

Backpropagation in Convolutional Neural Networks

- Three operations of convolutions, max-pooling, and ReLU.
- The ReLU backpropagation is the same as any other network.
 - Passes gradient to a previous layer only if the original input value was positive. 只pass正數
- The max-pooling passes the gradient flow through the largest cell in the input volume.
•
- Main complexity is in backpropagation through convolutions.

Backpropagating through Convolutions

- Traditional backpropagation is transposed matrix multiplication.
- Backpropagation through convolutions is transposed convolution (i.e., with an *inverted filter*).
- Derivative of loss with respect to each cell is backpropagated.
 - Elementwise approach of computing which cell in input contributes to which cell in output.
 - Multiplying with an inverted filter.
- Convert layer-wise derivative to weight-wise derivative and add over shared weights.

Backpropagation with an Inverted Filter [Single Channel]

a	b	c
d	e	f
g	h	i

FILTER DURING
CONVOLUTION

i	h	g
f	e	d
c	b	a

FILTER DURING
BACKPROPAGATION

- Multichannel case: We have 20 filters for 3 input channels (RGB) \Rightarrow We have $20 \times 3 = 60$ spatial slices.
- Each of these 60 spatial slices will be inverted and grouped into 3 sets of filters with depth 20 (one each for RGB).
- Backpropagate with newly grouped filters.

