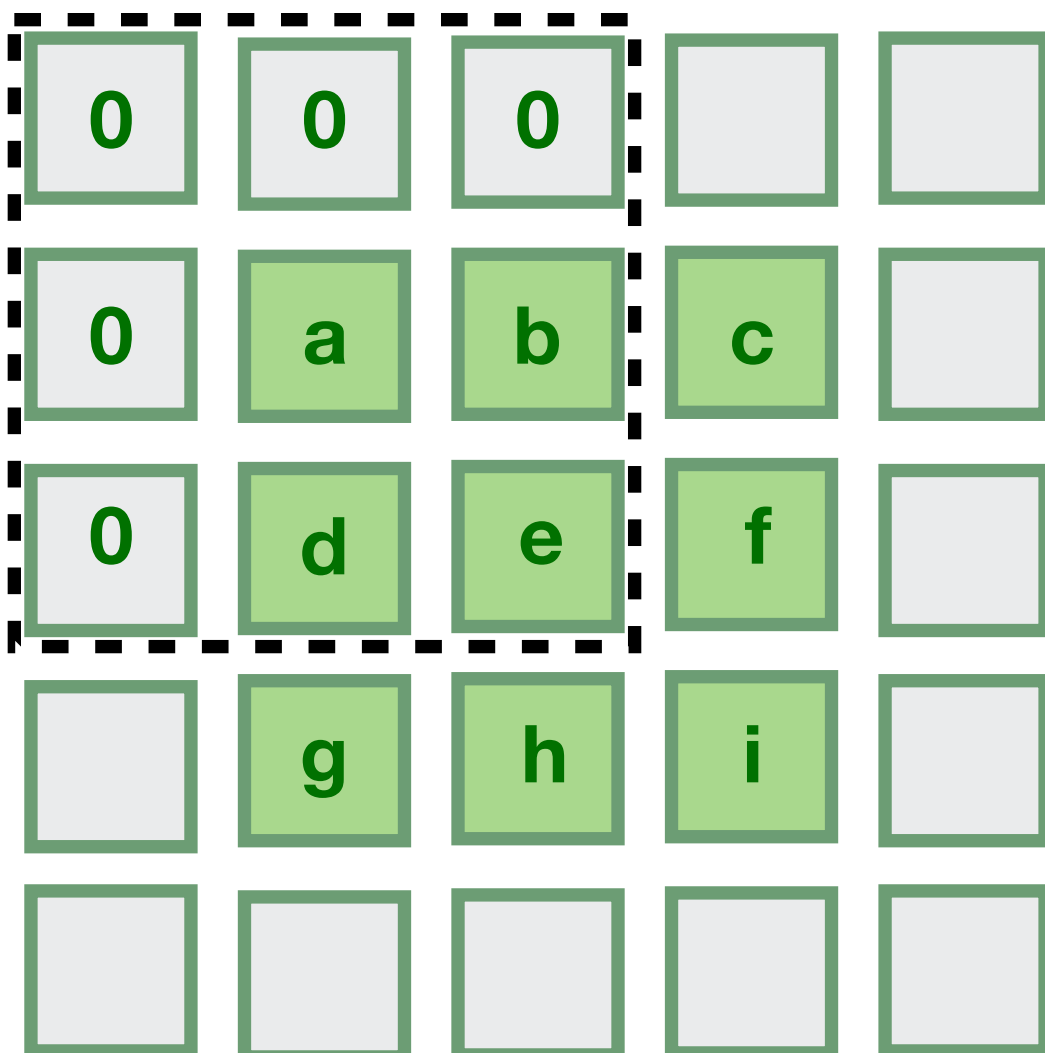


Input. InY=3, InX=3 InC=2

ic=0



Example:

KnY=3, KnX=3, KnC=2

Px=Py=1

Sx=Sy=1

Assume BS=1

Therefore:

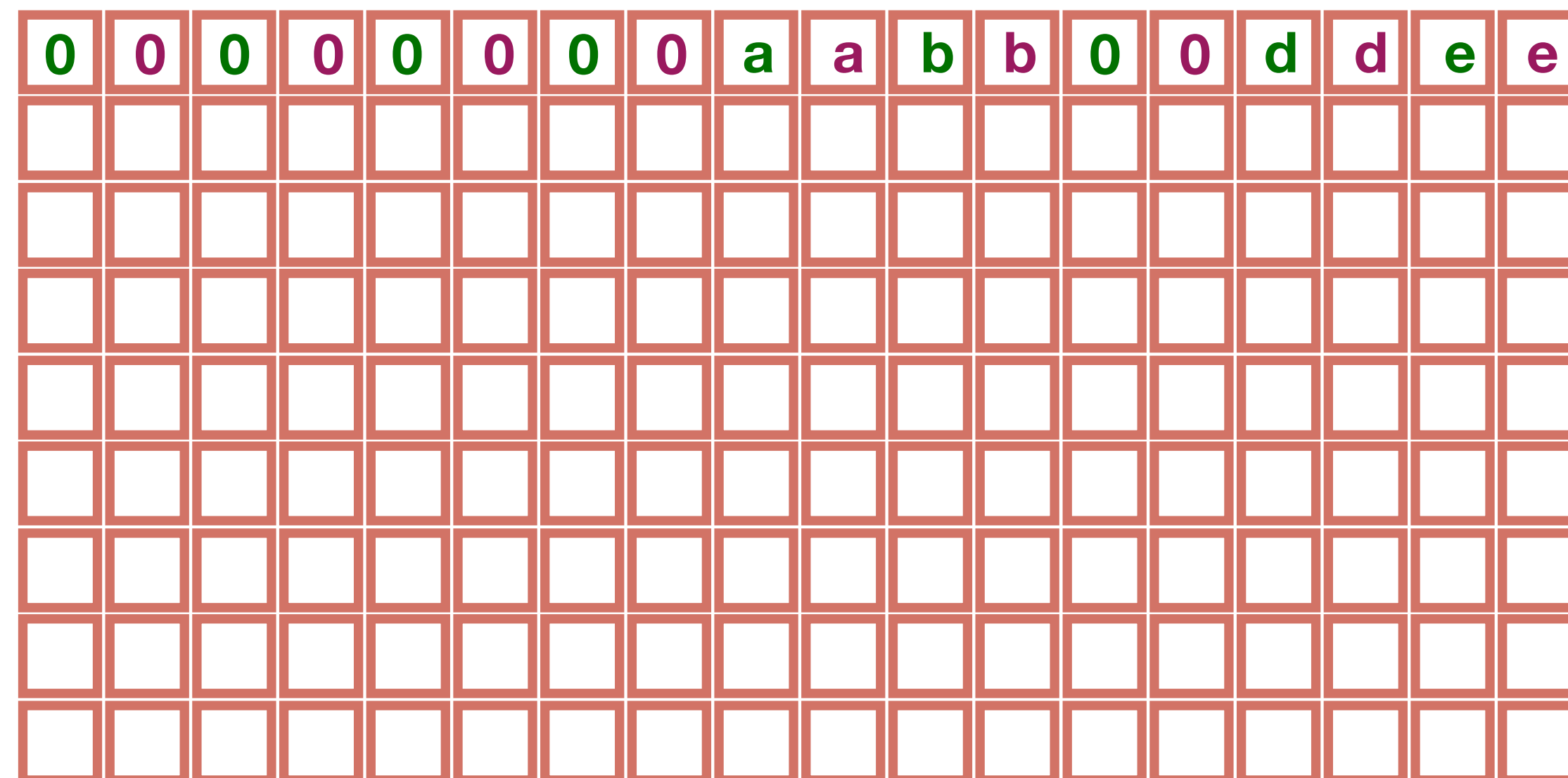
$OpY = (InY - KnY + 2Py) / Sy + 1 = 3$

$OpX = (InX - KnX + 