International Journal of Advanced Science and Technology
Vol.70 (2014), pp.1-10
http://dx.doi.org/10.14257/ijast.2014.70.01

Study of FIR Filter Designing By Using Self Organizing Map Neural Network

Navneet Gupta and Ravindra Pratap Narwaria

Electronics Engineering Department
Madhav Institute of Technology &
Science, Gwalior
navneetgupta19902000@gmail.com, narwaria10@yahoo.co.in

Abstract

In this paper, designing of a low pass FIR filter is done by artificial neural network. For this kind of application a different type of model is used in ANN. In this research work, self-organizing map (SOM) algorithm is used to train the Neural Network. The SOM algorithm is based on unsupervised learning technique. We also compare the result of neural network with the result of normal window method. The accuracy of SOM neural network is about 98%.

Keywords: *Neural Network, SOM, Accuracy*

1. Introduction

Digital filter designing techniques are widely used in different areas. The input and output signals in the digital filters are digital or discrete. Mainly, digital filters are linear time invariant (LTI) systems which are characterized by unit sample response. These filters are highly flexible and it has minimum interference noise and other effects. Digital filters are simple in storage, maintenance. The FIR filters have greater flexibility to control the shape of their magnitude and phase response over the IIR filters [1-2]. We are using the MATLAB software due to reason that the MATLAB is a high-performance language for technical computing integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation [3]. In this research work, DSP and neural networks were combined to produce an excellent algorithm for digital filter design. Self-organizing map (SOM) algorithm one of the most popular neural network models. SOM belongs to the category of competitive learning networks.

2. Designing FIR Filter Using Fda Toolbox in MATLAB

In MATLAB there is a toolbox for the designing of digital filter. That is filter design and analysis (fda) toolbox, by the using of this toolbox we can easily design any type of filer. For designing a low pass fir filter there are different methods like equiripple, least square, window techniques etc. The windowing method requires minimum amount of computational effort; so window method is simple to implement. In this paper, we use three types of windows as rectangular, hamming and Kaiser Window. Then compare the results of various training algorithm of neural network with the results from fda tool. The order of filter for all windows is set to value 6. Here we use normalized frequency (0 to 1) as input and filter coefficients h(n) as output. Starting input from 0.01 to 0.99, design the filter and export filter coefficients h(n) to MATLAB workspace. These are dataset of input and output.

3. Artificial Neural Network

ANN is an information processing network. In this network, the element neurons, process the information [4]. The signals are transmitted by the connection links. These links have an associated weight, which is multiplied by the incoming signal for any neural network. An important aspect of an ANN model is that it requires guidance in learning or not. Based on the way of learning, all artificial neural networks can be divided into two learning areas supervised and unsupervised.

3.1. Self-Organizing Map Neural Network

The Self-Organizing Map was developed by the professor Teuvo Kohonen. The SOM is a well-known unsupervised neural learning algorithm. The SOM learns from examples a mapping from a high-dimensional continuous input space X onto a low-dimensional discrete space (lattice) A of q neurons which are arranged in fixed topological forms, *e.g.*, as a rectangularThe main objective of an SOM is to convert an input signal of any dimension into a one or two dimensional discrete map, and perform this conversion in a topologically way. Therefore set up SOM by placing neurons at the nodes of a two dimensional lattice. Weight adjustment is performed until a steady state of global ordering of the weight vectors has been achieved. In this case, we say that the map has converged. The resulting map also preserves the topology of the input samples in the sense that adjacent patterns are mapped into adjacent regions on the map. Due to this topology-preserving property, the SOM is able to cluster input information and spatial relationships of the data on the map. Despite its simplicity, the SOM algorithm has been applied to a variety of complex problems and has become one of the most important ANN architectures.s. We can view this as a non-linear generalization of principal component analysis (PCA).

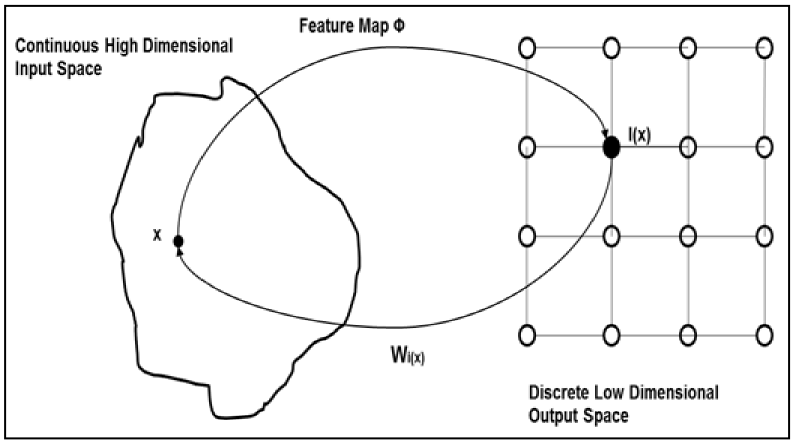

Feature Map Φ
Continuous High Dimensional Input Space
I(x)
x
Wi(x)
Discrete Low Dimensional Output Space

