Image Classification Using Network In Network

Qasim Mahdi haref
Al-Imam-kadhum college for Islamic science

Abstract*: Recently, image classification has become vital task using several methods. In this work, to achieve better performance comparing with other models, we adapted robust model of Convolutional Neural Networks (CNN) called Network In Network (NIN). One of the limitations of CNN models that they linearly aggregate input patterns of prior layers which can lead to hardly draw strong features from the patterns. However, to diminish this weakness inherited from those models NIN proposed a new technique to highly avoid local aggregation of given inputs. Thus, in this paper we revisit the model and we enhance further its performance by introducing and analyzing different parameters that can widely enhance model efficiency. Furthermore, different challenging benchmarks are used for evaluation. Specifically, CIFAR-10, CIFAR100, and MNIST are used in our final experiments. We showed that our model surpasses many former models evaluated on the same datasets.

Keywords: *Network In Network,* Convolutional Neural Networks

1. INTRODUCTION

CNNs achieved convolution in the lower layers of the network. In the classification, the feature maps of the last convolutional layer are vectored and fed into fully connected layers followed by a softmax layer [1]. We used the strategy called global average pooling presented by Min Lin at al. [2]. It makes one feature map for each corresponding category of the classification task in the last mlpconv layer. It replaces fully connected layers and the resulting vector is fed directly into the softmax layer. The authors argue that the linear filtering operation in the convolution layers is not expressive enough, leading to a necessity of many layers stacked on top of each other.

2. LITERATURE REVIEW

In recent years, neural networks and convolutional neural network currently represent dominated solutions to many problems in image recognition. Convolutional neural network is considered because it achieves state-of-the-art results for variety of computer vision tasks [3]. Xiao-Xiao Niu et al. [4] designed a novel hybrid CNN–SVM model for handwritten digit recognition. The hybrid model automatically extracts features from the raw images and generates the predictions.Matthew D. Zeiler et al. [5] improving on Krizhevsky et al. ’s (Krizhevsky et al., 2012) impressive ImageNet 2012 result. The author introduces a novel visualization technique that gives insight into the function of feature layers and the procedure of the classifier. Matthew D. Zeiler et al. have observed ImageNet model generalizes well to other datasets: when the softmax classifier is retrained. Hayder M. Albehadili et al. [1] have performed a new CNN architecture which achieves state-of-the-art classification results on the different challenge benchmarks. In their study, they showed on MNIST, CIFAR-10, and CIFAR-100 datasets. We investigate and demonstrate a powerful DC NIN’s method used for classification. Not only designing powerful NIN is presented but also critical parameters of CNN is carefully selected and tuned to produce final concrete model which achieves superior results. classification is illustrating prototype problem for learning about deep neural networks in general. CNN is a valuable method applied for variety of applications.

3. METHODOLOGY

Multilayer perceptron Convolution Layers (MLP) used a universal function for feature extraction of the local patches. Radial basis network and multilayer perceptron are two well-known universal functions approximates. Using multilayer perceptron is compatible with the structure of convolutional neural networks t. The new type of layer is called mlpconv [1]. It replaces the traditional conventional layer to convolve over the input. In the fig. 1 show the new layer mlpconv.

