Regularized Deep Convolutional Neural Networks for Feature Extraction and Classification

Khaoula Jayech(✉)

LATIS Research Lab, National Engineering School of Sousse,
University of Sousse, Sousse, Tunisia
ljayech_k@yahoo.fr

Abstract. Deep Convolutional Neural Networks (DCNNs) are the state-of-the-art in fields such as visual object recognition, handwriting and speech recognition. The DCNNs include a large number of layers, a huge number of units, and connections. Therefore, with the huge number of parameters, overfitting can occur. In order to prevent the network against this problem, regularization techniques have been applied in different positions. In this paper, we show that with the right combination of applied regularization techniques such as fully connected dropout, max pooling dropout, L2 regularization and He initialization, it is possible to achieve good results in object recognition with small networks and without data augmentation.

Keywords: Deep learning · Deep convolutional neural networks · Object recognition · Fully connected dropout · Max pooling dropout · L2 regularization

1 Introduction

Visual object recognition is an extremely hard computational problem in computer vision research. It has a lot of potential applications that touch a lot of areas of artificial intelligence including video data mining, object identification for mobile robots, and image retrieval. It searches to identify and localize categories, places and objects in order to recognize and classify images.

Visual object recognition has gained the interest of the research community and has been further applied successfully to a lot of other application areas [1–5]. However, it is still an open problem and a challenging task. The core problem is due to the high variability of the objects constituting an image. In fact, the object may have variation in the view point, the illumination, the scale and the imaging conditions [6, 7].

Recently, deep learning, especially the Convolutional Neural Networks (CNNs), has attracted huge attention among computer vision research communities thanks to its high performance in classification tasks [8, 9]. It has produced extremely promising results for various tasks of pattern recognition issues like handwritten digits, face recognition, sentiment analysis, object detection and image classification [9–11]. Those models have some advantages and disadvantages. Indeed, the main advantage of Deep

D. Liu et al. (Eds.): ICONIP 2017, Part II, LNCS 10635, pp. 431–439, 2017.
https://doi.org/10.1007/978-3-319-70096-0_45

CNNs (DCNNs) is their accuracy in image recognition problems. Also, they are very good at discovering high-dimensional data having intricate structures [8]. On the other hand, they have some disadvantages such as the high computational cost, so if you do not have a good GPU, they will quite slow to train the model (for complex tasks) and they will need a lot of training data.

This paper introduces a DCNN model to obtain a high multi-classification accuracy on object recognition. The architecture of the CNN model is neatly elaborated to extract deep hidden features and model small training datasets, which fits well for the used datasets. The CIFAR-10 and STL-10 datasets are labeled subsets of an 80-million-tiny-image dataset. These datasets are used to evaluate the performance of the CNN model. The experiment results prove that the CNN model with the right combination of regularization techniques has an adaptive accuracy rate on classification.

The main contributions of this study are the following:

- The method uses the CNN to classify images into 10 categories and produce an accuracy of 97% utilizing the CIFAR-10 and 75.4% using the STL-10.
- The CNN model utilizes three convolution layers, a regularization layer, and a high efficiency optimizer to be adaptive to the CIFAR and STL datasets.

The remaining of this paper is set as follows: In the second section, we present some related work. In the third section, we describe the details of applying the CNN model to object recognition. Our experimental study and results using this system are provided in the fourth section. In the final section, we present some concluding remarks and future directions.

2 Related Work

Although the problem of object recognition is still a very active and challenging task, good results have been recorded thanks to the new learning capabilities offered by deep neural networks.

In this context, Tobias et al. in [10] presented the implementation of light-weight CNN schemes on mobile devices for domain-specific objection recognition tasks. In the same optic, the DCNNs were investigated by Lorandet et al. in [12] for RGB-D based object recognition. The DCNNs outperformed other classifiers and proved a significant classification accuracy.

Krizhevsky in [13] suggested a large DCNN to classify 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into 1000 different classes. The model achieved top-1 and top-5 error rates. The neural network was composed of five convolutional layers, some of which were followed by max-pooling layers and three fully-connected layers with a final 1000-way softmax. In order to accelerate the training process, non-saturating neurons and a very efficient GPU implementation of the convolution operation were used. Nevertheless, to minimize overfitting, some regularization techniques like the dropout proved to be very effective.

