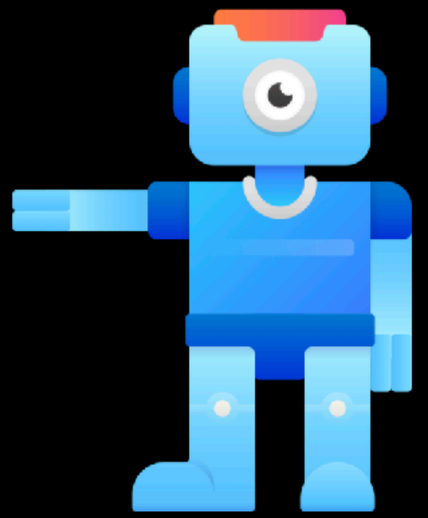


MODERN COMPUTER VISION

BY RAJEEV RATAN

Stride

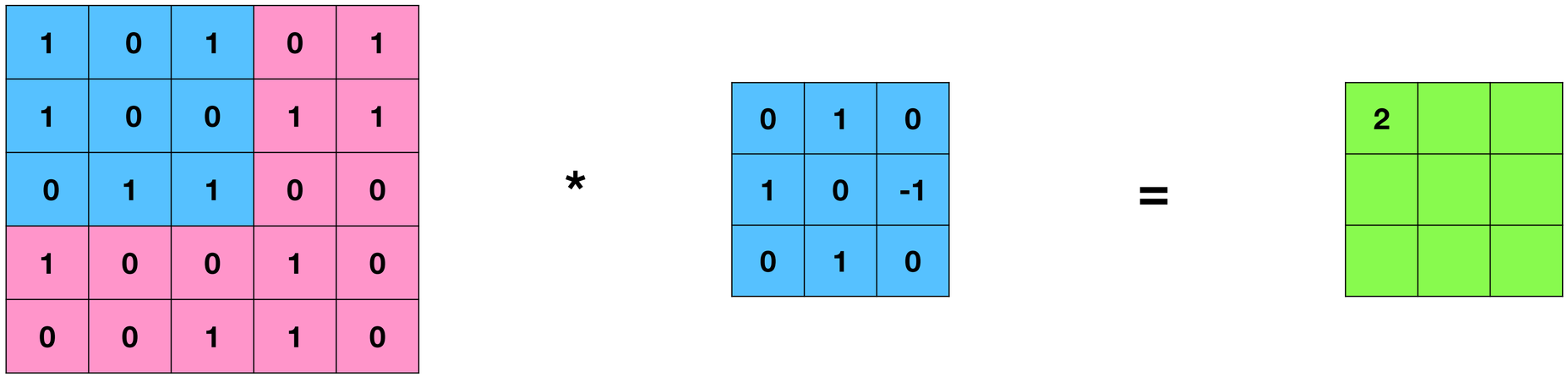
We look at a parameter that defines how we move our Conv Filter

Stride

Our Step Size

- Stride defines how many steps we take when sliding our Convolution Window across the input image

