4-2 Convolution Neural Network I

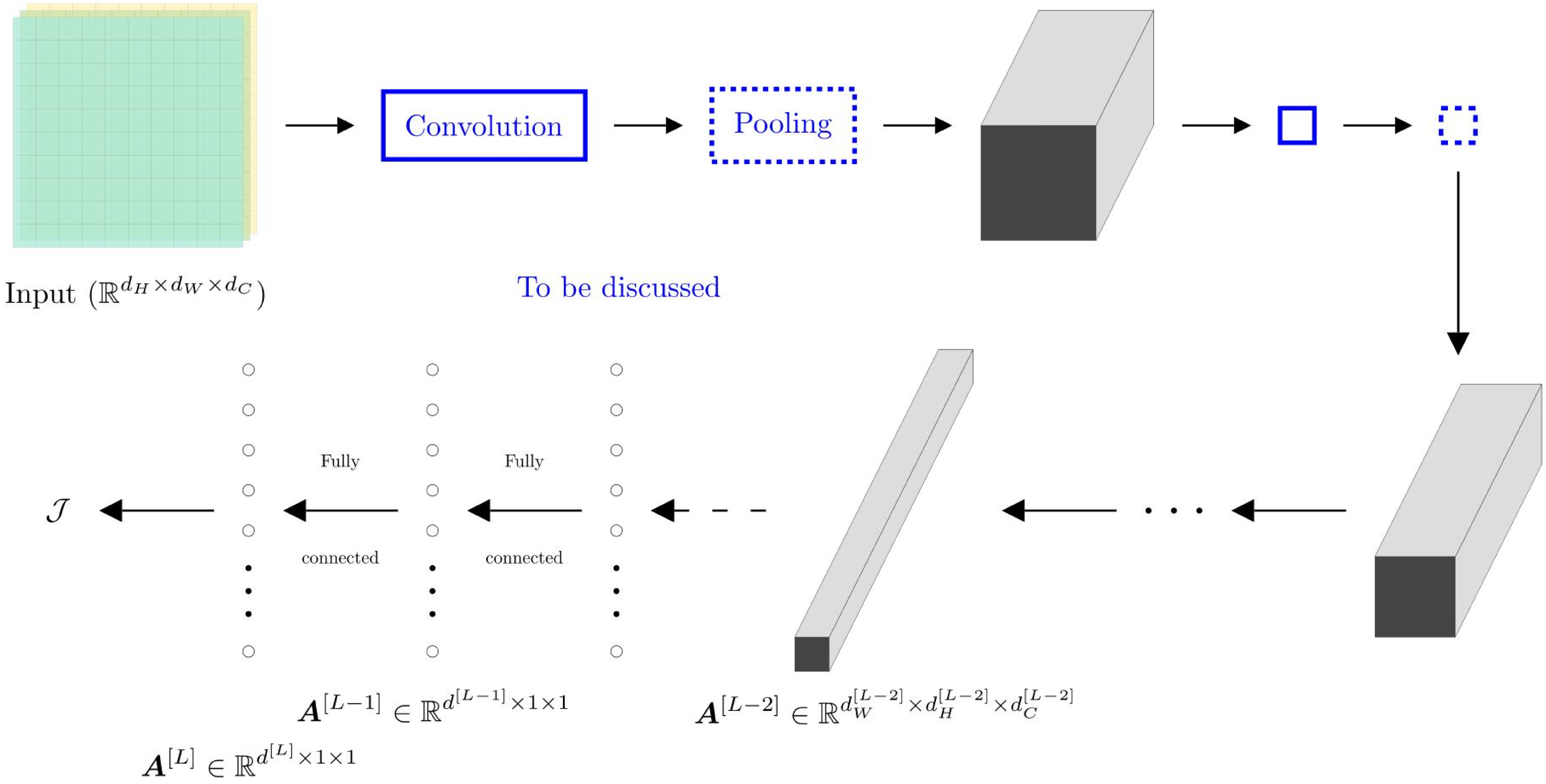
Zhonglei Wang

WISE and SOE, XMU, 2025

Contents

Introduction

1. Given an image, why not use a fully connected neural network?
    - Number of parameters is overwhelmingly large, easily leading to overfitting
    - Overlooks the spatial structure
    - Not location (translation) invariant
2. Consider a new network, convolution neural network (CNN), for image processing


Convolution
Pooling
Input ($\mathbb{R}^{d_H \times d_W \times d_C}$)
To be discussed
$\mathcal{J}$
Fully
connected
Fully
connected
$\boldsymbol{A}^{[L-1]} \in \mathbb{R}^{d^{[L-1]} \times 1 \times 1}$
$\boldsymbol{A}^{[L-2]} \in \mathbb{R}^{d_W^{[L-2]} \times d_H^{[L-2]} \times d_C^{[L-2]}}$
$\boldsymbol{A}^{[L]} \in \mathbb{R}^{d^{[L]} \times 1 \times 1}$

Summary

1. Generally, $d_C^{[l]}$ increases as l increases, but $d_W^{[l]}$ and $d_H^{[l]}$ decrease
2. The vectorization step does not involve any model parameter
3. Typically, there are three "layers" for CNN
   - Convolution layers (CONV)
   - Pooling layers (POOL)
   - Fully connected layers (FC)

Notation

1. l: layer index
2. $d_H^{[l]}$: height of the "image" of the lth layer
3. $d_W^{[l]}$: width of the "image" of the lth layer
4. $d_C^{[l]}$: number of channels of the lth layer
5. $f^{[l]}$: kernel size associated with the lth layer
6. $p^{[l]}$: padding associated with the lth layer
7. $s^{[l]}$: stride associated with the lth layer

Vectorization

1. $\boldsymbol{A}^{[l-1]} \in \mathbb{R}^{n \times d_H^{[l-1]} \times d_W^{[l-1]} \times d_C^{[l-1]}}$: input for the lth layer

2. $\boldsymbol{Z}^{[l]} \in \mathbb{R}^{n \times d_H^{[l]} \times d_W^{[l]} \times d_C^{[l]}}$: linear transformed result

$$\boldsymbol{Z}^{[l]} = \boldsymbol{A}^{[l-1]} \text{``}*\text{''} \boldsymbol{W}^{[l]} \text{``}+\text{''} \boldsymbol{b}^{[l]}$$
$$\boldsymbol{A}^{[l]} = \sigma^{[l]}(\boldsymbol{Z}^{[l]})$$

- $\boldsymbol{W}^{[l]} \in \mathbb{R}^{d_C^{[l]} \times f^{[l]} \times f^{[l]} \times d_C^{[l-1]}}$: kernels

- $\boldsymbol{b}^{[l]} \in \mathbb{R}^{d_C^{[l]} \times 1 \times 1 \times 1}$: bias term

