

Effect of Neighbourhood Size Selection in SOM-Based Image Feature Extraction

Sokchoo Ng and Mieowkee Chan

Abstract—Feature extraction is one of the processes in pattern recognition systems. It is the procedure of selecting the best possible subset of input features from a high dimensional problem space and transforming them into a lower dimensional space. Mathematically, feature extraction can be perceived as dimensionality reduction problem. Kohonen's self-organizing map (SOM) is an unsupervised learning approach in which a group of neurons regularly adjusts their basic structure as a function of its experience and environment. Preliminary research reported that neighbourhood functions is one of the necessary parameters that influence the results. The neighbourhood size refers to the region covered by the activated neighbouring neurons in relation to the winner neuron in each iteration of learning. This study aims to evaluate the implementation of SOM in image feature extraction. Different neighbourhood size value was tested in the SOM learning process to observe the quality of the extracted feature. Gaussian neighbourhood function was applied to the SOM-based feature extraction in current study. This study also compared the feature extraction performance of different neighbourhood functions. Experimental results show that the resulting feature loses its details as the neighbourhood size increases. Gaussian neighbourhood function gives more significant extracted feature.

Index Terms—Feature extraction, Gaussian neighbourhood function, high dimensionality, self-organizing map (SOM).

I. INTRODUCTION

A statistical pattern recognition system can be divided into three modules: preprocessing, feature extraction and classification (Fig. 1) [1]. Preprocessing module removes noise data from the object of interest. Normalization is conducted on the datasets to ensure that the result of image analysis in later stage is independent of the parameters' variation. Feature extraction module discriminates the class memberships (pattern) of the datasets. Mathematically, this module transfers the pattern from a n -dimensional data space, $X \in R^n$ into a low dimensional feature space, $f \in R^d$. Classification module defines decision boundaries in the feature space in order to map the observed object to its

respective class.

The current study focuses on the feature extraction module. There are two steps on feature extraction: feature construction and feature selection [2]. Feature construction is the process of modeling the problem space. It identifies the key attributes that form the observed novel object. For a visual object such as image, the attributes are pixel bits that hold the value of RGB, colour information and the resolution of the image. The following step, feature selection, selects the relevant and discriminative feature to create the class memberships for classification. In a high dimensional problem space such as image data, feature extraction task can be challenging as many variables (attributes) are involved in the problem space. This will cause the feature model to be highly complex. The solution to reduce the complexity of the model is to reduce the dimension of the problem space. The advantages of applying dimensionality reduction in feature extraction are cutting down the computational complexity yet at the same time retaining the discriminative dimension in the multidimensional problem space. Kohonen's self-organizing map (SOM) has been evidenced as effective dimensionality reduction method in [3]-[6]. SOM is a variation of artificial neural network which applies unsupervised learning mechanism to study its residing environment. SOM has been widely used in clustering, predictive system and data compression [7]-[11]. Natita, Wiboonsak and Dusadee [12] reported that learning rate and neighbourhood functions are necessary parameters in SOM which can influence the results. This study evaluates the application of SOM in image feature extraction. The effect of applying different neighbourhood size value in relation to the resulting feature is observed.

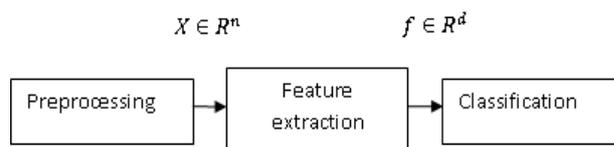


Fig. 1. Statistical pattern recognition system adapted from Jiang (2011).

II. RELATED WORK

Some of the well-established feature extraction techniques are Histogram analysis, Sobel edge detection, Canny edge detection and grid colour moment [13]-[16]. Sobel edge detection looks at images as functions of intensity. A pixel is identified as an edge when the intensity sharply changes to its neighbours. The edge is a linear shape along each maximal changed pixel. Working on the basis of "a maximum of the first derivative" property, discrete differential operator is applied to approximate the first derivative. Canny edge detection improves Sobel by applying non-maxima

Manuscript received September 5, 2018; revised February 7, 2019.

