CNN

Name	Dharini Baskaran
Identity Key	dhba5060

	Level	Completed
	Beginner	16
	Intermediate	3
	Advanced	
	Expert	

Goal	
4722	16
5722	18
Total Completed	
19	

1 Cross-Correlation vs Convolution

Cross-correlation

2	0
-1	1

$\otimes$

2	1	2	5
0	4	3	0
0	1	2	1
6	0	1	0

2	0
-4	3

a

1

Convolution

2	0
-1	1

$*$

2	1	2	5
0	4	3	0
0	1	2	1
6	0	1	0

0	4
4	-3

b

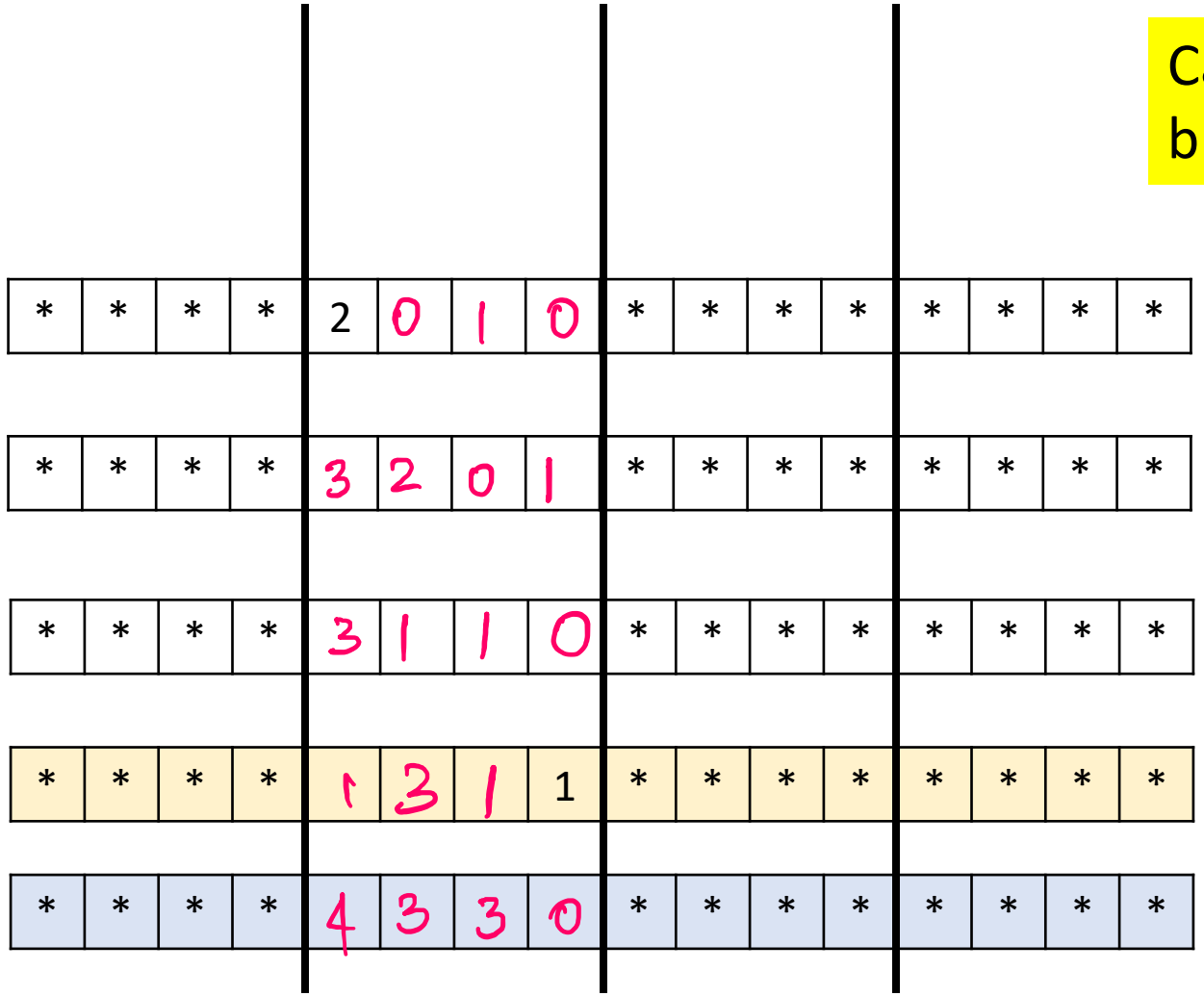
5

2 Calculate a Conv Layer (padding = 0)

2	0	0	2
1	3	1	1
3	2	0	1
1	1	0	4

1	2
-1	0

0	-1	2	1



Calculate all the blanks. Skip *'s.