Figure 1. Organizing of Map

4. Formulation of Problem

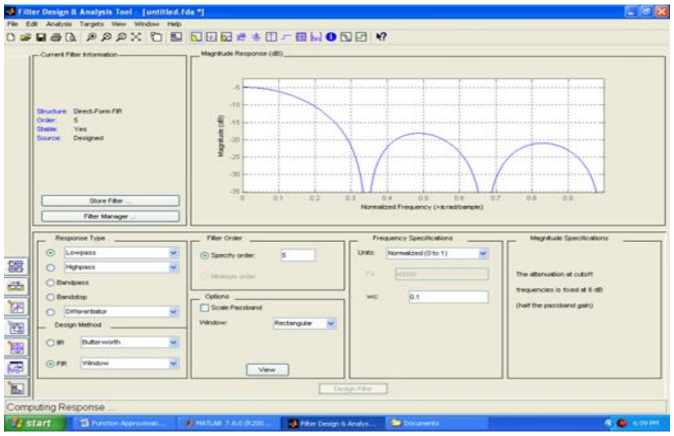
The aim of this paper is to estimate the cut off frequency of any given coefficients of fir filter and increase the accuracy of result from previous work. Accuracy is determined by comparing the result from fdatool and nntool. For training purpose we use input output dataset derived from fdatool, and for testing purpose we use a test input and simulate it using nntool.

5. Methodology

In this methodology we use kaiser window technique for filter designing and SOM neural network for training purpose.

5.1. Step 1

Low pass fir filter designed by fda tool. The order of filter is 10. We use cut of frequency (fc) range from 0 to 1. Then set the value of fc = 0.01 and design the filter. Repeat the same process for value of fc from 0.01 to 1.0. So this gives the 99 dataset of input and output data. Out of these 99 dataset we use 50 for training and 49 for testing. Here input is h(n) and output is fc. The screenshot of first step is given below in Figure 2.



Structure: Direct-Form FIR
Order: 5
Stable: Yes
Source: Designed

Specify order: 5
Scale Passband
Window: Rectangular
Units: Normalized (0 to 1)
wc: 0.1

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Figure 2. Filter Designing by *Fdatool*

5.2. Step 2

Export the coefficients on the MATLAB workspace. Design the neural network model by nntool. Training is done by SOM algorithm. After training, simulate the network by testing input.

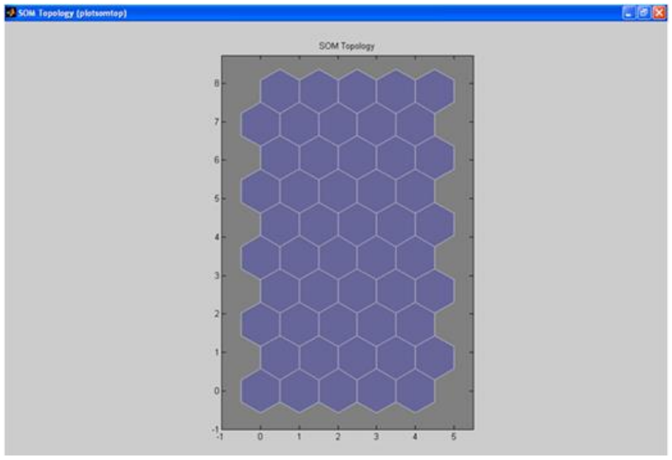


Figure 3. SOM Topology (Plotsomtop)

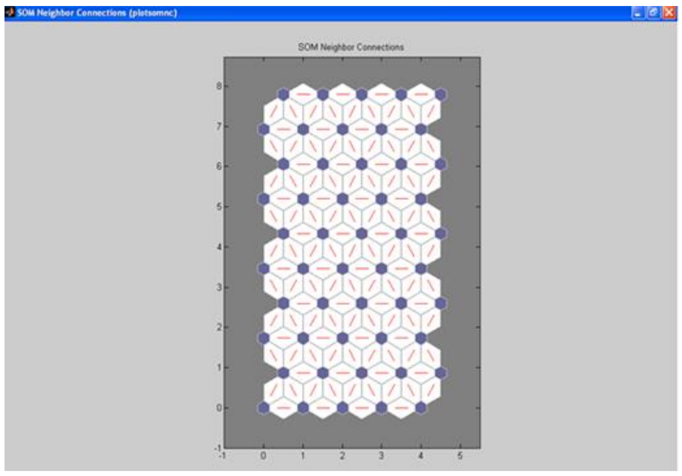


Figure 4. SOM Neighbor Connections (plotsomnc)

