International Journal of Engineering Research and Development
e-ISSN: 2278-067X, **p**-ISSN: 2278-800X, www.ijerd.com
Volume 7, Issue 12 (July2013), PP. 53-58

Optimal Palette Generation Using Trained Neural Network Approach (TNN)

Japneet Kaur[1], Ramanpreet Kaur[2]
[1] M-Tech Student, CEC, Landran, Mohali (Punjab)
[2] Assistant Professor, CEC, Landran, Mohali (Punjab)

Abstract:- Color is the most important properties for object discrimination. It is an object component that adds a new dimension to the machine vision. The color images on low devices are displayed having limited number of colors called as color palette reduction to cut down the storage requirements. In this paper we investigate the problem of averaging aggregation functions to select the appropriate function using neural network to reduce the color of an image. The neural network represents the large set of input features with a smaller set of rules to predict the exact penalty of the aggregation function to have good approximation. Then train the neural network with the image samples so that it creates the set of rules to predict the color values using aggregation functions as the features and it comes to an end with some experimental results and conclusions.

Keywords:- color reduction, color palette, neural networks, neural network learning, network training

I. INTRODUCTION

Color is one of the most important properties for object detection and also one of the most widely used information for image processing researchers. Several techniques have been proposed in the literature for color reduction, as reduction of colors is an important factor for segmentation, compression, presentation and transmission. Neural and fuzzy clustering algorithms are employed for color reduction. The techniques in this category attempt to find the optimal palette using vector classifiers like Growing Neural Gas(GNG)[14], FOSART[15-18], Fuzzy-ART[19,20] and FCM[21] and K-means[12]. Most of the above clustering techniques lead to spherical clusters, which is an major disadvantage in cases where we have dominant colors. There is a group of techniques that repeatedly divide the 3D color space into disjoint regions[13]. The methods of Octree[8,9],median-cut[10] and variance-based algorithms[11] are some of the splitting algorithms used for image color reduction.

Adaptive color reduction (ACR) technique achieves color reduction using a tree clustering procedure. In each node of the tree, a self-organized neural network classifier (NNC) is used which is fed by image color values and additional local spatial features. The output neurons of the NNC define the color classes for each node. The final image not only has the dominant image colors, but its texture also approaches the image local characteristics used.[1] The another neural technique uses KSOM and Neural Gas Network and typical GSONN, Growing Neural Gas for color reduction[2]. A Bi-criteria global optimization approach for the color display of hyper-spectral images uses the fusion model, useful in any multidimensional imagery color display which preserves the spectral distance criterion, and contrast, guaranteeing that colors are well distinguished or concretely allowing the good separability of each observed existing material in the final visualized color image[3]. A multi-scale edge detection algorithm which took soft threshold method to implement detail enhancement and noise reduction of the true color image[4]. The method called MOCK, uses multi-objective evolutionary algorithm as the mean of clustering employed to classify the colors in the input image into representative colors[6].

In this paper we recall the problem of image reduction through averaging aggregation functions[5,7] using neural network and then reconstruct the image. When reconstruct the image, the error with respect to the original image may be determined by the reduction method employed.

The structure of this paper is as follows: In section II, we have discussed the problem formulation and research methodology to be adopted for achieving the objectives. In Section III the architecture of neural network and the neural network toolbox has been discussed. In Section IV, the neural network approach is explained and in Section V, training of network is done and desired Trained Neural Network (TNN) is obtained. The experimental study has been presented next section and results are given in next Section VII. The paper is concluded with the conclusions in section VIII .

II. PROBLEM FORMULATION AND RESEARCH METHODOLOGY

Our problem is to minimize the penalties to generate the appropriate aggregation function values to reduce the colors of image. This problem arises in image processing when several values given in RGB, HSL, or another coding scheme needed to be combined. The minimization of penalties provides, with the probabilities to select the aggregated color values and objective, is to increase the probability of selection of aggregation function with better PSNR.

The main objective of research technology is to optimize the values of PSNR to have maximum value and the optimization of mean square error to have its minimum value. The application of penalty functions features for training the neural network in which we take aggregated functions as the features to train the neural network and concluding various parameters like PSNR and MSE in experimental analysis for computing percentage of improvement.

III. NEURAL NETWORK

A Neural Network is the information processing paradigm, which is inspired by the biological nervous systems, such as the brain which processes the information[22]. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements called as neurons, working in union to solve specific problems[22].

4.1 Why use Neural Networks?

Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyze[24]. Once it is trained it creates its own set of rules to predict the output for any arbitrary input through its learning capability as a simple neuron learns from teaching as shown in Fig.1. Neural networks do not perform miracles. But if used sensibly they can produce some amazing results.

