Comparison of Different Deep Structures for Fish Classification

M. Sarigül and M. Avci

Abstract—**The superior performances of convolutional neural networks in various fields have recently enabled deep learning to be popular. One of the most common problems in this regard is the determination of the structure of the deep artificial neural network according to the problem to be solved.**

In this paper, deep convolutional neural networks having different numbers of convolutional layers and different filter sizes are used for classifying challenging fish dataset. The results show that all of the tested structures succeeded in the learning data set, while the less deep structures with larger filters gave better results on the test data set. Increasing size of the filters may provide a performance boost up to 40.73 percent. In addition, tests were done by increasing the number of filters on each convolutional layer of successful structures. This operation led an extra performance boost up to 14.28 percent over the current performance of the structures.

Index Terms—**Fish classification, deep learning, convolutional neural network, image classification.**

I. Introduction

Deep learning is a branch of machine learning which uses multiple layered neural network topology to represent high-level abstractions of data. Using a network including multiple representational layers allows complex distinctive features of data to be realized by the artificial neural network. Deep learning studies has been started at 1940s. However after 2006, it became more popular and started to be called as deep learning. Deep learning algorithms have accomplished a thriving performance on large datasets with images, videos, texts and speeches. This caused the deep structures to be used in many research areas.

Difficulties in determining the convolutional network depth, filter sizes and number of filters according to the problem to be solved are experienced. In this work, deep artificial neural networks with different number of layers and various filter sizes were used for two different classification tasks on QUT fish dataset. Precision is used as a measure of performance in the experiments. Obtained results show that while structures with less number of convolutional layers achieve better performances for both tasks, enlarging filter sizes of the convolution may lead an increase over the performance up to 40.73 percent. It had also been observed that increasing the number of filters on the successful structures may result in an extra performance boost by 14.28% over last obtained performance.

II. Material and Methods

A. Deep Learning

Deep learning is a branch of machine learning based on learning representation of data. Deep learning has been used in many different areas such as image classification as in [1], object recognition as in [2], speech recognition as in [3], natural language processing as in [4] and audio classification as in [5], face representation as in [6], pedestrian detection as in [7] and even playing Atari games as in [8]. It has accomplished state-of-art results on different tasks. A deep neural network can be called as an artificial neural network with more than one hidden layers. Deep neural networks can contain convolutional layers to extract features from input data. Convolutional layers are similar to ordinary neural network layer. They include neurons which have learnable weights and biases. Each neuron takes inputs, executes dot product and selectively followed by activation function. The idea of convolutional neural network was born in 1985 [9]. Firstly it was used for temporal signals [10]. The idea was improved in 1998 [11] and generalized in 2003 [12]. Convolutional layers have been used successfully over image and speech recognition applications (see Fig. 1).

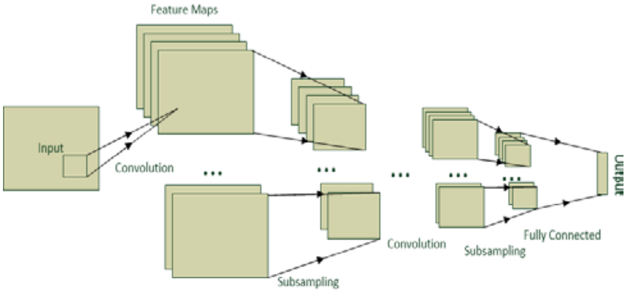

Fig. 1. Deep learning structure.

A standard convolutional layer takes data as an input and uses randomly weighted filters which are also called as kernels to extract different feature maps of the input. The filters are shifted over the image and activation values are calculated. As an example, calculations of a layer of convolution for a 2×2 kernel can be seen in Fig. 2.

Kernels can be selected in any size as needed. Also they don't need to be square, size of width and height of a kernel can be different. In each convolution layer any number of filters can be used according to the task. Each extra layer of convolution increases the complexity of the features that extracted via convolutional part of the network. Number of convolution layers must be carefully selected to be effective on the problem. However too many layer can increase the

Manuscript received January 5, 2017; revised March 23, 2017.

