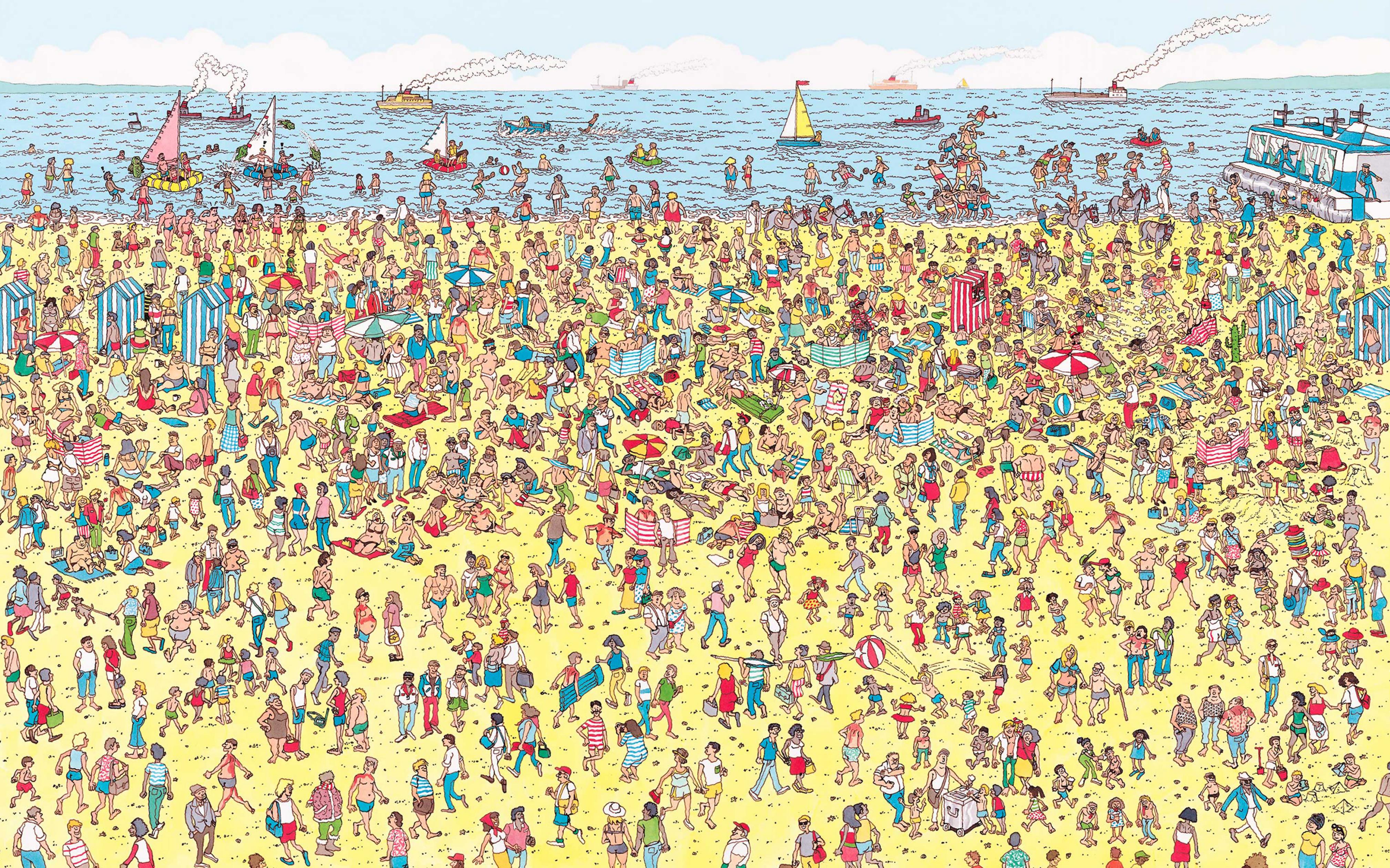
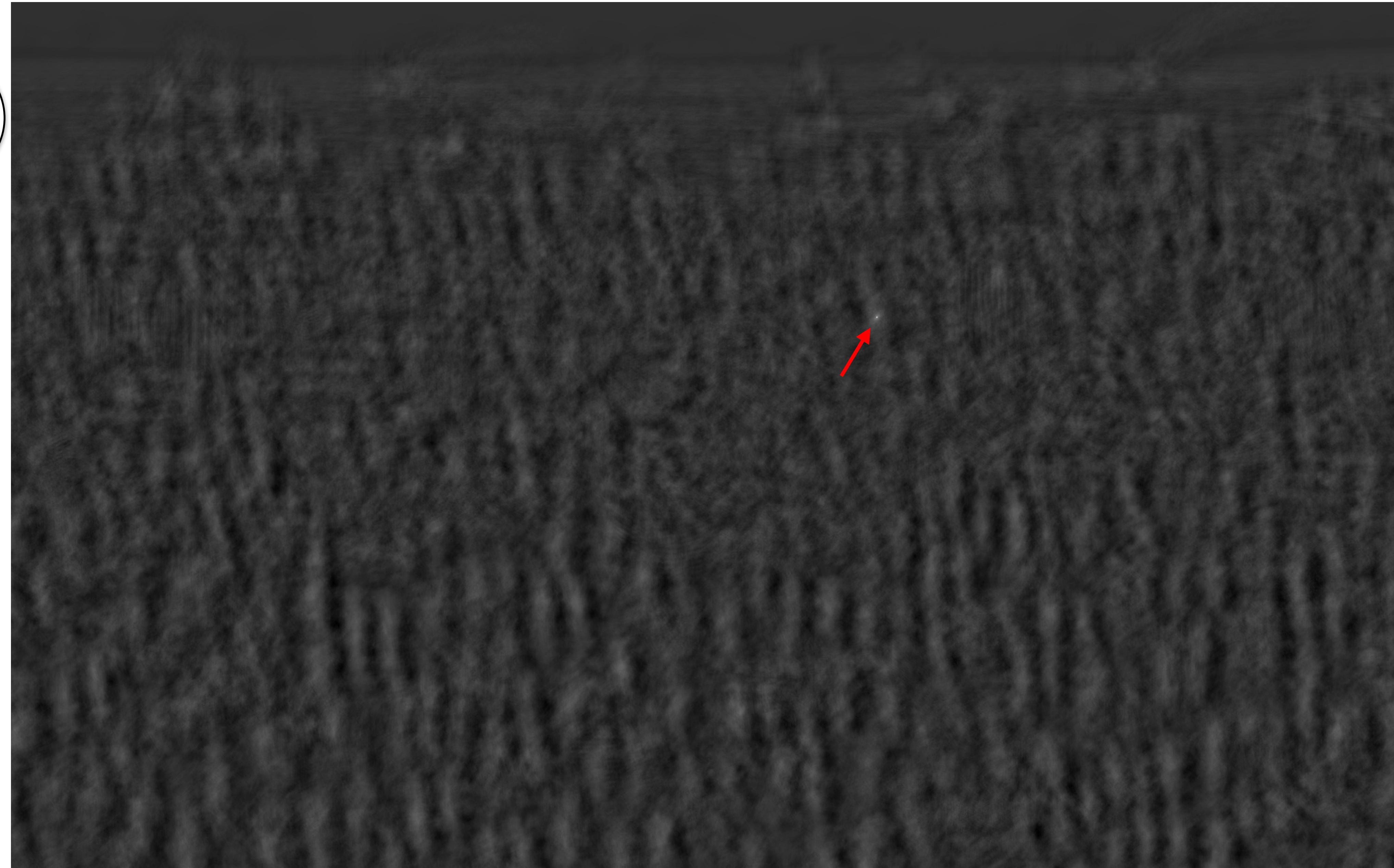
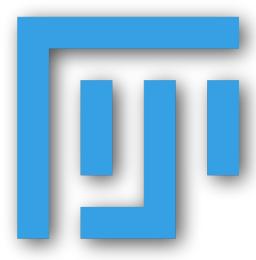


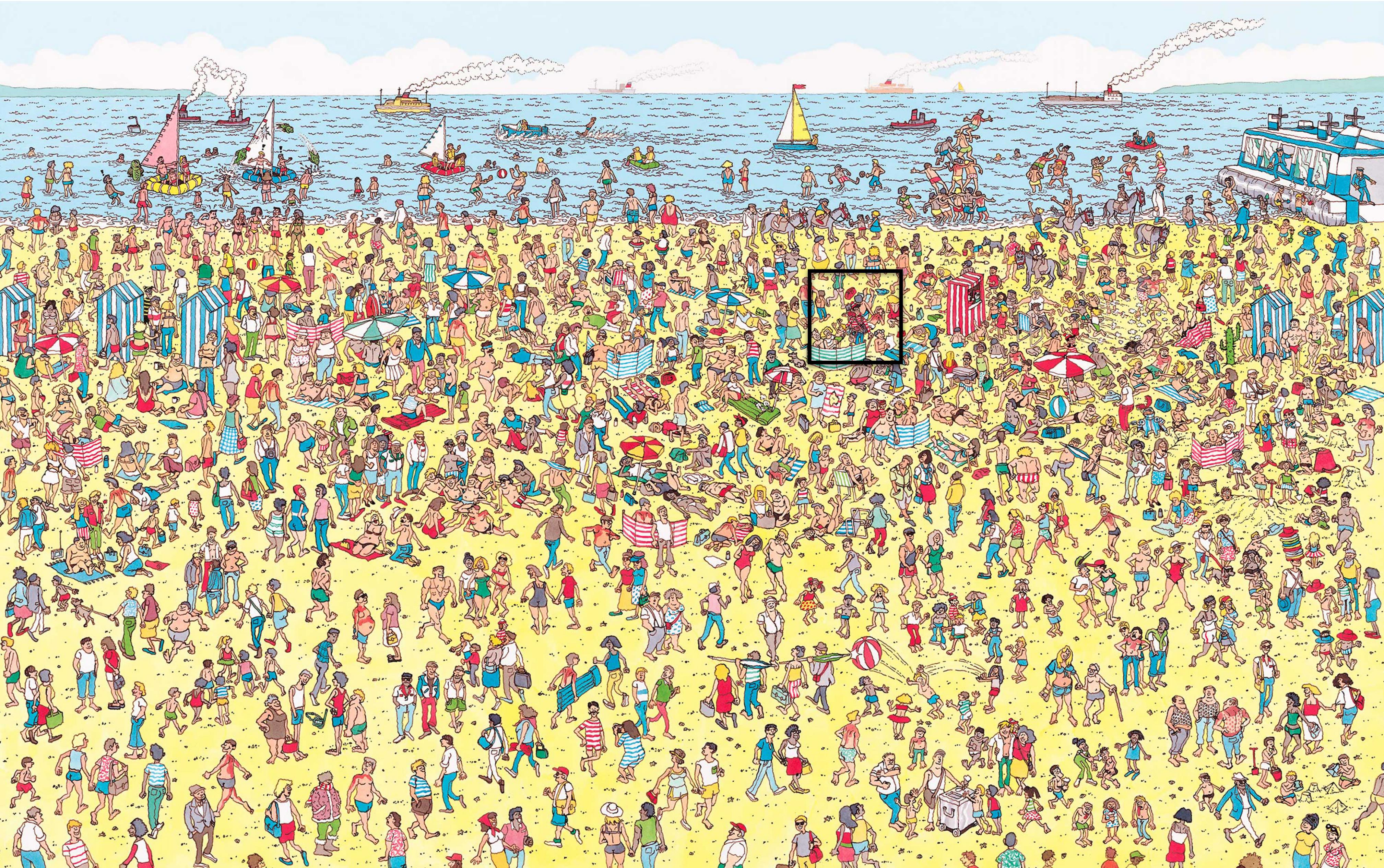
Filters

Where is Waldo?