Sokchoo Ng is with the Faculty of Science, Technology, Engineering and Mathematics, International University of Malaya-Wales, Malaysia (e-mail: ashleyng@iumw.edu.my).

Mieowkee Chan is with the Centre for Modelling and Simulation, Faculty of Engineering and the Built Environment, SEGi University, Malaysia (e-mail: mkchan@segi.edu.my).

suppression to get thin lines that represents the edges [17]. Similar to Sobel, Canny edge detection at the first step detect the pixels with maximal intensity change. Canny introduces another path with local maxima perpendicular to the edge direction in the second step. The pixel with the gradient magnitude less than a threshold, T_1 is suppressed. The later path stops as soon as the gradient magnitude falls below the second threshold, T_2 . The introduction of two thresholds can filter noise data better in order to obtain thin, clear edges. In recent years, machine learning is adopted in feature extraction such as Ant algorithms, self-organizing map and particle swarm optimization. Some of the literature can be found in [18]-[21]. The use of machine learning in feature extraction has advantages over traditional feature extraction techniques in terms of robustness and automation. The learning mechanism enables the system itself to search towards high image gradient change and converge onto the strongest edges of the image [22]. For instance, the distributed adaptive thresholding in Ant algorithm removes the requirement of a sensitivity threshold in many traditional methods [22]. Other research employed dimensionality reduction methods such as principal components analysis (PCA) [23], [24], linear discriminant analysis (LDA) [25], locality preserving projections (LPP) [26] and their numerous variants. PCA is a statistical approach to discard redundant features from a digital image. Ng [24] applied PCA to reduce digital image feature. The dimension reduction process consists of four major steps: normalize image data, calculate the covariance matrix from the image data, perform Single Value Decomposition (SVD) and lastly locate the projection of image data into a new basis with the reduced features. The digital image is represented as distributed datasets in a multi-axes system. This multi-axes system is perceived as a multi-dimensional problem space. PCA first identifies the greatest variance of the dataset by obtaining its covariance matrix which consists of eigenvector and eigenvalue sets. Eigenvector with the largest eigenvalue is the direction of the greatest variation in the image dataset, known as the principal features of the image data. A new image representation with reduced dimension is obtained by multiplying the original image dataset with its principal components (eigenvector). Ng's study compared the quality of the feature reduced images with difference variance values for the principal components. The results showed that PCA achieved 35.3% of the image file size reduction for the best acceptable human visual quality. However, PCA is limited to linear systems with uncorrelated variables. LDA is another linear dimension reduction method. Unlike PCA which finds the maximum variance of the dataset, LDA finds a set of vectors (axes) that maximize the separation of classes from the scattered dataset. LDA conducts feature selection by maximizing the difference between classes and minimizing the distance within classes [27], [28]. Since LDA involves two-level data processing (between-class and within-class scatter), it possess higher computational complexity compared to PCA.

III. KOHONEN'S SELF-ORGANIZING MAP

Proposed by Kohonen [3], self-organizing map is an

unsupervised neural network that projects multivariate datasets onto a one or two-dimensional grid of neurons or known as map, with the purpose of discovering the pattern (feature) in the data. The neurons of the map can be arranged on either a rectangular or hexagonal lattice. Fig. 2 presents the architecture of a SOM neural network. The network consists of one input and output layer. Each input, x_i in the input layer is connected to all output neurons in the output layer. Each neuron comprises a weight vector, $[w_1, w_2, \dots, w_i]$ having the same dimensionality as the input vectors, $[x_1, x_2, \dots, x_i]$. There are two phases in the SOM learning process: *competitive* and *cooperative*. Competitive phase is the process to identify the Best Matching Unit (BMU), which is the neuron having the distance closest to the input vector. BMU is known as the winner neuron. Followed by cooperative phase, the weight of the neighbouring neurons which are close to BMU in the SOM lattice are then adjusted towards the input vector (Fig. 3). For each iteration, BMU is found and the weight vectors for the activated neighbourhood will be updated.