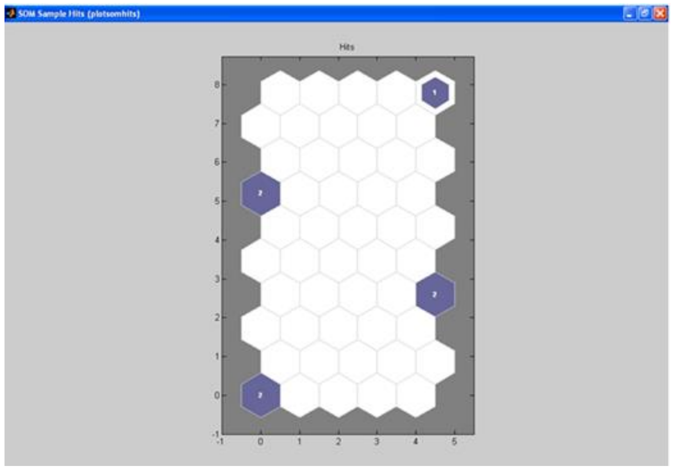


Figure 5. SOM Sample Hits (Plotsomhits)

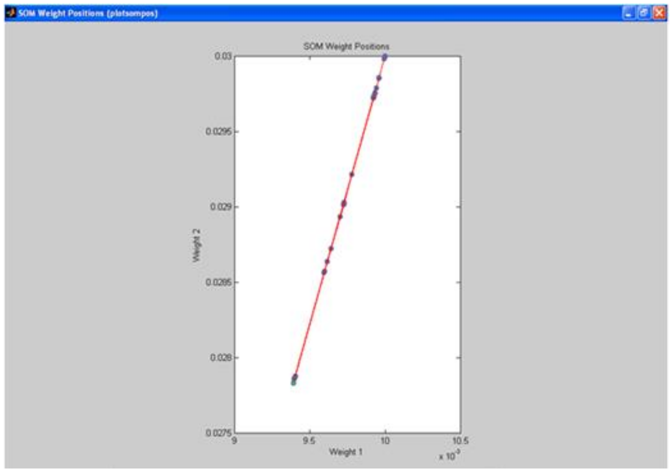


Figure 6. SOM Weight Positions (Plotsompos)

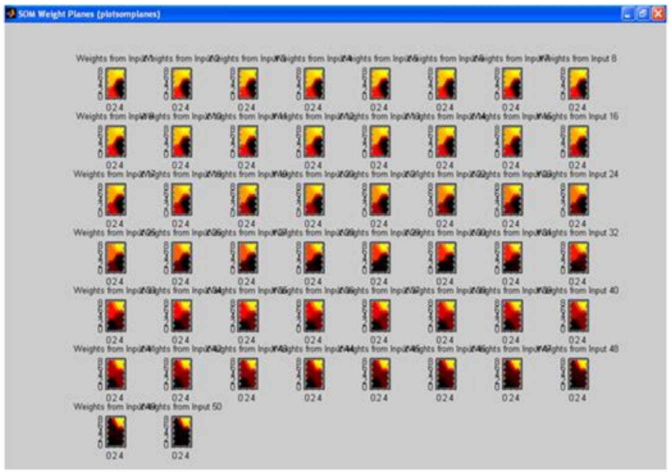


Figure 7. SOM Weight Planes (Plotsomplanes)

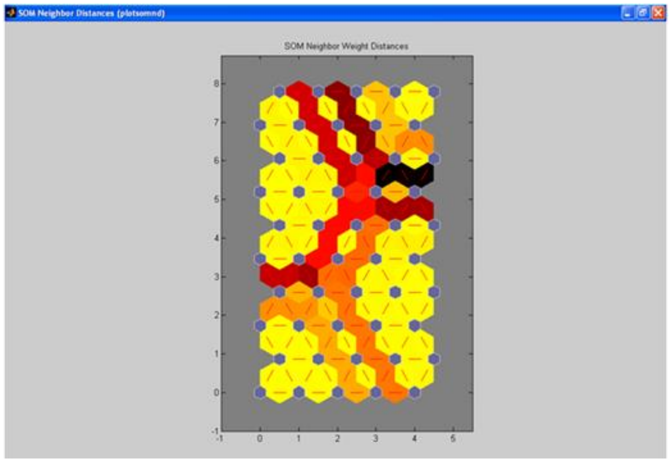


Figure 8. SOM Neighbor Distances (Plotsomnd)