What a Stride of 1 Looks Like

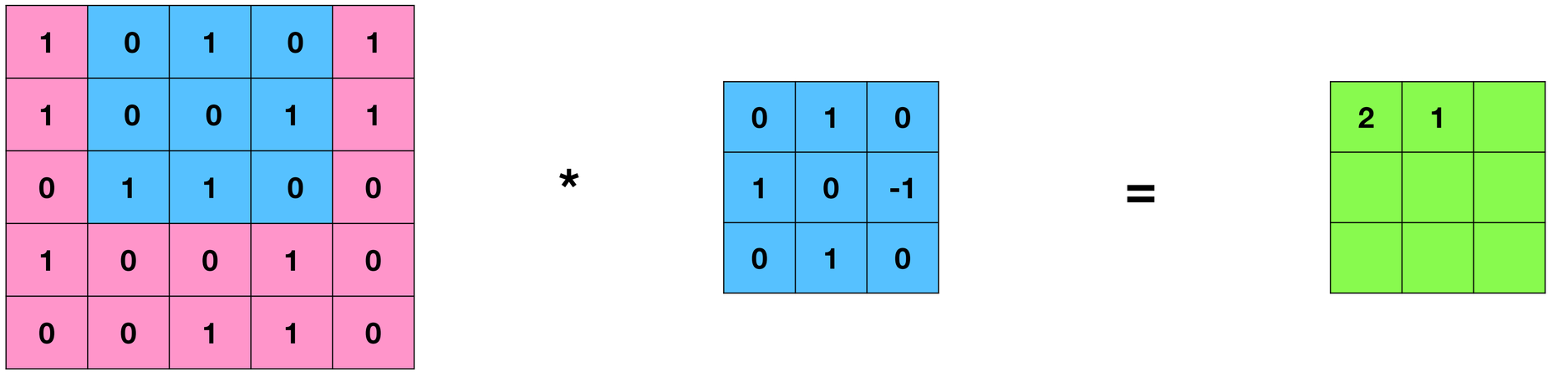


Input Image

Filter or Kernel

Output or Feature Map

What a Stride of 1 Looks Like

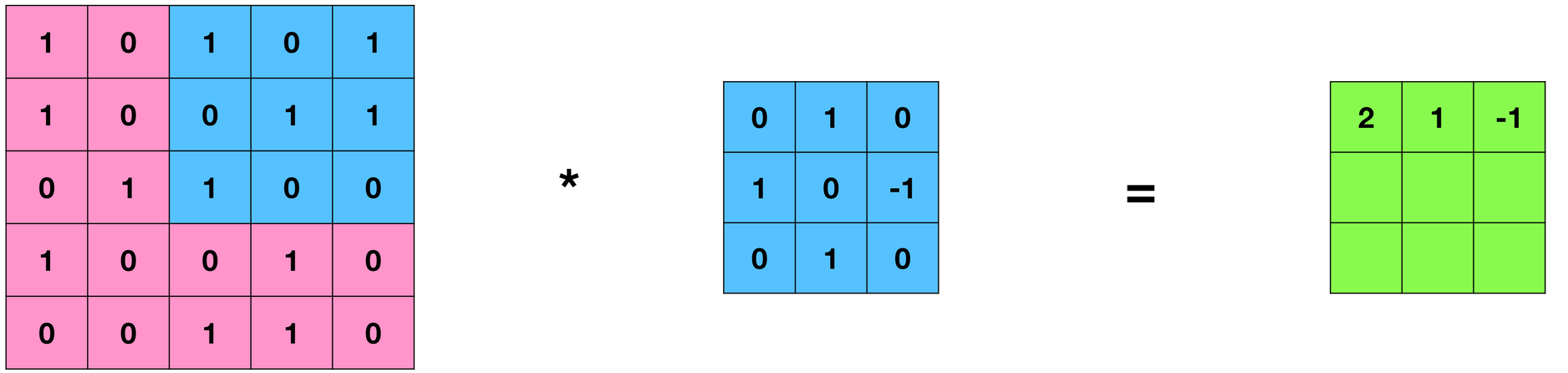


Input Image

Filter or Kernel

Output or Feature Map

What a Stride of 1 Looks Like

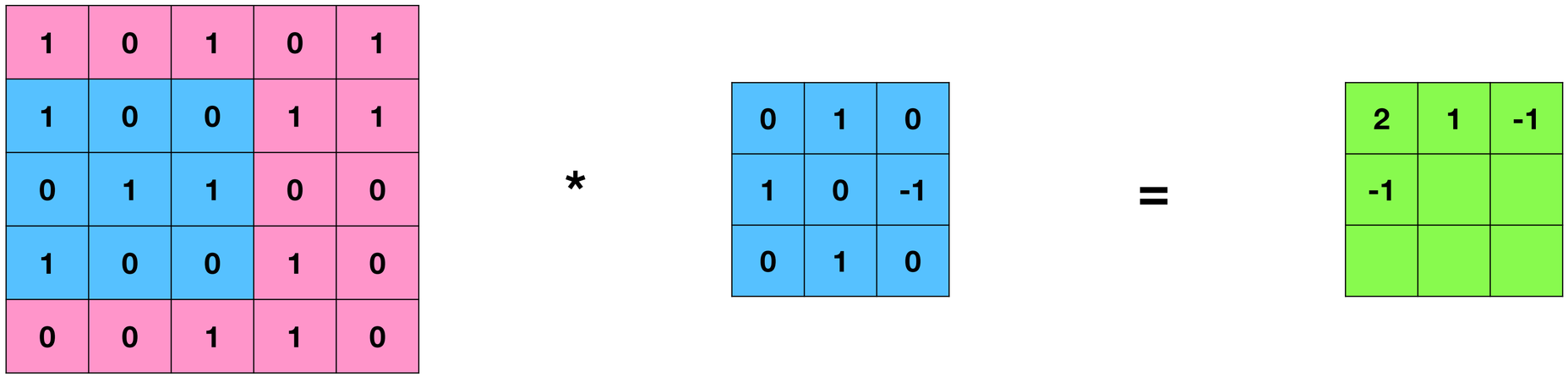