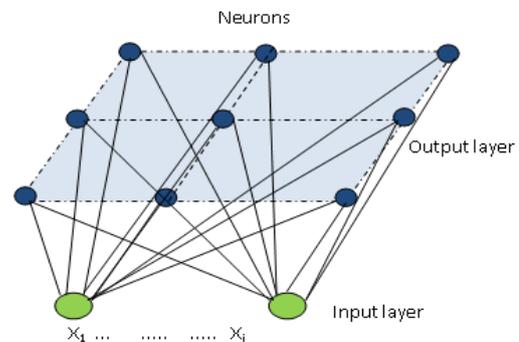


Fig. 2. Architecture of a SOM neural network.

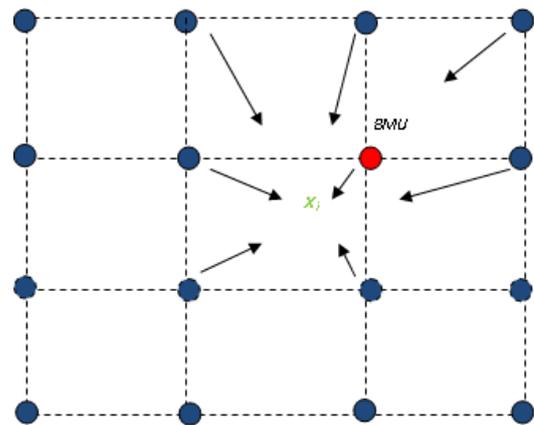


Fig. 3. Neighbouring neurons of BMU moves towards input vector.

The region of the activated neighbourhood depends on the neighbourhood function chosen and the neighbourhood size defined prior to the training. The learning process continues until the weights connecting the input data to the neurons have stabilized. SOM learns both the distribution and topology of the input vectors.

IV. SOM-BASED FEATURE EXTRACTION

Suppose that an image is a high dimensional problem space as it contains a huge amount of information. Mathematically, an image is represented by a group of pixels arranged in the

form of two-dimensional matrix as shown in (1) [29].

$$f(x, y) = \begin{bmatrix} f(0,0) & \cdots & f(0,m-1) \\ \vdots & \ddots & \vdots \\ f(n-1,0) & \cdots & f(n-1,m-1) \end{bmatrix} \quad (1)$$

where x and y are the coordinates of the pixels in the image and $f(x, y)$ is the corresponding colour or gray level depending on its value type. Feature extraction draws out the relevant information from the image data to form feature vectors. These feature vectors are known as class labels which is later used by classifiers to map the input unit with output unit in pattern recognition. Kohonen's self-organizing map [7] is applied to extract the discriminative feature from the problem space. The underlying principle of SOM-based feature extraction is that by removing the redundant data from problem space, the small size feature sets which contain discriminating information is retained. These feature sets are the class labels in pattern recognition. This feature extraction method has the potential advantage of preventing generating different feature codes for an object in the same class especially for high dimensional problem space which involves large amount of variables. There are four steps in SOM-based feature extraction: (1) parameters initialization (2) competitive (3) cooperative (4) feature vector construction.

A. Parameters Initialization

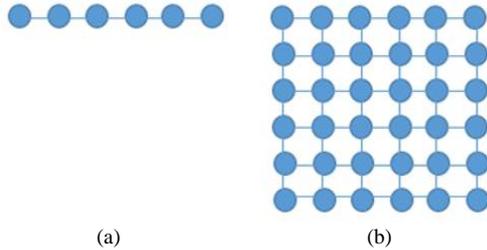


Fig. 4. Neuron arrangement (a) one dimensional array (b) two dimensional matrix lattice.

