CONVOLUTIONAL NEURAL NETWORKS:

DISCRETE CONVOLUTIONS

Convolution operations first published by D'Alembert in 1754

Discrete convolutions are matrix operations that can be used to apply **filters** to a matrix or array

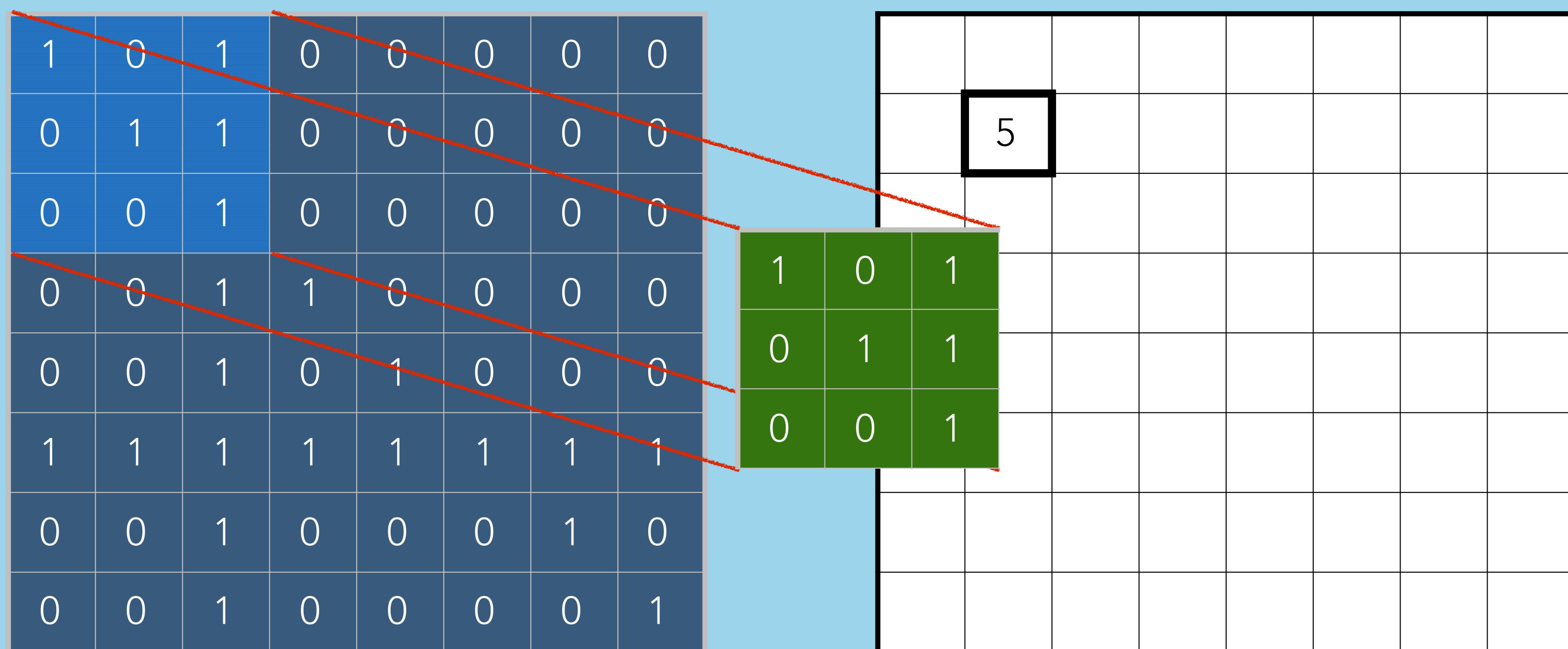
The convolutional neural network architecture was first described by Kunihiko Fukushima in 1980

pre-defined filters

Discrete convolutions are matrix operations that can be used to apply **filters** to a matrix or array

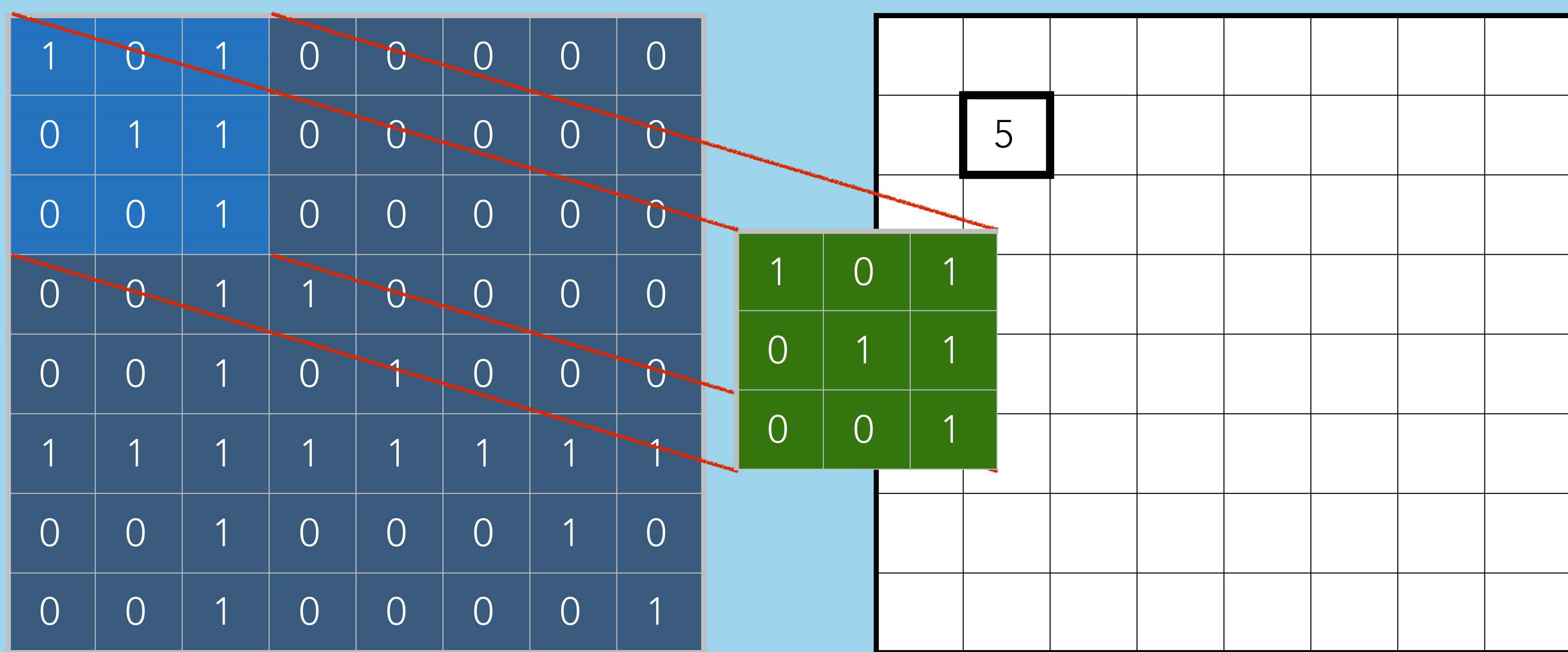
Discrete Convolutions $C = A \circledast h$

$$C[m,n] = \sum_{j=-\omega}^{\omega} \sum_{i=-\omega}^{\omega} h[i+\omega, j+\omega] * A[m+i, n+j]$$