In handwritten digit recognition, Calderon et al. in [14] and Alwzwazy in [15] proposed a robust DCNN for classification, which achieved superior results. A combination of the CNNs and the RNNs was presented by Peris in [16] and applied for the

generation of video and image descriptions. These models demonstrated that they outperformed the previous state of the art.

Indeed, Haiteng in [11] put forward advanced DCNNs for body constitution to simulate the function of pulse diagnosis, which is able to classify an individual's constitution, based on their pulse. The CNN model employed the latest activation unit and rectified the linear unit and the stochastic optimization. This model attained a recognition accuracy of 95% on classifying nine constitutional types.

Peyrard et al. in [17] proposed a blind approach to super-resolution based on the CNN architecture. The network could deal with different blur levels without any a priori knowledge of the actual kernel utilized to give LR images. The obtained results showed the success of the suggested approach for the blind set-up and were comparable with non-blind approaches.

A deep-neural-network-based estimation metric was investigated by Sholomon et al. in [18] to solve the jigsaw puzzle problem. The proposed metric indicated an extremely high precision even without extracting any manual feature.

Two CNN architectures were presented by Garcia et al. in [19] for emotion recognition in order to classify images into seven emotions. The first architecture checked the effects of minimizing the number of deep learning layers. However, the second architecture horizontally divided the given image into two streams based on eye and mouth positions. This method performed good results compared it other approaches proposed in the literature.

3 Object Recognition Based on DCNN Model

The success of any DCNN comes from the efficient use of GPUs, Rectified Linear Units (ReLUs), a new regularization technique such as a max pooling dropout, a fully connected dropout, and techniques for data augmentation to generate more training examples by deforming the existing ones. In the following section, we describe the major improvements and overall architecture of the DCNN model.

3.1 Initialization and Stochastic Optimization

Before starting to learn the parameters of the network, we must initialize its parameters. An initialization establishes the probability distribution function for the initial weights. The model uses a uniform initialization such as the He weight initialization. This initialization method effectively resolves the bottleneck of the extremely deep neuronal network training [11]. Yet, in order to optimize and update various CNN parameters, the Adam algorithm is used. It is a simple and efficient computational algorithm for optimization based on the gradients of stochastic objective functions. This algorithm is well suited for a CNN with a complex structure and large parameter spaces and it combines the strength of two newly popular optimization methods: the ability of AdaGrad to cope with sparse gradients and the ability of RMSProp to handle non-stationary objectives.

3.2 Leaky Rectified Linear Unit

In general, to learn a neural network, the saturated counterpart, such as a hyperbolic tangent or a logistic sigmoid, is used. However, in recent years, the most popular activation function for the deep network is the ReLU. It calculates the following function:

$$f(x) = \max(0, x). \tag{1}$$

There are some advantages in using the ReLU. First, the ReLU is faster to calculate because it does not require any normalization or exponential calculation (like those required in sigmoid activations or tanh). Second, the use of the ReLU accelerates the convergence of the stochastic gradient descent. This is argued to be caused by its linear and non-saturating form. Third, it does not face the problem of gradient degradation as for the sigmoid and tanh functions. It has been demonstrated that deep networks can be trained effectively utilizing the ReLU even without pre-training.

3.3 Over-Fitting Prevention and Regularization

Learning the CNNs uses a large number of layers, a huge number of units, and connections related to its complex structure and numerous filters in each convolutional layer. These are prone to overfitting, which is a serious problem. To deal with this problem, dropout learning and regularization methods have been developed to improve the CNN performance and reduce overfitting.

L2 and ridge regularization
In order to reduce the regression coefficient overfitting, regularization penalties are appended to CNN parameters. In fact, the L2 regularization and ridge are used in a fully connected layer. L2 weight regularization penalizes weight values by adding the sum of their squared values to the error term to drive all weights to smaller values. Nevertheless, ridge regularization decreases the approximated regression coefficients towards surmount overfitting, which is caused by high dimensionality [7]. The penalty parameter is set to 0.01.