- "$*$": convolution for each sample and each channel

- "$+$": one common bias for each channel

- $\sigma^{[l]}(\cdot)$: activation function for the lth layer

Vectorization

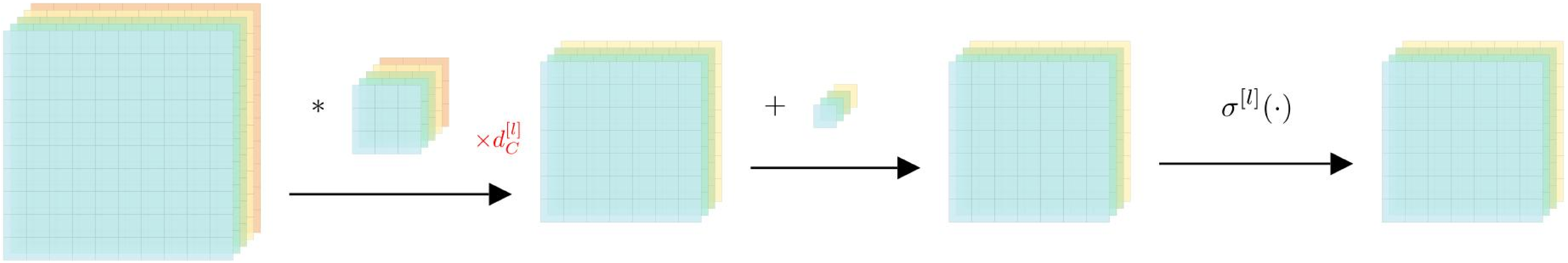
1. For simplicity, we show forward- and back-propagation for $n = 1$
   - $\boldsymbol{A}^{[l-1]} \in \mathbb{R}^{d_H^{[l-1]} \times d_W^{[l-1]} \times d_C^{[l-1]}}$
   - $\boldsymbol{Z}^{[l]} \in \mathbb{R}^{d_H^{[l]} \times d_W^{[l]} \times d_C^{[l]}}$
2. Forward propagation:

$$\boldsymbol{Z}^{[l]} = \boldsymbol{A}^{[l-1]} \text{``}*\text{''} \boldsymbol{W}^{[l]} \text{``}+\text{''} \boldsymbol{b}^{[l]}$$
$$\boldsymbol{A}^{[l]} = \sigma^{[l]}(\boldsymbol{Z}^{[l]})$$

Convolution

1. Consider:
    - $p^{[l]} = 0$
    - $s^{[l]} = 1$

Convolution



$\boldsymbol{A}^{[l-1]} \in \mathbb{R}^{d_H^{[l-1]} \times d_W^{[l-1]} \times d_C^{[l-1]}}$ $\boldsymbol{Z}_0^{[l]} = \mathbb{R}^{d_H^{[l]} \times d_W^{[l]} \times d_C^{[l]}}$ $\boldsymbol{Z}^{[l]} \in \mathbb{R}^{d_H^{[l]} \times d_W^{[l]} \times d_C^{[l]}}$

$\boldsymbol{W}^{[l]} \in \mathbb{R}^{d_C^{[l]} \times f^{[l]} \times f^{[l]} \times d_C^{[l-1]}}$ $\boldsymbol{b}^{[l]} \in \mathbb{R}^{d_C^{[l]} \times 1 \times 1 \times 1}$ $\boldsymbol{A}^{[l]} \in \mathbb{R}^{d_H^{[l]} \times d_W^{[l]} \times d_C^{[l]}}$

Pooling

1. Pooling is typically used after convolution layers,
    - Reduce height and width for each channel
    - Achieves robustness by neglecting useless or repeated information
    - Increase the receptive field
    - Achieve robustness small translations, rotations, and other distortions in the input
    - Prevent overfitting, especially when combined with dropout

Pooling

1. Two types of pooling:
   - Average pooling
   - Max pooling
2. Notice
   - Pooling is conducted for each channel
   - Pooling does not involve new model parameters

Average pooling

1. Input size: 6×6
2. Kernel size: 2×2
3. Stride: 2

Average pooling

Input

3	6	5	4	8	9
1	7	9	6	8	0
5	0	9	6	2	0
5	2	6	3	7	0
9	0	3	2	3	1
3	1	3	7	1	7

Kernel

0.25	0.25
0.25	0.25

Result

4.25	6.0	6.25
3.0	6.0	2.25
3.25	3.75	3.0

Max pooling

1. Input size: 6×6
2. Kernel size: 2×2
3. Stride: 2

Max pooling

Input

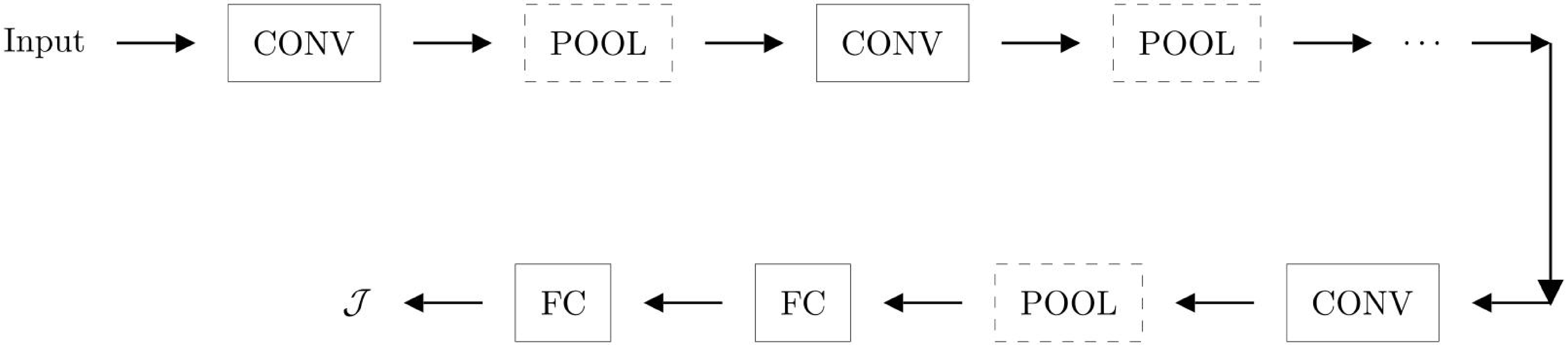
3	6	5	4	8	9
1	7	9	6	8	0
5	0	9	6	2	0
5	2	6	3	7	0
9	0	3	2	3	1
3	1	3	7	1	7

