Image classification optimization models using the convolutional neural network (CNN) approach and embedded deep learning system

Akbar Payandan, S. Hossein Hosseini Nejad

Faculty of Engineering, Ahar Branch, Islamic Azad University, Ahar, Iran,

Email: payandan7393@gmail.com, S.hosseininejad@gmail.com

Abstract

Deep learning has progressed rapidly in recent years and has been applied in many fields, which are the main fields of artificial intelligence. Traditional methods of machine learning most use shallow structures to deal with a limited number of samples and computational units. When the target objects have rich meanings, the performance and ability to generalize complex classification problems will be quite inadequate. The convolutional neural network (CNN), which has been developed in recent years, widely used in image processing; because it has high skills in dealing with image classification and image recognition issues and it has led to great care in many machine learning tasks and it has become a powerful and universal model of deep learning. The combination of deep learning and embedded systems has created good technical dimensions. In this paper, several useful models in the field of image classification optimization, based on convolutional neural network and embedded systems, are discussed. Since this paper focuses on usable models on the FPGA board, models known for embedded systems such as MobileNet, ResNet, ResNeXt and ShuffNet have been studied.

Keywords: Artificial Intelligence, Deep Learning, Image Classification, Convolution Neural Network, Deep Learning Algorithm.

1. Introduction

Today, deep learning technology is evolving rapidly and is used in many industrial fields. Deep learning, as part of machine learning, [1, 2] has become an important academic area since 2006. Deep learning is used in areas such as image classification and voice recognition with raw data [3]. Image recognition is a special part of computer vision that obtains outstanding information from images [4]. Computer vision is used in many areas, such as identifying traffic signs and optical character recognition. Most computer vision work is done on GPUs not only images but also pixels can be run in parallel on them. Image recognition in GPUs performs well, however, it limits tasks to computers or workstations.

Compared to desktop PCs, embedded systems (such as FPGAs) have lower power consumption, smaller size, and lower unit cost. These types of systems are used in several fields such as robots and smartphones. Given the growing needs of computer vision technology to be applied in FPGAs, combining the visual benefits of a computer

with an embedded system is a great way to solve problems in a variety of areas.

FPGA is a well-established operating system. Compared to microprocessors, FPGA has good flexibility, it can be rewritten several times, and it is fast to read [5].

An embedded system, such as an FPGA board, is a flexible platform for specific tasks and it is physically very small compared to a traditional deep learning platform such as a workstation. The combination of deep learning and embedded system provides a great aspect for development. Deep learning is more focused on software design because it is not allowed to change high quality hardware and framework. However, there is no perfect way to apply a high-performance deep learning network to the target embedded system.

It is important to study previous work on CNN models. In recent years, several models have been designed for CNN, such as AlexNet, GoogleNet, VGG-16, etc. [6]. Models can be evaluated using some features. These models cost less space for embedded systems, in which case there is a memory [7] on the board.

2. Research literature

In a study conducted by Tu and colleagues (2019) efficient DNN performance was evaluated and FPGA was used for fully connected layer and GPU was used for floating point operation aiming to use specialized devices such as FPGAs and GPUs in heterogeneous computations to accelerate deep learning computations with energy efficiency constraints. The proposed heterogeneous framework idea was implemented using the Nvidia TX2 GPU and the Xilinx Artix-7 FPGA. Experimental results show that the proposed framework can be calculated faster and much less energy consumption [8.[

Another study was conducted by Shawahna et al. (2019) to investigate FPGA-based accelerators and Deep Learning Networks to learning and classification, recent techniques to accelerate FPGA deep learning networks. The researchers identified the main features used in various techniques to improve acceleration performance. In addition, they provided recommendations for increasing the use of FPGAs to accelerate CNN. The techniques studied in this paper reflect recent trends in FPGA-based accelerators of deep learning networks [9.[

Monafi (2019) evaluated and analyzed deep learning in remote sensing by three different strategies. The experiments were performed using a remote sensing data set as well as the well-known fine-tuned neural networks. The results show that well-regulated convulsive neural networks have the best performance among the strategies. Using the features of convolutional neural networks well complemented by Linear SVM gives the best results. In fact, the best results are obtained by simultaneously using the features of well-tuned convolution neural networks with linearly tuned SVMs [10].

3. Convolution Neural Network (CNN)

Deep learning uses machine learning skills to produce multiple layers of nonlinear operations structure [11].