2Px) / Sx + 1 = 3$

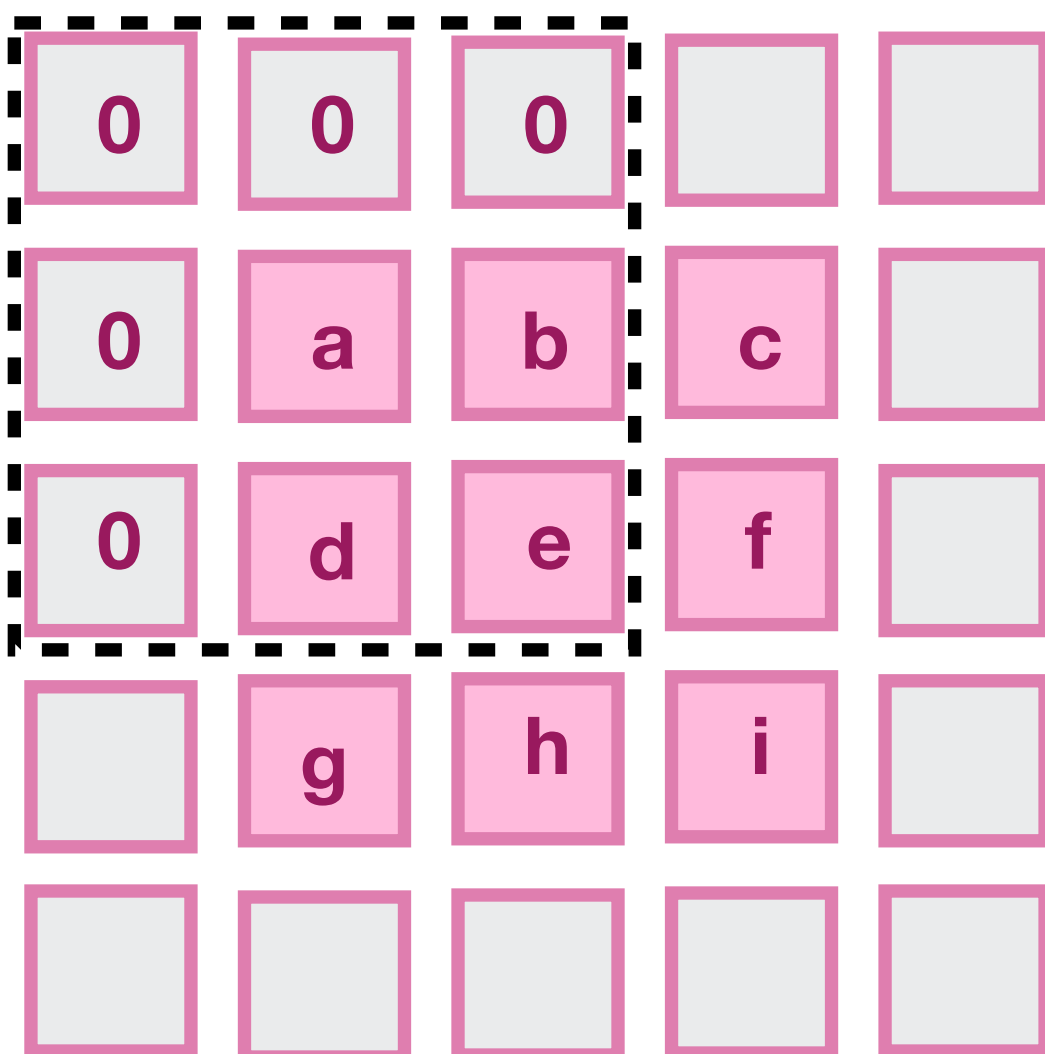
$OpC = KnC$

Output of Im2Col is a matrix:

$(BS * 3 * 3) \times (3 * 3 * 2)$



ic=1



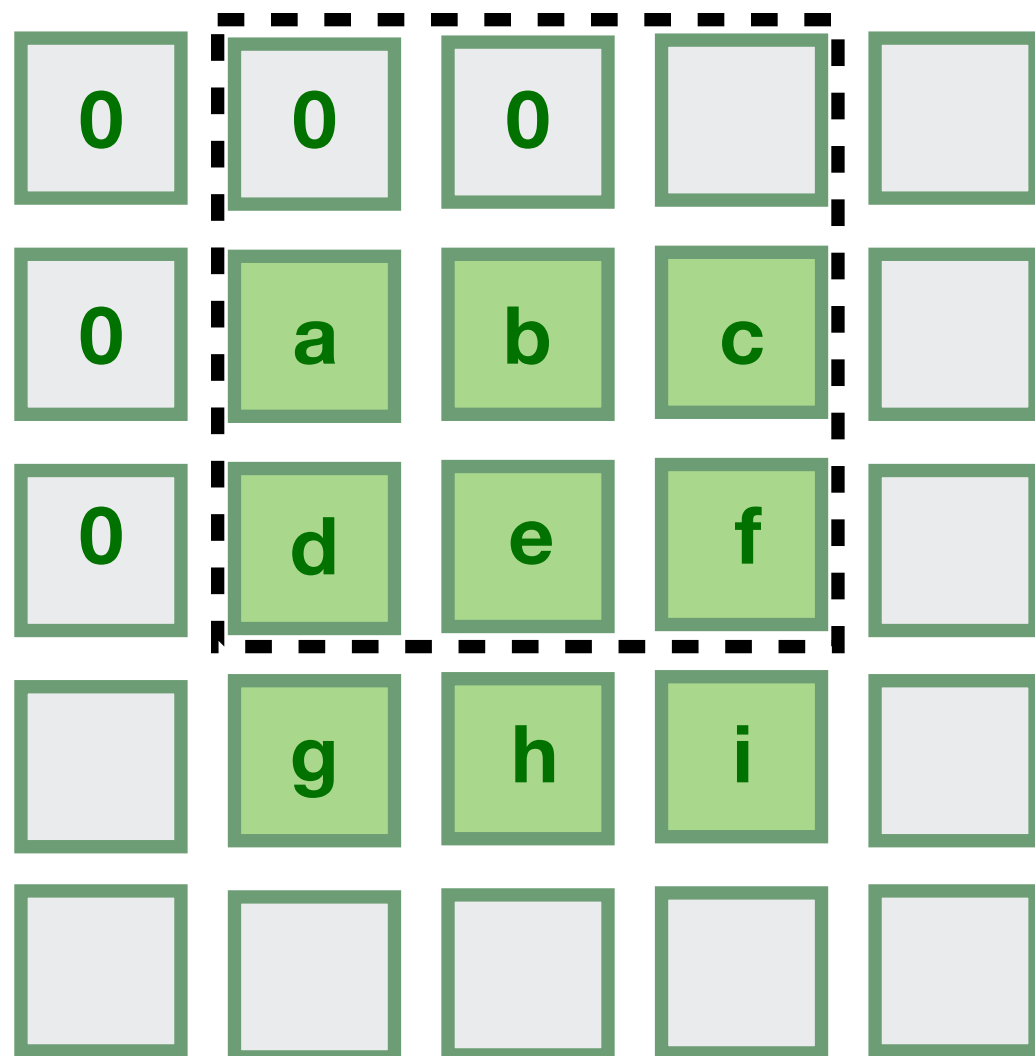
Kernel, dashed line on the left, takes $3 * 3 * 2$ inputs at the time, im2col lays them out in a row

Indices from slow to fast are:

- batch id
- y
- x
- color

Input. InY=3, InX=3 InC=2

ic=0



Example:

KnY=3, KnX=3, KnC=2

Px=Py=1

Sx=Sy=1

Assume BS=1

Therefore:

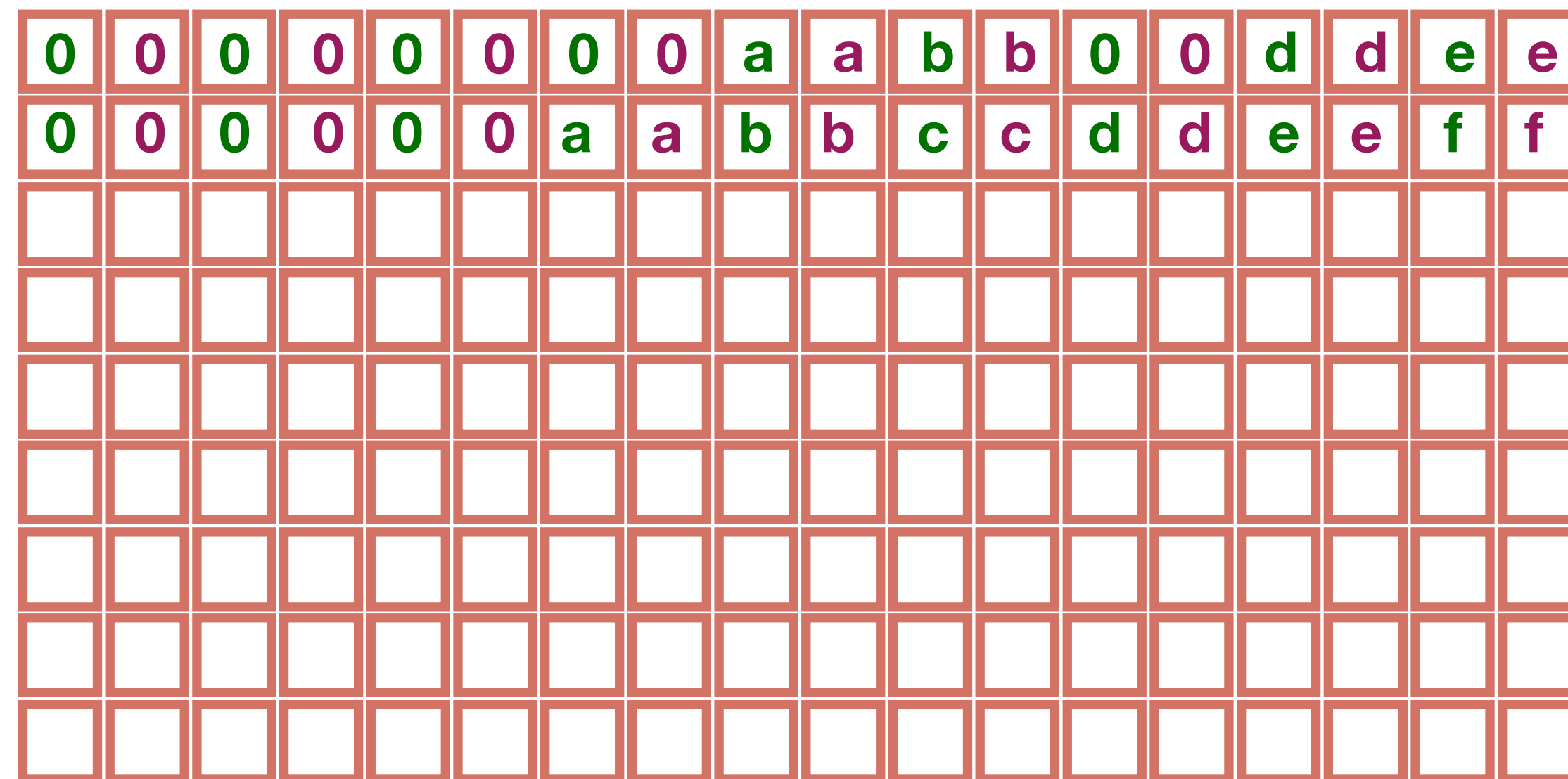
$OpY = (InY - KnY + 2Py) / Sy + 1 = 3$

$OpX = (InX - KnX + 2Px) / Sx + 1 = 3$

$OpC = KnC$

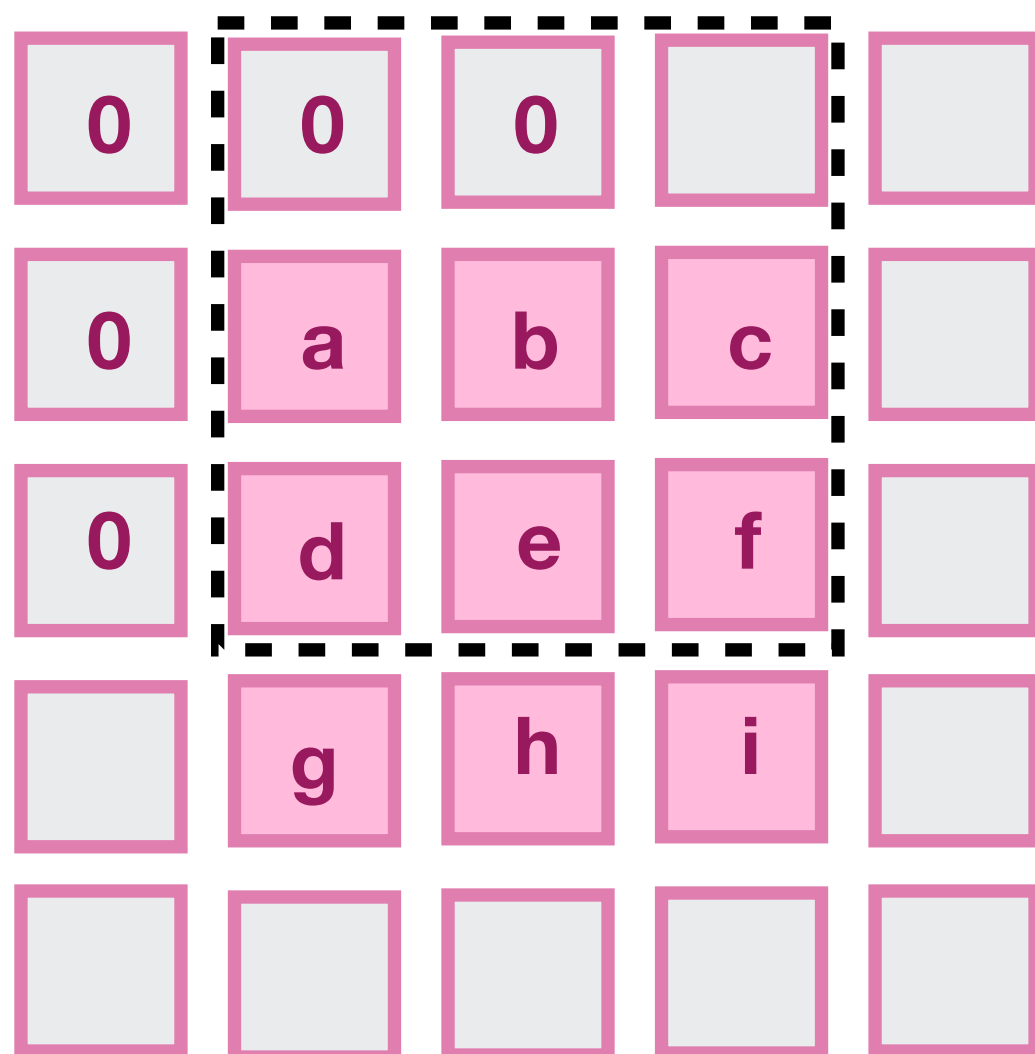
Output of Im2Col is a matrix:

$(BS * 3 * 3) \times (3 * 3 * 2)$



and so on...

ic=1



Kernel, dashed line on the left, takes $3 * 3 * 2$ inputs at the time, im2col lays them out in a row

Indices from slow to fast are:

- batch id
- y
- x
- color

Input. InY=3, InX=3 InC=2

ic=0

0	0	0		
0	a	b	c	
0	d	e	f	
	g	h	i	

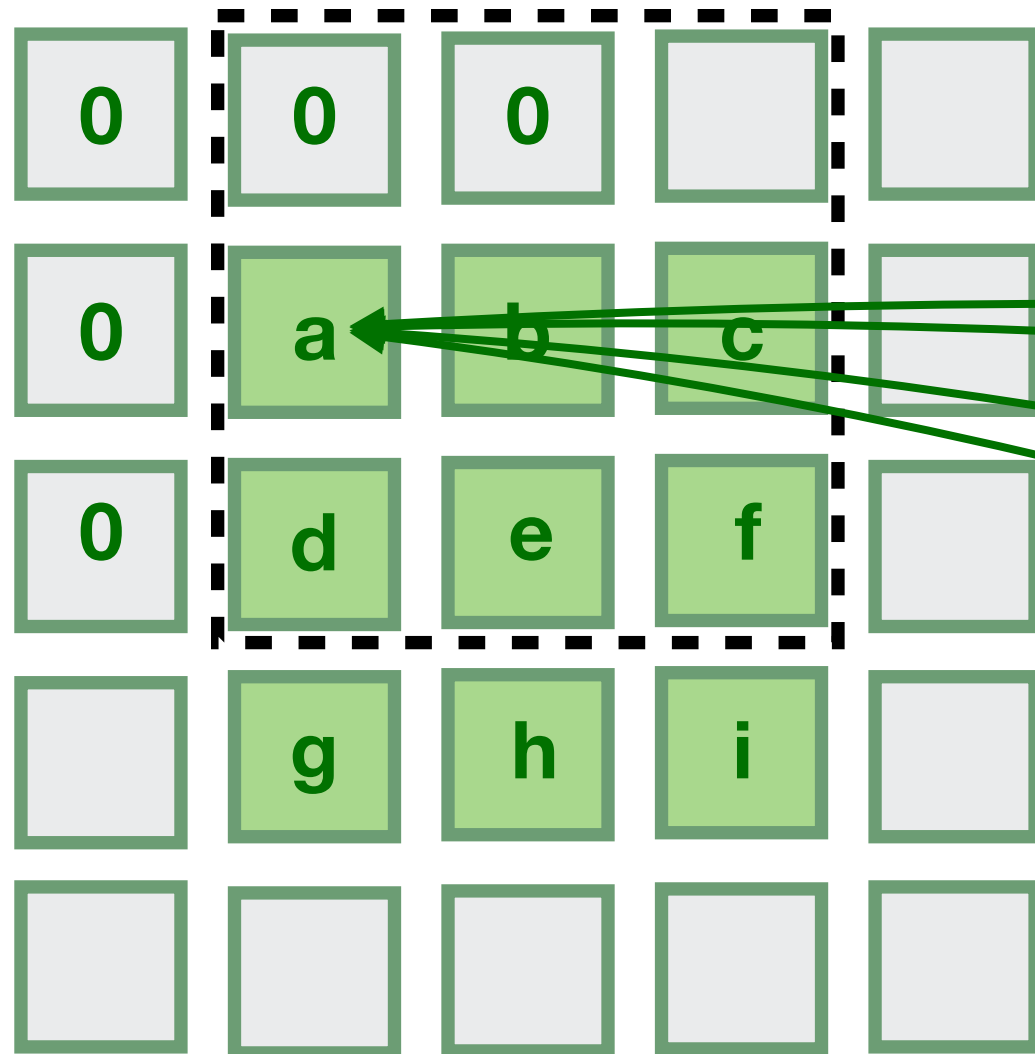
0	0	0	0	0	0	0	0	a	a	b	b	0	0	d	d	e	e
0	0	0	0	0	0	a	a	b	b	c	c	d	d	e	e	f	f
e	e	f	f	0	0	h	h	i	i	0	0	0	0	0	0	0	0