Input Image

Filter or Kernel

Output or Feature Map

What a Stride of 1 Looks Like

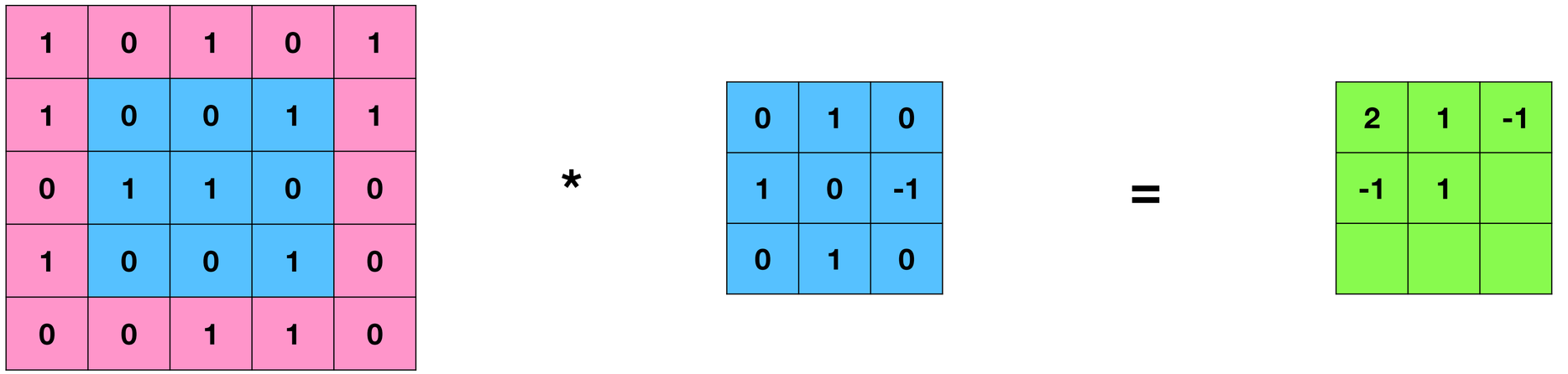


Input Image

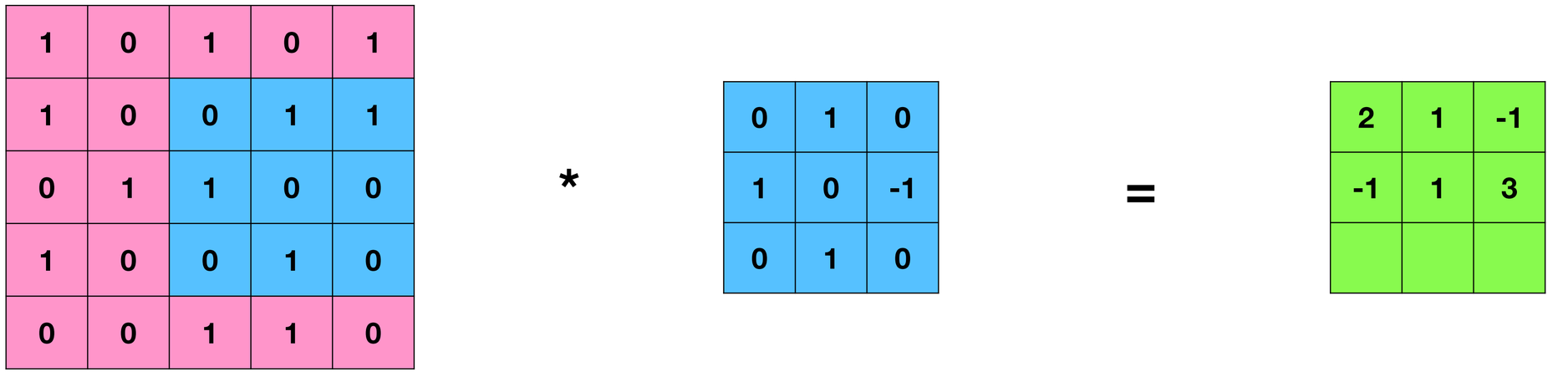
Filter or Kernel

Output or Feature Map

What a Stride of 1 Looks Like



What a Stride of 1 Looks Like

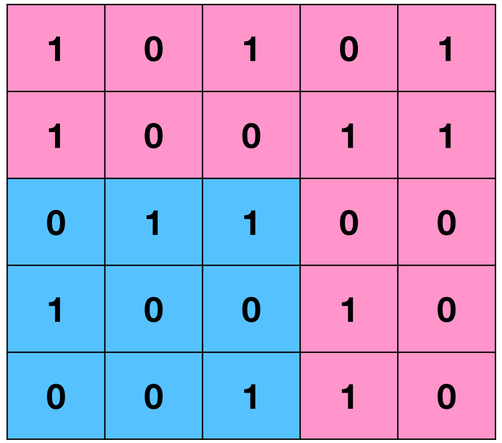


