Багатокритеріальна оптимізація в задачах структурно-параметричного синтезу згорткових нейронних мереж

Віктор Синєглазов
Національний Авіаційний Університет
Київ, Україна
svm@nau.edu.ua

Бориндо Ілля
Національний Авіаційний Університет
Київ, Україна
ibo.mistle@gmail.com

Multicriteria optimization for structural parametric synthesis of convolutional neural networks

Viktor Sineglazov
National Aviation University
Kyiv, Ukraine
svm@nau.edu.ua

Boryndo Illia
National Aviation University
Kyiv, Ukraine
svm@nau.edu.ua

***Анотація*—В роботі визначено перспективний клас згорткових нейронних мереж, а також розглянуті їх ключові параметри для подальшого структурно-параметричного синтезу. Показано, що ці мережі повинні включати крім традиційних компонентів (згорточні шари, об'єднуючі шари, шари прямого поширення, додаткові шари: шар пакетної нормалізації, згортковий шар 1x1, шар відсіву, залишковий блок, тощо) також і функціональні структірні блоки (SRU, CRU, блок щільної залишкової уваги, тощо). Пропонується застосування генетичного алгоритму для структурно-параметричного синтезу з використанням розглянутих шарів та структурних блоків.**

***Abstract*— The paper defines a promising class of convolutional neural networks and considers their key parameters for further structural and parametric synthesis. It is shown that these networks should include, in addition to traditional components (convolutional layers, pooling layers, feed-forward layers, additional layers: batch normalization layer, 1x1 convolutional layer, dropout layer, etc), also functional structural units (SRU, CRU, dense residual attention unit, etc). We propose to use a genetic algorithm for structural-parametric synthesis using the considered layers and structural blocks.**

Ключові слова—структурно-параметричний синтез; згорткові нейронні мережі; генетичний алгоритм

Keywords— structural-parametric synthesis; convolutional neural networks; genetic algorithm

I. Introduction

Nowadays, in modern world fully filled with image processing tasks as well as the application of CNNs to solve them, the problems with lack of accuracy, low performance and network over-complexity issues occurs. The urgency of this problem over time is only increasing due to the proliferation of the problem of digital identification.

Most of CNN architectures are highly restricted due to their performance results, low learning rates and at the same time they require a big number of high quality training materials. In order to solve this problem and increase results the complexity of convolutional neural networks were only increasing over the years. But such development leads to new problems when further CNN enrichment bumps into hardware limitations. In such conditions hybrid CNNs takes the place. To increase overall performance and accuracy the group of CNNs can be combined to form hybrid convolutional neural network. In this paper we will describe and present the research results on the usage and topology features of hybrid convolutional neural networks (HCCN)[3] as well as different structural blocks that should be used for their synthesis processes. It includes performance research of each structural blocks, analysis of modern CNN topologies and their application using different learning samples to solve most of the performance issues.

The main criteria of this research are to define optimal types and structures of modern CNNs, extract functional blocks and apply them in the process of HCNN synthesis to achieve suitable performance and accuracy results.

Genetic algorithms are part of evolutionary computing, a field of artificial intelligence. They are inspired by evolution and natural selection, where the strongest traits are passed

CNN Architecture	Features	Parameters	Error
FractalNet	- Different pathlength interaction with each other without residual connection	38.6M	CIFAR-10: 7.2 CIFAR-10+: 4.6 CIFAR-100: 28.20
DelugeNet	- Cross-layer information flow	20.2M	CIFAR-10: 3.76 CIFAR-100: 19.02
SCConvNet	- Introduced the Spatial and Channel Reconstruction convolutional block that could replace any default convolutional layer to increase overall performance	36.2M	CIFAR-10: 5.2 CIFAR-10+: 2.6 CIFAR-100: 15.11
NovelConvNet	- Application of continuous symmetry approach for 3HM architecture	28.3M	CIFAR-10: 3.88 CIFAR-100: 21.51
Contextual CNN	- Uses split transform merge idea and contextual attention-based modules	34M	ImageNet: 2.87
Dense Residual Attention-Based CNN	- Modern approach of combining the residual and dense mechanisms on top of attention-based 3HM architecture	38.2M	CIFAR-10: 4.12 CIFAR-10+: 2.31 CIFAR-100: 14.35 Multi-Crop: 2.81 ImageNet: 2.72
Competitive SE-WRN CNN	- Identity mapping used for rescaling the feature maps in pair with SE-WRN layer	36.9M	CIFAR-10: 3.5 CIFAR-100: 17.47

