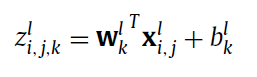
**The topics of Convolutional Neural Network (CNN)**

1. **Introduction**

Convolutional Neural Network (CNN) is an emerging deep learning architecture based on the visual perception of living creatures. In 1990, there was a modern framework of CNN published by Le-Cun et al. [1]. By this framework, LeNet-5 – a first successful application of CNN, was developed which could clarify the handwritten digits. This model has a multi-layer trained with back-propagation algorithm similar to other neural networks [2]. By acquiring representations of original images effectively, it can recognize visual pattern with little-to-none preprocessing [8]. In order to overcome the difficulties in training deep CNN, a study of Krizhevsky et al. [3] on AlexNet showed a significant development on classification of images. The architecture of AlexNet is similar to LeNet-5 except that the structure is deeper and bigger [8]. There are several works have been developed in order to improve the performance of AlexNet such as ZFNet [6], GoogleNet [5], VGGNet [4], ResNet [7], etc. The trend of increasing the depth in networks creates the increasing of complexity and makes it easier to overfitting as well as more difficult to optimize. However, it can improve the approximation to the target function of networks and feature representations [8].

1. **Architecture overview**

There are numerous CNN architectures. However, these are very similar in basic components. According to LeNet-5, there are three main types of layers including convolutional layer, pooling layer and fully-connected layer [8]. In the *convolutional layer*, the purpose is to learn the feature representations from the inputs. This layer consists of convolutional kernels computed the feature maps. Each neuron in the feature maps is linked to a region in the previous layers. By convolving input with s kernel, a new feature map can be created. After that, an element-wise nonlinear activation function will be applied on the convolved outputs [8]. The kernels can be shared by all spatial locations to produce each feature map, as a result, this can decrease model complexity as well as train network easily. An equation to calculate the feature value is as below [8]:



Where is bias, is weight vector, and are input patch center and feature value in the layer lth at location (i, j).

The activation function is to determine the nonlinear features desired for multi-layer networks. It introduces nonlinearities of CNN which is computed as based on the convolutional feature [8]. There are several typical activation functions such as ReLU [9], sigmoid, tanh [10].



In *the pooling layer*, the feature maps will be reduced the resolution to achieve shift-invariance. Each feature map in this layer is connected to previous feature map in convolutional layer. The pooling layer is usually set between two convolutional layers [8]. Therefore, the complex features of inputs can be developed by using multiple convolutional layers before operating of destructive pooling layer [11].

There are one or more *fully-connected layers* following convolutional and pooling layers in order to perform high level reasoning [4, 6, 12]. A global semantic information is created by connecting all neurons in previous layer to every neuron in fully-connected layer [8]. In addition, it is possible to replace a convolutional layer of instead of a fully-connected layer [8, 13]. The output of CNN with classification tasks can be released by using softmax operator or SVM to classify different tasks [3, 14, 15].

The architecture of LeNet-5 network is illustrated in the Figure 1. In this network, there are several convolutional layers composed of numerous kernels. In the first convolutional layer, the kernels are utilized to detect low-level features, e.g. edges, curves. In the higher layers, the kernels are trained to code more abstract features. The higher-level feature representations are extracted gradually by applying a stack of convolutional and pooling layers continuously [8].

