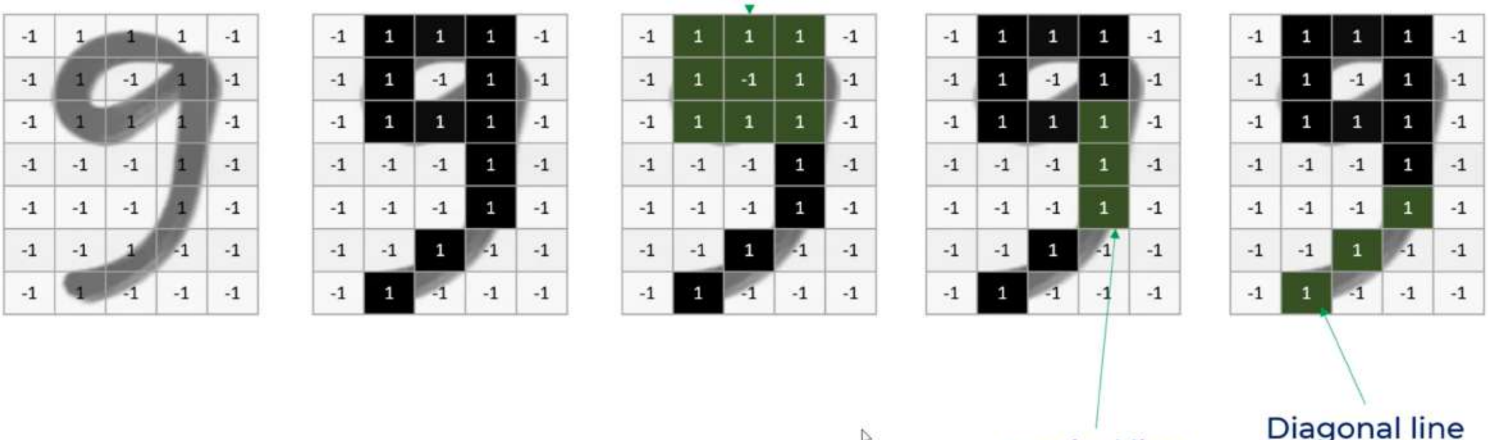
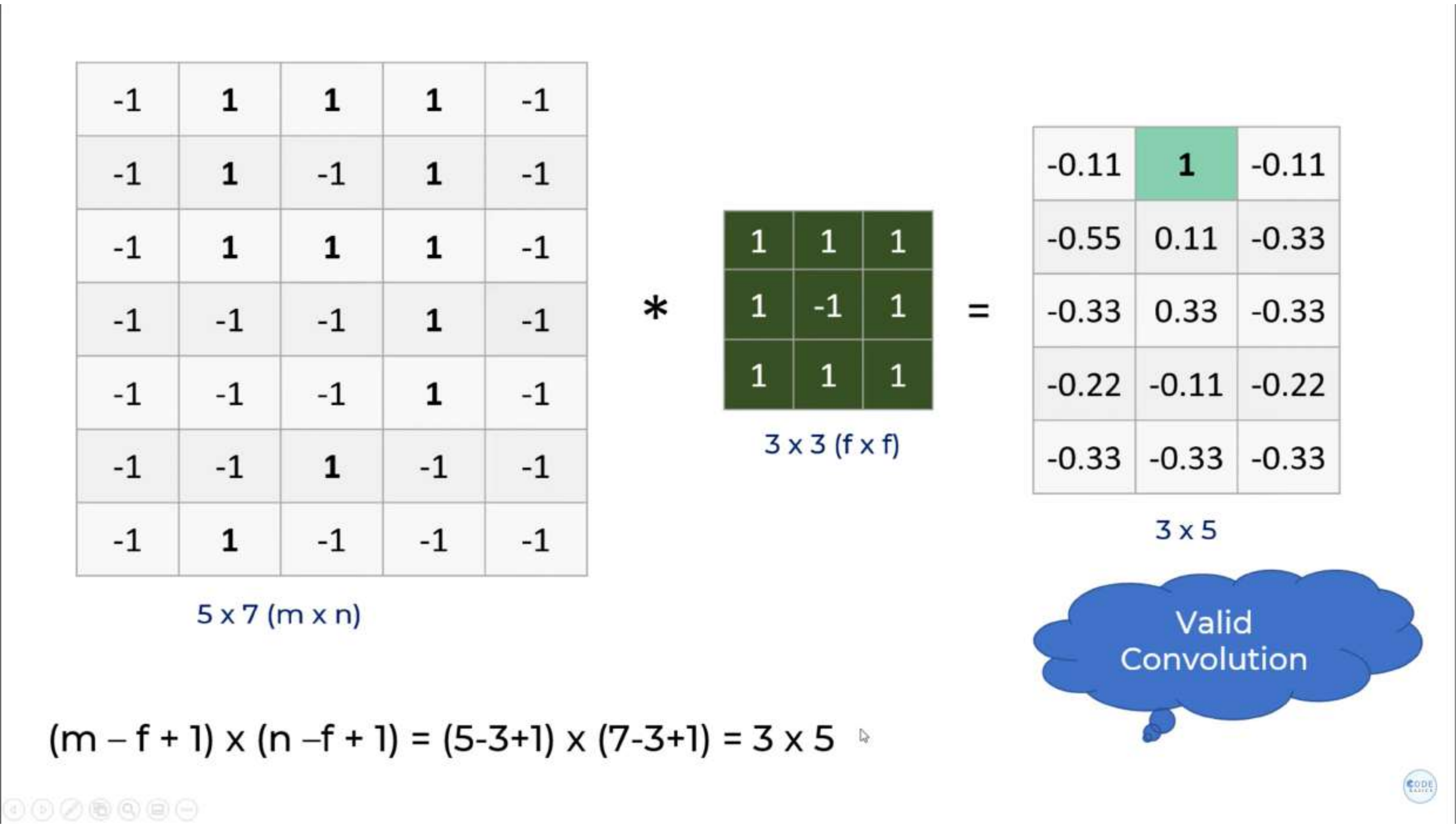


