IMPROVING DEEP CONVOLUTIONAL NEURAL NETWORKS WITH UNSUPERVISED FEATURE LEARNING

Kien Nguyen, Clinton Fookes, Sridha Sridharan

Image and Video lab, SAIVT Research Program
Queensland University of Technology, Brisbane, Australia

ABSTRACT

The latest generation of Deep Convolutional Neural Networks (DCNN) have dramatically advanced challenging computer vision tasks, especially in object detection and object classification, achieving state-of-the-art performance in several computer vision tasks including text recognition, sign recognition, face recognition and scene understanding. The depth of these supervised networks has enabled learning deeper and hierarchical representation of features. In parallel, unsupervised deep learning such as Convolutional Deep Belief Network (CDBN) has also achieved state-of-the-art in many computer vision tasks. However, there is very limited research on jointly exploiting the strength of these two approaches. In this paper, we investigate the learning capability of both methods. We compare the output of individual layers and show that many learnt filters and outputs of the corresponding level layer are almost similar for both approaches. Stacking the DCNN on top of unsupervised layers or replacing layers in the DCNN with the corresponding learnt layers in the CDBN can improve the recognition/classification accuracy and training computational expense. We demonstrate the validity of the proposal on ImageNet dataset.

Index Terms— Deep learning, Convolutional Neural Network, Deep Convolutional Belief Network, Unsupervised deep learning, Supervised deep learning

1. INTRODUCTION

Feature extraction plays a key role in the performance of recognition systems since this phase decides the discrimination capability of the systems. In traditional recognition approaches, features are normally carefully hand-crafted to maximise the discrimination capability. A multitude of hand-designed features have been explored (such as SIFT [1] and HOG [2]) and achieved great success in computer vision tasks. However, because these features are hand-designed, not learnt from the nature of data, they are subjective to perception of the designers and are not always the optimal feature set required for a given task. In addition, classifiers (such as the SVM and k-nearest neighbor) are generic and not robust to the diverse variation of the data. These traditional approaches represent a shallow architecture, which is severely challenged by the non-linear complexity of the features. Recently, deep learning researchers have proposed to learn feature representations in a hierarchy all the way from pixels to classifiers through multiple layers (deep) architecture [3, 4]. The deep architecture allows the system to learn to represent features by themselves based on the nature of the data, rather than the subjective nature of human perception. This deep architecture has been shown to achieve state-of-the-art in many computer vision tasks with little effort in tuning the model including text recognition [5], object detection [6], object recognition [7], face recognition [8], scene parsing/labeling [9].

There have been 2 major trends in deep learning approaches: supervised and unsupervised. While in supervised learning (e.g. Convolutional Neural Network [5, 10] and Recurrent Neural Network [11]), training data includes both the input and the desired output, in unsupervised learning (e.g. Deep Belief Network [12] and Deep (Sparse/Denoising) AutoEncoder [13]), the model is not provided with the desired output. While both supervised and unsupervised deep learning have achieved superior performance in several computer vision tasks, there has been little effort to jointly exploit the advantages of these two. Most recently, Erhan *et al.* have shown that unsupervised pretraining, which precedes the supervised deep learning, helps to guide the deep learning towards basins of attraction of minima that support better generalisation from the training set [14]. This is a premiliary first step towards exploiting the advantages of both approaches. In this work, we approach the integration of unsupervised and supervised deep learning from another perspective. We show that early layers' parameters in the supervised network (DCNN) can be learnt from an unsupervised network (CDBN), which not only reduces the number of parameters to learn and lessens the training burden, but also improves classification accuracy in some cases.

The remaining of this paper is organised as follows: Section 2 presents motivations for this research, Section 3 introduces background on Deep learning, Section 4 describes our proposal on how to employ unsupervised learning to improve the DCNN, Section 5 discusses and concludes our paper.

2. MOTIVATION

Our contribution in this paper - to learn early the DCNN layers parameters from an unsupervised CDBN network - is driven by the following motivations:

- Motivation 1: Training deep architecture neural networks is very expensive, for example, it took two weeks for Krizhevsky *et al.* to train their deep learning model on 2 Graphical Processing Units (GPUs) on 1.2 million training images provided by the ImageNet Large Scale Visual Recognition Challenge 2012 (ILSVRC) [10]. However, there are many pretrained models publicly available for the research community. The question is how to take advantage of this rich information to speed up and save computing resource without compromising the learning capability?

