Implementing Self Organising Map to Organise the Unstructured Data

S P Lim¹, C K Lee¹, J S Tan¹, S C Lim² and C C You¹
¹Faculty of Information and Communication Technology, Universiti Tunku Abdul Rahman, Jalan University, Bandar Barat, 31900 Kampar, Perak, Malaysia.
²Department of Computing and Information Technology, Tunku Abdul Rahman University College, Penang Branch Campus, 77, Lorong Lembah Permai Tiga, 11200 Tanjong Bungah, Pulau Pinang, Malaysia.

Email: limsp@utar.edu.my

Abstract. Surface reconstruction is significant in reverse engineering because it should present the correct surface with minimum error using the data available. It has become a challenging process when the data are in the unstructured type and the existing methods are still suffering from accuracy issues. The unstructured data will produce an incorrect surface because there is no connectivity information among the data. So, the unstructured data should undergo the organising process to obtain the correct shape. The Self Organising Map (SOM) has been extensively applied in previous works to solve surface reconstruction problems. However, the performance of the SOM models has remained uncertain. It can be evaluated and tested using different types of data sets. The objectives of this research are to examine the performance and to determine the weaknesses of SOM models. 2D SOM, 3D SOM, and Cube Kohonen (CK) SOM models are investigated and tested using three data sets in this research. As shown in the experimental results, the CKSOM model has proved to perform better because it can represent the correct closed surface with the lowest minimum error.

1. Introduction
Surface reconstruction is a crucial step and challenging task [1,2] in reverse engineering because it should be able to produce an accurate shape [3] with the data obtained [4-6]. Data collection is a vital process because the data will indicate reconstruction and representation methods. Generally, the data can be either in the structured or unstructured type. The shape of the surface will also be affected by the data type. The connectivity information of data is needed in surface reconstruction because unstructured data are usually obtained [6] during data collection. The organising process should perform on unstructured data to represent the correct and smooth surface with minimum error.

Recently, learning-based methods have been widely used in previous works to tackle surface reconstruction case studies [6]. Self-Organising Map (SOM) is one of the popular learning-based (unsupervised learning) methods introduced by Professor Teuvo Kohonen [7]. It solves and handles various types of surface reconstruction problems, such as organising the unstructured data, which can refer to the works of [8-13]. SOM is applied in the research because it can present high dimensional data with lower-dimensional [14] while preserving the topological relationship [15]. The performance of the SOM models has remained unstable because limitations still appear. Also, the models are not verified using different types of data sets [16]. As stated in [11,17], the 2D SOM model contains gaps or holes.
while producing the surface for a closed surface data, and the internal neurons in the 3D SOM model will incorrectly represent the object surface [11].

Besides that, not all the learning-based methods such as Deep Learning (DL) or enhancement are useful in solving surface reconstruction problems. When proposing a new or enhanced way to solve a case study, no further and additional issues should appear. As shown in [17], DL integrates with the SOM model to handle the surface reconstruction problems. The results have shown significant performance while performing with a large point cloud, yet an incorrect surface has produced for the shape. Additionally, the 3D reconstruction problems in [18] are solved using DL. The method is proposed based on Convolutional Neural Network, yet unable to correctly predict due to the outlier points in the sparse point cloud [18]. So, this research will not consider DL due to it is still suffering from limitations. The problems should not occur while proposing a new method or enhancement because sometimes it is only suitable for certain case studies with its existing framework or structure. Therefore, the objectives are to examine the performance and identify the limitation of different SOM models. 2D SOM, 3D SOM, and Cube Kohonen (CK) SOM models are investigated and tested using three datasets in this research.

2. Flow of Experiment

This section elaborates the steps followed for all SOM models. The steps are derived based on the works of [7,8,10,11,13,19,20]. The 2D SOM and 3D SOM models are referred from [7] whereas CKSOM model is referred from [11]. Generally, there are five steps involved in the flow.