Discrete Convolutions $C = A \circledast h$

Filter Stride Edges



Discrete Convolutions $C = A \circledast h$

Filter Stride Edges

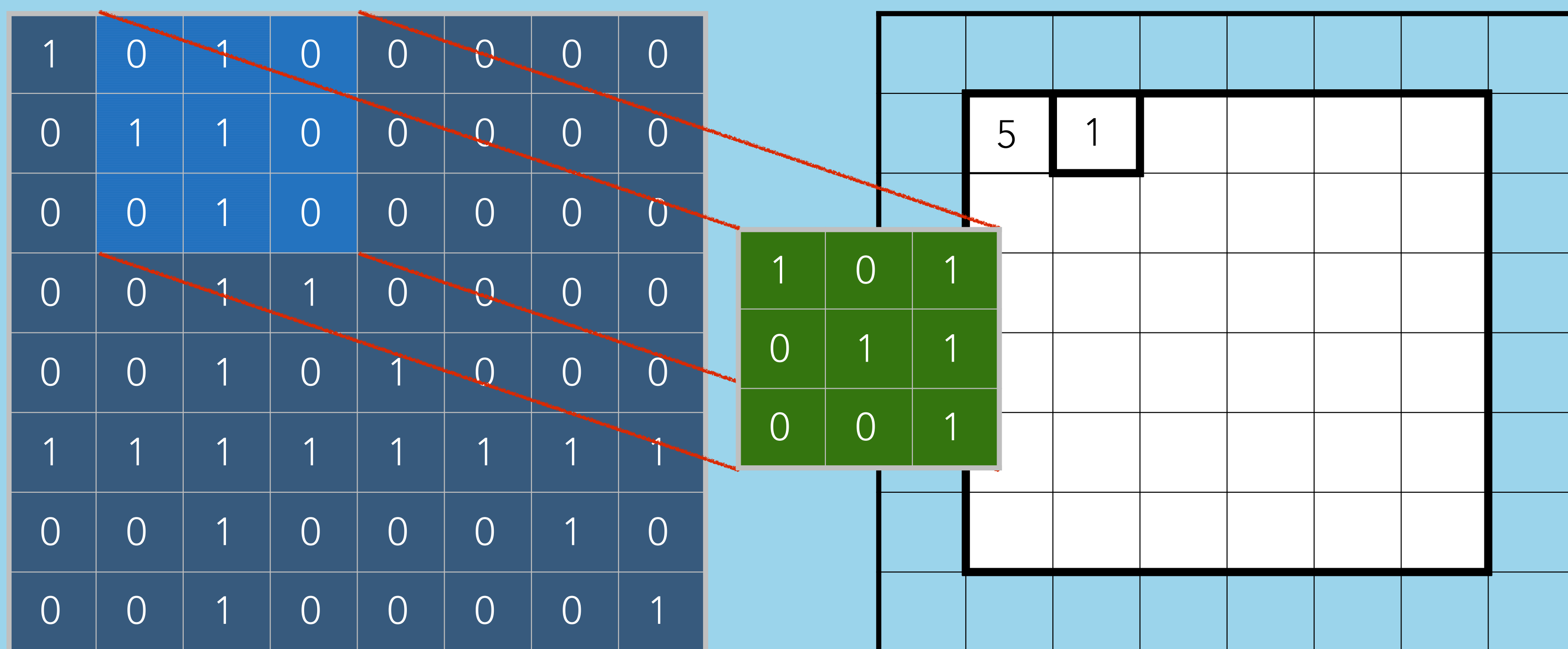
1	0	1	0	0	0	0	0
0	1	1	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	1	0	0	0	0
0	0	1	0	1	0	0	0
1	1	1	1	1	1	1	1
0	0	1	0	0	0	1	0
0	0	1	0	0	0	0	1

1	0	1
0	1	1
0	0	1

	5	1					

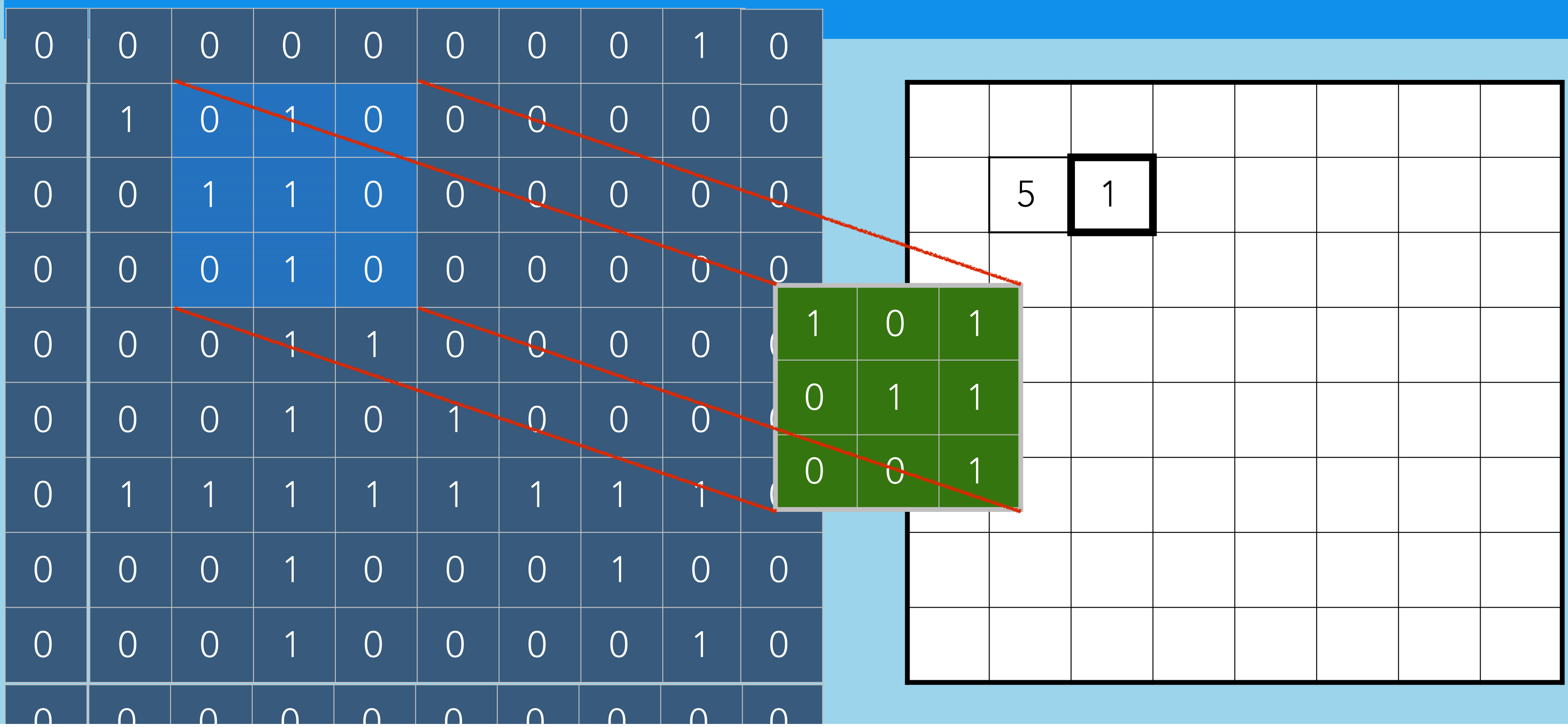
Discrete Convolutions $C = A \circledast h$