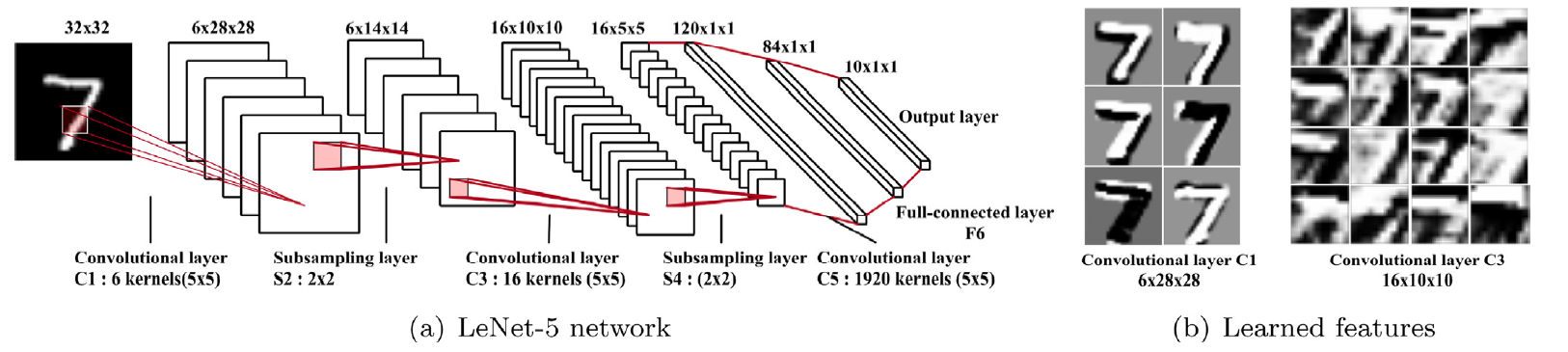


Figure 1: (a) The architecture of the LeNet-5 network, which works well on digit classification task. (b) Visualization of features in the LeNet-5 network. Each layer's feature maps are displayed in a different block [8].

1. **Typical applications of CNN**

There are numerous applications apply the CNN.

1. Image classification

CNN have been utilized in image classification [8]. Since CNN can connect classifier learning and feature, it can attain the better accuracy in classification on large scale datasets in compared with other techniques [8, 9, 16]. There are many works have been conducted in order to improve the classification accuracy significantly such as decreasing filter size and widening the network depth [9, 10,11].

Furthermore, CNN have been applied in object detection. As a long-standing and crucial issue in computer vision, object detection requires the locate the objects in accurately and efficiently in images or video frames [8].

**References**

1. B.B. Le Cun, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, “Handwritten digit recognition with a back-propagation network”, Proceedings of the Advances in Neural Information Processing Systems (NIPS), pp. 396 – 404, 1989.
2. R. Hecht-Nielsen, “Theory of the backpropagation neural network”, Neural Networks1 (Supplement-1), pp. 445 – 448, 1988.
3. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, “Imagenet large scale visual recognition challenge”, Int. J. Conflict Violence (IJCV) 115, vol.3, pp. 211 – 252, 2015.
4. K. Simonyan, A. Zisserman, “Very deep convolutional networks for large-scale image recognition”, Proceedings of the International Conference on Learning Representations (ICLR), 2015.
5. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, “Going deeper with convolutions”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1 – 9, 2015.
6. M.D. Zeiler, R. Fergus, “Visualizing and understanding convolutional networks”, Proceedings of the European Conference on Computer Vision (ECCV), pp. 818 – 833, 2014.
7. K. He, X. Zhang, S. Ren, J. Sun, “Deep residual learning for image recognition”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770 – 778, 2016.
8. J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, T. Chen, “Recent advances in convolutional neural networks”, Pattern Recognition, vol. 77, pp. 354 – 377, 2018.
9. V. Nair, G.E. Hinton, “Rectified linear units improve restricted Boltzmann machines”, Proceedings of the International Conference on Machine Learning (ICML), pp. 807 – 814, 2010.
10. Y.A. LeCun, L. Bottou, G.B. Orr, K.-R. Müller, “Efficient backprop”, Neural Networks: Tricks of the Trade - Second Edition, pp. 9 – 48, 2012.
11. Convolutional neural network for Visual Recognition lecture, CS231N – Standford University, <http://cs231n.github.io/convolutional-networks/>.
12. G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R. R. Salakhutdinov, “Improving neural networks by preventing co-adaptation of feature detectors”, CoRR abs/1207.0580, 2012.
13. M. Lin, Q. Chen, S. Yan, “Network in network”, Proceedings of the International Conference on Learning Representations (ICLR), 2014.
14. Y. Tang, “Deep learning using linear support vector machines”, Proceedings of the International Conference on Machine Learning (ICML) Workshops, 2013.
15. G. Madjarov, D. Kocev, D. Gjorgjevikj, S. Džeroski, “An extensive experimental comparison of methods for multi-label learning”, Pattern Recognition vol.45 (9), pp. 3084 – 3104, 2012.
16. M. Everingham, S.A. Eslami, L. Van Gool, C.K. Williams, J. Winn, A. Zisserman, “The pascal visual object classes challenge: a retrospective”, Int. J. Conflict Vio- lence (IJCV) vol. 111 (1), pp. 98–136, 2015.