Kernel

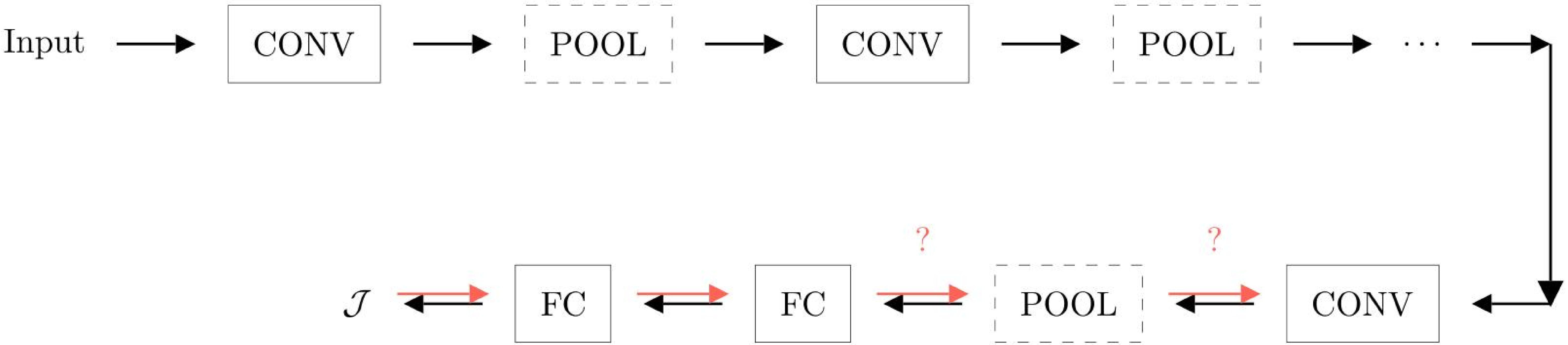
Result

7	9	9
5	9	7
9	7	7

Forward propagation



Backpropagation



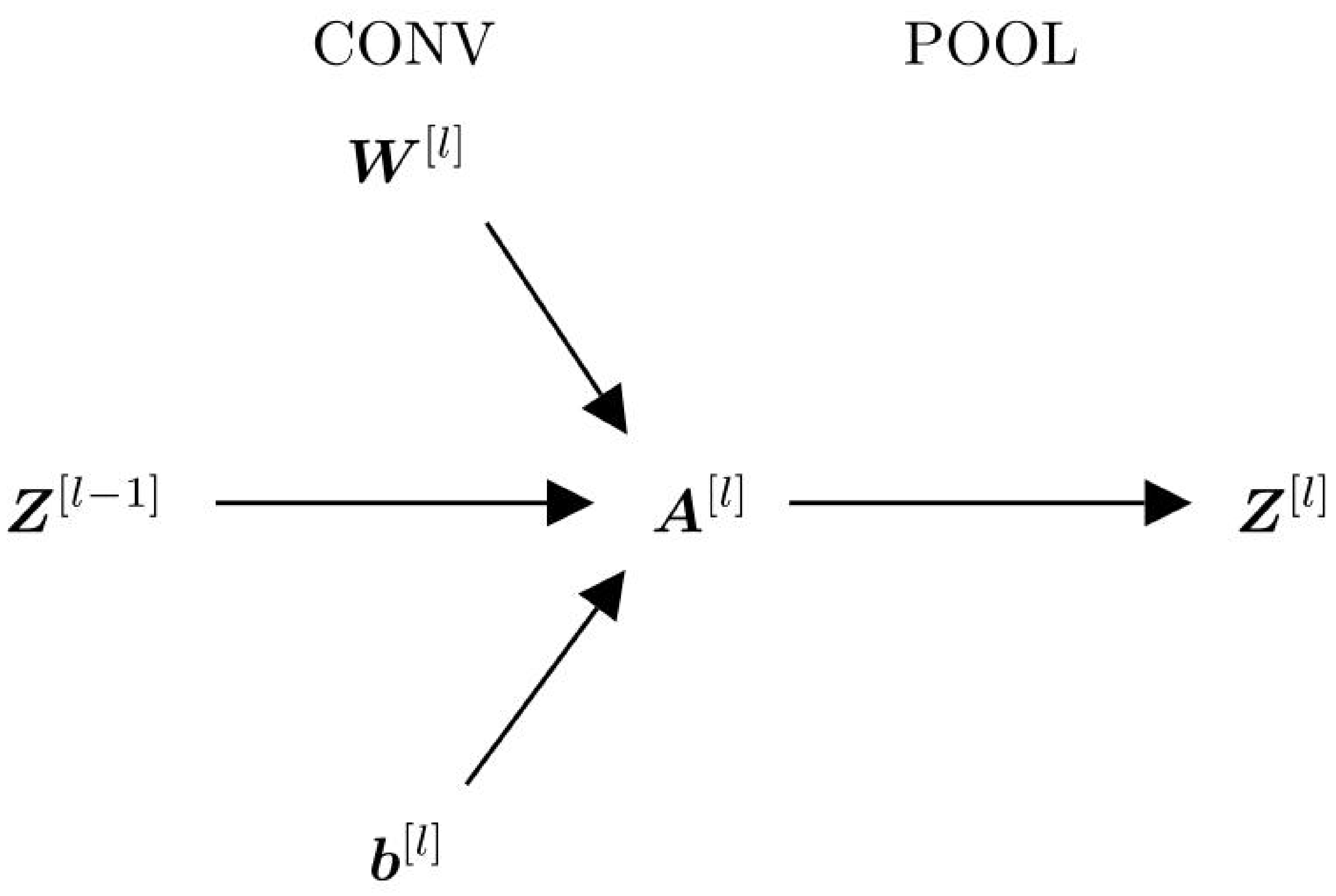
Backpropagation

1. We have learnt backpropagation for fully connected layers
2. Thus, it remains to show the backpropagation for
   - The pooling layer (POOL)
   - The convolution layer (CONV)
3. In the following, we assume a POOL is conducted right after a CONV

Backpropagation

1. Denote
   - $\boldsymbol{A}^{[l-1]} \in \mathbb{R}^{? \times ? \times d_C^{[l-1]}}$: output of the $(l-1)$th CONV
   - $\boldsymbol{Z}^{[l-1]} \in \mathbb{R}^{d_H^{[l-1]} \times d_W^{[l-1]} \times d_C^{[l-1]}}$: output of the $(l-1)$th POOL, taking $\boldsymbol{A}^{[l-1]}$ as input
   - $\boldsymbol{W}^{[l]} \in \mathbb{R}^{d_C^{[l]} \times f^{[l]} \times f^{[l]} \times d_C^{[l-1]}}$: kernel for the lth layer, taking $\boldsymbol{Z}^{[l-1]}$ as input
   - $\boldsymbol{b}^{[l]} \in \mathbb{R}^{d_C^{[l]} \times 1 \times 1 \times 1}$: bias for the lth layer, taking $\boldsymbol{Z}^{[l-1]}$ as input
   - $\boldsymbol{A}^{[l]} \in \mathbb{R}^{? \times ? \times d_C^{[l]}}$: output of the lth CONV
   - $\boldsymbol{Z}^{[l]} \in \mathbb{R}^{d_H^{[l]} \times d_W^{[l]} \times d_C^{[l]}}$: output of the lth POOL

