

Centroid Neural Network With Directional Spawning of New Weights

Dong-Chul Park

Abstract—This paper proposes Centroid Neural Network with Directional Spawning for efficient data clustering. The proposed algorithm locates each new neuron for cluster center by considering the location trajectory of previously added neurons during training process. Experiments on different data sets are performed in order to evaluate the performance of training error, training speed, and test error. The results show that the proposed algorithm outperforms conventional Centroid Neural Network and Fuzzy c-Means algorithm in terms of clustering errors and training speed.

Keywords—Unsupervised learning algorithm, CNN, FCM, data clustering

I. INTRODUCTION

COMPETITIVE unsupervised learning algorithms in artificial neural networks have been widely used for clustering data in complicated classification tasks [1]-[4]. They have been successfully applied to various problems including data compression, pattern recognition and data classification. One of the most widely used competitive learning algorithms is the k-means clustering algorithm which is simple and fast enough to be performed in real time[1]. However, k-means clustering algorithm shows also limitations in convergence in cases.

CNN is an unsupervised learning algorithm based on the k-means algorithm. It was first proposed in [4] and has some advantages when compared with other conventional unsupervised learning algorithms including SOM (Self Organizing Map) [2] and FCM (Fuzzy C-Means) [3]. The key points of CNN are the winner and loser neuron (or learning and forgetting) concepts in updating cluster weights. This idea makes CNN more stable and yields a competitive classification results[4]-[6].

CNN starts with two clusters and increases this number with certain conditions. A new cluster is only added to the network if the current clusters hold their member data and the membership of each data does not change. In CNN, this new neuron is randomly located near the neuron with most error. The location of a new neuron is somewhat random and is not controlled with the characteristics of given data set. The tendency or trajectory of locations for newly added neurons (cluster centers) can be a valuable information in determining the location of a new neuron. Random positioning of new neuron sometimes causes excessive training time and undesired clustering results.

Dong-Chul Park is with the Department of Electronics, Myongji University, YongIn, Rep. of KOREA (phone: +82-31-330-6756, fax: +82-31-3306977, parkd@mju.ac.kr)

In this paper, we proposed new version of CNN called CNN-DS(Directional Positioning of new neuron) to overcome this issue. In the proposed CNN-DS, the position of a newly added neuron is determined by considering the location trajectory of previously added neurons. By doing so, we can decrease the training time and training error.

This short paper is organized as follows: Section II introduces the conventional CNN and proposed CNN-DS. Feature extraction methods used in experiments are summarized in Section III. Experiments on two data sets are given in Section IV. Section V concludes this paper.

II. ALGORITHMS

A. CNN(Centroid Neural Network)

The CNN is an unsupervised competitive learning algorithm based on the classical k-means clustering algorithm [4]. It finds the centroids of clusters at each presentation of the data vector. The CNN updates its weights only when the status of the output neuron for the presenting data has changed when compared to the status from the previous epoch.

When an input vector x is presented to the network at epoch n , the weight update equations for winner neuron j and loser neuron i in CNN can be summarized as follows:

$$w_j(n+1) = w_j(n) + \frac{1}{N_{j+1}} [x(n) - w_j(n)] \quad (1)$$

$$w_i(n+1) = w_i(n) + \frac{1}{N_{i-1}} [x(n) - w_i(n)] \quad (2)$$

where $w_j(n)$ and $w_i(n)$ represent the weight vectors of the winner neuron and the loser neuron, iteration, respectively.

The CNN has several advantages over conventional algorithms such as SOM, FCM, or k-means algorithm. The CNN requires neither a learning gain parameter nor the total number of iterations. It converges to sub-optimal solutions while SOM may show unstable results depending on parameters. More detailed description on the CNN can be found in [4][5].

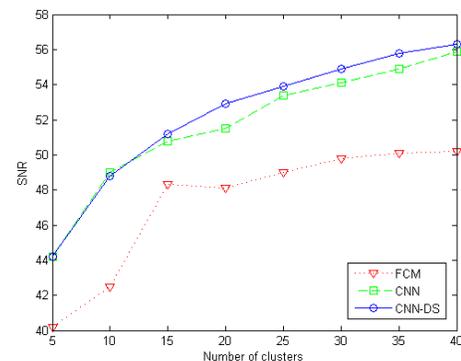


Fig. 1 Comparison of training errors among different algorithms

B. CNN with Directional Spawning

One of the most important feature with CNN is that CNN starts with setting the number of clusters two and increases it one by one until it reaches the predetermined number of clusters. During this process, CNN finds the cluster with most error and then spawns a new weight around this cluster with random positioning. However, this random positioning of the new spawning weight may cause undesired clustering results. When this randomness in positioning of new spawning weights is changed with a certain strategy, the clustering results may be improved.