Since output is $BS * 3 * 3 * KnC$
 output of im2col is matrix with $BS * 3 * 3$ rows

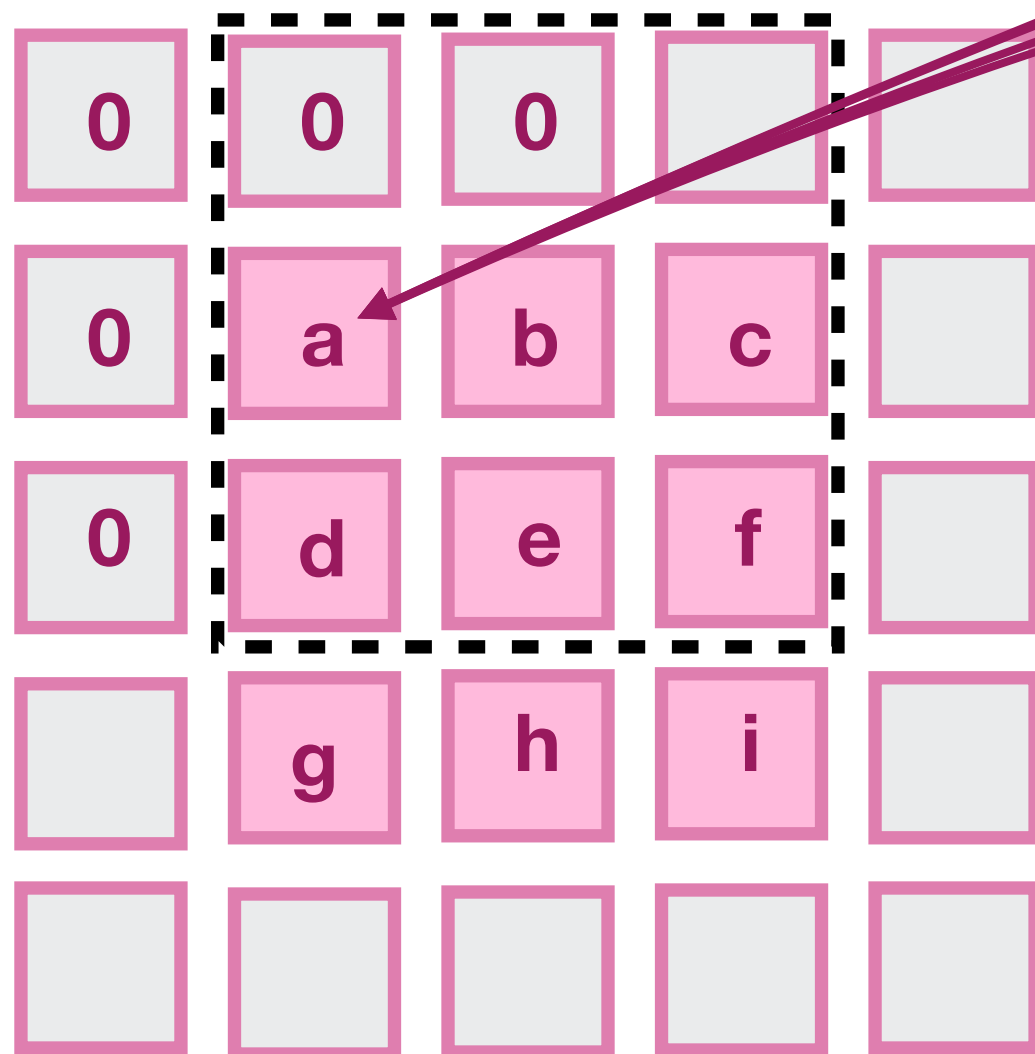
ic=1

0	0	0		
0	a	b	c	
0	d	e	f	
	g	h	i	

Input. InY=3, InX=3 InC=2



ic=0



ic=1

Backprop of im2col picks errors on the side of the matrix and adds them up on the side of the inputs.

This now represents dErr/dOut of im2col:

