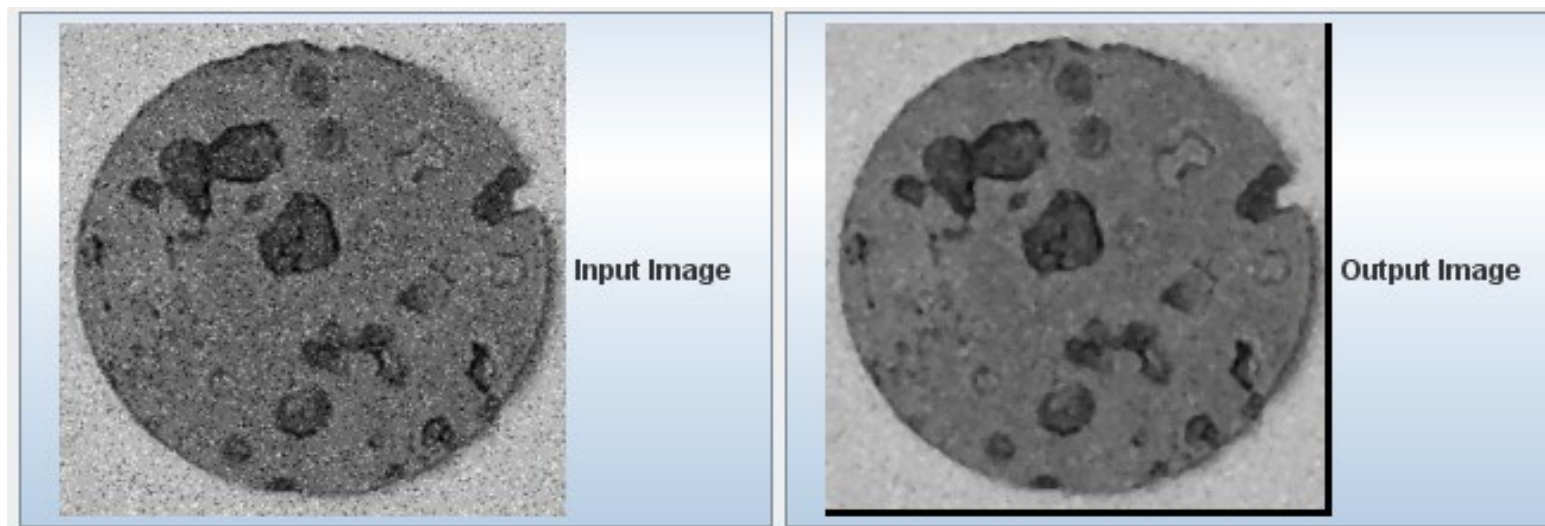
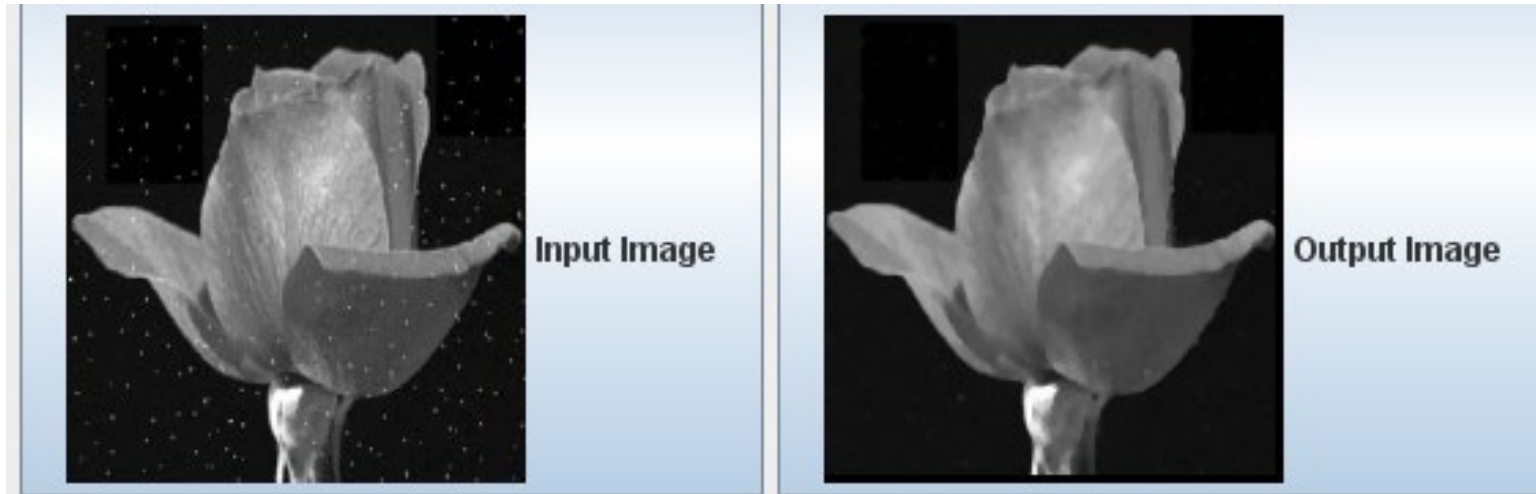