Loopy pattern filter

Vertical line filter

Diagonal line filter



-1	**1**	**1**	**1**	-1
-1	**1**	-1	**1**	-1
-1	**1**	**1**	**1**	-1
-1	-1	-1	**1**	-1
-1	-1	-1	**1**	-1
-1	-1	**1**	-1	-1
-1	**1**	-1	-1	-1

5 x 7 (m x n)

*

1	1	1
1	-1	1
1	1	1

3 x 3 (f x f)

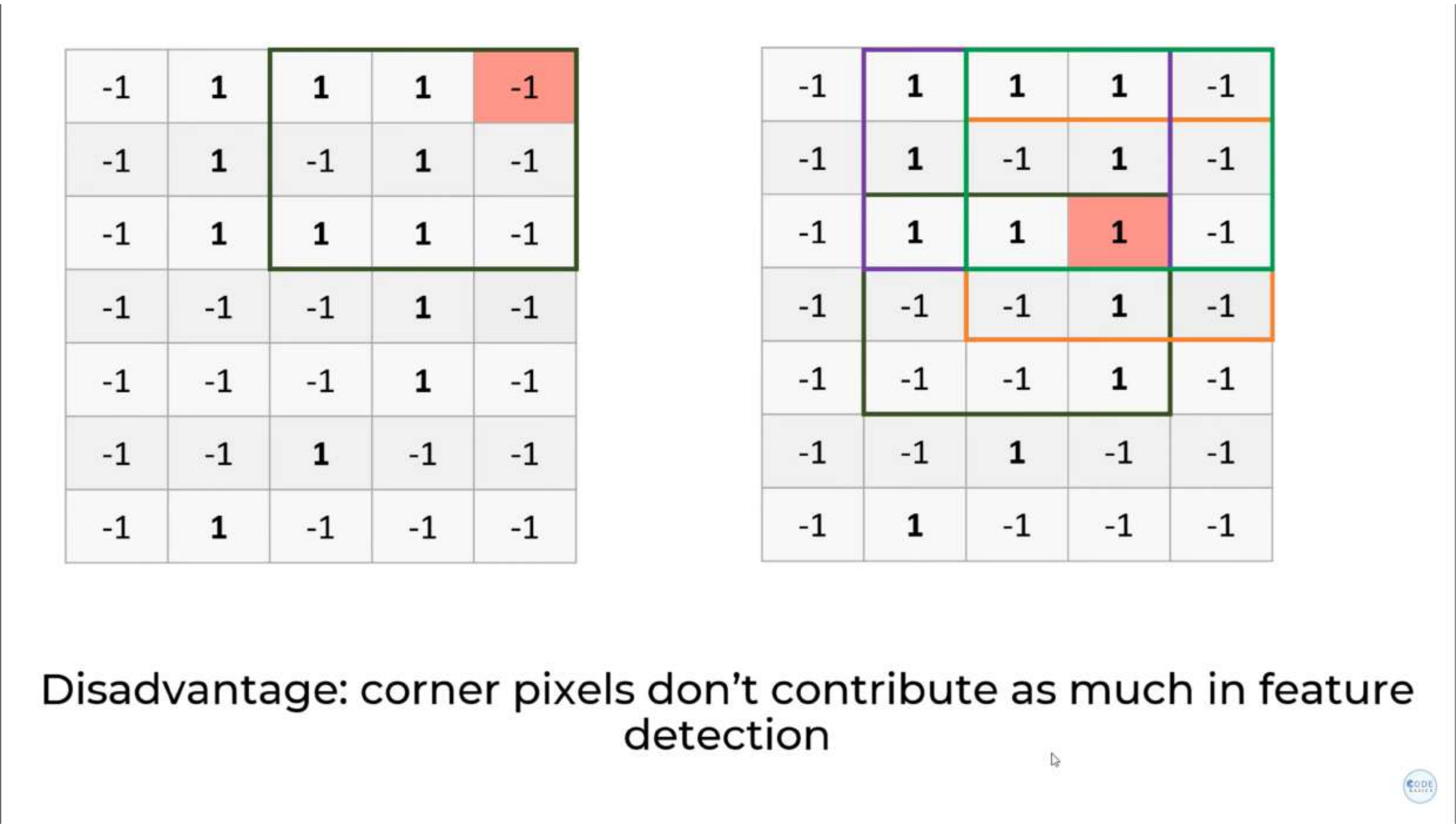
=

-0.11	**1**	-0.11
-0.55	0.11	-0.33
-0.33	0.33	-0.33
-0.22	-0.11	-0.22
-0.33	-0.33	-0.33

