A Comprehensive Research of the Development of Classical Convolutional Neural Networks

Changli Tao

Faculty of Engineering, The Hong Kong Polytechnic University, Hong Kong SAR, 999077, China
22102825d@connect.polyu.hk

Abstract. Since 2010, with the rapid emergence of deep learning, Convolutional Neural Networks (CNNs) have made significant progress across various domains. In particular, advancements in CNNs have profoundly impacted the field of computer vision, resulting in substantial improvements in tasks such as image classification, object detection, and segmentation. However, as task complexity increases and dataset sizes expand, traditional CNN models face a series of challenges. In response to these obstacles, researchers have devised multiple enhancements and optimization strategies from different perspectives and directions, fostering ongoing developments in structural design and model performance. This paper offers a comprehensive investigation into the evolution of CNNs. The study begins by introducing the standard architecture of CNNs, followed by a delineation of the three significant developmental stages that CNNs have undergone: 1) Traditional Architecture Network, 2) Connectivity-Enhanced Network, and 3) Hybrid Optimization Network. Furthermore, this paper conducts an exhaustive comparison and evaluation of representative models from each stage. Finally, promising directions for CNNs are identified to guide future research endeavors.

Keywords: Deep Learning, CNN, Model Architecture.

1 Introduction

Image processing and recognition play a vital role in the digital age. With the generation and widespread application of large volumes of image data, there is an increasing demand for accurate and efficient image analysis and understanding. In this context, Convolutional Neural Networks (CNNs), as powerful deep learning algorithms, have emerged as crucial tools and have achieved significant success. By progressively extracting and abstracting features from the input training data, CNNs can learn hierarchical feature representations. These representations capture local patterns and global context information, enabling high-performance image classification, target detection, and semantic segmentation.

Initially, CNN originated from the simulation of visual perception mechanisms in living organisms. As early as 1962, Hubel discovered receptive field cells in the

Y. Wang (ed.), *Proceedings of the 2024 2nd International Conference on Image, Algorithms and Artificial Intelligence (ICIAAI 2024)*, Advances in Computer Science Research 115,
https://doi.org/10.2991/978-94-6463-540-9_96

visual cortex of the cat brain [1]. These cells are responsible for extracting local spatial correlations in images, enabling preliminary processing and analysis of visual information. This pivotal finding laid the foundation for the subsequent invention of computer neural networks. In 1980, Fukushima proposed a multi-layer artificial neural network model called neurocognition [2]. Inspired by neurocognition, LeCun utilized the backpropagation algorithm to design and train a classic CNN based on gradient learning, known as LeNet-5, which successfully achieved recognition of handwritten digits [3]. Following its emergence, subsequent research in the field focused on refining different aspects of LeNet-5.

This research conducts a comprehensive review, comparison, and analysis of representative models in the development process of CNNs. By systematically summarizing and analyzing the pivotal research findings in model evolution, this article provides a concise and thorough perspective on the current state of CNN development, assisting readers in quickly grasping the breakthroughs in this domain.

2 Standard Architecture for CNNs

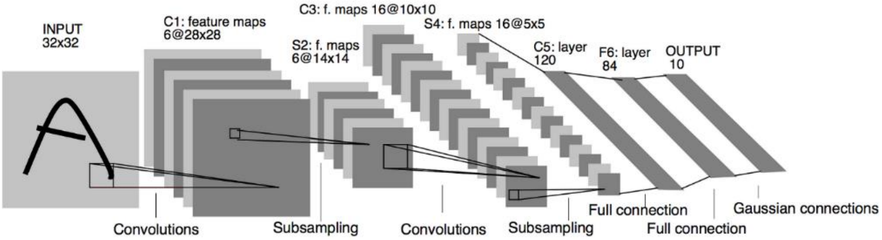

Fig. 1. A standard CNN architecture (based on LeNet-5) .

LeNet-5, which LeCun first proposed in 1998, is the initial CNN model [3]. Although the architecture is relatively simple, the major modules used for feature extraction and learning are complete. Its architecture has also become the standard architecture referenced by many subsequent CNNs. As depicted in Fig. 1, a typical CNN consists of several key components, including the input layer, convolutional layer, pooling layer, fully connected layer, and output layer.