An idea of the new strategy in positioning of new spawning weights is to control the randomness in positioning of new spawning weights by considering the movement history of the positions of clusters. The momentum term in error back propagation algorithm in multi-layer perceptron type neural network is a similar strategy. When we define a tracking function, \vec{p}_i , to estimate the moving direction over previous run of training as follows:

$$\vec{p}_i = \vec{w}_i - \vec{w}_{i-1} \quad (3)$$

where i denotes the number of clusters and \vec{w}_i denotes the position of the neuron with the most error when the number of clusters is i .

Therefore, the position of new spawning weight is determined by considering the moving direction of previous location of final weights as follows:

$$\vec{w}_{i+1} = \vec{w}_i + \alpha \vec{p}_i \quad (4)$$

where α denotes a predetermined constant.

Once a new spawning weight is determined with the new position, the rest of training procedure is the same as CNN. By doing so, the proposed algorithm still has the advantageous features of CNN with more stable and faster convergence over conventional CNN.

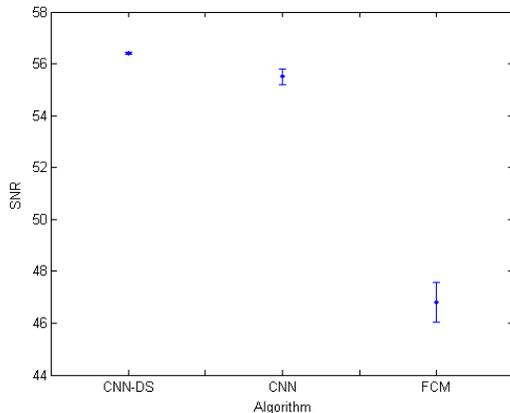


Fig. 2 Comparison of compression errors

III. FEATURE EXTRACTION METHODS

A. LBP(Local Binary Pattern)

The basic Local Binary Pattern(LBP) feature describes local texture information of gray-scale image and shows robust characteristic to illumination changes [6]. The LBP is defined by an ordered set of binary patterns with the difference of pixel intensities between the center pixel and its neighbor pixels.

B. HSV(Hue Saturation Value)

In image processing and photography, a color histogram is a representation of the color distribution in an image. The color histogram can be built for three-dimensional spaces like RGB or HSV. HSV stands for hue, saturation, and value, and is also sometimes called HSB (B for brightness). The hue value describes colors in 2-dimensional circle: locating the red primary at 0°, passing through the green primary at 120° and the blue primary at 240°, and laying out all of the possible mixtures including complement colors, secondary colors, and tertiary colors around in the circle. Saturation and value are represented by percentage from 0% to 100%. The original HSV histogram [7] has a total of 560 (H:360, S:100, V:100) dimensions. However, we use a feature with 140 (H:120, S:10, V:10) dimensions for less computational complexity in data.

C. DCT (Discrete Cosine Transform)

Discrete Cosine Transform (DCT) proposed by Ahmed [8] has been widely used in image compression and restoration such as JPEG, H. 261 and MPEG. DCT coefficient also used in image classification and recognition as an efficient feature representing frequency properties of an image. The DCT and inverse-DCT equations for 2-d case are given by the following:

$$X(u, v) = \frac{2}{N} A_u A_v \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} x(k, l) \cos \frac{(2k+1)u\pi}{2N} \cos \frac{(2l+1)v\pi}{2N} \quad (5)$$