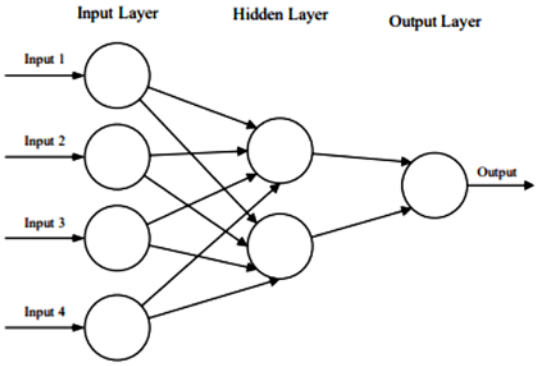

Fig.1. Model of artificial neural network derived from [12]

Deep learning is inspired by the structure and function of biological brains [13]. The deep learning model mimics the characteristics of neurons and their connections. Artificial neural network (ANN) is a common implementation method for deep learning. An ANN model consists of an input layer, a hidden layer, and an output layer, as shown in Figure (1).

In the field of neural networks, the most useful models are: Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN). RNN [14] is useful in natural language processing and language comprehension. CNN [15] works well in machine vision, like image recognition.

CNN has two main stages: training and inference. Training stage; It uses the Back propagation algorithm to update the model parameters and improve the forecast accuracy based on the target data set.

In the actual use of CNN training and experimentation, the data are controlled by tensors that are a multidimensional mathematical object with specific change rules [16]. Table (1) shows the tensors transferred between the layers and the parameters used in CNN. A batch means that several images are combined together as a map of the input features.

Table (1): Tensors in CNN inferred with the involved parameters [6]

Symbol	Description	Content
B	Batch size (number of input frames)	
W/H/C	Weight / height / depth of input FM	
U/V/N	Weight / height / depth of output FM	
K/J	Vertical / horizontal kernel size	
X	Input FMs	B×C×H×W
Y	Output FMs	B×N×V×U
Θ	Trained filters	N×C×J×K
β	Trained biases	N

CNN has a Feed Forward back propagation structure on several types of layers, including convolution, activation, merge, fully connected, and batch activation layers. Figure (2) shows a simple example of Feed Forward propagation that includes a layer of convolution, activation, and integration.

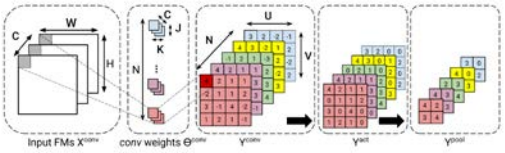

Fig.2.An example of Feed Forward back propagation in inference [6]

4- CNN models

Designing and developing CNN models takes a lot of time. In this paper, the usable models on the FPGA board are studied, which are MobileNet, ResNet, ResNeXt and ShuffNet. The main differences between the different models are in their convolution layers.

The cost of convolution layers is obtained from two parts: spatial connection and channel connection. As shown in Figure (3), if the convolution core size is 3 × 3, each pixel is spatially connected to the other three pixels with width and height dimensions. At the same time, in terms of channels, each input channel (depth) is connected to all output channels. Due to this feature, with the increase in FM size in all dimensions, the cost of connecting channels is a major part.

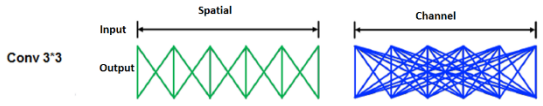

Fig.3. Spatial connections and normal convolution channel taken from [17].

As shown in Figure (4), pointwise convolution, grouped convolution, and channel shuffling are applied to improve the cost of the main network cost. The computational cost has some parameters: height (H), width (W), input channel (N), output channel (M), kernel size (K × K) and group number (G). According to [18], Conv 1 × 1 is also called point convolution.

Its kernel size is 1. 1. For this reason, spatial connection is easier than main convolution. Convolution function is the point of change of channel size.

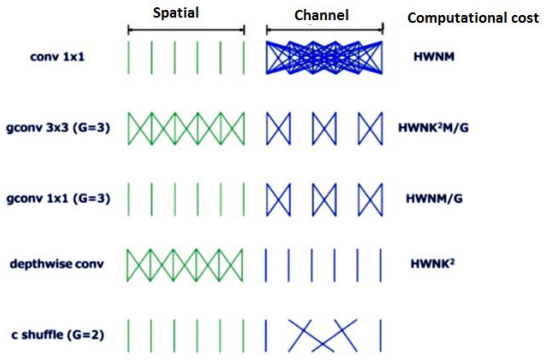

Fig.4. Different methods of convolution taken from [17].