Structure



1. Assume that $\mathrm{d}\boldsymbol{Z}^{[l]}$ is available

Backpropagation for average POOL

$\boldsymbol{A}^{[l]}$

3	6	5	4	8	9
1	7	9	6	8	0
5	0	9	6	2	0
5	2	6	3	7	0
9	0	3	2	3	1
3	1	3	7	1	7

Kernel $\boldsymbol{M}$

0.25	0.25
0.25	0.25

$\boldsymbol{Z}^{[l]}$

4.25	6.0	6.25
3.0	6.0	2.25
3.25	3.75	3.0

Backpropagation for average POOL

1. For average pooling, we have

$$\mathrm{d}\boldsymbol{A}^{[l]} = \mathrm{d}\boldsymbol{Z}^{[l]} \otimes \boldsymbol{M}$$

   - $\boldsymbol{A} \otimes \boldsymbol{B}$: Kronecker production of two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$

Backpropagation for Max POOL

$\boldsymbol{A}^{[l]}$

3	6	5	4	8	9
1	7	9	6	8	0
5	0	9	6	2	0
5	2	6	3	7	0
9	0	3	2	3	1
3	1	3	7	1	7

Kernel

$\boldsymbol{Z}^{[l]}$

7	9	9
5	9	7
9	7	7

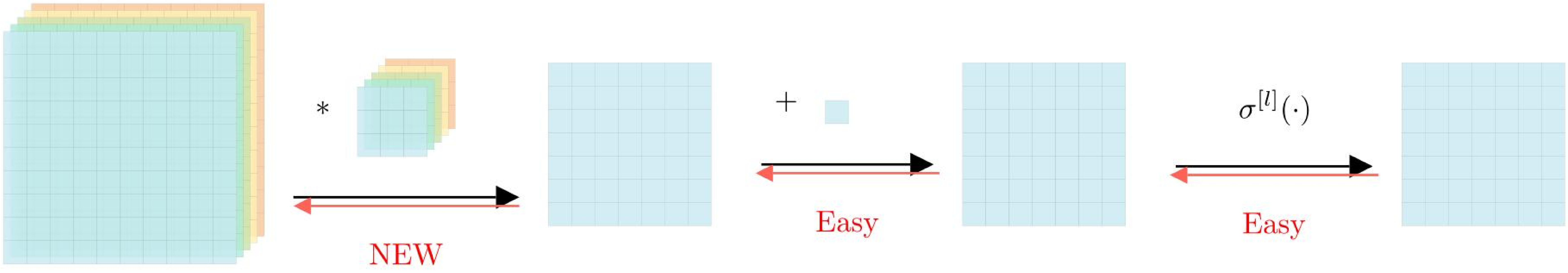
Backpropagation for Max POOL

1. For Max pooling, $\mathrm{d}\boldsymbol{A}^{[l]}$ is obtained by
   - stacking small matrices associated with each elements in $\boldsymbol{Z}^{[l]}$ times a 2×2 mask matrix
   - Those 2×2 matrices consist of 0s and only one 1
2. We have finished the backpropagation from $\mathrm{d}\boldsymbol{Z}^{[l]}$ to $\mathrm{d}\boldsymbol{A}^{[l]}$
   - POOL does not involve any model parameters
3. Assume the availability of $\mathrm{d}\boldsymbol{A}^{[l]}$ for the following analysis

Backpropagation for CONV

1. For simplicity, let $d_C^{[l]} = 1$.
2. Assume $\mathrm{d}\boldsymbol{A}^{[l]}$ to be available

Backpropagation for CONV



$\boldsymbol{Z}^{[l-1]} \in \mathbb{R}^{d_H^{[l-1]} \times d_W^{[l-1]} \times d_C^{[l-1]}}$ $\quad$ $\boldsymbol{Z}_0^{[l]} \in \mathbb{R}^{? \times ? \times 1}$ $\quad$ $\boldsymbol{Z}_1^{[l]} \in \mathbb{R}^{? \times ? \times 1}$

$\boldsymbol{W}^{[l]} \in \mathbb{R}^{f^{[l]} \times f^{[l]} \times d_C^{[l-1]}}$ $\quad$ $\boldsymbol{b}^{[l]} \in \mathbb{R}^{1 \times 1 \times 1}$ $\quad$ $\boldsymbol{A}^{[l]} \in \mathbb{R}^{? \times ? \times 1}$

$\mathrm{d}\boldsymbol{W}^{[l]} = ?$ $\quad$ $\mathrm{d}\boldsymbol{b}^{[l]} = \sum_{ij} \mathrm{d}Z_{1,ij}^{[l]}$

$\mathrm{d}\boldsymbol{Z}^{[l-1]} = ?$ $\quad$ $\mathrm{d}\boldsymbol{Z}_0^{[l]} = \mathrm{d}\boldsymbol{Z}_1^{[l]}$ $\quad$ $\mathrm{d}\boldsymbol{Z}_1^{[l]} = \sigma^{[l]\prime}(\boldsymbol{Z}^{[l]}) \circ \mathrm{d}\boldsymbol{A}^{[l]}$

Backpropagation for CONV

1. To find derivative with respect to something, we need to find where its information is contained
2. Initialize $\mathrm{d}\boldsymbol{Z}^{[l-1]}$ as a zero matrix of the same dimension as $\boldsymbol{Z}^{[l-1]}$
3. Informally, obtain

$$\mathrm{d}\boldsymbol{Z}^{[l-1]}_{\text{slice},ij} {\color{red}+=} \mathrm{d}Z^{[l]}_{0,ij} \times \boldsymbol{W}^{[l]}$$

- $\boldsymbol{Z}^{[l-1]}_{\text{slice},ij}$: the part of $\boldsymbol{Z}^{[l-1]}$ used to obtain $Z^{[l]}_{0,ij}$

Backpropagation for CONV

1. It is obvious that every element in $\boldsymbol{Z}_0^{[l]}$ contains information of $\boldsymbol{W}^{[l]}$
2. Thus, we have

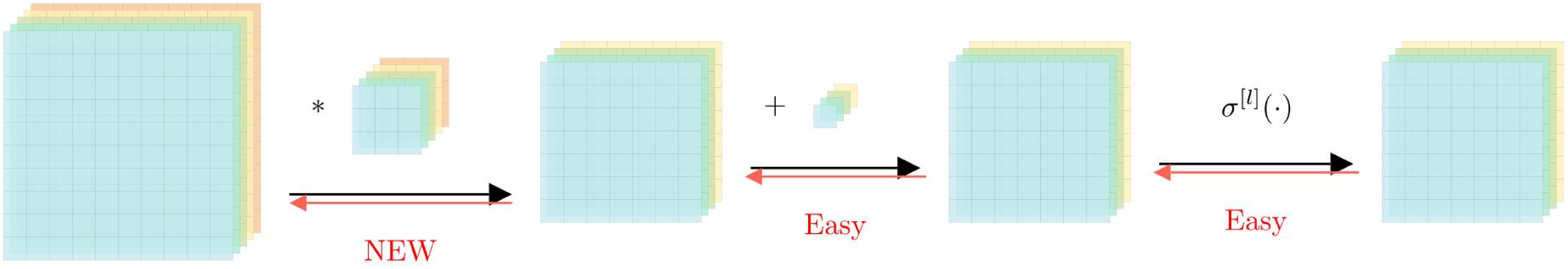
$$\mathrm{d}\boldsymbol{W}^{[l]} = \sum_{ij} \mathrm{d}Z_{0,ij}^{[l]} \times \boldsymbol{B}_{ij}^{[l-1]}$$

