NANYANG TECHNOLOGICAL UNIVERSITY

SELF-ORGANIZING CORTICAL PROCESSING WITH VISUAL FEATURE SELECTION FOR PATTERN RECOGNITION

NGUWI YOK YEN
SCHOOL OF COMPUTER ENGINEERING
2011

SELF-ORGANIZING CORTICAL PROCESSING WITH VISUAL FEATURE SELECTION FOR PATTERN RECOGNITION

NGUWI YOK YEN

School of Computer Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2011

Acknowledgements

I wish to record the deepest gratitude to my advisor, Dr. David Cho Siu-Yeung who offered prompt, wise, constructive feedback and advice despite his busy schedules. His thoughtful guidance, invaluable suggestions and constant encouragement has been and will continue to be the source of inspiration to me. I have learned a lot about how to be a researcher from his continual passion and serious attitude towards research for which I am truly grateful.

I would like to thank Nanyang Technological University and the School of Computer Engineering for granting me the research scholarship to pursue my PhD. I am also grateful to the staff and peers at the Centre of Computational Intelligence (C2I) who have made the centre a very good environment to do research and have helped me in different ways throughout the studies.

Lastly, I would like to thank my family for their continuous support, and encouragement.

Abstract

Pattern recognition has been studied extensively, and many algorithms have been established. It generally makes use of discriminant functions to learn the pattern in data. These discrimant functions are developed to be simplistic so as to warrant fast computations. In addition, simple evaluation functions are easier to learn because there are lesser parameters to estimate. However, this simplicity may not work well when new 'pattern' in data surfaces. Humans recognize an object or pattern from surrounding world in split second; however this involves many processing in the human visual system. Human gathers most of the sensory information through sight. Visual-perceptual processing covers approximately one-fourth of the cortex. Visual information processing is also the most complex, most studied sensory system of the brain.

It is envisaged that if the visual cortex can process information in such a lightning speed, there should exist some combinations of feature selection and pattern classification which are close enough to provide such capability. The motivation behind the research of this thesis is to establish a computational framework that attempts to emulate the visual cortical processing in the human brain. The aim is to recognize a pattern in short computation time even when sparse data is presented. Majority of the existing classifiers are only trained by datasets which are with balanced distribution (i.e. equal number of positive and negative samples). These classifiers will pose a problem when the data is imbalanced.

With these difficulties and motivation in mind, this thesis will unravel the establishment of a model in these directions. A Two-Tier Emergent Self-Organizing Map will thereby be proposed which have properties like incremental learning, discovery capabilities, and can adapt to the changing structures of data. The Two-Tier Emergent Self-Organizing Map achieves substantial improvements of accuracies over the convectional Self-Organizing Mapping models in terms of recognition rate and F-measure. Most recognition problems are solved through supervised

learning, whereas this work utilizes the topographic preserving, self-organizing, and emergent properties to resolve the recognition problem effectively.

Following that, we further model the human brain's cognitive process at the primary visual cortex to comprehend the region of interest in an image. The model consists of pre-cortical processing and cortical processing. Our pre-cortical processing of visual model encodes visual information using Gabor wavelet convolution and generalise the responses of on-centre and off-centre cells for the broad-band channels. There are six types of retinal and six types of LGN cells which are modelled using different Gabor wavelets, where each Gabor wavelet is applied to a particular spectral band. The outputs of these convoluted Gabor wavelets are a series of features that produce similar images as perceived by the visual pathway. The second stage (cortical processing) consists of two types of unit representing the cells in the primate striate cortex. The excitatory units represent spiny cells and the inhibitory one represent smooth cells. The adaptation process is achieved by the Two-Tier Emergent Self-Organizing Map. This thesis details how this model can fit into simulations like road sign recognition and emotion recognition. In the context of emotion recognition, the model is first trained as an Affect-Map (A-Map) that process positive and negative affects of human emotion. The second map called Emotion-Map (E-Map) is then established through the input from the Affect-Map. The results are encouraging; they are comparable to those of supervised-learning.

To demonstrate the capability of our model, we apply it for the problem of emotion profiling. This problem possesses the property of skewed distribution and is imbalanced in nature. Experimental results showed that the top rank features are in-line with the work of Wallbott and Scherer published in 1994 which approached emotions physiologically. Whilst the performance measures show that using the full features for classifications can degrade the performance, the selected features provide more convincing results in terms of accuracy and generalization.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Background

Human is the most complex living being in the world. The most distinctive features that separate human from other living beings are human intelligence, mental states, thoughts, and speech. The commander behind these is our brain. In the past several decades, human visual cortex has been the source of new theories and ideas about how the brain processes information. The visual cortex is easily accessible through a number of recording and imaging techniques. Several key ideas, such as input-driven self-organization, representing information on topographic maps, and temporal coding, originate from the mechanisms observed in the visual cortex. Understanding the computations in the visual cortex is therefore an important step towards a general computational brain theory.

Although computational theories of visual cortex have existed for about 30 years, it has been difficult to test these theories experimentally and computationally. In the last 10 years or so the situation has started to change, for two reasons. First, it has become technically possible to measure how the visual cortex develops in response to external input, and how visual functions

depend on low-level cortical mechanisms. Second, the available confluence makes it possible for the first time to constrain and test precise computational models about how the visual cortex develops and functions, and why it has the organization it does. Computational models have gradually become an integral part of neuroscience theory.

Only human can read human facial expressions well. Hence part of the endeavour of this work is to gel the model into facial expression recognition. Human expresses their mental states through facial expression, gesture and speech. A person may or may not unveil their inner mental states through several means such as facial emotion, voice, gaze, gesture, body language, posture and etc. Emotion and facial expression are sometimes confused. Emotion is the mental state of a human being, which may be reflected through facial expression or other means. Facial expression recognition deals with the classification of facial motion into defined classes that is based on visual information. However, some argue that the origin of all facial emotion is social context rather than emotion (Russell & Fernández Dols, 1997). As facial expression is assumed to change with respect to the occurrence of interaction between persons, it would seem more important to analyze human-to-human interaction precisely in spotting facial expression. Facial expression recognition has gained growing interest among the institutional researchers. This thesis is organised in such a manner from inspiration of cognitive model, to how human facial emotion recognition came about, its development and finally the system's performance analysis.

This research seeks to provide a framework that is inspired by our visual model for self-organizing the processing that emulates human-like performance to be able to handle different recognition problems. The effort concentrates on establishing a model that work in a similar way like our human visual model did in recognizing different objects. The motivation for this endeavour is that it is based on a neuroscience understanding of cognitive, auditory, visual and emotion signal interplay. The research will focus on the formulation of a neuro-cognitive framework of the brain-inspired systems for visual data understanding. The principles of the

model are described in details. Our perspective is to focus not only on the map-life structure of the cortex, but also take into account the dynamic processes that take place through lateral interaction and synchronization.

Comprehension of how emotion and visual model is related will draw us to entertain the possibility of having human behaviour and thought processes being replicated in machines based on connectionist networks models (Churchland & Churchland, 1990). Hence the scope of the research also includes the development of a computational model that emulates the visual cortex of human being in perceiving emotion and to explicate the cognitive processes that are controlled by the working memory and the cognitive learning processes from conscious awareness.

1.2 Motivations

Pattern recognition is a research area that studies the solution to recognize patterns in data or description of observations. The patterns associated with pattern recognition are not single instances. They are patterns of features that repeat across different samples. Recognition implies the act of associating a classification with a label. Strictly speaking, pattern recognition only tells a group of similar pattern posses certain similar characteristics that distinguish it from the other pattern. Pattern classification narrow this down to group this pattern with a name (label). This process generally takes two steps: feature extraction and classifications. The tools and methods of the field can be applied broadly. Feature selection is also known as variable selection, feature reduction, attribute selection or variable subset selection. It is a technique aimed at selecting a subset of relevant features for building robust learning model. The idea is that by removing the irrelevant and redundant features from data, feature selection helps to improve the performance of learning models by alleviating the curse of dimensionality, enhancing generalization capability, improve the learning time and the interpretability of model. Feature selection algorithm typically fall into two categories: feature ranking and subset selection. Feature ranking ranks the features by

a matrix and eliminates features that do not achieve sufficient score. Subset selection searches the set of possible features for optimal subset. Feature selection wrapper methods have been effective to eliminate noisy attributes, but are generally very slow because they apply the data mining algorithms repeatedly for different features as they follow certain search and stop criteria. The use of ranking method uses information gain measurement between individual attributes to give a ranking for the different features with reduced computation time.

This field has been studied extensively and many algorithms have already been established. It generally makes use of discriminant function to learn the pattern in data. These discrimant functions are developed to be simplistic so as to warrant fast computations. In addition, simple evaluation functions are easier to learn because there are lesser parameters to estimate. However, this simplicity may not work well when new 'pattern' in data surfaces. Human recognizes an object from surrounding world in split second; however this involves many processing in human visual system. Human gathers most of the sensory information through sight. Visual-perceptual processing covers approximately one-fourth of the cortex. Visual information processing is also the most complex, most studied and best understood sensory system of the brain.

The motivation behind this work is to establish a computational framework that emulates the visual cortical processing in human brain. The objective is to recognize a pattern in short computation time even when sparse data is presented. The aim of pattern classification models is to establish a working model that enables classification for any nature of data. However, majority of the classifiers are only trained by dataset which are with balanced or even distribution (i.e. equal number of positive and negative samples). These classifiers may have difficulties when the dataset is imbalanced. With these difficulties and motivation in mind, the thesis will unravel the establishment of a model in these directions.

1.3 Contributions

The major contributions of this thesis are listed as entries below:

1. Two-Tier Emergent Self-Organizing Map

The driving force behind SOM lies on the need to analyse data in a self adaptive and organized manner for reliability at the same time taking into account the inter-dependency among the data. The real world data are often complex and frequently updated, the implicit structures are hidden underneath the high dimensions of a data may have. A Two-Tier Emergent Self-Organizing Map is proposed which has properties like incremental learning, discovery capabilities and can adapt to the changing structures of data. The model is capable of handling large data up to one thousand datums. The concept of Emergent Self-Organizing Map originates from Ultsch (A. Ultsch & Mörchen, 2005). The Two-Tier Emergent Self-Organizing Map achieves substantial improvements of accuracies over other Self-Organizing Maps. Most recognition problems are solved through supervised learning, whereas this work utilizes topographic preserving, self-organizing, and emergent properties to resolve recognition problem effectively.

2. Self-Organizing Cortical Visual Processing Model

This model attempts to model human brain's cognitive process at the primary visual cortex to comprehend the region of interest in an image. The model consists of pre-cortical processing and cortical processing. The input intensity patterns perceived by human visual system are typically complicated functions of object surfaces and light sources in the world. Thus the visual system must be able to extract information from the input intensities that is relatively independent of the actual intensity values. Our pre-cortical processing of visual model encodes visual information using Gabor wavelet convolution and model the responses of on-centre and off-centre cells for the broad-band channels. There are six types of retinal and six types of LGN cells which are modelled using different Gabor wavelets, where each Gabor wavelet is applied to a

particular spectral band. The outputs of these LGN cells are a series of features that produce similar images as perceived by the visual pathway. The second stage (cortical processing) consists of two types of unit representing the cells in the primate striate cortex. The excitatory units represent spiny cells and the inhibitory one represent smooth cells. The adaptation process is achieved by the Two-Tier Emergent Self-Organizing Map. This thesis details how this model can fit into simulations like road sign recognition and emotion recognition. In the context of emotion recognition, the model is first trained as an Affect-Map (A-Map) that processes positive and negative affects of human emotion. The second map called Emotion-Map (E-Map) is then established through the input from the Affect-Map. The results are encouraging; they are comparable to supervised-learning.

3. Prototype Ranking for Imbalanced Data

Protype ranking is a subset of feature selection. The prototypes are batch trained together with pre-specified criterion. It works in a similar manner as survivor of the fittest. The features are eliminated one by one. The remaining values of surviving features continue to be recursively ranked till all criterions are met and all rankings are finalised. The prototype rankings are derived from Support Vector Machines and based on weight vector sensitivity with respect to a prototype. It was initially proposed by (Guyon, Weston, Barnhill, & Vapnik, 2002a) for selecting genes that is relevant for a cancer (Ahumada, Grinblat, Uzal, Granitto, & Ceccatto, 2008) classification problem. This is to complement the learning of classifiers to shrink down the number of prototypes that need to be processed and hence reducing the computation time and alleviates the skewness effect that is observed in imbalanced data. This work has been extended into the problem domain of medical imbalanced dataset and has successfully pinpointed the important prototypes that help in diagnosing illness.

4. Emotion Profiling and Understanding

Implementation of algorithms on Prototype Ranking and Self-Organizing Learning into the problem domain of emotion profiling and understanding would be proposed in this thesis. Experimental results showed that the top rank features are in-line with the work of Wallbott and Scherer published in 1994 which approached emotions physiologically (Scherer & Wallbott, 1994). Whilst the performance measures show that using the full features for classifications can degrade the performance, the selected features provide more convincing results in terms of accuracy and generalization.

1.4 Thesis Outline

This thesis consists of 8 chapters. The thesis begins with Chapter 2 where the literature reviews on self-organizing maps are presented which forms the foundation of this work. Here, the origins of self-organizing maps from Von Malsburg and Willshaw's Self-organizing Model to Kohonen SOM are introduced. We then proceed to introduce the measurements in SOM like topographic products, topographic function, topographic error. The various extensions of SOM like Adaptive Subspace SOM, Visualization induced SOM, Growing Hierarchical SOM and Self-Organizing Map Structured Data will be briefly discussed. The other part of this chapter is about feature ranking. Feature ranking is an important aspect to optimise the learning performance.

In Chapter 3, we introduce the concept of Two-Tier Emergent Self-Organizing Map (TtEsom). The TtEsom is evolved from an extension of SOM, Emergent Self-Organizing Map. This variant of SOM allows the emergence of intrinsic features of high dimensional data map onto a two dimensional map. The structure of TtEsom, the adaptation between the layers and its convergence are described in this chapter.

Chapter 4 investigates a cognitive-based Emergent Self-Organizing Visual Processing Model (ESOVPM). The framework emulates the neuro-cognitive structure of human visual processing pathway and topographic maps. Within this brain inspired framework, the object

detection and features extraction blocks are built to model the processing taken place from visual pathway. We then demonstrate how this model can fit into simulations like road sign recognition and emotion recognition. In the context of emotion recognition, the model is first trained by the Affect-Map that process positive and negative affects of human emotion. The second map called Emotion-Map is then established through the input from Affect-Map.

Chapter 5 illustrate the concept of Prototype Ranking based on Support Vector Machine to solve the problem of imbalanced dataset. The resultant of prototype ranking is a series of good features which can greatly enhance the learning and significantly reduce computation time. This chapter walks through some background of the statistical learning theory and support vector machine. The algorithm of prototype ranking is then presented.

Chapter 6 focuses on an important challenge in machine learning community, i.e. imbalanced dataset. As traditional machine learning algorithm may be biased towards majority class, thus producing poor predictive accuracy over the minority class. In this chapter, we introduce the problem domain and formulation. The analysis to discover the problems underlying imbalansed data will be performed. We then utilize the prototype ranking framework as described in Chapter 5 to solve the problem of imbalanced data under medical domain.

In Chapter 7, we adopted a case study with questionnaire based emotion profiling. We propose to implement our algorithm of Prototype Ranking with Self-Organizing Learning into this problem domain. Experimental results showed that the top rank features are interpreted in-line with the work of Wallbott and Scherer published in 1994 which approached emotions physiologically. Whilst the performance measures show that using full features for classifications would degrade the performance, selected features provide more convincing results in terms of accuracy and generalization.

Finally, Chapter 8 concludes the research described in this thesis and further studies.

Chapter 2 Literature Review

2.1 Self-Organizing Map

Self-Organizing Map concerns the formation of a topologically ordered map from signal space onto neural network by a special kind of adaption. Kohonen described this adaptation as regression (Kohonen, 1995) which involves fitting a number of ordered discrete reference vectors to the distribution of vector input samples. The 'feature maps' formed can be displayed to project and visualize high dimensional signal spaces onto one-dimensional or two-dimensional lattice-like structure. This chapter provides background of SOM and how the Malsburg's model and Kohonen's model relates. The properties and theories of SOM are presented in the second section. We will walk through SOM's generic algorithm, the learning and measures for topographic preservations. The third section discusses different variants of SOM which include Adaptive Subspace SOM (ASSOM), Visualization Induced SOM (ViSOM), Growing Hierarchical SOM (GHSOM), and Self-Organizing Map Structured Data (SOM-SD).

2.1.1 Biological Background

Physiological study has shown that human brain is organized in many places in such a way that different sensory inputs are represented by topologically ordered computational maps (Knudsen,

du Lac, & Esterly, 1987). In particular, sensory inputs such as tactile (Kaas, Merzenich, & Killackey, 1983), visual (Hubel & Wiesel, 1962b), and acoustic (Suga, 1985) inputs are mapped onto different area of the cerebral cortex in a topologically ordered manner. The spatial locations of the neurons on these topographic maps are indicative of statistical features of input patterns. These biological sensory systems are often organised in such a way that stimuli with properties that are close to each other often activate nearby cells. A good example of this is the visual system. At a very basic level, objects that are spatially adjacent in the visual world activate photoreceptors that are also adjacent to each other, and these in turn activate neurons with the cortical visual system.

Experimental evidence has shown that there exist two kinds of topographically organized computational maps in primary visual cortex (Blasdel & Salama, 1986; David & Torsten, 1974; Hubel & Wiesel, 1962b):

1. The maps of preferred line orientation which corresponds to the angle of tilt of a line stimulus.
2. The maps of ocular dominance which corresponds to the relative strength of excitatory influence of each eye.