Gconv 3×3 $(G = 3)$ is the convolution is grouped with kernel size 3×3 and group number 3. The difference with main convolution is that the channels are independent of the groups. This does not change the spatial connection, it only reduces the cost of calculating the channel. Gconv 1×1 $(G = 3)$ only spatially different from gconv 3×3 $(G = 3)$.

Depthwise convolution has the same channel number for FM input and FM output. All channels are in a separate group of size 1 and separate all channels from the connections, which reduces the cost of calculation in the channel range [19, 20, 21]. C Shuffle has no calculation. Just sorts the channels in a different order. For example, when group number 2 is a new channel list, both channels copy the list of incoming channels.

4-1-ResNet

A deep residual learning network stands for ResNet. The residual unit has a bottleneck structure as shown in Figure (5). The middle layer channel number is not necessarily 2, but

according to [21] the remaining unit is always smaller than the FM input and FM output.

Apart from the unique layer structure, the remaining network simultaneously with the convergence of deep networks [22] solves the problem of degradation. Figure (6) shows the structural idea of the additional structure. This map passes the input feature through the two-layer training convolution unchanged, and the map adds the main feature with the output.

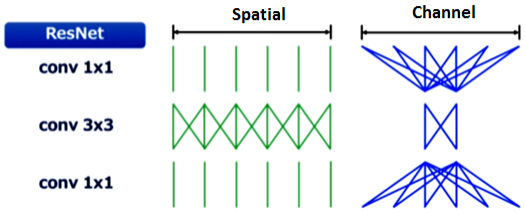

Fig.5. ResNet structure taken from [17].

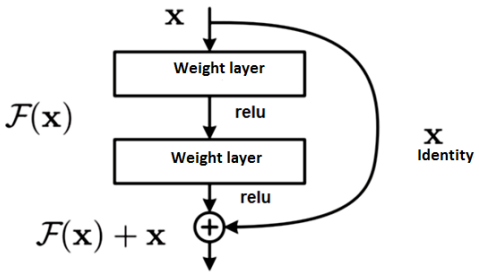

Fig.6.A sample of the resifual learning block taken from [23].

4-2- ResNeXt

ResNeXt is an improved version of the ResNet model. This model changes the middle convolution to the group convolution. ResNeXt has been shown to have better classification accuracy than ResNet with the same computational cost. As shown in Figure (7) and Figure (5), if the channel number is set evenly (n), the ResNeXt connection number is smaller than the ResNet; as shown in Equation (1).

$$N_{ResNet} = n^2 > N_{ResNeXt} = \left(\frac{n}{2}\right)^2 = \frac{n^2}{2} \qquad (1)$$

Because channel connectivity affects resource costs, ResNeXt is better than ResNet in consuming resources with similar parameters. In addition, ResNeXt has the remaining learning part of adding an identity input feature map to the output feature map.

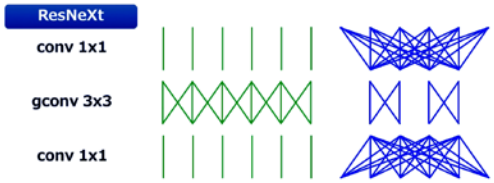

Fig.7. ResNeXt structure taken from [17].

4-3- MobileNet

MobileNet has two parts: deep convolution and point convolution, as shown in Figure (8) [24]. This model is separated from the main convolution. Conversion depthwise does not change the channel number. When the parameters are the same, the computational cost of the main convolution is $HWNMK^2$, but the computational cost of MobileNet is equal to $HWN(M + K^2)$. In this example it is K = 3. M is always 32, 64 or greater. Therefore, M is much larger than K^2. By ignoring K^2, MobileNet costs are reduced to about 1.9 main convolution.

Fig.8.MobileNet structure taken from [17].

Conclusion

Among the types of neural networks, convolutional neural networks (CNNs) usually provide good accuracy in classifying images. In this paper, deep learning neural network models are studied in order to optimize the classification of usable image on the FPGA board, which is known for embedded systems such as MobileNet, ResNet, ResNeXt and ShuffNet.

According to the results of the study, it is suggested that in a simulation-based comparative study to test the performance of the studied models, using different data sets to show what results these models will have on practical events in the field of image classification and which results model will be more successful.

.

References

[1] "Li Deng and DongYu. “Deep Learning: Methods and Applications”. In: Foundations and Trends ."

[2] "Yoshua Bengio. “Learning Deep Architectures for AI”. In: Foundations and Trends in Machine Learning 2.1 (2009), pp. 1–127. issn: 1935-8237 .".

