Neurocomputing 000 (2017) 1–10

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Hyperlayer Bilinear Pooling with application to fine-grained categorization and image retrieval

Qiule Sun[a,b], Qilong Wang[b], Jianxin Zhang[a,*], Peihua Li[b,*]

[a] Key Lab of Advanced Design and Intelligent Computing (Ministry of Education), Dalian University, Dalian, China
[b] School of Information and Communication Engineering, Dalian University of Technology, Dalian, China

ARTICLE INFO

Article history:
Received 4 July 2017
Revised 6 December 2017
Accepted 8 December 2017
Available online xxx

Communicated by Prof. X. Gao

Keywords:
Hyperlayer Bilinear Pooling (HLBP)
Convolutional neural networks
Fine-grained categorization
Image retrieval

ABSTRACT

With rapid development of deep convolutional neural networks (CNNs), more and more high–accuracy CNN architectures have been proposed for multimedia and vision applications. Among them, Bilinear CNN (B-CNN) performs outer product on the outputs of convolutional layers of two stream CNN models, and has attracted a lot of attentions due to its effectiveness in fine-grained categorization. However, the B-CNN fails to use the information inherent in different convolutional layers, meanwhile the cost of computing and storage is higher as evaluation requirements of two CNN models. In this paper, we propose a family of Hyperlayer Bilinear Pooling (HLBP) methods to overcome the limitations of B-CNN. Evaluating a single CNN model only once, our HLBP methods can effectively exploit the second-order statistics of features of multiple layers. Besides fine-grained categorization, we also apply the proposed HLBP methods to content-based image retrieval task. Extensive experiments on seven widely used benchmarks demonstrate that our HLBP methods are superior to the counterparts (e.g., B-CNN), achieving very competitive or better performance compared to state-of-the-arts on both fine-grained categorization and content-based image retrieval tasks.

1. Introduction

Deep convolutional neural networks (CNNs) have been rapidly spreading and achieving impressive performance in a broad range of multimedia and vision applications, such as content based image retrieval [1–5], image classification [6–8], action recognition in video [9] and pose estimation [10]. The classical CNN architectures are generally designed by stacking convolutions followed by non-linear activation functions, pooling operations and optional fully-connected layers for final representations fed to task-driven losses. Based on CNN models pre-trained on ImageNet dataset [11], through further coding and/or pooling methods, the outputs of convolutional or fully-connected layers generalize well to a diversity of benchmarks, often resulting in much better performance than the original CNN models in many tasks [12–15].

Recent researches have shown that integration of coding and/or pooling methods as structural layers into CNN architectures, trainable end-to-end, can obtain significant improvement [16–19]. Arandjelovic et al. [16] propose a novel neural network called NetVLAD, which plugs classical VLAD coding [20] as a trainable layer into CNN architectures and achieves promising results in place recognition task. Tang et al. [17] incorporate Fisher vector encoding [21] into the CNN models for end-to-end learning, which improves accuracies of deep CNNs in object recognition. Ionescu et al. [18] develop the theory of matrix back propagation and instantiate a DeepO_2P layer where O_2P representation [22] is embedded into CNN models for effective region classification. Among them, bilinear CNN (B-CNN) [19] has attracted a lot of attentions due to its effectiveness in fine-grained categorization. In B-CNN, the outer products of the outputs of convolutional layers of two CNN models are pooled to obtain image representations. In spite of its great success, B-CNN fails to use the information inherent in different convolutional layers of a single CNN model. Meanwhile, it has high costs of computing and storage than single CNN model based methods when two different CNN models are employed.

As different convolutional layers capture various levels (low, middle and high) of features as well as varying scale information, many works have demonstrated that it is beneficial to use the information across convolutional layers [8,23–25]. Liu et al. [23] propose a cross-convolutional-layer pooling (CrossLayer) method to perform weighted spatial pooling in two consecutive convolutional layers, in which the outputs of the former layer serve as local features while the latter one is used for weighting. Hariharan et al. [24] introduce the representation of pixel-wise hypercolumn,

* Corresponding authors: Jianxin Zhang and Peihua Li.
E-mail addresses: qiulesun@163.com (Q. Sun), qlwang@mail.dlut.edu.cn (Q. Wang), jxzhang0411@163.com (J. Zhang), peihuali@dlut.edu.cn (P. Li).