The renown scientist David Hubel won the Noble prize with his colleague Torsten Wiesel for their discoveries in the mammalian visual system (Hubel & Wiesel, 1962b). In his experiment, a flashed or moving bar of light is projected onto a screen. This is the stimulus that is being presented to the cat. The activity of a neuron recorded from the cat's brain can be heard. A cat has been anesthetized and placed in front of the screen, with its eyelids held open. The tip of a tungsten wire has been placed inside the skull, and lodged next to a neuron in a visual area of the brain. Although the cat is not conscious, neurons in this area are still responsive to visual stimuli. The tungsten wire is connected to an amplifier, so that the weak electrical signals from the neuron can be recorded. The response of the neuron consists of brief clicking sounds due to spikes in the waveform of the electrical signal from the neuron. Almost without exception, such spikes are

characteristic of neural activity in the vertebrate brain. The frequency of spiking is dependent on the properties of the stimulus. The neuron is activated only when the bar is placed at a particular location in the visual field. Furthermore, it is most strongly activated when the bar is presented at a particular orientation. It was that observation that led to the discovery that cortical neurons were most sensitive to oriented stimuli like edges or bars. They found that in visual area 1 of the mammalian brain neurons were tunes to the "orientation" of a stimulus. That is, if a grating of alternate dark and light lines is presented, the cells respond most strongly when they are orientated in a particular way. The researchers found that cells in the cortex were organized topographically with each adjacent cell having a slightly different preferred orientation.

2.1.2 From Von Malsburg and Willshaw's Self-organizing Model to Kohonen SOM

The idea of self-organizing map in the field of artificial neural networks may be traced back to the early work of Malsburg (Von Der Malsburg, 1973; Willshaw & Von Der Malsburg, 1976) in 1970s. They established Winner-Take-All (WTA) behavior where only a single winner is selected. They studied the self-organization of orientation sensitive nerve cells in the striate cortex. The first paper on the formation of biological inspired self-organizing maps was jointly published by Willshaw and Malsburg in 1976 (Willshaw & Von Der Malsburg, 1976). The Willshaw and Malsburg model explains the problem of retinotopic mapping from retina to visual cortex. One lattice represents presynaptic (input neuron) is projected onto the other lattice which represents postsynaptic (output) neurons. The basic idea of Willshaw and Malsburg model is for geometric proximity of presynaptic neurons to be coded in the form of correlations in their electrical activity and to use these correlations in the post-synaptic lattice so as to connect neighboring presynaptic neurons to neighboring post-synaptic neurons as shown in Figure 2-1(a). The pre-synaptic neurons make up the first lattice. A topologically ordered mapping is thereby produced through self-

organization of neurons. The states of nodes in cortical sheet represent activation rates in localized neuronal populations with a leaky integrator dynamics. Population models were derived by Wilson and Cowan (Wilson & Cowan, 1972) with separate excitatory and inhibitory populations, and subsequently studied with centre-surround architectures (Wilson & Cowan, 1973). Amari (Amari, 1977) further abstracted these models by combining the excitatory and inhibitory populations, and by using a continuous descriptions of the neural sheet. Malsburg and Willshaw (Willshaw & Von Der Malsburg, 1976) reached four conclusions as follows:

1. Maps develop in a step-by-step, orderly fashion.
2. Lateral connections within the map are initially widespread but during the learning process, they decay.
3. The orientation of map is formed during the self-organizing process, but the final pattern of connections take a longer time to develop.
4. Appropriate starting conditions are essential to the formation of meaningful map.

While the model of Malsburg and Willshaw are useful in exploring the biological implications of self-organization, they are not useful to act as a generalized learning tool. The motivation of their model is to model biological activity, they sacrifice computational efficiency in order to maintain consistency with observed biological structure.

In 1982, Kohonen's paper on Self-Organizing Feature Mapping (Kohonen, 1982) attracts much more attention than Willshaw and Malsburg's model in the literature. Kohonen's SOM model is structurally similar to that of Malsburg's: a set of neurons, each of which is connected to every other neuron through a connection of predetermined strength, and to a set of inputs via a set of initially random weights. The novel aspect of Kohonen's algorithm is the key change to the lateral interaction functions. This change significantly improves computational efficiency of the self-organization process. Kohonen's innovation was to replace the iterative process of reinforcement in Von der Malsburg's with algorithmic selection. Rather than having a single

winner (WTA), Kohonen's algorithm artificially select a winner according to some distance metric, and a second lateral interaction kernel (Eg. Gaussian) is artificially fitted about this winner. Neurons near the winner (neighbours) receive strong support as a result of the large value of the plasticity control kernel, while other neurons receive little or no support. In this way, the winner and its neighbours are reinforced.

Kohonen (Kohonen, 1990) describes SOM as a nonlinear, ordered, smooth mapping of high-dimensional input data manifolds onto the elements of a regular, low-dimensional array. The Kohonen model generates a mapping of input space onto a discrete lattice with pre-specified number of neurons as shown in Figure 2-1(b). The output can be in one dimensional or two dimensional. Figure 2-1(b) displays the two dimensional 5x4 lattice, all neurons are presented with the same input vector, and all neurons compute its distances between their synaptic weights. The neuron with the closest distance to the input vector produces an active output. The map is generated by establishing a correspondence between input and output neurons while maintaining a topological relationship that faithfully models the input space. The Kohonen model can be based on Winner-Takes-All (WTA) or Winner-Takes-Most (WTM) algorithms. The winner in WTA is simply the winning neuron which is the activated neuron during competition. WTM differs from WTA in that more than one neuron adapts their synaptic weights in one learning iteration. The close neighbourhood neurons of the winner are updated instead of only update the winner in the case of WTA. The further the neighboring neuron, the smaller the weight modification. The winner has weights which are most correlated to current input vector. The low-dimensional topological representation of high-dimensional dataset is achieved by codebook feature vector through competitive and unsupervised learning.

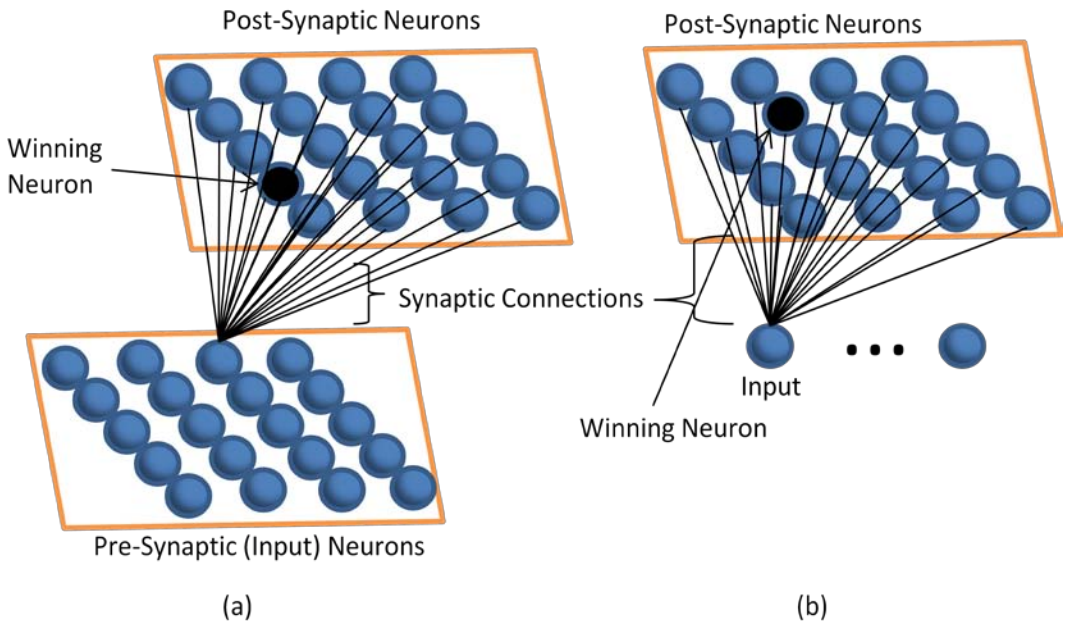


Figure 2-1 Two pioneering models in self-organizing maps: (a) Willshaw-von der Malsburg Model (b) Kohonen Model

2.1.3 SOM Generic Algorithm

The Self-Organizing Maps (Somers *et al.*) undergoes unsupervised competitive learning. The SOM projects high dimensional space onto grid-like lower dimensional space (1D or 2D) in an orderly fashion. The neurons on the map are randomized at the beginning. The input may be presented to the map one after another or by batch. After learning, the similar input patterns which are near to each other in the input domain are correspondingly mapped to nodes near each other in the output map, which is referred to as topology preservation.

Let $X=[x_1,x_2,x_3]^T$ denotes input vector with n dimensions. And $w_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T$ denote the corresponding synaptic weight of neuron i. Let $c_{i,j}$ denote the lateral feedback of node i and j is the boundary of lateral interaction. The map with L output is represented by $y_1, y_2, \dots, y_L$. Kohonen (Kohonen, 1989) describes the output to be obtained with the following equation:

$$y_i = \emptyset\left[I_i + \sum_{j=-J}^{J} c_{ij}\, y_{i+j}\right], \qquad i = 1,2,\dots,L \tag{2.1}$$

where ϕ is the sigmoid transfer function. c_{ij} represents the strength of the connection between i and j. When $c_{ij} < 0$, the connection is inhibitory. On the other hand, the connection is excitatory when $c_{ij} < 0$. The I_i represents the total external control exerted on node i by the weighted input,

$$I_i = \sum_{l=1}^{n} w_{il} x_l \tag{2.2}$$

Kohonen's simulations show that a node c with the largest I_c together with its neighbours tends to concentrate inside a spatially bounded cluster, while the outputs y_i of other nodes tend to be zero. The cluster is centred at the node with the largest response. The lateral range of cluster depends on the ratio of the excitatory and inhibitory interconnections. The adaptation of weights updates the nodes in the activity range,

$$\frac{dW_i}{dt} = \alpha(t) y_i X \tag{2.3}$$

where $\alpha < \alpha(t) < 1$ is the learning rate for input X.

The above mentioned algorithms can be organized as follows:

Step 1 – Initialization: The topology and size of network are determined. The weights of the neuron $w_i(0) = [w_{i1}, w_{i2}, \dots, w_{in}]^T$ are initialised with small random values where i denotes the index of neuron at iteration *0*. The initial learning rates $\alpha(0)$ are specified.

Step 2 – Competition: For each input vector or draw a sample input (this step is sometimes known as sampling) $\alpha(t)$, compute the distance from all neurons. The neuron with the minimum Euclidean distance or largest inner product $w_i x$ is the winning neuron or best matching unit (BMU) denoted by:

$$c = \arg\min_i \left\| x(t) - w_i \right\|, i \in \{1,\dots,n\} \tag{2.4}$$

Step 3 – Co-operation: The winning neuron denotes the centre of topological neighbourhood and determines the spatial location of topological neighbourhood of excited neurons. The neighbourhood function $h_{i,c}(t)$ decreases gradually during the learning process.

$$h_{c,i}(t) = \exp\left[-\frac{\|r_i - r_c\|^2}{2\sigma^2(t)}\right] \tag{2.5}$$

Where $\|r_i - r_c\|^2$ is the lateral distance between winning neuron c and excited neuron i and $\sigma(t)$ is the width of neighbourhood function that decreases monotonically with respect to iteration i.

$$\sigma(t) = \sigma_0 \cdot e^{-\frac{t}{\tau}log\,\sigma_0} \tag{2.6}$$

where σ_0 is the initial width, τ is a time constant that is set to be the maximum number of iterations.

Step 4 – Adaptation: This step allows the output to be self-organized and form a feature map between inputs and outputs. Adaptation is achieved by adjusting the synaptic weight vectors of all neurons according to the weight-updating rule:

$$w_i(t+1) = \begin{cases} w_i(t) + \eta(t)h_{c,i}(t)[x(t) - w_i(t)] & for\ i \epsilon N_c \\ w_i(t) & otherwise \end{cases} \tag{2.7}$$

$\eta(t)$ is the learning rate which decreases monotonically with respect to iteration t .

$$\eta(t) = \eta_0 \bullet \exp\left(-\alpha\frac{t}{\tau}\right) \tag{2.8}$$

where η_0 is the initial learning rate and α is the exponential decay constant. At the beginning, SOM tries to organize itself globally and subsequently it performs more local organization when the learning rate and neighbourhood get smaller.

Step 5 – Convergence: The convergence of network occurs when the changes in neurons' weights are asymptotically small or the pre-specified iteration is reached.

2.1.4 Learning Algorithms

Kohonen's SOM follows two-stage learning: an initial phase of learning, followed by offline learning where no new information is learned. The initial phase of learning relates to all the initialization needed and the start of learning using these parameters. At this stage, we initialize the initial random weights, set the desired number of iterations, learning rate, number of neurons and etc. The first phase of learning adapts the weights of neuron together with the default values prior to subsequent adaptation. The two stage approach is enforced through decreasing (decaying) values of parameters. When the parameters are zeroed, the learning remains static and no more adaptation of new training data.

The learning algorithm in Kohonen's SOM is based upon the repeated application of updating rule over the training data set. The weights of the map are seeded with small random values. The parameters α and σ are given initial values and decrease over the predefined schedule. During each epoch, a random order of presentation is chosen, and each pattern is presented to the updating mechanism in this order.

Algorithm 2.1: SOMLearn

W=SOMLearn(n,m,e):

1. Assign initial value of α and σ
2. Initialize W with array of α x m neurons, each neuron contains J weights.
3. Initialize all J in W with small random values.
4. For $t = 1$ to e (max iteration) do
   - Initialize random ordered map of presentation R
   - for $t = 1$ to I (total neurons) do
     - W_x = weights of neurons x input
   - end for
   - $W = \max(W_x)$
   - for $i = 1$ to I (total neurons) do
     - Update the weights W
   - end for
   - end for
   - return W

The traditional SOM adapts neurons radially around the stimulating input x_i. At t iteration, the weights of the neurons are updated by:

$$w_r(t+1) = w_r(t) + \Delta w_r(t) = w_r + \eta_r[x_i - w_r(t)] \quad (2.9)$$

where $\eta_r(t)$ denotes the learning rate for neuron r and is decreased gradually.

2.1.5 Topological Ordering

The feature map Φ computed by the SOM algorithm is topologically ordered in the sense that the spatial location of a neuron in the lattice corresponds to a particular domain or feature of input patterns. Approximate the input space k by pointers or prototypes in the form of synaptic weight vectors w_j in such a way that the feature map Φ provides a faithful representation of the important features that characterize the input vectors $x \in \kappa$ in terms of a certain statistical criterion.

Kohonen's algorithm (Kohonen, 1982) defines a SOFM from data manifold M embedded in a d-dimensional input space $\Re^d$ onto a d_A-dimensional lattice $A$of neurons. Synaptic weight vector w_i is assigned to each node i of A defining the Voronoi polyhedron V_i of each unit i by the set of all data points $v \in M$ which are matched best by this reference vector. This mapping from data manifold M onto the lattice A is called topology preserving, if neighbouring units i have Voronoi polyhedron V_i adjacent to M. The topology preserving property of the SOFM makes it possible to exploit the similarity relations of the input space.

Definition 1: Let $X \subset \Re^n$ be a given manifold and $W = \{W_1, W_2, W_3, \ldots, W_L\}, W_i \neq W_j$, for $i \neq j$, $W_i \in \Re^n, i, j = 1, 2, \ldots, L$. The Voronoi polyhedron is a set given by (T. Martinetz, 1993):

$$V_i = \{X \in R^n \mid \|X - W_i\| \leq \|X - W_j\| \text{ for } j \neq i, i, j = 1, 2, \ldots L\}. \quad (2.10)$$

Kohonen defined a measure of the degree of ordering of weights, called index of disorder as follows:

$$D = \sum_{i=2}^{L} | w_i - w_{i-1} | - | w_L - w_1 | \tag{2.11}$$

The equality holds only if all the weights form a monotonic sequence in ascending or descending order. An one-dimensional SOFM is topologically ordered if $D = 0$.

Lo and Bavarian (Lo & Bavarian, 1991) defined SOFM of arbitrary dimensionality to be topology ordered if the followings hold:

$$\| X - W_a \| < \| X - W_b \| \tag{2.12}$$

$$\| r_c - r_a \| < \| r_c - r_b \| \tag{2.13}$$

where c is the index of winning neuron, a and b are the indexes of any two nodes in the output map. r_a, r_b, r_c are locations of the corresponding nodes in the output map. They further prove that SOFM converge to a topologically ordered configuration if the neighbourhood function is monotonically decreasing.

However, Erwin (Erwin, Obermayer, & Schulten, 1992) proves that Lo's convergence proof are not absorbing, the topology ordered configuration could become disordered again. He had shown that the one-dimensional SOFM using any monotonically decreasing neighbourhood function and a constant learning step size can be guaranteed to converge to an ordered mapping. The convergence time depends on the neighbourhood function.

To quantify the performance of SOFM, it is important to define a measure that can determine the degree of topology preservation. However, different definition of topology preservation may lead to different measures. The first formal definition of topology preservation is proposed by Martinetz and Schulten (T. Martinetz & Schulten, 1994). They defined a topology

preserving feature map is determined by a mapping Φ from a manifold $M \subseteq \Re^D$ onto the neuron i. The mapping from M to A is determined by pointers $w_i \in \Re^D, i = 1,2,\dots,N$ attached to the vertices i. The output map A forms a topology preserving map of M if and only if the mapping Φ from M to A as well as the inverse mapping Φ^{-1} from G to A is neighbourhood preserving.

2.1.6 Topographic Product (TP)

There are several measures to quantify the degree of topology preserving. The first known measure was proposed by Bauer and Pawelzik (Bauer & Pawelzik, 1992) called topographic product. It measures the preservation of neighbourhood between the neuron in A and their reference vectors w_i lying on M. However, the topographic product does not consider the neighbourhood relations of reference vectors lying on M, but only the neighbourhood relations of the reference vectors within the embedding space V.