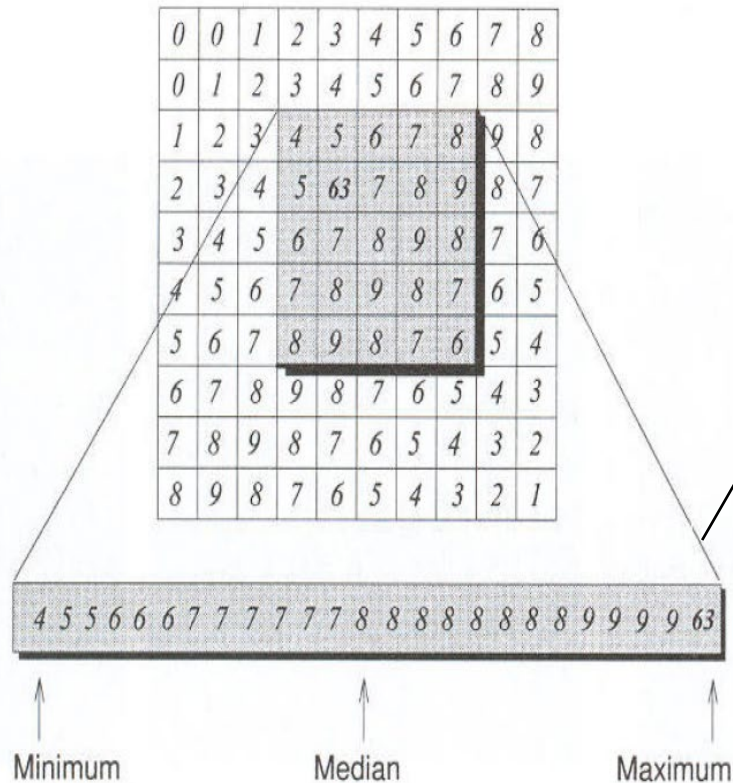