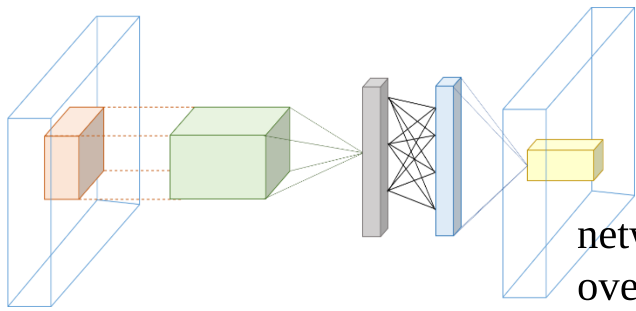
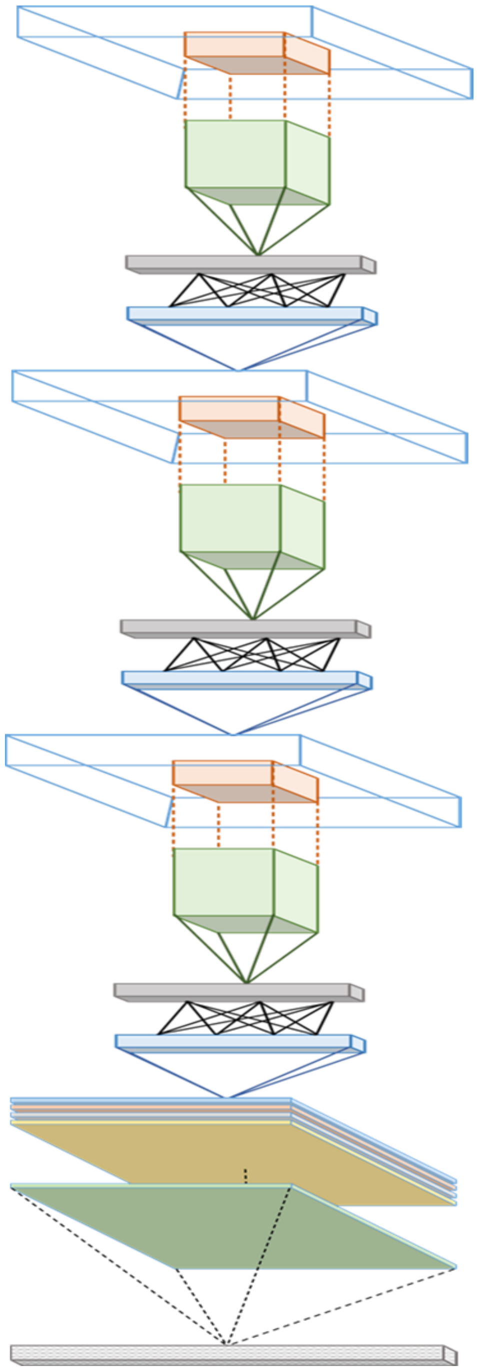


Fig. 1: Network in network layer

In fig. 1, we have a two-layer network, with layers X and Y, being slid over the input channels. In the fig. above, the neurons in the X box are actually the same as a traditional convolutional layer each one corresponds to a linear filter and non-linearity. For instance, if we have 16 neurons in the X box, it is the same as a convolutional layer with 16 filters/output channels for each neuron in the Y box takes a linear combination of the outputs of neurons in X, and pass those through a non-linearity. Layer Y is close to having a pack of 1x1 convolutional layers. If there are more layers in the "inner network", it achieves 1x1 convolutional layer operation again on the output of Y [7]. In another word, the idea is to generate one feature map for each category of the classification task in the last layer, take the average of each feature map, and the resulting vector is fed directly into the SoftMax layer.

The complete structure of NIN is a stack of mlpconv layers, with the global average pooling. Sub-sampling layers would be added in between the mlpconv layers as in CNN and maxout networks. In the fig. 2 shows the structure which we used in our work. three mlpconv layers are used and in



each mlpconv layer; there are a three-layer perceptron.

Fig.2: A simple 3 layer NIN + Global Average Pooling

4. EXPERIMENTAL SAETUP

To further analysis and evaluate our model architectures, we used three different datasets which are heavily used before. The first dataset used in our experiments is MNIST [3] which is a standard and large database of handwritten digits. The second dataset is CIFAR-10 [8] where it has 10 classes and it is more challenge. The last dataset that we used in our work CIFAR-100 [8] and this dataset are similar to CIFAR-10 but it has 100 classes containing 600 images each. It is one of challenge dataset that used in image classification, the size image for CIFAR-100 are similar to CIFAR-10 but the different in class number where CIFAR-100 have 100 class and CIFAR-10 have 10 class.

- **MNIST dataset**

It is widely used to trainings based on MNIST dataset in the literature, suggesting much diverse approach. One of the major tasks in the recognition of handwritten digits is the within class variance, Therefore, the best way to get different class by handwriting digit because people write the digit in different way. The MNIST designed SD-3 for training set and SD-1 for test set. It is a worth mention that SD-3 is much cleaner and easier than SD-1. These datasets are collected from 500 writers, training samples SD-3 that was taken from American Census Bureau employees and the test samples SD-1 that was taken from American high school students. MNIST database contains 70,000 digits from (0 – 9) for training the digit recognition system used 60,000, and rest digits as test data. The size original black and white images were fit to 20 x 20 pixel box. In recently work used 28 x 28 pixel box. In our work, For each digit is normalized and centered with size 28x28 as the features [3].