*	*	*	*
4	3	3	0
*	*	*	*
*	*	*	*

3 X:2@3x4 | Conv:2@[2x2]x2 | Y:2@2x3

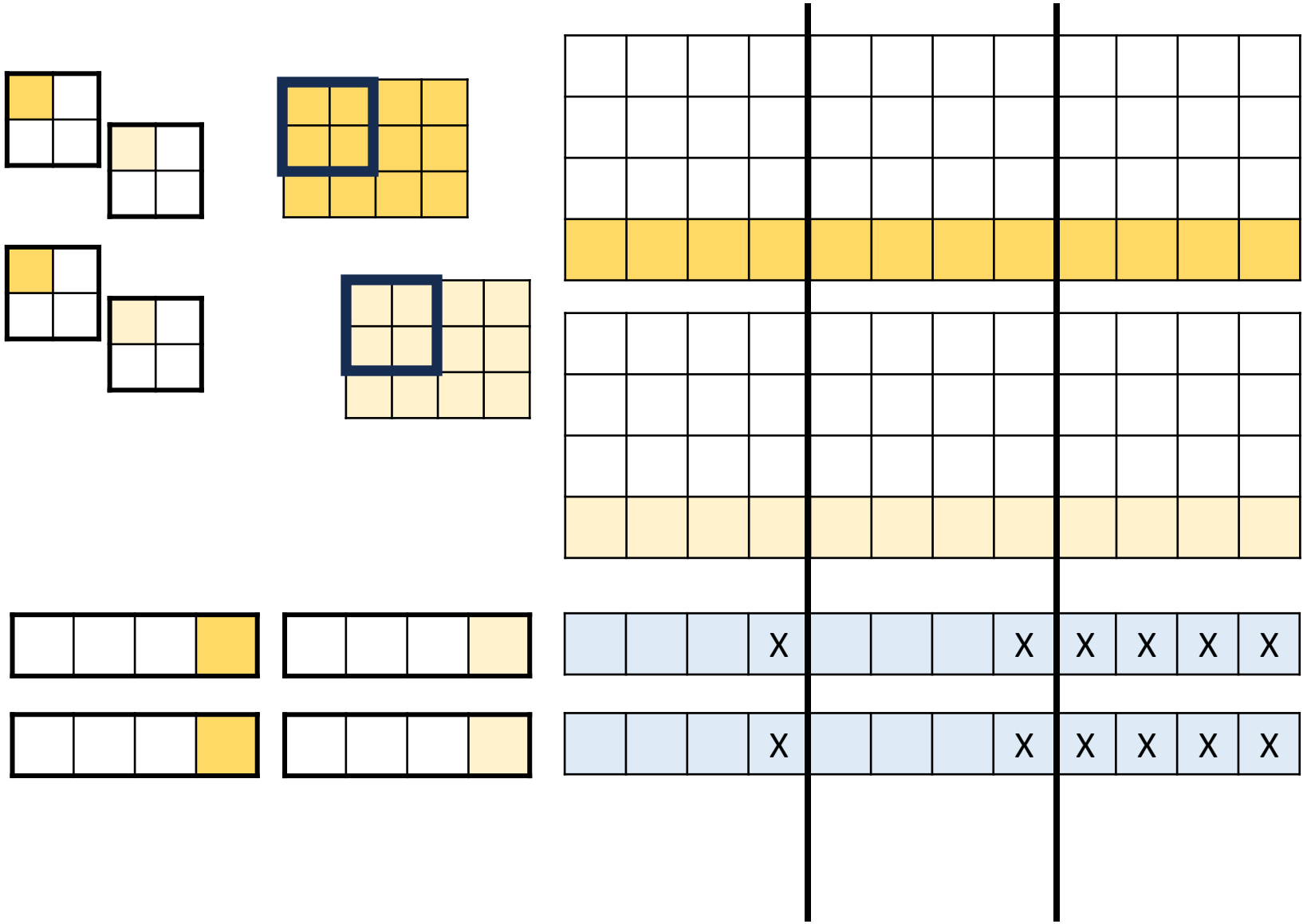
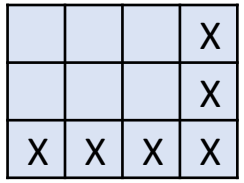
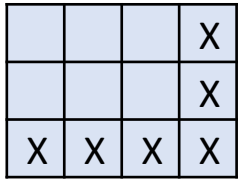


Figure out the dimensions. Fill the blanks __.





4 Calculate a Conv Layer (no padding: X)

Calculate all the blanks. Skip *'s.

2	0
1	0

0	2	1	1
3	1	1	0
0	4	0	2

1	2
-1	0

2	0	0	1
0	3	1	0
4	0	0	1

*	*	*	*	1	1	0	X	*	*	*	*
*	*	*	*	3	1	1	0	0	4	0	2

*	*	*	*	3	1	0	X	*	*	*	*
*	*	*	*	0	3	1	0	4	0	0	1

*	*	*	*
6	5	3	X
*	*	*	*

0	2
0	1

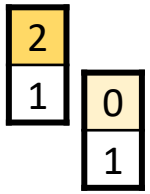
*	*	*	*	6	5	3	X	*	*	*	*

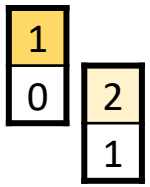
2	1
0	-1

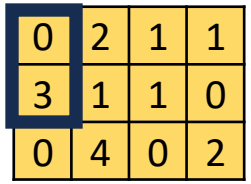
*	*	*	*	5	0	0	X	*	*	*	*

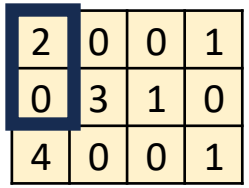
*	*	*	*
5	0	0	X
*	*	*	*

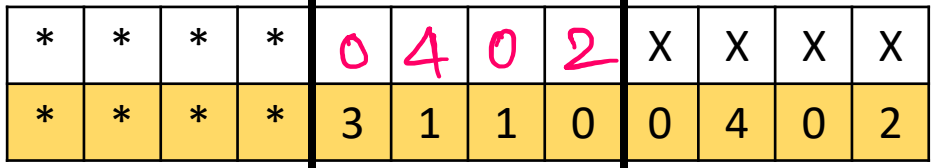
5 Calculate a Conv Layer

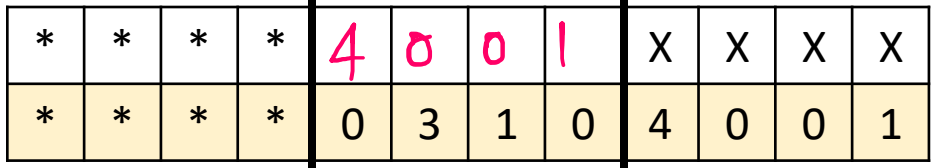


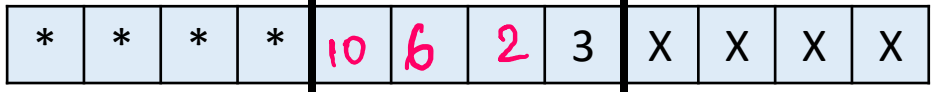


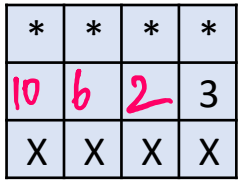


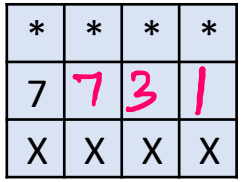






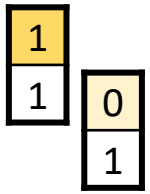


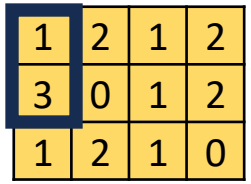


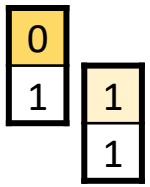


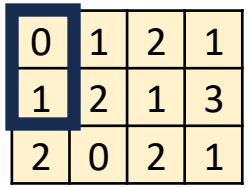
Calculate all the blanks. Skip *'s.