Filter Stride Edges



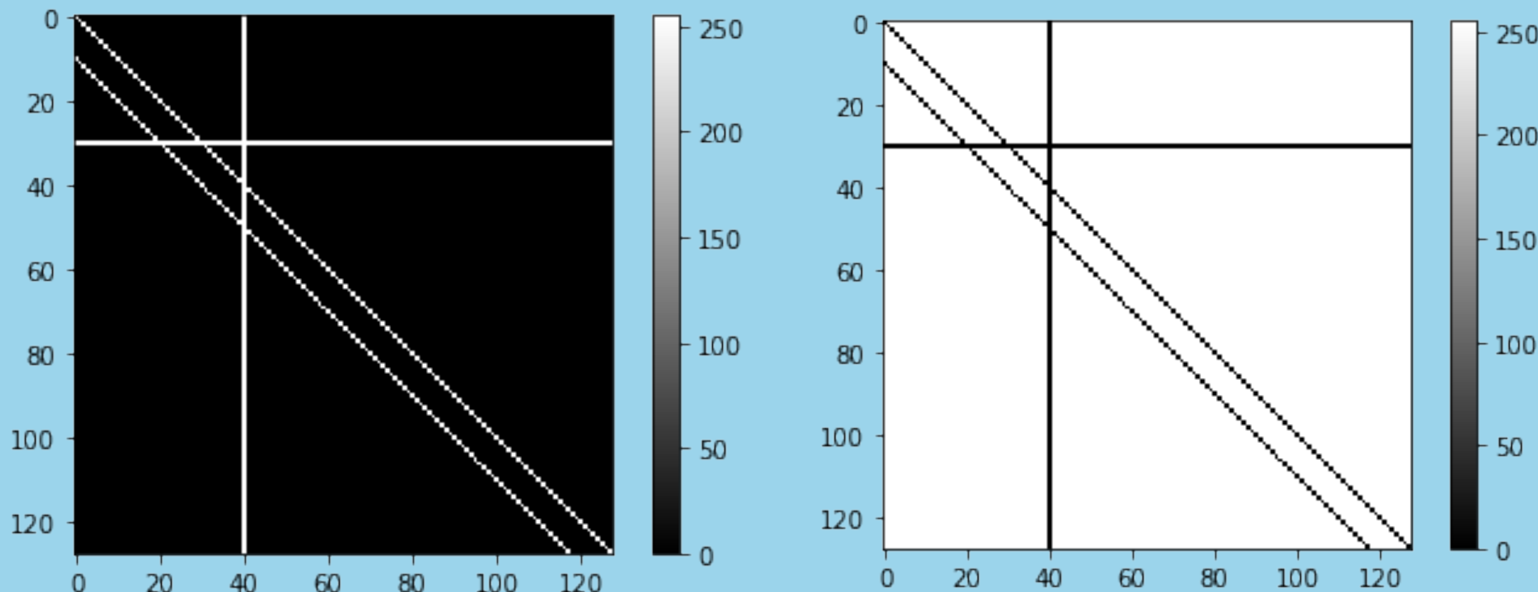
Discrete Convolutions $C = A \circledast h$

Filter Stride Edges



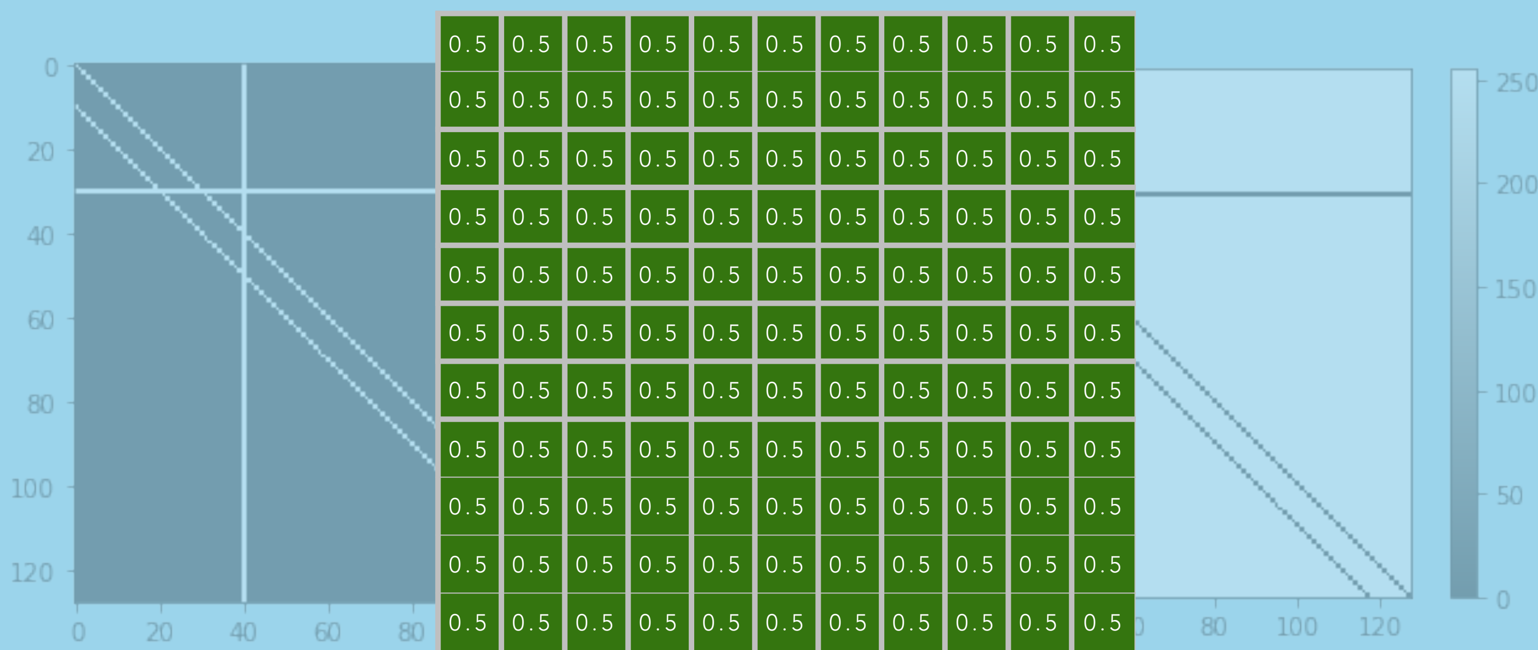
Discrete Convolutions $C = A \circledast h$