Fig. 1. Topology of modern CNN architectures and their relative features and parameters.

down from generation to generation. The multicriteria genetic algorithm (MCGA) is an extension of this process. It focuses on optimizing multiple objectives simultaneously. Each solution provided by the algorithm is associated with a set of objective function values. The BCGA optimizes these values and provides a set of Pareto-optimal solutions.[2]

The advantage of the MCGA is that it explores global solutions, not limited to local minima and maxima, including the ability to simultaneously process numerous parameters. When applied to neural networks, MCGA can help determine the optimal set of weights and biases in the network. They can also be useful in matters such as network design decisions, such as choosing the right number of hidden layers and nodes in each layer.

In multi-objective optimization problems, there are several conflicting objectives that need to be optimized. This results in a set of possible solutions, known as Pareto solutions, where no other solution can improve all objectives simultaneously. Therefore, the goal is not to find a single optimal solution, but to generate a set of Pareto-optimal solutions that provide a trade-off between the conflicting objectives.[5, 6]

Multi-criteria genetic algorithms (MCGAs), such as NSGA-II1, MOEA/D2, SPEA3, have shown good performance in many engineering optimization problems. Inspired by the evolutionary theory of "survival of the fittest," competitive individuals can be produced by using selection, mutation, and crossover operators through iteration. These individuals, which cannot outperform each other in all respects, create a set, the so-called non-dominance front. From the point of view of physical optimization problems, in which evaluations are always computationally complex, the population size in MOGA is usually small due to limited computing resources.

II. Topology Analysis of Modern Convolutional Neural Networks

Due to fast-forward development of convolutional neural networks the new constructive approaches are being developed actively. Instead on further increasing of structure complexity and depth development switched on utilizing functional elements such as structural blocks and layers presets. Such blocks are the paternal representation of combinations of simple layers with special connectivity approach and predefined core parameters with usage of global or local functions. These blocks by itself can provide superior improvements by applying it singly or as a combination of several different blocks.[4]

A. Spatial and Channel Reconstruction Convolution

In modern convolutional neural networks the most undergo bottleneck elements are 3x3 convolutional layers that are responsible for the most of computational load. It caused the development of different efficient convolutional operation functions such as GWC, DWC, PWC, etc. The latest substitution for convolution operation is SCConv[1] that consists of special reconstruction unit and channel reconstruction unit.

• the aim of SRU is to separate redundant features based on weighting coefficients and reconstructs them to reduce redundancy in the spatial dimension and improve feature representation;

• the aim of CRU element is to use a “divide-transform-merge” approach to lower the redundancy in channel dimensions and computational load.

• the structural block SCConv is combines SRU and CRU elements and is the replacement for default convolution operation for default CNN architectures. Utilizing the SCConv by far reduces computational load and improves model performance on complex problems.

B. Channel Boosting Based Convolutional Block

The CNN training performance results also relies on the input representation and its parameters. The lack of different qualities and lack of class-specific information within target sample may affect resulted CNN performance. To solve this problem, the concept of channel boosting CNN was designed. It based on the technology of input channel dimension and by using auxiliary learners that were introduced in CNN to boost the representation of the network.

• increasing representational capacity of CNN by means of rising quantity of input channels;

• inductive transfer learning is used in a novel way to build a boosted input representation for CNN;

• increases in computational load may happen due to the generation of auxiliary channels.

C. Multipath Based Structural Blocks

Multipath blocks are based on the approach of skipping some layers, connections and dependencies where it’s necessary to improve overall qualities of neural network. Some of these shortcut connections are the following: densely-connected block, zero padded layer, projections, dropout, 1x1 connections, etc. The most common for structural synthesis is densely connected block and has the following parameters:

• introduced depth or cross-layer dimension;

• ensures maximum data flow between the layers in the network;

• avoid relearning of redundant feature-maps;

• low and high level both features are accessible to decision layers;

III. Structural parametric synthesis of convolution neural networks

Based on number of different modern CNNs that was reviewed beforehand we can analyze and extract their unique structural blocks and apply them for structural synthesis of our own unique CNN architecture applying multi-criteria genetic algorithms. Since these blocks has their own conceptual structure and features they should be analyzed as independent structural units as well as a different pairs of such blocks placed together or separately within one neural network. Then it is necessary to apply performance testing to investigate their internal parameters, influence on overall system performance and accuracy shifts.