The important properties that define a self-organizing map are the number of neurons, neurons' weight, learning rate, neighbourhood function and neighbourhood size. To implement SOM learning, the first step is to set the values for these properties. The neurons can be arranged in either a one-dimensional array or a two-dimensional matrix lattice (map) (see Fig. 4). To reduce data dimension, SOM applies the neuron array (one dimension) or map (two dimensions) to plot the similarity of the data by grouping the similar data items together. For image processing, the neurons will be arranged in the form of two-dimensional matrix lattice. Random values will normally be assigned to the neurons' weight vector at the initial stage of the training. After several iterations, the neurons' weight values will be updated. Data with similarity will gradually converge to form a group. Some of the choices for neighbourhood functions are Gaussian, bubble and epanechicov. Neighbourhood size dictates the region of activated neighbouring neurons from BMU. This study works on the hypotheses that the selection of neighbourhood size and function will influence the resulting feature vector. The learning rate is a decay function over time.

The learning rate is initialized and reduced over time to gravitate to a convergence of the map.

B. Competitive Phase

This stage reads the input, $f_i(x,y)$ from the problem space and computes its Euclidean distance d_j , between the input and each output neuron using (2).

$$d_j = \sum_{i=0}^{n-1} (f_i(x, y) - w_{ij})^2 \quad (2)$$

where w_{ij} is the weigh from input f_i to neuron j . The neuron having the smallest d_j value is identified as BMU.

C. Cooperative Phase

Two activities are carried out in this stage: activated neighbouring neurons identification and updating their weight vectors. Neurons are adapted in a recursive manner [30]. The neighbourhood function's radius is decremented over time as training progresses to enable the neurons to converge to a final mapping. The region that covers the activated neighbouring neurons depends on the choice of neighbourhood function selected. Fig. 5 shows some common choices for neighbourhood function. The neighbourhood's region spans symmetrically around the BMU. Bubble neighbourhood function is a constant function around the winner unit. It means every neuron in the neighbourhood is updated with the same proportion of the difference between the neuron and the sample vector. For the case of Gaussian neighbourhood function, the activated neighbourhood $N_{j,i}$ is calculated by (3):

$$N_{j,i(x,y)} = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2}\right), d_{j,i}^2 = \|x_j - y_i\|^2 \quad (3)$$

where σ^2 is variance of the Gaussian function and $d_{j,i}$ is the magnitude of the x and y -axis for the map. The new weight of BMU and its activated neighbouring neurons is computed by (4):

$$w_j(t+1) = w_j(t) + \theta \cdot N_{j,i(x,y)}(t) \cdot f_i(x, y) - w_{ij}(t) \quad (4)$$

where t is the training iteration number and θ is the learning rate of neuron. The outcome of this stage is a prototype with its adjusted weight vectors.

D. Feature Vector Construction

As a result of these collective and cooperative learning, the network is tuned to create localized responses to the input vectors and thus reflect the topological ordering of the input vectors. The ordering reflects the feature space for the class label (pattern) [31]. This stage creates the image feature vectors, P from the feature space by taking the diagonal matrix of the product of the prototype, obtained from (4) with the image pixels by using (5):

$$P = \text{diag}(W_j * f(x, y)) \quad (5)$$

where W_j is prototype and $f(x,y)$ is the input vectors. Sampling

of the original image pixels according to the minimum value feature vector is then conducted to yield the class label, which is the pattern of the image object.

