Neurocomputing 175 (2016) 427–432

Contents lists available at ScienceDirect

Neurocomputing

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

Brief Papers

Local visual feature fusion via maximum margin multimodal deep neural network

Zhiquan Ren, Yue Deng, Qionghai Dai *

Tsinghua University, China

ARTICLE INFO

Article history:
Received 18 June 2015
Received in revised form
21 August 2015
Accepted 11 October 2015
Communicated by XIANG Xiang Bai
Available online 31 October 2015

Keywords:
Image categorization
Deep learning
Feature fusion
Discriminative learning

ABSTRACT

In this letter, we consider improving the image categorization performance by exploiting multiple local descriptors on the image. To achieve this goal, a novel deep learning configuration called maximum margin multimodal deep neural network (3mDNN) is proposed to learn joint feature from different data views. The local feature representations encoded by 3mDNN exhibit two significant advantages: (1) involving the information of multiple descriptors and (2) exhibiting discriminative ability. The whole deep architecture is well solved by the typical back propagation (BP) method and its performances are verified on three benchmark image datasets.

1. Introduction

Learning effective representations of an image [1] is a long-standing pursuit in both the fields of computer vision and machine learning. Among conventional image coding approaches, bag-of-feature (BoF) method is the most prevalent one due to its flexibility in treating various image sizes. In details, these methods follow diverse mathematical concepts, e.g. sparse coding [2,3], to code local image patches as some statistical representations. Then, these local features are combined together via spatial pyramid pooling [4] forming the image-level histogram for intelligent categorization. While the successes of these typical methods have been widely witnessed, there are still some approaches to further improve their performances.

First, typical BoF always exploits single descriptor, e.g. the widely used SIFT, to summarize local image contents. In fact, when describing an image patch, multiple visual descriptors can be generated to quantify different statistical properties, e.g. texture, color and gradient. The single visual descriptor may be partial in dissecting the full information of one patch. In order to allow multiple visual descriptors, one promising approach is to fuse different local descriptors as a single one using subspace [5] or probabilistic models [6]. Alternatively, it is also possible to generate the image-level histograms from different descriptors. Then, the inherent differences of these descriptor-specified histograms are evaluated with multiple kernel learning [7]. In [8], an inspiring sparse coding framework is established to jointly reconstruct the information and penalize their correlations from multiple views. The prevalent structured sparse learning concept has also been incorporated into multiview framework for simultaneous data clustering and group-level feature selection [9].

Besides, conventional bag-of-feature model is a fully generative model that sheds no light on the discriminative side. It widely known that learning discriminative features will nontrivially improves the classification performances. Among them, discriminative dictionary learning is a prevalent direction which can be implemented in various ways including sparse learning [10,11], codeword selection [12] and probabilistic inference [13]. It is also flexible to incorporate the discriminative information into the histogram generation parts by exploiting discriminative pooling [14].

While diverse concepts have been exploited to address the issues of multiple descriptors and discriminative learning, there is a lack of a unified paradigm. In a nutshell, existing works always consider the aforementioned two tasks separately with different mathematical formulations. Moreover, rather than using the existing shallow representations, many pioneering works [15–17] have shown the great promises of hierarchical deep representation. Accordingly, it is natural to ask whether there is a joint mathematical formulation that “deeply” addresses the two desired properties altogether?

In this letter, we will propose maximum margin multimodal deep neural network (3mDNN), a deep learning model to discriminatively fuse multiple visual descriptors for robust local information coding. The schematic configuration of 3mDNN has been summarized in Fig. 1. The bulk of the system is a multimodal

* Corresponding author.

http://dx.doi.org/10.1016/j.neucom.2015.10.076
0925-2312/

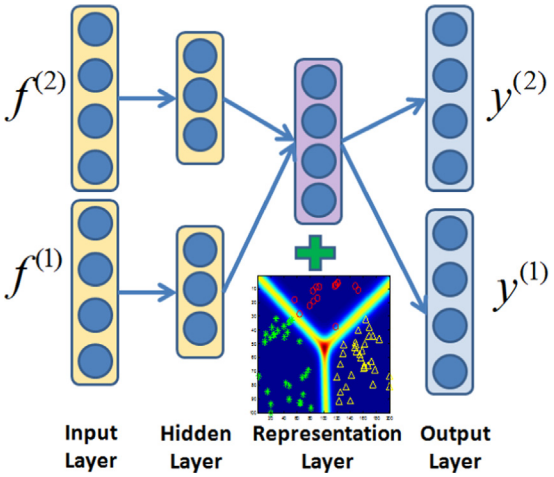

Fig. 1. A conceptual explanation of the maximum margin multimodal deep neural network (3MDNN).

deep neural network (DNN) with multiple descriptors as inputs. By passing the neural network, multiple independent descriptors are coded altogether forming a high-level feature representation [18] on the representation layer. To improve the discriminative property of the representation, a maximum margin regularization is placed on the nodes of the representation layer. The performances of 3mDNN are tested on three benchmark datasets including fifteen-scene [19], Caltech101 [20], VOC 2007 [21] and MIT indoor scene [22] with comparisons to other leading image coding and categorization methods.