- $\boldsymbol{B}_{ij}^{[l-1]}$: the part of $\boldsymbol{Z}^{[l-1]}$ used to obtain $Z_{0,\boldsymbol{ij}}^{[l]}$

Backpropagation for CONV

1. Consider the general case where there exist $d_C^{[l]}$ channels in the lth layer.

Backpropagation for CONV



$\boldsymbol{Z}^{[l-1]} \in \mathbb{R}^{d_H^{[l-1]} \times d_W^{[l-1]} \times d_C^{[l-1]}}$ $\quad$ $\boldsymbol{Z}_0^{[l]} \in \mathbb{R}^{? \times ? \times d_C^{[l]}}$ $\quad$ $\boldsymbol{Z}_1^{[l]} \in \mathbb{R}^{? \times ? \times d_C^{[l]}}$

$\boldsymbol{W}^{[l]} \in \mathbb{R}^{d_C^{[l]} \times f^{[l]} \times f^{[l]} \times d_C^{[l-1]}}$ $\quad$ $\boldsymbol{b}^{[l]} \in \mathbb{R}^{d_C^{[l]} \times 1 \times 1 \times 1}$ $\quad$ $\boldsymbol{A}^{[l]} \in \mathbb{R}^{? \times ? \times d_C^{[l]}}$

$\mathrm{d}\boldsymbol{W}^{[l]} = ?$

$$\mathrm{d}\boldsymbol{b}_c^{[l]} = \sum_{ij} \mathrm{d}Z_{1,ijc}^{[l]} \quad (c = 1, \ldots, d_C^{[l]})$$

$\mathrm{d}\boldsymbol{Z}^{[l-1]} = ?$

$$\mathrm{d}\boldsymbol{Z}_0^{[l]} = \mathrm{d}\boldsymbol{Z}_1^{[l]}$$

$$\mathrm{d}\boldsymbol{Z}_1^{[l]} = \sigma^{[l]\prime}(\boldsymbol{Z}^{[l]}) \circ \mathrm{d}\boldsymbol{A}^{[l]}$$

Backpropagation for CONV

1. Just think about where the information is contained when calculating derivatives.
2. The information about $\boldsymbol{Z}^{[l-1]}$ is contained in every channel of $\boldsymbol{Z}_0^{[l]}$
3. Initialize $\mathrm{d}\boldsymbol{Z}^{[l-1]}$ as a zero matrix of the same dimension as $\boldsymbol{Z}^{[l-1]}$
4. Informally, obtain

$$\mathrm{d}\boldsymbol{Z}_{\text{slice},ij}^{[l-1]} \mathrel{+}= \sum_{c=1}^{d_C^{[l]}} \mathrm{d}Z_{0,ijc}^{[l]} \times \boldsymbol{W}_c^{[l]}$$

- $\boldsymbol{Z}_{\text{slice},ij}^{[l-1]}$: the part of $\boldsymbol{Z}^{[l-1]}$ used to obtain $Z_{0,ijc}^{[l]}$ for $c = 1, \ldots, d_C^{[l]}$
- $\boldsymbol{W}_c^{[l]}$: the cth kernel in the lth layer

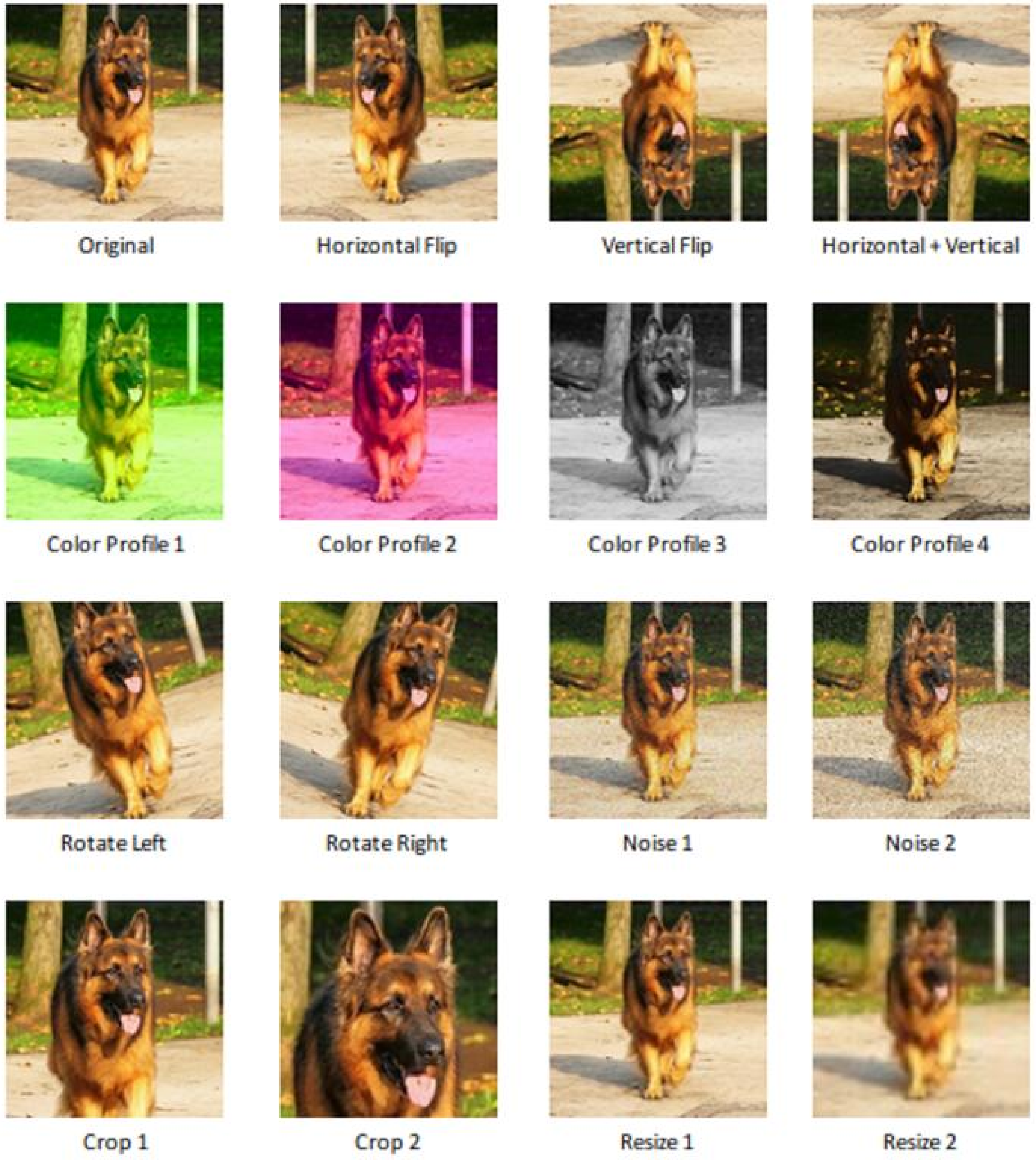
Backpropagation for CONV

1. It is obvious that every element in $\boldsymbol{Z}_{0,c}^{[l]}$ contains information of $\boldsymbol{W}_c^{[l]}$ for $c = 1, \ldots, d_C^{[l]}$
2. Thus, we have