Max pooling dropout
In recent years, Wu and Gu in [20] proposed to use a special dropout variant with the CNN, known as the max pooling dropout. Actually, the traditional CNN is composed of alternating convolutional and pooling layers, with fully-connected layers on top. However, the max pooling dropout can be seen as a special variant of stochastic pooling. It is used within the pooling layers to introduce stochasticity into the learning process with the difference that activations are utilized with a probability proportional to their rank, instead of the strength of their activation.

Fully connected dropout
Dropout learning is used in deep learning to avoid overfitting. A small number of data compared with the size of a network may cause overfitting [21]. Dropout learning follows two processes. At the training stage, some hidden units are neglected with a probability p, and this process reduces the network size. If a dropout probability p of

0.5 is used, roughly half of the activations in each layer will be deleted for every training sample, thus preventing hidden units from relying on other hidden units being present. At the testing stage, the neglected inputs and hidden units are combined with the learned hidden units and multiplied by p to express the final output. As a result, the weights are rescaled proportional to the dropout probability. For example, for a dropout probability of 0.5, all weights are divided by two [21]. This regularization can improve the network performance and significantly reduce the error rate.

3.4 Architecture of Proposed DCNN

The suggested system is presented in Fig. 1. The CNN architecture is composed of six layers: three convolutional layers with 15, 20 and 25 filters, where each filter has a size of 5*5 and each convolutional layer is followed by a max pooling layer of a size of 2*2; two fully connected layers with 600 and 300 units performed after the convolutional layers, and the softmax layer, which is the final layer of the CNN model classifying the output into 10 class labels.

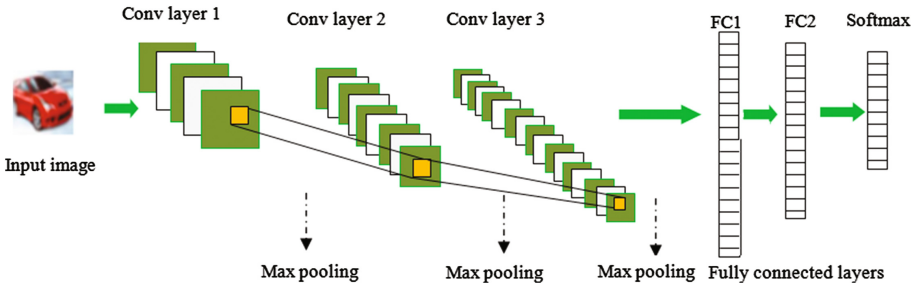

Fig. 1. Overall CNN architecture

A dropout layer is applied to the output with a probability of 0.5 on the 1st, 2nd and 3rd convolutional layers and to two fully connected layers. With this fixed architecture we then proceed to test the effects of the different regularization techniques on the set classification task.

4 Experimentation

We conduct our experimental studies using the proposed DCNN for object recognition. This architecture and the previously described regularization methods are trained and tested to classify images from two datasets: CIFAR-10 and STL-10. The next section introduces the dataset and the overall performance of the DCNN model.

4.1 Database Description

To evaluate the performance of the proposed system, the experiments are conducted on the benchmark object recognition datasets: CIFAR-10 and STL-10.

The CIFAR-10 dataset [13] consists of 60,000 color images of 32 × 32 pixels in 10 classes: airplanes, automobiles, birds, cats, deers, dogs, frogs, horses, ships, and trucks. The total dataset is split into 50,000 training images and 10,000 testing ones. The last 10,000 training images are used for validation. Here are the classes in the dataset, as well as 10 random images from each class (Fig. 2):

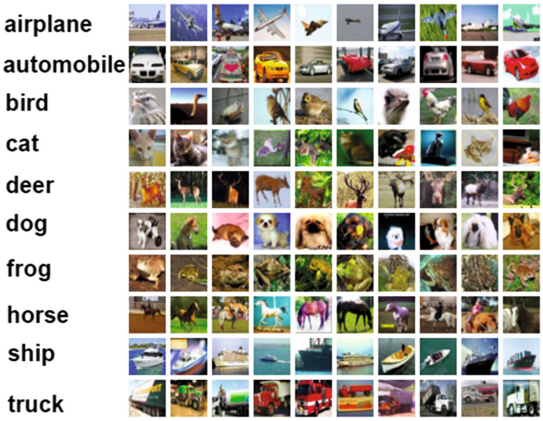

Fig. 2. CIFAR-10 dataset

We use also the STL-10 dataset that contains 96 × 96 RGB images in 10 categories. This dataset has 5,000 labeled training images and 8,000 test images. Additionally, it includes 100,000 unlabeled images for unsupervised learning algorithms, which are extracted from a similar but broader distribution of images [22].

