Contents

1 Convolutional Neural Networks 3
- 1.1 Some classic CNN models 3
  - 1.1.1 LeNet-5, AlexNet and VGG 4
  - 1.1.2 ResNet 4
  - 1.1.3 pre-act ResNet 5

References 7

1

Convolutional Neural Networks

1.1 Some classic CNN models

In this section, we will use these convolutional operations introduced above to give a brief description of some classic convolutional neural network (CNN) models. Firstly, CNNs are actually a class of special DNN models. Let us recall the DNN structure as:

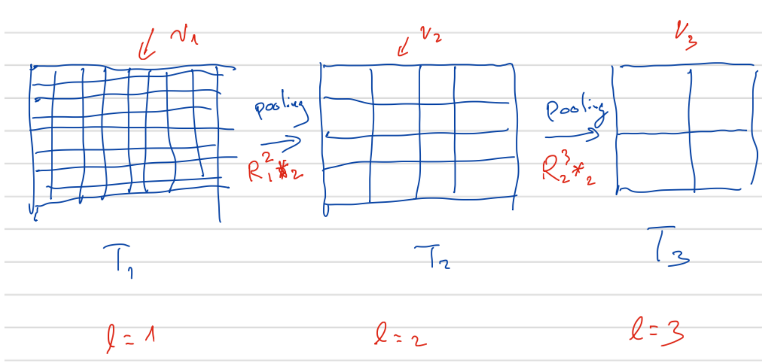
$$
\begin{cases} f^0(x) & = x \\ f^\ell(x) & = \sigma(\theta^\ell(f^{\ell-1})) \quad \ell = 1 : L \,, \\ f(x) & = W^L f^L + b^L \end{cases} \tag{1.1}
$$

where

$$
(\theta^\ell(f^{\ell-1}) = W^\ell f^{\ell-1}(x) + b^\ell. \tag{1.2}
$$

So the key features of CNNs is

1. Replace the general linear mapping to be convolution operations with multi-channel.
2. Use multi-resolution of images as shown in the next diagram.



Then we will introduce some classical architectures in convolution neural networks.

1.1.1 LeNet-5, AlexNet and VGG

The LeNet-5 [6], AlexNet [5] and VGG [7] can be written as:

Algorithm 1 h = Classic CNN($f; J, \nu_1, \cdots, \nu_J$)

1: Initialization: $f^{1,0} = f_{\rm in}(f)$.
2: **for** $\ell = 1 : J$ **do**
3: **for** $i = 1 : \nu_\ell$ **do**
4: Basic Block:

$$f^{\ell,i} = \sigma\left(\theta^{\ell,i} * f^{\ell,i-1}\right) \tag{1.3}$$

5: **end for**
6: Pooling(Restriction):

$$f^{\ell+1,0} = R_\ell^{\ell+1} *_2 f^{\ell,\nu_\ell} \tag{1.4}$$

7: **end for**
8: Final average pooling layer: $h = R_{\rm ave}(f^{L,\nu_\ell})$.

Here $R_\ell^{\ell+1} *_2$ represents for the pooling operation to sub-sampling these tensors into coarse spatial level (lower resolution). Here we use $R_\ell^{\ell+1} *_2$ to stand for the pooling operation. In general we can also have

- average pooling: fixed kernels such as

$$R_\ell^{\ell+1} = \frac{1}{9}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \tag{1.5}$$

- Max pooling $R_{\max}$ as discussed before.

In these three classic CNN models, they still need some extra fully connected layers after h as the output of CNNs. After few layers of fully connected layers, the model is completed by following a multi-class logistic regression model.

These fully connected layers are removed in ResNet to be described below.

1.1.2 ResNet

The original ResNet developed in [2] is one of the most popular CNN architectures in image classification problems.