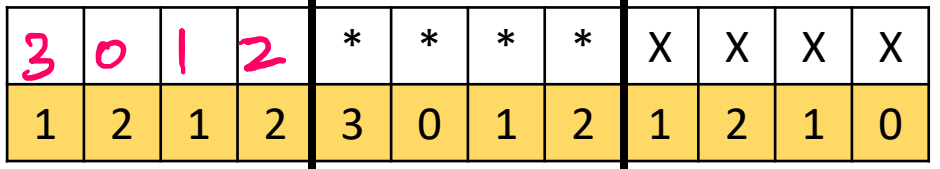
6 Calculate a Conv Layer

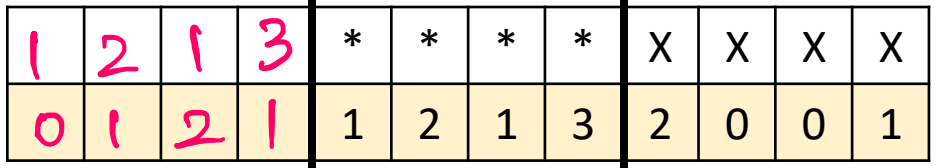


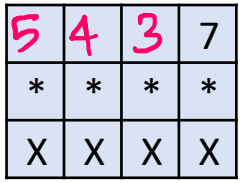


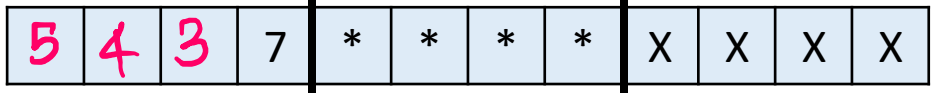


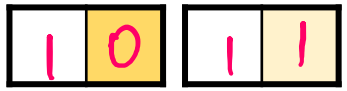


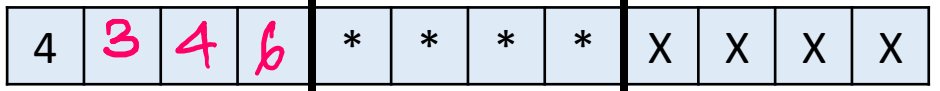


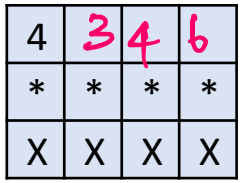












Calculate all the blanks. Skip *'s.

7 Calculate a Conv Layer

1	0
0	1

1	1
0	0

0	2	1	1
3	1	1	0
0	4	0	2

1	0
0	-1

0	0
0	1

2	0	0	1
0	3	1	0
4	0	0	1

1	1	0	*	*	*	*	*	*	*	*	*
3	1	1	*	*	*	*	*	*	*	*	*
2	1	1	*	*	*	*	*	*	*	*	*
0	2	1	1	3	1	1	0	0	4	0	2

3	1	0	*	*	*	*	*	*	*	*	*
0	3	1	*	*	*	*	*	*	*	*	*
0	0	1	*	*	*	*	*	*	*	*	*
2	0	0	1	0	3	1	0	4	0	0	1

1	0	0	1

0	0	1	1

3	3	2	X	*	*	*	X	X	X	X	X
2	2	1	X	*	*	*	X	X	X	X	X

-1	0	0	1

1	0	0	0

Calculate all the blanks. Skip *'s.

	3	2	X
*	*	*	X
X	X	X	X

2	2	1	X
*	*	*	X
X	X	X	X

8 ✓ ○ Stride: (1, 2) | sum

2	4	2	5
1	3	1	1
2	2	4	1
1	3	3	2

Subsample ↓

10	9
8	7
8	10

1	1	2	1	3	2
3	1	2	1	1	3
4	5	3	5	2	1
2	2	1	2	2	4

sum

10	9	8	7	8	10

9 Stride: (2, 2) | sum

2	4	2	5
1	3	2	1
2	2	4	1
1	3	3	2

3	1	3	2
1	2	1	3
4	5	2	1
2	2	2	4

Subsample ↓

sum

10	10	8	10

10	10
8	10

10

Calculate Padding

Padding: Same Size; Repeat

Calculate all the blanks. Skip *'s.

2	2	1	3
0	3	0	1
5	2	4	2
6	1	0	3

*	*	*
*	*	*
*	*	*

4	2	2	2	0	3	3	3	*	*	*	*	*	*	*	*
2	4	2	2	1	0	3	3	*	*	*	*	*	*	*	*
5	2	4	2	6	1	0	3	*	*	*	*	*	*	*	*
0	1	1	1	4	2	2	2	0	3	3	3	0	3	3	3
3	0	1	1	2	4	2	2	1	0	3	3	1	0	3	3
0	3	0	1	5	2	4	2	6	1	0	3	6	1	0	3
1	3	3	3	0	1	1	1	4	2	2	2	0	3	3	3
2	1	3	3	3	0	1	1	2	4	2	2	1	0	3	3
2	2	1	3	0	3	0	1	5	2	4	2	6	1	0	3

*	*	*	*	*	*	*	*	*

*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

11

Input: (height x 8, width x 6, depth x 5)

(Padding = Same)