$$\mathrm{d}\boldsymbol{W}_{c}^{[l]} = \sum_{ij} \mathrm{d}Z_{0,ijc}^{[l]} \times \boldsymbol{B}_{ij}^{[l-1]} \quad (c = 1, \ldots, d_C^{[l]})$$

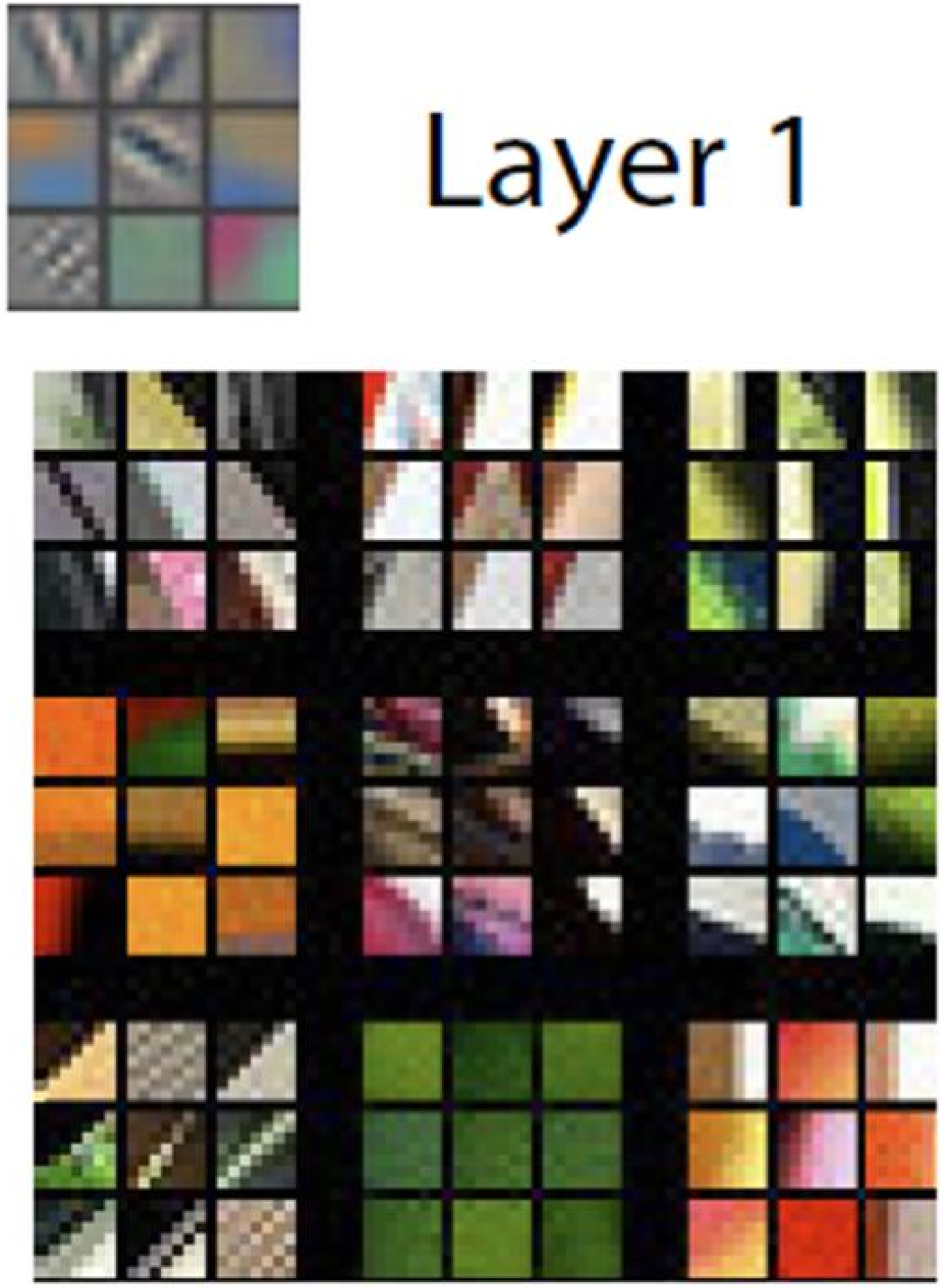
- $\boldsymbol{B}_{ij}^{[l-1]}$: the part of $\boldsymbol{Z}^{[l-1]}$ used to obtain $Z_{0,ij\boldsymbol{c}}^{[l]}$