M. Sarıgül is with the Computer Engineering Department, Cukurova University Sarıçam/Adana, Turkey (e-mail: msarigul@cu.edu.tr) & Computer Engineering Department, Iskenderun Technical University, İskenderun/Hatay, Turkey (e-mail: mehmetsarigul@iste.edu.tr).

M. Avci is with the Biomedical Engineering Department, Cukurova University Sarıçam/Adana, Turkey (e-mail: mavci@cu.edu.tr).

DOI: 10.7763/IJCTE.2017.V9.1167

training time of the network and may lead poor performance on the test data.

A convolutional layer is usually followed by a pooling layer. Task of the pooling layer is to reduce the size of the representation to decrease amount of parameters and calculations over the network, thus pooling operation plays an important role in reducing computational complexity. Most common pooling operation is done with 2 by 2 pictures with maximum operation. This operation can be seen in Fig. 3. This operation reduces number of parameters need to be calculated by 75 percent in a layer. Pooling operation can be done with any size of picture. Average pooling and L2-norm pooling are some other pooling operators.

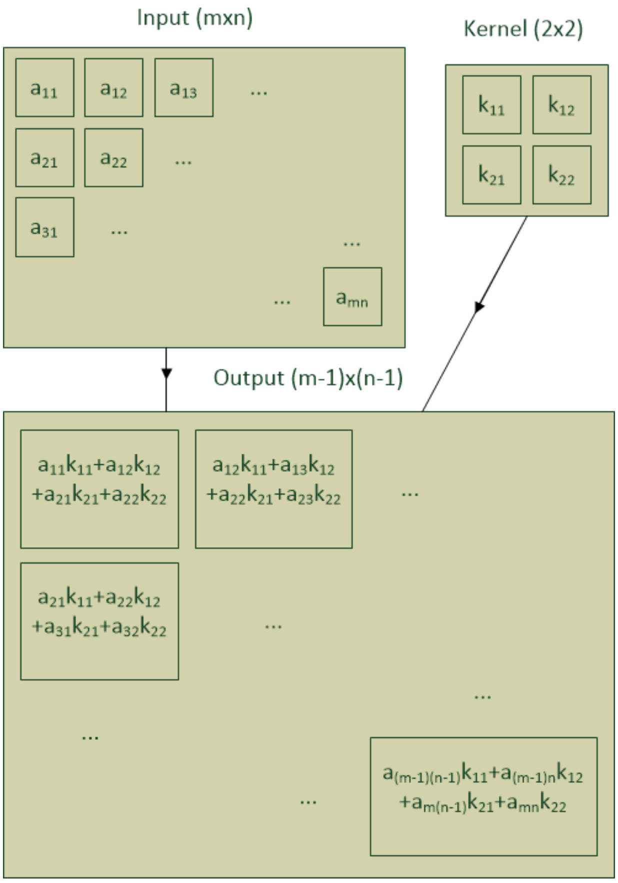
Input (mxn)
Kernel (2x2)
a_{11} a_{12} a_{13} ...
a_{21} a_{22} ...
a_{31} ...
... a_{mn}
k_{11} k_{12}
k_{21} k_{22}
Output (m-1)x(n-1)
$a_{11}k_{11}+a_{12}k_{12}+a_{21}k_{21}+a_{22}k_{22}$
$a_{12}k_{11}+a_{13}k_{12}+a_{22}k_{21}+a_{23}k_{22}$
...
$a_{21}k_{11}+a_{22}k_{12}+a_{31}k_{21}+a_{32}k_{22}$
...
$a_{(m-1)(n-1)}k_{11}+a_{(m-1)n}k_{12}+a_{m(n-1)}k_{21}+a_{mn}k_{22}$

Fig. 2. Convolution example.