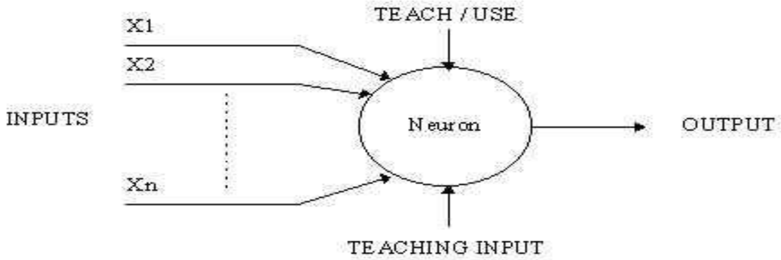

Fig.1: A simple neuron which learns from teaching

4.2 Neural Network Toolbox

Neural Network Toolbox supports a variety of supervised and unsupervised network architectures. With the toolbox's modular approach to create networks, we can develop and custom the network architectures for our specific problem. We can view the network architecture including all inputs, layers, outputs, and interconnections.

4.2.1 Supervised Networks

Supervised neural networks are trained to produce desired outputs in response to sample inputs, making them particularly well-suited to modeling and controlling dynamic systems, classifying noisy data, and predicting future events[23-25].

4.2.2 Unsupervised networks

Unsupervised neural networks are trained by letting the network continually adjust itself to new inputs. Neural Network Toolbox includes two types of self-organizing, unsupervised networks: competitive layers and self-organizing maps[23-25].

The neural network toolbox consists of following tools as: Fitting Tool, Pattern Recognition tool, Clustering tool and Time Series Tool as shown in Fig.2

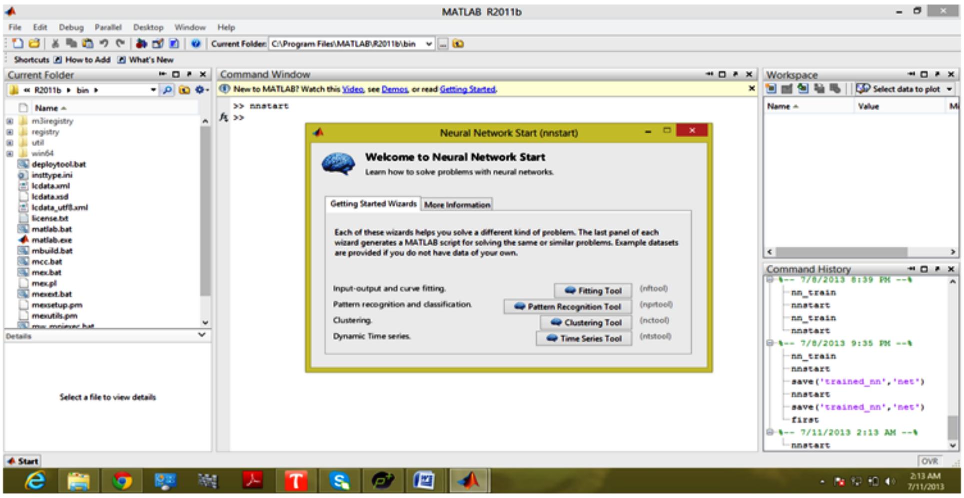

Fig.2 Neural Network Toolbox

IV. NEURAL NETWORK APPROACH FOR COLOR REDUCTION

Consider any arbitrary image of size 321*480 and load into MATLAB using image acquisition toolbox. Then create the samples from that image as inputs which can be related to particular outputs specified. The features are selected as aggregation functions which are appropriate. A feature vector table is created which relates the inputs to the outputs through the samples taken from the arbitrary image. Then we train the neural network with our feature vector table so that it predicts the most efficient pixel value with the minimum penalty. The entire work is done in neural network toolbox. The neural network check for the penalties from all the probabilities created from its own set of rules and fix up the pixel with the appropriate value.

Table I: Feature vector table to train the neural network

Features	Samples(inputs) S1----Smn				outputs
X1	S1	S2	S3	S4	Y1
X2	S11	S12	S13	S14	Y2
X3	S21	S22	S23	S24	Y3

V. TRAINING OF NEURAL NETWORK

Various tools of neural network are available for the different purposes. The fitting tool of neural network toolbox is used to train the neural network. The neural network can be trained for an arbitrary image of any size so that it learns how the color values be adjusted for predicting the color of any other input image. Each neural network possesses the knowledge, which is contained in the values of the connections weights. Modifying the knowledge stored in the network as a function of experience, implies a learning rule for changing the values of the weights to generate the set of rules.