X:1@4x4 | Conv:1@[2,2]x1 | Y:2@4x4 =>

X: 5 @ 32 x 24 | Conv: 1 @[2 , 2]x 5 | Y: 1 @ 32 x 24

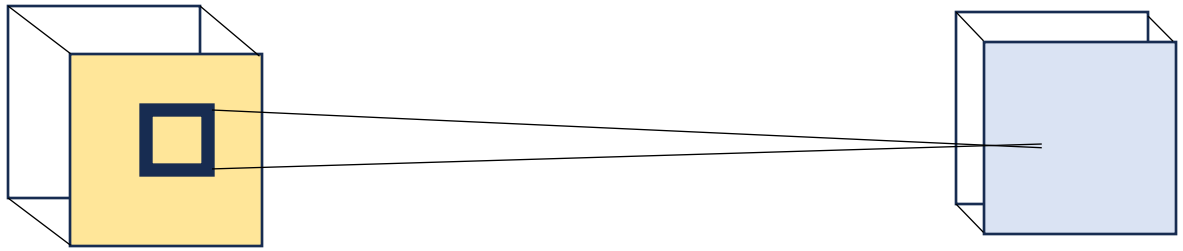
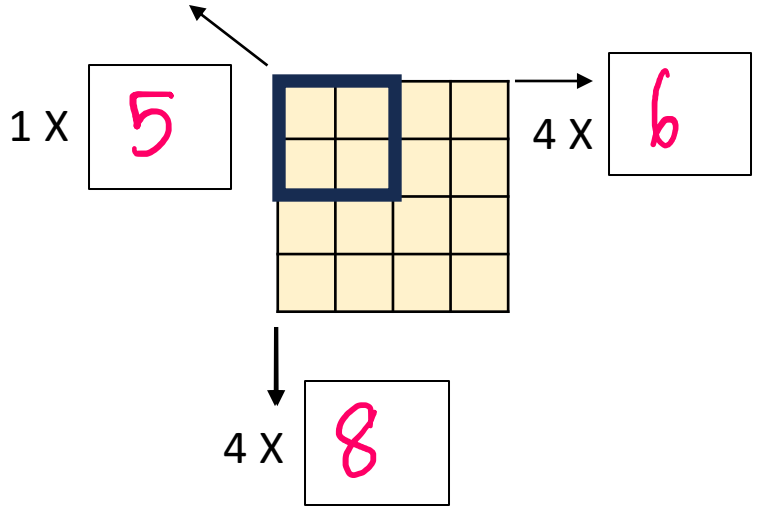


Figure out the dimensions. Fill the blanks __.



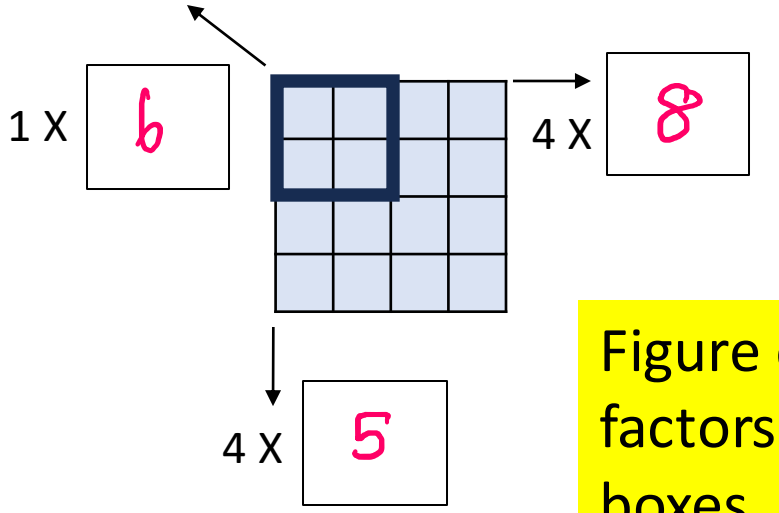


Figure out scaling factors. Fill the boxes.

12 

# of filters x 12, # of input maps x 16

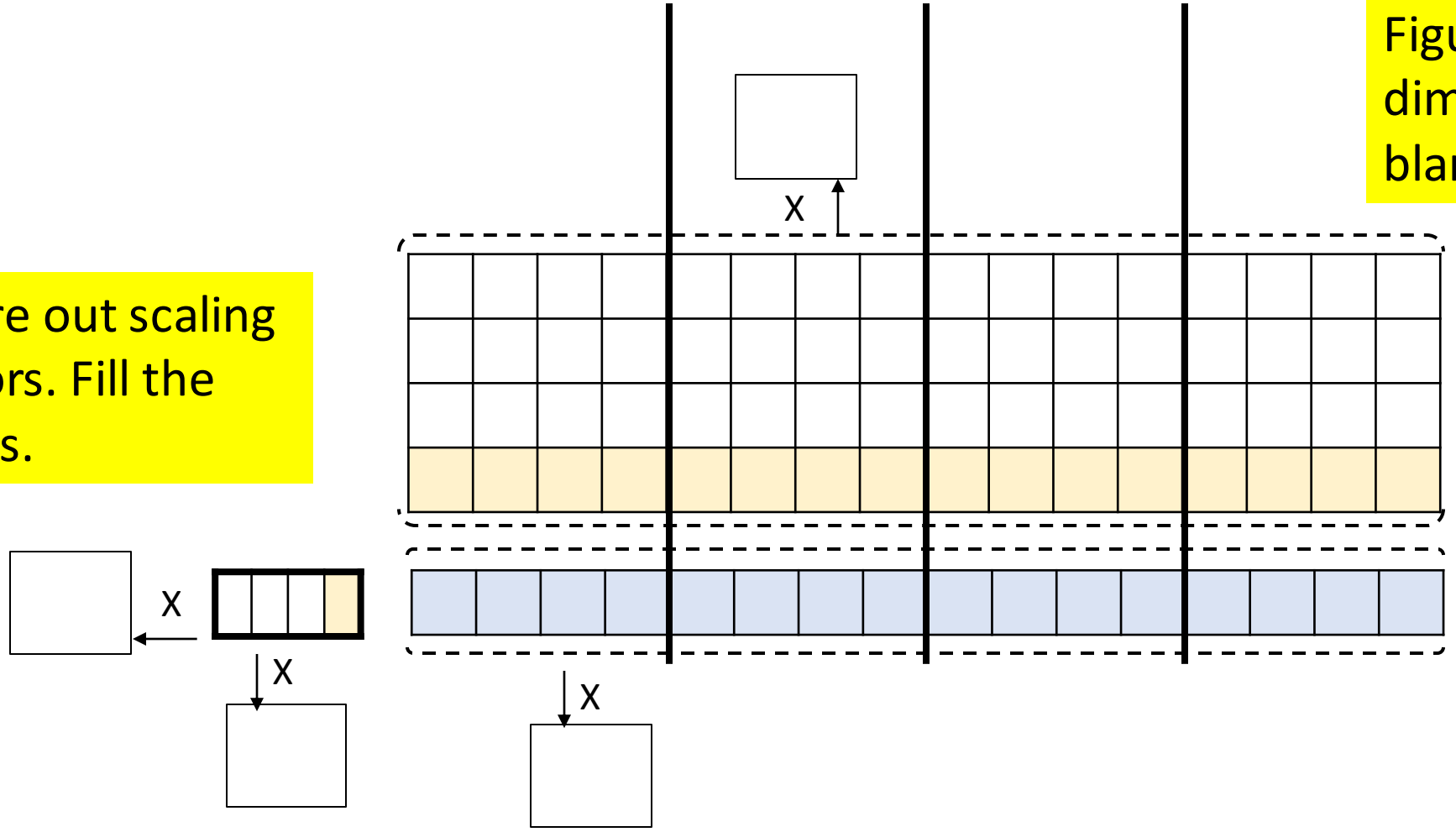
(Padding = Same)