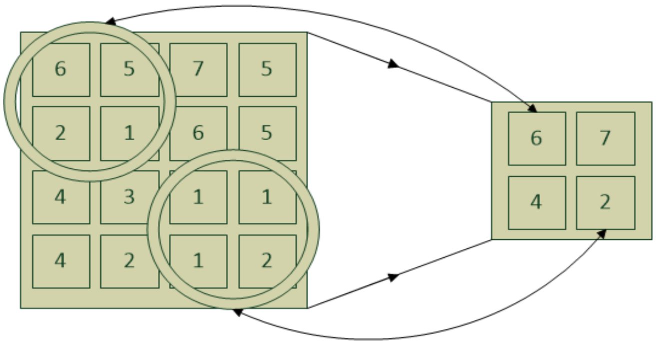

Fig. 3. Pooling operation.

Convolutional layers are followed by fully connected neural network. Neurons in this layer have connections to all neurons in previous layer as in regular neural networks. At the output layer a loss function is needed to calculate the error of the network. An accurate loss function can be used such as softmax loss, sigmoid cross-entropy or Euclidean loss. After computing the error a training algorithm such as back-propagation can be used to train the deep neural network. Standard training methods of neural networks are also suitable for this structure.

B. QUT Fish Data Set

The QUT fish database was used to compare the deep learning structures. This dataset was firstly used in [13] for a method called Local ISV. Local ISV is a classification method using different classes of data for feature extraction, training operation and testing operation, therefore there is no possibility to compare its performance directly with deep learning structures.

This dataset contains 3960 images of 468 species of fish in different environments. The pictures are divided into 3 different classes in terms of their backgrounds. The first one includes pictures of fish taken with a constant background. The second one contains pictures of fish that are not in water but without background regulation. The last one includes pictures of fish taken in its natural environment. Some sample pictures from the dataset can be seen in Fig. 4. Because it contains very few images for some fish species, two different datasets containing pictures of 98 and 93 species of fish selected from this dataset to be used for training task.

The fact that the fish are quite similar to each other makes it very difficult to determine the fish species. Obtained best performance in the previous work is about 49.3 percent.

The performances of the deep learning networks of different structures were compared on this task. The images of fish in RGB format, cut into 160×64 pixels size, were the inputs of the deep learning neural networks.

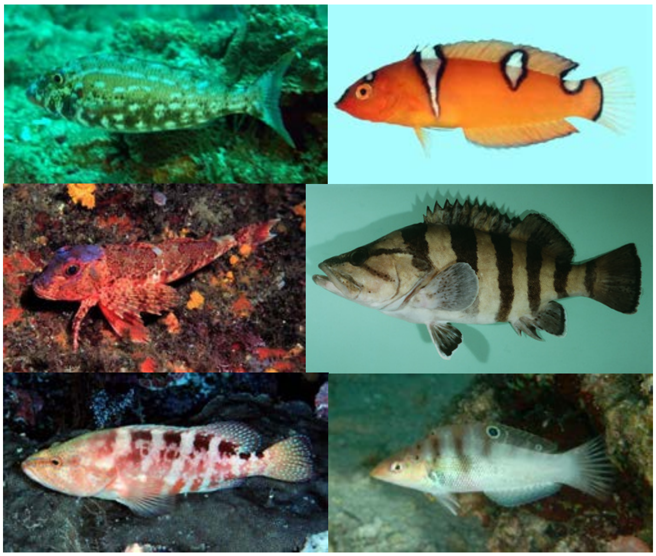

Fig. 4. Some samples from QUT fish dataset.

III. Experiments

Artificial neural networks with different number of layers have been tested on the data sets for filters with different sizes. The general idea is to choose number of convolutional layers a few for this kind of classification tasks. Larger networks need a huge amount of learning time. Therefore as numbers of convolutional layers, 3, 4 and 5 were determined for the experiments. For each neural network, small, medium and large filters were used. Each convolutional layer of the neural networks contains 128 filters respectively. Each convolutional layer is followed by a pooling layer which carrying out a max pooling operation. Depending on the depth of the network various size of pictures are used for pooling operation for different structures of neural networks. For all trained structures having same number of layers, numbers of the neurons in the fully connected part were also kept the same to ensure that this part of the neural network does not affect the classification performance. Artificial neural network structures used in experiments can be seen in Table I, Table II and Table III.