3 x 5

Valid Convolution

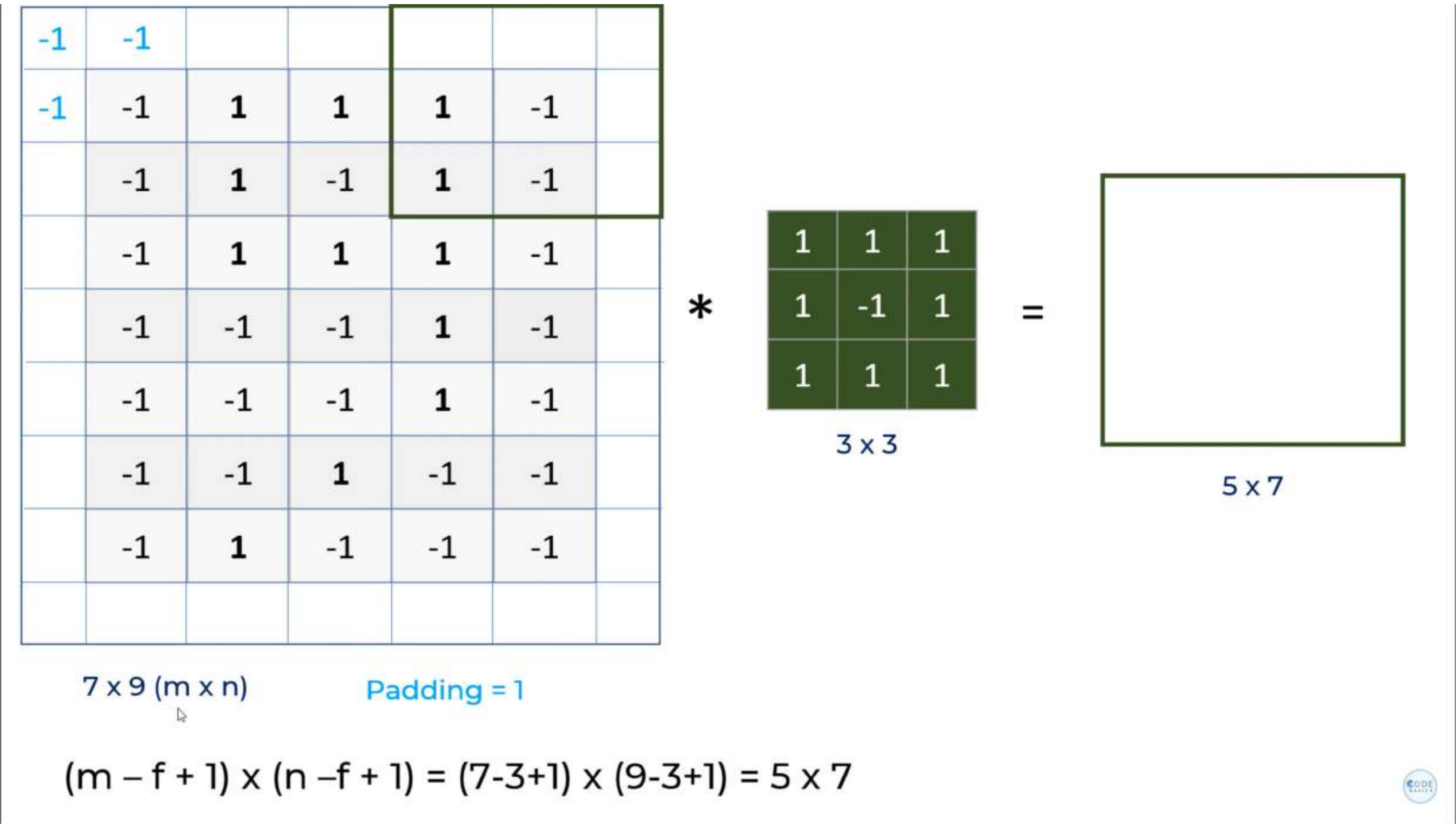
(m – f + 1) x (n –f + 1) = (5-3+1) x (7-3+1) = 3 x 5



-1	1	1	1	-1
-1	1	-1	1	-1
-1	1	1	1	-1
-1	-1	-1	1	-1
-1	-1	-1	1	-1
-1	-1	1	-1	-1
-1	1	-1	-1	-1

-1	1	1	1	-1
-1	1	-1	1	-1
-1	1	1	1	-1
-1	-1	-1	1	-1
-1	-1	-1	1	-1
-1	-1	1	-1	-1
-1	1	-1	-1	-1

Disadvantage: corner pixels don't contribute as much in feature detection



-1	-1					
-1	-1	**1**	**1**	**1**	-1	
	-1	**1**	-1	**1**	-1	
	-1	**1**	**1**	**1**	-1	
	-1	-1	-1	**1**	-1	
	-1	-1	-1	**1**	-1	
	-1	-1	**1**	-1	-1	
	-1	**1**	-1	-1	-1	

7 x 9 (m x n)