X:1@4x4 | Conv:1@[2,2]x1 | Y:1@4x4 =>

X:16@4x4 | Conv:12@[2,2]x16 | Y:12@4x4

Figure out the dimensions. Fill the blanks __.

Figure out scaling factors. Fill the boxes.



13

# of filters x 512, # of input maps x 128

(Padding = Same)

X:1@4x4 | Conv:1@[2,2]x1 | Y:1@4x4 =>

X:128@4x4 | Conv:512@[2,2]x128 | Y:512@4x4

Figure out the dimensions. Fill the blanks __.

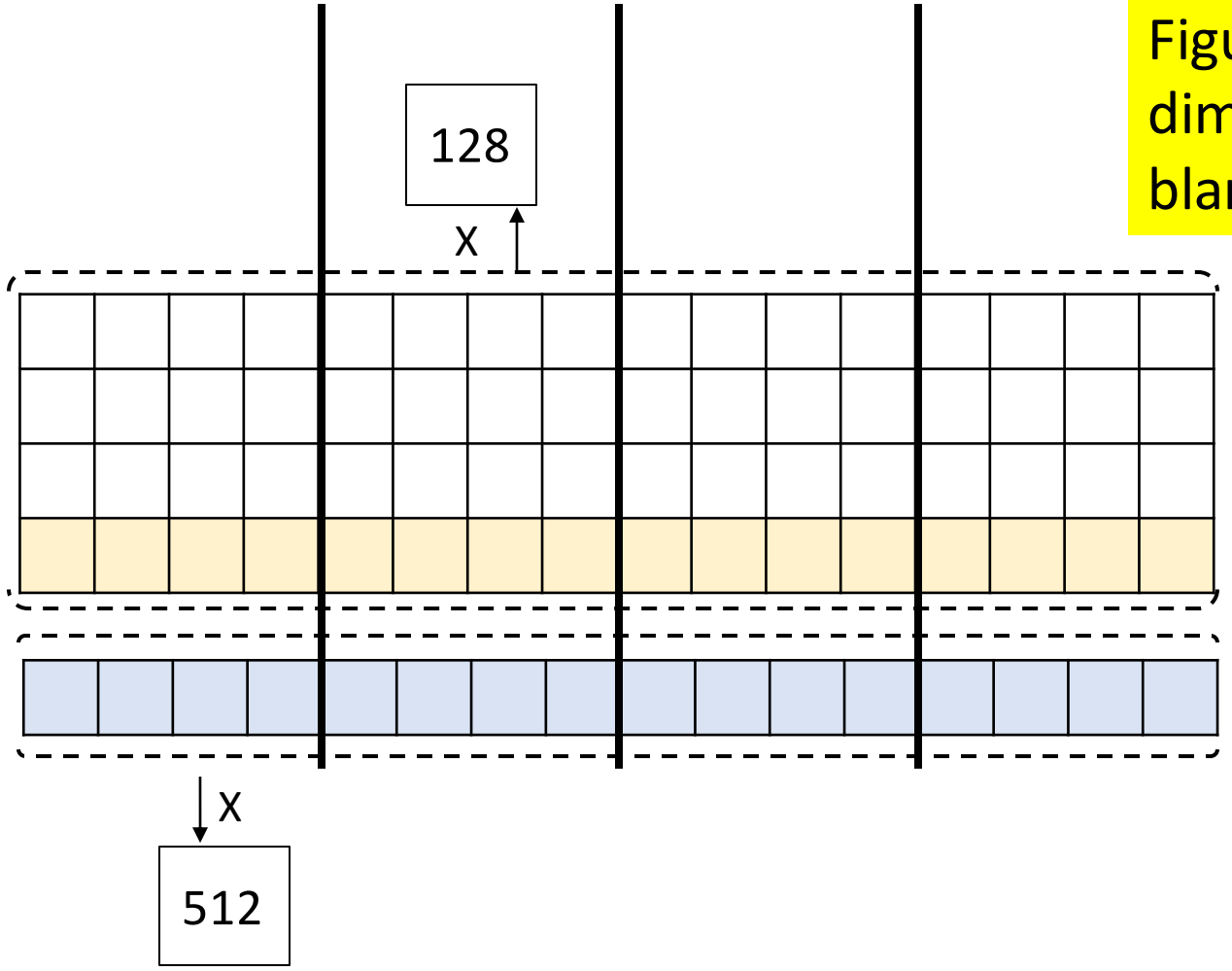
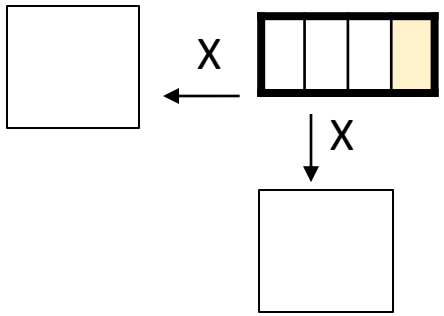