Where is Waldo?





Where is Waldo?

Filtering

Neighborhood Transformation

Linear Transformation

Smoothing: Mean, Gaussian

Edge detection: Sobel, Canny

Non-linear Transformation

Smoothing: Median

Cleaning: Morphology

Weights are defined by the kernels

Weighted average = Multiply, add, divide

One neighborhood at a time

Neighborhood defined by the kernel size

Convolution is weighted average in local neighborhood

Terminologies

A **kernel** is a small matrix used in image processing for convolution.

A **filter** is a broader term that refers to any technique that modifies an image.

Convolution operation

Image

px1	px2	px3	px4	px5	px6	px7	px8	px9
px10	px11	px12	px13	px14	px15	px16	px17	px18
px19	px20	px21	px22	px23	px24	px25	px26	px27
px28	px29	px30	px31	px32	px33	px34	px35	px36
px37	px38	px39	px40	px41	px42	px43	px44	px45
px46	px47	px48	px49	px50	px51	px52	px53	px54
px55	px56	px57	px58	px59	px60	px61	px62	px63
px64	px65	px66	px67	px68	px69	px70	px71	px72
px73	px74	px75	px76	px77	px78	px79	px80	px81

convolution
★

k1	k2	k3
k4	k5	k6
k7	k8	k9

$$\text{ksum} = k_1 + k_2 + k_3 + k_4 + k_5 + k_6 + k_7 + k_8 + k_9$$

3 x 3 kernel

$$\begin{aligned}
 & (\text{px1} * \text{k1} / \text{ksum}) + \\
 & (\text{px2} * \text{k2} / \text{ksum}) + \\
 & (\text{px3} * \text{k3} / \text{ksum}) + \\
 & (\text{px10} * \text{k4} / \text{ksum}) + \\
 & (\text{px11} * \text{k5} / \text{ksum}) + \\
 & (\text{px12} * \text{k6} / \text{ksum}) + \\
 & (\text{px19} * \text{k7} / \text{ksum}) + \\
 & (\text{px20} * \text{k8} / \text{ksum}) + \\
 & (\text{px21} * \text{k9} / \text{ksum}) =
 \end{aligned}$$

?

c1

Convolution operation

Image

px1	px2	px3	px4	px5	px6	px7	px8	px9
px10	px11	px12	px13	px14	px15	px16	px17	px18
px19	px20	px21	px22	px23	px24	px25	px26	px27
px28	px29	px30	px31	px32	px33	px34	px35	px36
px37	px38	px39	px40	px41	px42	px43	px44	px45
px46	px47	px48	px49	px50	px51	px52	px53	px54
px55	px56	px57	px58	px59	px60	px61	px62	px63
px64	px65	px66	px67	px68	px69	px70	px71	px72
px73	px74	px75	px76	px77	px78	px79	px80	px81

convolution
★

k1	k2	k3
k4	k5	k6
k7	k8	k9

3 x 3 kernel

$$\begin{aligned}
 & (\text{px2} * \text{k1} / \text{ksum}) + \\
 & (\text{px3} * \text{k2} / \text{ksum}) + \\
 & (\text{px4} * \text{k3} / \text{ksum}) + \\
 & (\text{px11} * \text{k4} / \text{ksum}) + \\
 & (\text{px12} * \text{k5} / \text{ksum}) + \\
 & (\text{px13} * \text{k6} / \text{ksum}) + \\
 & (\text{px20} * \text{k7} / \text{ksum}) + \\
 & (\text{px21} * \text{k8} / \text{ksum}) + \\
 & (\text{px22} * \text{k9} / \text{ksum}) = \\
 & \text{ksum} = \text{k1} + \text{k2} + \text{k3} + \text{k4} + \text{k5} + \text{k6} + \text{k7} + \text{k8} + \text{k9}
 \end{aligned}$$

=

	c1	?						

c2

Convolution operation

Image

px1	px2	px3	px4	px5	px6	px7	px8	px9
px10	px11	px12	px13	px14	px15	px16	px17	px18
px19	px20	px21	px22	px23	px24	px25	px26	px27
px28	px29	px30	px31	px32	px33	px34	px35	px36
px37	px38	px39	px40	px41	px42	px43	px44	px45
px46	px47	px48	px49	px50	px51	px52	px53	px54
px55	px56	px57	px58	px59	px60	px61	px62	px63
px64	px65	px66	px67	px68	px69	px70	px71	px72
px73	px74	px75	px76	px77	px78	px79	px80	px81

convolution
★

3 x 3 kernel

k1	k2	k3
k4	k5	k6
k7	k8	k9

$$\text{ksum} = k_1 + k_2 + k_3 + k_4 + k_5 + k_6 + k_7 + k_8 + k_9$$

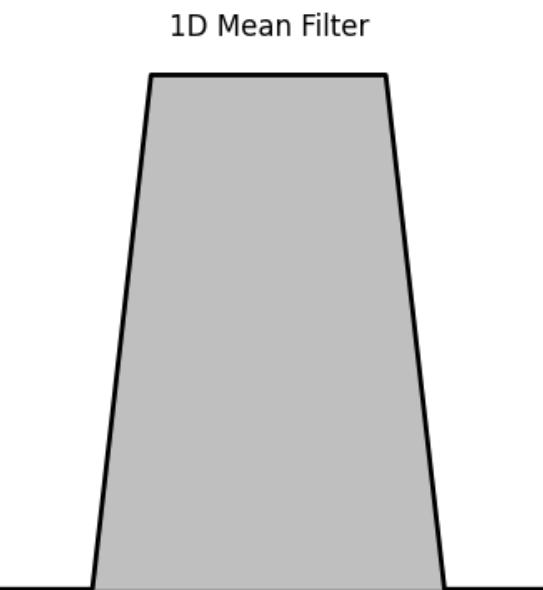
$$\begin{aligned}
 & (\text{px}_3 * \text{k}_1 / \text{ksum}) + \\
 & (\text{px}_4 * \text{k}_2 / \text{ksum}) + \\
 & (\text{px}_5 * \text{k}_3 / \text{ksum}) + \\
 & (\text{px}_{12} * \text{k}_4 / \text{ksum}) + \\
 & (\text{px}_{13} * \text{k}_5 / \text{ksum}) + \\
 & (\text{px}_{14} * \text{k}_6 / \text{ksum}) + \\
 & (\text{px}_{21} * \text{k}_7 / \text{ksum}) + \\
 & (\text{px}_{22} * \text{k}_8 / \text{ksum}) + \\
 & (\text{px}_{23} * \text{k}_9 / \text{ksum}) =
 \end{aligned}$$

c₃

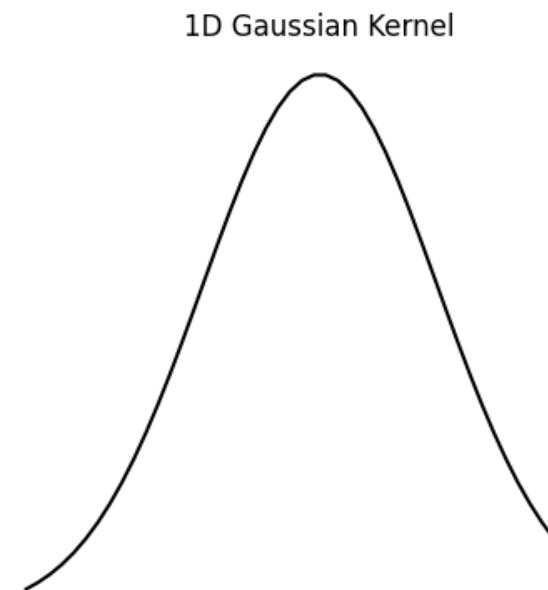
	c ₁	c ₂	?					

Different Types of Kernel

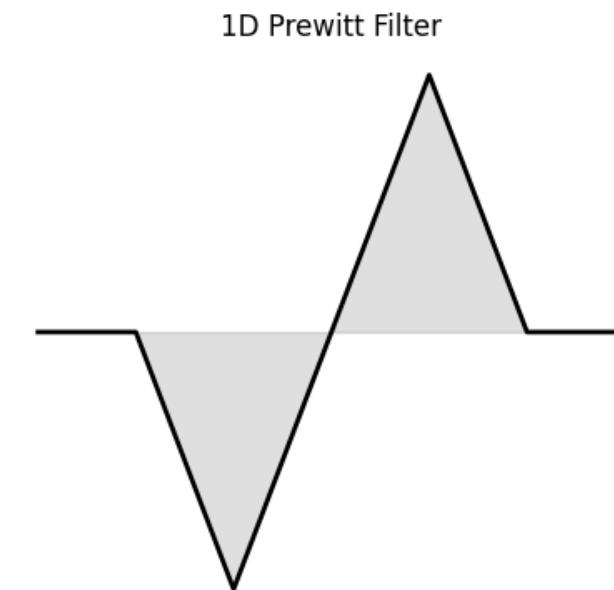
0	0	0	0	0	0
0	1	1	1	1	0
0	1	1	1	1	0
0	1	1	1	1	0
0	0	0	0	0	0