2. Maximum margin multimodal deep neural network

2.1. Model

In Fig. 1, we define input vector $f^{(i)}$ as a visual descriptor extracted from the ith view. The whole multimodal deep neural network plays the role of recovering the same input information of $f^{(i)}$ at the output $y^{(i)}$. Accordingly, we define the objective of this deep auto-encoder as

$$L=\sum_{ij}\|y_j^{(i)}-h_\Theta(f_j^{(i)})\|_2^2+\eta\sum_l\|w^l\|_2^2 \tag{1}$$

where parameter set $\Theta=(w^l,b^l),\forall l$ record the weights and bias between the $(l-1)$th and the lth layer; $h_\Theta(\cdot)$ denotes the feed-forward transformation from f to y through the DNN. The last quadratic term in (1) places a regularization on the weights to avoid over-fitting.

It is worthwhile to note that the DNN in Fig. 1 has four layers, i.e. input layer, hidden layer, feature representation layer and output layer. Specifically, for the ith node on the lth layer, the activation $a^{li}=s(\sum_j w^{lj}a^{(l-1)j}+b^l)$, where $s(\cdot)$ is the sigmoid function. In the neural network, the hidden layer preliminarily removes the noises in the input layer and further passes the coded information to the representation layer. The representation layer fuses the information from hidden layer (from different views) and generates the second level activations. Finally, the output layer is placed after the representation layer to reconstruct the same input information. In training phase, the output layer is added to make the whole DNN perform like an auto-encoder. In the testing procedures of feature coding, the output layer is not explicitly used.

The representation layer here just addresses the generative property of the training data. In image classification task, a number of previous works have suggested the importance of discriminative structure learning [23,14]. In this work, we incorporate the label information into the representation layer according to the famous maximum margin strategy. In details, when defining a_i^r as the feature representation of the ith sample, the maximum margin pursuit is placed between them, i.e.

$$C(a_i^{(r)})=\begin{cases}\sum_j\|a_i^r-a_j^r\|_2^2, & \text{if } s_i=s_j\\ \sum_j\max(0,d-\|a_i^r-a_j^r\|_2^2), & \text{if } s_i\neq s_j\end{cases} \tag{2}$$

In the above formulation, s_i denotes the label information of the ith sample. The meaning of this constraint is obvious. It encourages small distances for samples in the same class. However, for samples in different classes, when their inherent feature distance is less than a margin d, they are penalized to be away from each other as far as possible. In this study, we simply set $d=1$ following the suggestions in existing work [24]. Accordingly, the whole objective function of maximum margin multimodal deep neural network (3mDNN) is obtained:

$$\min\cdot F=L+\lambda\sum_i C(a_i^r) \tag{3}$$

2.2. Coding locally and pooling globally

We clarify the differences between 3mDNN and other DNNs for image categorization. Existing DNNs always generate the feature from the whole image and require the same size for input images. To fulfill this restriction, down-sampling implementations are usually adopted to normalize different images into the same resolution. Then, with the unique size input, a complicated convolution layer [17] is then placed after the input data with different convolutional kernels. The multiple convolutional kernels inevitably bring in a large number of unknown weighting parameters to be inferred. Unlike these typical DNNs, the 3mDNN is established on the local patch level which relaxes the unique image size constraint and reduces the number of latent parameters.

In implementation, different types of local descriptors are generated from image patches. After feeding these patch-level local descriptors through the 3mDNN, the activations on the representation level r are recoded as the joint representation in $\mathbb{R}^c$. c is the number of nodes on the rth layer. Loosely speaking, the representation layer resembles the typical codebook with c centers/bases. Then, all the local representations on the image are pooled altogether forming the final image-level histogram as conventional BoF implementations. In this letter, we follow the same implementations in [2,4] to use the spatial pyramid approach to compose local representations as the global histogram.

2.3. Optimization

We follow the benchmark protocol to train the DNN by two sequential steps of layer-wise initialization and fine tuning. In the layer initialization step, all the parameters between layers are initialized by an layer-wise auto-encoder. It is emphasized here that the layer-wise auto-encoder is not the same as the global auto-encoder discussed in Eq. (1). It is only used to initialize the parameters in one specific layer by addressing the optimal reconstruction on two sides of the layer (not the global neural network). The initializations with layer-wise auto-encoder have been extensively discussed in previous works and we refer interested readers to [17,25] for details.

Given each layer well initialized, the fine tuning step further adjusts the parameters by minimizing the objective function in Eq. (3). The whole optimization procedures are subject to the well-known