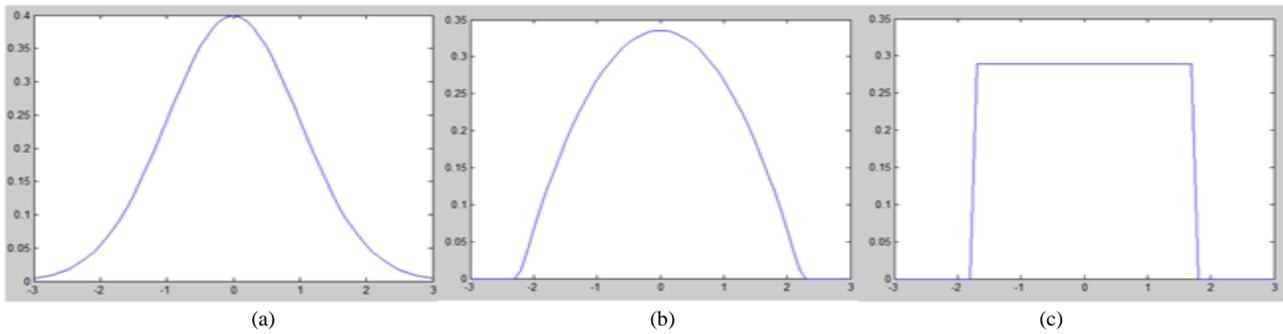


Fig. 5. Neighbourhood functions (a) Gaussian kernel (b) Epanechnikov kernel (c) Bubble kernel.

V. RESULTS AND DISCUSSION

The SOM-based feature extraction was implemented on polyvinylidene fluoride blended membranes used in waste water treatment. The original image of polyvinylidene fluoride blended membranes (Fig. 5a) was taken using atomic force microscopy (AFM). Fig. 6a shows the surface of the polyvinylidene fluoride blended membranes. The initial setting of the SOM parameters is as follows: $\theta = 0.1$, number of neurons = 16, Gaussian neighbourhood function is used. Fig. 6 shows the comparison of the original image and the resulting feature extracted with neighbourhood size scale, $n = 1, 2$ and 3.

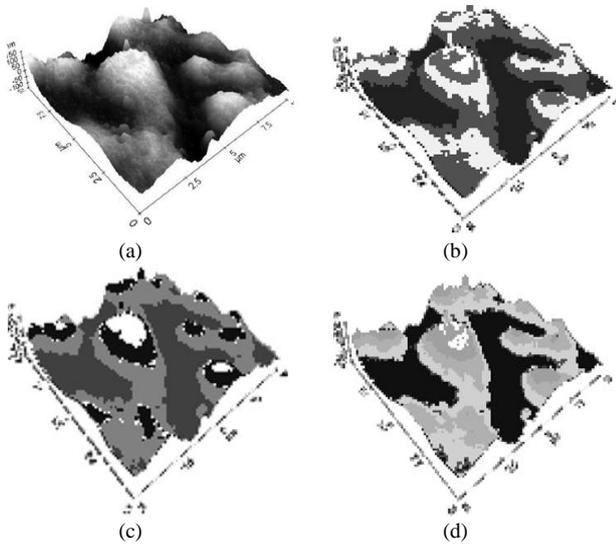


Fig. 6. (a) Image of a polyvinylidene fluoride blended membrane (b) feature extracted with $n=1$ (c) feature extracted with $n=2$ (d) feature extracted with $n=3$.

The neighbourhood size determines the variance of the Gaussian neighbourhood function. Fig. 7 shows the region covered by the activated neighbouring neurons with different n values. As the neighbourhood size increases, the width (variance) of the Gaussian neighbourhood function increases. This brings forth the wider span of activated neighbouring neuron area. Experimental results show that as the neighbourhood size increases, a better smoothing effect is obtained. But this also results in the loss of details for the image. Fig. 6 shows that the increase of neighbourhood size also causes a better segmentation of features.