Padding = 1

*

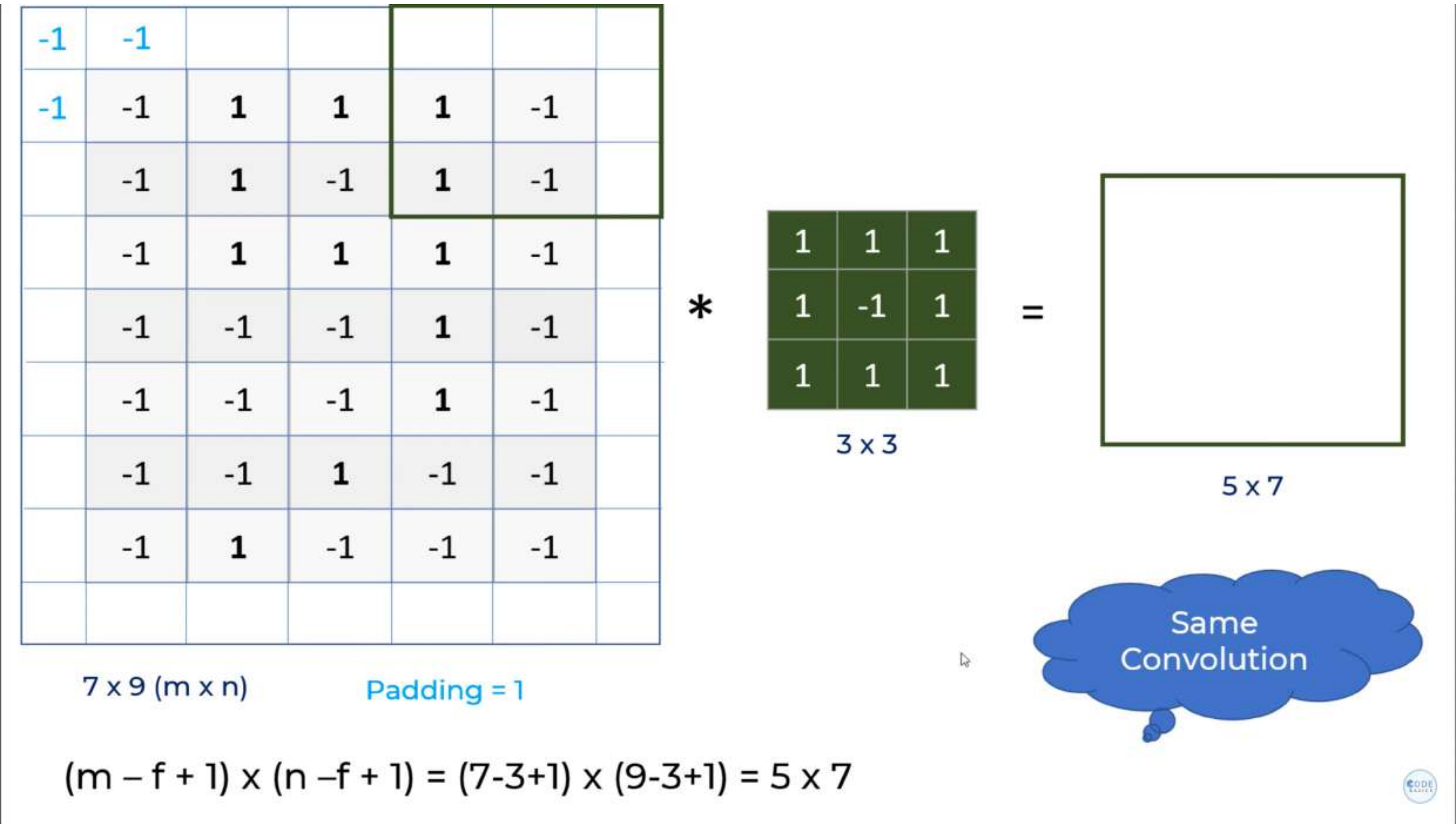
1	1	1
1	-1	1
1	1	1

3 x 3

=

5 x 7

$$(m - f + 1) \times (n - f + 1) = (7-3+1) \times (9-3+1) = 5 \times 7$$



-1	-1					
-1	-1	**1**	**1**	**1**	-1	
	-1	**1**	-1	**1**	-1	
	-1	**1**	**1**	**1**	-1	
	-1	-1	-1	**1**	-1	
	-1	-1	-1	**1**	-1	
	-1	-1	**1**	-1	-1	
	-1	**1**	-1	-1	-1	

7 x 9 (m x n)