4.2 Results and Discussion

We perform classification experiments on CIFAR-10 and STL-10. Then, we proceed to compare the effects of various regularization methods on seven different classifiers. The following seven settings are tested:

1. No regularization
2. CNN withLReL
3. CNN with LReL, max pooling dropout
4. CNN with LReL, max pooling dropout, fully connected dropout
5. CNN with LReL, max pooling dropout, fully connected dropout, L2
6. CNN with LReL, max pooling dropout, fully connected dropout, L2, ADASYS
7. CNN with LReL, max pooling dropout, fully connected dropout, L2, He

The results of each method are illustrated in Fig. 3a and b. Rate recognition is defined as to the number of correctly recognized samples divided by the total number of test samples. The objective is to classify the input image into 10 class labels. Using CIFAR-10, the initial CNN model achieves a higher recognition rate of 75.5% with the LReL as shown by Fig. 3a. With the LReL and the max pooling dropout, the recognition rate significantly increases by 25.5%. With the addition of the fully connected dropout, the recognition rate goes up by 3.08%. Applying the L2 regularization slightly raises the rate to 94.3%. With the ADASYS method, the rate decreases to 90.07%. The final CNN model with the He initialization gives the highest performance of 97.15%.

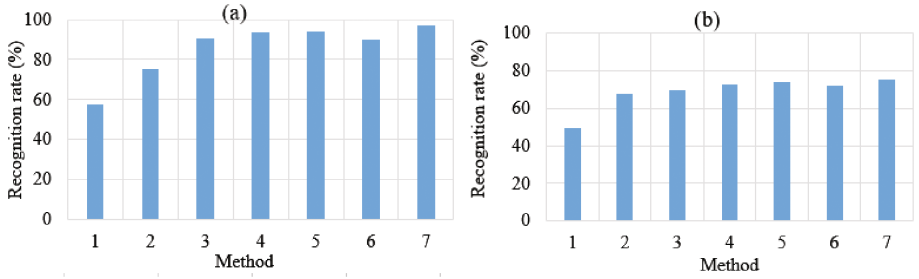

Fig. 3. (a) Comparison of results on CIFAR-10 dataset and (b) Comparison of results on STL dataset

The experimental results show the outperformance of the DCNN based on the max pooling dropout and the fully connected dropout compared to the standard CNN. In fact, the DCNN model demonstrates its superiority based on a very complex and high dimensional dataset with limited samples and without any data augmentation. However, the training and test DCNNs are time-consuming tasks due to the implementation of the fully connected dropout and the max pooling dropout.

5 Conclusion

In this paper, we have presented a DCNN model for object recognition and examined the effects of regularization techniques on the training of DCNNs. This regularized model is able to surmount the shortcomings of traditional recognition methods and improves the multi-classification recognition rate. The experiments have proven that the combination of the DCNN with the max pooling dropout and fully connected dropout can avoid the problem of overfitting. In addition, we have shown that the right combination of regularization techniques can have a big impact on the performance of DCNNs and their trained features by giving an adaptive recognition rate on an extremely complex dataset. As a perspective, these regularization techniques can be used together with data augmentation and more complex CNNs with more filters or more layers, to potentially achieve good results and minimize the execution time on challenging datasets.