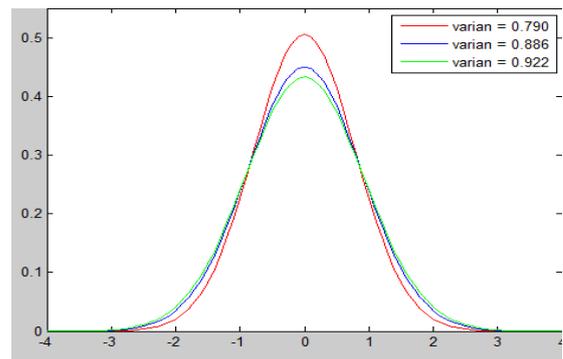


Fig. 7. Gaussian neighbourhood function with different variance, σ^2 .

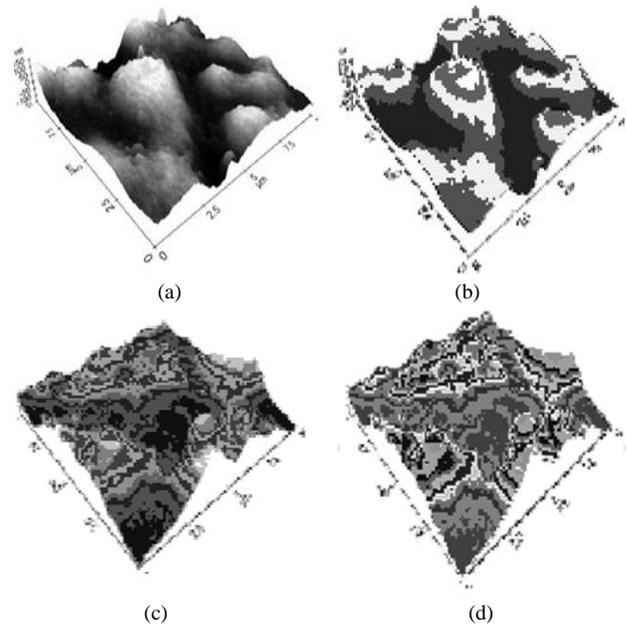


Fig. 8. (a) Image of a polyvinylidene fluoride blended membrane; feature extracted by (b) Gaussian kernel (c) Bubble kernel and (d) Epanechnikov kernel.

The second experiment was conducted to investigate the effect of neighbourhood function to the extracted feature. In this experiment, three neighbourhood functions, namely Gaussian kernel, Bubble kernel and Epanechnikov kernel were applied respectively in the SOM-based feature extraction. Fig. 8 shows the results of the extracted feature. For the same neighbourhood size, Gaussian neighbourhood function provides a more significant feature segmentation comparing to Bubble kernel and Epanechnikov kernel. Both Bubble and Epanechnikov kernel gives a very similar

extraction outcome. One possible explanation to the result is that the neighbourhood neurons induce an attenuation to the adaptation as the grid distance to the winning neuron grows [26].

VI. CONCLUSION

Feature extraction is the stage prior to classification in pattern recognition systems. In image processing, its purpose is to identify the basis information that forms the image object. Traditional image feature extraction techniques include transforms and series expansion, histogram analysis and mathematical differentiation. SOM-based feature extraction is a technique that utilizes the spatial information of the image object to construct the image's pattern. SOM extracts the feature vector from image dataset by reducing its dimensionality. The application of SOM to extract prototype patterns from a polyvinylidene fluoride blended membrane has been presented in this paper. Experiments were conducted to investigate the effect of neighbourhood size to the performance of the feature extraction. The finding showed that more details are missing in the prototype pattern as the neighbourhood size increases. This aspect applies to Gaussian neighbourhood function. This study also investigates the effect of neighbourhood function to the extracted feature. Experiment results showed that Gaussian neighbourhood function outperformed Bubble and Epanechnikov neighbourhood function to give more significant feature segmentation. As future development of this work, the experiment will be extended to include other performance measure to the extraction method such as the classification of pixel in image space.