6. Results and Discussion

According to the Table 1, there are 50 samples of input output dataset. These input output dataset are derived from fdatool. In the neural network, this input output dataset using as the training dataset. According to the Table 2, there are also 50 samples of input output dataset. The input of this dataset is working as the test input for the neural network and the output of this dataset is used to compare the result from nntool simulation result. According to the Table 3, there are also 50 samples of input output dataset. This dataset is derived from nntool. Where input is same as test input in Table 2 and the output is simulation result of nntool. So after comparison of output from Table 2 and Table 3, it shows almost equal. So the result from nntool is about 98% accurate.

Table 1. Result from *fdatool* (for Train Input)

h(n) (Train Input)							Cut off frequency (Target)
h(0)	h(1)	h(2)	h(3)	h(4)	h(5)	h(6)	Fc
0.00938	0.00972	0.00993	0.01	0.00993	0.00972	0.00938	0.01
0.02783	0.02902	0.02975	0.03	0.02975	0.02902	0.02783	0.03
0.04529	0.04786	0.04945	0.05	0.04945	0.04786	0.04529	0.05
0.06114	0.06595	0.06897	0.07	0.06897	0.06595	0.06114	0.07
0.07483	0.08299	0.08820	0.09	0.08820	0.08299	0.07483	0.09
0.08587	0.09873	0.10709	0.11	0.10709	0.09873	0.08587	0.11
0.09387	0.11291	0.12556	0.13	0.12556	0.11291	0.09387	0.13
0.09854	0.12531	0.14353	0.15	0.14353	0.12531	0.09854	0.15
0.09972	0.13573	0.16094	0.17	0.16094	0.13573	0.09972	0.17
0.09736	0.14401	0.17771	0.19	0.17771	0.14401	0.09736	0.19
0.09156	0.15002	0.19378	0.21	0.19378	0.15002	0.09156	0.21
0.08251	0.15367	0.20908	0.23	0.20908	0.15367	0.08251	0.23
0.07054	0.15489	0.22356	0.25	0.22356	0.15489	0.07054	0.25
0.05607	0.15367	0.23716	0.27	0.23716	0.15367	0.05607	0.27
0.03962	0.15002	0.24982	0.29	0.24982	0.15002	0.03962	0.29
0.02176	0.14401	0.26149	0.31	0.26149	0.14401	0.02176	0.31
0.00313	0.13573	0.27214	0.33	0.27214	0.13573	0.00313	0.33