The topographic product (S. Lin, 1997),

$$\mathrm{P} = \frac{1}{N(N-1)} \sum_{j=1}^{N} \sum_{k=1}^{N-1} \log\left(\prod_{l=1}^{k} Q_1(j,l) Q_2(j,l) \right)^{\frac{1}{2k}} \tag{2.14}$$

$$Q_1(j,k) = \frac{\left\| \overrightarrow{w_j} - \overrightarrow{w_{n_k^X(j)}} \right\|^2}{\left\| \overrightarrow{w_j} - \overrightarrow{w_{n_k^M(j)}} \right\|^2} \tag{2.15}$$

$$Q_2(j,k) = \frac{\left\| \overrightarrow{x_j} - \overrightarrow{x_{n_k^X(j)}} \right\|^2}{\left\| \overrightarrow{x_j} - \overrightarrow{x_{n_k^M(j)}} \right\|^2} \tag{2.16}$$

Where $n_k^M(j)$ denotes the index of kth nearest neighbour of neuron j in the map space, and $n_k^X(j)$ denotes the index of kth nearest neighbour of neuron j in the input space. The ratio $Q_1(j,k)$ and $Q_2(j,k)$denotes a comparison between nearest neighbours in the map space and the input

space respectively. The topographic product of 1 denotes the input space coincide with output space.

The problem with TP as pointed out by (Villmann, Der, & Martinetz, 1994) is its inability to distinguish between the folding of the map along nonlinearities in the data space and folding within the data space itself. The team (Villmann, Der, Herrmann, & Martinetz, 1997) later proposed another measurement approach called topographic function (TF) as described following section.

2.1.7 Topographic Function (TF)

The topographic function compares the neighbourhood relationship between receptive fields on the map. The receptive field of a neuron M is the range of patterns in the input space which will cause M to be selected as a winner. On a well ordered map, only units that are neighbours on the map may have adjacent receptive fields. The topographic function (Thomas, Ralf, Herrmann, & Thomas, 1994) is defined as:

$$\Phi_X^M(d) = \sum_i f_i(d) \tag{2.17}$$

$$f_i(d) = \#\{j \mid \|i-j\|_{\max} > d_i R_i \cap R_j \neq f_i 0\} \tag{2.18}$$

The function $f_i(d)$ determines the number of units j which have receptive field R_i adjacent to R_j while at the same time have a distance on the map larger than d. $\#\{X\}$ is the cardinality of the set X, and the maximum norm $\|x\|_{\max} = \max_i |x_i|$. The topographic function is then computed for d values in the range $[1, \ldots, I]$.

2.1.8 Topographic Error (TE)

The topographic error was proposed by Kiviluoto (Kiviluoto, 1996). The proportion of samples for which the nearest and second nearest units reside in non-adjacent positions on the map is first calculated. The topographic error is defined as:

$$\varepsilon_t = \frac{1}{N}\sum_{i=1}^{N} u(\vec{x}_i) \tag{2.19}$$

$$u(\bar{x}) = \begin{cases} 1, & iff\ winner\ and\ nearest\ units\ for\ x\ are\ non-adjacent \\ & otherwise \end{cases} \tag{2.20}$$

This measure does not consider the extent of discontinuities. Given two similar points in the input space, there is no difference between mapping them one neuron apart, or to opposite corners of the map. Kiviluoto claimed that the TE allows for distinction between a small number of major discontinuities and a large number of minor discontinuities.

2.1.9 Variants of SOM

2.1.9.1 Adaptive Subspace SOM (ASSOM)

The ASSOM (Bailing, Minyue, Hong, & Jabri, 1999; Kohonen, 1996; Kohonen, Kaski, & Lappalainen, 1997) creates a set of local subspace representations through competition and cooperative learning. Conventional SOM organize the neurons to partition the input space. In ASSOM, a number of sub-models are topologically ordered, with each sub-model responsible for describing a specific region of the input space by its local principal subspace. The neuron in conventional SOM is replaced by module consisting of linear input layer and a quadratic neuron. The input pattern is compared with the signal subspace represented by the module. During the training, different feature filters emerge and be tuned to different low-dimensional manifolds. The ASSOM starts with grouping input vectors as episodes and present to the network.

The training algorithm proceeds as follow:

Step 1 – Find the winner with maximum projection energy.

$$c = \arg\max_i \left\{ \sum_{t_p \in S(t)} \left\| \hat{x}^{(i)}(t_p) \right\|^2 \right\} \tag{2.21}$$

Step 2 –Rotate the basis vectors of the winner unit and its neighborhood.

$$b_h^{(i)}(t+1) = \left[1 + \lambda(t) h_c^{(i)}(t) \frac{x(t_p) x(t_p)^T}{\left\| x^{(i)}(t_p) \right\| \left\| x(t_p) \right\|} \right] b_h^{(i)}(t) \tag{2.22}$$

Step 3 – Dissipate the components of basis vector.

Step 4 – Orthonormalize the basis vectors of each module.

2.1.9.2 Visualization Induced SOM (ViSOM)

The ViSOM (Yin, 2002) is a visualization method that regularizes the inter-neuron distances such that the inter-neuron distances in the input space resemble those in the output space after the completion of training. This feature can be useful to some applications because it is able to preserve the topology information as well as the inter-neuron distances. This characteristic is attributed to the output topology pre-defined in a regular 2-D grid so that the trained neurons are almost regularly distributed in the input space. The ViSOM delivers better data visualization compared to conventional SOM and other visualization methods.

The ViSOM uses a similar grid structure of neurons to the SOM. Each node, indexed $c_{i,j}$ with associated weight vector $w_c = \left[w_{c1}, w_{c2}, ... w_{cn} \right]^T$ of n dimensions. The algorithms are as follows:

Step 1 – Find the winner using the minimum distance in map space Ω

$$c = \arg\min_{c \in \Omega} \left\| x(t) - w_c \right\| \tag{2.23}$$

Step 2 – Updates the weights of winner

$$w(t+1) = w(t) + \alpha(t)[x(t) - w(t)] \tag{2.24}$$

where $\alpha(t)$ is the learning rate at time t.

Step 3 – This step and the next step differ from traditional SOM because it decomposes the updating force into two components u, v. The neighbourhood weights are updated as follows:

$$w_k(t+1) = w_k(t) + \alpha(t)\eta(v,k,t)\{[x(t) - w_v(t)] + [w_v(t) - w_k(t)]\frac{(d_{vk} - \Delta_{vk}\lambda)}{\Delta_{vk}\lambda}\} \tag{2.25}$$

where d_{vk} and Δ_{vk} are the distances between neurons in data space and on the map. λ affects the resolution of the map.

The key feature of ViSOM is on the distances between neurons on the map reflect the corresponding distances in the data space. The mapped data points on the map resemble approximately those in the original space which makes visualization more direct and quantitatively measurable.

2.1.9.3 Growing Hierarchical Self-Organizing Map (GHSOM)

The main idea behind GHSOM (Rauber, Merkl, & Dittenbach, 2002) is to represent different layers of SOM in a hierarchical manner where each layer consists of a number of independent SOM. Each unit in the map can be added to the next layer of the hierarchy. The architecture starts from one map, and subsequently expanded if the mapping unit has high quantization error q_i above threshold τ_2. Figure 2-2 shows the architecture of typical GHSOM. The root node starts with one neuron and the map at level 1 consists of 2x2 units, each unit further expands to level 2.

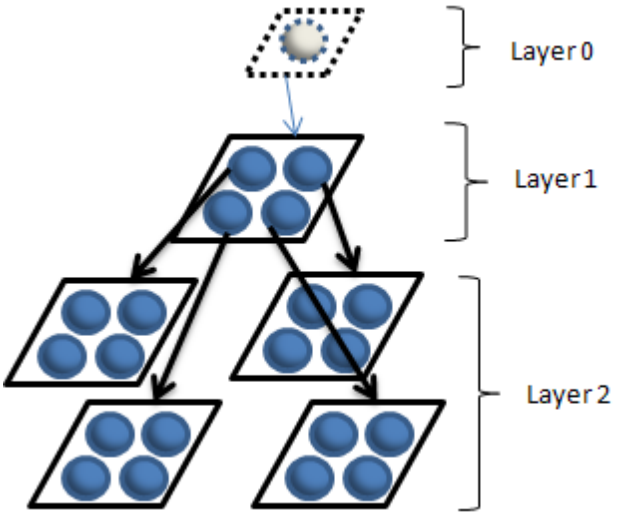


Figure 2-2 Architecture of GHSOM reflecting the hierarchical structure of the input data

The steps of GHSOM are as follows:

Step 1 **- Initialization**: The first layer map is initially set to 2x2 map and its model vectors have random initial value. The training starts from mapping a subset C_i of input vector x_j onto unit i. The mean quantization error of first layer is found using the following equation:

$$mqe_i = \frac{1}{n_c}.\sum_{x_j \in C_i} \left\| m_i - x_j \right\|, \tag{2.26}$$

$$n_c = \left| C_i \right|, C_i \neq 0 \tag{2.27}$$

where the n_c input vectors x_j are elements of the set of input vectors C_i.

Step 2 **- Training and Growth**: The training process is the same as conventional SOM training procedure. After λ training iterations, the quantization errors are analyzed.

Step 3 **- Termination of Growth Process**: The training process continues until all units satisfy the global stopping criterion which is defined as:

$$mqe_m < \tau \bullet \mathcal{R}_u \tag{2.28}$$

where qe_u is the quantization error of the corresponding unit u in the upper layer.

The training and growth process results in lower hierarchies with detailed refinements presented at each subsequent layer or deeper hierarchies which provide a stricter separation of various sub-clusters by assigning separate maps.

2.1.9.4 Self-Organizing Map Structured Data (SOM-SD)

The SOM-SD (Hagenbuchner, Sperduti, & Ah Chung, 2003) is another extension of SOM for recursive processing of tree-structured data. Structured data are jointly and severally related to each other according to specific modalities. The data are represented in structured tree manner. The parent nodes contain features information and child information. The outputs of leaf node are used as another input for the map, the coordinates of the winning neuron acts as pointer within a parent node. Figure 2-3 shows the sequence and mapping of a simple tree with 3 nodes. First, the winner neurons of two leaf nodes are found on the map, the positions of the winner neurons are then used together with root nodes features for the input representation of root node. Other nodes are processed in a similar fashion. The training algorithm is similar to that of generic SOM, with slight differences as follows:

Step 1 – **Competition**: The closest weight vector for input $v(t)$ at iteration t is selected as

$$c = \arg \min_{c_i} \left\| \Lambda(v(t) - w(t) \right\| \tag{2.29}$$

where Λ is a $(m + 2c) \times (m + 2c)$ diagonal matrix that is used to balance the importance of the label versus the importance of the pointers. Both the input vector and weight vector share the same dimension of $m + 2c$. c is the maximum number of children nodes and m is the dimension of current node on the map under training.

Step 2 – **Cooperation**: This step is to attract winning neurons and its neighborhood to move closer to input vector $v(t)$. The subsequent steps include gradually decrease the learning rate and neighborhood radius.

The computation is quite expensive due the updating of coordinates for different sub-graph prior to training. Another drawback is its difficulty to visualize the trained map.

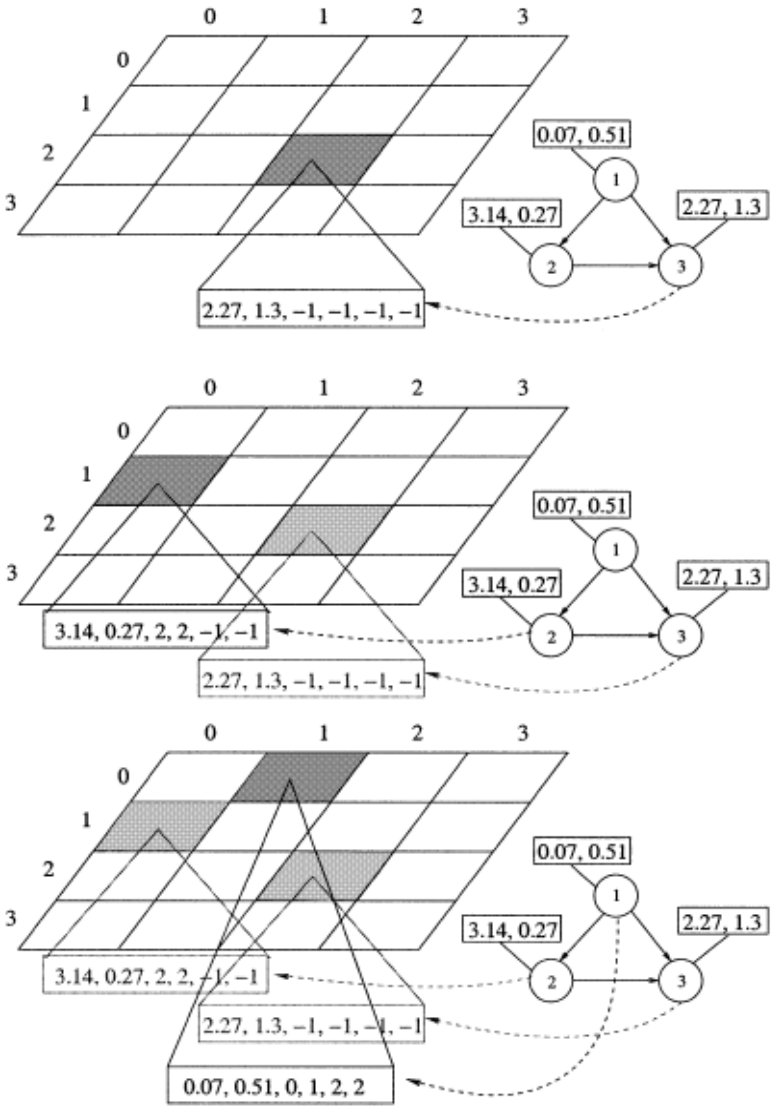


Figure 2-3 Example of SOM-SD. Null coordinates are represented by (-1, -1). Nodes are being mapped from leaf node to root node. The parent node 1 consists of the children's coordinates (Hagenbuchner et al., 2003)

2.2 Feature Ranking

Feature ranking is an important technique for data understanding as well as analysing the relevant features associated with it. To optimise the learning performance, it is desirable to discard irrelevant features prior to learning and provide only meaningful features. Features ranking is a relaxed formalization of feature selection. In a feature ranking, one selects the top ranked features, where the number of features is specified by user or analytically determined. For proper predictions to be made on test data sets, various feature ranking methods can be used to gain insights into the data (Guyon *et al.*, 2008). The approaches used for evaluation of relevant features include wrapping methods, classification based methods as well as general method (Guyon et al., 2002a). Feature selection can be formalised as a combinatorial optimization problem by finding the feature set that maximizes the quality of the hypothesis learned from these features. Wrapping methods are such global approach to the optimization problem. Classification methods can be based on statistical characteristics such as correlation coefficients. Statistical based approaches such as Support vector Machines (SVMs) are widely used for ranking various features to achieve the expected optimum knowledge on the data being investigated (B. Boser, I. Guyon, & V. Vapnik, 1992). Support vector machines are important for complex data classifications due to their ability to identify separating hyperplanes between two distinct classes of data with maximum margins (Hsu & Lin, 2002).

2.2.1 Applications

One of the most practical aspects in applying features ranking is to assist user in making a purchasing decision. In traditional product design setups, effective product design depends on the artistic capabilities of a designer which in most cases did not give the expected results for a competitive advantage in the market (Duan, Rajapakse, Wang, & Azuaje, 2005). Great amount of research have been conducted to get a deeper understanding of the end user preferences for

development of competitive product designs that meets the market standards. Despite the various techniques, methods and approaches that have been developed for effective feature selection and design, some issues remain outstanding (Shimizu & Jindo, 1995). End user preferences are usually intensively influenced by the features that are available in the product whose purchase decision is to be made. A good example is in the design of mobile phones where the decision to buy or not to buy is deeply influenced by the features available on the phone such as internet, music, video, camera, Bluetooth, memory card, gallery support and so on. People use features ranking to gain understanding on the features of which users value.

Due to the complexity of high dimensional features, reliable and effective techniques are required to rank them. Most product design industries rely on the use of opinions from leading industry experts as well as focus groups for development of attractive and competitive features in their products (Han and Kim, 1989). Although this technique applies to selection and ranking of product features, it has several drawbacks and lacks of tools support. Use of opinions from various industry experts as well as focus groups lacks objectivity, the qualification as well as availability of industry experts and therefore cannot be relied upon for complex and continuous products design based on the features. Han & Kim (S. H. Han & Kim, 2003) used various traditional statistical techniques for screening various critical design features. Some of the methods used included Principal Component Regression (PCR), Cluster Analysis as well as Cluster Least Squares (Y.-W. Chen & C.-J. Lin, 2006) will be discussed in the following section.

2.2.2 State-of-the-art

This section introduces some of the feature ranking algorithms. The training set ε includes n examples, $\varepsilon = \left\{ \left(x_i, y_i \right), x_i \in \Re^d, y_i \in \left\{ -1,1 \right\}, i = 1,...,n \right\}$. The i–th example is described as d continuous feature values; label y_i indicates the example pertains to the target concept or not.

2.2.2.1 Univariate Feature Ranking

In univariate approach, a feature score is computed after a statistical test to quantify the discriminative strength of a feature. Univariate approach is hindered by feature redundancy; i.e. features correlated to target concept will be ranked first with little regard on the information it offers (Jong, Mary, Cornuejols, Marchiori, & Sebag, 2004). Pepe et. al. (Pepe, Longton, Anderson, & Schummer, 2003) reported to support the identification of differentially relevant features and associates to the k-th feature the score defined as the fraction of pairs between the positive and negative examples $P_r(x_{i,k} > x_{j,k} \mid y_i > y_j)$. This criterion coincides with the Wilcoxon rank sum test, which is equivalent to the AUC criterion (Yan, Dodier, Mozer, & Wolniewicz, 2003).

2.2.2.2 Principal Component

Principal Component Regression variables have been frequently used in estimation of the values of response variables on the basis of the chosen principal components (K. Z. Mao, 2004). The principle component regression of variables is used due to the nature of explanatory variables which usually have multi-colinearity resulting to inaccurate estimations of the least square regression coefficients. Multi-collinearity (Farrar & Glauber, 1967) is a high degree of correlation (linear dependency) among several independent variables. It commonly occurs when a large number of independent variables are incorporated in a regression model. Robust Principle Components Regression (PCR) and Principle Components Analysis (PCA) were developed for identification of outliers (Park & Han, 2004).

