OVMMSOM: A variation of MMSOM and VMSOM as a clusterization technique

Franco Alejandro Sánchez Huertas
Computer Science School
San Pablo Catholic University, UCSP
Arequipa, Peru
franco.sanchez@ucsp.edu.pe

Yván Túpac
Computer Science School
San Pablo Catholic University
Arequipa, Peru
ytupac@ucsp.pe

Raquel Esperanza Patiño Escarcina
Computer Science School
San Pablo Catholic University
Arequipa, Peru
rpatino@ucsp.edu.pe

***Abstract*—This paper proposes the Optimized Vector and Marginal Median Self-Organizing Map (OVMMSOM), a new clustering model that is a Self-Organizing Map (SOM) variant based on order statistics, Marginal Median SOM (MMSOM) and Vector Median SOM (VMSOM). This model combines MMSOM and VMSOM defining their particular importance through a λ participation index. To demostrate the effectiviness of the proposal, images from the COIL100 dataset was clusterized and the Composity Density between and within clusters (CDbw) validity index was used. The performed experiments show that the proposed model outperforms standard SOM network trained by batch and even the MMSOM and VMSOM separately training.**

***Keywords*-images clustering; som; neural network; cdbw index;**

I. INTRODUCTION

The self-organizing maps (SOM) are a type of artificial neural network [1] that establishes a mapping from an input data space onto a 2 or 3 dimensional lattice of nodes so that a number of well defined neuron prototypes is produced. The nodes are organized by a map and they compete in order to win the input patterns [2]. This model are based on the class of SOM training algorithms that employ multivariate order statistics, such as the marginal median (MMSOM) and the vector median (VMSOM) [3]. These SOM variants as well as the standard SOM, are trained using the batch algorithm, and are applied to pattern clustering. The contribution of this work is in the assessment of both SOM training algorithms together with a participation index represented by λ. This index will be chosen based on the best CDbw index [4] obtained during testing. The superiority of this model against the BL-SOM and his parents MMSOM and VMSOM is demonstrated by experiments carried out using the COIL100 data vectorized by Color Layout Descriptor (CLD) [5].

II. SELF-ORGANIZING MAP AND VARIANTS

The SOM forms a nonlinear mapping of an arbitrary D-dimensional input space into a two or three dimensional lattice of nodes. Each node is associated with a weight vector $w = (w_1, w_2, \dots, w_D)^T$ in the input space. The SOM is iteratively trained and it learns the input patterns one at time. The map neurons compete each other in order to be activated by winning the input patterns. Only one neuron wins at each iteration and becomes the winner or the best matching unit (BMU) or as in this case, the pattern winner will update the BMU.x_j and jth are the D-dimensional input feature vector and w_i is the ith D-dimensional weight vector. The first step is the weight vector initialization performed using the linear initialization algorithm. The weight vectors w_i define the Voronoi tessellation of the input space [1], [2]. Each Voronoi cell is represented by its centroid that corresponds to the weight vector w_i. Each input pattern x_j is assigned to a Voronoi cell based on the nearest neighbor condition. The BMU $c(j)$ of the input pattern x_j is defined by:

$$c(j) = \arg\min(\|x_j - w_i\|) \tag{1}$$

where $||$ denotes the Euclidean distance. The updating rule of the ith weight vector, w_i, is computed as:

$$w_i(t+1) = \frac{\sum_{j=1}^{N} \alpha(t) h_{ic(j)}(t) x_j}{\sum_{j-1}^{N} h_{ic}(t)} \tag{2}$$

where N defines the number of patterns x_j that have been assigned to the ith neuron up to the tth iteration and $h_{ic(j)}(t)$ denotes the neighborhood function around the BMU $c(j)$. The learning rate $\alpha(t)$ is a decreasing function in time.

A. MMSOM

The MMSOM calculates the marginal median of all patterns assigned to the winner neuron and updates only the BMU's weight vector. The MMSOM relies on the concept of marginal ordering. The marginal ordering of N input vectors $x_1, x_2, \dots, x_N$, where $x_j = (x_{1j}, x_{2j}, \dots, x_{Dj})^T$, is performed by ordering the winner neuron's vector components independently along each of the D dimensions [3] [6]:

$$x_{q(1)} \le x_{q(2)} \le ... \le xq(N), q = 1, 2, ..., D \tag{3}$$

where q denotes the index of the vector component into consideration. The new weight vector of the BMU emerges from the calculation of the marginal median of all patterns indexed by the BMU. The calculation of the marginal median is defined by [7]

$$marginal\_median\{x_1, x_2 \dots, x_n\} = \\ = \begin{cases} (x_{1(v+1)}, \dots, x_{D(v+a)})^T, & N = 2v+1 \\ (\frac{x_{1(v)} + x_{1(v+a)}}{2}, \dots, \frac{x_{D(v)} + x_{D(v+a)}}{2})^T, & N = 2v \end{cases} \tag{4}$$

where N denotes the number of patterns assigned to the BMU, w_c. The winner neuron is updated by:

$$w_c(t+1) = marginal\_median\{R_c(t) \cup x(t)\} \tag{5}$$

B. VMSOM

The VMSOM calculates the vector median of all patterns assigned to the winner neuron and updates only the BMU's weight vector. The vector median operator is the vector that belongs to the set of input vectors indexed by the BMU, which is the closest one to all the current input vectors. The vector median of N input vectors $x_1, x_2, \ldots, x_N$ is defined by [8]

$$vector\_median\{x_1, x_2 \ldots, x_n\} = x_l \text{ where}$$

$$l = \arg\min_k \sum_{j=1}^{N} |x_j - x_k| \tag{6}$$

The winner neuron is updated by

$$w_c(t+1) = vector\_median\{R_c(t) \cup x(t)\} \tag{7}$$

C. OVMMSOM

The OVMMSOM calculates the new BMU, for every neuron, from the resulting BMU's of MMSOM and VMSOM trained separately. The model are described in the following pseudocode:

Algorithm 1 Pseudocode of OVMMSOM training

Require: dataset $P_{starter} = \{p_1, \ldots, p_n\}$ not empty.
Ensure: clusterized data.

```
while CDbw acceptable do
    MMSOM trainning
        Get BMU: →MMBMU
    VMSOM trainning
        Get BMU: →VMBMU
    Generate a new BMU: →OVMMBMU
        →OVMMBMU = λ × →MMBMU + (1 − λ) × →VMBMU
    OVMMSOM trainning
    Calculate CDbw
end while
return true
```

III. Experiment Results

In order to evaluate this proposal, an experiment looking to solve the clustering problem was performed. This experiment consist in generate clusters from dataset COIL100[1] with 7200 patrons using the Color Layout Descriptor, from the Standard MPEG-7 [5], in order to generate the image's characteristics vectors. Figure 1 shows the experiments made to choose 0.35 as the best value for λ to be used on OVMMSOM, based on the average of values of CDbw on 100 trainings.

After choosing the best value of lambda, we made trainings with the 4 methods: BL-SOM, MMSOM, VMSOM and OVMMSOM. BL-SOM was trained with 50 epochs each time. Table I shows an average result of 100 experiments.

[1]Columbia University Image Library - http://www.cs.columbia.edu/CAVE/software/softlib/coil-100.php

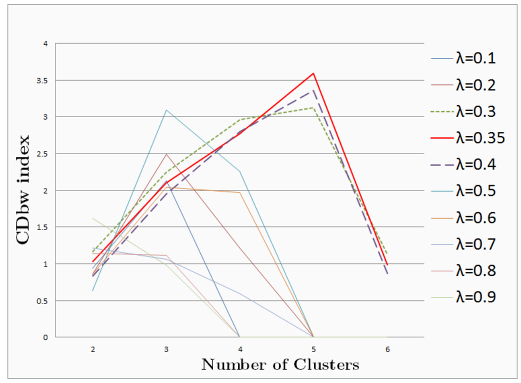

Figure 1. Experiments to choose the best λ

Table I
Comparative Results with COIL100 dataset (C = Number of clusters obtained)

Map Size	BLSOM		MMSOM		VMSOM		OVMMSOM	
	CDbw	C	CDbw	C	CDbw	C	CDbw	C
4x4	1.76	2	2.64	2	1.54	3	2.38	3
5x5	1.93	4	3.15	3	2.89	4	2.48	5
6x6	2.07	5	2.61	3	2.77	4	2.99	5
7x7	2.38	4	2.96	5	2.56	4	2.65	4
8x8	2.59	5	2.49	4	3.12	5	3.56	5
9x9	2.29	5	3.42	4	3.28	4	3.78	5

(Method spans BLSOM, MMSOM, VMSOM, OVMMSOM.)

IV. Conclusion

It was determined experimentally that the union of MMSOM and VMSOM training with an index of participation reaches better results compared with the application of two separate, where results are the validation index CDbw value applied to the resulting groups in the set of images used. Through the experiments could be determinate the best value for λ as 0.35. With this index both traning methods can offer better results that their individual application. Another pro of the model is the continuity to get the best number of clusters without dependency of the size of the map.

References

[1] S. Haykin, "Neural networks: A comprehensive foundation," *Journal of Multi-Criteria Decision Analysis, Upper Saddle River, Prentice-Hall*, 1999.
[2] T. Kohonen, "Self-organizating maps," *Springer-Verlag*, 2000.
[3] I. Pitas, C. Kotropoulos, N. Nikolaidis, R.Yang, and M. Gabbouj, "Order statistics learning vector quantizer," *IEEE Trans. Image Processing, vol. 5, pp. 1048-1053*, 1996.
[4] S. Wu and T. Chow, "Clustering of the self-organizing map using a clustering validity index based on inter-cluster and intra-cluster density," *Pattern Recognition Journal, págs. 175-188*, 2003.
[5] P. Salembier and J. R. Smith, "Mpeg-7 multimedia, description schemes," *Transcriptions on Circuits and Systems For video Technologies, pags. 748-759.*, 2002.
[6] C. Kotropoulos and I. Pitas, "Self-organizing maps and their applications in image processing, information organization, and retrieval," in *Nonlinear Signal and Image Processing: Theory, Methods, and Applications*, K. E. Barner and G. R. Arce, Eds. Boca Raton, FL: CRC Press, 2004.
[7] I. Pitas and P. Tsakalides, "Multivariate ordering in color image restoration," 1991.
[8] J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filters," 1990.