- Motivation 2: We observe that even though there are a plethora of deep architectures, the key to all these architectures is the hierarchy of the feature representation, which means each layer in the deep learning multiple layer model learns to represent images from pixels, to edges, to textons, to motifs, to parts and to objects in that order [15]. Notably, both supervised (DCNN) [16] and unsupervised (DCBN) [12] deep learning approaches learn the hierarchy in the same hierarchical manner. It is obvious that the feature representation will differ at higher layers of both approaches; however, the similar representation of low-level and mid-level layers raises a question of how to incorporate the learnt information from both approaches?

- Motivation 3: Transfer learning has been successfully used to transfer knowledge learnt from one domain to another. Recently, Oquab *et al.* discussed how image representation learnt with CNN can be efficiently transferred to other visual recognition tasks [17]. In other words, the parameters learnt from one dataset can be transferred to other datasets. Now rather than transfer learnt parameters from one dataset to another dataset, an interesting question is whether we can transfer learnt parameters from one learning category (unsupervised) to another (supervised)?

- Motivation 4: The benefit of unsupervised learning is the unlabeled training data, which allows it to train on huge quantity of data. Training on big data enables unsupervised learning learn parameters accurately. If the parameters learnt from unsupervised learning (CDBN) can be transferred to supervised learning (DCNN), not only the burden of training on DCNNs is less computational expensive, but the accuracy may also be improved due to more accurate training on big data of the unsupervised learning.

3. DEEP LEARNING

This section introduces a typical supervised and a typical unsupervised deep network, which are used in this research.

3.1. Deep Convolutional Neural Network (DCNN)

Inspired by the visual cortex of the human and animal brain, convolutional neural networks (CNN) were first introduced in 1980 by Kunihiko Fukushima [18]. The first major influential model of CNN is the deep architecture called LeNet-5 introduced by Lecun *et al.* [5]. Lenet-5 consists of a series of layers, including an input layer, followed by a number of feature extracting Convolutional and Subsampling layers, and finally a number of fully connected layers that perform the classification as shown in Figure 1. The famous network can classify digits successfully, which is applied to recognize checking numbers. However, without efficient computing resources and methods to prevent from overfitting if the numbers of layers becomes larger and larger, it did not perform well with more complex problems. The second major influential model of CNN is KrizhevskyNet [10]. KrizhevskyNet is an extension of LeNet-5 fulfilled by the power of latest computing hardware. KrizhevskyNet is deeper with 7 layers as shown in Figure 2, larger with 60 million parameters. This was made possible by the fast hardware (GPU-optimised code), big dataset (1.2 million training images) and better regularisation (dropout).

3.2. Convolutional Deep Belief Network (CDBN)

Deep belief network (DBN) is a generative graphical model consisting of a layer of visible units and multiple layers of hidden units [19]. Each layer, which is a Restricted Boltzmanzz Machine (RBM), encodes correlations in the units in the layer below to learn higher-level feature representations from unlabeled data, suitable for use in tasks such as classification. A breakthrough in training DBN was proposed by Hinton *et al.* with layer-wise training, which greedily trains each layer (from bottom to top) [19]. Recently, Lee *et al.* proposed to employ Convolutional RBM (CRBM) as a replacement for RBM with the advantage that the weights between layers are shared among all locations in an image [12]. A CDBN is constructed simply by stacking multiple CRBMs following by max-pooling layers. The authors reported the CDBN is able to learn the hierarchical representation of natural images. The first layer has been shown to learn oriented, localised edge filters. The second layer learn contours, corners, angles and surface boundaries in the image. The third layer has been shown to learn object parts, even though the algorithm was not given any label specifying the locations of either objects or their parts. Higher layers in the CDBN learn features which are not only higher level, but also more specific to particular object categories.