-0.01560	0.12531	0.28170	0.35	0.28170	0.12531	-0.01560	0.35
-0.03379	0.11291	0.29016	0.37	0.29016	0.11291	-0.03379	0.37
-0.05078	0.09873	0.29747	0.39	0.29747	0.09873	-0.05078	0.39
-0.06597	0.08299	0.30361	0.41	0.30361	0.08299	-0.06597	0.41
-0.07883	0.06595	0.30855	0.43	0.30855	0.06595	-0.07883	0.43
-0.08889	0.04786	0.31227	0.45	0.31227	0.04786	-0.08889	0.45
-0.09580	0.02902	0.31476	0.47	0.31476	0.02902	-0.09580	0.47
-0.09932	0.00972	0.31601	0.49	0.31601	0.00972	-0.09932	0.49
-0.09932	-0.00972	0.31601	0.51	0.31601	-0.00972	-0.09932	0.51
-0.09580	-0.02902	0.31476	0.53	0.31476	-0.02902	-0.09580	0.53
-0.08889	-0.15489	0.31227	0.55	0.31227	-0.15489	-0.08889	0.55
-0.07883	-0.15459	0.30855	0.57	0.30855	-0.15459	-0.07883	0.57
-0.06597	-0.15367	0.30361	0.59	0.30361	-0.15367	-0.06597	0.59
-0.05078	-0.15215	0.29747	0.61	0.29747	-0.15215	-0.05078	0.61
-0.03379	-0.15002	0.29016	0.63	0.29016	-0.15002	-0.03379	0.63
-0.01560	-0.14731	0.28170	0.65	0.28170	-0.14731	-0.01560	0.65
0.00313	-0.14401	0.27214	0.67	0.27214	-0.14401	0.00313	0.67
0.01250	-0.14015	0.26149	0.69	0.26149	-0.14015	0.01250	0.69
0.02176	-0.13573	0.24982	0.71	0.24982	-0.13573	0.02176	0.71
0.03083	-0.13078	0.23716	0.73	0.23716	-0.13078	0.03083	0.73
0.03962	-0.12531	0.22356	0.75	0.22356	-0.12531	0.03962	0.75
0.04806	-0.11934	0.20908	0.77	0.20908	-0.11934	0.04806	0.77
0.05607	-0.11291	0.19378	0.79	0.19378	-0.11291	0.05607	0.79
0.06359	-0.10603	0.17771	0.81	0.17771	-0.10603	0.06359	0.81
0.07054	-0.09873	0.16094	0.83	0.16094	-0.09873	0.07054	0.83
0.07687	-0.09104	0.14353	0.85	0.14353	-0.09104	0.07687	0.85
0.08251	-0.08299	0.12556	0.87	0.12556	-0.08299	0.08251	0.87
0.08742	-0.07462	0.10709	0.89	0.10709	-0.07462	0.08742	0.89
0.09156	-0.06595	0.08820	0.91	0.08820	-0.06595	0.09156	0.91
0.09488	-0.05702	0.06897	0.93	0.06897	-0.05702	0.09488	0.93
0.09736	-0.04786	0.04945	0.95	0.04945	-0.04786	0.09736	0.95
0.09898	-0.02902	0.02975	0.97	0.02975	-0.02902	0.09898	0.97
0.09972	-0.00972	0.00993	0.99	0.00993	-0.00972	0.09972	0.99