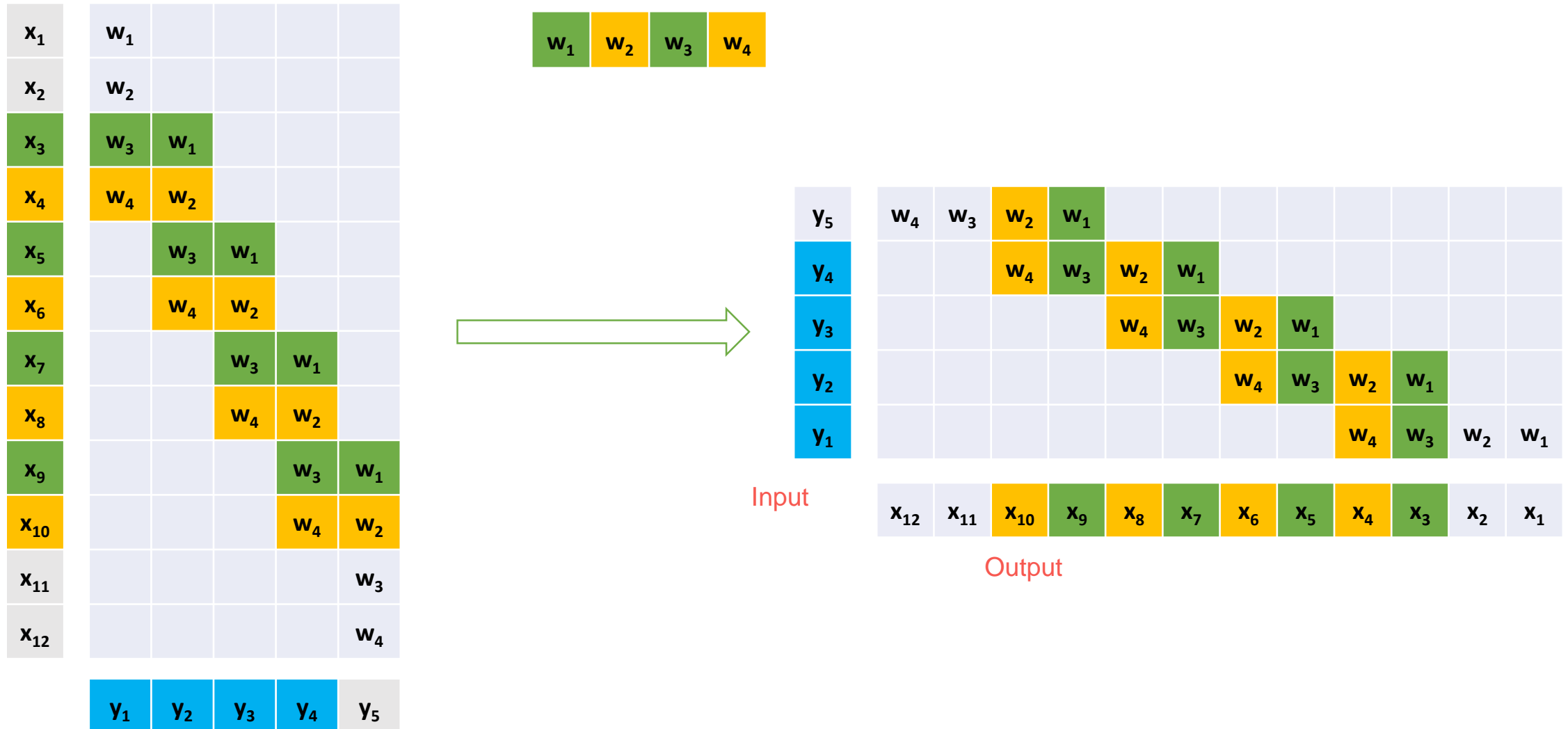
1-D Convolution Layer with stride 2

x_1	w_1				
x_2	w_2				
x_3	w_3	w_1			
x_4	w_4	w_2			
x_5		w_3	w_1		
x_6		w_4	w_2		
x_7			w_3	w_1	
x_8			w_4	w_2	
x_9				w_3	w_1
x_{10}				w_4	w_2
x_{11}					w_3
x_{12}					w_4
	y_1	y_2	y_3	y_4	y_5

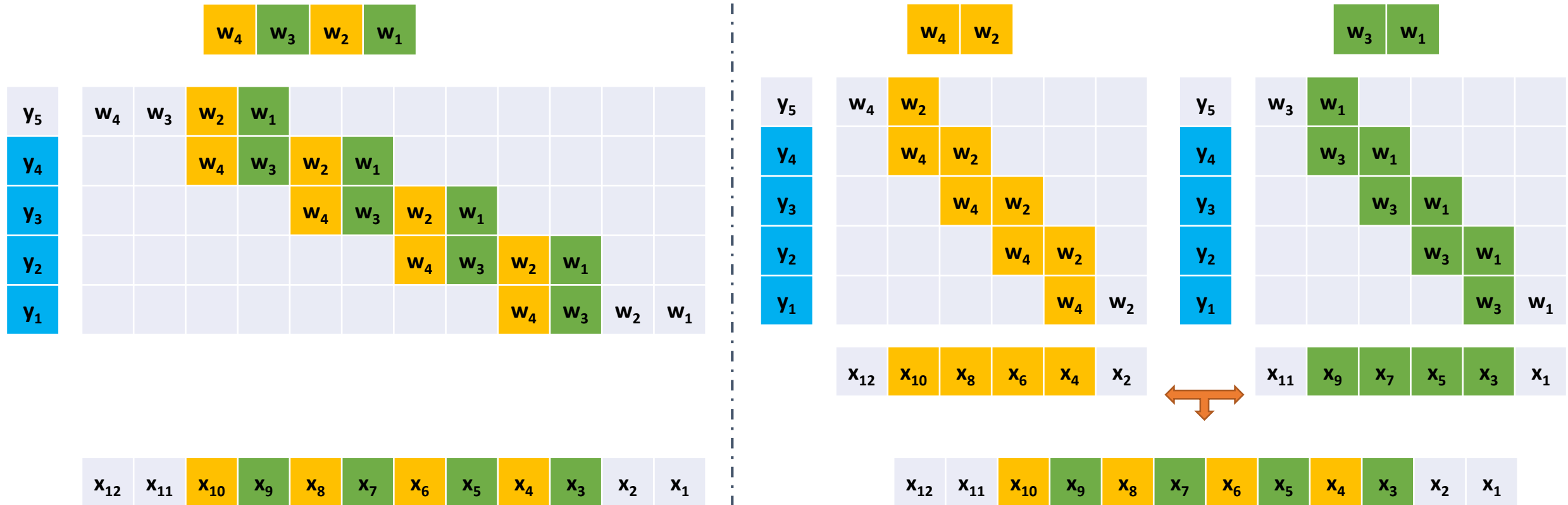
w_1	w_2	w_3	w_4
-------	-------	-------	-------

$$C = \begin{bmatrix} w_1 & & & & & & & & \\ w_2 & & & & & & & & \\ w_3 & w_1 & & & & & & & \\ w_4 & w_2 & & & & & & & \\ & w_3 & w_1 & & & & & & \\ & w_4 & w_2 & & & & & & \\ & & w_3 & w_1 & & & & & \\ & & w_4 & w_2 & & & & & \\ & & & w_3 & w_1 & & & & \\ & & & w_4 & w_2 & & & & \\ & & & & & w_3 & & & \\ & & & & & w_4 & & & \end{bmatrix}$$