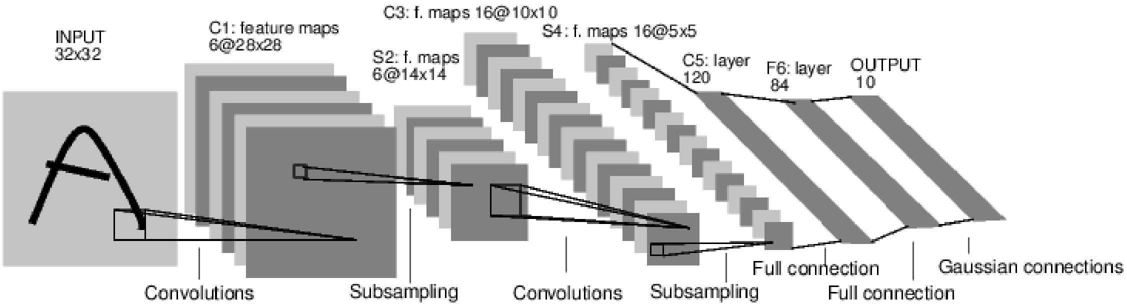

Fig. 1. The first major influential model of CNN: LeNet-5 [5]

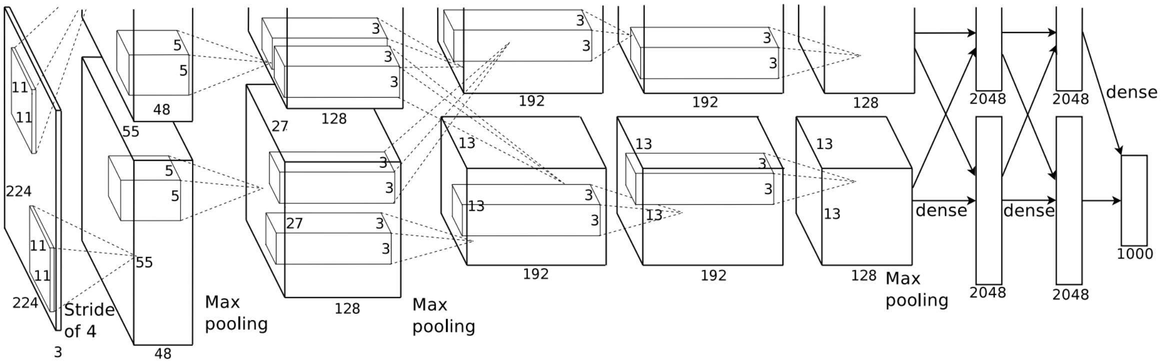

Fig. 2. The second major influential model of CNN: KrizhevskyNet [10]

4. IMPROVING DCNN BY UNSUPERVISED FEATURE LEARNING

4.1. Network architecture

4.1.1. DCNN

In this work, we employ the network architecture presented by Krizshevesky *et al.* [10] as shown in Figure 2 on a public framework called CAFFE [20]. Caffe is a clean and modifiable C++ framework with state-of-the-art deep learning algorithms for training and deploying general-purpose convolutional neural networks and other deep models efficiently on commodity architectures. By separating model representation from actual implementation, Caffe allows experimentation and seamless switching among platforms for ease of development and deployment from prototyping machines to cloud environments. The benefits of using Caffe is the richness of reference models from the community.

The network, which we name as KrizhevskyNet, is composed of eight layers: five successive convolutional layers, C1,...,C5, and three fully connected layers, FC6,...,FC8. Response-normalisation layers follow the first and second convolutional layers. Max-pooling layers follow both response-normalisation layers as well as the fifth convolutional layer. Readers should refer to [10] for details of convolutional, response-normalisation and max-pooling layers.

4.1.2. CDBN

For CDBN, we implement our own version of Honglak Lee proposal based on the paper [12] as described in Section 3.2. Three layers, are trained in the Kyoto dataset to learn the parameters of the CDBN model. To be compatible with the early layers of the DCNN, the first two layers of the CDBN architecture are followed by probabilistic max-pooling layers. The third layer is not followed by any max-pooling operation. Lee *et al.* have shown that the CDBN can learn hierarchical representations of the natural images [12].