6.1 Steps to train the neural network

I. Load the image of any size in neural network using MATLAB image acquisition tool.
II. Neural network is then adjusted that it takes the samples from that loaded image and relate those samples with our inputs and outputs specified by us.
III. Number of samples can be adjustable. The work has been presented using number of samples of that particular picture =1000.
IV. Number of hidden neurons= "variable"

Then train the neural network with the toolbox and best training is supposed to be done when the best validated performance and minimum MSE is obtained.

Fig.3 Picture used to create samples from it and 1000 samples have been taken from this picture.

VI. EXPERIMENTAL STUDY

A. SIMULATION AND TESTING:

- **Number of samples=1000**
- **Number of hidden neurons=10**

I. FIRST ATTEMPT

Table II: Data obtained from trained NN

No. of times neural network is trained	MSE	No. of epochs	Validation performance
1	17.9018	14	22.0867 at epoch 8
2	19.8190	14	32.5389 at epoch 8
3	19.6779	19	24.8081 at epoch 13
4	17.1395	19	23.3407 at epoch 13
5	15.0303	16	35.6246 at epoch 10
6	14.3710	55	32.9545 at epoch 45

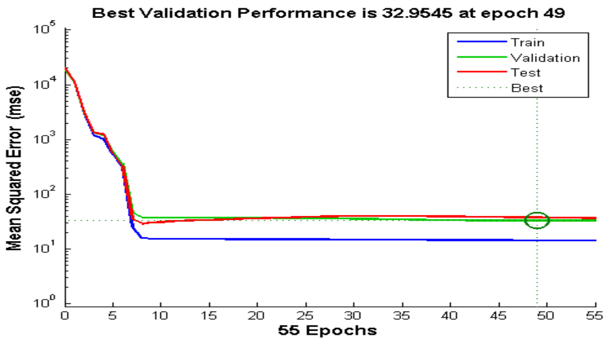

Fig.4 Performance of best trained NN in First Attempt.

- **Number of samples =1000**
- **Number of hidden neurons=8**

II. SECOND ATTEMPT

Table II. Data obtained from trained NN

No. of times neural network is trained	MSE	No. of epochs	Validation performance
1	18.7806	14	24.4557 at epoch 8
2	18.2245	25	27.4389 at epoch 8
3	23.1209	14	33.5150 at epoch 13
4	19.0735	21	17.0525 at epoch 13
5	15.7801	19	31.4611 at epoch 15
6	25.0832	55	36.2045 at epoch 45

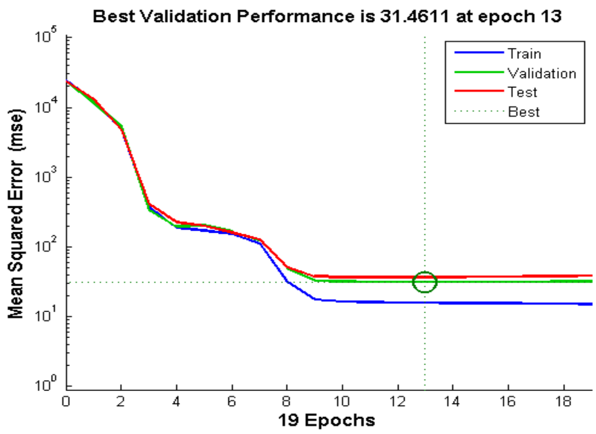

Fig.5 Performance of best trained NN in Second Attempt.

B. REDUCTION OF COLORS OF IMAGE USING TNN APPROACH

The results of best trained neural network are stored and then apply the rule set of saved neural network to other image.

Fig.6 Original image on which TNN is tested

Fig.7 Color reduction using TNN approach

VII. RESULTS

The best training is obtained in first attempt at 55 epochs and having the minimum MSE of 14.3710 and In second attempt MSE is 15.7801 at 19 epochs. The best TNN (Trained Neural Network) results are saved and applied to any other image and we conclude that the colors of image are reduced to visible extent.

VIII. CONCLUSION AND FUTURE WORK

The Trained Neural Network (TNN) belongs to computational intelligence paradigm as a smart tool which can be used for image processing and object detection. The TNN select the pixel values from the set of rules and set the values of image having minimum penalty value decided by its learning capability. We conclude that the image obtained after TNN approach has lesser number of colors as compared to original image. The computational cost of this method is small and is important because it results smaller number of colors efficient for transmission, segmentation and presentation of an image.

The TNN approach can be applied to color reduction of images and enhancing the performance in terms of PSNR and MSE is our next research. Noise factors would be considered in our further work and emphasis on local characteristics used so that image structure approximates the original image.