1D Deconvolution Layer

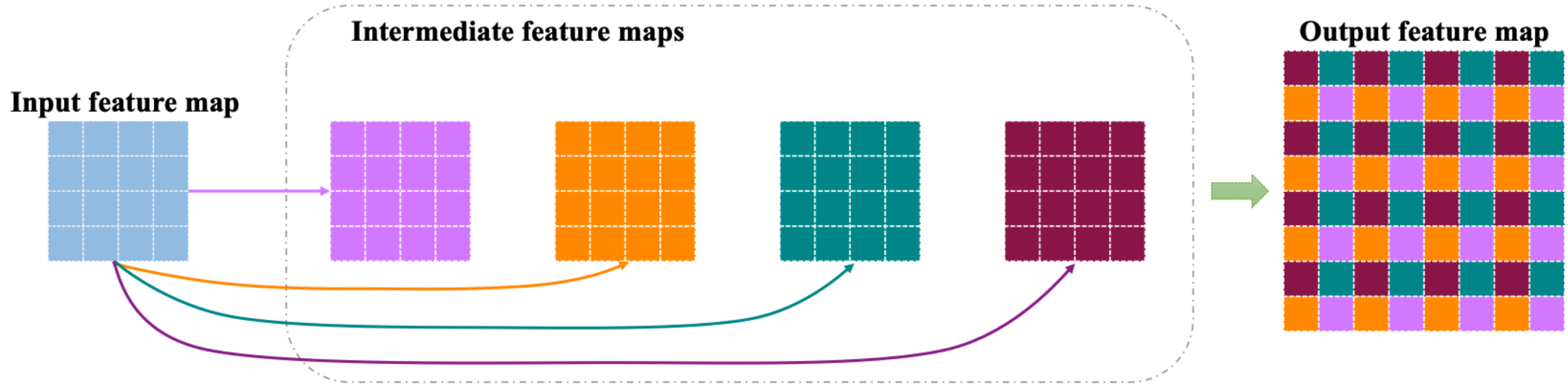


1-D Deconvolution VS. 1-D Convolution

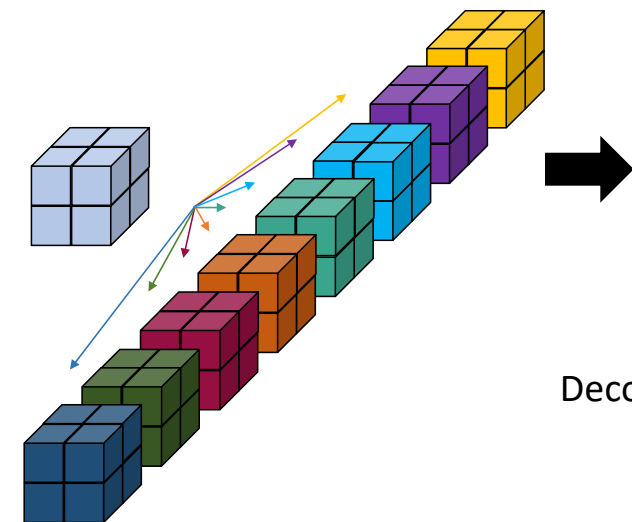
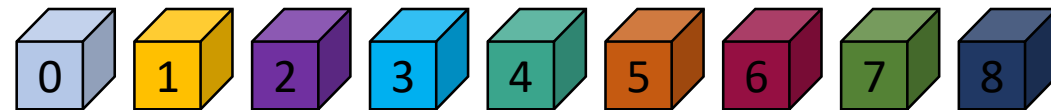


1D Deconvolution with up-sampling factor 2 is equivalent to two 1-D convolutions.