Assuming we have a data vector X with p variables. The idea of PCA is to locate a linear combination of the variables such that it achieves large variance in new variable. The j-th principal component PC_i (de Sá et al., 2007)is the linear combination of the original variables $X_1, X_2, \dots, X_p$:

$$PC_j = a_{j1}X_1 + a_{j2}X_2 + a_{j3}X_3 + \cdots + a_{jp}X_p = \sum_{i=1}^{p} a_{ji}X_i \quad (2.30)$$

Under the condition of

$$\boldsymbol{a_{j1}^2 + a_{j2}^2 + \cdots a_{jp}^2 = 1} \quad (2.31)$$

Where are coefficients assigned to the original *p* variables for *PCj*. The following algorithm (Smith, 2002) is the general process of PCA:

Step 1 – Data preparation: The data is usually in large dimensional or large features. These data must be pre-processed before computing the various matrixes.

Step 2 – Mean subtraction: The mean for a given sample set *X*, is computed using the following formula.

$$\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n} \quad (2.32)$$

However, the mean only shows some centre point of the whole dataset. It is not sufficient to show the nature of the data. We have to consider how spread out the data is.

Step 3 – Compute co-variance matrix: Variance is the spread of data. Variance and standard deviation only show the relationship of data within its own dimension. Covariance matrix shows the variance between dimensions. The covariance can be computed as follows:

$$cov(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)} \quad (2.33)$$

The result shows how similar or dissimilar the data are. Positive result shows that the dimensions increase together. Negative result shows that the dimensions decrease together. With covariance equals to zero, it shows that two dimensions are independent of each other.

Given a matrix C^{mxn} with m rows and n columns, and Dim_x is x^{th} dimension, the covariance matrix is given by:

$$C^{mxn} = (c_{i,j}, c_{i,j}, = cov(Dim_i, Dim_j)) \quad (2.34)$$

Step 4 – Compute eigenvectors and eigenvalues: Eigenvectors of transformation are vectors which are either left unaffected or simply multiplied by a scale factor after the transformation. The eigenvalue of an eigenvector is the scale factor by which it has been multiplied (Wikipedia).

$$\lambda v = Cv \tag{2.35}$$

Where λ is an eigenvalue of *C* and *v* is the associated eigenvector of λ. Eigenvector with the highest eigenvalue is generally the principal component of the data set. It is the most significant relationship between the data dimensions.

Step 5 – Determine the components and construct feature vector: Feature vector is constructed by taking the eigenvectors of interest, and forms a matrix in columns.

$$Feature\ Vector = (eig_1 eig_2 eig_3 \dots eig_n) \tag{2.36}$$

The feature vector is then multiplied by the original dataset.

2.2.2.3 Cluster Analysis

Cluster Analysis involves formation of groups from similar items on the basis of several measurements for the different types of objects. Partial Least Squares (PLS) is one of the most recent techniques that use generalization and combination of features from PCA and multiple regressions. When a set of dependent variables is to be predicted from a large set of independent variables or predictors, Cluster Analysis is the best technique to use (Hermes & Buhmann, 2000). Sun & Yang (S. H. Han & H. Yang, 2004) applied a generic algorithm based on PLS (Partial Least Squares) method for selection and ranking of screen design variables. Feature selection difficulties are encountered in various displaces such as software engineering, chemistry, medicine and many others. The biggest task in feature ranking is usually in the identification of a subset of persistent features that have the least generalization errors. In addition, feature ranking is in most cases characterized by difficulties in selection of the smallest subsets with a specific discrimination capability (Chang & Lin, 2008).

2.2.2.4 Rough Sets

Another generally used method is rough sets. The rough sets method provides a mathematical approach for solving imperfect knowledge (Ning, Andrzej, & Setsuo, 1999). Imperfect knowledge problems have been solved for a long time by logicians, philosophers as well as mathematicians. Rough sets methods are widely used for solving complex problems in artificial intelligence. This method provides various approaches that can be used for problem understanding as well as manipulation of imperfect knowledge. The rough set theory has been developed by various developers and practitioners and has been widely used in feature ranking for development of numerous applications in several disciplines. The rough set theory has served fundamental and critical importance in cognitive sciences as well as artificial intelligence. Some of the fields in cognitive sciences and Artificial Intelligence (AI) where rough set theory have been intensively utilized for feature ranking includes machine learning, decision analysis, knowledge acquisition, data mining, expert systems, pattern recognition and inductive reasoning (Casti, 1989). The main advantage of rough sets in feature ranking is that it does not require preliminary information about data. The theory provides effective and efficient algorithms that can be used for identification of hidden patterns in features, evaluation of minimal features through feature reduction, evaluation of feature significance, generation of sets of decision rules from data, interpretation of results as well as parallel reasoning (Hsu & Lin, 2002) . Set theory is critical in mathematical evaluations of complex problems such as feature ranking. The use of rough set theory displays vagueness when boundary regions of a set are employed (Anderberg, 1973). Empty boundary regions results to crisp sets while none empty boundary regions results to rough sets. Rough boundary regions are an inexact boundary region which means that comprehension of sets that not provide sufficient knowledge for precise definition of sets. Fuzzy sets theory is one of the most successful approaches for tackling vagueness in sets. The theory defines sets using partial membership as opposed to crisp membership, which is used in classical set definition techniques. Classical

mathematical sets have contradictions such as the powerset contradiction (antinomy) that can be solved using Axiomatic Set Theory (Han & Yang, 2004), Type Theory (Hindley, 2008) and Classes Theory (Evgeniou, Pontil, Papageorgiou, & Poggio, 2003).

2.3 Conclusion

This chapter provides a study on self-organizing map and feature ranking. The origins of self-organizing maps from Von Malsburg and Willshaw's Self-organizing Model to Kohonen SOM are introduced. We then proceed to introduce the measurements in SOM like topographic products, topographic function, topographic error. The various extensions of SOM like Adaptive Subspace SOM, Visualization induced SOM, Growing Hierarchical SOM and Self-Organizing Map Structured Data are discussed. The other part of this chapter is about feature ranking. Feature ranking is an important aspect to optimise the learning performance.

We shall now proceed to the introduction of Two-Tier Emergent Self-Organizing Map (TtEsom). The TtEsom is evolved from an extension of SOM, Emergent Self-Organizing Map. This variant of SOM allows the emergence of intrinsic features of high dimensional data map onto a two dimensional map. The structure of TtEsom, the adaptation between the layers and its convergence are described in chapter 3.

Chapter 3 Two-Tier Emergent Self-Organizing Map (TtEsom)

3.1 Introduction

Emergent Self-Organizing Map (Esom) (A. Ultsch & Mörchen, 2005) is an extension of SOM that allows the emergence of intrinsic features from high dimensional data map onto a two dimensional map. The SOM is one of the most widely used artificial neural networks which have been applied successfully in the field of data analysis and clustering. The driving force behind SOM lies on the need to analyse data in a self adaptive and organized manner for reliability at the same time taking into account the inter-dependency among the data. The real world data are often complex and frequently updated, the implicit structures are hidden underneath the high dimensions a data may have. Examples of such data are protein data, Online Analytical Processing (OLAP) database (Berson & Smith, 1997), and etc. Clustering of these data requires a sound and reliable model that can span on thousands of datums. This model should have properties like incremental learning, discovery capabilities and can adapt to the changing structures in data. A dynamic and self

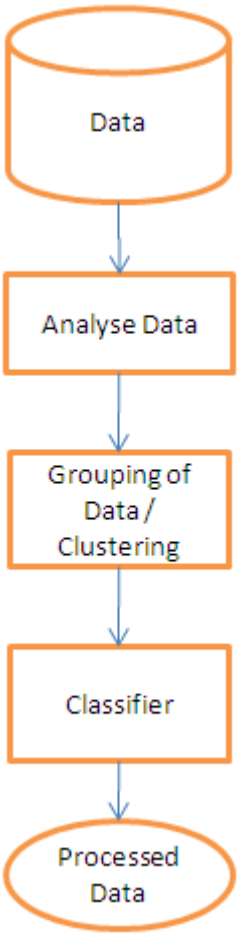


Figure 3-1 Process of analyzing data

organizing model with powerful adaptiveness is therefore desired to cope with these dynamics in data. The general data analysis could be visualised as having the process shown in Figure 3-1.

K-means clustering can be used to complement the self-organizing process but it is inadequate as it is only applicable to applications involving small number of neurons within the network. Esom on the other hand is capable of handling larger maps which are necessary to reflect the topological structure of data. K-means algorithm has been proven to be stable but its biggest limitation is the need to define the number of cluster beforehand which makes it difficult when no prior knowledge is available about the data distribution (MacQueen., 1967). The other disadvantage of K-means algorithm is its intense confinement to local minima (T. M. Martinetz,

Berkovich, & Schulten, 1993). The weaknesses can be solved through modification of the reference vectors where the winning vector is modified and the reference vector adjusted based on their proximity to different input vector (Nowlan, 1990).

This chapter will illustrate a two-tier emergent self organizing map (TtEsom) which differs from the traditional self organizing map. The TtEsom can be used for visualization, clustering and classification. This model is build on the foundation of self-organizing map (Kohonen, 1982) and emergent self organizing map (A. Ultsch & Mörchen, 2005). It provides fast and reliable incremental structure discovery, self adaptation and effective topology that preserve the original data structure.

3.2 Emergent Self-Organizing Map (ESOM)

The Emergent Self-Organizing Map is a non-linear projection technique using neurons arranged on a map. The ESOM forms a low dimensional grid of high dimensional prototype vectors (A. Ultsch & Mörchen, 2005). The density of data in the vicinity of the models associated with the map neurons, and the distances between the models, are taken into account for better visualization. An ESOM map consists of a U-Map (from U-Matrix), a P-Map (from P-Matrix) and a U*-Map (which combines the U and P map). The three maps show the layout for a landscape like visualization for distance and density structure of the high dimensional data space. Structures emerge on top of the map by the cooperation of many neurons. These emerging structures are the main concept of ESOM. It can be used to achieve visualization, clustering, and classification. The different maps (A. Ultsch & Mörchen, 2005) for visualization and the clustering algorithm are introduced in the following sub-sections.

3.2.1 Map Visualization

Let $m: D \to M$ be a mapping from a high dimensional data space $D \subset \Re^n$ onto a finite set of positions $M = \{n_1, \ldots, n_k\} \subset \Re^2$ arranged on a grid for k dimensions. Each position has its two dimensional coordinates and a weight vector $W = \{w_1, \ldots, w_k\}$ which is the image of a Voronoi region in *D*: the data set $E = \{x_1, \ldots, x_d\}$ (where $x_i \in D$ and d denotes the total number of data) is mapped to a position in M such that a data point x_i is mapped to its best-match $bm(x_i) = n_b \in M$ with $d(x, w_b) \leq d(x, w_j); \forall w_j \in W$, where d is the distance on the data set. $d(x, w_b)$ denotes the distance between input and best matching unit and $d(x, w_j)$ denotes the distance between the input and current neuron. The neuron with the shortest distance to input is the best matching unit. The set of immediate neighbours of a position n_i on the grid is denoted by N(i).

3.2.2 U-Map (Distance-based Visualization)

The U-height for each neuron n_i is the average distance of n_i's weight vectors to the weight vectors of its immediate neighbours $N(i)$. The U-height, denoted as $uh(i)$, is calculated as follows:

$$uh(i) = \frac{1}{|N(i)|} \sum_j d(w_i, w_j),\ j \in N(i) \tag{3.1}$$

A display of all U-heights on top of the map is called a U-Matrix (Ultsch and Siemon, 1990). The height value will be large in area where no or few data points reside, creating mountain ranges for cluster boundaries. The sum will be small in area of high densities. A U-map is a height-field like visualization of the U-matrix. The local distance structure is displayed at each neuron as a height value creating a 3D landscape of the high dimensional data space.

3.2.3 P-Map (Density-based Visualization)

The P-height $ph(i)$ for a neuron n_i is a measure of the density of data points in the vicinity of w_i:

$$ph(i) = |\{x \in E \mid d(x, w_j) < r\},\ r > 0,\ r \in \Re\}| \quad (3.2)$$

A display of all P-heights on top of the grid G is called a P-Matrix. The radius r should be chosen such that $ph(i)$ approximates the probability density function of the data points. Distance based methods usually work well for clearly separated clusters, problems can occur with slowly changing densities and overlapping clusters. Density-based methods directly measure the density in the data space sampled at the prototype vectors. The P-matrix displays the local density measures with Pareto Density Estimation (PDE) (Ultsch, 2003).

3.2.4 U*-Map (Distance and Density based Visualization)

For the identification of clusters in data sets, it is sometimes not enough to consider distances between the data points. The local distance depicted in an U-Matrix are presumably distances measured inside a cluster for dense region, such distance is often disregarded for clustering purpose. However, in less dense region, such distance is important where the U-Matrix heights correspond to cluster boundaries. This leads to the definition of an U*-Matrix described in (Ultsch, 2005). The U*-matrix combines the distance-based U-matrix and the density-based P-matrix. The U*-matrix shows significant improvement over U-matrix in dataset with clusters that are not clearly separated in the high dimensional space. Since $uh(i)$ denotes the U-height of a neuron *i*, $\overline{ph}$ denotes the mean of all P-heights, and $\max_i \{ph(i)\}$ is the maximum of all P-heights. The U*-height, denoted as $u^*h(i)$, of an U-Matrix for neuron *i* is calculated as:

$$u^*h(i) = uh(i) \cdot \lambda(i), \quad (3.3)$$

where $\lambda(i)$ is denoted as a scaling factor which is calculated as:

$$\lambda(i) = \frac{ph(i) - \overline{ph}}{\overline{ph} - \max_i \{ph(i)\}} + 1. \quad (3.4)$$

With this equation, the scaling factor is a linear function of P-heights. Using this scaling factor, the U*-height is equal to U-height if $ph(i) = \overline{ph}$ of neuron *i*. We expect the probability density function of P-height is bimodal and its density distribution is a combination of within cluster density distribution and the densities of weight vectors in between clusters.

3.2.5 ESOM Clustering and Classification

Unlike standard Kohonen map (Kohonen 1990) and the other mapping methods like manifold embedding (Rao et al. 2006, Nie et al. 2010) which is only a single mapping approach, the ESOM is a multiple mapping approach which consists of three maps, i.e., a U-Map (from U-Matrix), a P-Map (from P-Matrix) and a U*-Map (which combines the U and P map), in which these three maps would show a landscape to visualize the distance and density structure of the high dimensional data space. Such structures emerge on top of the map by incorporating many neurons. These emerging structures are the main concept of ESOM which can be used to achieve visualization, clustering, and classification. The clustering algorithm is described by the following.

The clustering of ESOM is based on the U*C clustering algorithm described by (Ultsch, 2005). Consider a normalised data point *x* at the surface of a cluster *C*, with the best match of $n_i = bm(x)$. The neighbourhood size starts from 25 neurons and gradually reduced to 1. The weight vectors of its neighbors $N(i)$ are either within the cluster, in a different cluster or interpolate between clusters. Assume that the inter cluster distances are locally larger than the local within-cluster distances, then the U-heights in $N(i)$ will be large in such directions which point away from the cluster C. Thus, an immersive movement will perform to lead away from cluster borders. It is a movement from one position n_i to another position n_j with the result that

w_i is more within a cluster C than w_i. This immersive movement is performed which starts from a grid position, keeps decreasing the U-matrix value by moving to the neighbor with the smallest value, then keeps increasing the P-matrix value by moving to the neighbor with the largest value.

Finally, the classification phase is performed to determine the category C_j of which the test data belongs to. The clustering process assigns input data *x* to a cluster C_j according to its feature vector. After that, the clusters are pre-labelled with the respective category. The class is determined by taking the cluster whose Euclidean distance is nearest to the test vector. The details of this clustering algorithm can be referred to (Ultsch, 2005), and the algorithm is summarized as follows:

Algorithm 3.1: ESOM Clustering

Initialization:

1. Determine the topology and size of network
2. Initialize the weights of the neurons and initial learning rates
3. The input vectors are normalized to the range of 0 to 1.

Immersion:

For each input *x*, we find the best matched *n* neurons and and construct the matrix as follows:

1. From position *n* follow a descending movement on the U-Matrix until the lowest distance value is reached in position *u*.
2. From position *u* follow an ascending movement on the P-Matrix until the highest density value is reached in position *p*.
3. $I = I\ U\ \{p\};\ \ Immersion(n) = p$

Cluster assignment:

1. Calculate the watersheds for the U*-Matrix using the algorithm in (Luc and Soille, 1991).
2. Partition *I* using these watersheds into clusters $C_1 \ldots C_c$.
3. Assign a data point *x* to a cluster C_j if $Immersion(bm(x)) \in C_j$.

Classification:

1. For each test vector; calculate the Euclidean distance from each cluster.
4. The class of the test vector belongs to the label of the cluster with nearest Euclidean distance to test vector.

Classification:

2. For each test vector; calculate the Euclidean distance from each cluster.
3. The class of the test vector belongs to the label of the cluster with nearest Euclidean distance to test vector.