Input Image

Filter or Kernel

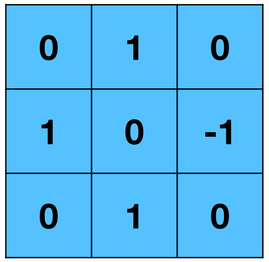
Output or Feature Map

What a Stride of 1 Looks Like



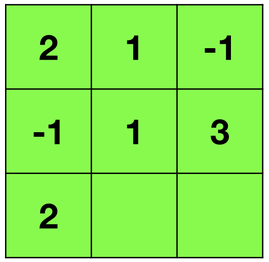
Input Image

*



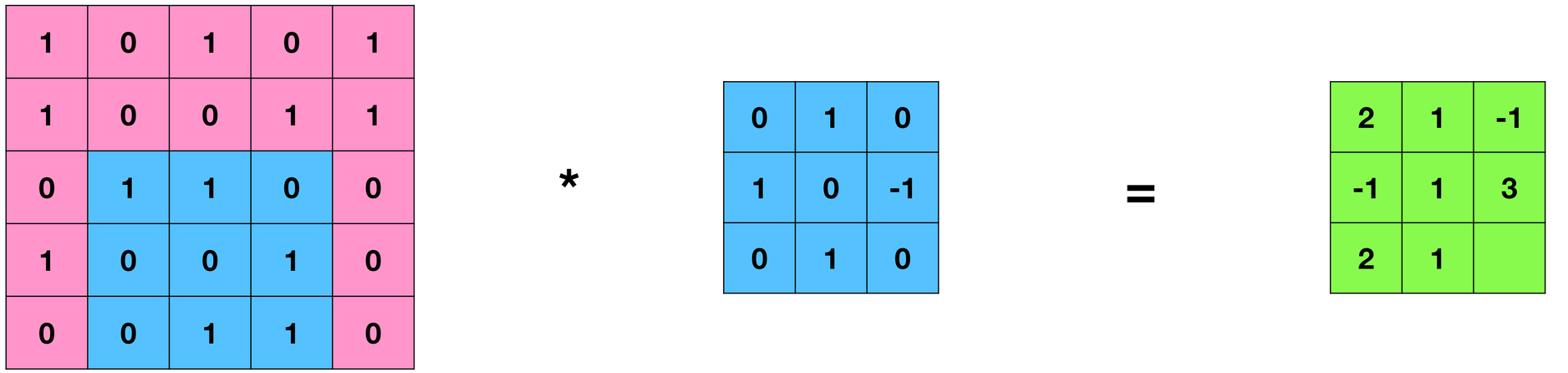
Filter or Kernel

=



Output or Feature Map

What a Stride of 1 Looks Like