2.1. Acquiring Data

Acquiring data is the first step for the experiment. The data are inserted into the SOM model to perform the organising process. Two sets of primitive objects (Cube and Sphere) and one set of medical image data (talus bone) are used for this research. All the data are in 3D coordinates (x, y, z) and unstructured type. Refer to [13, 21] for the characteristics and details of the data. For the SOM model, the input vector will randomly select from the data sets.

2.2. Initialising Parameters

All the parameters involved in the SOM model are assigned in the second step, Initialising Parameters. Table 1 shows the parameters set for the SOM model. Refer to [11] for the values of the parameters assigned. Rectangular topology is used for the SOM model in this research. For the SOM model, the total input layer neuron is determined based on the dimension size. As the data are in 3D, hence three neurons are assigned. Besides that, the grid size, \( n \), Initial Learning Rate, \( \omega_0 \) and Initial Radius, \( \sigma_0 \) of SOM models are defined in this step. The learning rate and radius will gradually decrease to 0.01 and 1 during the learning process. Maximum iteration, \( T \) is the total number of iterations for the model in learning towards the input vector. The Time Constant, \( \lambda \) is applied to manage the decay rate for the learning rate and the radius. The value is derived based on the maximum iterations and initial radius. The Merging Neurons and Detecting Neighbours for CKSOM model from [11] are also grouped in this step to standardise the flow of experiment. Refer to [11] for the details.

| Parameter                  | Value                                      |
|----------------------------|--------------------------------------------|
| Total of input layer neuron| 3                                          |
| Grid size                  | 10, 20, 30                                 |
| Initial Learning Rate      | 0.9                                        |
| Initial Radius             | Half of the grid size                      |
| Maximum iteration          | 30000                                      |
| Time constant              | Based on Maximum Iteration and Initial Radius |

Table 1. Parameters Involved
2.3. Generating Weights
The weights for each neuron, \( W \), are randomly generated within 0 to 1 in the third step, Generating Weights. They are used to learn towards the input vector. The weights will be updated with new values during the learning process.

2.4. Learning Process
Competition, cooperation and adaptation processes are gathered in the fourth step, Learning Process. In Competition, one input vector, \( V \), is randomly chosen from the data. The Euclidean distance formula is used to determine the closest neuron to \( V \). The neuron with the least distance compared to \( V \) will be picked as the winning neuron. In Cooperation and Adaptation, Gaussian function formula is used to update the weights of the winning and neighbouring neurons. The learning process will be terminated when the maximum iteration reached.

2.5. Producing Output
Producing Output is the last step in the flow. The output produced by the SOM model is the weights of each neuron (coordinates \( x \), \( y \) and \( z \)) and it is in the structured type.

3. Analysis and Discussion
This section presents the analysis and discussion based on the experimental results. The experimental results of 2D SOM, 3D SOM, and CKSOM models are analysed and compared accordingly. Table 2 and Table 3 show the metric evaluation and visualisation for different SOM models and data sets. Three data sets and grid sizes are used to test the efficiency of the SOM models. The quantisation error is the metric evaluation recommended by [11,22-24]. The error can prove the accuracy because it measures the fitting of the neural map towards the data. Also, minimum and maximum errors and CPU time are the metric evaluation suggested by [11]. The errors can show the minimum and maximum errors for models while CPU time can prove the speed in producing the result. As for visualisation, it can represent the surface of the models. All the experiments are performed using Dev C++ and the results are visualised using GNUPlot.