3.3 Structure of Two-Tier Emergent Self-Organizing Map (TtEsom)

We now present the structure of TtEsom which is shown in Figure 3-2. We employ 2 maps that are interconnected for classification. The input vector $\{f_1, f_2, \dots, f_p\}$ represents the properties that collectively describe the problem setting. The input is first presented to Map-1. The formation of Map-1 adopts Algorithm 3.1. The proposed method was used in an attempt to project the high dimensional data hierarchically using emergent self organizing map with a narrow mapping space. The classification is then done based on their topological characteristics. The second map (Map-2) is learned using Algorithm 3.2. Figure 3-3 denotes the topology of connection for two-tier map. Layer 1 and 2 belongs to Map-1, the first-tier map. The input x-vector goes through competitive learning and forms the output as y'-vector in layer 2. Layer 3 matches the y'-vector to provide input as x'-vector for Map-2, the second-tier map. The final output is denoted as y-vector. The adaptive learning between the two maps adopts a learning algorithm similar to Grossberg learning rule (Hecht-Nielsen, 1988). It is a forward-only counter propagation network (CPN) as shown in Figure 3-3 (Hecht-Nielsen, 1988). The basic idea is that, during adaptation, pairs of example vectors (x, y) are presented to Map-1 and Map-2. These vectors then propagate through the network in a counter-flow manner to yield output vectors x' and y' that are intended to be approximations of x and y. The architecture consists of three layers: an input layer containing n fan-out units that multiplex the input signals $x_1, x_2, \dots, x_n$, an ESOM layer with N processing elements that have output signals $\{z_1, z_2, \dots, z_n\}$ and a final Grossberg layer with m processing element with output $y_1^{'}, y_2^{'}, \dots, y_n^{'}$. The output represents approximations to the components $y_1, y_2, \dots, y_m$ of $y = \emptyset(x)$.

The operation of the network consists of two stages: training and normalization. During training, the network is trained using the training pairs of input and output of mapping ϕ. The input vectors x are drawn from $\Re^n$ in accordance with a fixed probability density function ρ. After training, the w_i vectors arrange themselves in $\Re^n$ in such a way that they are approximately equi-probable in a nearest neighbour manner in respect to x vectors. In other words, given any i, $1 \le i \le N$, and given an x vector drawn from $\Re^n$, the probability that x is closest to w_i is approximately $\frac{1}{N}$. After equilibration of the entire network, the output vector will be approximately $\emptyset(w_i)$, where i is the index of the winning processing element on Map-2. See (Hecht-Nielsen, 1988) for further details.

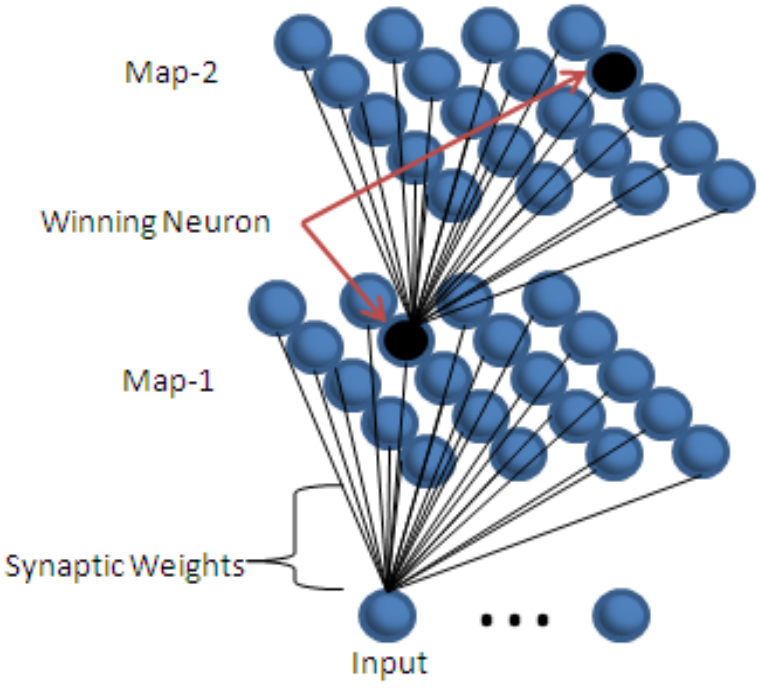


Figure 3-2 Structure of TtEsom

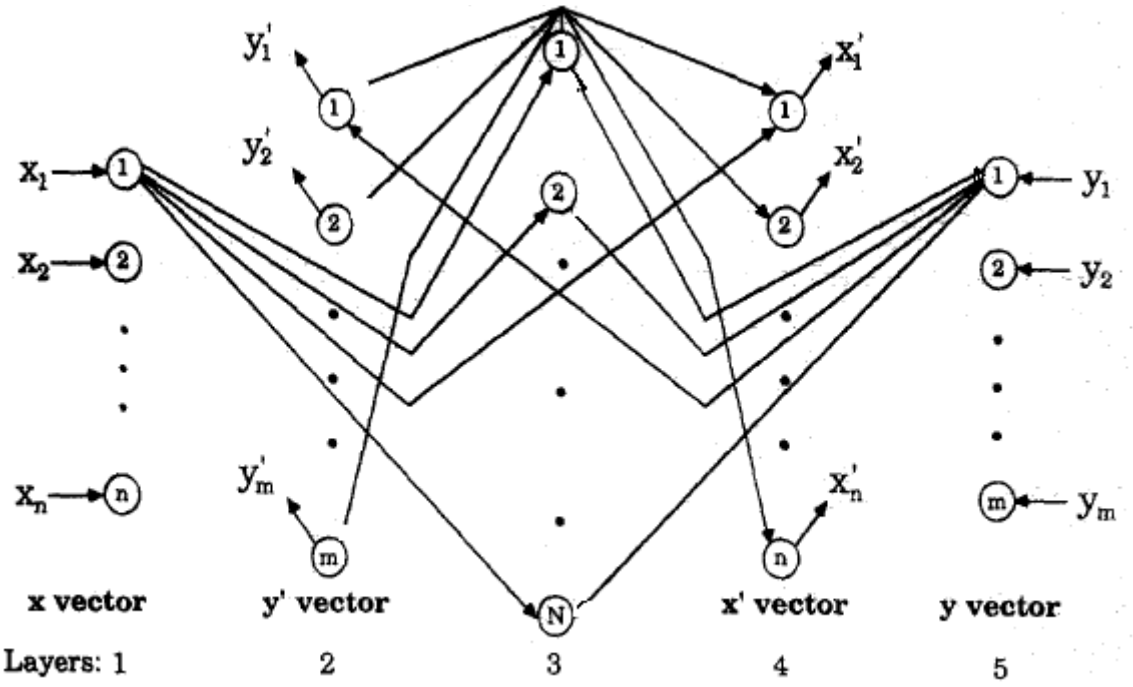


Figure 3-3 Topology of counter propagation network

Algorithm 3.2: Adaptation of Map-1 to Map-2

1. Let $w^i_{n,m}(t)$ denotes the weight of best matching unit (n,m) from Map-1 and $w^j_{n,m}(t)$ indicate the weights of E-Map unit (n,m) at time t which was initialized to small random numbers.
2. Let $x_i(t)$ be the input data to the input layer unit at time t, we calculate the Euclidean distance $d_{n,m}$ between $x_i(t)$ and $w^i_{n,m}(t)$.
3. Update weights and $w^i_{n,m}(t)$ and $w^j_{n,m}(t)$ in the neighborhood region of the winner unit using the following:

   $w^i_{n,m}(t+1) = w^i_{n,m}(t) + \alpha(t)(x_i(t), w^i_{n,m}(t))$

   $w^j_{n,m}(t+1) = w^j_{n,m}(t) + \beta(t)(N_i(t), w^j_{n,m}(t))$

   where $\alpha(t)$ and $\beta(t)$ denotes learning coefficients, and $N_i(t)$ denotes winner in Map-2.
4. Repeat the above steps until the maximum iteration is reached.

The TtEsom can be visualised as a network of neurons with varying patterns in accordance with the input vector in an orderly fashion. The competition among neurons and the learning requirements determine the node into which the input is mapped. We shall now present the full process as follows:

1. Let us denote the network as Ω, an interconnection set denoted as C and a parameter set denoted as ρ. Our TtEsom model can therefore be denoted as shown in the equation below:

$$\xi^t = (\Omega^t, C^t, \rho) \tag{3.5}$$

2. Each neuron denoted as $W_i \subset \Omega^t$ is a vector of dimension d and $i = 1...N$. Where N is the current number of neurons in Ω^t.

3. When an input vector denoted as x enters the network, and there is no neuron that matches the input vector X within a given distance threshold denoted as ϵ for all $i = 1...N$, and $d(W_i, X) = \|W_i - X\| > \in$, then we can create a new relation within our network to represent the input value X as shown in the equations below:

$$W_{n+1} = X \tag{3.6}$$

$$\Omega^{t+1} = \Omega^t \bigcup W_{N+1} \tag{3.7}$$

$$N \leftarrow N + 1 \tag{3.8}$$

4. We then connect the discovered neuron within the network with its two adjacent and nearest neighbors denoted as W_{n1} and W_{n2}. If W_{n1} and W_{n2} are not yet connected, a new connection is established for the next instance is denoted as follows:

$$C^{t+1} = C^t \bigcup C(X, W_{n1}) \bigcup C(X, W_{n2}) \tag{3.9}$$

5. The connection C^{t+1} is the connection which has been established between neurons W_{n1} and W_{n2} at the next time interval.

6. The neurons which require updates are the winning neuron as well as its neighbours. The corresponding neurons are updated according to their input vector X. This relation can be represented by a function f as shown below:

$$\Omega^{t+1} = f(\Omega^t, X) \tag{3.10}$$

7. Each neuron denoted as W is modified as follows:

$$\Delta W = \Upsilon e^{d^2(W,X)/2\sigma^2}(X - W) \quad (3.11)$$

where Υ is used to represent a small constant for the adaption rate while σ is the constant which controls the rate of spreading in the neighbourhood neurons.

8. In the next step, the winner neuron is created. We then continue learning its connections with the neighbouring neurons.
9. Let us denote the connection strengths between the winning neurons W_i and W_j as $s(i, j)$ and the interconnection as $C(W_i, W_j)$.
10. Since the connection strength is dependent on the distance between the neighbouring neurons, we can represent its relationship with the connection as shown in the equation below, where the constant $\in$ controls the spreading between neurons.

$$s(i, j) = \in / d(W_i, W_j) \quad (3.12)$$

11. All the steps above are repeated till convergence.

The TtEsom is aimed at achieving a continuous learning process with a strict convergence of the algorithm. In our TtEsom model, the distance between the neighbouring neurons is used to determine the weight vectors as opposed to the single tier models where the grid distance is used. The TtEsom model overcomes the challenges of large neighbourhoods through the straight forward allocation of weight connections. Determination of winning neurons requires specification of distance thresholds with a higher accuracy value as opposed to the other single layer self organizing map. The model allows for an effective neuron translation process with superior capabilities for recalling novel stimuli, coding it effectively and adapting it to any future changes with speed and accuracy.

The augmented feature map-1 is presented to the second-tier of map. From the best matching unit, the output value is determined and selected. The relationships between the tiers are established through the extension of k-nearest neighbour making use of the k-best matching prototypes. Below is the architectural diagram of the flow.

The projections of TtEsom from high dimensional data spaces to two dimensions have some related errors which must be eliminated for the model to be fully effective. These include similar data points errors and close neighbourhood position errors. For the similar data points errors, we assume that we have points x_1, x_2 which are assigned to distant positions $(m(x_1), m(x_2))$ equivalent to $p(i), p(j)$. This implies that the distance $d(x_1, x_2)$ is smaller than the distance $(m(x_1), m(x_2))$ resulting in a forward projection error. For the close neighbouring position errors, we consider two neighbouring positions $p(i)$ and $p(j) = p(i, j)$. The neighbouring positions may be an image of other distance data points within the data space resulting to backward projection errors (Alfred Ultsch & Herrmann, 2005).

The existence of gaps within the two tiers can cause division of datasets into different classes of coherent elements complicating determination of the forward and backward projection errors. To overcome such challenges, techniques such as the minimal U ranking measure can be used. If we have a path denoted as P_{ij}, as the set of all random paths between nodes $i, j \in I$, the U distance for nodes $i, j \in I$ is the minimal distance between i and j. The following equations hold.

$$\mathrm{PATH_{ij}} = \{\mathrm{f}(\mathrm{i}_1; :::; \mathrm{i}_n) : \mathrm{n} \in \mathrm{N} \setminus \{0,1\}, \mathrm{i}_1 = \mathrm{i}, \mathrm{i}_n = \mathrm{j}, \mathrm{i}_{k+1} \in \mathrm{N}_{ik}^{M}\} \tag{3.13}$$

$$\mathrm{PathDistance}\ (i_1 ..., i_n) = \sum_{k=1}^{n-1} d(w_{ik}, w_{ik+1}) \tag{3.14}$$

$$\mathrm{UDistance}\ (i\ , j) = \min_{q \in PATH_{ij}} \mathrm{PathDistance} \tag{3.15}$$

Path distance (Alfred Ultsch & Herrmann, 2005) is the set of arbitrary path distances within the network while *UDistance* is the smallest distance between the two arbitrary neurons. U

distances can be used for definition of a rank based architecture on a set of neurons which helps to further overcome the problems of gaps within the network of neurons. This can be done as follows:

If we let a distance (udistance(i,i_1) …, udistance(i,i_n)) be the ordered sequence of all the u distances within the network towards a neuron k for all neurons i in $\{i_1...i_n\}$ which is equivalent to k and $udistance(i,i_k) \leq udistance(i,i_{k+1})$ for *k=1,…,n-1*, the ranking of the *UDistance* denoted as $uRank_i$ (j)=$\tau \in \{1...n\}$ can be determined. This implies that $i_r = j$ and our minimal u ranking (*MUR*) measure can therefore be defined as shown in the equation below.

$$\text{MUR (i)} = \sum_{j \in N^D(i)}^{n} urank_i(j) \qquad (3.16)$$

High dimensional data can be visualized through the use of ESOM prototypes within the neuron space. Other algorithms such as Sammon's algorithm can be used for projection of high dimensional data (Sammon, 1969). Using this algorithm, all the data components are projected onto low dimensional space while maintaining an optimum distance ordering within the cell space for faster computation. The required prototype vectors are projected in a two dimensional space while minimizing the mapping errors.

3.4 Conclusion

This chapter introduces the concept of Two-Tier Emergent Self-Organizing Map (TtEsom). The TtEsom is evolved from an extension of SOM, Emergent Self-Organizing Map. This variant of SOM allows the emergence of intrinsic features of high dimensional data map onto a two dimensional map. The structure of TtEsom, the adaptation between the layers was described.

Chapter 4 investigates a cognitive based Emergent Self-Organizing Visual Processing Model (ESOVPM). The framework emulates the neuro-cognitive structure of human visual

processing pathway and topographic maps. Within this brain inspired framework, the object detection and features extraction blocks are built to model the processing taken place from visual pathway. We then demonstrate how this model can fit into simulations like road sign recognition and emotion recognition. In the context of emotion recognition, the model is first trained by the Affect-Map that process positive and negative affects of human emotion. The second map called Emotion-Map is then established through the input from the Affect-Map.

Chapter 4 Self-Organizing Cortical Visual Processing Model

4.1 Overview

Recent advances in behavioral neuroscience enhance our understanding of human visual cortex. In the past few years, several new results on the structure, development, and functional role of lateral connectivity in the cortex emerged. These results have led to a new understanding of the cortex as a continuously-adapting dynamic system shaped by competitive and cooperative lateral interactions (Miikkulainen & Sirosh, 1996). The properties of simple cells have been widely studied (Atick & Redlich., 1990; J. G. Daugman, 1985; Hawken & Parker, 1991). The more extensive works on visual processing model can be categorised into two groups. The first group focuses on the receptive field of neurons. The other group focuses on high level visual processing.

The first group of works provide simulation on the reaction of receptive field. Polat et. al. (Polat, M.Norcia, & Sagi, 1996) presented psychophysical and neurophysiological evidence for facilitation and suppression of responses beyond classically defined receptive fields, proposing that long-range lateral interactions could be responsible for such phenomena. Possible

mechanisms for long range excitatory and inhibitory interaction were reviewed. Gestalt theory was discussed, the central idea of it is that the whole is different from the sum of its part and the visual system organizes parts into wholes. The facilitation effect can be found only when three Gabor patches share the same orientation and spatial frequency as predicted by Gestalt grouping rule. Sabatini (Sabatini, 1996) analyzes the effect of medium-range clustered connections in an orientation map mathematically and shows that they give rise to Gabor-like receptive fields observed in the visual cortex. They argued that intra-cortical inhibition may well be the substrate for a variety of influences observed between RF center and its surround. (Somers et al., 1996) presented a computational simulation showing how fixed-length lateral connections can facilitate or suppress a group of neurons depending on their level of activation. They also present experimental evidence for such gain control, and postulate that it could play a role in perceptual filling-in and discrimination phenomena.

The second group of works explores high-level visual processing. Sirosh et. al. (Sirosh, Miikkulainen, & Bednar., 1996) presented simulations where lateral connections self-organize synergetically with cortical orientation columns, and also mediate reorganization of receptive fields in response to lesions. Lateral connections are shown to de-correlate cortical activity patterns, and thereby play a central role in forming a sparse and redundancy-reduced representation of visual input. Marshall and Alley (Marshall & Alley, 1996) developed a neural network model that learns to detect and represent depth relations, after a period of exposure to motion sequences containing occlusion and dis-occlusion events. The model has two parallel opponent channels: On-chain and Off-chain. Two results obtained from the studies: First, a visual system can learn a non-metric representation of the depth relations arising from occlusion events. Second, parallel opponent On and Off channels that represent both modal and a-modal stimuli can be learned through the same process. Wiskott and Malsburg (Wiskott & Malsburg, 1996) showed how face recognition could take place through a dynamic mapping of the image to a set of models.

The topological constraints that define a match are implemented through lateral interactions. Functional role of lateral connections in visual cortex was studied.

Most simple cells in the primary visual cortex (V1) are selective for the direction and orientation of a moving stimulus. Recent measurement techniques have made it possible to plot the neurons' full spatiotemporal receptive fields, which include specific excitatory (ON) and inhibitory (Hoffman, Grinstein, Marx, Grosse, & Stanley) sub-regions that vary overtime (DeAngelis, Ohzawa, & Freeman, 1995). The functional properties of these cells form a mosaic across V1, with patches of nearby neurons preferring similar directions and orientations (Berkes & Wiskott, 2002a). Apart from their afferent input from the LGN, the neurons in these maps are connected intra-cortically through specific long-range lateral connections (C.D. Gilbert, 1989).