TABLE I: 3 CONVOLUTIONAL LAYERED STRUCTURES

Layer No	Small Size Filtered NN	Medium Size Filtered NN	Large Size Filtered NN
1	2x2 Convolution	3x3 Convolution	5x5 Convolution
2	3x3 Pooling	2x2 Pooling	3x2 Pooling
3	2x2 Convolution	4x4 Convolution	5x5 Convolution
4	4x2 Pooling	4x2 Pooling	3x2 Pooling
5	2x2 Convolution	4x3 Convolution	5x5 Convolution
6	4x4 Pooling	4x4 Pooling	3x3 Pooling
7	128 Neuron MLP	128 Neuron MLP	128 Neuron MLP
8	128 Neuron MLP	128 Neuron MLP	128 Neuron MLP
9	98 or 93 Neuron Softmax Output	98 or 93 Neuron Softmax Output	98 or 93 Neuron Softmax Output

TABLE II: 4 CONVOLUTIONAL LAYERED STRUCTURES

Laye r No	Small Size Filtered NN	Medium Size Filtered NN	Large Size Filtered NN
1	3x3 Convolution	3x3 Convolution	5x5 Convolution
2	2x2 Pooling	2x2 Pooling	2x2 Pooling
3	2x2 Convolution	4x4 Convolution	5x5 Convolution
4	3x2 Pooling	2x2 Pooling	2x2 Pooling
5	3x2 Convolution	2x2 Convolution	4x4 Convolution
6	2x2 Pooling	3x3 Pooling	2x1 Pooling
7	3x2 Convolution	3x2 Convolution	3x2 Convolution
8	2x2 Pooling	2x1 Pooling	3x3 Pooling
9	128 Neuron Fully Connected	128 Neuron Fully Connected	128 Neuron Fully Connected
10	128 Neuron Fully Connected	128 Neuron Fully Connected	128 Neuron Fully Connected
11	98 or 93 Neuron Softmax Output	98 or 93 Neuron Softmax Output	98 or 93 Neuron Softmax Output

Each dataset was used for two different classification tasks. One of the tasks is to predict the species of fish from pictures that have regularized background. Since there will not be anything to mislead the neural network, it is easier to predict fish species from the images with edited background. The other classification task is to estimate the species of fish from pictures taken in the natural environment of the fish. It is relatively a harder classification task because of the background of the picture makes it harder to realize the fish. These tasks are called “first classification task” and “second classification task” through the paper. Each experiment was repeated ten times and test results were averaged. After that, the filter numbers of the structures with the best performance for each classification task were increased to 4 times and the tests were repeated for the altered structures. Torch framework was used for all experiments ([14]).

TABLE III: 5 CONVOLUTIONAL LAYERED STRUCTURES

Laye r No	Small Size Filtered NN	Medium Size Filtered NN	Large Size Filtered NN
1	3x3 Convolution	3x3 Convolution	5x5 Convolution
2	2x2 Pooling	2x2 Pooling	2x2 Pooling
3	2x2 Convolution	4x4 Convolution	5x3 Convolution
4	2x2 Pooling	2x1 Pooling	2x1 Pooling
5	2x2 Convolution	3x3 Convolution	4x3 Convolution
6	2x2 Pooling	2x2 Pooling	2x2 Pooling
7	2x2 Convolution	4x3 Convolution	4x4 Convolution
8	2x1 Pooling	1x1 Pooling	2x2 Pooling
9	2x1 Convolution	4x3 Convolution	4x3 Convolution
10	2x2 Pooling	3x3 Pooling	1x1 Pooling
11	128 Neuron Fully Connected	128 Neuron Fully Connected	128 Neuron Fully Connected
12	128 Neuron Fully Connected	128 Neuron Fully Connected	128 Neuron Fully Connected
13	98 or 93 Softmax Output Layer	98 or 93 Softmax Output Layer	98 or 93 Softmax Output Layer

IV. RESULTS

Obtained results show that 3 convolutional layered structures were more successful for both tasks. It was also observed that increasing the filter size in the majority of experiment also increased the performance.