Figure out scaling factors. Fill the boxes.



14

Filter size (Width+1, Height+1)

(Padding = Same)

X:1@4x4 | Conv:1@[2,2]x1 | Y:1@4x4 =>

X:__1__@__5__x__5__ | Conv:__1__@[__1__,__2__]x__2__ | Y:__1__@__5__x__5__

Figure out how more cols and rows to grow.

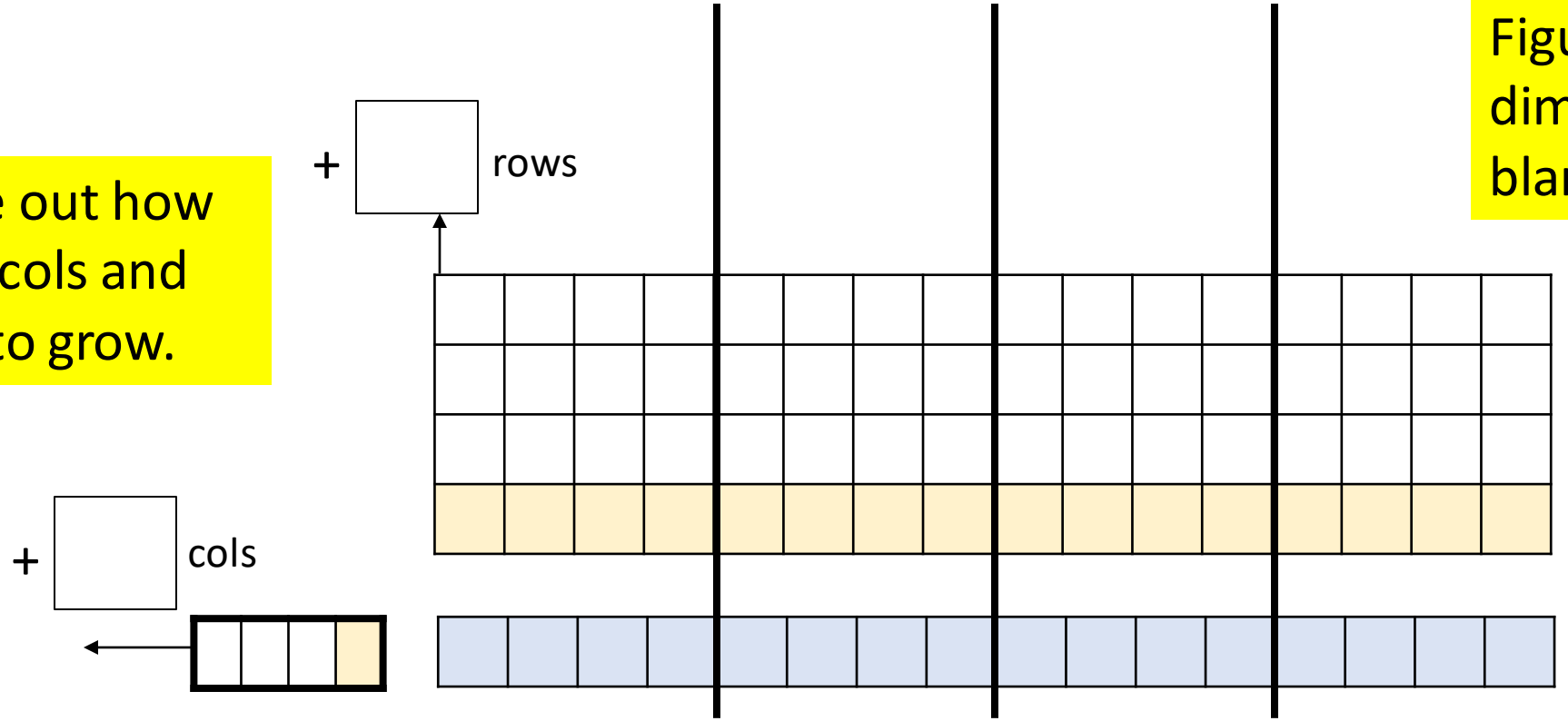


Figure out the dimensions. Fill the blanks __.

Trainable Parameters

of Trainable Parameters

X:6@5x5 | Conv: 2 @[2,2]x 6 | Y:2@5x5 => 2(2.2.6 +1) = 50

X:2@4x4 | Conv: 5 @[2,3]x 2 | Y:5@4x4 => 5(2.3.2 +1) = 65

# of Trainable Parameters

X:2@4x4 | Maxpool | Y:2@4x4 => 0

X:2@4x4 | Meanpool | Y:2@4x4 => 0

16

# of filters x 3, input: (Width x 5, Depth x 4)

(Padding = Same)