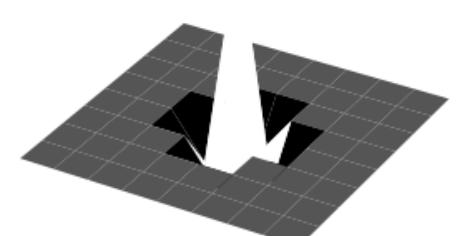
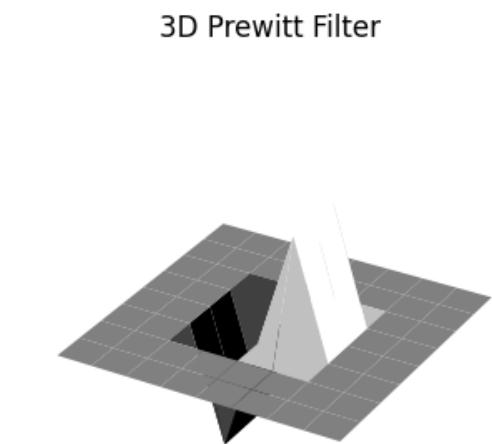
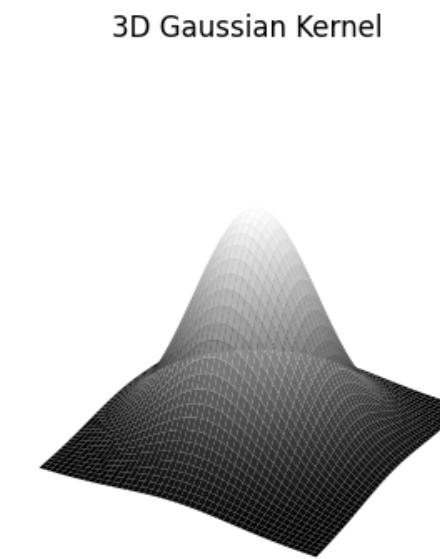
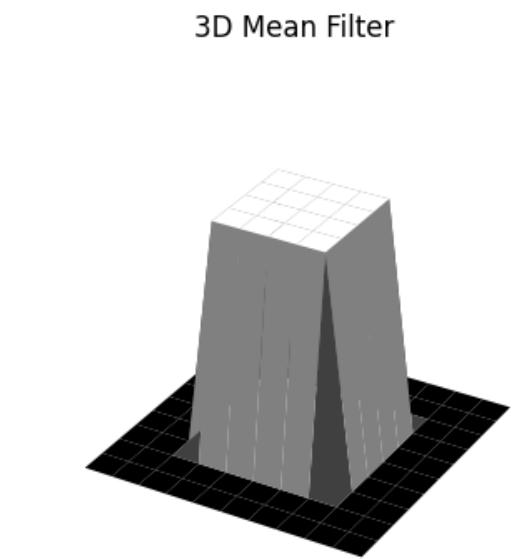
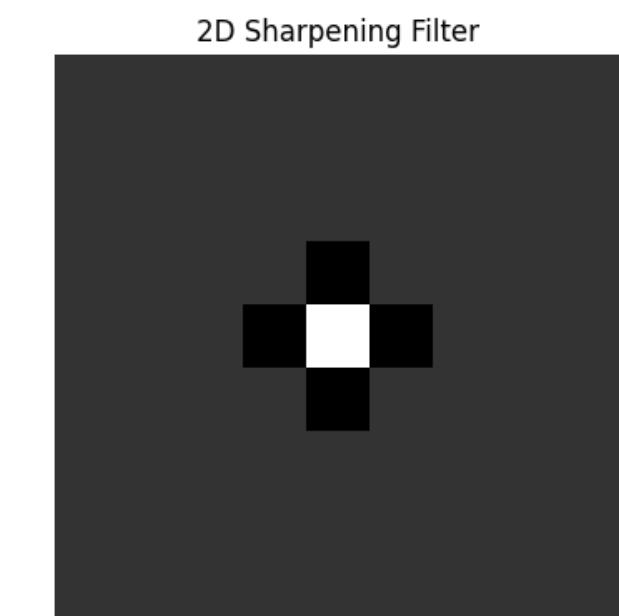
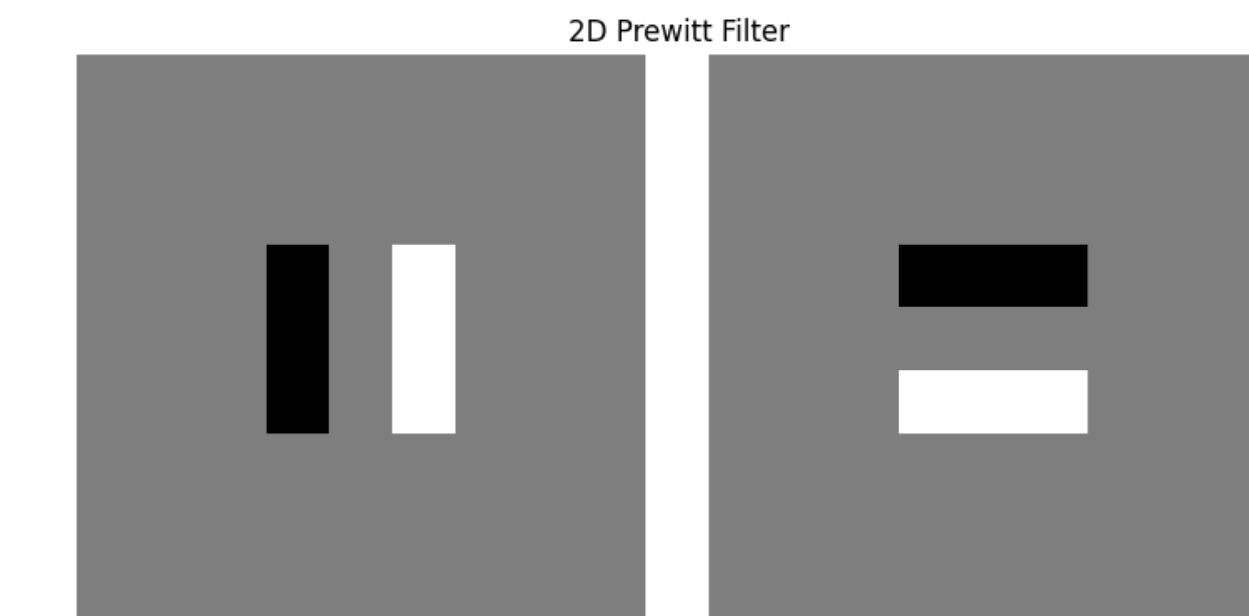
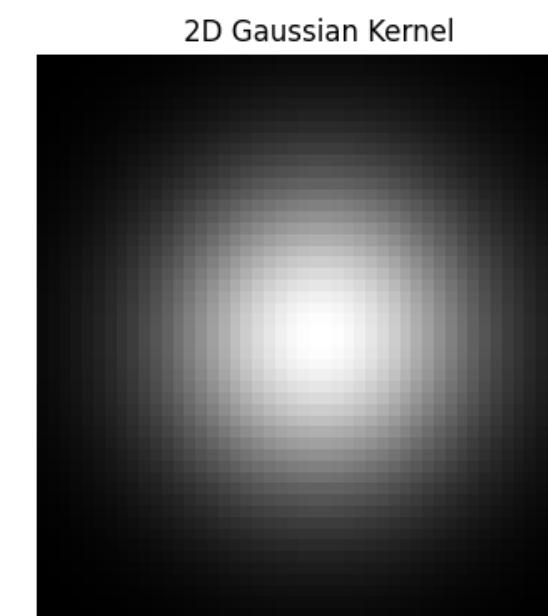
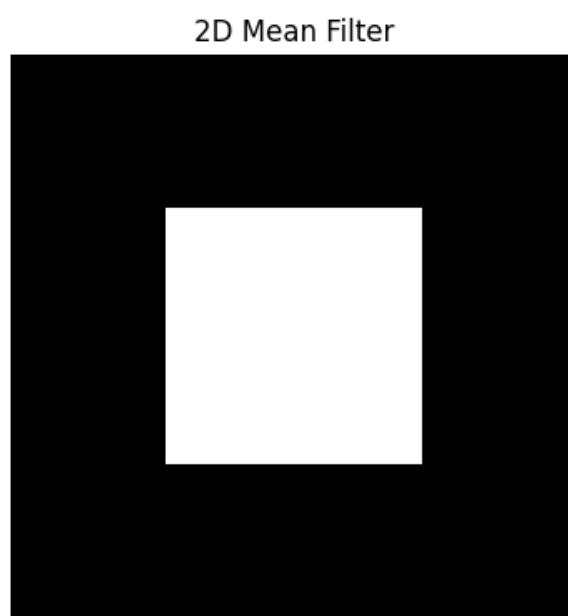
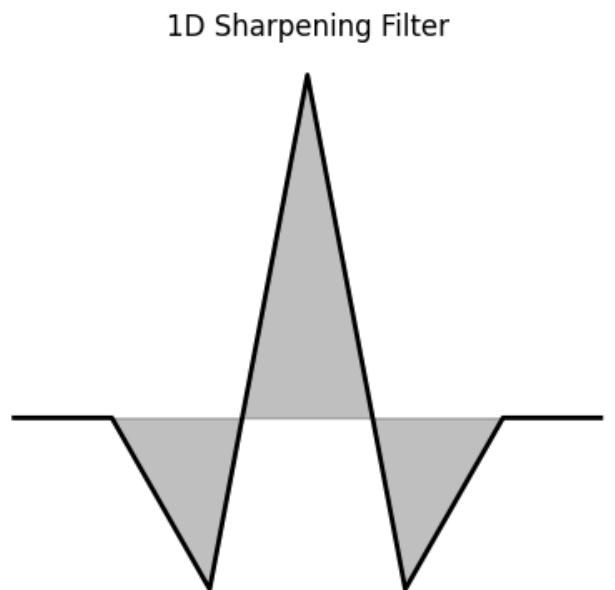
0	0	0	0	0	0
0	1	2	1	0	0
0	2	5	2	0	0
0	1	2	1	0	0
0	0	0	0	0	0



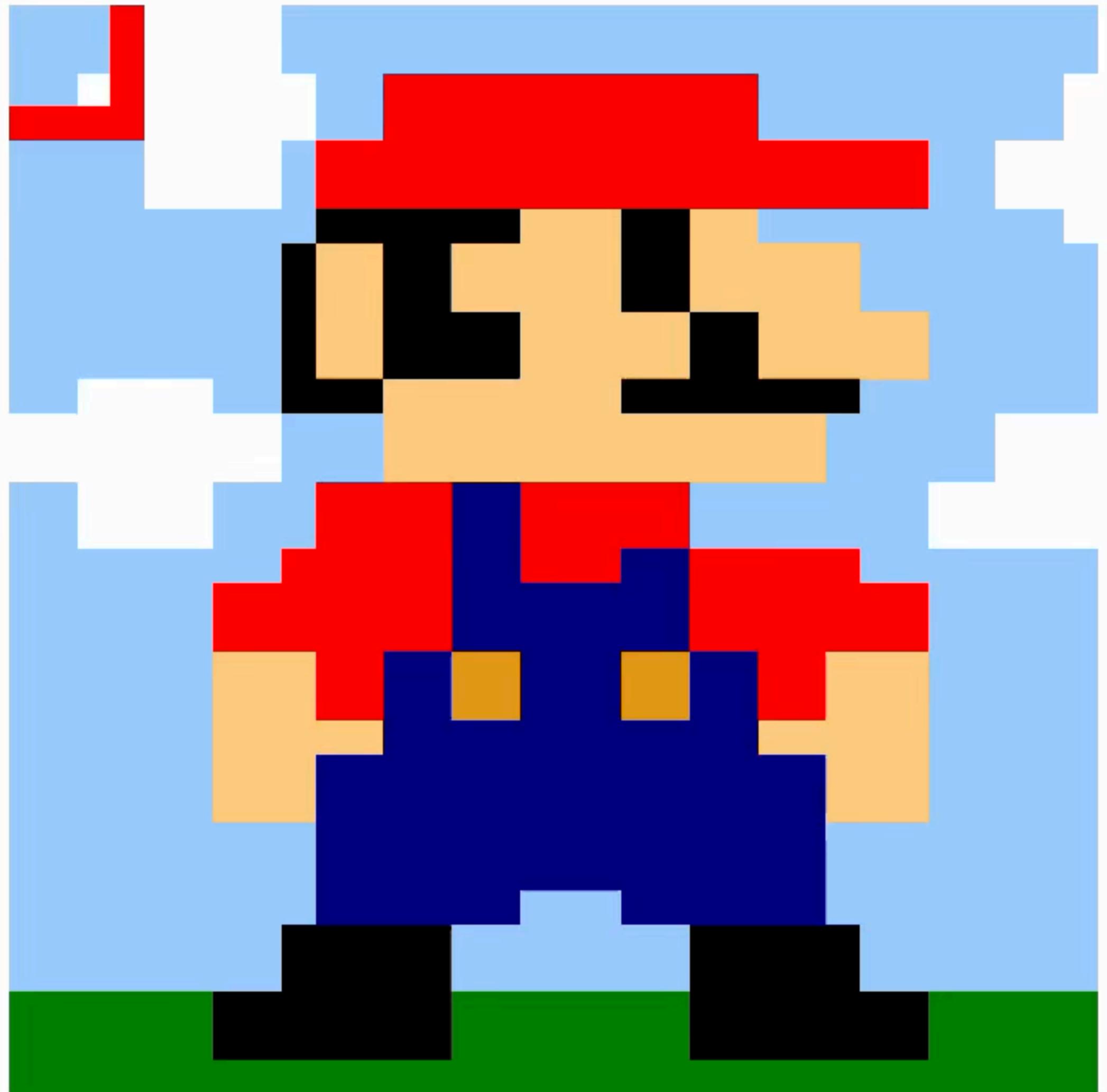
0	0	0	0	0	0
0	-1	0	1	0	0
0	-1	0	1	0	0
0	-1	0	1	0	0
0	1	1	1	0	0