Filter	Stride	Edges
(11x11)	1	None



Discrete Convolutions $C = A \circledast h$

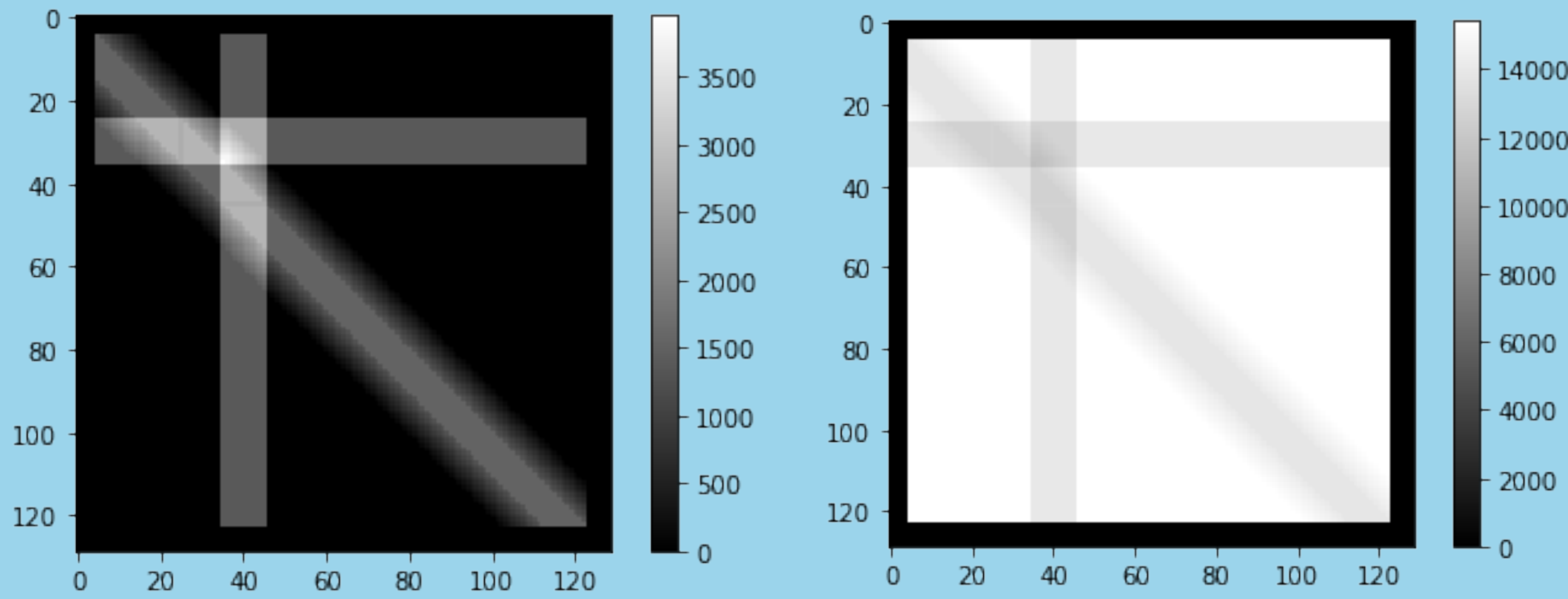
Filter	Stride	Edges
(11x11)	1	None



0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5

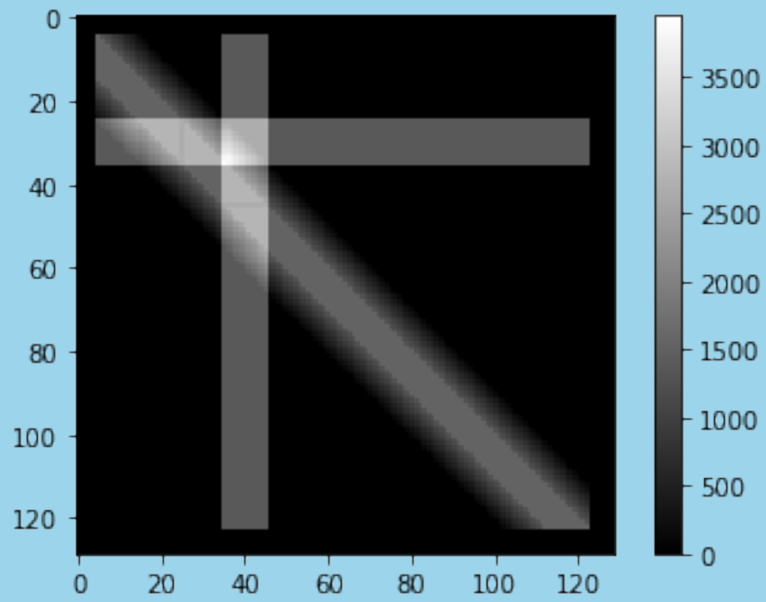
Discrete Convolutions $C = A \circledast h$