REFERENCES

- [1] X. Jiang, "Linear subspace learning-based dimensionality reduction," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 16-26, 2011.
- [2] L. S. Shafiq and E. Pérez, 2009, "Feature construction and feature selection in presence of attribute interactions," in *Proc. the 4th International Conference on Hybrid Artificial Intelligence Systems*, Springer-Verlag, Berlin, Heidelberg, pp. 589-596.
- [3] M. Loderer, J. Pavlovičová, M. Föder and M. Oravec, 2014, "Data dimension reduction in training strategy for face recognition system," *IWSSIP 2014 Proceedings*, Dubrovnik, pp. 263-266.
- [4] J.-L. Wu and I.-J. Li, "A SOM-based dimensionality reduction method for KNN classifiers," in *Proc. 2010 International Conference on System Science and Engineering*, Taipei, pp. 173-178, 2010.
- [5] C. Grévisse, I. Muller, J. L. Laredo Jim énez, M. Ostaszewski, G. Danoy and P. Bouvry, 2014, "Visualization and classification of protein secondary structures using self-organizing maps," in *Proc. 2014 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making (MCDM)*, Orlando, FL, pp. 129-136.
- [6] R. Lasri, "Clustering and classification using a self-organizing MAP: The main flaw and the improvement perspectives," in *Proc. 2016 SAI Computing Conference*, London, United Kingdom, 2016, pp. 1315-1318.
- [7] P. Yodkhad, A. Kawewong, and K. Patanukhom, "Approximate nearest neighbor search using self-organizing map clustering for face recognition system," in *Proc. 2014 International Conference on Computer Science and Engineering*, Khon Kaen, 2014, pp. 151-156.
- [8] R. S. Pircoveanu, M. Stevanovic, and J. M. Pedersen, "Clustering analysis of malware behavior using self organizing map," in *Proc. 2016 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (CyberSA)*, London, 2016, pp. 1-6.
- [9] V. D. S. Almendra and D. Enachescu, "Using self-organizing maps for fraud prediction at online auction sites," in *Proc. 2013 15th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, Timisoara, 2013, pp. 281-288.
- [10] G. Abaei, Z. Rezaei, and A. Selamat, "Fault prediction by utilizing self-organizing map and threshold," in *Proc. 2013 IEEE International Conference on Control System, Computing and Engineering (ICCSCCE)*, Mindeb, 2013, pp. 465-470.
- [11] K. Kalaivani, C. Thirumaraiselvi, and R. Sudhakar, "An effective way of image compression using DWT and SOM based vector quantisation," in *Proc. 2013 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, Enathi, 2013, pp. 1-5.
- [12] W. Natita, W. Wiboonsak and S. Dusadee, 2016, "Appropriate learning rate and neighborhood function of self-organizing map (SOM) for specific humidity pattern classification over Southern Thailand," *International Journal of Modeling and Optimization*, vol. 6, no. 1, pp. 61-65.
- [13] Vanishree and R. K. V. Ramana, 2013, "Implementation of pipelined Sobel edge detection algorithm on FPGA for High speed applications," in *Proc. 2013 International Conference on Emerging Trends in Communication, Control, Signal Processing & Computing Applications (C2SPCA)*, Bangalore, pp. 1-5.
- [14] Z. Zhang and G. Zhao, 2011, "Butterworth filter and Sobel edge detection to image," in *Proc. 2011 International Conference on Multimedia Technology (ICMT)*, Hangzhou, pp. 254-256.
- [15] H. Y. Sun and S. X. Tian, 2011, "Image retrieval based on blocked histogram and Sobel edge detection algorithm," in *Proc. 2011 International Conference on Computer Science and Service System*, Nanjing, pp. 3277-3281.
- [16] H. N. A. Che, N. Jamil, S. Nordin and K. Awang, 2013, "Plant species identification by using Scale Invariant Feature Transform (SIFT) and Grid Based Colour Moment (GBCM)," in *Proc. 2013 IEEE Conference on Open Systems (ICOS)*, Kuching, pp. 226-230.