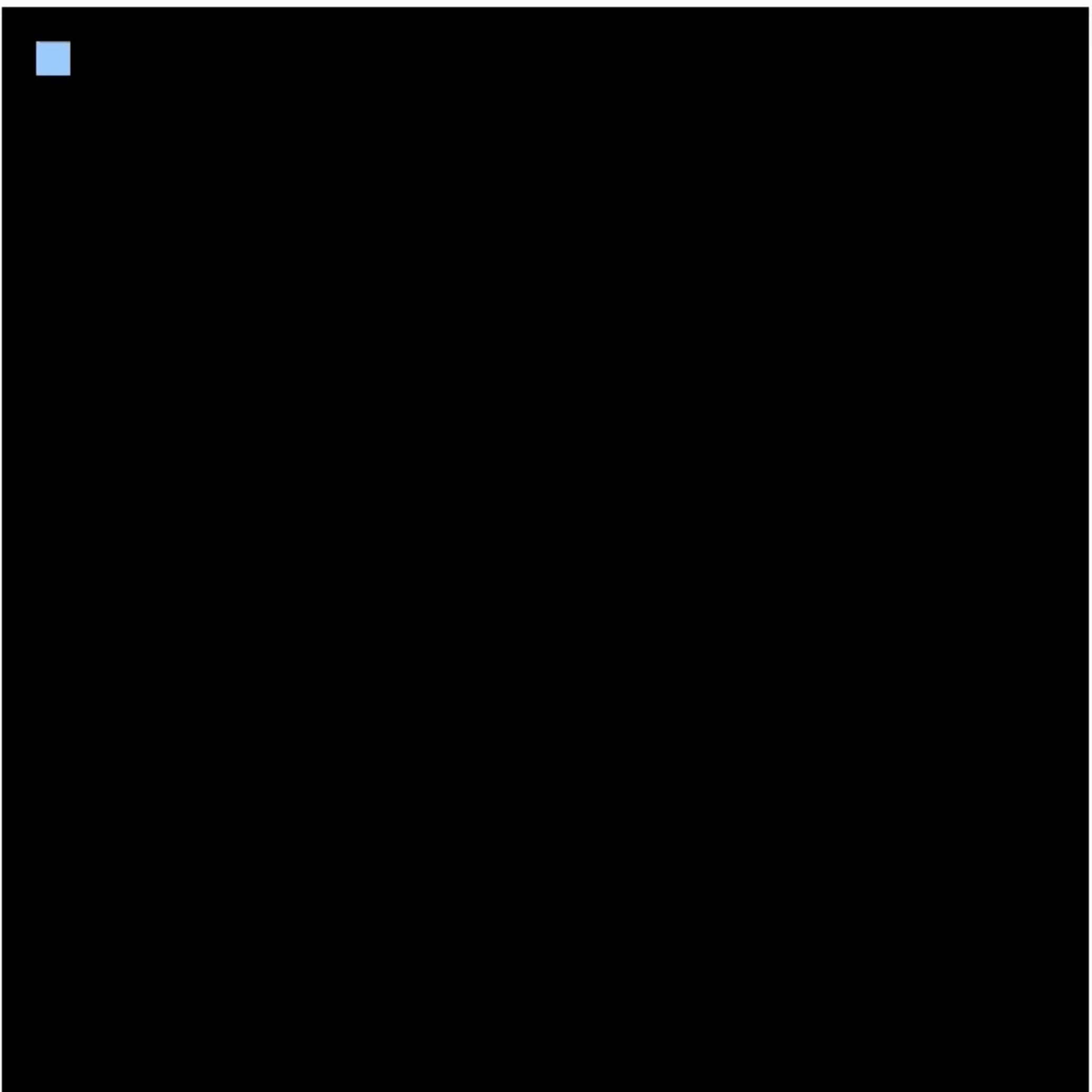
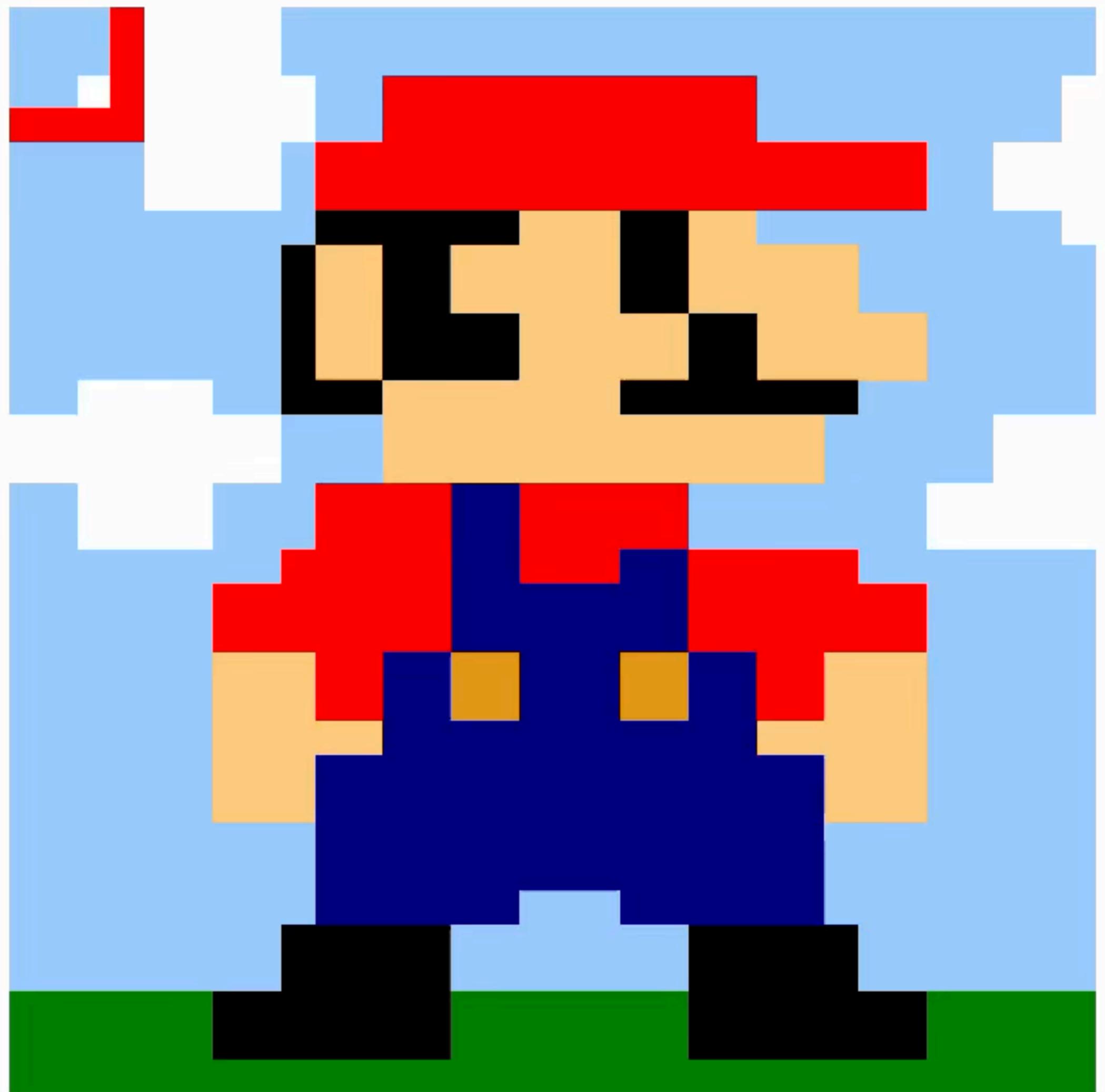
0	0	0	0	0	0
0	1	-1	1	0	0
0	-1	2	-1	0	0
0	1	-1	1	0	0
0	0	0	0	0	0