X:1@4x4 | Conv:1@[2,2]x1 | Y:1@4x4 =>

X:4@4x20 | Conv:3@[2,2]x4 | Y:3@4x20

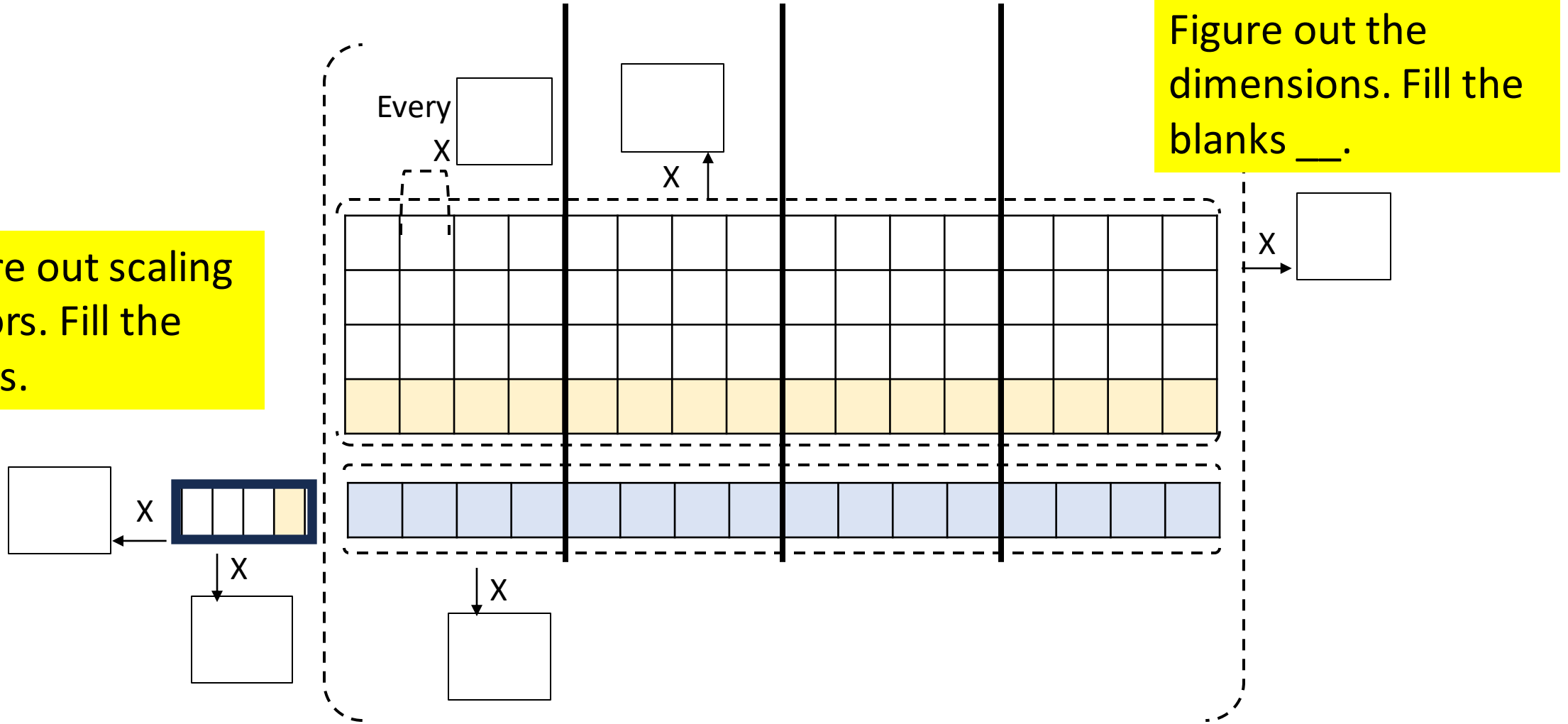


Figure out the dimensions. Fill the blanks __.

Figure out scaling factors. Fill the boxes.

17 ☐ ☐ # of filters x ___, input: (Height x ___, Width x ___, Depth x ___)