Firstly, convolutional layers, along with multiple convolutional kernels, play a crucial role in generating feature maps from the input image. Through the sliding and operation of convolution kernels on the input image, effective extraction of image features can be achieved. Secondly, the pooling layer downsamples the feature map to reduce the data dimension and computational effort. Common downsampling methods, such as Max Pooling or Mean Pooling, are employed to extract the maximum or average value within a pooling window. Subsequently, the fully connected layer summarizes local information from the previous layer to form global information. It often employs the Rectified Linear Unit (ReLU) activation function to enhance network performance and incorporates dropout techniques to prevent overfitting. Finally, the output layer utilizes diverse activation functions to process the output, thereby matching the requirements of different types of tasks.

In general, the training objective of CNNs is to optimize the network weights with the backpropagation algorithm, aiming to minimize the loss function while mitigating the risk of overfitting. Therefore, the selection of an appropriate loss function holds paramount importance in achieving optimal results. Among the commonly employed loss functions, cross-entropy loss is typically favored for classification tasks, while mean squared error and mean absolute error find utility in regression tasks. Additionally, image segmentation tasks often benefit from the adoption of the Dice loss, whereas smooth L1 loss can be advantageous for object detection tasks. By leveraging the progressive operations of convolution, pooling, and fully connected layers, CNNs can effectively extract and integrate features, thereby supporting efficient image recognition and processing.

3 The Evolution of CNNs

3.1 Traditional Architecture Network (CNNs based on Standard Architecture)

AlexNet. AlexNet, proposed by Krizhevsky in 2012 as an extension of the LeNet-5 architecture, stands as a milestone in the development of CNNs [4]. Compared to LeNet-5, AlexNet proposed a series of major innovative measures. Notably, it employs ReLU as the activation function. In contrast to the Sigmoid function used by LeNet-5, ReLU offers advantages such as simplified computations, faster convergence, and improved mitigation of the vanishing gradient problem. Additionally, AlexNet incorporated the Dropout technique, randomly discarding neurons in the fully connected layer to effectively prevent overfitting. Furthermore, by employing overlapping MaxPooling and data augmentation techniques, AlexNet enriched feature extraction and bolstered the model's generalization capabilities. These optimization methods led to a significant refinement in the performance of CNNs. Consequently, AlexNet achieved a Top-5 error rate of 15.3% on the ILSVRC 2012 dataset, surpassing the second-place model by a large margin of 26.2% [4].

VGGNet. Following the groundbreaking success of AlexNet, subsequent advancements were made in the field of CNNs. One notable contribution came from Simonyan with the introduction of VGGNet [5]. VGGNet presented novel approaches by replacing large convolution kernels with a series of smaller ones and increasing the depth of the network. This architectural modification allowed VGGNet to learn more intricate and detailed features while addressing the parameter explosion issue associated with larger kernels.

GoogLeNet. Furthermore, the following innovation in convolutional approaches should not be overlooked. In 2015, Szegedy introduced GoogLeNet, which featured the Inception Module as its core structure [6]. The distinctive characteristic of the

Inception Module is the integration of multiple parallel branches into a single layer. Each branch employs convolution kernels of varying sizes, such as 1x1, 3x3, and 5x5 convolutions, to capture information at different scales. Moreover, additional 1x1 convolutions are employed to reduce the dimensionality of feature maps. By employing this branch convolution followed by merging approach, GoogLeNet further enhances the network's width and its ability to perceive features at different scales.

Overall, the innovative contributions of the above three classic CNNs, including the selection of activation functions, the choice of convolutional kernel sizes, and advancements in convolutional methods, have propelled enhancements in the performance and computational efficiency of traditional architecture networks.

3.2 Connectivity-Enhanced Network (CNNs based on Cross-Layer Connection)

ResNet. As the traditional architecture networks have evolved, it has been observed that a deep network formed by directly stacking multiple shallow layers often fails to leverage the theoretically powerful feature extraction capabilities of deep networks, while leading to an anomalous decline in model performance. However, the issue is not attributed to overfitting, as the performance of deeper networks deteriorates even on the training set.

In 2016, another groundbreaking model in the evolution of CNNs named ResNet was introduced by He, effectively tackling the issue of network degradation that arises with the increasing depth in traditional architecture networks [7].