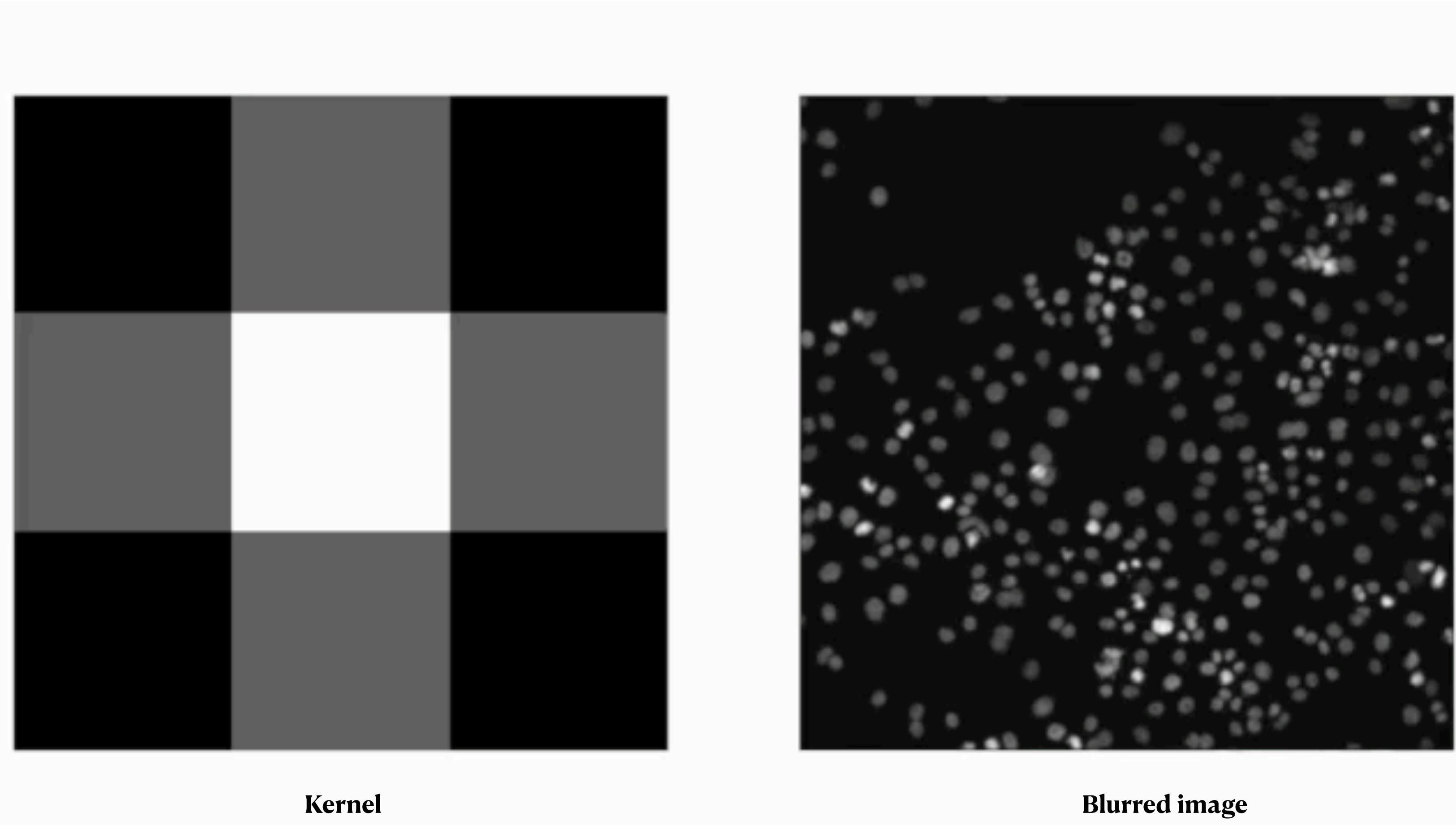
Mean Filter



Gaussian Filter

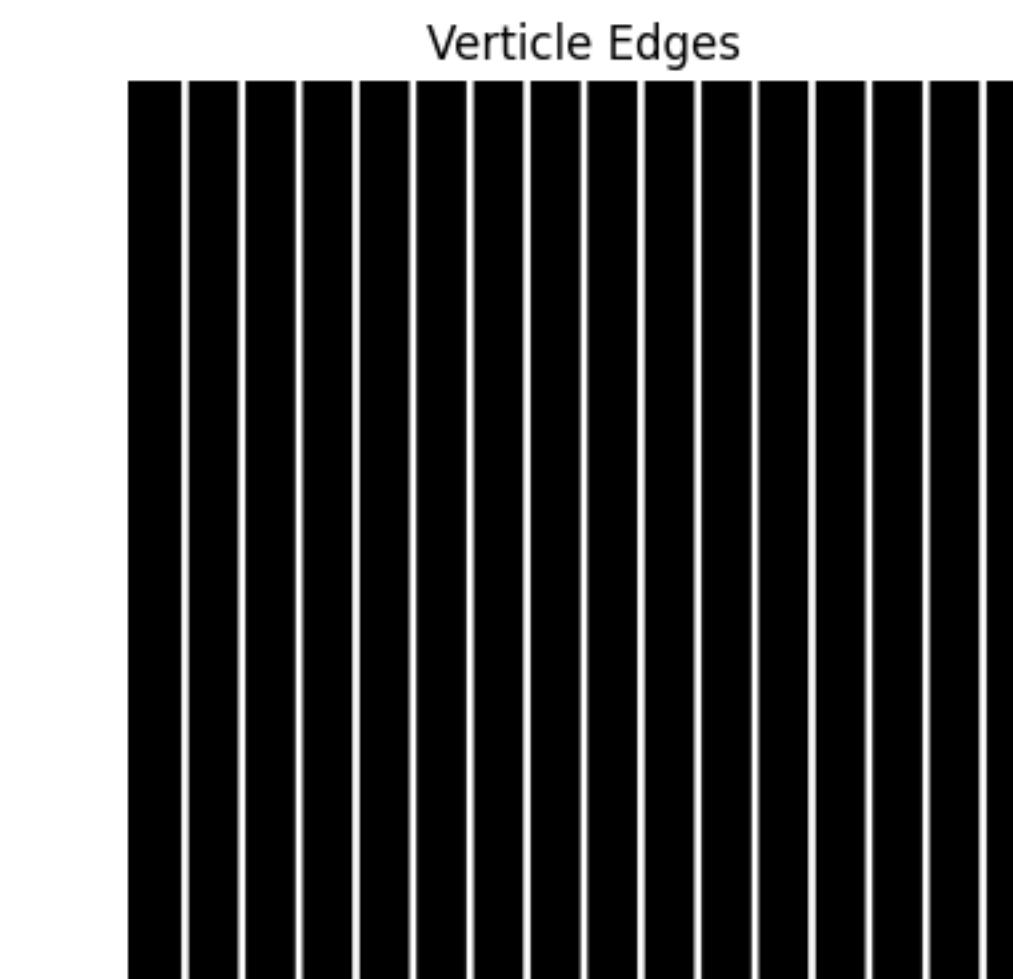
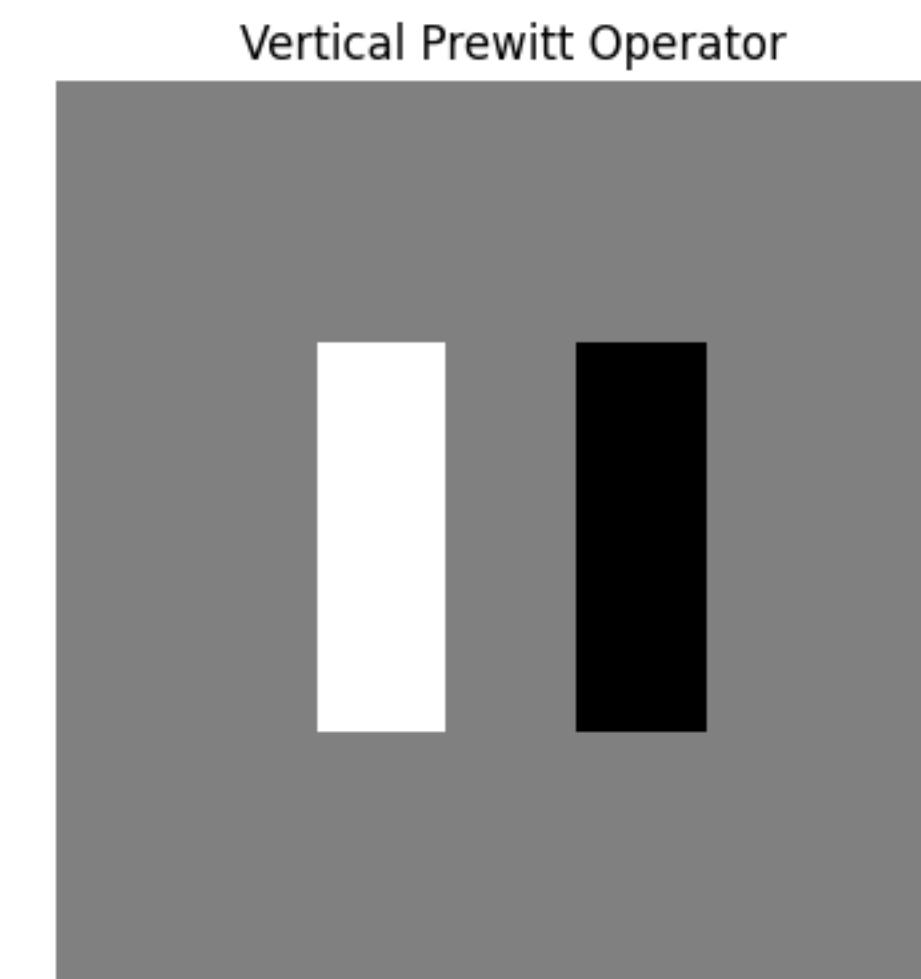
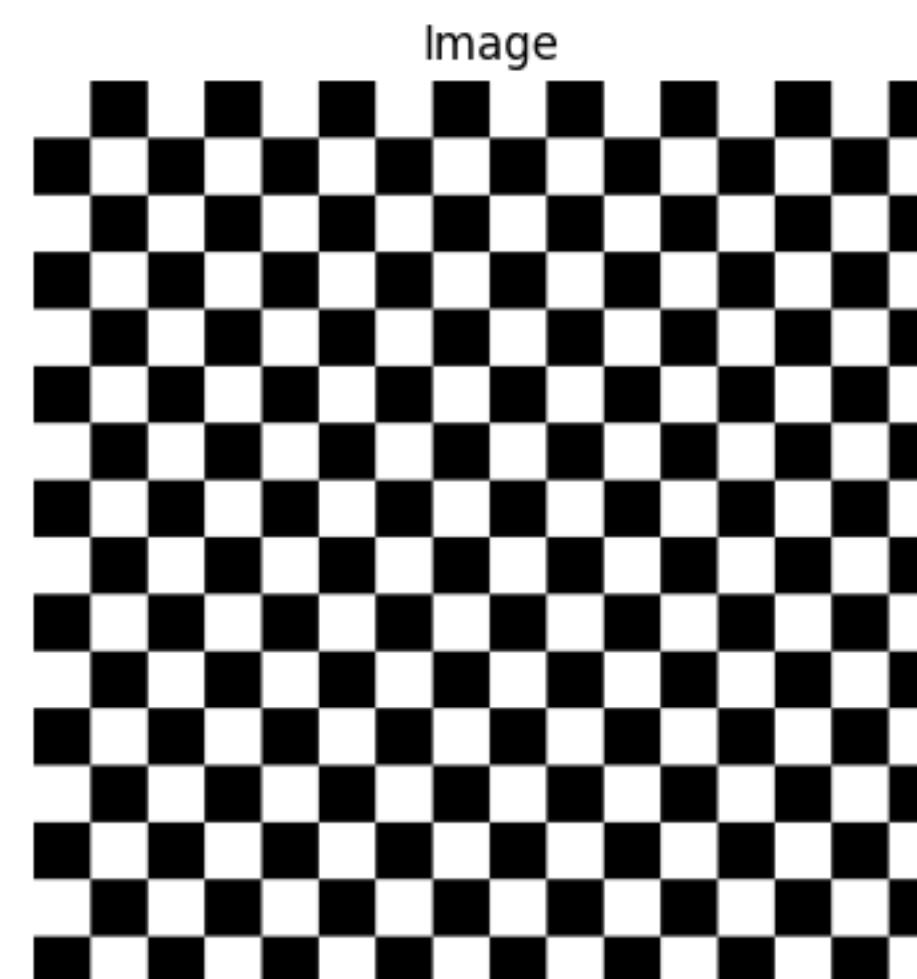
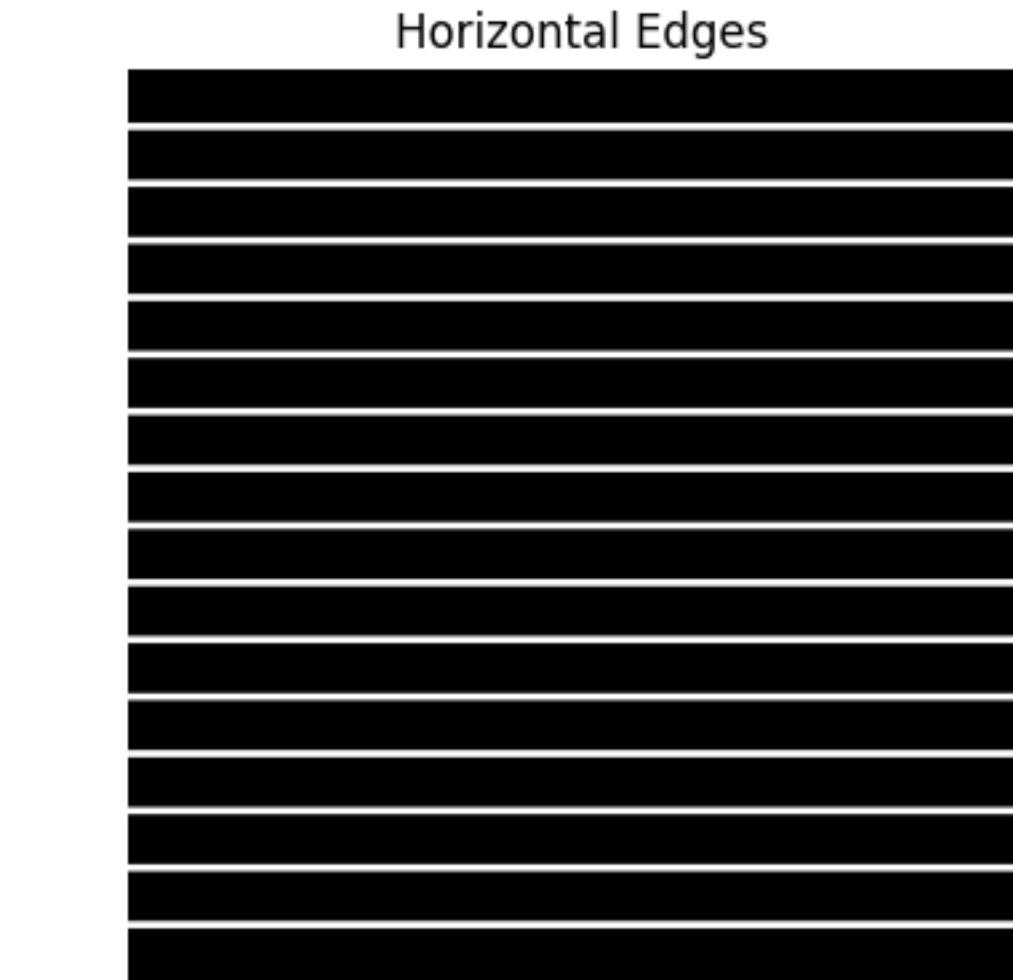
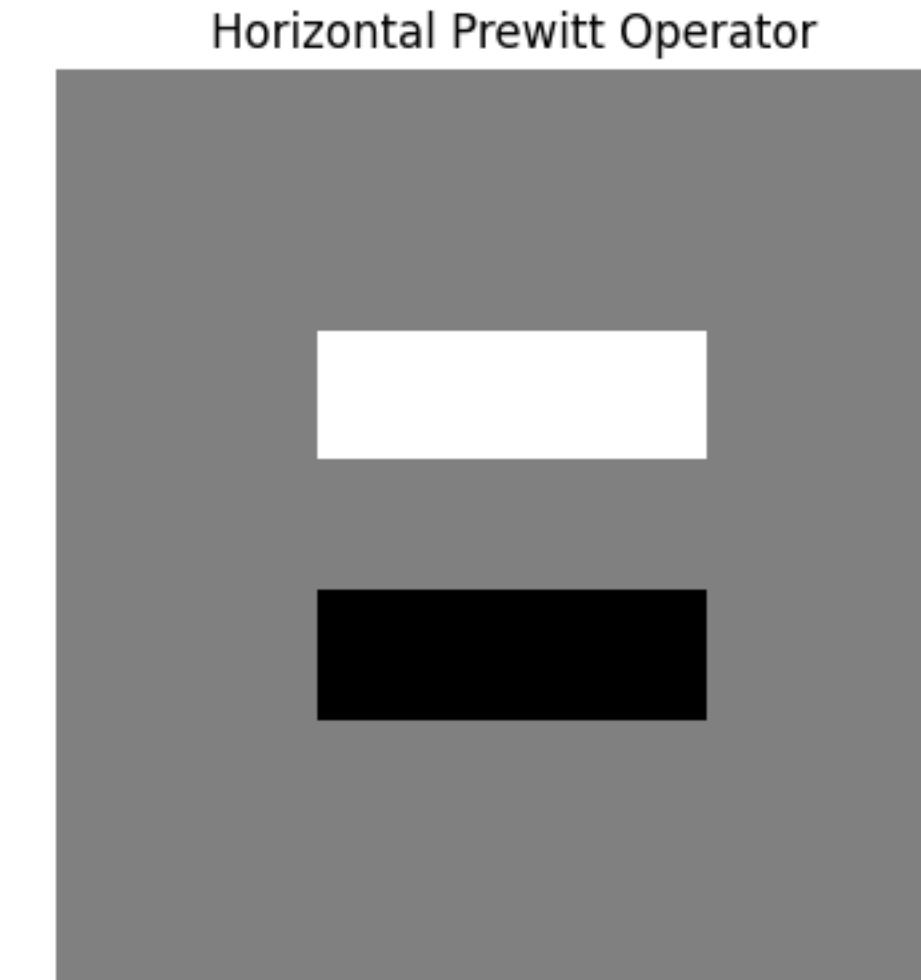
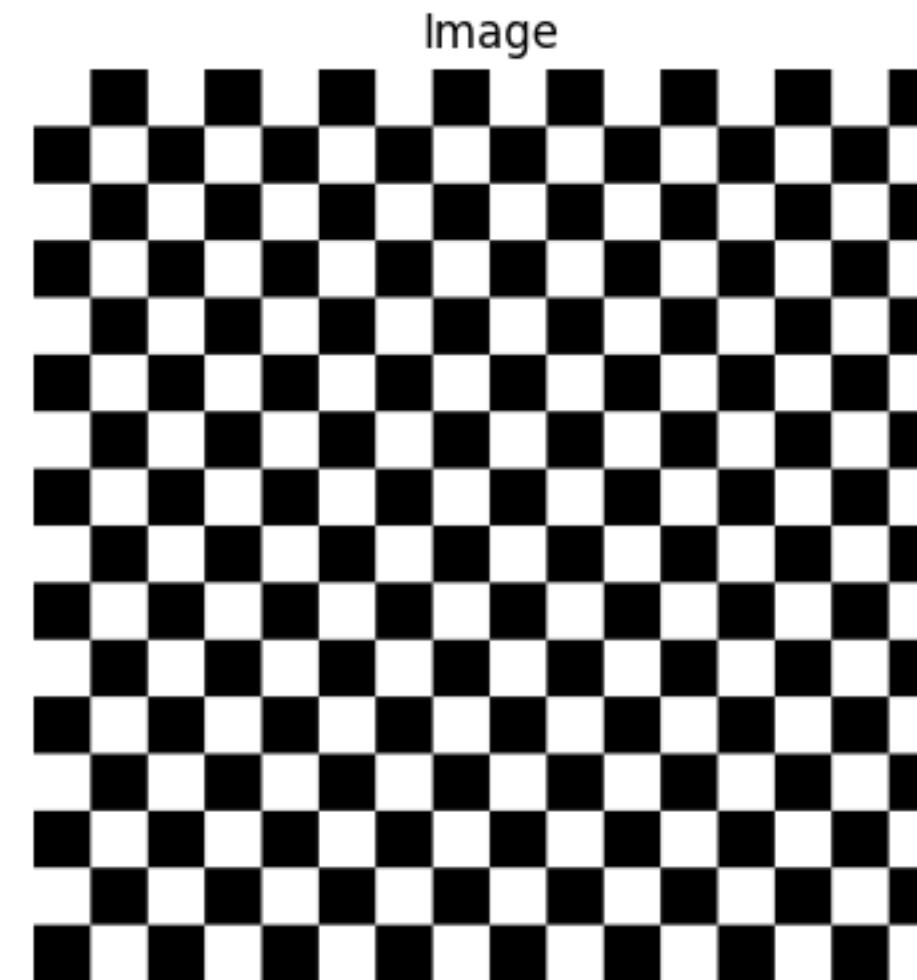