For our practical investigation we’ll test these blocks both solo and by pairs during HCNN synthesis using genetic algorithms. The target CNN should be very simple and straightforward. The simplicity will diminish external factors, randomness and allow us to clearly highlight the internal influence caused by each of the block. All the tests will be done using training and testing sample called “CIFAR-100”. It contains low resolution images within the number of classes that fits well to process them using relatively simple system.[7]

After applying genetic algorithms to generate basic CNN without specific blocks, the initial accuracy of such system is 86.3% and learning process lasts 5.3 hours. It will be the initial values for further performance test comparison. Take note that the overall learning time depends on the hardware and only the time differences should be taken into consideration

Input: N *(population size)*
$\overline{N}$ *(archive size)*
T *(maximum number of generations)*
Output: $\boldsymbol{A}$ *(nondominated set)*

Step 1: ***Initialization***: *Generate an initial population* $\mathbf{P}_0$ *and create the empty archive (external set)* $\overline{\mathbf{P}}_0 = \emptyset$. *Set* $t = 0$.

Step 2: ***Fitness assignment***: *Calculate fitness values of individuals in* $\mathbf{P}_t$ *and* $\overline{\mathbf{P}}_t$ *(cf. Section 3.1).*

Step 3: ***Environmental selection***: *Copy all nondominated individuals in* $\mathbf{P}_t$ *and* $\overline{\mathbf{P}}_t$ *to* $\overline{\mathbf{P}}_{t+1}$. *If size of* $\overline{\mathbf{P}}_{t+1}$ *exceeds* $\overline{N}$ *then reduce* $\overline{\mathbf{P}}_{t+1}$ *by means of the truncation operator, otherwise if size of* $\overline{\mathbf{P}}_{t+1}$ *is less than* $\overline{N}$ *then fill* $\overline{\mathbf{P}}_{t+1}$ *with dominated individuals in* $\mathbf{P}_t$ *and* $\overline{\mathbf{P}}_t$ *(cf. Section 3.2).*

Step 4: ***Termination***: *If* $t \geq T$ *or another stopping criterion is satisfied then set* $\boldsymbol{A}$ *to the set of decision vectors represented by the nondominated individuals in* $\overline{\mathbf{P}}_{t+1}$. *Stop.*

Step 5: ***Mating selection***: *Perform binary tournament selection with replacement on* $\overline{\mathbf{P}}_{t+1}$ *in order to fill the mating pool.*

Step 6: ***Variation***: *Apply recombination and mutation operators to the mating pool and set* $\mathbf{P}_{t+1}$ *to the resulting population. Increment generation counter* $(t = t + 1)$ *and go to Step 2.*

Fig. 2. Algorithmic description of SPEA-3 evolutionary algorithm.

For the structural synthesis we propose to utilize SPEA-3 evolutionary algorithm to overcome aforementioned problems shown of Figure 2.

Based on the algorithm the basic test-driven CNN architecture in generated using the set of preestablished blocks. Then the performance analysis is done on the generated result model. The main criteria for following optimization approach are target accuracy, learning rate (performance-based) and resulted CNN structure complexity.

By adding structural blocks to generation process in comparison to predefined default CNN structure the result numbers were changed as following:

• SCConv-A – accuracy 88,5%, learning process lasts 5.1h (performance boost = +3.8%);

• SCConv – accuracy 89,2%, learning process lasts 5.5h (performance boost = -3.37%%);

• DenRes-Att – accuracy 90,1%, learning process lasts 5.1h (performance boost = +3.77%%);

• SE-BN-Inception – accuracy 87,92%, learning process lasts 4.92 h (performance boost = +7.17%);

• Convolutional block attention module (take note that this block may perform better with the image samples of high resolution) – accuracy 92,1%, learning process lasts 5.42h (performance boost = -2.26%);