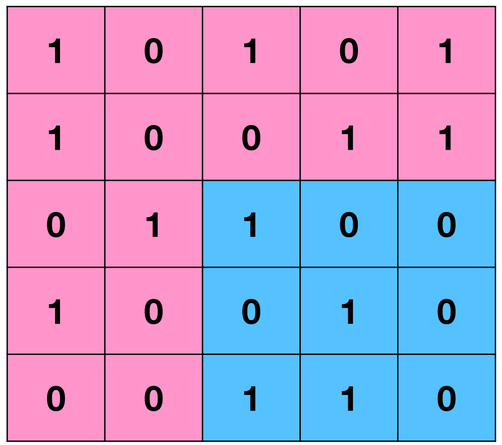


Input Image

Filter or Kernel

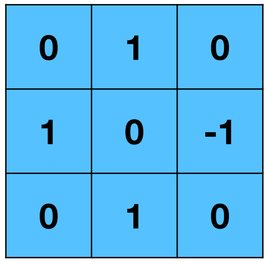
Output or Feature Map

What a Stride of 1 Looks Like



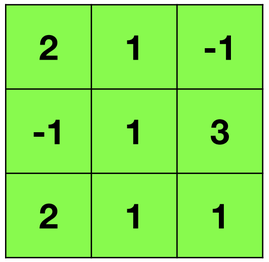
Input Image

*



Filter or Kernel

=

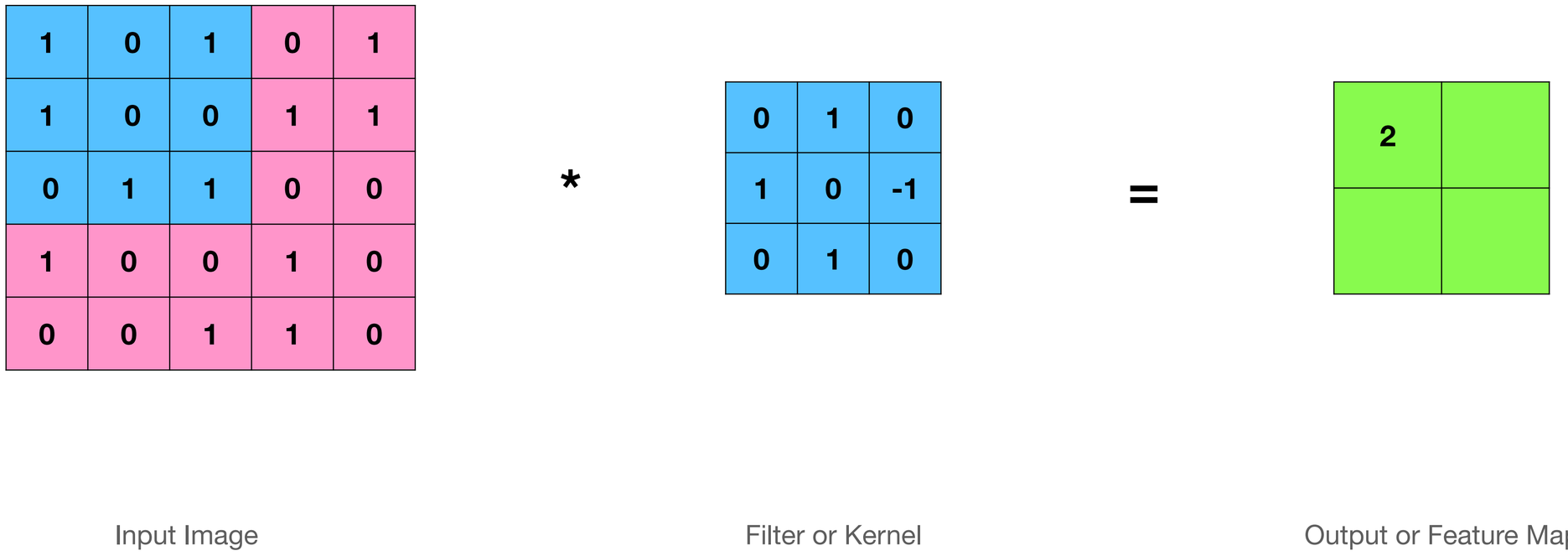


Output or Feature Map

What about a Stride of 2?