Several computational models have shown that directional selectivity and interleaved orientation and direction maps can be developed through activity-dependent self-organization (Farkas & Miikkulainen, 1999; H. Shouno, 2001; Wimbauer, Wenisch, Hemmen, & Miller, 1997). Studies of orientation maps in primary visual cortex (V1) suggest that lateral connection mediate competition and cooperation between orientation selective units, but their role motion perception has not been established. Bednar and Miillulainen (Bednar & Miikkulainen, 2003) demonstrated that using a self-organizing model of V1 with moving oriented patterns, the afferent weights of each neuron organize into Gabor-like spatiotemporal receptive fields with ON and OFF lobes. They also showed that these receptive fields form realistic joint direction and orientation maps and the lateral connections develop between patches with similar orientation and direction preferences. Their result suggested that a single self-organizing system may underlie the development of orientation selectivity, direction selectivity, and lateral connectivity.

This chapter encompasses the details of the Self-Organizing Visual Processing Model.

4.2 Architecture

In this section, we outline the structure of the visual model based recognition system. This section describes the modelling of cognitive process of human brain in recognizing emotions.

Humans recognize an object from surrounding world in split second; however this involves a lot of processing in human visual system. Human gather most of the sensory information through sight. Visual-perceptual processing covers approximately one-fourth of the cortex. Visual information processing is also the most complex, most studied and best understood sensory system of the brain. Light enters human eye through the pupil. The eye maps visual image and invert it to be further processed by primary visual processing area of the cortex. The rods and cones transduce electromagnetic wavelengths of light energy and extract properties of objects. Light activates cones by different colours and rods by black and white. The structure is shown in Figure 4.1. Bipolar cells connect the receptors to the ganglion cells. The ganglion cells then provide output. Light hyper-polarises rods and cones that decreases the voltage, and darkness depolarises and increases the voltage. Rods, cones, horizontal and bipolar cells provide graded changes in potential that cannot travel long distances. Ganglion cells convert that visual information coded by graded potential changes into a discrete code based on the frequency of action potentials. The information is then transmitted by Ganglion cell and travel down the optic nerve. Receptive field of a Ganglion cell is an area of retina over which light stimuli change the activity of a particular Ganglion cell. There are two types of Ganglion cells. The first type is ON centre and OFF surround which measure how much brighter an object is than its background. The second type is OFF centre and ON surround which measure how much darker. As shown on the right side of Figure 4.2 is the response of ON centre OFF surround ganglion cell when a spot of light is stimulated. Initially there is no change in the tonic activity. Then the spot in excitatory area

increase firing and followed by the spot in inhibitory area decrease firing. The spot outside the receptive field shows no change.

There are two visual pathways in human brain which are responsible for face and emotion recognition called Dorsal and Vectral routes. The dorsal stream begins with V1, goes through visual area V2, then to the dorso-medial area and visual area V5 and to the inferior parietal lobule. The dorsal stream, sometimes referred to as 'Where Pathway", is associated with motion, representation of object locations, and control of the eyes and arms (Goodale & Milner, 1992). The ventral stream begins with V1, goes through visual area V2, then through visual area V4, and to the inferior temporal lobe (See Figure 4.3). The ventral stream, sometimes called the 'What Pathway', is associated with form recognition and object representation, and also storage of long-term memory.

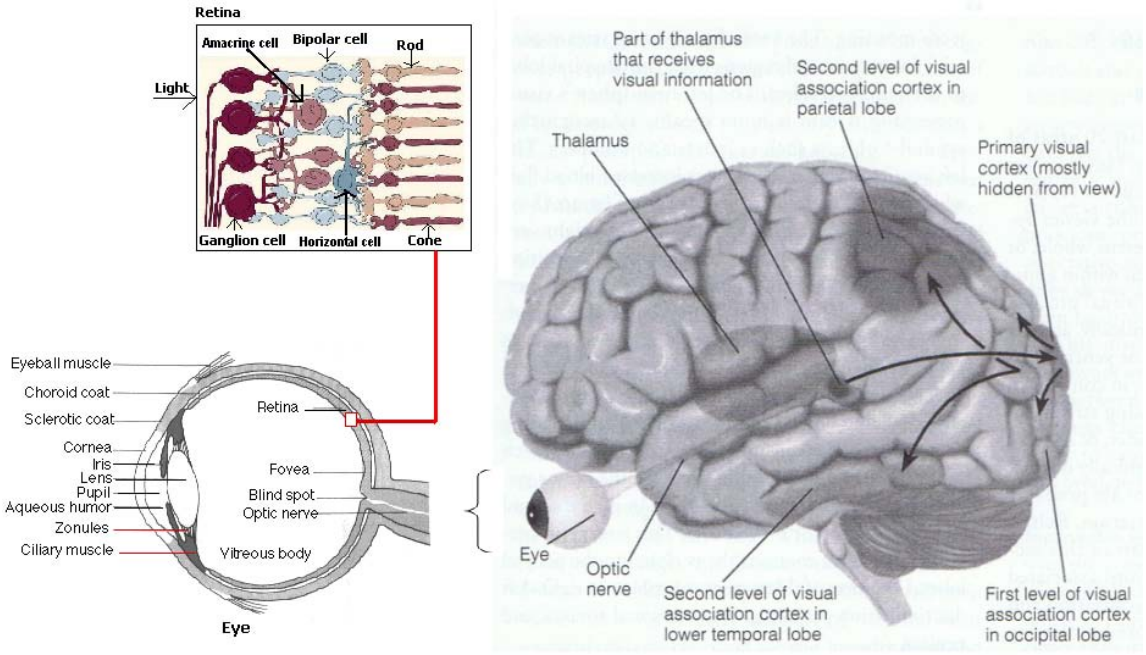


Figure 4-1 The structure of eye and cortical visual processing(Kanndel, Martinetz, & Schulten, 2000).

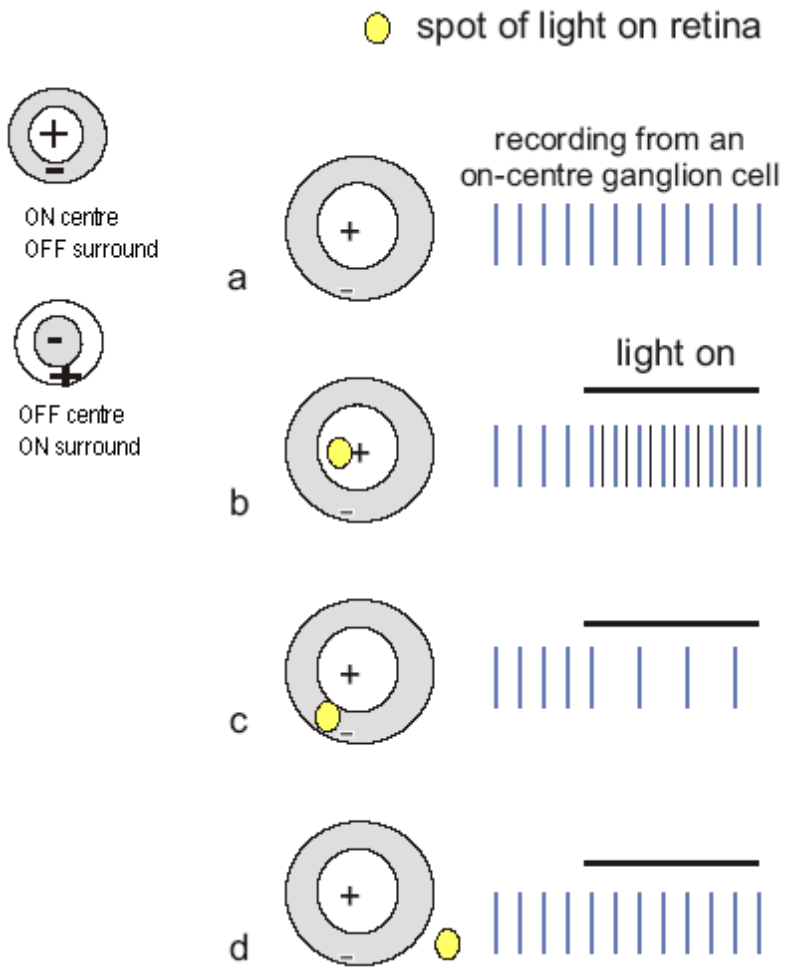


Figure 4-2 Response of Ganglion cell (Kanndel et al., 2000).

Optic nerve's bundle of axons disseminates the visual information and send to various brain areas. The optic tracts synapse with the thalamus in the dorsal portion at the lateral geniculate nucleus (LGN) and followed by projecting it to primary visual cortex. The electrical signals that are converted from wavelength of light terminate at the LGN located at thalamus. The LGN coordinates visual information from the two eyes and relays them to the visual cortex as shown in Figure 4.1. Images seen on one side are processed by the opposite side of the brain which is achieved by crossing optic chiasm at the ganglion cell. The information of each eye does not mix until the primary visual cortex. The LGN consists of 6 layers, 3 receive information from one eye, 3 from the other. The Ganglion cells are primarily M-cells and P-cells. Axons from the

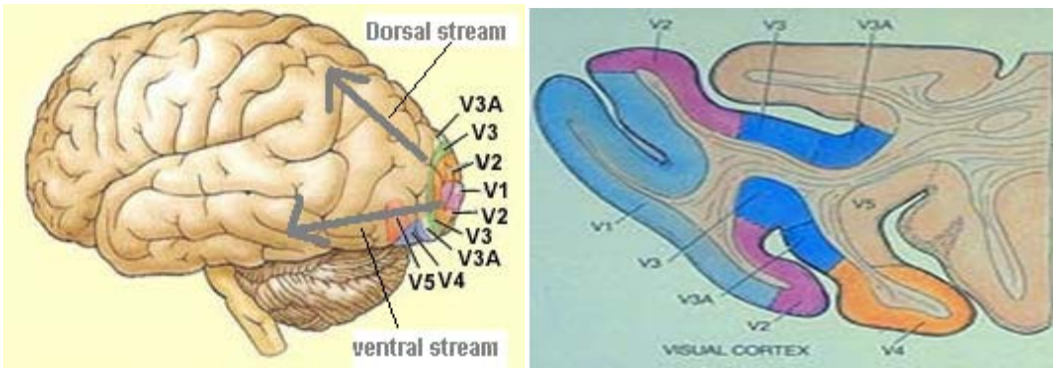


Figure 4-3 The V1, V2, V3, V4 and V5 distribution (Barrow, Bray, & Budd, 1996)

M-cells project to the two lower layers (magnocellular M-layers) of LGN. Axons of the P-cells project to the four upper layers (parvocellular P-layers) of LGN. The parallel processing that begins in the retina is maintained through LGN.

We now present a framework which emulates the neuro-cognitive structure of human visual processing pathway and topographic maps. Within this brain inspired framework, the object detection and features extraction block is built to model the processing takes place from visual pathway. Human brain activation patterns are evoked when we look at an object. Figure 4-4 shows the architecture of the entire system and how it maps to human visual system. Retina consists of a large number of photoreceptor cells. It converts light falling on them into electrical signals. These electrical signals terminate at the LGN (Lateral Geniculate Nucleus), which is a structure found in thalamus. The LGN coordinates visual information from the two eyes and relays them to the visual cortex. Nearly all visual information reaches the primary visual cortex (Area V1). Hubel and Wiesel (Hubel & Wiesel, 1962b) described the primary visual cortex as a structure composes of 'hypercolumns', which consist of cells responding to same spatial location in the retina and orientation but with different position in the visual space (David & Torsten, 1974; Hubel & Wiesel, 1962b).

The process starts with modelling of the cognitive processes of feature extraction. When a stimulus is first detected, it activates in a feed-forward manner. The exemplar (input pattern) is

first transformed to greyscale image. The exemplar is then transformed using Gabor wavelet. Gabor wavelet is known for its capability to approximate mammals' visual cortex. The receptive field of cortical cell (LGN) can be reproduced fairly well using Daugman's Gabor function (Jones & Palmer, 1987). There is considerable evidence that the parameterised family of 2D Gabor filters, proposed by Daugman in 1980, suitably models the profile of receptive cells in the primary visual cortex. Gabor filters models the properties of spatial localization, orientation selectivity, and spatial frequency selectivity and phase relationship of the receptive cells (J. Daugman, 1985).

Following the pre-cortical processing, the features go through features selection process which is similar to the selective tuning (Fairhall & Ishai, 2007) taking place in cognitive states. The features selected resembles to those extracted by the V1, V2, and V4 regions of visual cortex. The amygdale is closely connected with hypothalamic and midbrain motivational areas and is involved in all emotional response, from the most primitive to the most cognitively driven. The multiple features received from the various parts of the visual cortex are forwarded in a hierarchical structure to the amygdale. Then the recognition stage takes the output from the previous stage as input. The procedures involved in this stage are Self-Organizing adaptation, knowledge representation, and classification. Human brain can be thought of as a group of many brain maps designed for different functions. These maps consist of interconnected neurons. They are topological in nature, and certain area of the map respond to only specific stimuli. The discovery in the brain map shows clearly that different sensory inputs (motor, visual and auditory, etc) are mapped onto corresponding areas of the cerebral cortex in an orderly fashion. The essential point of the discovery in the neurobiology lies in the principle of topographic map formation. The principle states that the spatial location of an output neuron in the topographic map corresponds to a particular domain or feature of the input data. This leads to the development of Self-Organizing Map (Somers et al.).

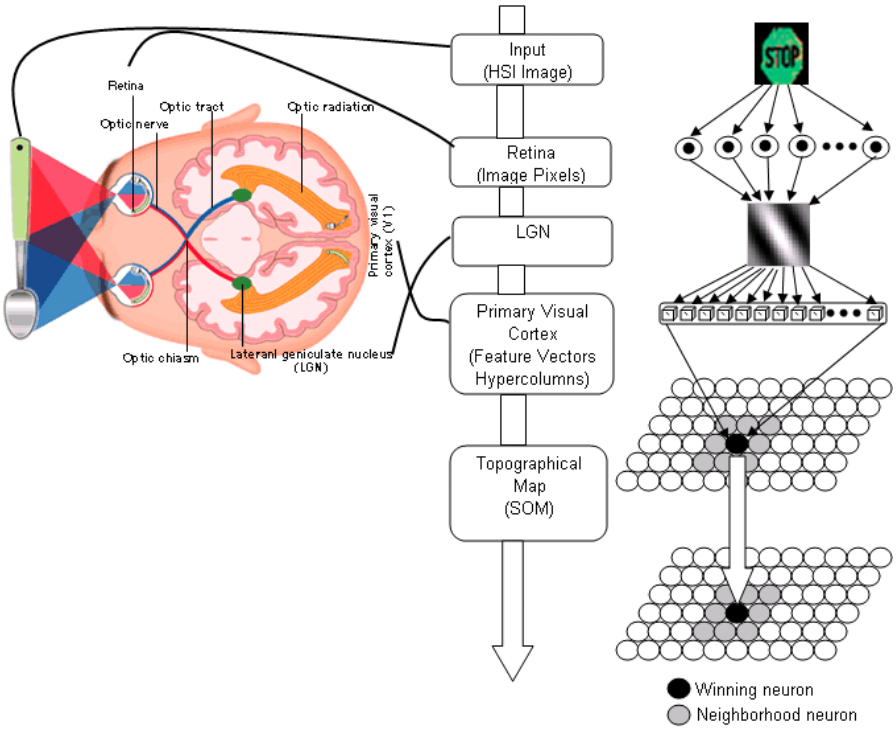


Figure 4-4 Mapping of system design and cognitive process

Figure 4-4 shows the architecture of the entire system and how it maps to human visual system. Retina consists of a large number of photoreceptor cells. It converts light falling on them into electrical signals. These electrical signals terminate at the LGN (Lateral Geniculate Nucleus), which is a structure found in thalamus. The LGN coordinates visual information from the two eyes and relays them to the visual cortex. Nearly all visual information reaches the primary visual cortex (Area V1). Hubel and Wiesel (Hubel & Wiesel, 1962b) described the primary visual cortex as a structure composes of 'hypercolumns', which consist of cells responding to same spatial location in the retina and orientation but with different position in the visual space (C. Liu & Wechsler, 2002). Following the pre-processing stage, the recognition stage takes the output from the previous stage as input. The procedures involved in this stage are Gabor wavelets extraction, Self-Organizing adaptation, knowledge representation, and classification. Gabor wavelets

extraction models the cognitive process of feature selection. It extracts essential information and passes information for Self-Organizing adaptation. A biologically similar map is then achieved through competitive learning.

Algorithm 4.1 presents the flow of Emergent Self-Organizing Visual Processing Model (ESOVPM). The algorithm consists of pre-cortical processing, cortical processing and recognition. The details of emersion clustering algorithm can be referred to (A. Ultsch, 2005).

Algorithm 4.1: Emergent Self-Organizing Visual Processing Architecture

Given pixels of static input vector.

Pre-cortical processing:

1. Determine the spatial frequency W, and spatial variances σ_x^2 and σ_y^2 for the Gabor wavelets formation.

   $g(x,y) = g_1(x,y)\exp(j2\pi Wx)$

   $$g_1(x,y) = \left(\frac{1}{2\pi\sigma_x\sigma_y}\right) exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)\right)$$

2. Convolute the gabor wavelets with step 1 output. For the hue component in HSI case, the convolution is done as follows:

   ON-channel: $h^+ = g_c * I_h - k \cdot g_s * I_h$

   OFF-channel: $h^- = g_s * I_h - k \cdot g_c * I_h$

3. Dimensionality reduction to lower dimension space by finding the *j*-th principal component PC_j by:

   $$PC_j = a_{j_1} X_1 + a_{j_2} X_2 + a_{j_3} X_3 + \ldots + a_{j_p} X_p = \sum_{i=1}^{p} a_{j_i} X_i$$

   under the condition,

   $a_{j_1}^2 + a_{j_2}^2 + \ldots + a_{j_p}^2 = 1$

Cortical Processing:

1. Find the U-Matrix, P-Matrix, U*-Matrix and let I={}
2. For all positions *n* of the grid:
   - From position *n* follow a descending movement on the U-Matrix until the lowest distance value is reached in position *u*.
   - From position *u* follow an ascending movement on the P-Matrix until the highest density value is reached in position *p*.
   - $I = I \cup \{p\}$; $\text{Immersion}(n) = p$.