Table 2. Result from fdatool (for Test Input)

h(n) (Test Input)							Cut off frequency (Output)
h(0)	h(1)	h(2)	h(3)	h(4)	h(5)	h(6)	Fc
0.01869	0.01941	0.01985	0.02	0.01985	0.01941	0.01869	0.02
0.03672	0.03852	0.03962	0.04	0.03962	0.03852	0.03672	0.04
0.05345	0.05702	0.05924	0.06	0.05924	0.05702	0.05345	0.06
0.06829	0.07462	0.07862	0.08	0.07862	0.07462	0.06829	0.08
0.08071	0.09104	0.09770	0.10	0.09770	0.09104	0.08071	0.10
0.09027	0.10603	0.11638	0.12	0.11638	0.10603	0.09027	0.12
0.09663	0.11934	0.13461	0.14	0.13461	0.11934	0.09663	0.14
0.09957	0.13078	0.15231	0.16	0.15231	0.13078	0.09957	0.16
0.09898	0.14015	0.16941	0.18	0.16941	0.14015	0.09898	0.18
0.09488	0.14731	0.18583	0.20	0.18583	0.14731	0.09488	0.20
0.08742	0.15215	0.20153	0.22	0.20153	0.15215	0.08742	0.22
0.07687	0.15459	0.21643	0.24	0.21643	0.15459	0.07687	0.24
0.06359	0.15459	0.23047	0.26	0.23047	0.15459	0.06359	0.26
0.04806	0.15215	0.24361	0.28	0.24361	0.15215	0.04806	0.28
0.03083	0.14731	0.25578	0.30	0.25578	0.14731	0.03083	0.30
0.01250	0.14015	0.26695	0.32	0.26695	0.14015	0.01250	0.32
-0.00626	0.13078	0.27706	0.34	0.27706	0.13078	-0.00626	0.34
-0.02481	0.11934	0.28607	0.36	0.28607	0.11934	-0.02481	0.36
-0.04247	0.10603	0.29396	0.38	0.29396	0.10603	-0.04247	0.38
-0.05864	0.09104	0.30069	0.40	0.30069	0.09104	-0.05864	0.40
-0.07272	0.07462	0.30623	0.42	0.30623	0.07462	-0.07272	0.42
-0.08423	0.05702	0.31056	0.44	0.31056	0.05702	-0.08423	0.44
-0.09276	0.03852	0.31367	0.46	0.31367	0.03852	-0.09276	0.46
-0.09800	0.01941	0.31554	0.48	0.31554	0.01941	-0.09800	0.48
-0.09976	0.00	0.31616	0.50	0.31616	0.00	-0.09976	0.50
-0.09800	-0.01941	0.31554	0.52	0.31554	-0.01941	-0.09800	0.52
-0.09276	-0.03852	0.31367	0.54	0.31367	-0.03852	-0.09276	0.54
-0.08423	-0.15459	0.31056	0.56	0.31056	-0.15459	-0.08423	0.56
-0.07272	-0.15367	0.30623	0.58	0.30623	-0.15367	-0.07272	0.58
-0.05864	-0.15215	0.30069	0.60	0.30069	-0.15215	-0.05864	0.60
-0.04247	-0.15002	0.29396	0.62	0.29396	-0.15002	-0.04247	0.62
-0.02481	-0.14731	0.28607	0.64	0.28607	-0.14731	-0.02481	0.64
-0.00626	-0.14401	0.27706	0.66	0.27706	-0.14401	-0.00626	0.66
0.00938	-0.14015	0.26695	0.68	0.26695	-0.14015	0.00938	0.68
0.01869	-0.13573	0.25578	0.70	0.25578	-0.13573	0.01869	0.70
0.02783	-0.13078	0.24361	0.72	0.24361	-0.13078	0.02783	0.72
0.03672	-0.12531	0.23047	0.74	0.23047	-0.12531	0.03672	0.74
0.04529	-0.11934	0.21643	0.76	0.21643	-0.11934	0.04529	0.76
0.05345	-0.11291	0.20153	0.78	0.20153	-0.11291	0.05345	0.78
0.06114	-0.10603	0.18583	0.80	0.18583	-0.10603	0.06114	0.80
0.06829	-0.09873	0.16941	0.82	0.16941	-0.09873	0.06829	0.82
0.07483	-0.09104	0.15231	0.84	0.15231	-0.09104	0.07483	0.84
0.08071	-0.08299	0.13461	0.86	0.13461	-0.08299	0.08071	0.86
0.08587	-0.07462	0.11638	0.88	0.11638	-0.07462	0.08587	0.88
0.09027	-0.06595	0.09770	0.90	0.09770	-0.06595	0.09027	0.90
0.09387	-0.05702	0.07862	0.92	0.07862	-0.05702	0.09387	0.92
0.09663	-0.04786	0.05924	0.94	0.05924	-0.04786	0.09663	0.94
0.09854	-0.03852	0.03962	0.96	0.03962	-0.03852	0.09854	0.96
0.09957	-0.01941	0.01985	0.98	0.01985	-0.01941	0.09957	0.98