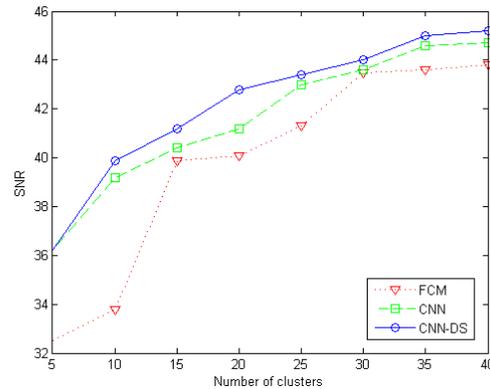


Fig. 3 Comparison of compression errors with test image data

$$x(k, l) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} A_u A_v X(u, v) \cos \frac{(2k+1)u\pi}{2N} \cos \frac{(2l+1)v\pi}{2N} \quad (6)$$

for $k, l = 0, 1, \dots, N - 1$

$$A_u, A_v = \begin{cases} \frac{1}{\sqrt{2}}, & u, v = 0 \\ 1, & u, v = 1, 2, \dots, N - 1 \end{cases} \quad (7)$$

In experiments, input images are divided into 8×8 pixel non-overlapping blocks and DCT is applied to each block. DCT coefficients are arranged by the zigzag scanning from the upper left corner to the lower right corner. Since more important information of an image is stored in lower band of DCT coefficients, only 32-dimensional low-frequency coefficients are used. In order to obtain feature vectors on a image data

using this DCT coefficients, the mean and variance of each component of DCT coefficients for whole blocks in the image data are calculated and finally the 64-dimensional DCT feature vectors are obtained.

IV. EXPERIMENTS AND RESULTS

In order to evaluate the proposed CNN-DS, experiments on image compression and data classification are performed.

A. Image compression experiments

Image compression experiments are performed on Lena image (256×256). The image is divided into a set of 4 dimensional vectors with a 2×2 block size and it results in 16, 384 data points. In order to evaluate the image compression performance of the proposed CNN-DS, the performance of CNN-DS is compared with those of conventional clustering algorithms such as FCM(Fuzzy c-Means) and CNN in terms of SNR for both of training and test results.

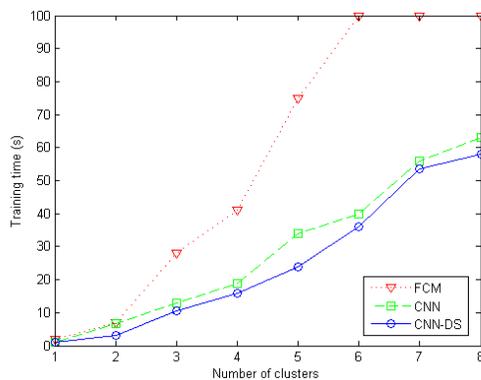


Fig. 4 Comparison of training speed

Fig. 1 compares the training errors for different algorithms in terms of SNR, the average and standard deviation. The results on training errors in Fig 1 show that the proposed CNN-DS gives the highest average SNR over the other two algorithms with the smallest standard deviation. Since the proposed CNN-DS begins the location of new cluster center with an optimal value by considering previous cluster centers instead of a random value of CNN, the training error of CNN-DS has much smaller value of standard deviation than that of CNN.

A comparison of compression errors for different algorithms in terms of SNR with different numbers of clusters is summarized in Fig.2. The results show that the proposed CNN-DS improves the compression error of conventional CNN and FCM for most cases of cluster numbers. Fig.3 shows a comparison of compression errors with test image data. Fig. 4 shows the training speeds with different numbers of clusters for different algorithms.

B. Data classification experiments

Experiments on UCI are also performed in order to evaluate the performance of the proposed CNN-DS. Abalone, Image segmentation, Iris, Meta, Multifeature, and Wine data sets from the UCI database are used for experiments [9]. Features are obtained by HSV (200 dimensions), Uniform LBP (59 dimensions) and DCT (64 dimensions). All three features are concatenated and results in one feature vector ($200 + 59 + 64 =$

323 dimensions) for each data.

Each of 6 data sets, Abalone, Image segmentation, Iris, Meta, Multifeature, and Wine, is partitioned into two subsets: 90% randomly selected samples for the training data set and the remaining 10% samples for the test data set. Experiments are performed with 10 different combinations of training and test data sets. The classification accuracies reported in Table I are the averages and standard deviations of the obtained accuracies from the 10 different cases of the training and test data sets.

Results summarized in Table I show that the proposed

TABLE I
RESULTS OF CLASSIFICATION ACCURACY ON UCI DATABASE

data	CNN	CNN-DS
Abalone	47.83%±9.6%	48.08%±8.1%
I.S	54.33%±11.5%	56.92%±11.5%
Iris	66.67±0%	91.11%±0%
Meta	86.37%±4.5%	88.18%±2.7%
M.F	62.5%±2.5%	62.7%±1.3%
Wine	54%±21%	70.83±12.5%
average	61.95%±8.2%	69.64%±6%

CNN-DS outperforms the conventional CNN in all data sets, especially in Iris and Wine data sets, in terms of average and standard deviation of classification accuracy. This result implies that the proposed CNN-DS improves the randomness in classification accuracy resulting from the randomness in initial position of spawning weights.

V. CONCLUSIONS

In this paper, CNN-DS(Directional Positioning of new neuron) is proposed to overcome the randomness in clustering performance of conventional CNN resulting from the randomness in initial position of spawning weights. In the proposed CNN-DS, the position of a newly added neuron is determined by considering the location trajectory of previously added neurons.

In order to evaluate the performance of the proposed CNN-DS, experiments on an image compression and one data classification task are performed. Experiments on Lena image compression shows that the proposed CNN-DS improves the compression error of conventional CNN and FCM for most cases of cluster numbers in terms of SNR on compressed images. In the subsequent experiments, data classification task on UCI database is considered. 6 data sets, Abalone, Image segmentation, Iris, Meta, Multifeature, and Wine, are trained and tested using CNN and the proposed CNN-DS. Results show that the proposed CNN-DS outperforms the conventional CNN by 7.69% in terms of average classification accuracy. More importantly, the average standard deviations for CNN and CNN-DS are reported 8.2% and 6.0%, respectively. This result shows that the proposed CNN-DS improves the randomness in classification accuracy resulting from the randomness in initial position of spawning weights.

ACKNOWLEDGMENT

This work was supported by the IT R&D program of The MKE/KEIT (10040191, The development of Automotive Synchronous Ethernet combined IVN/OVN and Safety control

system for 1Gbps class). The author would like to thank Mr. J. Han and members of ICRL for their help in preparing this manuscript.

REFERENCES

- [1] J.B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations," in *Proc. of 5th Berkeley Symposium on Mathematical Statistics and Probability I*, University of California Press, pp. 281-297, 1967.
- [2] T. Kohonen, "The self-organizing map," *Proceedings of IEEE*, vol. 78, no.9, pp. 1464-1480, 1990.
- [3] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.
- [4] Dong-Chul Park, "Centroid Neural Network for Unsupervised Competitive Learning," *IEEE Trans. on Neural Networks*, vol. 11, no. 2, pp.520-528, May, 2000.
- [5] Dong-Chul Park, Oh-Hyun Kwon, and Jio Chung, "Centroid neural network with a divergence measure for gpdf data clustering," *IEEE Trans. on Neural Networks*, vol. 19, no. 6, pp. 948-957, June 2008.
- [6] T. Ahonen et al, "Face Description with Local Binary Patterns," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no.12, pp. 2037-2041, 2006.
- [7] C.L. Novak and S.A. Shafer, "Anatomy of a color histogram," *IEEE Computer Society. Conf.*, pp. 299-605, 1992.
- [8] N. Anmed et al, "Discrete Cosine Transform," *IEEE Trans. on Computer*, vol. 23, no.1, pp. 90-93, 1974.
- [9] K. Bache, M. Lichman, M. UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science. 2013
- [10] http://www.vision.caltech.edu/Image_Datasets/Caltech101

Dong-Chul Park (M'90-SM'99) received the B.S. degree in electronics engineering from Sogang University, Seoul, Korea, in 1980, the M.S. degree in electrical and electronics engineering from the Korea Advanced Institute of Science and Technology, Seoul, Korea, in 1982, and the Ph.D. degree in electrical engineering, with a dissertation on system identifications using artificial neural networks, from the University of Washington (UW), Seattle, in 1990. From 1990 to 1994, he was with the Department of Electrical and Computer Engineering, Florida International University, The State University of Florida, Miami. Since 199-4, he has been with the Department of Electronics Engineering, MyongJi University, Korea, where he is a Professor. From 2000 to 2001, he was a Visiting Professor at UW. He is a pioneer in the area of electrical load forecasting using artificial neural networks. He has published more than 130 papers, including 40 archival journals in the area of neural network algorithms and their applications to various engineering problems including financial engineering, image compression, speech recognition, time-series prediction, and pattern recognition. Dr. Park was a member of the Editorial Board for the IEEE TRANSACTIONS ON NEURAL NETWORKS from 2000 to 2002.