Table 1 shows our final results comparing with state-of-the-art results of prior works. It is noticeable that our model outperforms all former models. The result achieved on MNIST dataset is 0.44%which is highest results compare with others.

Table 1: MNIST classification errors of various methods

Reference method	Reference	Error %
SVM	**[11]**	**1.4 %**
LeNet5	**[3]**	**0.95 %**
VSVM	**[11]**	**0.8 %**
boosted-LeNet4	**[3]**	**0.7 %**
VSVM2	**[11]**	**0.68 %**
Unsupervised Learning	**[9]**	**0.64 %**
K-NN	**[11]**	**0.63 %**
Sparsenet	**[12]**	**0.59 %**
VSVM2+deskewing	**[10]**	**0.56 %**
2-LayerCNN+2-Layer NN	**[13]**	**0.53%**
Stochastic Pooling	**[14]**	**0.47%**
Ours	**Ours**	**0.44%**

In addition, we showed how the error loss can gradually drop with iteration as shown in fig. 3.

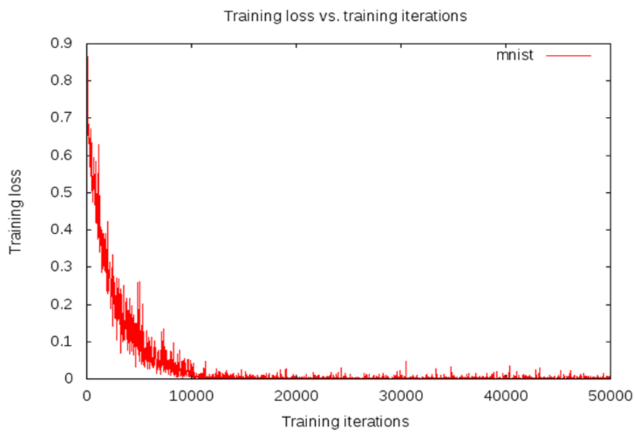


Fig. 3: The fig. shows that the error loss can gradually drop with iteration in MNIST dataset.

- **CIFAR-10 dataset**

Reference method	**Reference**	**Accuracy %**
Logistic regression	**[15]**	**36.0%**
Support Vector Machines	**[16]**	**39.5%**
GIST	**[15]**	**54.7%**
SIFT	**[16]**	**65.6%**
fine-tuning GRBM	**[17]**	**64.8%**
GRBM two layers	**[17]**	**56.6%**

mcRBM	**[15]**	**68.3%**
mcRBM-DBN	**[15]**	**71.0%**
Tiled CNNs	**[18]**	**73.1%**
Improved LCC	**[19]**	**74.5%**
Fast-Learning Shallow +CNN	**[20]**	**75.86 %**
KDES + EMK + linear SVMs	**[16]**	**76.0%**
PCANet	**[21]**	**78.67 %**
Convolutional RBM	**[22]**	**78.9%**
K-means (Triangle, 4k features)	**[22]**	**79.6%**
HKDES + linear SVMs	**[23]**	**80.0%**
Cuda-convnet2	**[28]**	**82.00 %**
Stochastic Pooling	**[14]**	**84.87 %**
Maxout Units	**[26]**	**90.61 %**
Maxout Networks	**[29]**	**90.65 %**
Ours	**Ours**	**92.20 %**

The dataset consists of six batches distributed into five training and one test. The test batch contains are 1000 randomly-selected images from each class. In the training batches contain the remaining images. It is choosing random images for each class [8]. In table 3, the result of the used model is depicted comparing with many prior model results. Again in our experiments we achieve non trivial results comparing with existing models. The results accomplished on CIFAR-10 is 92.20% which is superior results comparing with depicted models in table 2.

Table 2: CIFAR-10 classification errors of various methods.

Fig. 4 shows the training loss vs training iteration. It can be seen that the error highly drops at the first few iterations then it saturates at approximately 2000 iterations.