Using large size filters made a performance boost up to 17.28 percent for the structures with 3 convolutional layers on the first classification task. The filters with different sizes gave different results for 4 convolutional layered structures for two dataset. While using medium sized filters instead of small sized filters, provided 3.8 percent of performance boost for the first dataset in the first classification task, using large sized filters instead of small sized ones, provided 22.99 percent of performance boost for the second dataset in the first classification task. There were the medium-sized filters that give the best result for 5-layer structures for the first classification task. Using medium size filters made a performance boost up to 20.37 percent for the structures with 5 convolutional layers on the first classification task. Performance of different structures for the first classification task can be seen in Table IV.

TABLE IV: PERFORMANCE OF THE CLASSIFICATION TASK 1

	Filter Size	Dataset 1	Dataset 2
3 convolutional layers	Small	39.24	31.53
	Medium	43.01	39.08
	Large	46.02	42.24
4 convolutional layers	Small	39.78	31.53
	Medium	41.29	37.75
	Large	38.49	38.78
5 convolutional layers	Small	36.24	28.57
	Medium	40.32	34.39
	Large	40.22	33.47

Using large size filters makes a performance boost up to 21.71 percent for the first dataset and 40.73 percent for the second dataset for the structures with 3 convolutional layers on the second classification task. There were the large-sized filters that give the best result for 4-layer structures for the second classification task. Using large size filters made a performance boost up to 17.37 percent for the first dataset and 27.09 percent for the second dataset for the structures with 4 convolutional layers on the second classification task. The filters with different sizes gave different results for 5 convolutional layered structures for two dataset for the second classification task. While using large sized filters instead of small sized filters, provided 35.89 percent of performance boost for the first dataset in the second classification task, using medium sized filters instead of small sized ones, provided 25.00 percent of performance boost for the second dataset in the second classification task. Performance of different structure for the second classification task can be seen in Table V.

TABLE V: PERFORMANCE OF THE CLASSIFICATION TASK 2

	Filter Size	Dataset 1	Dataset 2
3 convolutional layers	Small	25.38	24.80
	Medium	30.54	29.69
	Large	**30.89**	**34.90**
4 convolutional layers	Small	22.90	22.96
	Medium	22.47	26.73
	Large	26.88	29.18
5 convolutional layers	Small	19.78	22.04
	Medium	25.91	27.55
	Large	26.88	26.12

The neural network structure with 3 convolutional layers and large filters which has the most accurate performance in the both classification tasks was altered by increasing the number of filters in each convolutional layer from 128 to 512 and tested on the same tasks. This process made a performance boost up to 14.28 percent for the first dataset and 6.30 percent for the second data set on the first classification task. Same process also made a performance boost up 7.90 percent for the first dataset and 8.17 percent for the second data set on the second classification task. This shows that increasing the number of filters without changing the artificial neural network structure allows an undeniable improvement in the test data set performance.

V. CONCLUSION

Deep learning structures have resulted successful results in many studies, which is why these constructs are commonly being used for many tasks. However the answer to the question of which depth and structure must be used according to the selected task is still an open question.

In this work, convolutional neural networks with 3, 4 and 5 convolutional layers and various filter sizes are used for two different classification tasks over QUT fish dataset. Obtained results show that while all the structures were able to classify the learning data by 100 percent efficiency, larger filtered structures with less convolutional layers were more successful over the test data. 3 layered and large filtered structures were the most successful structures over the dataset. It is obtained that enlarging filters may lead a performance increase up to 40.73 percent for 3 convolutional layered structures, 27.09 percent for 4 convolutional layered structures, 35.89 percent for 5 convolutional layered structures.