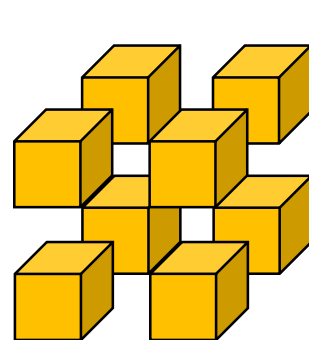
2-D Deconvolution



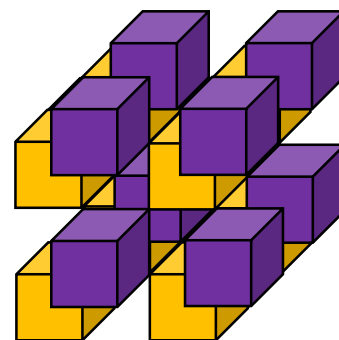
3-D Deconvolution



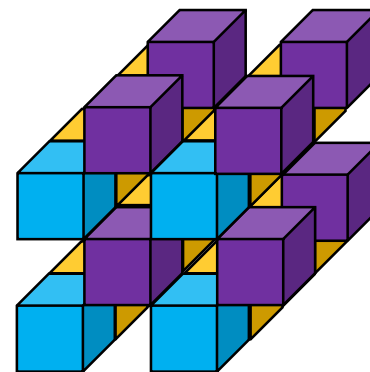
Deconvolution



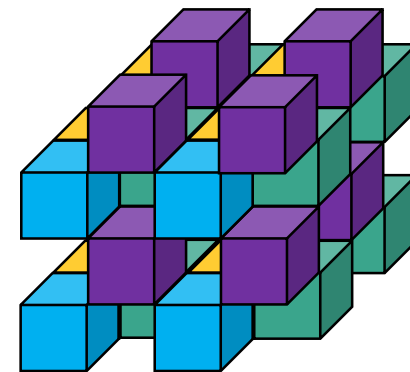
$0 \rightarrow 1$



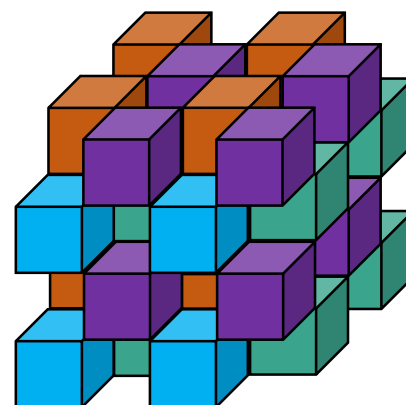
$0 \rightarrow 2$



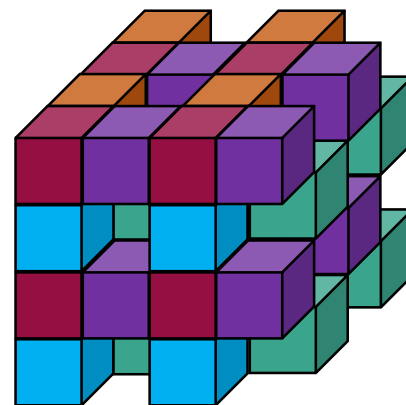
$0 \rightarrow 3$



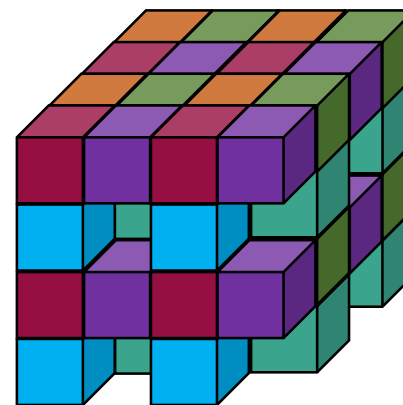
$0 \rightarrow 4$



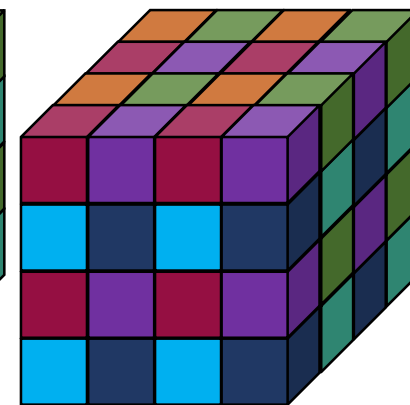
$0 \rightarrow 5$



$0 \rightarrow 6$



$0 \rightarrow 7$



$0 \rightarrow 8$

Deconvolution

(Output)

Reading

- Wenzhe Shi, et al. : Is the deconvolution layer the same as a convolutional layer?
- Vincent Dumoulin and Francesco Visin : A guide to convolution arithmetic for deep learning