Effects of Different Gaussian Kernel

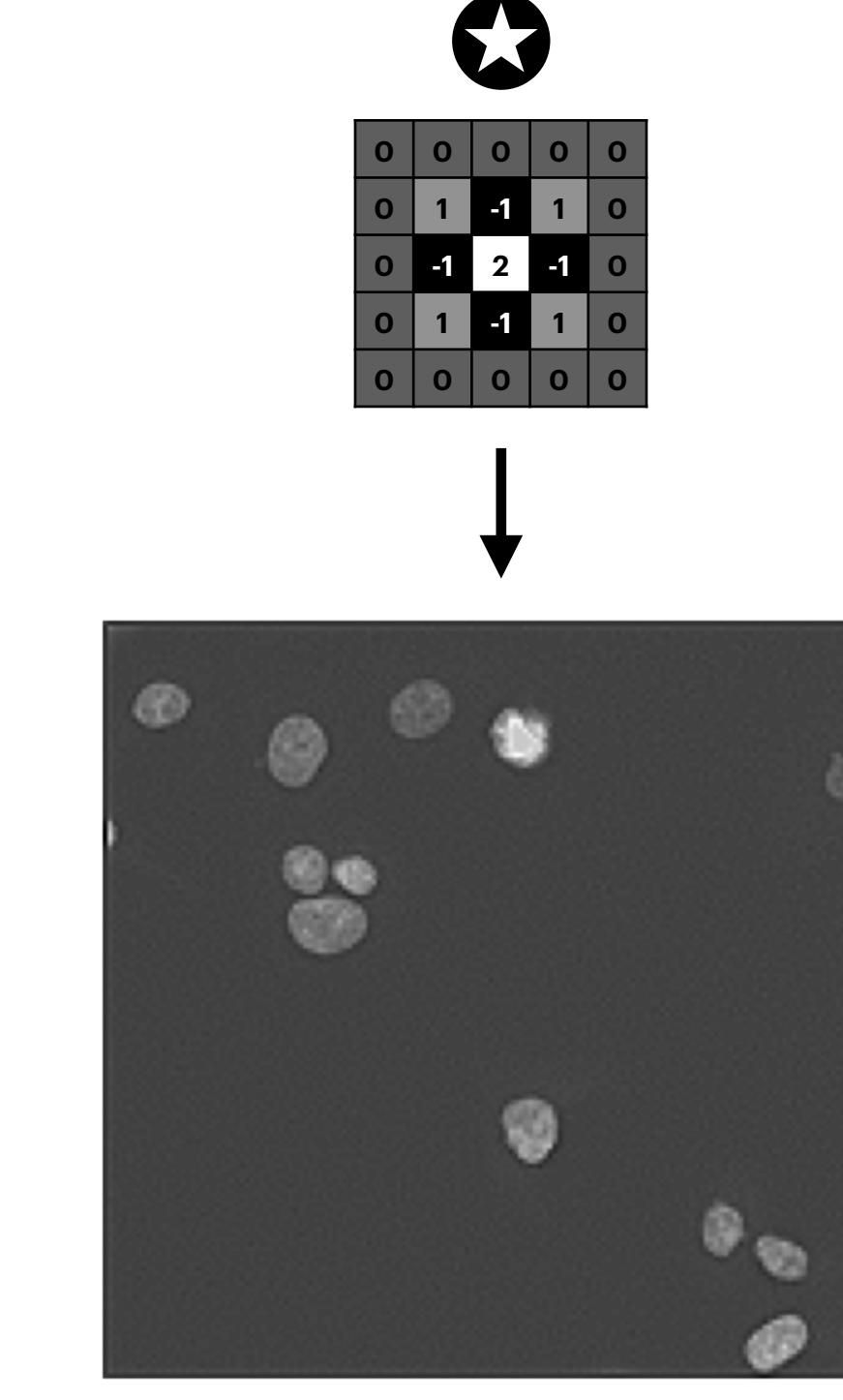
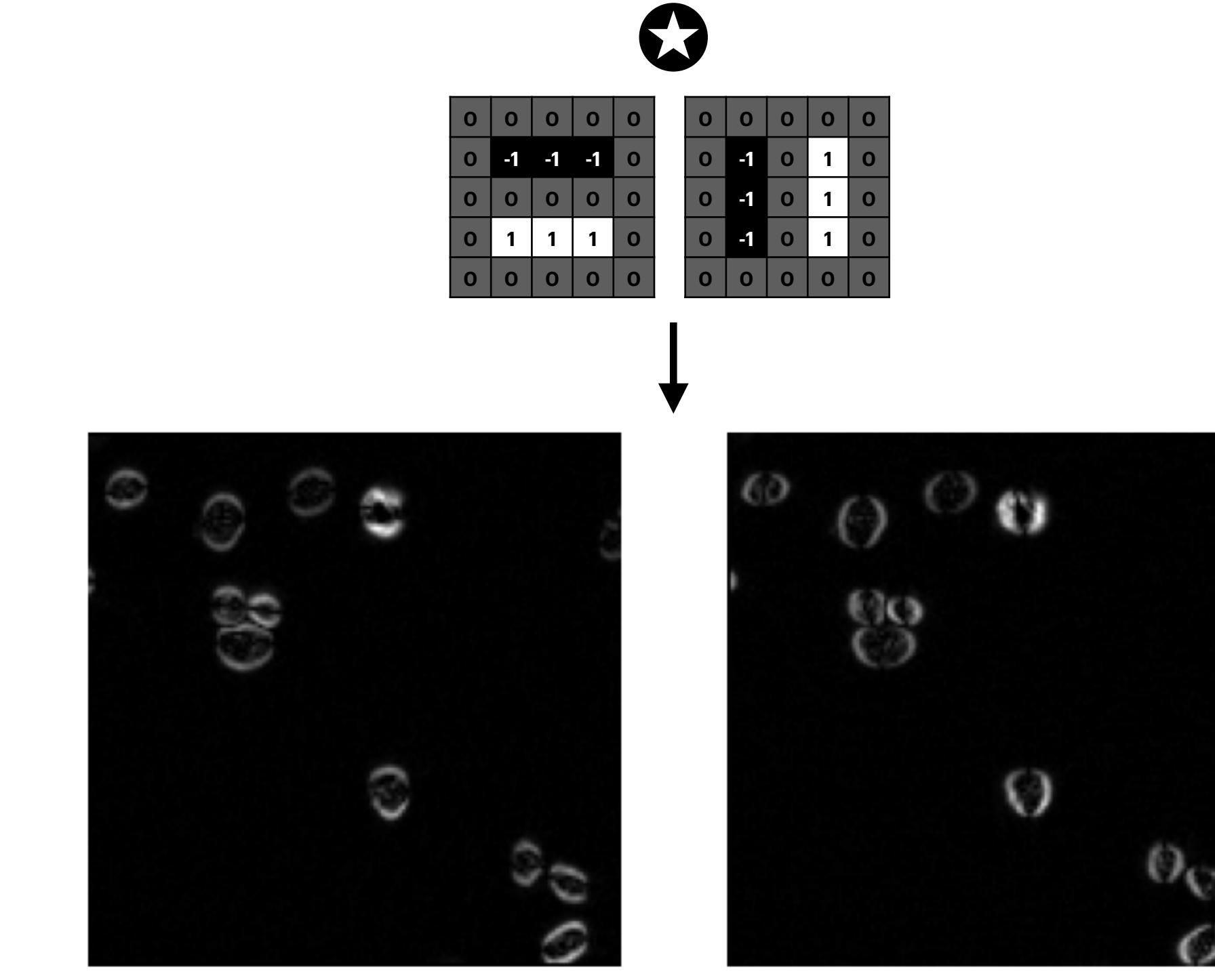
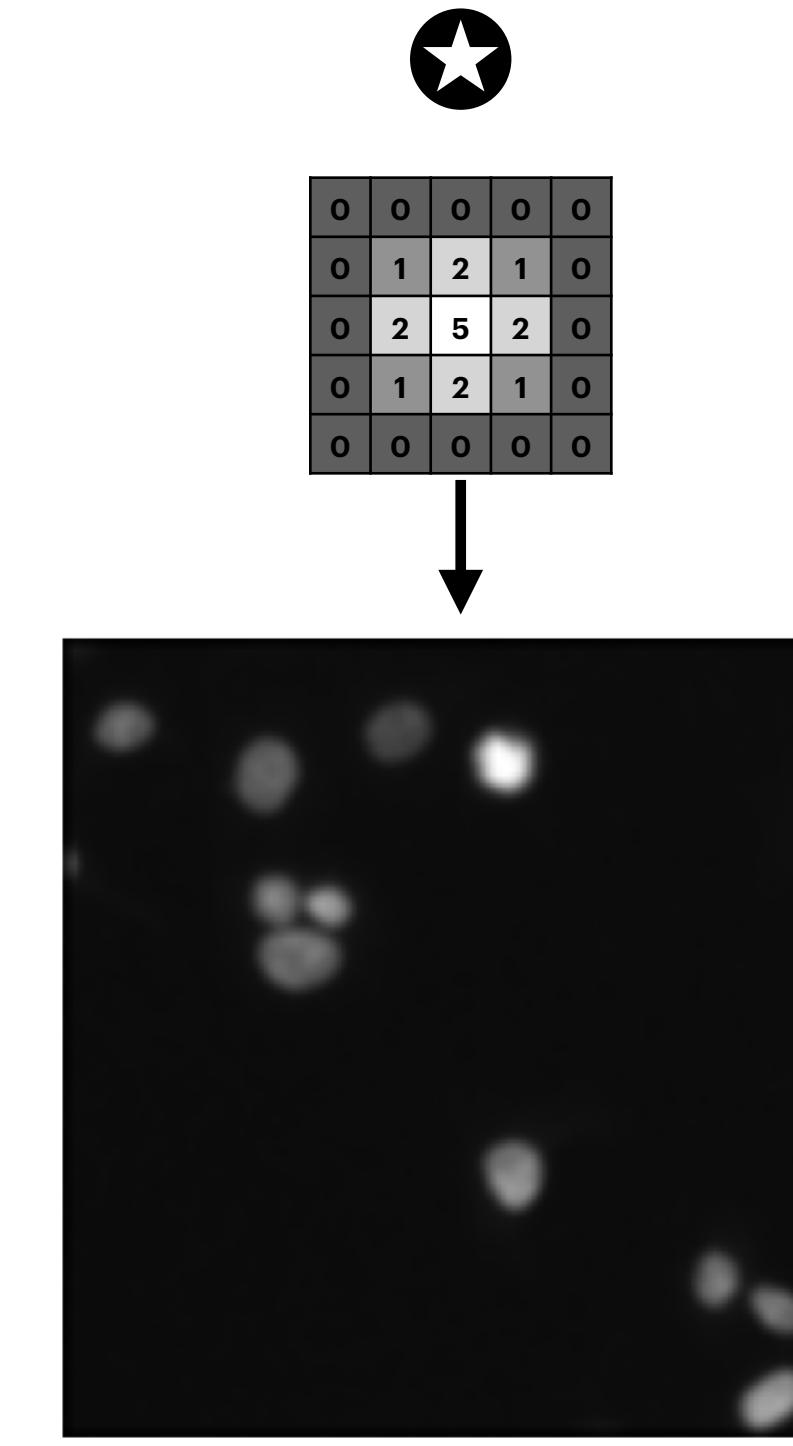
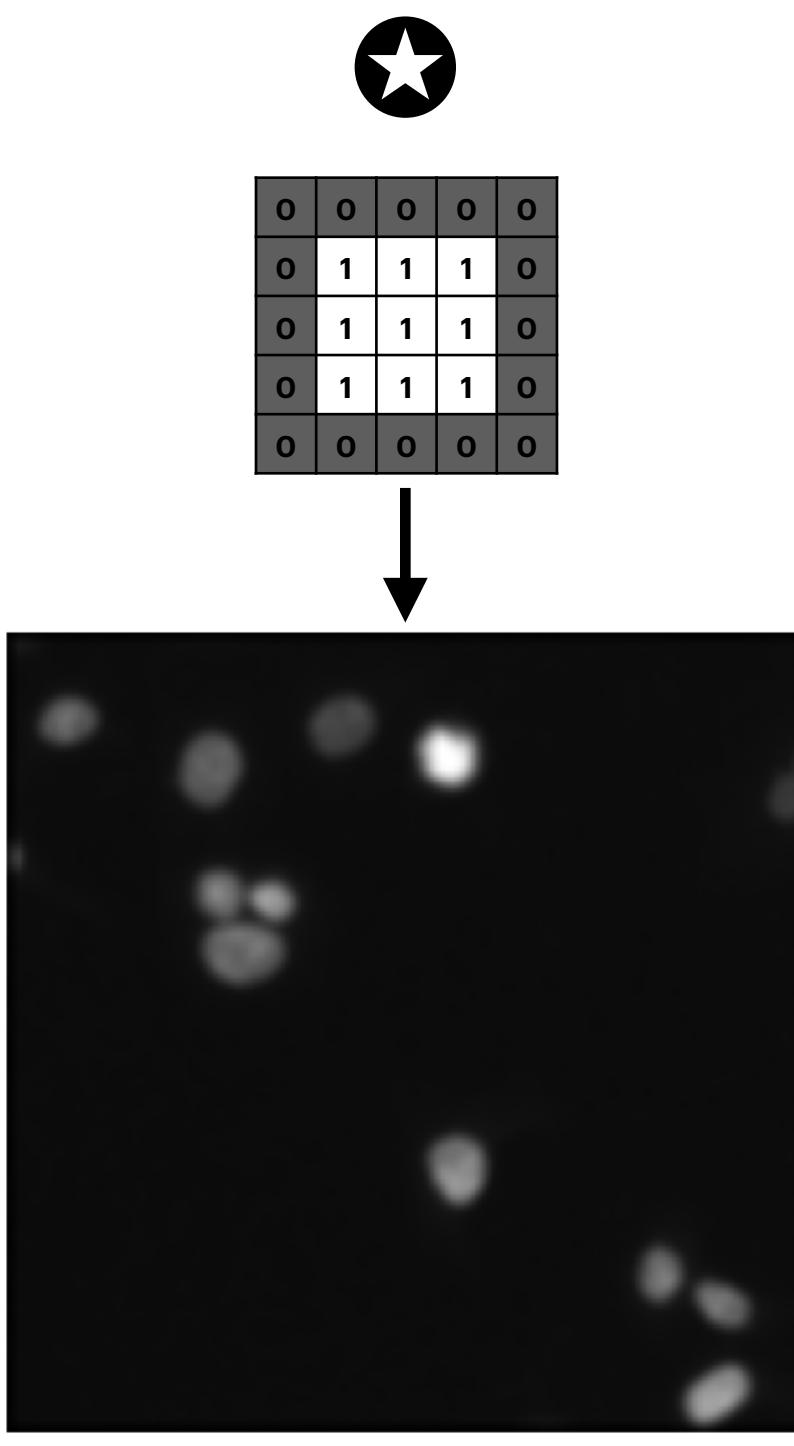
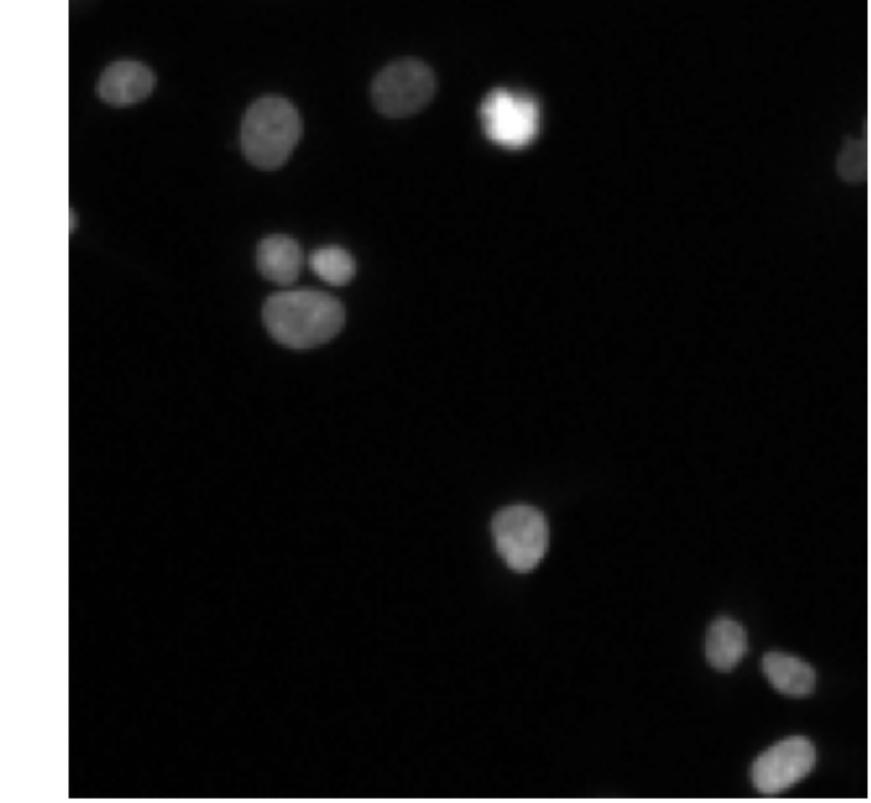
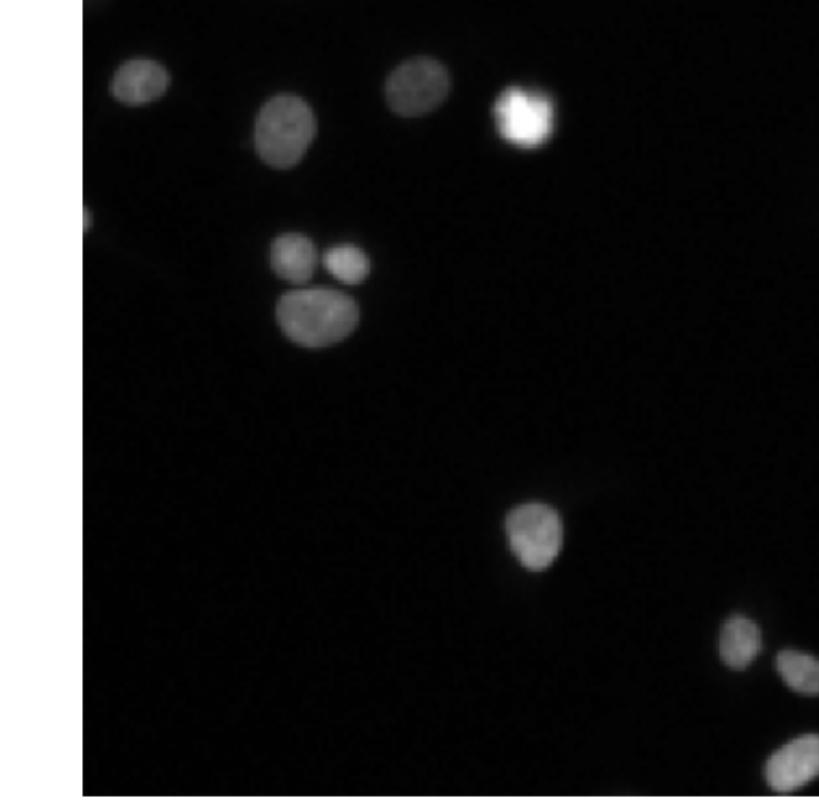
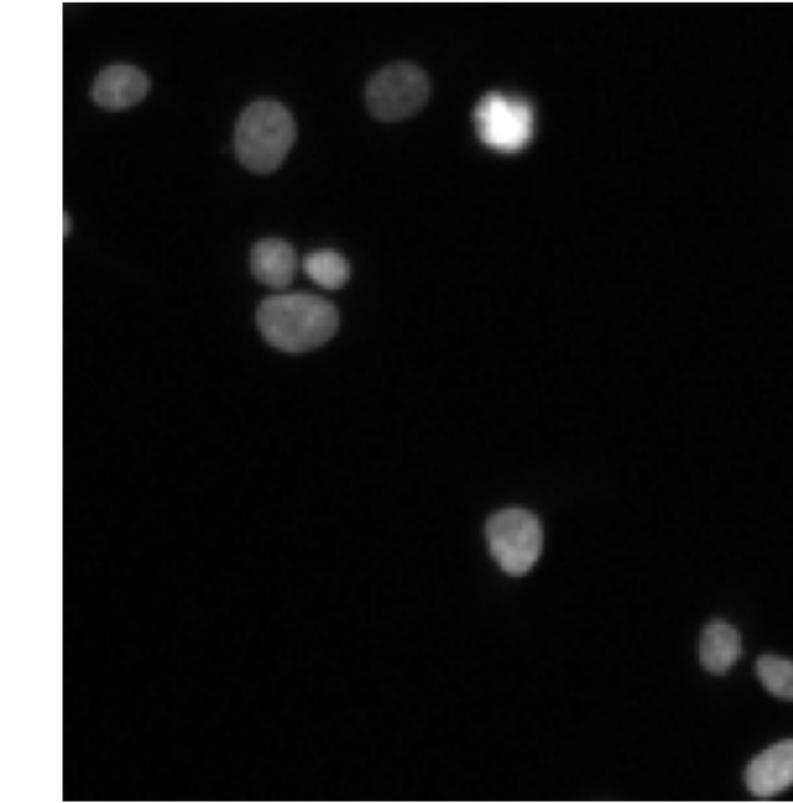
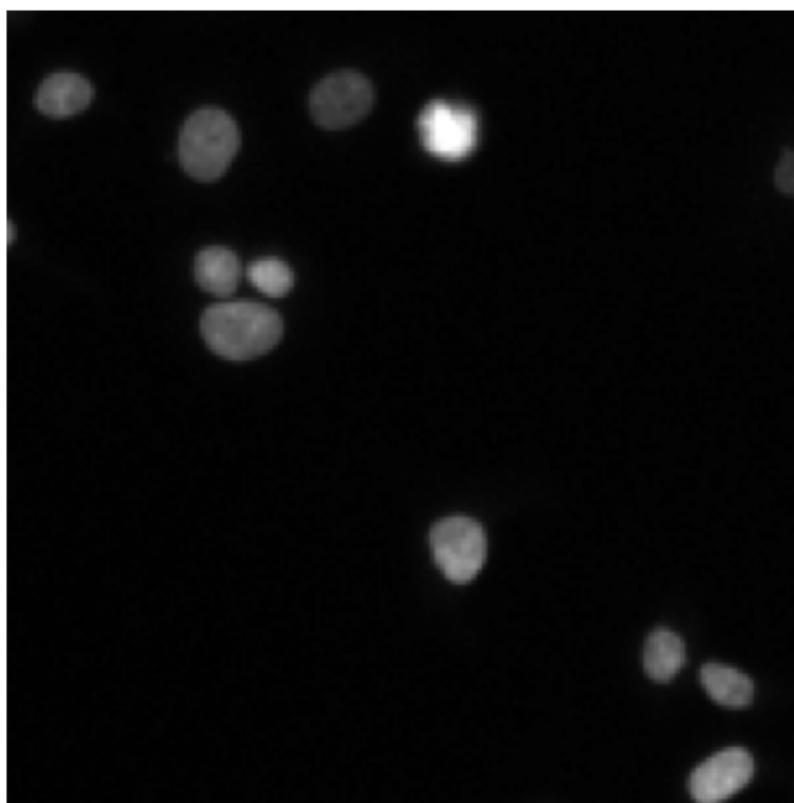