| Data  | Grid Size | Min Error | Max Error | Quantisation Error | CPU Time (s) |
|-------|-----------|-----------|-----------|--------------------|--------------|
| 2D SOM | 3D SOM | CK SOM | 2D SOM | 3D SOM | CK SOM | 2D SOM | 3D SOM | CK SOM | 2D SOM | 3D SOM | CK SOM |
| Cylinders | 10 | 0.001784 | 0.000950 | 0.000648 | 1.222400 | 1.152200 | 1.129200 | 0.123322 | 0.047082 | 0.054727 | 0.548 | 2.243 | 1.023 |
| 20 | 0.000600 | 0.000371 | 0.000164 | 1.170800 | 1.129100 | 1.100100 | 0.057975 | 0.002352 | 0.002970 | 1.261 | 37.763 | 0.562 |
| Sphere | 30 | 0.000153 | 0.000018 | 0.000004 | 1.139300 | 1.034200 | 0.839000 | 0.051461 | 0.001950 | 0.021550 | 2.263 | 72.360 | 10.731 |
| 10 | 0.02093 | 0.00556 | 0.000794 | 1.361500 | 1.152700 | 0.948100 | 0.097754 | 0.003039 | 0.003733 | 0.410 | 3.199 | 2.05 |
| 20 | 0.00996 | 0.000505 | 0.000056 | 1.173200 | 1.193900 | 0.825100 | 0.047708 | 0.001007 | 0.001069 | 1.588 | 25.836 | 7.765 |
| 30 | 0.001002 | 0.000034 | 0.000063 | 1.149200 | 1.123000 | 0.697200 | 0.053757 | 0.001494 | 0.004457 | 2.727 | 75.159 | 19.253 |
| Tables | 10 | 0.00072 | 0.000100 | 0.000022 | 1.221500 | 1.179400 | 1.023400 | 0.091611 | 0.003483 | 0.001003 | 0.555 | 2.418 | 1.401 |
| 20 | 0.001004 | 0.000065 | 0.000062 | 1.137700 | 1.020900 | 0.805100 | 0.051606 | 0.001707 | 0.002725 | 1.209 | 27.936 | 6.645 |
| 30 | 0.001013 | 0.000074 | 0.000018 | 1.087900 | 0.980672 | 0.607988 | 0.074825 | 0.002166 | 0.001064 | 2.494 | 70.809 | 16.275 |

As shown in Table 2 and Table 3, for all SOM models, when the grid size increased, the quantisation error reduced and CPU time increased for all data sets. Also, when the grid size increased, the surface produced will be better because more neurons are involved. The 2D SOM model contains the highest quantisation error and lowest CPU time, as shown in Table 2. When quantisation error is larger, the accuracy is lower because the surface produced is less approximated towards the data sets. Although 2D SOM model can produce the results faster, the surface still contains holes or gaps, which can be referred to the circles as shown in Table 3. The 2D SOM model is incapable of reconstructing the surface of closed surface data [11,20,25]. Hence, it is considered weaker than 3D SOM and CKSOM models because it contains higher quantisation error and is unable to produce correct closed surface.

As for the 3D SOM model, it has the lowest quantisation error and highest CPU time. The model consists of a larger number of total output neurons (for example 270000 for the grid size of 30), which indirectly increases the possibility for the model to obtain better winning neurons during the learning process. However, it requires a longer time to produce the result and the surface is still in the
unstructured type, as shown at the triangles in Table 3. The connectivity information of internal and external neurons of 3D SOM model [11] have updated during the learning process. Hence, the incorrect surface is presented as result. So, it is also not considered as the best SOM model because incapable to represent the correct closed surface.

Table 3. Visualisation for different SOM models and data sets

| Data     | Grid Size, n | 2D SOM | 3D SOM | CKSOM |
|----------|--------------|--------|--------|-------|
| Cube     |              |        |        |       |
| 10       |              |        |        |       |
| 20       |              |        |        |       |
| 30       |              |        |        |       |
| Sphere   |              |        |        |       |
| 10       |              |        |        |       |
| 20       |              |        |        |       |
| 30       |              |        |        |       |
| Talus Bone |             |        |        |       |
| 10       |              |        |        |       |
| 20       |              |        |        |       |
| 30       |              |        |        |       |

For CKSOM model, it has the lowest minimum and maximum errors with medium speed. The values of minimum error show that CKSOM model can learn closer towards the data set. Also, the quantisation error for CKSOM is near to the value produced by 3D SOM model. If CPU time remained constant, definitely CKSOM model can obtain a better result with lower quantisation error. Besides that, CKSOM model can produce correct closed surface, as shown in Table 3. If can improve on the accuracy, it will be better as lower quantisation error is preferred. It is suggested to integrate with optimisation methods in optimising the error. Therefore, CKSOM model is performed better in this research because it can obtain the lowest minimum and maximum errors with a correct closed surface.
4. Conclusion and Future Work
In this research, three types of SOM models have been implemented and tested with various data sets in solving surface reconstruction problems. The performance of each SOM model has been evaluated through quantitative methods (error and CPU time) and validated through qualitative method (visualisation). The main target of surface reconstruction is to obtain good accuracy and speed with correct surface. As shown in the experimental results, CKSOM model contains better performance. It can achieve the lowest minimum error, medium speed along with the correct closed surface.