Algorithm 2 h = ResNet($f; J, \nu_1, \cdots, \nu_J$)

1: Initialization: $r^{1,0} = f_{\rm in}(f)$.
2: **for** $\ell = 1 : J$ **do**
3: **for** $i = 1 : \nu_\ell$ **do**

Basic Block:

$$r^{\ell,i} = \sigma\left(r^{\ell,i-1} + A^{\ell,i} * \sigma \circ B^{\ell,i} * r^{\ell,i-1}\right). \tag{1.6}$$

end for

Pooling(Restriction):

$$r^{\ell+1,0} = \sigma\left(R_{\ell}^{\ell+1} *_2 r^{\ell,\nu_\ell} + A^{\ell+1,0} \circ \sigma \circ B^{\ell+1,0} *_2 r^{\ell,\nu_\ell}\right). \tag{1.7}$$

end for

Final average pooling layer: $h = R_{\mathrm{ave}}(r^{L,\nu_\ell})$.

Here $f_{\mathrm{in}}(\cdot)$ may depend on different data set and problems such as $f_{\mathrm{in}}(f) = \sigma \circ \theta^0 * f$ for CIFAR [4] and $f_{\mathrm{in}}(f) = R_{\max} \circ \sigma \circ \theta^0 * f$ for ImageNet [1] as in [3]. In addition $r^{\ell,i} = r^{\ell,i-1} + A^{\ell,i} * \sigma \circ B^{\ell,i} * \sigma(r^{i-1})$ is often called the basic ResNet block. Here, $A^{\ell,i}$ with $i \geq 0$ and $B^{\ell,i}$ with $i \geq 1$ are general 3×3 convolutions with zero padding and stride 1. In pooling block, $*_2$ means convolution with stride 2 and $B^{\ell,0}$ is taken as the 3×3 kernel with same output channel dimension of $R_{\ell}^{\ell+1}$ which is taken as 1×1 kernel and called as projection operator in [3]. During two consecutive pooling blocks, index ℓ means the fixed resolution or we ℓ-th grid level as in multigrid methods. Finally, R_{ave} ($R_{\max}$) means average (max) pooling with different strides which is also dependent on datasets and problems.

1.1.3 pre-act ResNet

The pre-act ResNet [3] shares a similar structure with ResNet.

Algorithm 3 h = pre-act ResNet($f; J, \nu_1, \cdots, \nu_J$)

Initialization: $r^{1,0} = f_{\mathrm{in}}(f)$.

for $\ell = 1 : J$ **do**

for $i = 1 : \nu_\ell$ **do**

Basic Block:

$$r^{\ell,i} = r^{\ell,i-1} + A^{\ell,i} * \sigma \circ B^{\ell,i} * \sigma(r^{\ell,i-1}). \tag{1.8}$$

end for

Pooling(Restriction):

$$r^{\ell+1,0} = R_{\ell}^{\ell+1} *_2 r^{\ell,\nu_\ell} + A^{\ell+1,0} \circ \sigma \circ B^{\ell+1,0} *_2 \sigma(r^{\ell,\nu_\ell}). \tag{1.9}$$

end for

Final average pooling layer: $h = R_{\mathrm{ave}}(r^{L,\nu_\ell})$.

Here pre-act ResNet share almost the same setup with ResNet.

The only difference between ResNet and pre-act ResNet can be viewed as putting a σ in different places. The connection of those three models are often shown with next diagrams:

Without loss of generality, we extract the key feedforward steps on the same grid in different CNN models as follows.

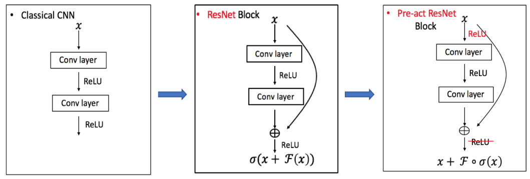


Fig. 1.1. Comparison of CNN Structures

Classic CNN

$$f^{\ell,i} = \xi^i \circ \sigma(f^{\ell,i-1}) \quad \text{or} \quad f^{\ell,i} = \sigma \circ \xi^i(f^{\ell,i-1}). \tag{1.10}$$

ResNet

$$r^{\ell,i} = \sigma(r^{\ell,i-1} + \xi^{\ell,i} \circ \sigma \circ \eta^{\ell,i}(r^{\ell,i-1})). \tag{1.11}$$

pre-act ResNet

$$r^{\ell,i} = r^{\ell,i-1} + \xi^{\ell,i} \circ \sigma \circ \eta^{\ell,i} \circ \sigma(r^{\ell,i-1}). \tag{1.12}$$

References

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[3] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pages 630–645. Springer, 2016.

[4] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[6] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[7] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.