4.2. Learning early layers in DCNN from an unsupervised network CDBN

The first layer of the DCNN [10] has 96 filters of size 11×11 (actual size is 11×11×3, but here we will not focus on the color image, so we reduce the filter to gray filter). The DCNN model is trained on the ImageNet dataset using for ImageNet Large Scale Visual Challenge [7]. The learnt filters of the first layer of the DCNN is illustrated in Figure 3. We also train one unsupervised learning layer of the CDBN to learn the same number of filters with the same size. The first layer of the CDBN is configured with 96 filters of size 11×11. We applied layer-wise training approach to train the first layer of the CDBN on Kyoto dataset as described in [12]. The learnt

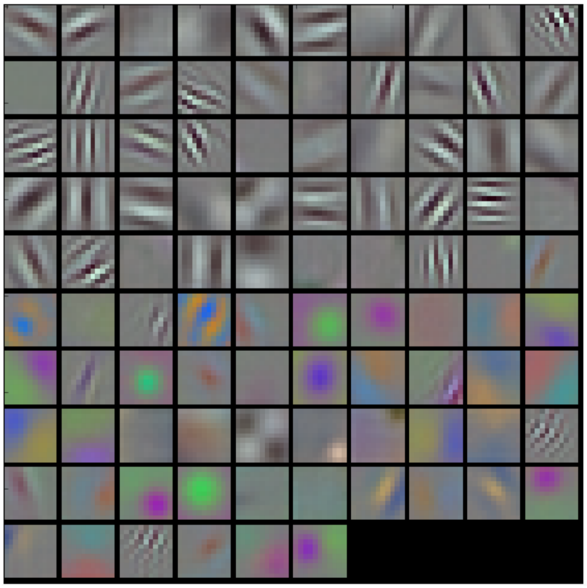

Fig. 3. The filters learnt supervisedly in the first layer of the DCNN [10]

96 filters of the CDBN are shown in Figure 4. Observe that even though two models are trained with different datasets, interestingly, the 96 filters learnt look a bit similar in a way as edge detectors and corner detectors.

From the above observation, our proposal is to replace the early layers' filters of the DCNN with the corresponding filters learnt from the unsupervised CDBN approach. While DCNN training is done with backpropagation on the whole parameters set, which is computationally expensive with 60 million parameters for KrizshevskyNet, learning these filters from unsupervised approach not only reduces the computational expense, but also achieve competitive classification performance as illustrated in Table 1. By fixing the parameters of the early layer, the number of parameters of the DCNN to learn reduces (96×11×11×3). Consequently, the training time reduces by 30% from 2 weeks as in [10] to 10 days. Investigating the learning capability of the DCNN model, we observe a slight increase in the classification accuracy from 18.2% top-5 error rate down to 17.7%. Similarly, replacing the first and second layers' filters with the filters learnt in the second layer of unsupervised CDBN results in less training burden (7 days) but the classification accuracy does not improve (18.1%). Substituting the first, second and third layers' filters with those learnt from the unsupervised CDBN leads to further decrease in training time (5 days) but a raise in the error rate (18.9%). Thus transferring the learnt parameters from unsupervised learning to supervised learning can improve performance in terms of accuracy and training computational expense with a single layer replacement. Further reduction in training time can be achieved through the replacement of higher layers with a small reduction in accuracy.

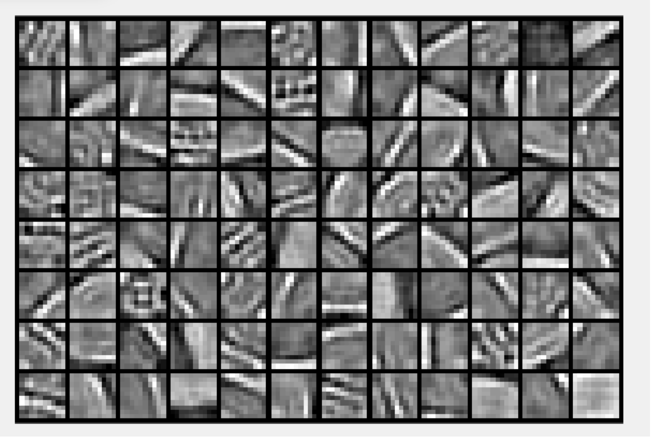

Fig. 4. The filters learnt unsupervisedly in the first layer of the CDBN [12]

Table 1. Effect of learning filters from unsupervised CDBN to training time and accuracy of DCNN.