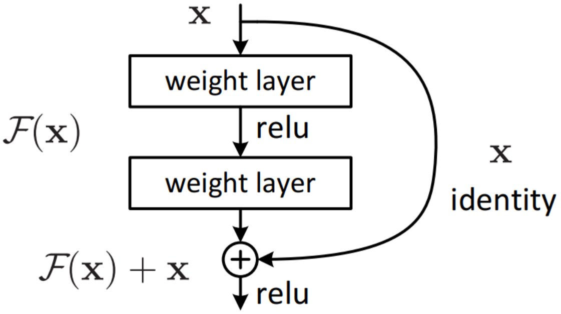
x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Fig. 2. Residual block.

The core of ResNets is the small unit called residual block. The residual block introduces an innovative structure that incorporates skip connections into the standard architecture, connecting across one or more layers. Additionally, it incorporates two different types of mappings: Residual Mapping and Identity Mapping. As shown in Fig. 2, the residual mapping refers to the intermediate $\mathcal{F}(x)$, while the identity mapping corresponds to the curve on the right representing x. Therefore, a typical underlying mapping of the residual block can be expressed as:

$$\mathcal{H}(x) = \mathcal{F}(x, \{W_i\}) + x \tag{1}$$

Where x is the input to the residual block (also the output from the previous layer or residual block), $\mathcal{F}(x,\{W_i\})$ stands for the residual function (typically composed of several convolutional layers and non-linear activation functions), and W_i represents the weights of the layers. By adding the output x from a previous layer to the residual function $\mathcal{F}(x)$, this unique residual connection enables direct information transfer within the residual block.

Theoretically, if the network has reached an optimal state, further increasing the network depth might cause the residual function of the new residual blocks to approach zero, i.e., $\mathcal{F}(x,\{W_i\}) \approx 0$. At this point, the output of the residual block would essentially be the input itself, i.e., $\mathcal{H}(x) \approx x$. This implies that the newly added layers would not significantly affect the output. Therefore, the performance of networks would not degrade significantly with increased depth. By introducing the residual block structure with skip connections, ResNet effectively addresses the network degradation problem that plagues traditional architecture networks, propelling the performance and development of CNNs to a new level.

DenseNet. Residual networks and other stochastic depth training methods share a common characteristic, which is the creation of cross-layer information flow within the network. Subsequently, to guarantee optimal information flow between layers, Huang proposed DenseNet [8]. Generally, a DenseNet is composed of multiple dense blocks, where all layers are directly connected. In a densely connected architecture, each layer receives input from all preceding layers and passes its feature map as input to every subsequent layer, facilitating comprehensive information flow throughout the network.

In comparison with residual connections, the primary innovation of dense connections resides in the approach employed for feature map combination. Instead of using summation in the residual block, the dense block directly merges the feature maps through channel concatenation. By employing dense connections, DenseNet can adequately transfer and reuse feature information from previous layers throughout the network. For this reason, it alleviates the gradient vanishing problem and promotes feature propagation, enabling the model to learn more diverse and intricate feature representations. Moreover, due to direct access to feature maps from all previous layers, the overall parameter quantity of the model is also reduced.

3.3 Hybrid Optimization Network (CNNs Integrated with Transformers)

Since 2017, Transformer-based models proposed by Vaswani have experienced rapid development in the field of computer vision [9]. Considering this, Liu introduced a hybrid optimization model called ConvNeXt by integrating some structural and parameter characteristics of the Swin Transformer into a pure ConvNet (ResNet-50) [10]. By adopting ResNet-50 as a baseline and training it with techniques similar to those utilized to train visual Transformers, Liu discovered that many architectural choices from Transformers can be incorporated into CNNs, leading to improved performance.

Liu explored various adjustments to the CNN architecture in light of Transformers, such as modifying the convolutional kernel size, minimizing the number of activation and normalization layers, introducing a separate downsampling layer, and substituting batch normalization (BN) with layer normalization (LN), among other modifications. Through a series of structural and parameter experiments, ConvNeXt ultimately achieved an accuracy rate of 82.0% on ImageNet-1K, surpassing the performance of Swin Transformer (81.3%). Furthermore, these experiments implicitly demonstrated that excessive non-linear mappings may introduce an excessive level of complexity, hampering the effective learning and representation of crucial features by the network.

4 Summary Analysis of Classic CNNs

Table 1. Comparative analysis of pros & cons and innovations of different CNNs.