For future work, it is suggested to overcome the issues discussed in the analysis and discussion section. The holes or gaps for 2D SOM model can be handled by connecting the neuron at the boundaries. The internal neurons for 3D SOM model can be removed and rejoined by avoiding the connectivity issues. It can also reduce the number of neurons involved in the learning process for 3D SOM model. As for CKSOM model, improvement can be applied to the quantisation error by integrating with optimisation methods [12], such as Particle Swarm Optimisation [26]. The parameters in the SOM model, especially the grid size can be tested with different values to obtain better results.

References
[1] T Elmidany, A Elkeran, A Galal, M Elkhateeb 2011 J. of Control Eng. and Tech. pp 34
[2] A Gálvez, A Iglesias, J P Pey 2012 Info. Sc. 182 pp 56
[3] R Ann Joachim Martin et al 2020 IOP Conf. Ser.: Mater. Sci. Eng. 912 032015
[4] G Guo, X Wu, M Y Wang, J Wu 2010 IEEE ICMA pp 1783
[5] M Zhou 2011 Procedia Engineering 23 pp 594
[6] V L DalleMole, R L M E Do Rêgo, A F R Araújo 2010 Self-Organizing Maps pp 167
[7] T Kohonen 1990 Proceedings of the IEEE 78 pp 1464
[8] F Forkan 2009 Master Thesis Universiti Teknologi Malaysia
[9] R L M E Do Rêgo, A F R Araújo 2010 Inter. Joint Conf. on Neur. Net. pp 1
[10] P Pandunata 2011 Master Thesis Universiti Teknologi Malaysia
[11] S P Lim, H Haron 2013 IEEE Trans. on Neur. Net. and Learn. Sys. 24 pp 1414
[12] A Iglesias, A Gálvez 2014 Math. Problems in Eng. 2014 pp 1
[13] S P Lim 2015 PhD Thesis Universiti Teknologi Malaysia
[14] D Miljković 2017 MIPRO 2017 pp 1252
[15] S Aly, S Almotairi 2020 IEEE Access 8 pp 107035
[16] S P Lim, H Haron 2014 Art. Int. Rev. 42 pp 59
[17] W P Lee, S Hasan, S M Shamsuddin, N Lopes 2017 Int. J. Adv. Soft Com. App. 9 pp 1
[18] P Mandikal, R V Babu 2019 WACV 2019 pp 1
[19] M Hoffmann 1999 Publ. Math. 54 pp 857
[20] F Boudjema, P B Enberg, J G Postaire 2003 IEEE ICSMC 3 pp 2418
[21] R Daud, M R Abdul Kadir, S Izman, A P Md Saad, M H Lee, A Che Ahmad 2013 J. of F. & Ank. Sur. 52 pp 426
[22] E A Uriarte, F D Martin 2005 Inter. J. of App. Maths. and Com. Sc. 1 pp 19
[23] M Kanimozhi, C H H Bindu 2013 Inter. J. of Adv. Res. in Com. and Comm. Eng. 2 pp 3968
[24] R Mancini, A Ritacco, G Lanciano, T Cucinotta 2020 IEEE 32nd SBAC-PAD pp 209
[25] J Chao, K Minowa, S Tsujii 1992 IEEE ICSE pp 24
[26] S P Lim et al 2020 IOP Conf. Ser.: Mater. Sci. Eng. 864 012068

Acknowledgement
The research has been carried out under Fundamental Research Grant Scheme project FRGS/1/2019/ICT02/UTAR/02/3 provided by Ministry of Higher Education of Malaysia.