References

1. Bai, S.: Growing random forest on deep convolutional neural networks for scene categorization. Expert Syst. Appl. **71**, 279–287 (2017)
2. Zhao, W., Xiong, L., Ding, H.: Automatic recognition of loess landforms using Random Forest method. J. Mt. Sci. **14**(5), 885–897 (2017)
3. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images (2009)
4. Gecer, B., Azzopardi, G., Petkov, N.: Color-blob-based COSFIRE filters for object recognition. Image Vis. Comput. **57**, 165–174 (2017)
5. Liang, M., Hu, X.: Recurrent convolutional neural network for object recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3367–3375 (2015)
6. Dicarlo, J., Cox, D.: Untangling invariant object recognition. Trends Cogn. Sci. **11**(8), 333–341 (2007)
7. Zhang, L., He, Z., Liu, Y.: Deep object recognition across domains based on adaptive extreme learning machine. Neurocomputing **239**, 194–203 (2017)
8. Lecun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
9. Chen, W., Wilson, J.T., Tyree, S., Weinberger, K.Q., Chen, Y.: Compressing convolutional neural networks. arXiv preprint arXiv:1506.04449 (2015)
10. Tobias, L., Ducournau, A., Rousseau, F.: Convolutional neural networks for object recognition on mobile devices: a case study. In: IEEE 23rd International Conference on Pattern Recognition (ICPR), pp. 3530–3535 (2016)
11. Li, H., Xu, B., Wang, N., Liu, J.: Deep convolutional neural networks for classifying body constitution. In: Villa, A.E.P., Masulli, P., Pons Rivero, A.J. (eds.) ICANN 2016. LNCS, vol. 9887, pp. 128–135. Springer, Cham (2016). doi:10.1007/978-3-319-44781-0_16
12. Madai-Tahy, L., Otte, S., Hanten, R., Zell, A.: Revisiting deep convolutional neural networks for RGB-D based object recognition. In: Villa, A.E.P., Masulli, P., Pons Rivero, A. J. (eds.) ICANN 2016. LNCS, vol. 9887, pp. 29–37. Springer, Cham (2016). doi:10.1007/978-3-319-44781-0_4
13. Krizhevsky, I., Sutskever, A., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems (NIPS), pp. 1097–1105 (2012)
14. Calderon, A., Roa, S., Victorino, J.: Handwritten digit recognition using convolutional neural networks and gabor filters. In: Proceedings of the International Congress on Computational Intelligence (2003)
15. Alwzwazy, H.A., Albehadili, H.M., Alwan, Y.S.: Handwritten digit recognition using convolutional neural networks (2016)
16. Peris, Á., Bolanos, M., Radeva, P.: Video description using bidirectional recurrent neural networks. arXiv preprint arXiv:1604.03390 (2016)
17. Peyrard, C., Baccouche, M., Garcia, C.: Blind super-resolution with deep convolutional neural networks. In: Villa, A.E.P., Masulli, P., Pons Rivero, A.J. (eds.) ICANN 2016. LNCS, vol. 9887, pp. 161–169. Springer, Cham (2016). doi:10.1007/978-3-319-44781-0_20
18. Sholomon, D., David, Omid E., Netanyahu, Nathan S.: DNN-Buddies: a deep neural network-based estimation metric for the jigsaw puzzle problem. In: Villa, A.E.P., Masulli, P., Pons Rivero, A.J. (eds.) ICANN 2016. LNCS, vol. 9887, pp. 170–178. Springer, Cham (2016). doi:10.1007/978-3-319-44781-0_21
19. Ruiz-Garcia, A., Elshaw, M., Altahhan, A., Palade, V.: Deep learning for emotion recognition in faces. In: Villa, A.E.P., Masulli, P., Pons Rivero, A.J. (eds.) ICANN 2016. LNCS, vol. 9887, pp. 38–46. Springer, Cham (2016). doi:10.1007/978-3-319-44781-0_5

20. Wu, H., Gu, X.: Towards dropout training for convolutional neural networks. Neural Netw. **71**, 1–10 (2015)
21. Hara, K., Saitoh, D., Shouno, H.: Analysis of dropout learning regarded as ensemble learning. arXiv preprint arXiv:1706.06859 (2017)
22. Miclut, B.: Committees of deep feedforward networks trained with few data. In: Jiang, X., Hornegger, J., Koch, R. (eds.) GCPR 2014. LNCS, vol. 8753, pp. 736–742. Springer, Cham (2014). doi:10.1007/978-3-319-11752-2_62