- [17] M. N. Hasan *et al.*, "Performance analysis of Canny's edge detection method for modified threshold algorithms," in *Proc. 2015 International Conference on Electrical & Electronic Engineering (ICEEE)*, Rajshahi, 2015, pp. 93-96.
- [18] Y. Zhang and X. B. Chen, "Application of ant colony algorithm in image features extraction and identification," in *Proc. 8th World Congress on Intelligent Control and Automation*, 2010, Jinan, pp. 873-877.
- [19] S. Jadon, M. Kumar, and Y. Rathi, 2015, "Face recognition using Som Neural Network with Ddct facial feature extraction techniques," in *Proc. 2015 International Conference on Communications and Signal Processing (ICCSP)*, Melmaruvathur, pp. 1070-1074.
- [20] P. K. Prajapati and M. Dixi, "Un-supervised MRI segmentation using self organised maps," in *Proc. 2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, Jabalpur, India, 2015, pp. 471-474.
- [21] C. Tao, L. Zhao, X. Su, and P. Ma, "PSO-based feature extraction for high dimension small sample," in *Proc. 2012 IEEE Fifth International Conference on Advanced Computational Intelligence (ICACI)*, Nanjing, 2012, pp. 229-233.
- [22] R. J. Mullen, M. N. Dorothy, and R. Paolo, "Ant algorithms for image feature extraction," *Expert Systems with Applications*, vol. 40, no. 11, no. 1, 2013, pp. 4315-4332.
- [23] L. I. Kuncheva and W. J. Faithfull, 2014, "PCA feature extraction for change detection in multidimensional unlabeled data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 1, pp. 69-80.
- [24] S. C. Ng, 2017, "Principal component analysis to reduce dimension on digital image," *Procedia Computer Science*, vol. 111, pp. 113-119.
- [25] G. Y. Aliyari, R. Frank, and M. H. Abrishami, 2015, "Fast incremental LDA feature extraction," *Pattern Recognition*, vol. 48, no. 6, pp. 1999-2012.
- [26] Y. Tan, Y. Zhao and X. Ma, "Contourlet-Based Feature Extraction with LPP for Face Recognition," in *Proc. 2011 International Conference on Multimedia and Signal Processing (CMSP)*, Guilin, Guangxi, pp. 122-125, 2011.
- [27] V. B. Shereena and M. D. Julie, 2015, "Significance of dimensionality reduction in image processing," *Signal & Image Processing: An International Journal*, vol. 6, no. 3, pp. 27-42.
- [28] J. Ye and Q. Li, "A two-stage linear discriminant analysis via QR-decomposition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 6, pp. 929-941, 2005.
- [29] R. D. Santo, *Principal Component Analysis Applied to Digital Image Compression*, Einstein, 2012, vol. 10, no. 2, pp. 135-139.
- [30] A. L. John and V. Michel, "Self-organizing maps with recursive neighborhood adaptation," *Neural Networks*, vol. 15, no. 8-9, pp. 993-1003, 2002.

- [31] P. S. Manu and D. R. Sharma, "Optimization of stochastic networks using simulated annealing for the storage and recalling of compressed images using SOM," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 10, pp. 2383-2396, 2013.



Sokchoo Ng received her bachelor's and master's degrees in computer science from Universiti Putra Malaysia in 2004 and 2007 respectively. She received her Ph.D degree from Institute of Advanced Technology, Universiti Putra Malaysia in 2013. Her research interests are in artificial intelligence, data mining and high dimensionality data analysis. She is a senior lecturer in the Faculty of Science, Technology, Engineering and Mathematics, International University of Malaya-Wales, Malaysia.



Mieowkee Chan completed her bachelor's and doctorate degrees in chemical-bioprocess engineering in Universiti Teknologi Malaysia. Her research interests are in membrane fabrication and separation, waste water treatment technology and haemodialysis. She is a senior lecturer in the Centre for Modelling and Simulation, Faculty of Engineering and the Built Environment, SEGi University, Malaysia.