Aufgabe 1

Compute a noise reduction with a morphological median
Medianfilterung.



Hinweis



```
int[] ar=new int[9];
```

```
Arrays.sort(ar);
```

Dann ist:

ar[8] das Maximum

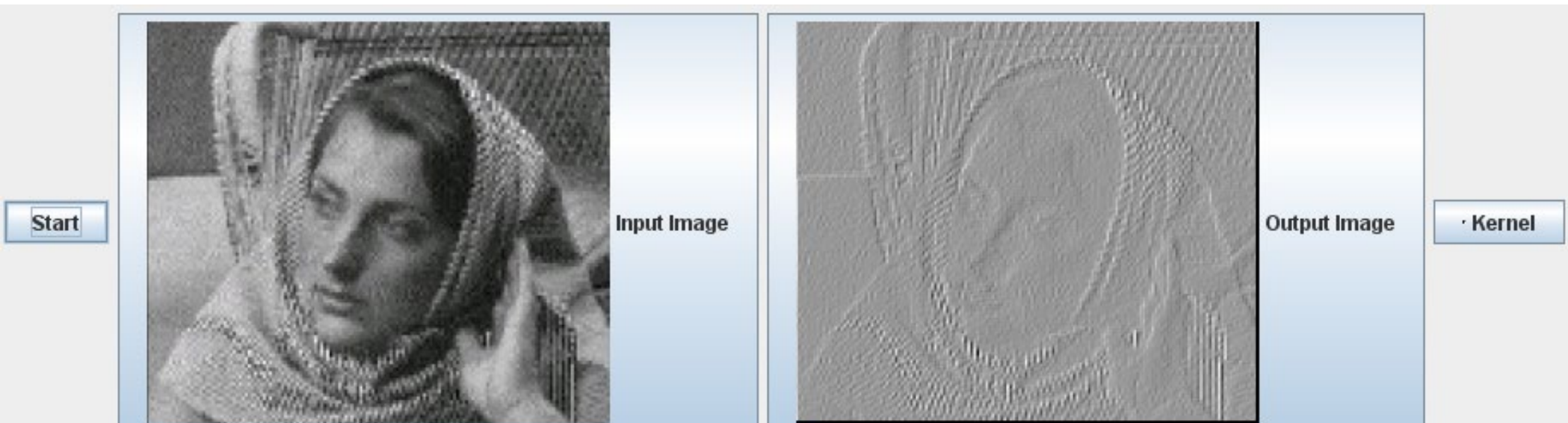
ar[0] das Minimum

ar[4] der Median

Aufgabe 2

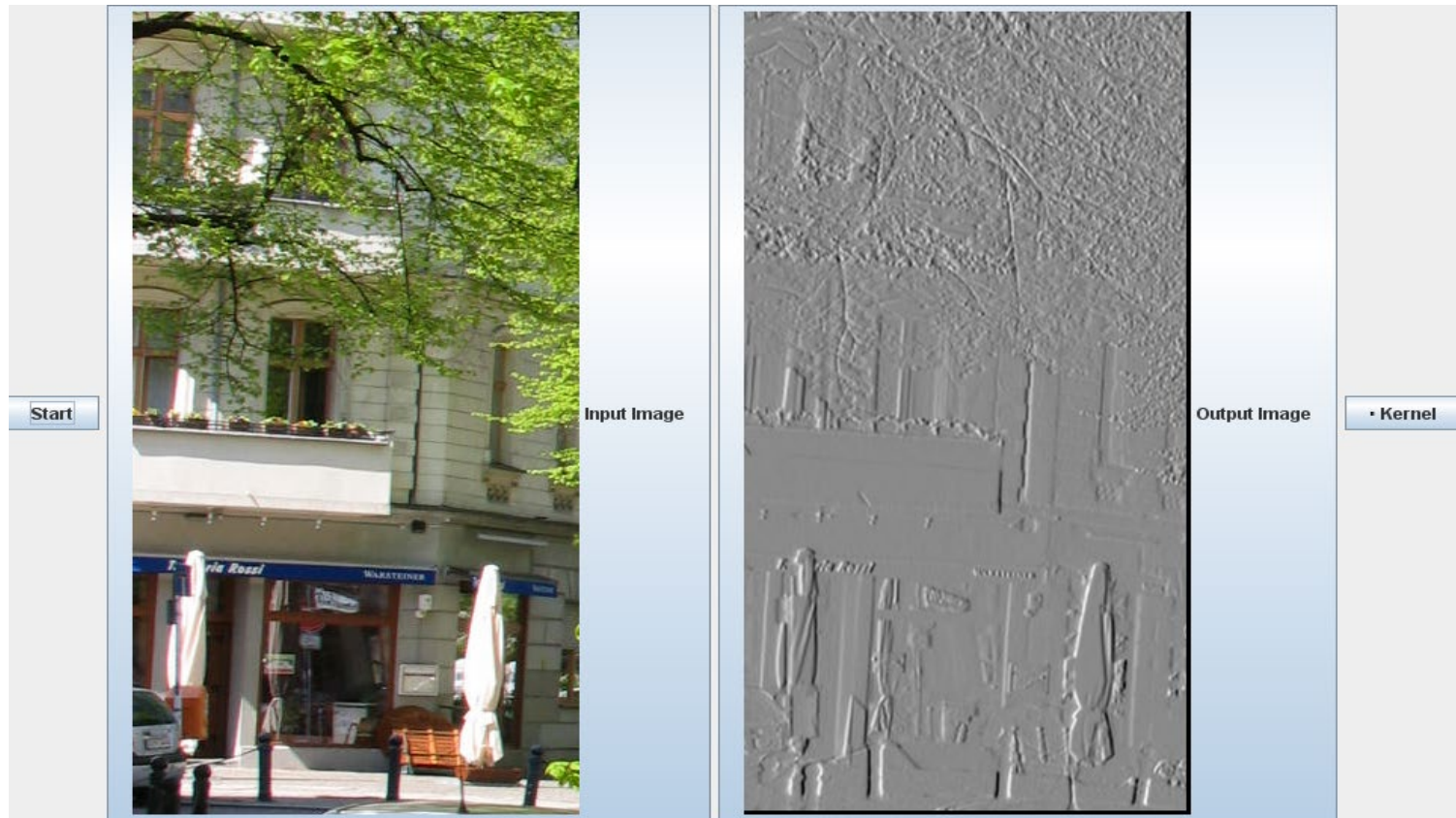
Implement linear filtering (convolution) with different Kernels:
edge detection and smoothing.