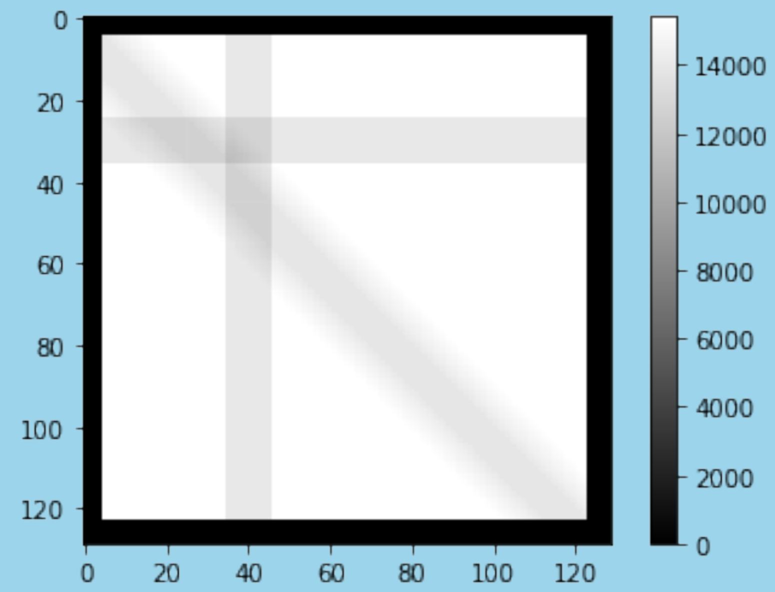
Filter	Stride	Edges
(11x11)	1	None



Discrete Convolutions $C = A \circledast h$

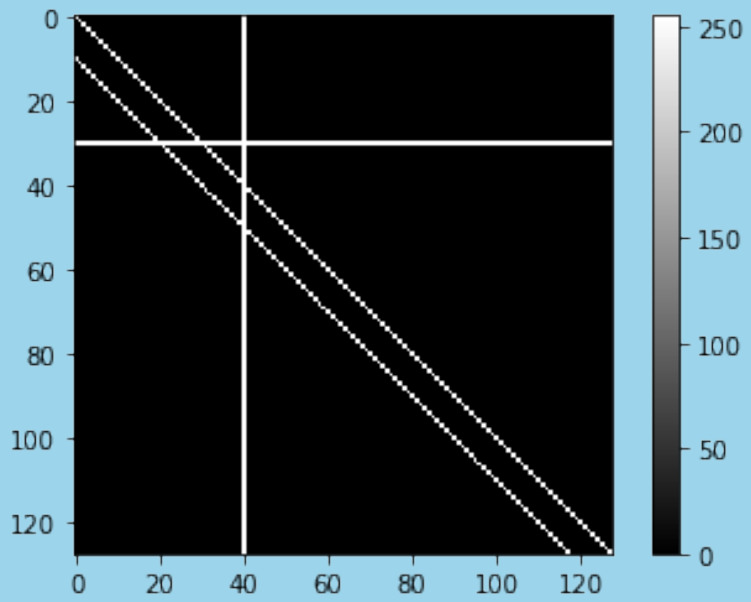
Blurring Filter

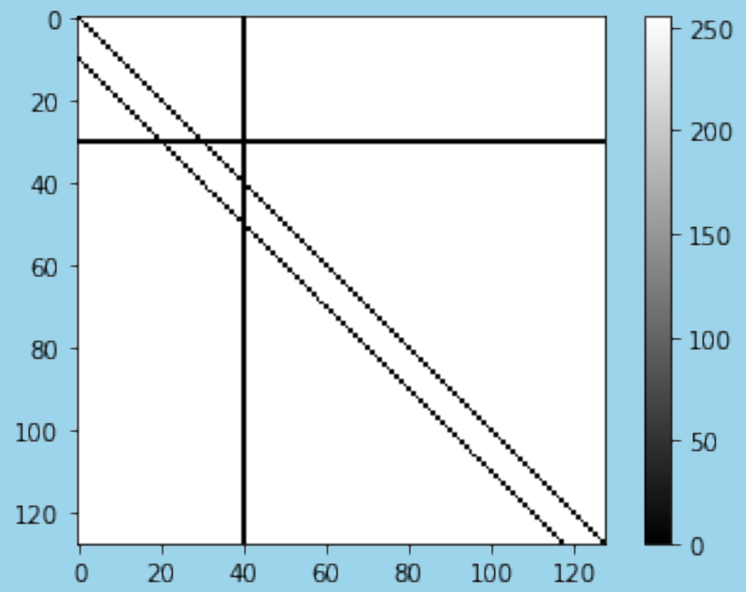




Discrete Convolutions $C = A \circledast h$