Padding = 1

*

1	1	1
1	-1	1
1	1	1

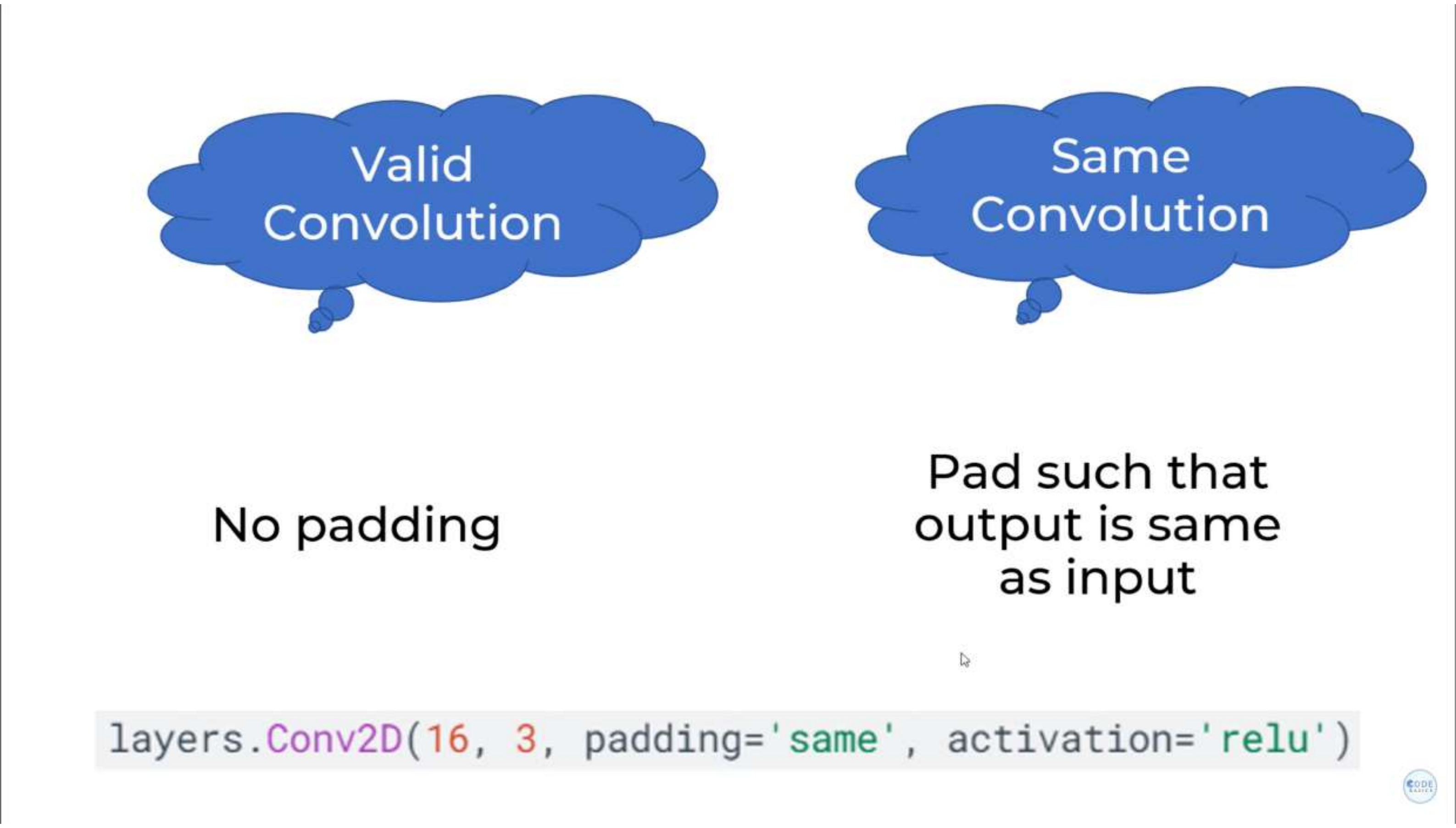
3 x 3

=

5 x 7

Same Convolution

$$(m - f + 1) \times (n - f + 1) = (7-3+1) \times (9-3+1) = 5 \times 7$$



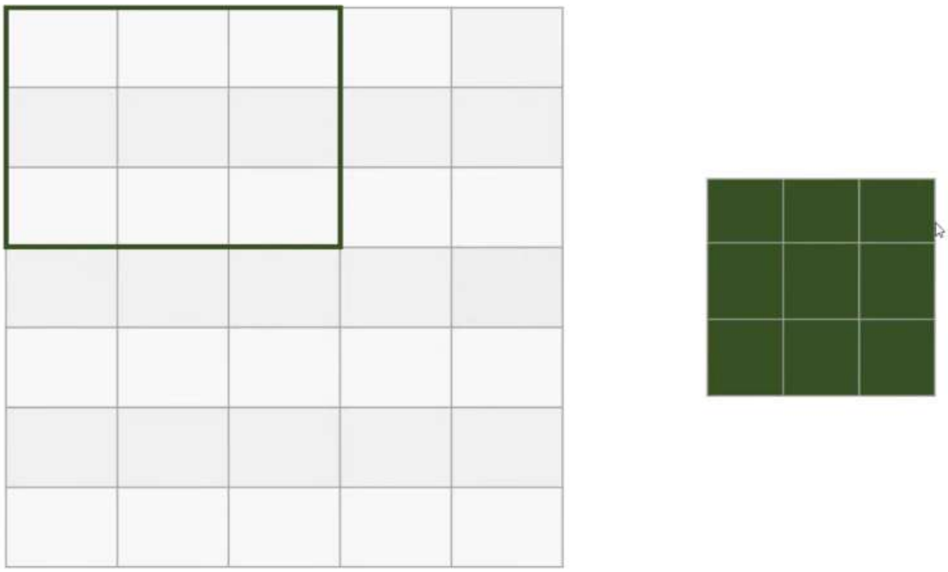
Valid Convolution

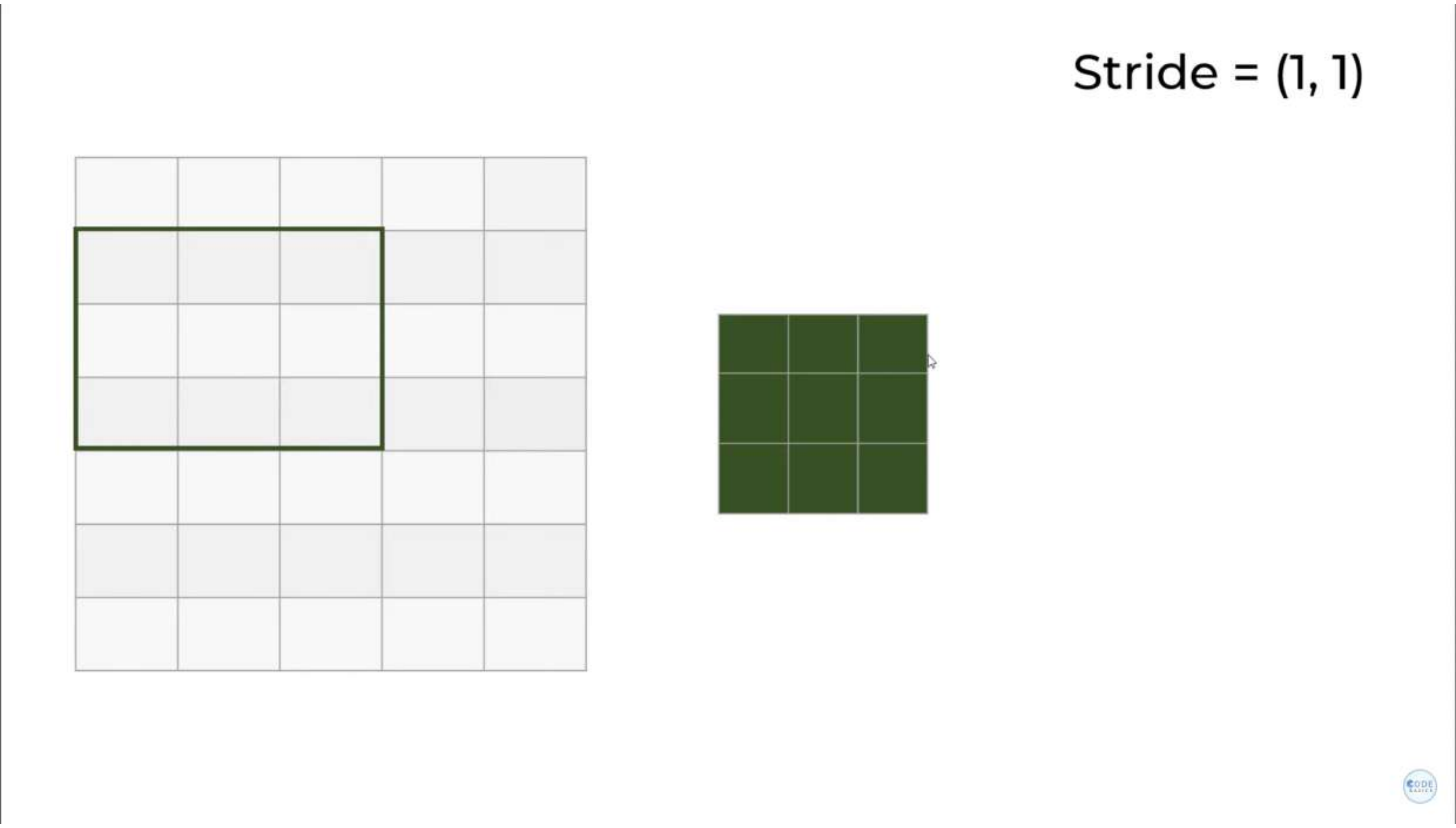
No padding