Approaches	**Training time**	**Accuracy (EER)**
Original DCNN	14 days	18.2%
One layer replaced	11 days	17.7%
Two layers replaced	8 days	18.1%
Three layers replaced	5 days	18.9%

This result aligns with and adds to recent findings by Boureau *et al.* in [21], which showed that even though different categories of classes lead to different statistics of the data, the low-level and mid-level features learnt in one dataset can be useful for another regarding the similarity of low- and mid-level image representations. We have further shown that the similarity in low- and mid-level image representations can be exploited to cross learn between unsupervised and supervised networks. Thus by training the unsupervised layers one time, these layers parameters can be employed as early layers in supervised networks to reduce the training time without compromising the recognition performance.

5. CONCLUSION

Both our approach and others employ unsupervised learning to improve the performance of the supervised learning. However, while others utilise learnt parameters as initial values to iteratively train the model, our approach encodes the parameters in various layers, resulting in competitive recognition/classification performance while reducing the training burden. Even though the experiments have been shown with the DCNN and the CDBN, we believe that our proposed approach should also work with other supervised and unsupervised networks. Our next step is to investigate the application to other network architectures to gain further insight into the proposed approach.

6. REFERENCES

[1] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[2] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, June 2005, vol. 1, pp. 886–893 vol. 1.

[3] Y. Bengio, “Learning deep architectures for AI,” *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[4] J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural Networks*, vol. 61, no. 0, pp. 85 – 117, 2015.

[5] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov 1998.

[6] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. Lecun, “Pedestrian detection with unsupervised multi-stage feature learning,” in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, Washington, DC, USA, 2013, CVPR ’13, pp. 3626–3633, IEEE Computer Society.

[7] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, “ImageNet Large Scale Visual Recognition Challenge,” 2014.

[8] Y. Taigman, Ming Y., M. Ranzato, and L. Wolf, “Deepface: Closing the gap to human-level performance in face verification,” in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, June 2014, pp. 1701–1708.

[9] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, “Learning hierarchical features for scene labeling,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 8, pp. 1915–1929, Aug 2013.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Curran Associates, Inc., 2012.

[11] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J. Schmidhuber, “A novel connectionist system for unconstrained handwriting recognition,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 5, pp. 855–868, May 2009.

[12] H. Lee, R. Grosse, R. Ranganath, and A. Ng, “Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations,” in *Proceedings of the 26th Annual International Conference on Machine Learning*, New York, NY, USA, 2009, ICML ’09, pp. 609–616, ACM.

[13] P. Vincent, H. Larochelle, Y. Bengio, and P. Manzagol, “Extracting and composing robust features with denoising autoencoders,” in *Proceedings of the 25th International Conference on Machine Learning*, New York, NY, USA, 2008, ICML ’08, pp. 1096–1103, ACM.

[14] D. Erhan, Y. Bengio, A. Courville, P. Manzagol, P. Vincent, and S. Bengio, “Why does unsupervised pre-training help deep learning?,” *J. Mach. Learn. Res.*, vol. 11, pp. 625–660, Mar. 2010.

[15] M. Ranzato, “Deep learning for large scale visual recognition,” Tutorial Large-Scale Visual Recognition in International Conference on Computer Vision and Pattern Recognition, 2014.

[16] M. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in *Computer Vision ECCV 2014*, vol. 8689 of *Lecture Notes in Computer Science*, pp. 818–833. Springer International Publishing, 2014.

[17] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, “Learning and transferring mid-level image representations using convolutional neural networks,” Los Alamitos, CA, USA, 2014, pp. 1717 – 24.

[18] K. Fukushima, “Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position,” *Biological Cybernetics*, vol. 36, no. 4, pp. 193–202, 1980.

[19] G. Hinton and O. Simon, “A fast learning algorithm for deep belief nets,” *Neural Computation*, vol. 18, pp. 2006, 2006.

[20] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” *arXiv preprint arXiv:1408.5093*, 2014.

[21] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, “Learning mid-level features for recognition,” in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, June 2010, pp. 2559–2566.