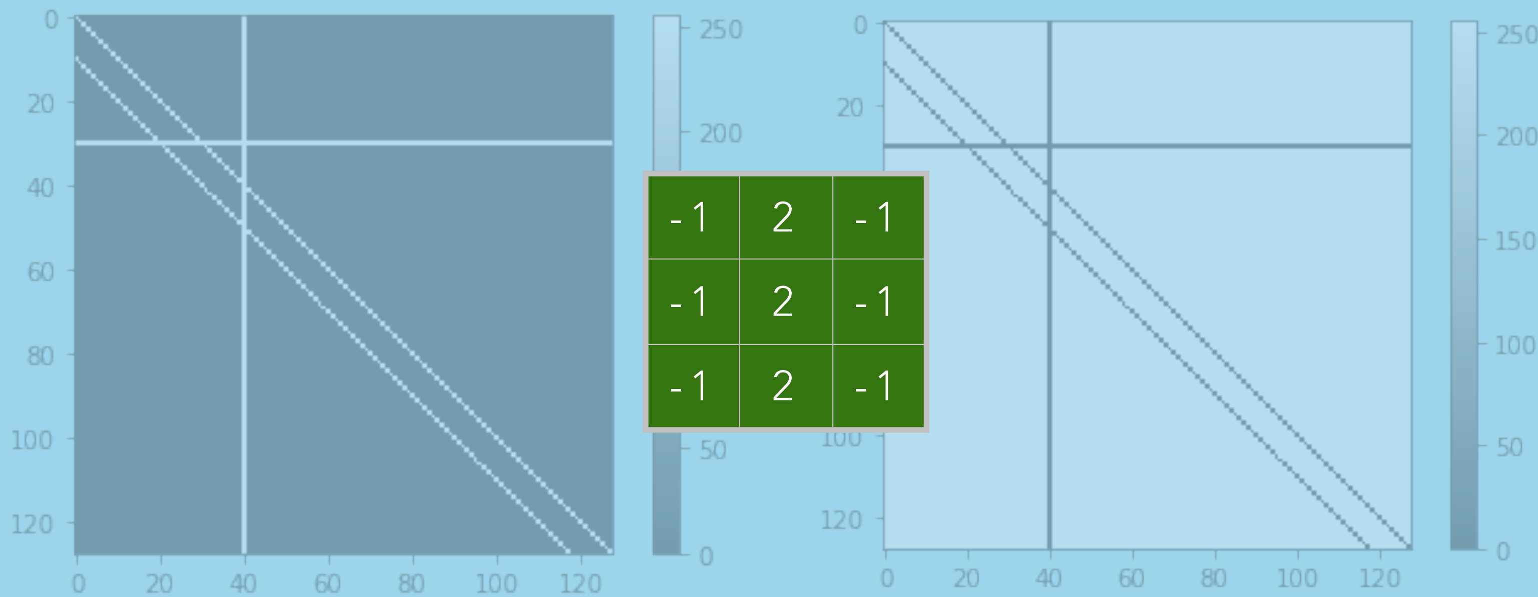
Filter	Stride	Edges
(3x3)	1	None





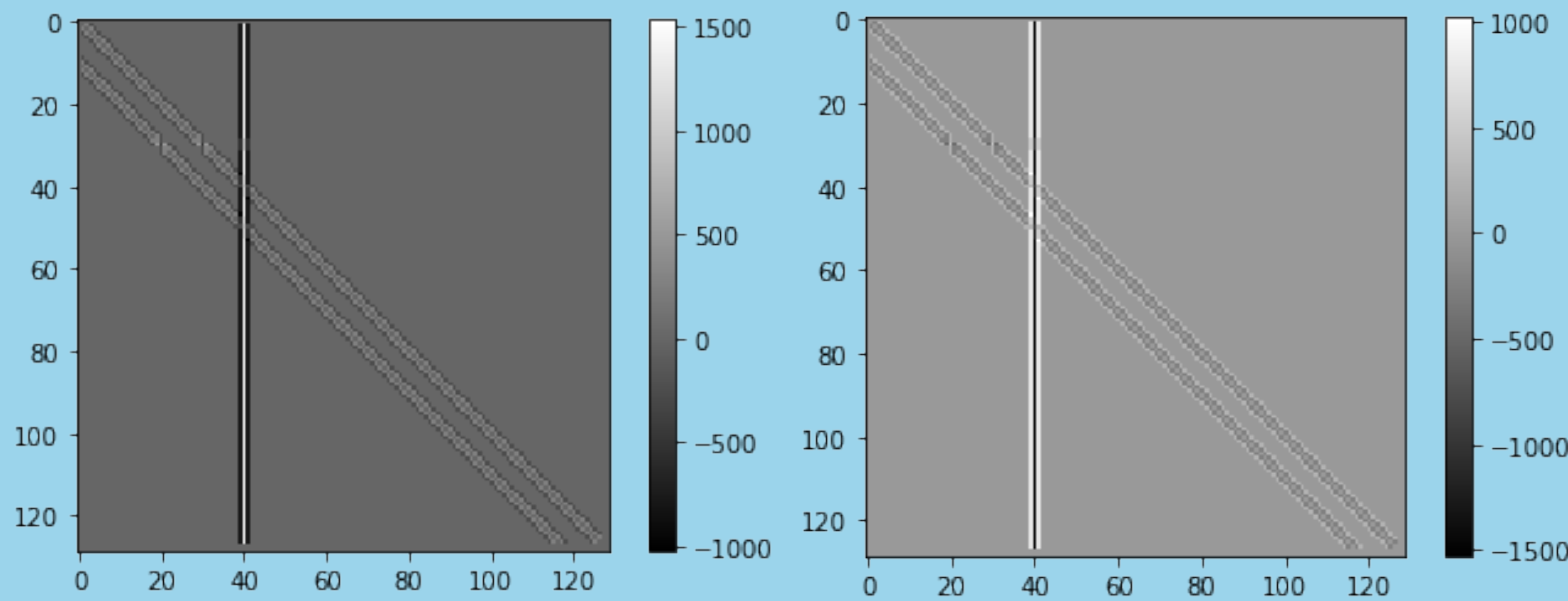
Discrete Convolutions $C = A \circledast h$

Filter	Stride	Edges
(3x3)	1	None



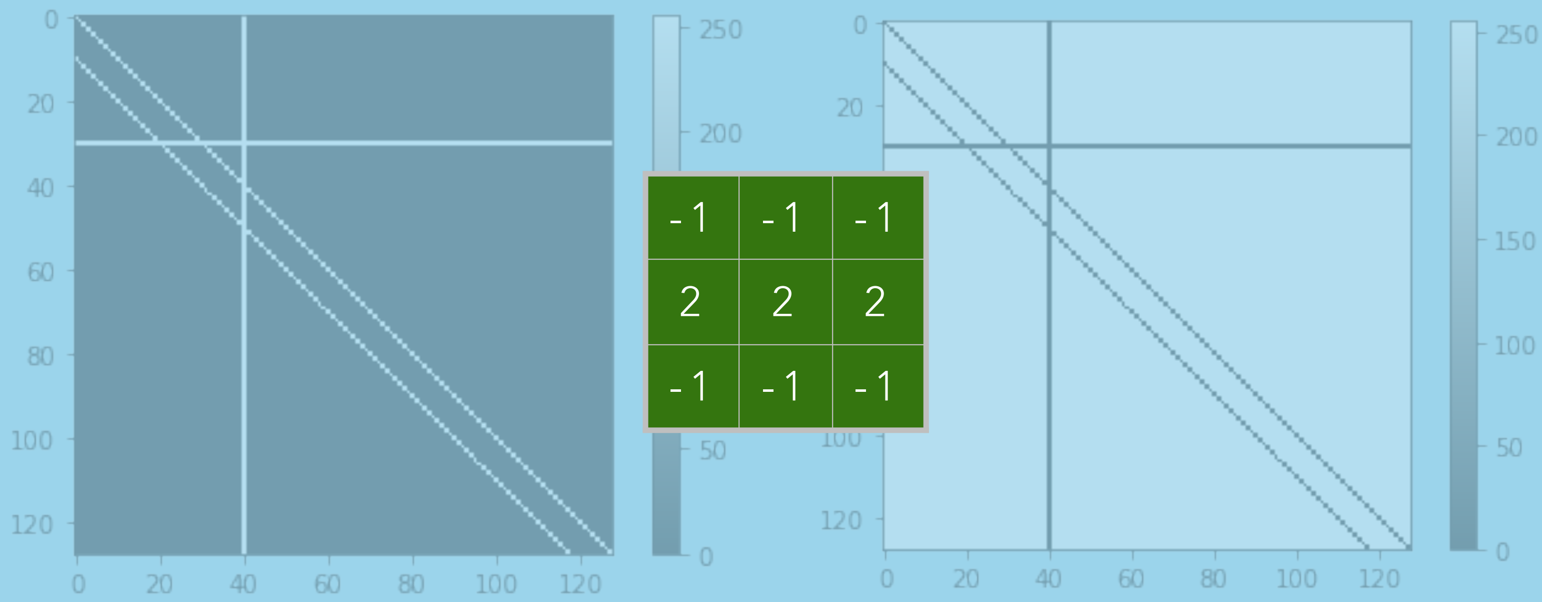
Discrete Convolutions $C = A \circledast h$