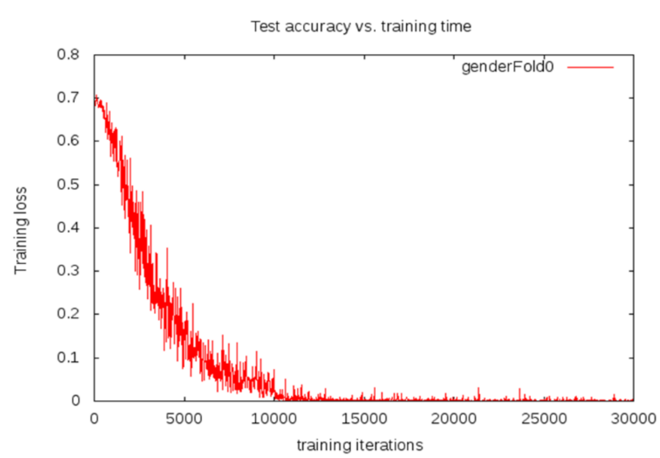


Fig. 4: Training loss vs training iteration in CIFAR-10 dataset

- **CIFAR-100 dataset**

The mode challenging dataset is CIFAR-100 which is a set of natural color image. It has 100 classes containing 600 images each. It divided to 500 training images and 100 testing images per class. The pixels are scaled to be between [0, 1] before the training. The 100 classes in the CIFAR-100 are grouped into 20 super classes. CIFAR-100 dataset is considered is the most challenging dataset because there are rare samples for each class. However, the results achieved are adequate. The results accomplished on CIFAR-10 is 63.32% which is superior results comparing with depicted models in table 3.

Table 3: CIFAR-100 classification errors of various methods.

Reference method	**Reference**	**Accuracy %**
Smooth Pooling	**[24]**	**56.29%**
Stochastic Pooling	**[14]**	**57.49%**
NOMP encoder	**[25]**	**60.8%**
Maxout Networks	**[23]**	**61.43%**
Maxout Units	**[26]**	**61.86%**
Smooth Pooling Regions	**[8]**	**56.29%**
Beyond Spatial Pyramids	**[27]**	**54.23%**
Hybrid PSO-SGD Network1	**[6]**	**53.52%**
Hybrid PSO-SGD Network2	**[6]**	**59.85%**
Ours	**Ours**	**63.32%**

training loss and training iterations are sketched to show how the error behaves during training iterations. Since we used very robust toolbox for the training, error rapidly decreases with each epoch as shown in fig. 5.

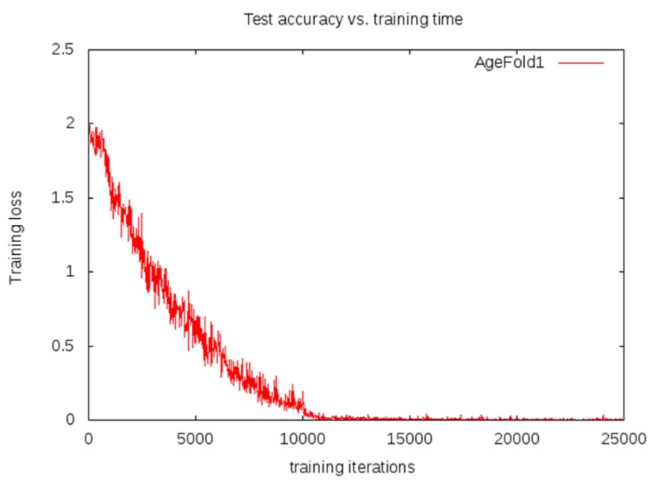


Fig. 5: Training loss and training iterations are sketched to show how the error behaves during training iterations on CIFAR-100 dataset

5. CONCLUSION AND FUTURE WORK

In this work, a robust model of CNN called NIN is recruited in our work. NIN is one of former work which achieves superior results and achieves state-of-the-art on many datasets. We further analyzed and explored parameters that can highly influence model performance. our model is evaluated and test on different datasets. Also, we compare our work with different models of former work, we achieve superior results comparting with other works.