• PolyInception module – accuracy 84%, learning process lasts 6.2h (performance boost = -16,98%);

• Non-local Block – accuracy 87,45%, learning process lasts 5.88h (performance boost = -10.91%);

• Densely connected layer (take note that this block may perform differently based on overall system depth) – accuracy 89,45%, learning process lasts 5.88h (performance boost = -10.94%);

The combination of multiple blocks applied to following CNN structure gives us following results:

• SCConv + DenRes-Att – accuracy 94,25%, learning process lasts 6.9h (performance boost = -30.18%);

• SCConv + SE-ResNeXt – accuracy 91,8%, learning process lasts 6.6h (performance boost = -24.52%);

• CBAM + Non-local block – accuracy 92,4%, learning process lasts 6.17h (performance boost = -16.41%);

• SCConv + Non-local – accuracy 91,13%, learning process lasts 5.74h (performance boost = -8.3%);

• SCConv + PolyInception – accuracy 91%, learning process lasts 6h (performance boost = -13.2%);

• Attention merge + CBAM – accuracy 94,4%, learning process lasts 6.34h (performance boost = -19.6%);

When the problem of low learning process performance takes place, there come out a number of solutions. One of them is to populate the architectural structure of current system with the supportive blocks. The main ones are:

- batch normalization layer;

- 1x1 convolution layer;

- dropout layer;

- residual block.

IV. CONCLUSIONS

It is considered the problem of hybrid convolution neural networks (HCNN) topology analysis. It is shown that it is necessary to pay attention on some layers (blocks) possess some useful properties, which permit to increase the problem solution accuracy and decrease the complexity of HCNN. For structural parametric synthesis it’s advices to use of SPEA-3 evolutionary algorithms with usage of CNN structural blocks. As such blocks, it was proposed to consider the ones that are listed in Table 1. Also it’s considered to use supportive structural layers such as: batch size: 512, 1x1 convolution layer, dropout layer, residual block. Based on the listed blocks, test-driven CNN architectures were synthesized using multi-criteria genetic algorithms (SPEA-3) to perform the analysis of functional block performance parameters. The number of specified functional blocks were tested in terms of performance and the results are listed at Table 1. Based on the performance analysis the optimal subset of structural blocks and target CNN were defined and application criterias were described.

ТАБЛИЦЯ I. RESULT PARAMETER COMPARISON TABLE OF SINGLE-USED BLOCKS

Block type	***Accuracy(%)***	***Time (H)***	***Diff (~)***
Densely connected layer	***0.8945***	***5.88***	***3.2***
SCConv block	***0.885***	***2.2***	***2.2***
SCConc-A block	***0.892***	***2.6***	***2.6***
SE-BN-Inception module	***0.8792***	***4.92***	***1.8***
Convolutional block attention module	***0.921***	***5.42***	***5.9***
DenRes-Att module	***0.8781***	***8.81***	***1.7***
PolyInception module	***0.89***	***8.9***	***2.7***
Non-local Block	***89.45***	***7.45***	***3.1***

REFERENCES

[1] Jiafeng Li, Ying Wen, “SCConv: Spatial and Channel Reconstruction Convolution for Feature Redundancy“ 2022.

[2] Shouyong Jiang, Shengxiang Yang, “A Strength Pareto Evolutionary Algorithm Based on Reference Direction for Multiobjective and Many-Objective Optimization” in *IEEE Transactions on Evolutionary Computation*, 2017.

[3] Iveta Mrazova, Marek Kukacka. "Hybrid convolutional neural networks" IEEE International Conference on Industrial Informatics 10.1109/INDIN.2008.4618146..

[4] Chaitanya Nagpal and Shiv Ram Dubey. “A Performance Evaluation of Convolutional Neural Networks for Face Anti Spoofing”.

[5] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” in Proceedings of the 22nd ACM international conference on Multimedia. ACM, 2014, pp. 675–678.

[6] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.

C. Cao, X. Liu, Y. Yang, Y. Yu, J. Wang, Z. Wang, Y. Huang, L. Wang, C. Huang, W. Xu, D. Ramanan, and T. S. Huang, “Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks,” in ICCV, 2015.