Same Convolution

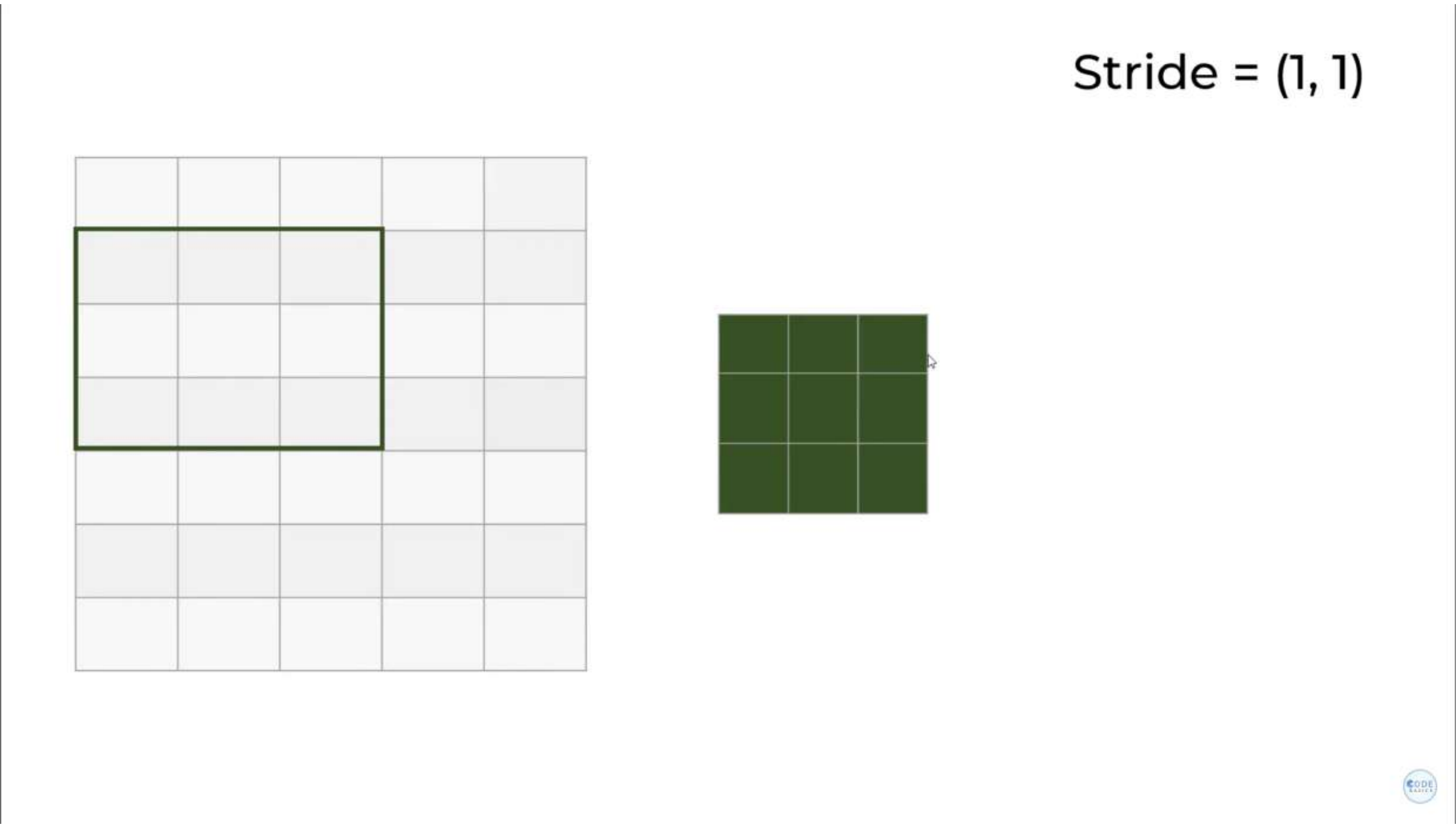
Pad such that output is same as input

```
layers.Conv2D(16, 3, padding='same', activation='relu')
```