Data augmentation



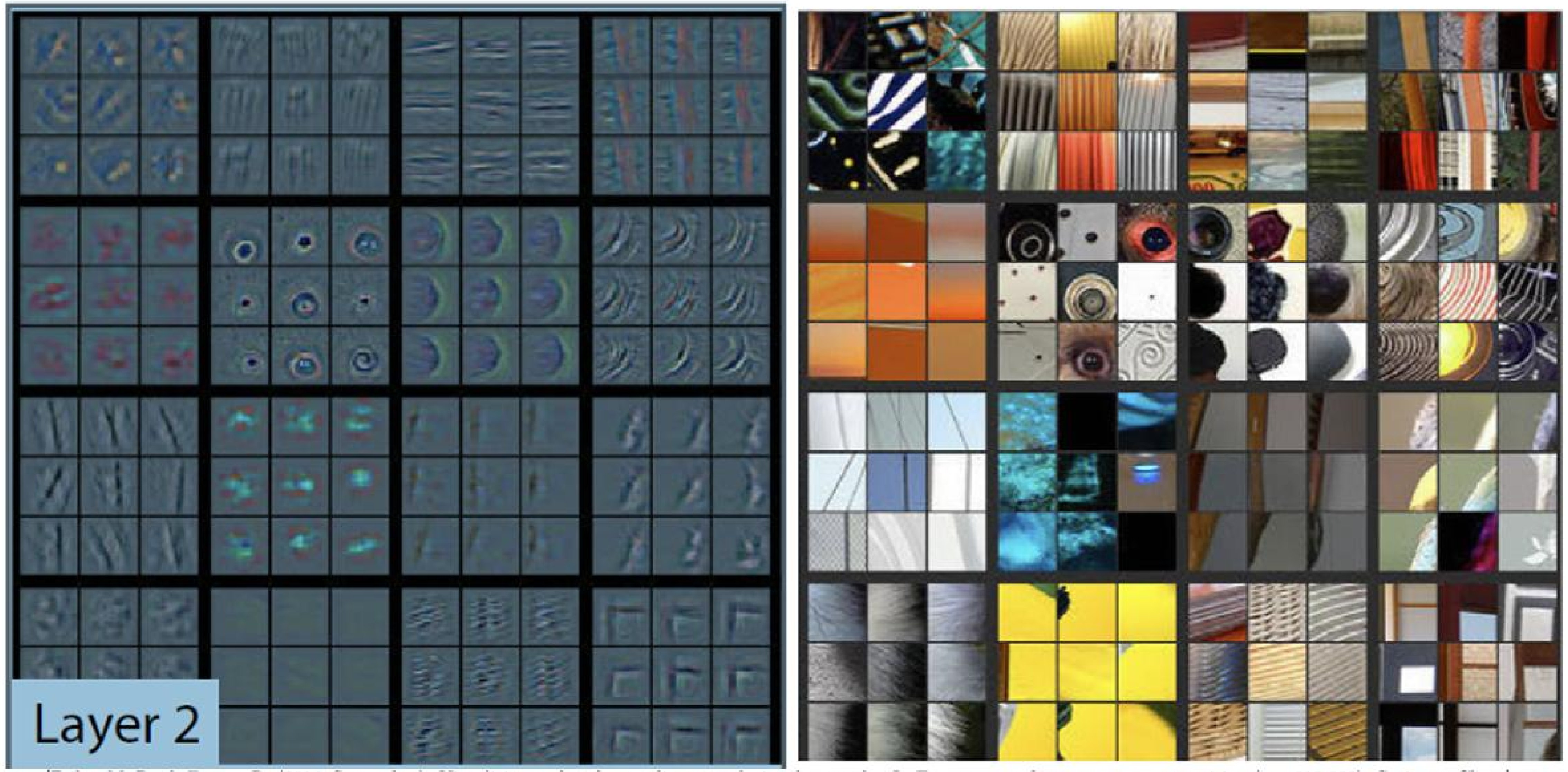
[https://www.volansys.com/blog/data-augmentation-in-ml/]

Understanding convolution networks



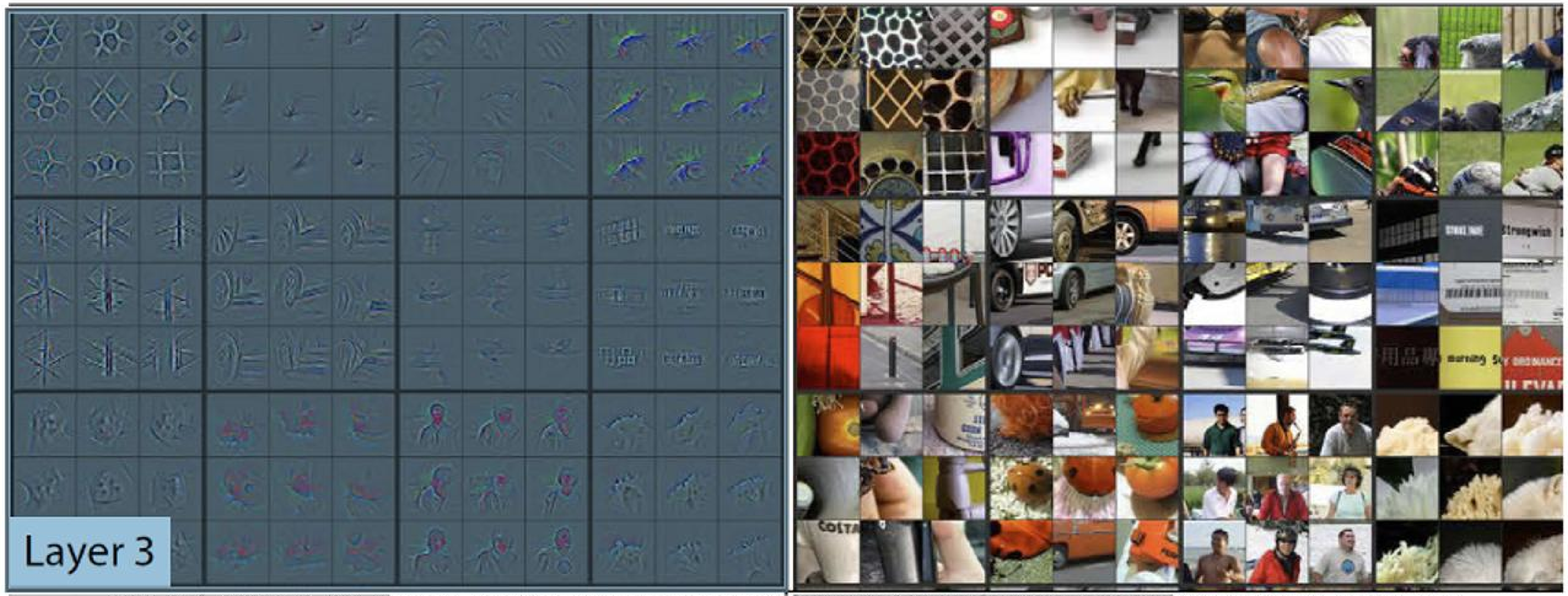
[Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In European conference on computer vision (pp. 818-833). Springer, Cham]