What a Stride of 2 Looks Like

We start off in the same position



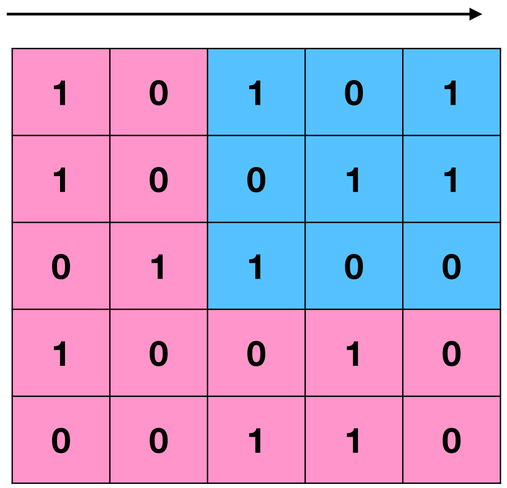
Input Image

Filter or Kernel

Output or Feature Map

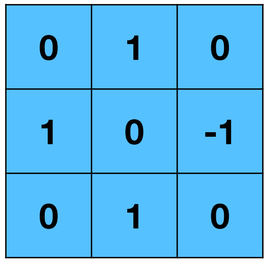
What a Stride of 2 Looks Like

Now we jump two spots to the left



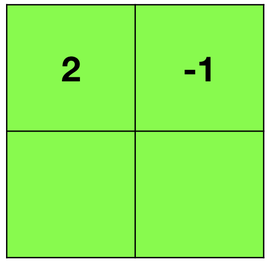
Input Image

*



Filter or Kernel

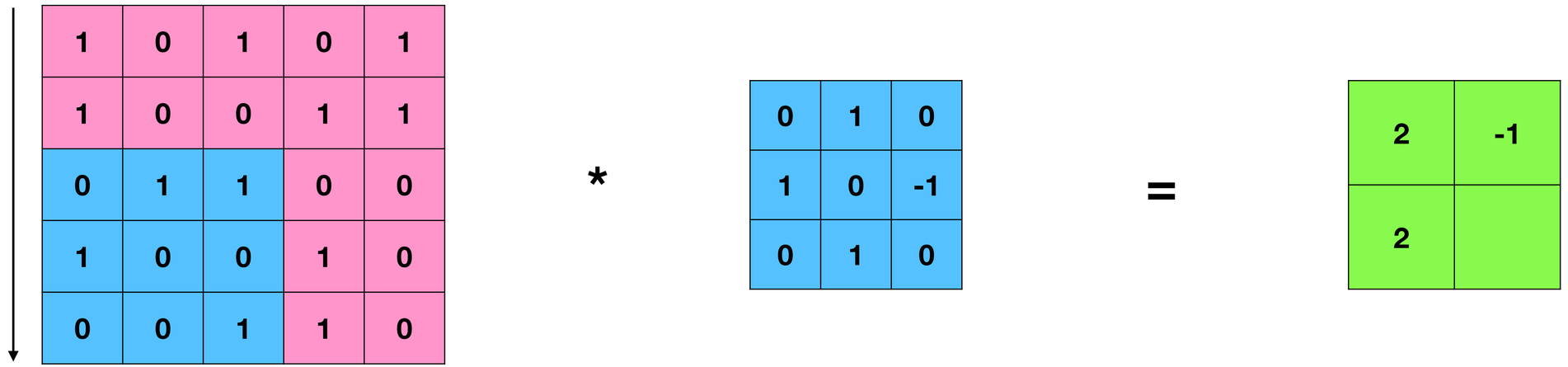
=



Output or Feature Map

What a Stride of 2 Looks Like

Now we go down, but by also 2 spots



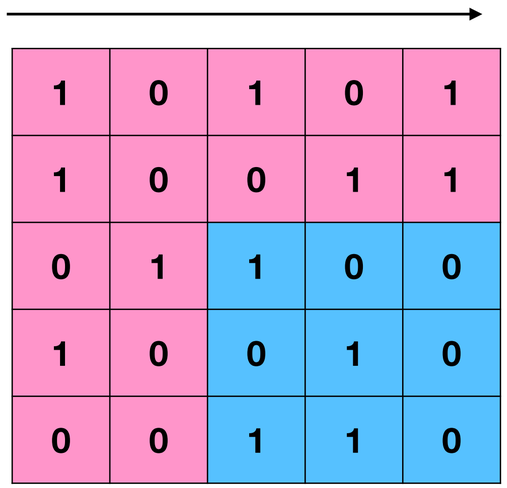
Input Image

Filter or Kernel

Output or Feature Map

What a Stride of 2 Looks Like

Now we jump two spots to the left



Input Image

*

0	1	0
1	0	-1
0	1	0

Filter or Kernel

=

2	-1
2	1

Output or Feature Map

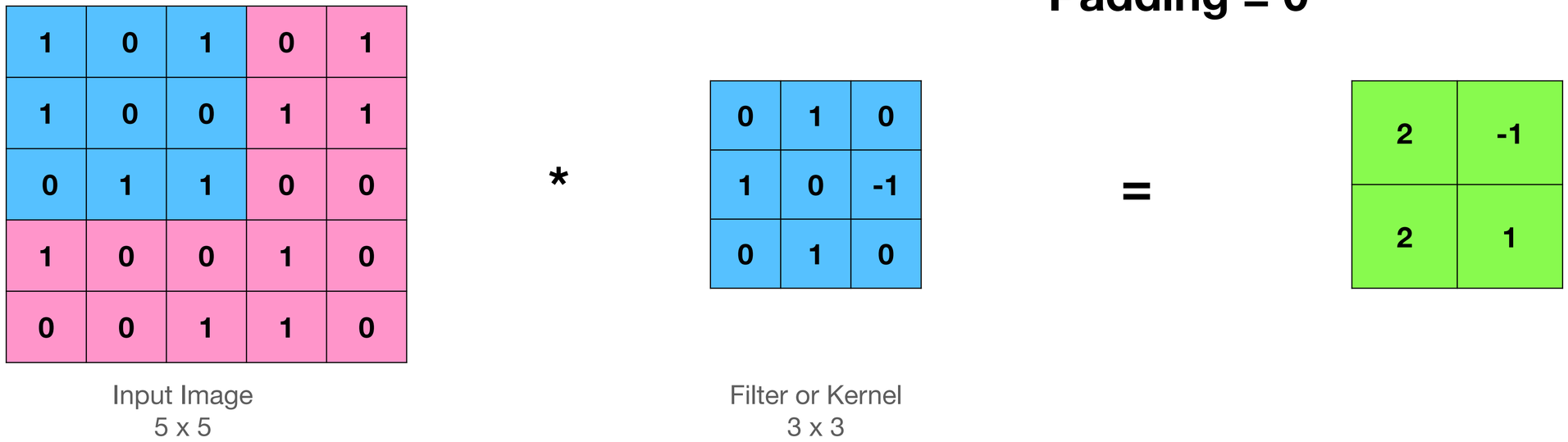
Stride Observations

- A larger Stride produced a **smaller** Feature Map output
- Larger Stride has **less overlap**
- We can use stride to **control the size of the Feature Map output**

Calculating Output Size

Using Stride and Padding

Stride = 2
Padding = 0



Input Image
5 x 5

Filter or Kernel
3 x 3