Table 3. Simulation Result from *nntool* (for Test Input)

h(n) (Test Input)							Cut off frequency (Output)
h(0)	h(1)	h(2)	h(3)	h(4)	h(5)	h(6)	Fc
0.01869	0.01941	0.01985	0.02	0.01985	0.01941	0.01869	0.0201
0.03672	0.03852	0.03962	0.04	0.03962	0.03852	0.03672	0.0387
0.05345	0.05702	0.05924	0.06	0.05924	0.05702	0.05345	0.0621
0.06829	0.07462	0.07862	0.08	0.07862	0.07462	0.06829	0.0816
0.08071	0.09104	0.09770	0.10	0.09770	0.09104	0.08071	0.0932
0.09027	0.10603	0.11638	0.12	0.11638	0.10603	0.09027	0.1185
0.09663	0.11934	0.13461	0.14	0.13461	0.11934	0.09663	0.1434
0.09957	0.13078	0.15231	0.16	0.15231	0.13078	0.09957	0.1698
0.09898	0.14015	0.16941	0.18	0.16941	0.14015	0.09898	0.1738
0.09488	0.14731	0.18583	0.20	0.18583	0.14731	0.09488	0.2012
0.08742	0.15215	0.20153	0.22	0.20153	0.15215	0.08742	0.2256
0.07687	0.15459	0.21643	0.24	0.21643	0.15459	0.07687	0.2375
0.06359	0.15459	0.23047	0.26	0.23047	0.15459	0.06359	0.2569
0.04806	0.15215	0.24361	0.28	0.24361	0.15215	0.04806	0.2629
0.03083	0.14731	0.25578	0.30	0.25578	0.14731	0.03083	0.3011
0.01250	0.14015	0.26695	0.32	0.26695	0.14015	0.01250	0.3267
-0.00626	0.13078	0.27706	0.34	0.27706	0.13078	-0.00626	0.3416
-0.02481	0.11934	0.28607	0.36	0.28607	0.11934	-0.02481	0.3554
-0.04247	0.10603	0.29396	0.38	0.29396	0.10603	-0.04247	0.3858
-0.05864	0.09104	0.30069	0.40	0.30069	0.09104	-0.05864	0.4010
-0.07272	0.07462	0.30623	0.42	0.30623	0.07462	-0.07272	0.4170
-0.08423	0.05702	0.31056	0.44	0.31056	0.05702	-0.08423	0.4392
-0.09276	0.03852	0.31367	0.46	0.31367	0.03852	-0.09276	0.4511
-0.09800	0.01941	0.31554	0.48	0.31554	0.01941	-0.09800	0.4894
-0.09976	0.00	0.31616	0.50	0.31616	0.00	-0.09976	0.4904
-0.09800	-0.01941	0.31554	0.52	0.31554	-0.01941	-0.09800	0.5273
-0.09276	-0.03852	0.31367	0.54	0.31367	-0.03852	-0.09276	0.5396
-0.08423	-0.15459	0.31056	0.56	0.31056	-0.15459	-0.08423	0.5677
-0.07272	-0.15367	0.30623	0.58	0.30623	-0.15367	-0.07272	0.5720
-0.05864	-0.15215	0.30069	0.60	0.30069	-0.15215	-0.05864	0.6005
-0.04247	-0.15002	0.29396	0.62	0.29396	-0.15002	-0.04247	0.6118
-0.02481	-0.14731	0.28607	0.64	0.28607	-0.14731	-0.02481	0.6463
-0.00626	-0.14401	0.27706	0.66	0.27706	-0.14401	-0.00626	0.6592
0.00938	-0.14015	0.26695	0.68	0.26695	-0.14015	0.00938	0.6772
0.01869	-0.13573	0.25578	0.70	0.25578	-0.13573	0.01869	0.7063
0.02783	-0.13078	0.24361	0.72	0.24361	-0.13078	0.02783	0.7194
0.03672	-0.12531	0.23047	0.74	0.23047	-0.12531	0.03672	0.7368
0.04529	-0.11934	0.21643	0.76	0.21643	-0.11934	0.04529	0.7605
0.05345	-0.11291	0.20153	0.78	0.20153	-0.11291	0.05345	0.7739
0.06114	-0.10603	0.18583	0.80	0.18583	-0.10603	0.06114	0.7952
0.06829	-0.09873	0.16941	0.82	0.16941	-0.09873	0.06829	0.8250
0.07483	-0.09104	0.15231	0.84	0.15231	-0.09104	0.07483	0.8416
0.08071	-0.08299	0.13461	0.86	0.13461	-0.08299	0.08071	0.8582
0.08587	-0.07462	0.11638	0.88	0.11638	-0.07462	0.08587	0.8748
0.09027	-0.06595	0.09770	0.90	0.09770	-0.06595	0.09027	0.9033
0.09387	-0.05702	0.07862	0.92	0.07862	-0.05702	0.09387	0.9196
0.09663	-0.04786	0.05924	0.94	0.05924	-0.04786	0.09663	0.9459
0.09854	-0.03852	0.03962	0.96	0.03962	-0.03852	0.09854	0.9629
0.09957	-0.01941	0.01985	0.98	0.01985	-0.01941	0.09957	0.9837

7. Conclusion

From this experiment we estimate the cut off frequency and other parameter from filter coefficient by the help of SOM, and it is quite simple method than complex calculative window method. The above figures show that results come from kaiser window method and artificial neural network is almost same. For the filter designing purpose This SOM training algorithm is much better than other training algorithm like MLP, RBF *etc*. In the previous work, when MLP is used as a training algorithm, result is about 93% accurate. By the using of SOM algorithm, the accuracy of result is almost 98%. So there is increment of 5% in accuracy, which is very effective.