Model	Year	Innovations	Advantages	Limitations
LeNet-5 [3]	1998	1. Introducing the backpropagation algorithm 2. Inventing standard CNN architecture	Simple network architecture makes it easy to understand and implement	Shallow network depth makes it impossible to fully capture high-level semantic features
AlexNet [4]	2012	1. Activation function changed from Sigmoid to ReLU 2. Dropout technique is used in fully connected layer to prevent overfitting 3. Using overlapping max pooling and data augmentation technique	1. Deep architecture and overlap pooling make AlexNet have powerful feature extraction capability 2. ReLU activation function significantly accelerates the training speed and lessens the gradient vanishing problem	The high complexity of the model and the large number of parameters (more than 600,000 parameters in total) make it costly to compute and store
VGGNet [5]	2014	1. Replacing larger convolutional kernels with consecutive 3×3 convolutional kernels 2. Introducing very deep networks such as VGG-16 and VGG-19	The combination of deep network architecture and small convolutional kernels makes it powerful for feature extraction and representation	1. Deeper networks lead to higher computation and storage costs 2. Network degradation problem that occurs with the increase of network depth
GoogLeNet [6]	2015	1. Using the Inception module to extract features with multiple scales from the same layer 2. Reducing the dimensionality with 1x1 convolution 3. Employing global average pooling to replace the fully-	1. The computational efficiency is significantly improved by the design of 1x1 convolution and Inception module 2. Multi-scale feature extraction enables GoogLeNet to excel in complex image recognition tasks	The complex design of the Inception module increases the difficulty of model implementation and debugging

		connected layer		
ResNet [7]	2016	1. Inventing the residual block structure and adding skip connections to the network 2. Introducing identity mapping and residual mapping	Skip connections significantly mitigated the network degradation problem, allowing gradients to propagate efficiently and permitting the training of ultra-deep networks	Although skip connections alleviate gradient vanishing, for ultra-deep networks, they may still face overfitting problems and require effective regularization
DenseNet [8]	2017	1. Inventing the dense block structure 2. Dense connections allow features to be maximally reused throughout the network	1. Enhanced feature propagation 2. Mitigated the vanishing gradients problem 3. Reduced the number of parameters	High memory consumption and computational cost due to densely connected layers
ConvNeXt [10]	2022	1. Integrating the advantages of CNNs and Transformer 2. Reducing normalization layers and activation functions	The model performance has further improved by drawing on and integrating the architectural design and hyperparameter selection of Swin Transformer	The performance of ConvNeXt can be sensitive to hyperparameters, requiring more discreet fine-tuning

In summary, the development of CNNs has undergone three predominant stages: Traditional Architecture Networks, as exemplified by LeNet-5; Connectivity-Enhanced Networks, represented by ResNet; and the current Hybrid Optimization Networks, exemplified by ConvNeXt. Table 1 provides a summary of the innovations introduced by these classic CNN models compared to their predecessors, along with their respective advantages and limitations. It concisely depicts the continuous innovation and evolution of CNNs with respect to model design and performance optimization.

5 Conclusion

This paper conducts comprehensive research on the canonical architecture and evolutionary history of CNNs. It summarizes the three major stages in the development of CNNs and provides detailed descriptions of representative models from each stage. Currently, CNNs continue to demonstrate untapped potential for further advancement. Future studies can concentrate on the following key areas.

First, model architecture innovation: exploring new network structures, such as hybrid convolutions and integrating Transformers with self-attention mechanisms, to boost the model's expressive power and efficiency. Second, optimization algorithms: introducing more advanced optimization algorithms and training strategies, such as adaptive learning rate adjustment and mixed precision training, to accelerate the training process and improve model performance. Third, multi-modal fusion:

combining multiple data modalities, such as images, text, and videos, to advance the model's understanding ability and expand its application scope.

References

1. Hubel, D. H., Wiesel, T. N.: Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of Physiology, 160(1), 106 (1962)
2. Fukushima, K.: Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics, 36(4), 193-202 (1980)
3. LeCun, Y., Bottou, L., Bengio, Y., et al.: Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324 (1998)
4. Krizhevsky, A., Sutskever, I., Hinton, G. E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25 (2012)
5. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
6. Szegedy, C., Liu, W., Jia, Y., et al.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-9 (2015)
7. He, K., Zhang, X., Ren, S., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778 (2016)
8. Huang, G., Liu, Z., Van, D. M. L., et al.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700-4708 (2017)
9. Vaswani, A., Shazeer, N., Parmar, N., et al.: Attention is all you need. Advances in neural information processing systems, 30 (2017)
10. Liu, Z., Mao, H., Wu, C. Y., et al.: A convnet for the 2020s. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11976-11986 (2022)