Increasing the number of filters, instead of increasing the depth or filter size, also leads to increased performance, as well as another result observed in the tests. Using 512 filters instead of 128 filters increased the performance of the structure with best performance from 46.02 percent to 52.59 percent for the first classification task and from 30.89 percent to 33.33 percent for the second classification task for the first dataset. Same alteration over the network structure also increased the performance from 42.24 percent to 44.90 percent for the first classification task and from 34.90 percent to 37.75 percent for the second classification problem for the second dataset. This means an extra performance boost up to 14.28 percent without changing the artificial neural network structure. However the training time of the artificial neural networks are also increased 4 times.

These results show that it is possible to alter the network structure to increase the performance of learning on the test data regardless of the performance in the training data. While different results may be obtained on different datasets, instead of increasing the depth of the deep artificial neural network in order to increase performance, it is possible to try to enhance the size of the filters or to increase the number of the filters on the deep learning structure.

ACKNOWLEDGMENT

M. Sarıgül would like to thank Asst. Prof. Melis Özyıldırım for her support and assistance during this work.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet Classification With Deep Convolutional Neural Networks,” *Advances In Neural Information Processing Systems,* pp. 1097-1105, 2012.

[2] A. Krizhevsky and G. Hinton, “Convolutional deep belief networks on cifar-10. Unpublished manuscript, 40,” 2010.

[3] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. R. Mohamed, N. Jaitly, and B. Kingsbury, “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups,” *IEEE Signal Processing Magazine,* vol. 29, no. 6, pp. 82-97, 2012.

[4] R. Collobert and J. Weston, “A unified architecture for natural language processing: Deep neural networks with multitask learning,” in *Proc. the 25th international conference on Machine learning,* July 2008, pp. 160-167.

[5] H. Lee, P. Pham, Y. Largman, and A. Y. Ng, “Unsupervised feature learning for audio classification using convolutional deep belief networks,” *Advances in Neural Information Processing Systems,* pp. 1096-1104, 2009.

[6] Y. Sun, X. Wang, and X. Tang, “Deep learning face representation from predicting 10,000 classes,” in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1891-1898.

[7] W. Ouyang and X. Wang, “Joint deep learning for pedestrian detection,” in *Proc. the IEEE International Conference on Computer Vision*, 2013, pp. 2056-2063.

[8] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, “Playing atari with deep reinforcement learning,” 2013.

[9] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning internal representations by error propagation,” *California Univ San Diego La Jolla Inst for Cognitive Science,* 1985.

[10] L. E. Atlas, T. Homma, and R. J. Marks II, "An artificial neural network for spatio-temporal bipolar patterns: Application to phoneme classification," in *Proc. Neural Information Processing Systems (NIPS)*, 1988, p. 31.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998.

[12] S. Behnke, "Hierarchical neural networks for image interpretation," *Springer Science & Business Media*, vol. 2766, 2003.

[13] K. Anantharajah, Z. Y. Ge, C. McCool, S. Denman, C. B. Fookes, P. Corke, D. W. Tjondronegoro, and S. Sridharan, "Local inter-session variability modelling for object classification," 2014.

[14] Ronan, Clément, Koray, and Soumith. [Online]. Available: http://torch.ch/

M. Sarıgül was born on July 7th, 1989. He completed his undergraduate and graduate education in Computer Engineering Department at Cukurova University. He is a Ph.D student in Computer Engineering Department at Cukurova University. He also works as a research assistant at Cukurova University in Adana.

His research areas are artificial intelligence, reinforcement learning and deep learning.

M. Avci completed his undergraduate and graduate education in electrical and electronics engineering and Ph.D. in 2005 in electronics and communication engineering. He is working as an associate professor in the Department of Biomedical Engineering at Çukurova University.

His research areas are artificial intelligence, microelectronics, analog and mixed VLSI.

Computer Information Theory and Applications