a) Detection of fine edges



$$h = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

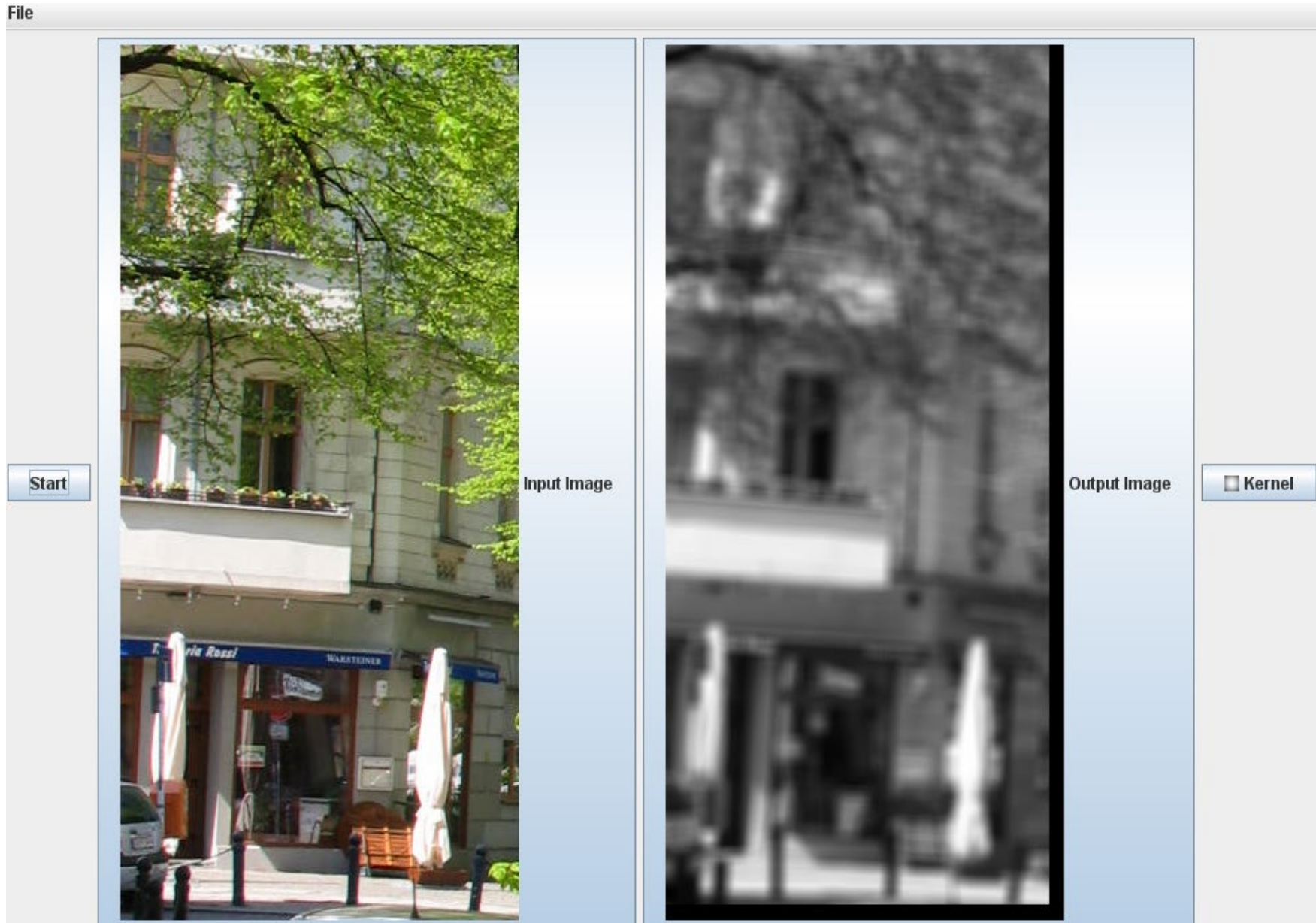
b) Detection of fine vertical edges vertikaler Kanten mit $P=Q=3$



$$h = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

c) Smoothing with a gaussian filter

$\text{dim}(h)=(P,Q)$ $P=Q=11$



Hinweis

Gauss'sche Faltungskerne sollten in die Mitte des von P und Q definierten Rechtecks verschoben werden. Dies kann durch folgende Translation erreicht werden:

$$x' = x - ((P-1)/2)$$

$$y' = y - ((Q-1)/2)$$

Ferner sollte die Gaussfunktion am Rande des Rechtecks schon weitestgehend abgeklungen sein. Dies kann durch folgende Skalierung erreicht werden:

$$x'' = x' / (P/4)$$

$$y'' = y' / (Q/4)$$

Anschließend werden die Gewichte durch die Gaussfunktion definiert:

$$h(x,y) = \exp(- ((x'')^2 + (y'')^2))$$