Cluster assignment and recognition:

1. Calculate the watersheds for the U*-Matrix using the algorithm in [25].
2. Partition *I* using these watersheds into clusters $C_1, \ldots, C_c$.
3. Assign a data point *x* to a cluster C_j if $Immersion(bm(x)) \in C_j$.
4. Adopt K nearest neighbor approach for recognition.
   X_i: number of neurons in K corresponding to emotion category
   $C_i = \max(X_i)$
5. where *K* in the range of $[1, \sqrt{N}]$ with $N = 400$.

4.2.1 Pre-cortical Processing

The input intensity patterns received by human visual system are typically complicated functions of object surfaces and light sources in the world. It seems probable, that human perceive the world in terms of surfaces and objects (Nakayama & Shimojo, 1992). Thus the visual system must be able to extract information from the input intensities that is relatively independent of the actual intensity values. Our model takes in input from colour images, represented as arrays in the hue, saturation, and intensity bands. We attempt to model responses of on-centre and off-centre cells for the broad-band and colour channels. There are six types of retinal and six types of LGN cells are modelled as using different Gabor wavelets, where each Gabor wavelet is applied to a particular spectral band.

The primary cortex of human brain interprets visual signals. It consists of neurons, which respond differently to different stimuli attributes. The receptive field of cortical cell consists of a central ON region is surrounded by 2 OFF regions, each region elongates along a preferred orientation (C. Liu & Wechsler, 2002). According to Jones and Palmer, these receptive fields can be reproduced fairly well using Daugman's Gabor function (Jones & Palmer, 1987).

The Gabor wavelet function can be represented by:

$$g(x,y) = g_1(x,y)\exp(j2\pi Wx) \tag{4.1}$$

where

$$g_1(x,y) = \left(\frac{1}{2\pi\sigma_x\sigma_y}\right) exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)\right) \tag{4.2}$$

We consider the receptive field (RF) of each cortical cell consists of a central ON region (a region excited by light) surrounded by two lateral OFF regions (excited by darkness) (G.E. La Cara, 2003). Spatial frequency (W) determines the width of the ON and OFF regions. σ_x^2 and σ_y^2 are spatial variances which establish the dimension of the RF in the preferred and non-preferred orientations.

The input vector is projected onto different Gabor wavelets to generate the output signals that resemble electrical signals in visual cortex. Different orientations and special frequencies produce different wavelets. After the convolution of input and wavelets, a set of feature vectors is formed that acts like the 'hypercolumns' as described by Hubel and Wiesel (Hubel & Wiesel, 1962a).

As shown in Figure 4-4, the Gabor wavelets are represented with different orientations and frequencies. These Gabor wavelets act as stimuli to the system. Figure 4-5 (Berkes & Wiskott, 2002b) gives an overview of the optimal stimuli for the first 48 units resulting from one typical simulation. Since the processing at the retina and LGN cortical layers are used by Gaussian kernels applying to some spectral bands to simulate local inhibition, most stimuli resemble Gabor wavelets.

In this cortical processing, there are six types of outputs from LGN cells. They are modelled using a set of Gabor wavelet convolution, where each Gabor kernel is applied to a particular domain. For example, the output of a hue opponent cell is a function of the difference between a small light sensitive central region and a larger dark sensitive surround region. Since the lateral inhibition is applied to the ON- and OFF- channels separately and independently in the LGN cells, thus the ON-channel and OFF-channel outputs of hue cell are essential to be shown as the following equations respectively:

ON-channel: $h^{+} = g_c * I_h - k \cdot g_s * I_h$ (4.3)

OFF-channel: $h^{-} = g_s * I_h - k \cdot g_c * I_h$ (4.4)

Assuming a dark centre Gabor function g and light surround Gabor function g, are used to convolute with the intensity of the hue component I_h, where * is the convolution operator. k denotes the relative weighting of centre and surround.

The ON-channel and OFF-channel outputs of the saturation (s^+, s^-) and intensity (v^+, v^-) cells can be produced in the similar manner as the above hue cell. A set of features can then be formed as $(h^+, h^-, s^+, s^-, v^+, v^-)$. The dimension of the features set is dependent on the number of Gabor functions being used.

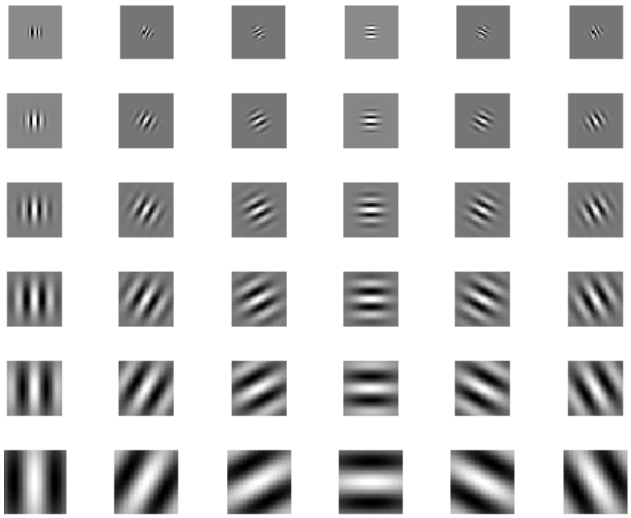


Figure 4-5 Gabor wavelets representations

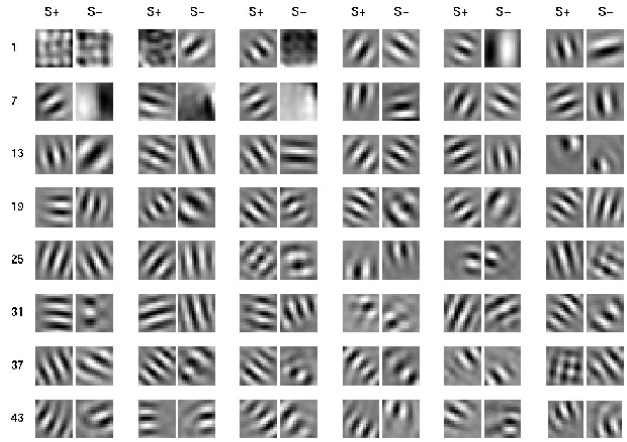


Figure 4-6 Overview of optimal excitatory and inhibitory stimuli (S+ res. S-)

4.2.2 Cortical Processing

Our model of simple cell connectivity and self-organizing has its base on kohonen model (Kohonen, 1982) and emergent self-organizing model (A. Ultsch & Mörchen, 2005). It consists of two types of unit representing two broad categories of cells in the primate striate cortex. The excitatory units represent spiny cells and the inhibitory one represent smooth cells. Both types of cells receive excitatory feedforward input from the LGN. Excitatory cells excite all neighboring cells within a short radius, and the inhibitory cells inhibit all neighbours within a larger radius (Barrow et al., 1996).

In the visual cortex, cooperation and competition between axons influence the development of ocular dominance columns primarily during critical period of development (Kanndel et al., 2000). The balance of activity in the fibers from the two eyes affects the segregation of afferent fibers and the establishment of the ocular dominance columns. Synchronous activity in neighboring afferent fibers sharpens the topographic mapping of retical axons onto their central targets. Connection becomes stable and much less susceptible to change after the critical period. These findings can be interpreted as an example of Hebbian learning-synapses are strengthen when presynaptic and postsynaptic elements are synchronously active. This forms the basis of models of neural competition and cooperation. The cortical units receive feedforward input through their connections to geniculate cells. The network settles into a stable state where intra-cortical feedback is no longer changing. All units then adapt their connections to the geniculate cells using Hebbian-type learning rule. The cortical cell can be modelled using the neuron in the map having an initial random value. The connections between cortical and geniculate cells are initialised to small random values.

4.2.3 Cluster Assignment and Recognition

Consider a data point *x* at the surface of a cluster C, with $n = bm(x)$. The weight vectors of its neighbours $N(i)$ are either within the cluster, in a different cluster or interpolate between clusters. If we assume that the inter cluster distances are locally larger than the local inner cluster distances, then the U-heights in $N(i)$ will be large in such directions which point away from the cluster. Thus, a gradient descent on the U-matrix will perform to lead away from cluster borders. A movement from one position n_i to another position n_j with the result that w_j is closer within a cluster C than w_i is called *Immersive (A. Ultsch & Mörchen, 2005)*. For data points well within C, a gradient decent on a U-matrix will, however, not necessarily be immersive. The P-heights follow the density structure of a cluster. Under the assumption that the core parts of a cluster are those regions with largest density, a gradient ascent on the P-Matrix is immersive. Clusters may also be defined by density alone instead of distance.

At the borders of a cluster, the measurement of density is, however, critical. At cluster borders, the local density of the points should decrease substantially. In most cases, the cluster borders are defined either by low point densities or by "empty space" between clusters (i.e. large inter cluster distances). For empirical estimates of the point density, a gradient ascent on a P-Matrix may therefore not be immersive for points at cluster borders. Let *I* denotes the end points of immersion starting from every position on a grid. If the density within a cluster is constant, immersion will not converge to a single point for a cluster for all starting points within a cluster. The U*-Matrix is then used to determine which points in *I* belong to the same cluster. The watersheds of the U*-Matrix are calculated using algorithm described in (A. Ultsch & Mörchen, 2005). Data points that are separated by a watershed are assigned to different clusters, data points within the same basin to a single cluster.

After the map has been adapted, we need to annotate and represent the cluster according to its nature. Physiologically, knowledge representation is a meta-cognitive function belonging to

Layer 5 of the Brain Reference Model. Knowledge representation is an important step because recognition, a Layer 6 higher cognitive function, will require the represented knowledge to carry out recognition.

The category to which a neuron would respond to is found by first measuring the Euclidean distances between weight vectors and training input vectors obtained. Followed by marking the winning neuron corresponding to each input image with the category it belongs to. These neurons shall be called Input-Marked Neurons for differentiation purposes. For the rest of the unmarked neurons, determine which Input-Marked Neuron is the unmarked neuron closer to using Euclidean distance of their weights. The unmarked neurons are then labelled following the most similar Input-Marked Neuron.

Recognition is a higher cognitive process belonging to Layer 6 of the Brain Reference Model. In this work, we adopted K-Nearest Neighbour for recognition. KNN is a method that can be used to evaluate the quality of clusters. With the feature vector extracted using Gabor wavelets, the Euclidean distance of feature vector and neurons' weight are determined. K neurons whose weights are closest to test vector are selected. The number of neurons X_i in K corresponds to emotion category V_i is determined. The cluster $C_i = \max(X_i)$, where K is determined empirically is often in the range $\left[1, \sqrt{N}\right]$ (N is the number of neurons under investigation).

4.2.4 Simulation I – Road Sign Recognition

Road sign recognition system is a very important area that deserves attention. A road sign recognition system provides timely alert to warn the driver of any critical sign ahead. The objective of a road sign recognition system is to detect and classify one or more road signs from colour images captured by camera. There exist many challenges that such a system should address. For instances, lighting condition is a very difficult problem to regulate. The strength of

the light depends on the time of the day and season, and also on the weather conditions. In addition, road sign patterns within images can be affected by shadows from surrounding objects.

In general, a road sign recognition system will first detect the road sign of interest in the image followed by classifying it into different classes. Most of the solutions rely on the colour and shape of the road sign. Colour is a visual feature that represents the most significant clue that can be easily noticed by the driver. The colours that are used in road signs are regulated by different countries and often include simple primary colours (red, green, or blue) with the exception of yellow, a secondary colour. Colour-based detection methods aim to segment the typical colours of road signs in order to provide a region of interest for further processing. Some colour-based detection methods are colour thresholding (Benallal & Meunier, 2003), colour indexing, dynamic pixel aggregation (S.Vitabile, G.Pollaccia, G.Pilato, & F.Sorbello, 2001), and region growing. Shape, being one of the two important attributes of road signs, can also be used for road sign recognition. Shape detection does not require colour information. However the selection of a scheme for the detection of road signs based on their shapes will have to address more issues than their colour. For example such issues as road signs in cluttered scenes, imperfect shape, as well as variance in scale and size make the detection task very challenging. Some implemented shape-based methods are Hierarchical Spatial Feature Matching (HSFM) (Paclik & Novovicova, 2000), Template Matching (Lauziere, Gingras, & Ferrie, 2001), Similarity Detection (S.Vitabile et al., 2001), and Distance Transform Matching (Gavrila, 1999). The other methods that have been implemented successfully in road sign recognition include Genetic Algorithm (Aoyagi & Asakura, 1996), and Histographic Recognition (Torresen, Bakke, & Sekanina, 2004).

The rationale behind using a self-organizing approach in the road sign recognition is that in the context of a driving support system, the recognition method cannot always be taught in advance with all possible road signs; for instance, some countries might have their own particular road signs. We thereby introduce the use of Emergent Self-Organizing Map (ESOM) that forms

clusters of different road signs by itself. It would be useful to detect and identify the new kinds of road signs by means of interactive training (or active learning) system. On the other hand, the maps used by most SOM applications are usually small and face significant performance degradation when run on large data sets. Training a small SOM on a data set is similar to k-means clustering with k equals to the number of nodes in the map. ESOM is an extension of SOM in which large numbers of neurons are used to allow topology preservation and to allow data structure to emerge on the maps. It has been demonstrated that using ESOM is a significantly different process from using k-means (A. Ultsch & Mörchen, 2005).

4.2.3.1 Framework for Road Sign Recognition

The road sign image first goes through acquisition and extraction to segment the input image and extract out the areas that contain road sign patterns. It consists of two components: segmentation and filtering. In our work, the Hue-Saturation-Intensity (HSI) transformation is appealing in colour segmentation because it gives unique information for different colour component. The HSI segmentation performs the segmentation according to the nature of the raw images. The raw image could be in varying sizes and resolutions, so it is crucial to first resize it to a fixed pixel width and height. In this work, the image is being resized to 200x200 pixels. Then the resized image is transformed from the original RGB colour space to the HSI colour space. Next, the system searches for pixels of interest. A pixel is marked if it is found to be in red colour using the following criteria, which have been obtained empirically:

$$\begin{cases} Either\ Hue < 0.027\ or\ Hue > 0.97 \\ Saturation > 0.6 \\ Intensity > 0.02 \end{cases} \tag{4.5}$$

The resulting image is then translated to a binary image with the pixels of interest being white whereas the rest being black. The binary image is next labelled according to the pixel's 8-connectivity (Shapiro & Stockman, 2001). Figure 4-6 shows the raw image is transformed into

HSI domain and subsequently the road sign object is extracted from binary image. To further narrow down the search for pixels of interest, a set of criteria is employed as follows to exclude objects that are too small (e.g. red traffic light) or too large (e.g. blocks of building, red vehicles or red soils):

$$\begin{cases} 200\ pixels < Area < 5000\ pixels \\ 0.4 < Aspect\ Ratio < 1.1 \end{cases} \tag{4.6}$$

We now consider the receptive field (RF) of each cortical cell consists of a central ON region (a region excited by light) surrounded by two lateral OFF regions (excited by darkness). Spatial frequency (W) determines the width of the ON and OFF regions.

The input image is in HSI domain and is projected into different Gabor wavelets to generate the output signals that resemble electrical signals in visual cortex. Different orientations and special frequencies produce different wavelets. After the convolution of input and wavelets, a set of feature vectors is formed that acts like the 'hypercolumns' as described by Hubel and Wiese (Hubel & Wiesel, 1962b).

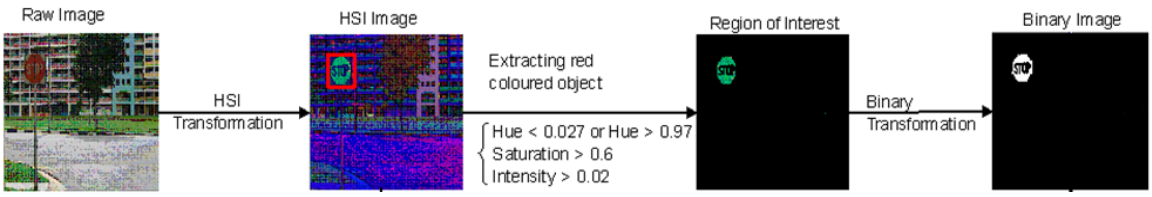


Figure 4-7 The intermediate pictures of road sign acquisition and extraction

We have developed two maps to recognize road signs. The first map is used to distinguish between road sign and non-road sign. The second map is used to classify the six categories of road signs. The input to the first map, namely *detecting map* (top map), is a convoluted image from the previous extraction model. The detecting map consists of 400 neurons, which is stimulated by feature vectors of image. $x_{top} = \{x_1, x_2, ..., x_n\}$ denotes the input vector to the top map and

$w_j = \{w_{j1}, w_{j2}, ..., w_{jn}\}$ denotes the weight vector of neuron j. The best match $i(x)$ is found to be the maximum inner product given by using:

$$i(x) = \arg\min \|x_{top} - w_j\| \quad (4.7)$$

The neuron with the minimum distance is the winning neuron. The winning neuron then locates the centre of a topological neighbourhood of cooperating neurons. We next adjust the synaptic weight vectors of all neurons using the updating formula.

$$w_j(n+1) = w_j(n) + \eta(n) h_{ji} [x - w_j(n)] \quad (4.8)$$

where h_{ji} is the Gaussian neighbourhood function and $\eta(n)$ denotes the learning rate at iteration *n*. Usually, the learning rate of training is controlled using the formula: $\eta = \eta_0 \exp\left(-\frac{n}{\tau}\right)$, where τ is the time constant and η_0 is the initial learning rate. The process of competition and weight adaptation continue until no further changes are observed.