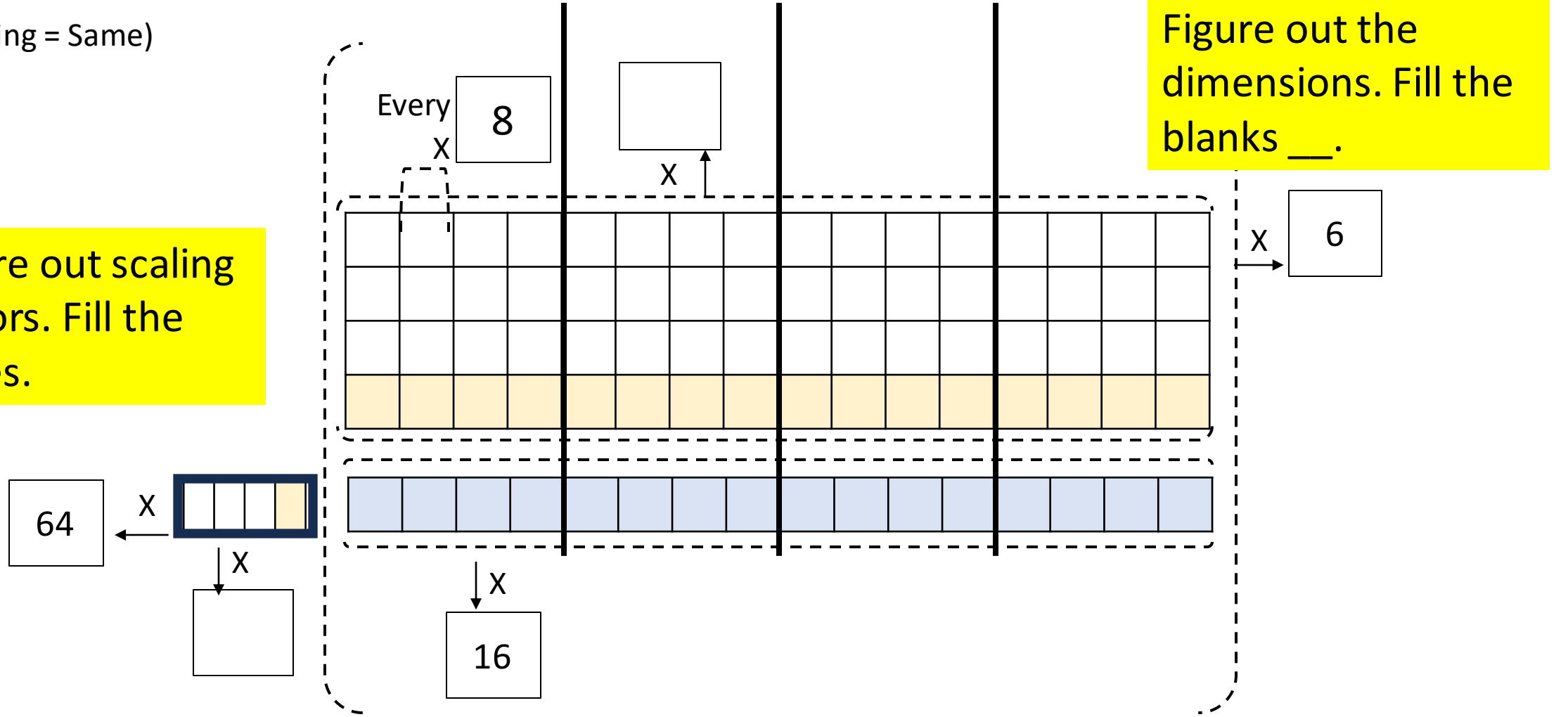
X:1@4x4 | Conv:1@[2,2]x1 | Y:1@4x4 =>

X:____@____x____ | Conv:___@[___,___]x____ | Y:____@____x____

(Padding = Same)

Figure out the dimensions. Fill the blanks __.

Figure out scaling factors. Fill the boxes.



18 

X: 2@4x4 | max pool:[2x2],s:(2,2) | Y: 2@2x2

2	0	3	0
3	1	0	1
0	4	1	0
0	0	0	0

0	1	2	0
1	0	2	1
3	0	1	1
6	0	0	0

3	3
4	1

1	2
6	1

1	1	0	0
3	0	0	0
0	0	4	0
2	3	0	1

0	1	0	0
1	2	6	0
1	0	0	1
0	2	3	1

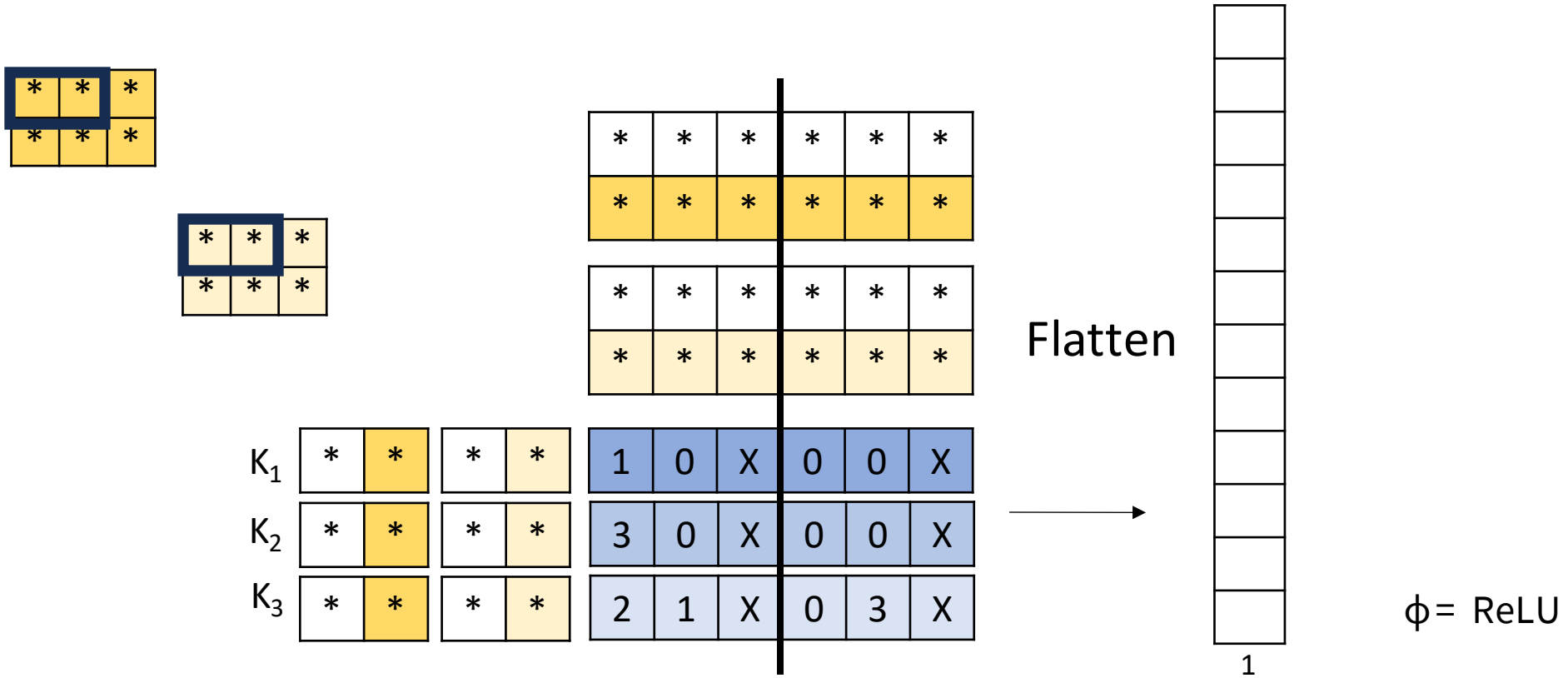
3	3	4	1

1	2	6	1

19

+ FC Layer for multi-class classification (c= 3)

X:2@3x3 | Conv:3@[1,2]x2 | H:3@2x2 | FC: 12 -> 3 | Y: 3 x 1



20

X:2 @ 4 x 4 | Conv: 3 @ [2,2] x 2 s:(2,2) | H1: 3 @ 2 x 2 | Maxpool s:(1,1) | H2: 1 @ 3 x 2 | FC: 6 -> 1 | Y: 3 x 1