$$(n \times n) * (f \times f) = (\frac{n + 2p - f}{s} + 1) \times (\frac{n + 2p - f}{s} + 1) = (\frac{5 + (2 \times 0) - 3}{2} + 1) \times (\frac{5 + (2 \times 0) - 3}{2} + 1) = 2 \times 2$$

Calculating Output Size

Using Stride and Padding

Stride = 1
Padding = 0

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

Input Image
5 x 5

*

0	1	0
1	0	-1
0	1	0

Filter or Kernel
3 x 3

=

2	-1	1
2	1	0
1	-1	1

$$(n \times n) * (f \times f) = (\frac{n + 2p - f}{s} + 1) \times (\frac{n + 2p - f}{s} + 1) = (\frac{5 + (2 \times 0) - 3}{1} + 1) \times (\frac{5 + (2 \times 0) - 3}{1} + 1) = 3 \times 3$$

Calculating Output Size

When using Stride & Padding

Stride = 1

Padding = 1

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

Input Image
7 x 7

*

0	1	0
1	0	-1
0	1	0

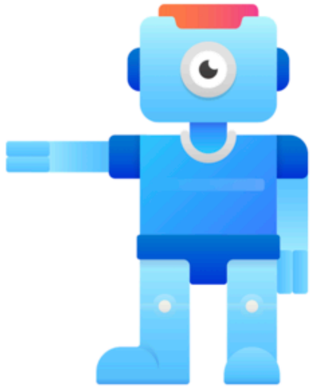
Filter or Kernel
3 x 3

=

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

$$(n \times n) * (f \times f) = (\frac{n + 2p - f}{s} + 1) \times (\frac{n + 2p - f}{s} + 1) = (\frac{5 + (2 \times 1) - 3}{1} + 1) \times (\frac{5 + (2 \times 1) - 3}{1} + 1) = 5 \times 5$$

Note if we get non integer value for our output size, we found it down to the nearest integer



MODERN COMPUTER VISION

BY RAJEEV RATAN

Next...

Activation Layer ReLU