Understanding convolution networks



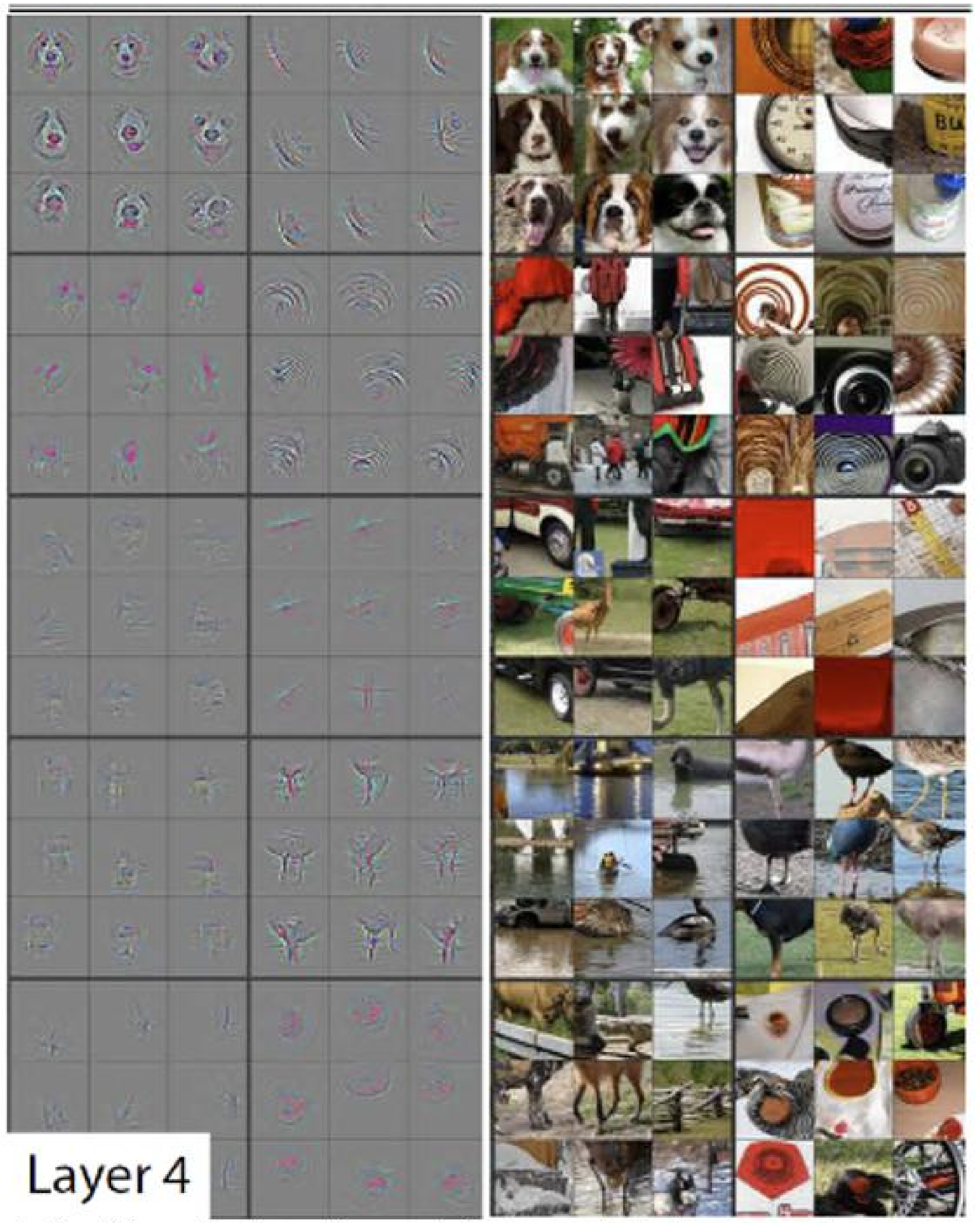
[Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In European conference on computer vision (pp. 818-833). Springer, Cham]

Understanding convolution networks



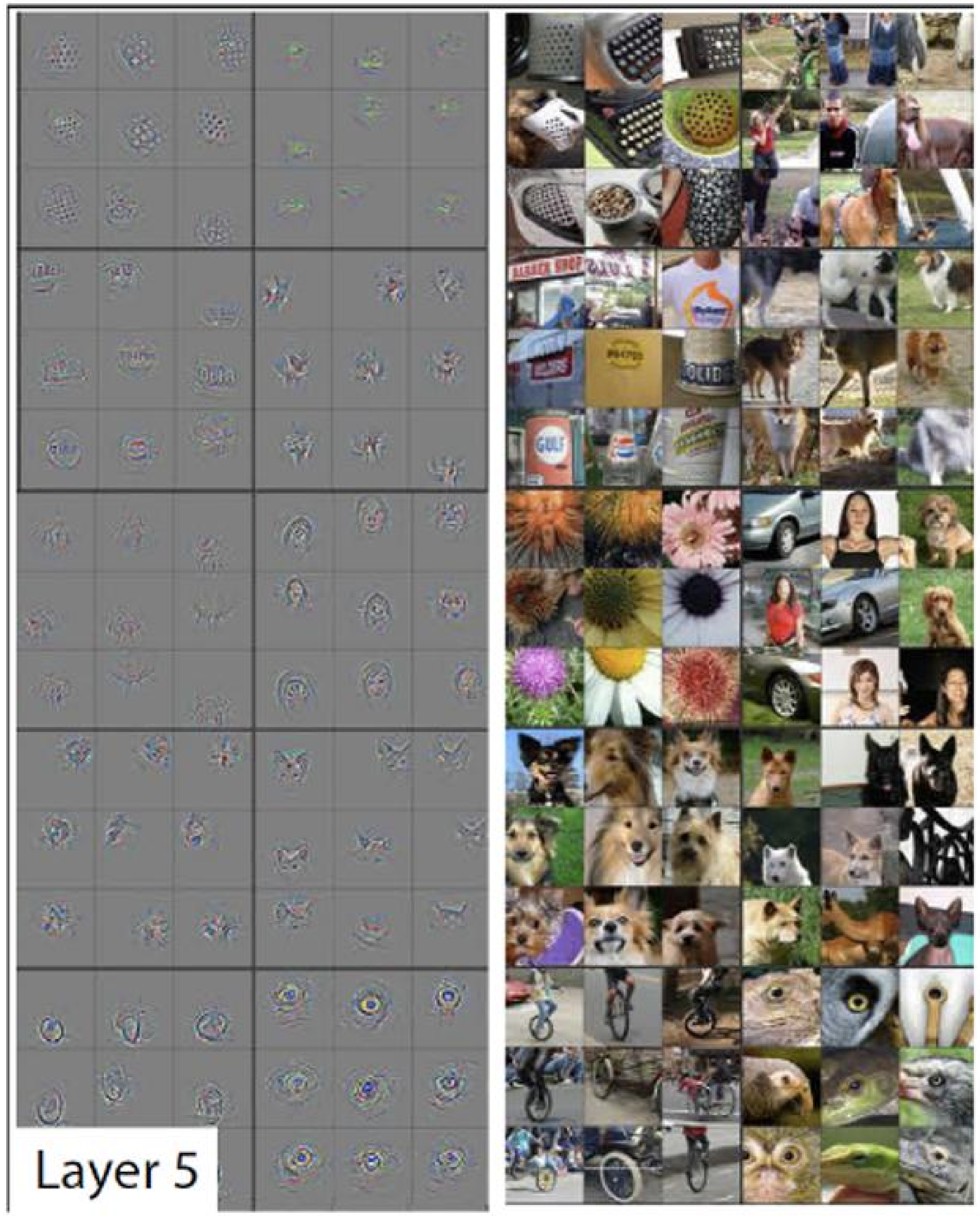
[Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In European conference on computer vision (pp. 818-833). Springer, Cham]

Understanding convolution networks



[Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In European conference on computer vision (pp. 818-833). Springer, Cham]

Understanding convolution networks



[Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In European conference on computer vision (pp. 818-833). Springer, Cham]