Filter	Stride	Edges
(3x3)	1	None



Discrete Convolutions $C = A \circledast h$

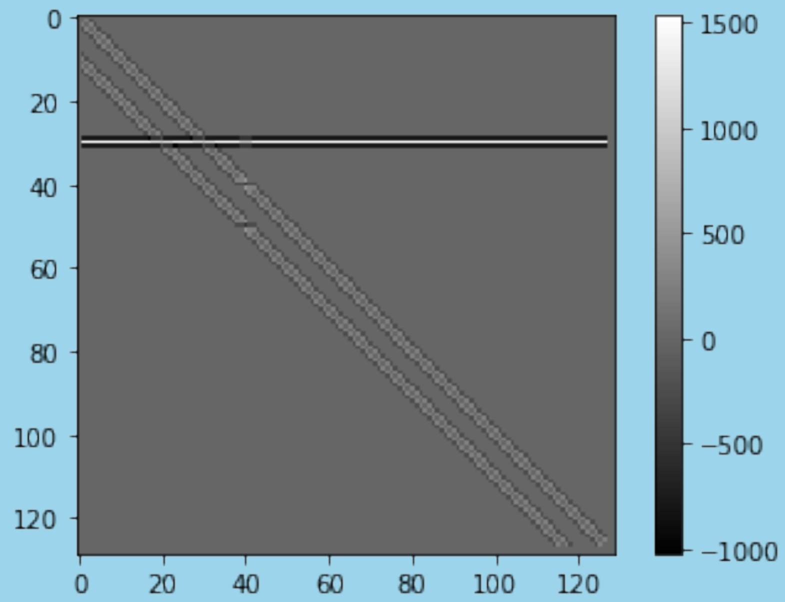
Filter	Stride	Edges
(3x3)	1	None

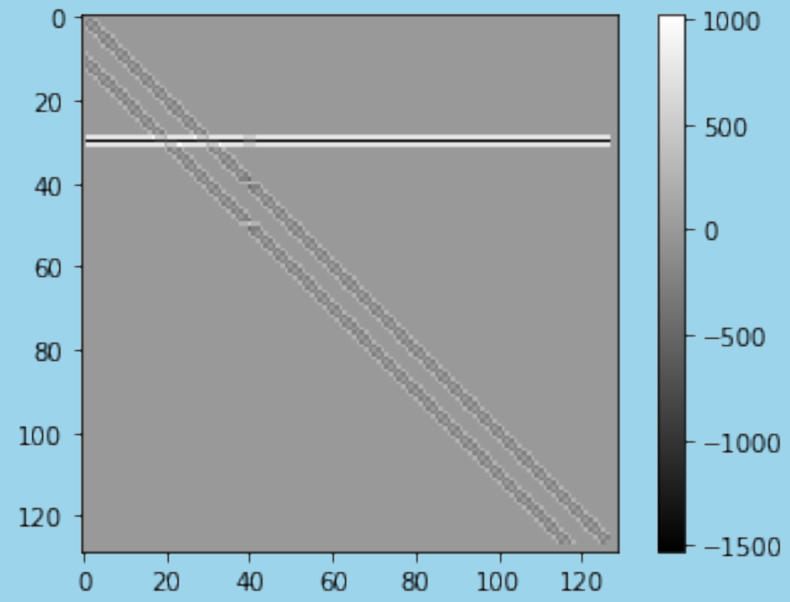


-1	-1	-1
2	2	2
-1	-1	-1

Discrete Convolutions $C = A \circledast h$

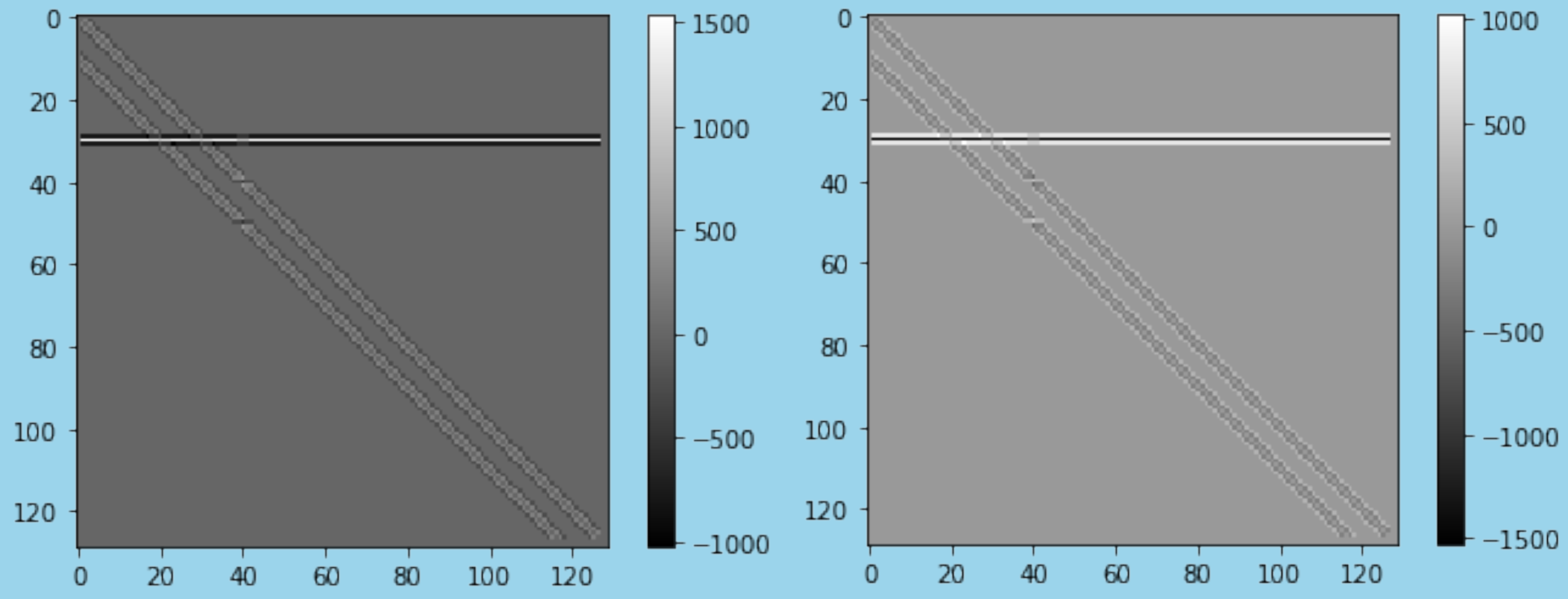
Filter	Stride	Edges
(3x3)	1	None





Discrete Convolutions $C = A \circledast h$

Line Detection / Extraction



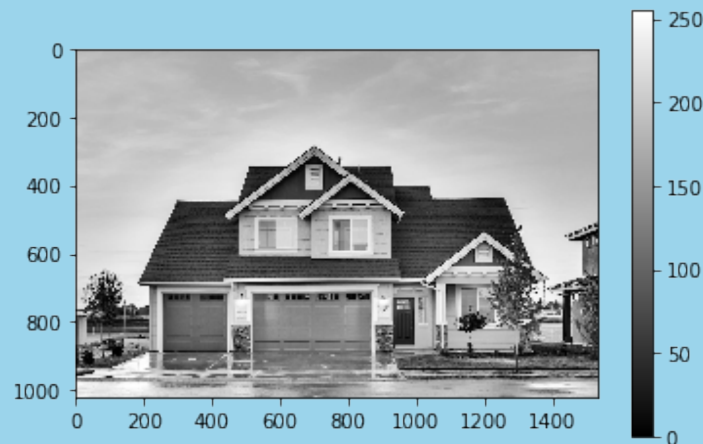
Discrete Convolutions $C = A \circledast h$