ACKNOWLEDGMENT

The authors would like to thank the editors and anonymous reviewers for their valuable comments and suggestions on the manuscript.

REFERENCES

[1] Nikos Papamarkos, Antonis E. Atsalakis and Charalampos P. Strouthopoulos “Adaptive Color Reduction” 2002 IEEE

[2] Guojian Cheng, Jinquan Yang, Kuisheng Wang, Xiaoxiao Wang “Image Color Reduction based on Self-organizing Maps and Growing Self-Organizing Neural Networks” 2006 IEEE

[3] Max Mignotte “A Bicriteria-Optimization-Approach-Based Dimensionality-Reduction Model for the Color Display of Hyperspectral Images” 2012 IEEE

[4] Feng Xiao , Mingquan Zhou , Guohua Geng “Detail Enhancement and Noise Reduction with True Color Image Edge Detection” 2011 IEEE

[5] Gleb Beliakov, Humbeto Bustince and Daniel Paternain “Image Reduction Using Means on Discrete Product Lattices”2011 IEEE

[6] Motonari Sadohira, Akio Saito, Hernan Aguirre and Kiyoshi Tanaka "A Color Palette Reduction Method Using Multi objective Evolutionary Clustering Algorithm" 2011 IEEE
[7] Daniel Paternain, Aranzazu Jurio, Javier Fernandez and Humberto Bustince, Gleb Beliakov "Color Image Reduction by Minimizing Penalty Functions" 2012 IEEE
[8] Gervautz, M., Purgathofer, W.: A simple method for color quantization: Octree quantization. Graphics Gems, A. S. Glassner, Ed. Academic New York (1990) 287–293
[9] Ashdown, I.: Octree color quantization in Radiosity. A Programmer's Perspective. Wiley New York (1994)
[10] Heckbert, P.: Color image quantization for frame buffer display. Comput. Graph., vol. 16 (1982) 297–307
[11] Wan, S. J., Prusinkiewicz, P., Wong, S. K. M.: Variance based color image quantization for frame buffer display. Color Res. Applicat., vol. 15, no. 1 (1990) 52–58
[12] O. Verevka, "The local K-means algorithm for color image quantization," M.Sc. dissertation, Univ. Alberta, Edmonton, AB, Canada, 1995
[13] Scheunders, P.: A comparison of clustering algorithms applied to color image quantization. Pattern Recognit. Lett., vol. 18. (1997) 1379–1384
[14] Fritzke, B.: A growing neural gas network learns topologies. Tesauro, G., Touretzky, D.S., Leen, T.K. (Eds.), Advances in Neural Information Processing Systems, vol. 7. MIT Press, Cambridge, MA (1995) 625–632
[15] Baraldi, A., Blonda, P.: A survey of fuzzy clustering algorithms for pattern recognition—part I. IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics 29 (6) (1999) 778–785
[16] Baraldi, A., Blonda, P.: A survey of fuzzy clustering algorithms for pattern recognition-part II. IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics 29 (6) (1999) 786–801
[17] Baraldi, A., Parmiggiani, F.: Novel neural network model combining radial basis function, competitive Hebbian learning rule, and fuzzy simplified adaptive resonance theory. Proceedings of the SPIE's Optical Science, Engineering and Instrumentation 1997: Applications of Fuzzy Logic Technology IV, vol 3165. San Diego CA (1997) 98–112
[18] Baraldi, A., Parmiggiani, F.: A fuzzy neural network model capable of generating/ removing neurons and synaptic links dynamically. Blonda, P. Castellano, M. Petrosino, A. (Eds.) Proceedings of the WILF 1997-II Italian Workshop on Fuzzy Logic. World Scientific Singapore (1998) 247–259
[19] Carpenter, G., Grossberg, S., Rosen, D. B.: Fuzzy ART: fast stable learning and categorization of analog patterns by an adaptive resonance system. Neural Networks 4. (1991) 759–771
[20] . Carpenter, G., Grossberg, S., Maukuzon, N., Reynolds, J., Rosen, D.B.: Fuzzy ARTMAP: a neural network architecture for incremental supervised learning of analog multidimensional maps. IEEE Transactions on Neural Networks 3 (5). (1992) 698–713
[21] Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press. New York (1981)
[22] An introduction to neural computing. Aleksander, I. and Morton, H. 2nd edition
[23] Neural Networks, Eric Davalo and Patrick Naim.
[24] DARPA Neural Network Study (October, 1987-February, 1989). MIT Lincoln Lab.
[25] An Introduction to Computing with Neural Nets (Richard P. Lipmann, IEEE ASSP Magazine, April 1987)