[3] "Keiron O’Shea and RyanNash. “An Introduction to ConvolutionalNeuralNetworks”. In: CoRR abs/1511.08458 (2015). arXiv: 1511.08458.url: http://arxiv.org/abs/1511.08458 .".

[4] "Dana H. (Dana Harry) Ballard and 1945- Brown Christopher M. Computer vision. English. Includes bibliographies and indexes. Englewood Cliffs, N.J. : Prentice-Hall, 1982. isbn: 0131653164. url: http://homepages.inf.ed.ac.uk/rbf/BOOKS/BANDB/bandb.htm .".

[5] "Lee Cloer. FPGA vs Microcontroller – Advantages of Using An FPGA.2017. url: https : / / duotechservices . com / fpga - vs -microcontroller-advantages-of-using-fpga .".

[6] "Kamel Abdelouahab et al. “Accelerating CNN inference on FPGAs: A Survey”. In: CoRR abs/1806.01683 (2018). arXiv: 1806.01683. url:http://arxiv.org/abs/1806.01683 .".

[7] "Andrew G. Howard et al. “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications”. In: CoRR abs/1704.04861 (2017). arXiv: 1704.04861. url: http://arxiv.org/abs/1704.04861 .".

[8] "Y. Tu, S. Sadiq, Y. Tao, M. Shyu and S. Chen, "A Power Efficient Neural Network Implementation on Heterogeneous FPGA and GPU Devices," 2019 IEEE 20th International Conference on Information Reuse and Integration for Data Science (IRI), Los Angeles, CA, US ."

[9] "A. Shawahna, S. M. Sait and A. El-Maleh, "FPGA-Based Accelerators of Deep Learning Networks for Learning and Classification: A Review," in IEEE Access, vol. 7, pp. 7823-7859, 2019, doi: 10.1109/ACCESS.2018.2890150 .".

[10] "Manafi Bojoshin, Safa, 2019, Classification of Remotely Recorded Images Using CNN Deep Learning Algorithm, Annual National Congress of New Research Ideas in Engineering and Technology, Electrical and Computer Science, Sari, Target Higher Education Institu ."

[11] "Li Deng and DongYu. “Deep Learning: Methods and Applications”.," In: Foundations and Trends in Signal Processing 7.3–4 (2014), pp. 197–387. issn: 1932-8346 ..

[12] "Keiron O’Shea and RyanNash. “An Introduction to ConvolutionalNeuralNetworks”. In: CoRR abs/1511.08458 (2015). arXiv: 1511.08458. url: http://arxiv.org/abs/1511.08458 .".

[13] "Adam H. Marblestone, Greg Wayne, and Konrad P. Kording. “Toward an Integration of Deep Learning and Neuroscience”. In: Frontiers in Computational Neuroscience 10 (2016), p. 94. issn: 1662-5188 .".

[14] "Rafal Józefowicz et al. “Exploring the Limits of Language Modeling”. In: CoRR abs/1602.02410 (2016). arXiv: 1602.02410 .".

[15] "S. Haykin and B. Kosko. “GradientBased Learning Applied to Document Recognition”. In: Intelligent Signal Processing. IEEE, 2001. isbn:9780470544976 .".

[16] "Todd Rowland and EricW.Weisstein. Tensor. url: http://mathworld.wolfram.com/Tensor.html .".

[17] "Why MobileNet and Its Variants (e.g. ShuffleNet) Are Fast. url: https://medium.com/@yu4u/why-mobilenet-and-its-variantse-g-shufflenet-are-fast-1c7048b9618d .".

[18] "Min Lin, Qiang Chen, and Shuicheng Yan. “Network In Network”. In: arXiv e-prints, arXiv:1312.4400 (Dec. 2013), arXiv:1312.4400. arXiv:1312.4400 [cs.NE .".[

[19] "Laurent Sifre. “Rigid-Motion Scattering For Image Classification”. PhD thesis. CMAP, 2014 .".

[20] "Laurent Sifre and Stephane Mallat. “Rotation, Scaling and Deformation Invariant Scattering for Texture Discrimination”. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). June 2013 .".

[21] "Francois Chollet. “Xception: Deep Learning With Depthwise Separable Convolutions”. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). July 2017 .".

[22] "Kaiming He et al. “Deep Residual Learning for Image Recognition”.In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). June 2016 .".

[23] "Kaiming He et al. “Deep Residual Learning for Image Recognition”.In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). June 2016 .".

[24] "Andrew G. Howard et al. “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications”. In: CoRR abs/1704.04861 (2017). arXiv: 1704.04861. url: http://arxiv.org/abs/1704.04861.".