Edge Filters

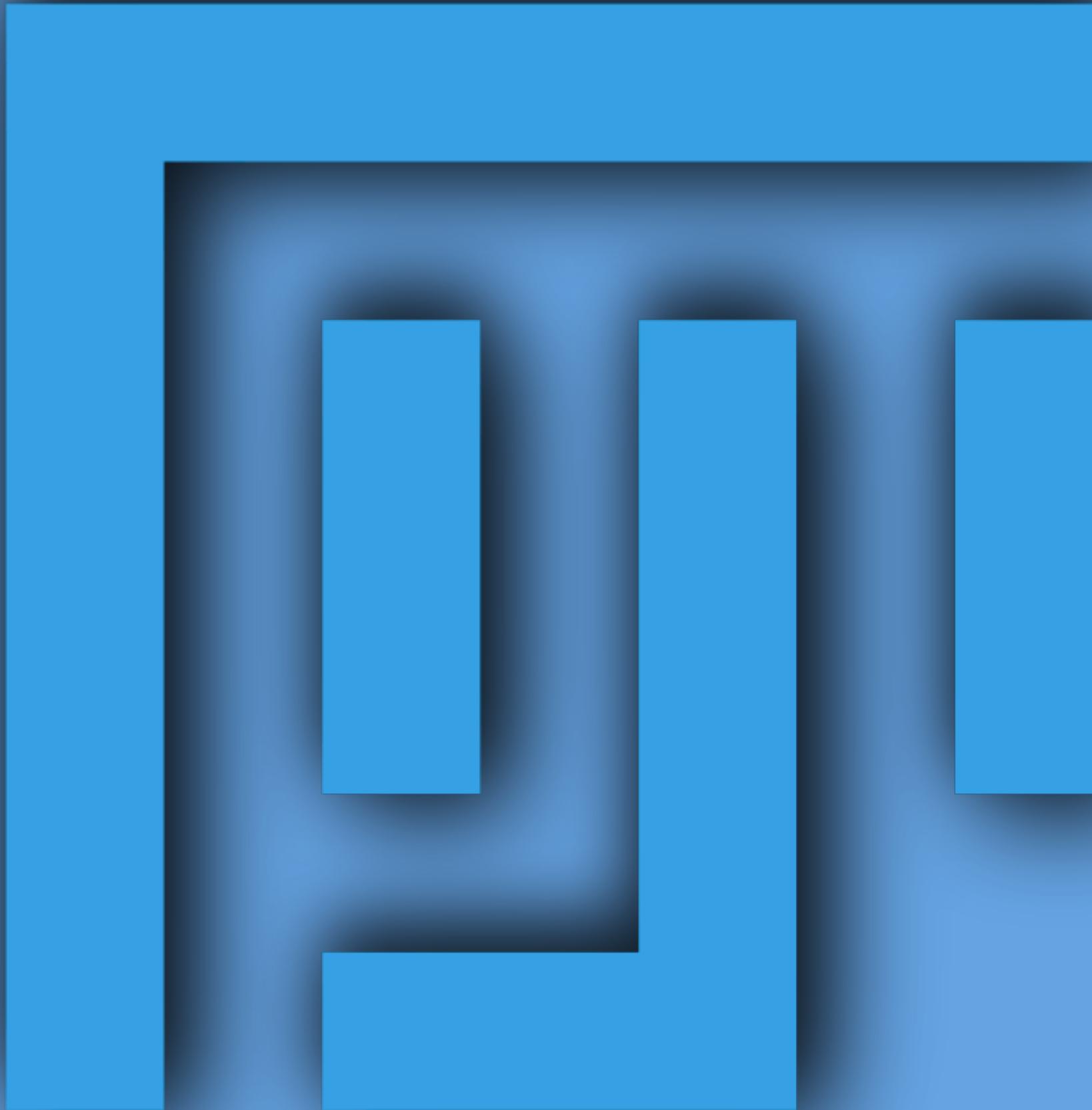
Prewitt Operator



Different Filters in Action



Filtering — exercises



- 3.1 edge filters - vertical stripes**
- 3.2 edge filters - horizontal stripes**
- 3.3 edge filters - checkerboard**

Filtering — exercises



4.2 DAPI segmentation with filters and thresholding