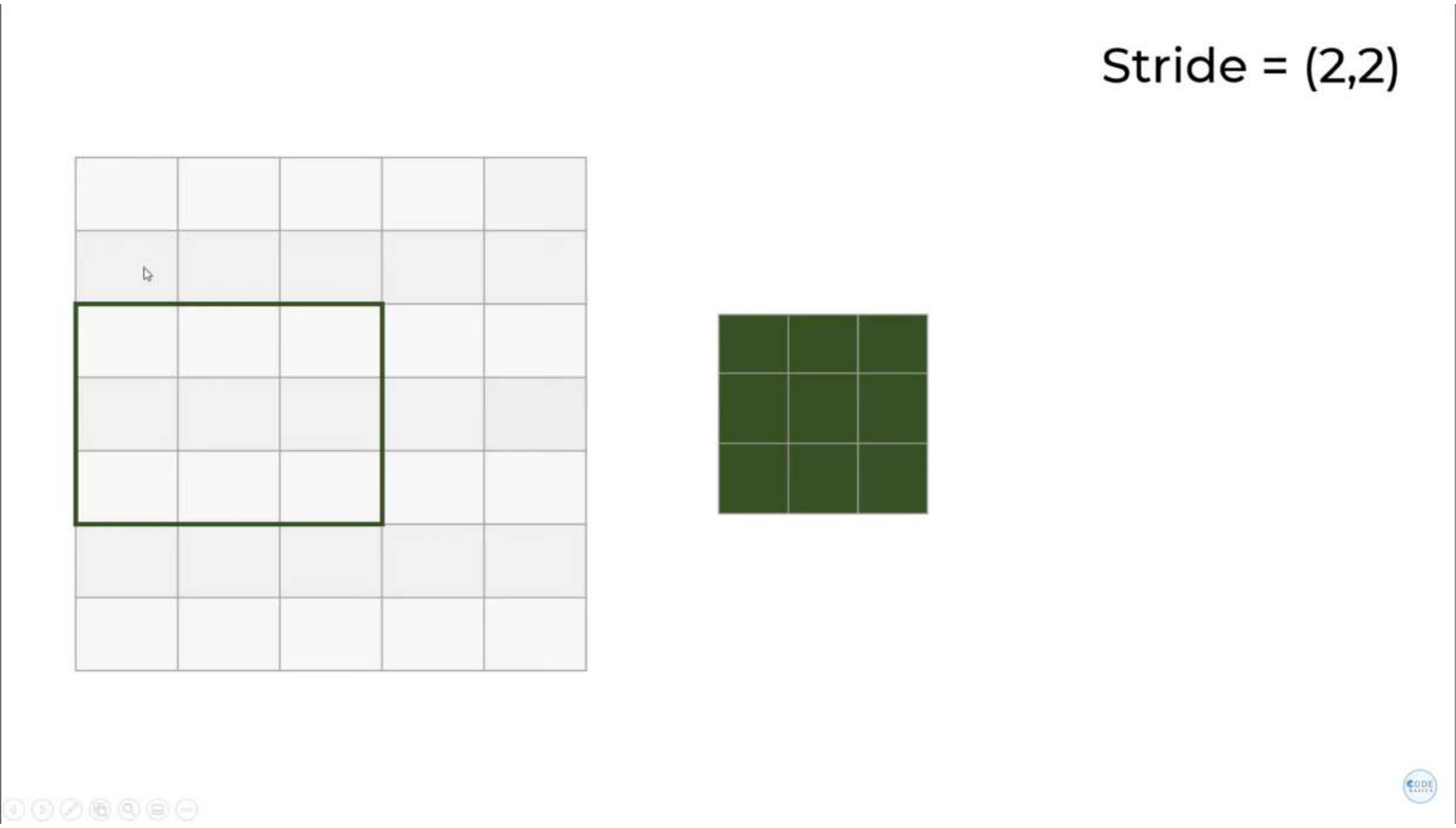




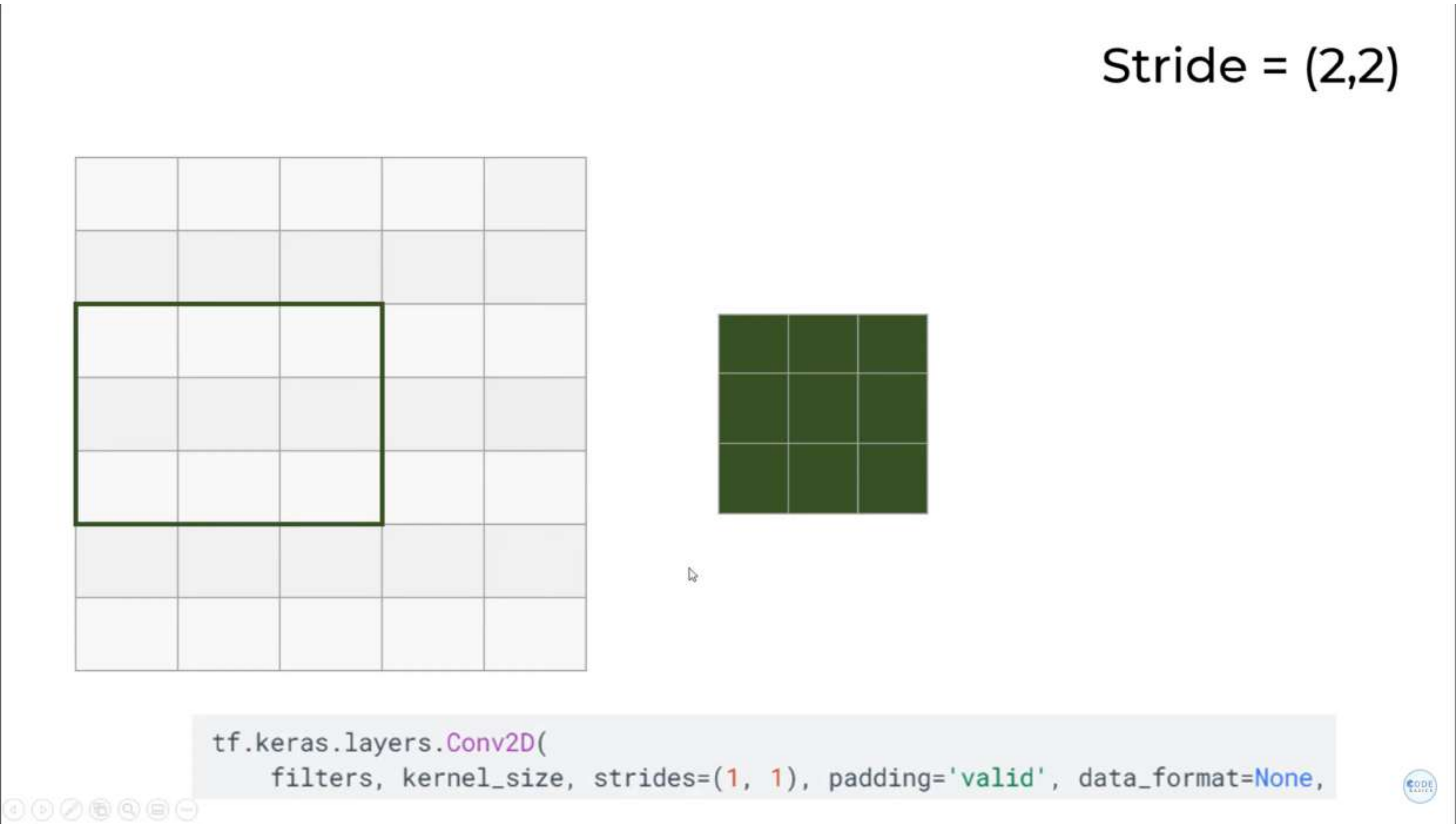
Stride = (1, 1)



Stride = (1, 1)



Stride = (2,2)



Stride = (2,2)

```
tf.keras.layers.Conv2D(
    filters, kernel_size, strides=(1, 1), padding='valid', data_format=None,
```