https://doi.org/10.1016/j.neucom.2017.12.020
0925-2312/

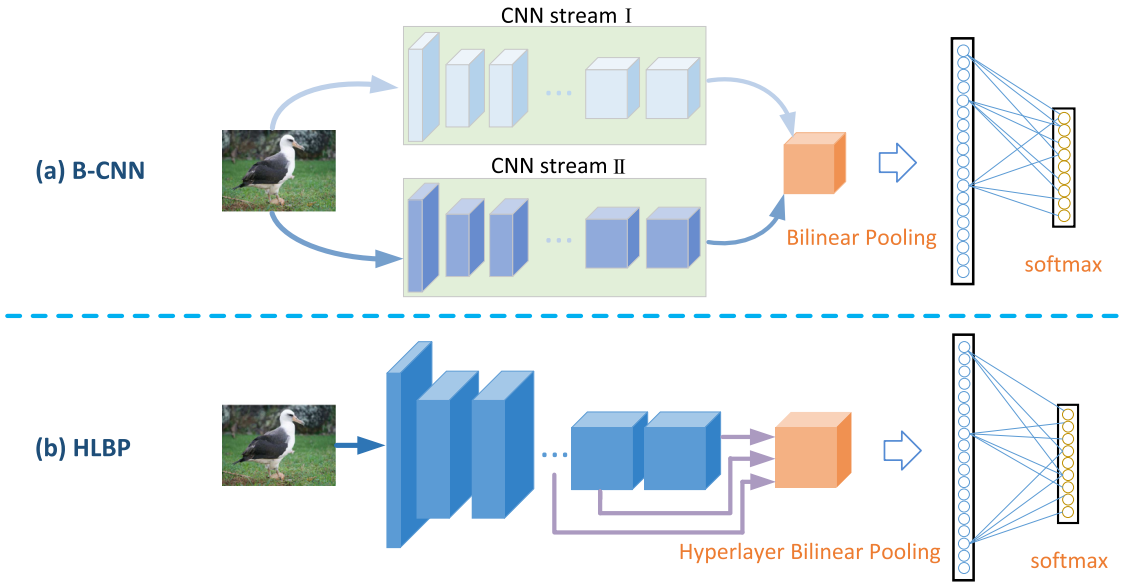

Fig. 1. Comparison of (a) B-CNN with (b) the proposed Hyperlayer Bilinear Pooling (HLBP). B-CNN computes outer product of the outputs of convolutional layers from two CNN models, while our HLBP methods employ the outer product of the outputs of individual convolutional layers and cross-convolutional layers in a single CNN model. We present three methods (detailed in Section 3) for mining information of multiple layers, namely, feature-level addition (FA) scheme, representation-level addition (RA) scheme, and the scheme of feature-level stacking followed by dimensionality reduction (FSDR). Our HLBP methods mine the second-order statistics, i.e., autocorrelation of intra-layer features and cross-correlation of inter-layer features, which is different from the existing cross-layer CNN methods [8,23–25] works mining only the first-order statistics.

which is defined as the vector of responses of all CNN units above one pixel. Significant performance gains are reported therein by using the hypercolumn representation in object segmentation and fine-grained localization tasks. In addition, cross-layer connection has played a key role in some state-of-the-art CNN architectures such as ResNet [8] and DenseNet [25]. In the forward propagation of ResNet, the outputs of two different convolutional layers are combined with element-wise addition followed by a non-linear function, which is then connected to the next convolutional layer. In DenseNet, for each dense block the inputs of the convolutional layers are stacking the outputs of all its preceding layers as input.

In order to circumvent the limitations of B-CNN while inheriting its merits, this paper proposes a family of Hyperlayer Bilinear Pooling (HLBP) methods to effectively mine the information inherent in different convolutional layers. The differences between B-CNN and our HLBP are illustrated in Fig. 1. Firstly, our HLBP exploits different convolutional layers in a single CNN model instead of two different CNN models. It thus has lower cost of computing and storage comparing to evaluate two CNN models simultaneously. Motivated by ResNet [8] and DenseNet [25], we present three kinds of fusion schemes for our HLBP to exploit multi-layer statistical information under the B-CNN framework, including feature-level addition (FA), representation-level addition (RA) and feature-level stacking followed by dimensionality reduction (FSDR). Meanwhile, we apply the proposed HLBP to fine-grained categorization and image retrieval tasks, and experimental results demonstrate that our HLBP is superior to its counterparts (e.g., B-CNN) and can perform very competitively or better compared to state-of-the-art methods. Last but not least, while exploiting features of multiple layers, our method mines the second-order statistics (i.e., autocorrelation and cross-correlation), which is different from the existing cross-layer CNN methods [8,23–25] where only the first-order statistics are mined by max or average pooling of features.

The main contributions of this paper can be summarized in three folds: (1) We, to our best knowledge, make the first attempt to study systematically the multi-layer second-order pooling in the CNN architectures. (2) To this end, we proposes a family of Hyperlayer Bilinear Pooling (HLBP) methods using various fusion schemes, which are significantly distinguished from the existing, first-order cross-layer CNN methods [23,24]. (3) We evaluate the proposed HLBP with various CNN architectures and apply them to fine-grained categorization and content-based image retrieval tasks. Extensive experiments on two fine-grained and four image retrieval benchmarks demonstrate that our HLBP greatly improves its counterparts (e.g., B-CNN) and achieves very competitive or better performance compared to state-of-the-arts.

2. B-CNN and cross-layer B-CNN

This section briefly describes the B-CNN method as well as its variant, cross-layer B-CNN, which are both very related to our HLBP.

2.1. B-CNN

The B-CNN model [19] is concerned with cross-correlation information of local features extracted from two CNN models. Let $\mathbf{X} \in R^{N\times d}$ and $\mathbf{Y} \in R^{N\times d}$ be the feature matrices of the last convolutional layers of the two CNN models, where N and d are the number and dimension of features, respectively. As shown in Fig. 2 (a), B-CNN model computes the outer product of $\mathbf{X}$ and $\mathbf{Y}$, i.e.,

$$\mathbf{Z} = \mathbf{X}^T\mathbf{Y}, \tag{1}$$

which is then subject to power normalization, i.e., element-wise signed square root operation, followed by ℓ_2 normalization. The final bilinear pooling representation is connected with the softmax loss function. Given $\frac{\partial L}{\partial \mathbf{z}}$ the gradient of the loss function L with respect to representation $\mathbf{Z}$, the backward propagation of bilinear pooling layer can be efficiently achieved by the following chain rule of gradients,

$$\frac{\partial L}{\partial \mathbf{X}} = \mathbf{Y}\left(\frac{\partial L}{\partial \mathbf{Z}}\right)^T, \quad \frac{\partial L}{\partial \mathbf{Y}} = \mathbf{X}\left(\frac{\partial L}{\partial \mathbf{Z}}\right). \tag{2}$$

Please cite this article as: Q. Sun et al., Hyperlayer Bilinear Pooling with application to fine-grained categorization and image retrieval, Neurocomputing (2017), https://doi.org/10.1016/j.neucom.2017.12.020