REFERENCES

[1] Hossein Khosravi, Ehsanollah Kabir, " Introducing a very large dataset of handwritten Farsi digits and a study on their varieties" Pattern Recognition LettersVolume 28, Issue 10, 15 July 2007, Pages 1133–1141.
[2] Min Lin, Qiang Chen, and Shuicheng Yan "Network In Network" arXiv 1312.4400v3,4 Mar 2014.

[3] Yann LeCun, L´eon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.
[4] Vapnik and Lerner "Support Vector Machines (SVMs)"introduce the Generalized Portrait algorithm, 1963
[5] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional neural networks," arXiv:1311.2901, 2013.
[6] Hayder M. Albeahdili, Haider A. Alwzwazy, Naz E. Islam." Robust Convolutional Neural Networks for Image Recognition". (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 6, No. 11, 2015.
[7] "Lacking capital" http://bbabenko.tumblr.com/post/ 10705903294/deep-net-highlights-from-2014.
[8] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Master's thesis, Department of Computer Science, University of Toronto, 2009.
[9] Raia Hadsell, Sumit Chopra, Yann LeCun, "Dimensionality Reduction by Learning an Invariant Mapping" CVPR , vol. 2, pp. 1735-1742, 2006.
[10] D. Decoste, B. Sch¨olkopf, Training invariant support vector machines, Machine Learning 46 (2002) 161–190.
[11] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Trans. on Pattern Analysis and Machine Intelligence 24 (2002) 509–522.
[12] K. Labusch Inst. for Neuro- & Bioinf., Univ. of Lubeck, Lubeck E. Barth T. Martinetz," Simple Method for High-Performance Digit Recognition Based on Sparse Coding", IEEE Transactions on Neural Networks, Volume 19 Issue 11, November 2008
[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. FeiFei. Imagenet: A large-scale hierarchical image database. In CVPR, 2009.
[14] M. D. Zeiler and R. Fergus, "Stochastic pooling for regularization of deep convolutional neural networks," arXiv preprint arXiv:1301.3557, 2013
[15] M. Ranzato and G. Hinton. Modeling pixel means and covariances using factorized third-order boltzmann machines. In CVPR, 2010. 1730, 1734
[16] L. Bo, X. Ren, and D. Fox. Kernel Descriptors for Visual Recognition. In NIPS, December 2010. 1729, 1730, 1731, 1732, 1734
[17] M. Ranzato, K. A., and G. Hinton. Factored 3-way restricted boltzmann machines for modeling natural images. In AISTATS, 2010. 1734

[18] Q. Le, J. Ngiam, Z. C. Chia, P. Koh, and A. Ng. Tiled convolutional neural networks. In NIPS. 2010. 1734.
[19] K. Yu and T. Zhang. Improved local coordinate coding using local tangents. In ICML, pages 1215–1222, 2010. 1730, 1734.
[20] Mark D. McDonnell, Tony Vladusich," Enhanced Image Classification With a Fast-Learning Shallow Convolutional Neural Network". arXiv:1503.04596 [cs.NE],2015
[21] David Stutz, "Understanding Convolutional Neural Networks", Seminar Report, August 30, 2014
[22] A. Coates, H. Lee, and A. Ng. An analysis of single-layer networks in unsupervised feature learning. In NIPS*2010 Workshop on Deep Learning, 2010. 1734, 1735.
[23] Liefeng Bo , Kevin Lai , Xiaofeng Ren , Dieter Fox," Object Recognition with Hierarchical Kernel Descriptors", In Proc. of CVPR, 2011.
[24] M. Malinowski and M. Fritz," Learning Smooth Pooling Regions for Visual Recognition", British Machine Vision Conference (BMVC), Conference Paper, 2013 .
[25] Tsung-han Lin and H. T. Kung," Stable and Efficient Representation Learning with Nonnegativity Constraints", Proceedings of the 31st International Conference on Machine Learning (ICML-14), 2014 .
[26] J. T. Springenberg and M. Riedmiller, "Improving deep neural networks with probabilistic maxout units," arXiv preprint arXiv:1312.6116, 2013.
[27] Y. Jia, C. Huang, and T. Darrell, ―Beyond spatial pyramids: Receptive field learning for pooled image features‖, CVPR 2012
[28] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In CVPR, 2006.
[29] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," arXiv preprint arXiv:1302.4389, 2013.