Photograph

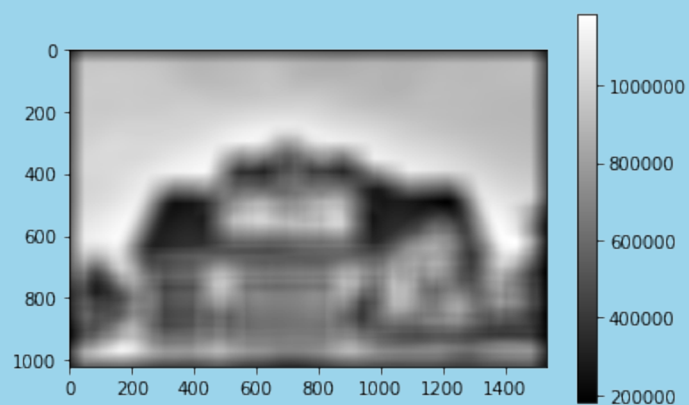
Discrete Convolutions $C = A \circledast h$

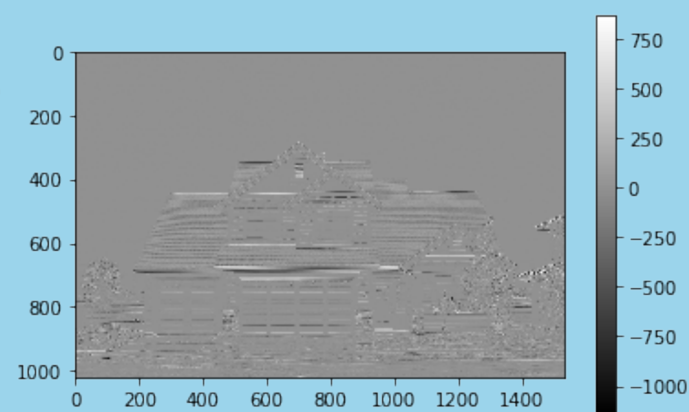
Photograph

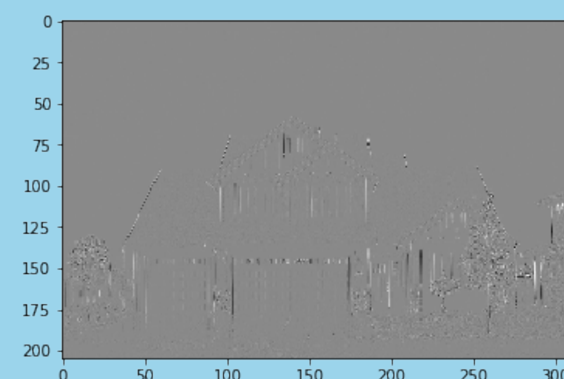
Discrete Convolutions $C = A \circledast h$



Filter
(101x101)

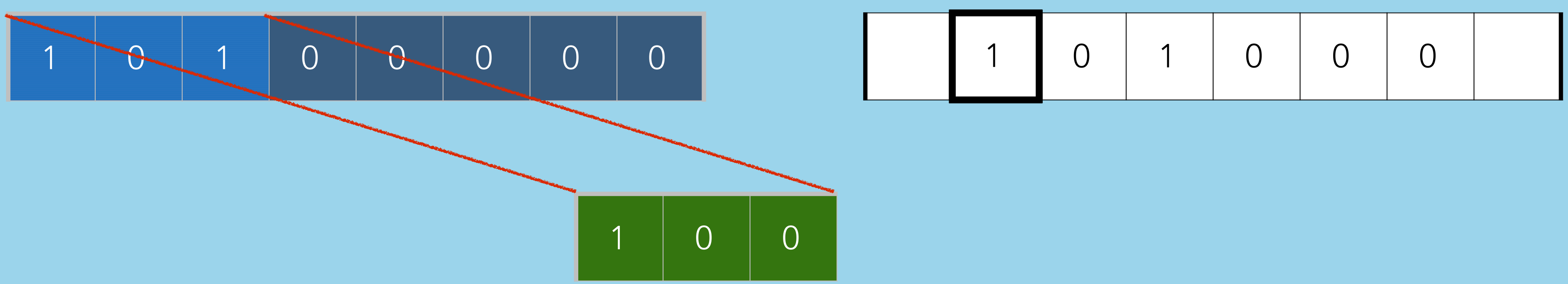






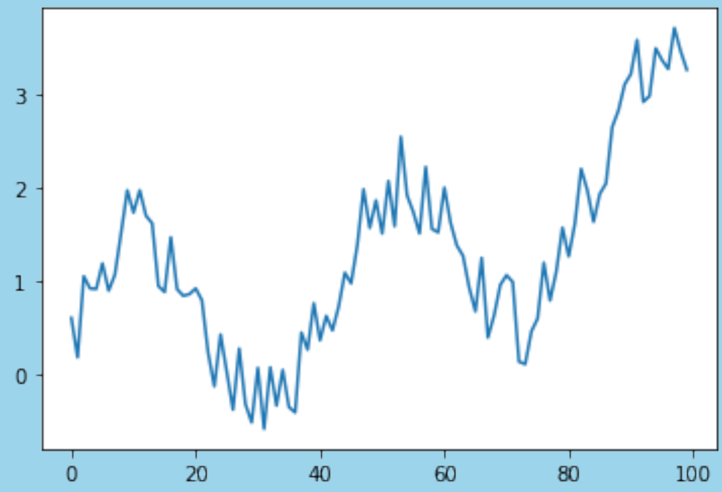
N-D Discrete Convolutions

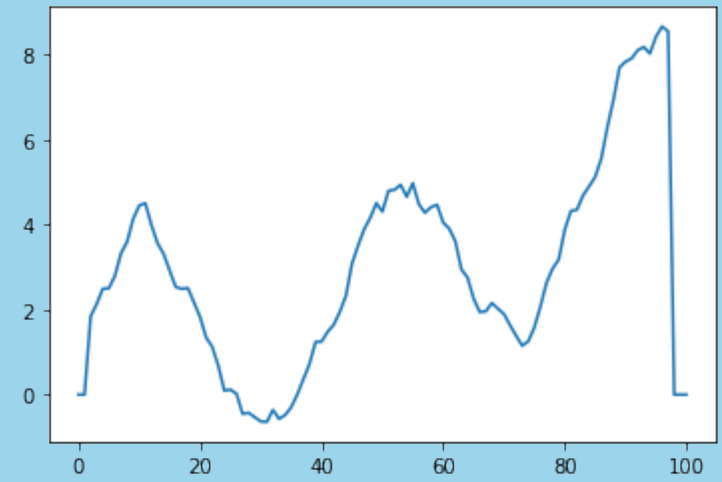
$$C[m,n] = \sum_{i=-\omega}^{\omega} h[i+\omega] * A[m+i]$$



N-D Discrete Convolutions

Filter	Stride	Edges
(1x5)	1	None





0.5	0.5	0.5	0.5	0.5

Discrete Convolutions

Discrete convolutions apply to **array** or **matrix**-like data

Discrete convolutions are matrix operations that can be used to apply **filters** to a matrix or array

These filters can **extract features** or transform the input