References

[1] S. K. Mitra, “Digital Signal Processing: A Computer-Based Approach”, Second Edition, McGraw-Hill Science/Engineering/Math, **(2001)**, pp. 446-472.
[2] S. Salivahanan, “Digital Signal Processing”, McGraw-Hill, **(2000)**, pp. 735-749.
[3] B. A. Shenoi, “Introduction to Signal Processing and Filter Design”, Wiley Inter science, **(2006)**, pp. 273-285.
[4] S. Haykins, “Neural Networks – A comprehensive foundation”, Prentice – Hall of India Private Limited, New Delhi, **(2003)**.
[5] D. Bhattacharya and A. Antoniou, “Real Time Design of FIR Filters of Feedback Neural Network”, vol. 3, **(1996)**, pp. 1070.
[6] Z.-Z. Zeng, Y. Chen and Y.-N. Wang, “Optimal design study of high-order FIR digital filters based on Neural Network Algorithm”, IEEE international conference, dahan, **(2006)** August 13-16.
[7] G. R. Marthy, “Finite Impulse Response FIR filter Model of Synapses: Associated Neural Network”, Proceeding of the Fourth Annual IEEE International Confidences on Natural Computation, **(2008)**, pp. 3304–3309.
[8] K. J. Hintz and J. J. Spofford, “Evoling Neural Network”, Proceedings of the IEEE Transactions on Communication and Intelligence, **(1990)** May, pp. 333–338.
[9] A. Varun, J. O. Wesley and M. O. Una, “Filter approximation using Explicit Time and Frequency Remain specification”, Proceeding of the Annual Symposium on Artificial Intelligence, Seattle, Washington, **(2006)**, pp. 174 –165.
[10] I. F. C. Emmonual and J. W. Barriel, “Digital Signal Processing, A Practical Approch”, Person Education Allinson N, Yin H, Allinson L, Slack J (eds.) Advances in Self-Organising Maps. Springer-Verlag, London, UK. (Singapore) Ltd., Second Edition, **(2001)**.
[11] J. G. Proakis and D. G. Manolakis, “Digital Signal Processing-Principle”, Algorithms and Applications, Prentice –Hall of India, New Delhi, **(2000)**.
[12] J. Larsen, “Design of Neural Network Filters”, Ph.D. Thesis, Electronics Institute, Technical University of Denmark, **(1993)**.
[13] M. O. Ahmad and J. D. Wang, “An analytical least square solution to the design problem of two-dimensional FIR filters with quadrantally symmetric or anti-symmetric frequency response”, IEEE Trans. Circuits Syst., vol. CAS-36, **(1989)**, pp. 968-979.
[14] S. Burrus, J. A. Barreto and I. W. Selesnick, “Iterative reweighted least-squares design of FIR filters”, IEEE Trans. Signal Processing, vol. 42, **(1994)**, pp. 2926-2936.
[15] C. M. Verleysen, **“**Special Issue on Advances in Self-Organizing Maps”, Neural Networks, vol. 19, no. (5–6), **(2006)**, pp. 721–976.
[16] L. R. Rabiner and B. Gold, “Theory and Application of Digital Signal Processing”, Englewood Cliffs, NJ: Prentice-Hall, **(1989)**.
[17] R. Algazi and M. Suk, “On the frequency weighted least squares design of finite duration filters,” IEEE Trans. Circuits Syst., vol. CAS-34, **(1987)**, pp. 80-95.
[18] E. Torbet, M. J. Devlin and W. B. Dorwart, “A measurement of the angular power spectrum of the microwave background made from the high Chilean Andes,” The Astrophysical Journal, vol. 521, **(1999)**, pp. L79–L82.
[19] T. Saramaki, S. K. Mitra and J. F. Kaiser, “Finite impulse response filter design”, Handbook for Digital Signal Processing, Eds.,Wiley, New York, NY, USA, **(1993)**.
[20] D. F. Specht and P. D. Shapiro, “Training speed comparison of probabilistic neural networks with backpropagation networks,” in Proc. Int. Neural Network Conf., (Paris, France), vol. 1, **(1990)** July, pp. 440- 443.
[21] D. F. Specht, “Series estimation of a probability density function,” Technometrics, vol. 13, no. 2, **(1971)** May.
[22] P. Burrascano, “Learning vector quantization for the probabilistic neural network,” IEEE Trans. Neural network, vol. 2, **(1991)** July, pp. 458- 461.

[23] H. B. Celikoglu, "Application of radial basis functions and generalized regression neural networks in non-linear utility function specification for travel mode choice modeling", Math Comput. Model, vol. 44, **(2006)**, pp. 640–58.
[24] E. J. Tkacz and P. Kostka, "An application of wavelet neural network for classification patients with coronary artery disease based on HRV analysis", Proceedings of the Annual International Conference on IEEE Engineering in Medicine and Biology, **(2000)**, pp. 1391–1393.
[25] H. Demuth and M. Beale, "Neural Network Toolbox for Use with MATLAB", User's Guide, Version 3.0

Authors

Navneet Gupta, he received the B.Tech. Degree in Electronics & Communication from Uttar Pradesh Technical University, Lucknow in 2011 and Masters in Microwave Engineering (Persuing) from Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal. He is currently student in the Department of Electronics Engineering at Madhav Institute of Technology & Science, Gwalior.

Ravindra Pratap Narwaria, he received the B.E. degree in Electronics & Comm. from Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal in 2003 and Masters in Measurement & control from Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal in 2005. He has 08 years of teaching experience. He is currently working as Assistant Professor with the Department of Electronics Engineering at Madhav Institute of Technology & Science, Gwalior