Once the map is formed, we use knowledge representation to label the neurons. Physiologically, knowledge representation is a meta-cognitive function. It is an important step because recognition requires the represented knowledge to carry out recognition of road signs. Here we use colour information to represent road sign or non-road sign that a neuron would respond to. The neuron will be marked according to the smallest Euclidean distances between the weight and the input.

Recognition is supposed to be a higher cognitive process that is performed by another brain maps at a higher level. In this work, to determine if an input is road sign or non-road sign category C_i, where $i \in \{\text{road sign, non-road sign}\}$. We obtain the number of neurons X_i in K, the class is determined by taking the maximum of it $C_i = \max(X_i)$.

Once the input is categorized to be a road sign, we further process it with the second map at a higher level in the visual cortical processing system, namely *recognizing map*. The learning

algorithm of this map is similar to the previously mentioned methods. In the recognizing map, we have six classes correspond to different road signs, C_i, where,

$i \epsilon \{giveway, noLeftTurn, noRightTurn, speedLimit60, speedLimit\ 90, stop\}$.

4.2.3.2 Experimental Results for Road Sign Recognition

We consider six classes of road signs namely stop sign, give-way sign, no left turn sign (NLT), no right turn (NRT), speed limit 60 (SL60), and speed limit 90 (SL90). The database consists of road sign images. Fifty images are used for each category of road sign for testing. The result shows recalling capability of the system. Test image are first fed into the first map, to all the 400 neurons (20x20 map). If the input is acknowledged as a road sign, it then further enters the second map. The second map also consists of 400 neurons. Figure 4-7 shows some of the intermediate images produced during the process. The first column shows the raw images in RGB color space. Second column shows the images in HSI color space, we can see that the images still preserve some of the distinct properties like shape and numbers similar to RGB color space. Third column shows the Gabor wavelet in use. The last row presents the convoluted Gabor images.

The convoluted images are used to train the self-organizing maps. Figure 4-8 shows the two-tier maps formed by road sign images. The learning and adaptation details may be found in Section 3.3 in the earlier section of this thesis. In the top map, the fuchsia color represents road sign and yellow color represents non- road sign. Road sign neurons are centralized in the centre of the map, whereas the boundary edges are surrounded by non-road sign. The weight vectors of road sign neurons are used as the input vectors for the second map. The bottom map in Figure 4-8 shows the six categories of road signs learnt using the average of four Gabor wavelets. The “No left turn” road sign gives the best clustering result amongst the road signs.

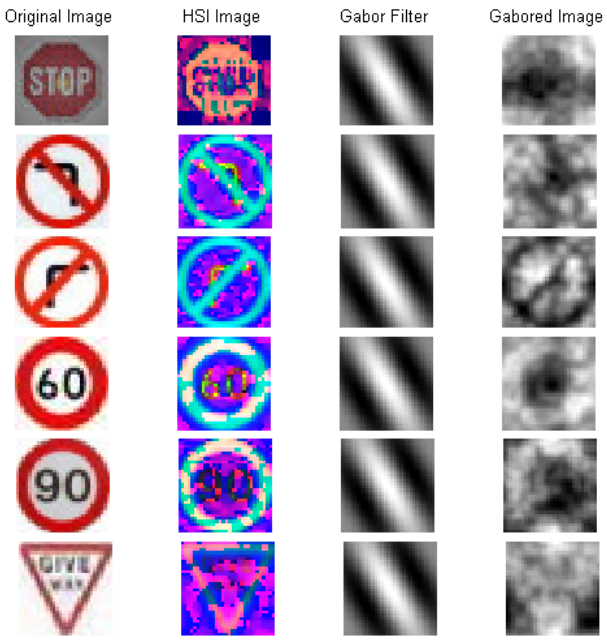


Figure 4-8 Intermediate images with one example of Gabor features generated

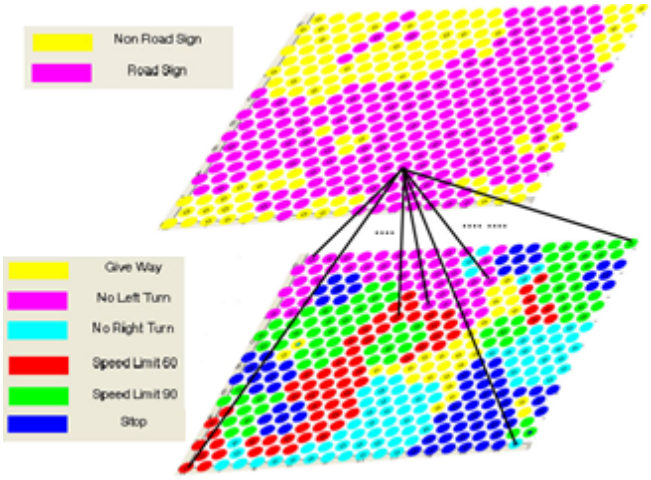


Figure 4-9 Two-tier maps with 4 Gabor wavelets

For the purpose of validation, we simulate the recognition map with a number of Gabor filters ranging from one Gabor up to 24 Gabors and the resulting road sign features would be used for visualization and classification by the self-organizing feature map. The visualization results are shown in Figure 4-9. The six road signs are labelled using different colours. The little circle shown inside each neuron represents the test output that maps to the corresponding neuron. The colour of

the little circle denotes the class of the test point. We have adopted 2D toroid structure with Euclidean distance for *recognizing map*. The *recognizing map* output is then shown to have connected boundary effect, for example the stop sign in deep blue from Figure 4-9b and the speed limit 60 in red that is distributed around the corners of the map. The *recognizing map* provides a low dimensional projection preserving the topology of the input space, thus the high dimensional distances can be visualized with the canonical U-Matrix, P-Matrix and U*-Matrix together so that the cluster boundaries can be distinguished easily. In addition, the visualization by *recognizing map* can be interpreted as height values on top of the conventional two dimensional grid of the SOM, leading to an intuitive paradigm of a landscape. Based on looking at the results, it seems that using the *recognizing map* with 4 Gabor filtering has the best visualization quality of mapping effects among the maps with other number of Gabor filters. Three of the road signs included *Give-way*, *No Left Turn* and *Speed Limit 90* are able to form closed regions on the map, whereas the other three road signs, i.e., *Stop*, *No Right Turn* and *Speed Limit 60*, formed more than one region in the map. In particular, the *Stop* and *Speed Limit 60* are often mapped in different and separate regions. It is because the *Stop* sign may contain outliers within both areas of the *Give-way* sign and *Speed Limit 90*. It is similar to the mapping of the *No Right Turn*, which contains some outliers within both areas of the *Give-way* and *Stop* signs. The mapping of the *Speed Limit 60* sign is even worse as it became outliers and noises by the other road signs so therefore it populated to four corner regions. This probably "confused" by the combination of Gabor and PCA feature extraction in which the extracted features of *Speed Limit 60* are quite similar to the other road signs so that it creates a rather random mapping. In summary, most of the road signs can be mapped by the *recognizing map* and the visualization results obtained by the *recognizing map* can help in recognizing and classifying consistent road signs in unlabeled datasets.

For classification task, labelling all (or most) neurons instead of only the best matches created a classification method based on unsupervised training which is similar to a k-nearest

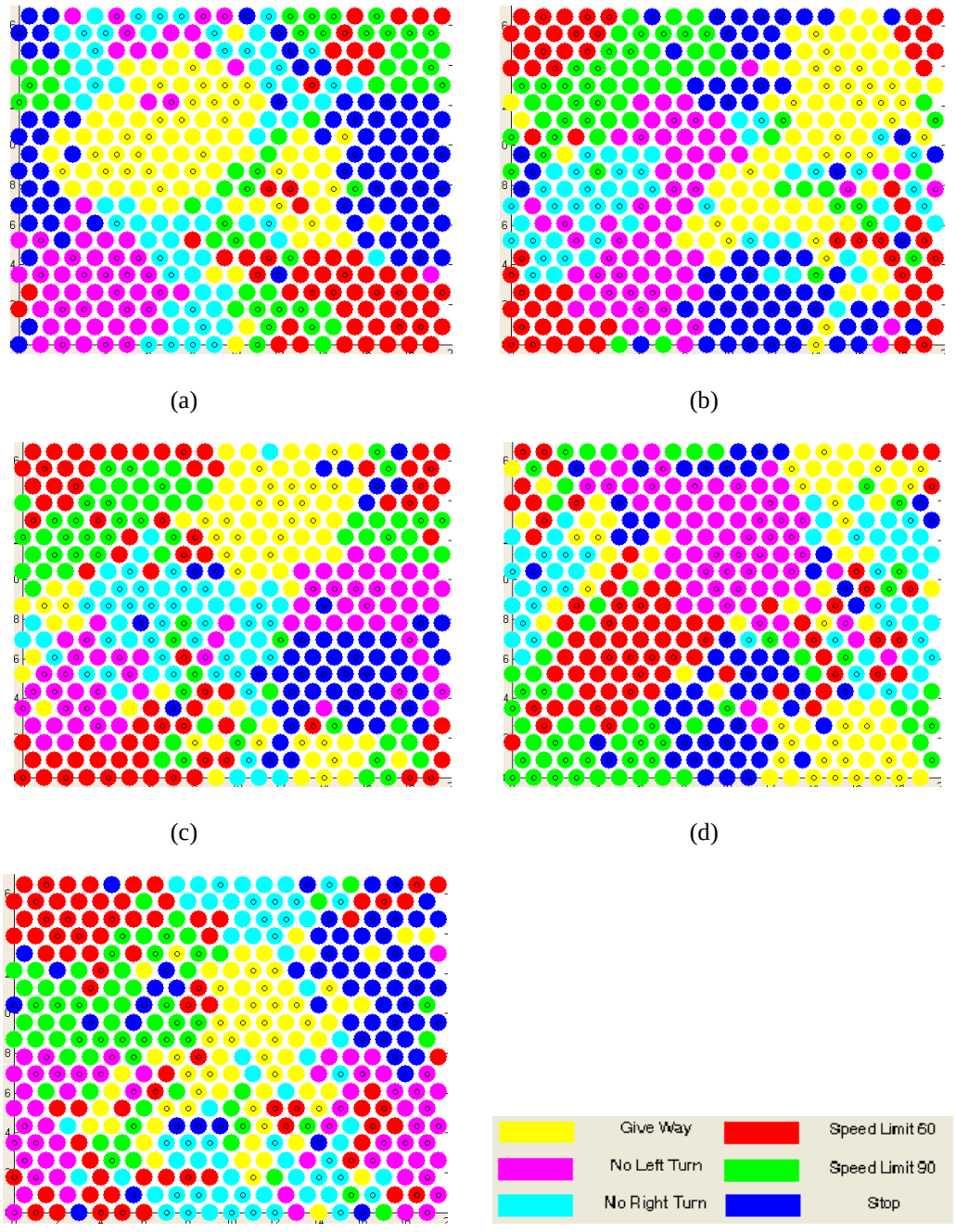


Figure 4-10 The Maps for Road Sign Images. (a) 1 Gabor used, (b) 4 Gabors used, (c) 8 Gabors used, (d) 12 Gabors used, (e) 24 Gabors used, (f) the color legend

neighbor (kNN) classifier with k=5 that can be applied to new data automatically. The main difference to kNN is that the user can use the visualization of the *recognizing map* to create the labelling whereas kNN does not give any visualization that could be used for this purpose. Further kNN classification always classifies a point, no matter how near (or far) the neighbors are. In contrast, *recognizing map* classification offers an unknown class by leaving neurons unlabeled, for example, for sparsely populated regions separating clusters.

Table 4-1 Road Sign Classification Results by Two-tier Map with Different Numbers of Gabor Filters Used

Number of Gabor filters used	Classification Rate (%)						
	Give Way	No Left Turn	No Right Turn	Speed Limit 60	Speed Limit 90	Stop	Average
1	100	100	90	72	64	44	78.3
4	100	98	86	96	78	82	90.0
8	100	100	78	92	76	58	84.0
12	100	92	80	82	82	66	83.7
24	100	96	72	88	60	74	81.7
Average	100	97.2	81.2	86	72	64.8	

Ten-folds cross validation has been conducted. Each fold contains eight images for each class. Table 4-1 summarizes the classification results of ten-folds cross validation obtained from the two-tier maps. There are five sets of results involved in the feature map. The first set uses only one Gabor wavelet to convolute all the input images. It gives an average hit rate of 78.3%. The second set use four Gabor wavelets and take PCA of the convoluted images and it produces average hit rate of 90.0%. The third set extracts from PCA image of eight wavelets and its hit rate is about 84.0%. The fourth set uses twelve Gabor wavelets; the hit rate is about 83.7%. The last set takes twenty-four Gabor wavelets and gives the hit rate, 81.7%. It is observed that the best hit-rate of 90% could be obtained by using 4 Gabor wavelets for feature extraction. Looking into individual road signs, the five classes of road signs are quite comparable excluding *Stop* sign in which about 65% hit-rate was obtained. Referring to the visualization results in Figure 4-9, we see

that the clusters of *Stop* sign are quite scattered around. It is due to the fact that the Gabored image of the *Stop* sign is unable to provide informative clue about the road sign.

In addition, a comparison with other approaches of existing road sign recognition systems would be necessary for us to investigate what the recognition performance of the proposed approach can be achieved. However, it is difficult to compare directly our results with the others since different research groups had conducted different types of experiments under different environments and databases used. Therefore, we discuss and compare here only the independent tests. Table 4-2 shows the recognition results of the different approaches for road sign recognition conducted in between 1999 to 2005. In this comparison, we are the only one group to adopt the unsupervised learning to this road sign image recognition, whereas the other models were learnt by supervised manner. According to the results shown, our SOM based approach is quite comparable with other techniques since the best one was used by Adaboost model which was able to achieve 98% hit rate, but only testing on 50 images. Our approach is able to achieve up to 90% hit rate with testing on 500 images. It is demonstrated that our result is quite encouraging and comparable.

Table 4-3 subsequently compares the performance of a few methods using road sign data with 1, 4, 8, 12, and 24 Gabor filters. As with the case of ESOM, the approach achieves higher accuracy compared to Naïve Bayesian, Adaboost and Bayes Net. The highest accuracy goes to J48 decision tree (92.33%), as this is a very well established approach. Decision tree resembles supervised approach to classification that test on one or more features to reach a decision. The average classification accuracy of ESOM with Gabor is 83.54% using Gabor filters ranging from 1 to 24. The accuracy for Naïve Bayesian and Bayes Net are similar, 82.33% and 82% respectively. The 1R classifier may not be suitable for this problem that causes its average poor result of 53.67%. The features with 4 Gabor filters are shown to have the best result throughout the experiments with other methods.

Table 4-2 Comparison with other road sign recognition approaches

References	Techniques used	Database nature	Performance
(Soetedjo & Yamada, 2005)	Ring Partitioned	180 test images, circular road signs (No entry, speed limit signs only)	93.9%
(Escalera & Radeva, 2004)	Adaboost and model matching	21 classes, 50 test images	98% on 50 test images
(Torresen et al., 2004)	Template Matching	7 classes of speed limit signs, 198 images	90.9%
(Gao *et al.*, 2002)	Behavioural Model of Vision	41 British road sign images	88~90%
(Vitabile, Gentile, & Sorbello, 2002)	MLP neural networks	620 images of 24 classes circular signs	~90% of hit rate
(Paclik, Novovicova, Pudil, & Somol, 2000)	Laplace Kernel classifier	1200 images of 50 classes	~95% of hit rate
(Gavrila & Philomin, 1999)	Distance Transform Matching	1000 traffic sign images	~95%
Our approach	ESOM based with Gabor features	480 images, 6 classes running under 10-folds cross validation	78.3~90%

Table 4-3 Benchmarking with other methods using road sign data

Method	Settings	1 G	4 G	8 G	12 G	24 G	Avg	High	Var
Naïve Bayesian	Default	67.00%	82,33%	71.00%	69.00%	72.33%	69.83%	82.33%	0.041
1R Classifier	Bucket Size=6	52.33%	58.67%	53.67%	52.67%	51.00%	53.67%	58.67%	0.070
Bayes Net	K2 Search Algorithm	70.33%	82.00%	79.00%	73.67%	70.67%	75.13%	82.00%	0.214
J48 Decision Tree	Tree Size=39; Leaves=20	88.67%	92.33%	91.33%	90.00%	91.67%	90.80%	92.33%	0.017
SVM	Default	69.8%	74.5%	73.4%	73.3%	72.5%	72.7%	74.5%	0.013
Our Model	ESOM with Gabor	78.30%	90.00%	84.00%	83.70%	81.70%	83.54%	90.00%	0.146

4.2.5 Simulation II – Emotion Recognition

The emotion recognition research has seen large numbers of published work centering on facial expressions recognition. Facial expressions represent a straightforward means of expressing the emotion of a person. A facial expression is formed by contracting or relaxing different facial muscles on human face which results in temporally deformed facial features like wide open mouth, raising eyebrows etc.

Emotions positively affect intelligent functions such as decision making, perception and empathic understanding (Bechara, Damasio, & Damasio, 2000 ; Isen, 2000). Emotion is a state of feeling involving thoughts, physiological changes, and an outward expression. There are five theories which attempt to understand the sequence of processes that we are experiencing when we are feeling certain type of emotion. They are James-Lange theory (K. Z. Mao, 2004), Cannon-Bard theory (Park & Han, 2004), Lazarus theory (Hermes & Buhmann, 2000), Schachter-Singer theory (Chang & Lin, 2008), and Facial Feedback theory(Informatique et al., 2008). According to the facial feedback theory (Christopher L), emotion is the experience of changes in our facial muscles. In other words, when we smile, we experience pleasure or happiness. When we frown, we experience sadness. It is the changes in our facial muscles that direct our brains and provide the basis for our emotions. As there are many possibilities of muscle configurations in our face, there is seemingly unlimited number of emotions. Figure 4-10 shows the facial expressions which are formed by contracting or relaxing the facial muscles from different part of face like eyebrows, upper eyelids, cheeks and lips. The characteristics of facial expressions are summarized in Table 4-4.

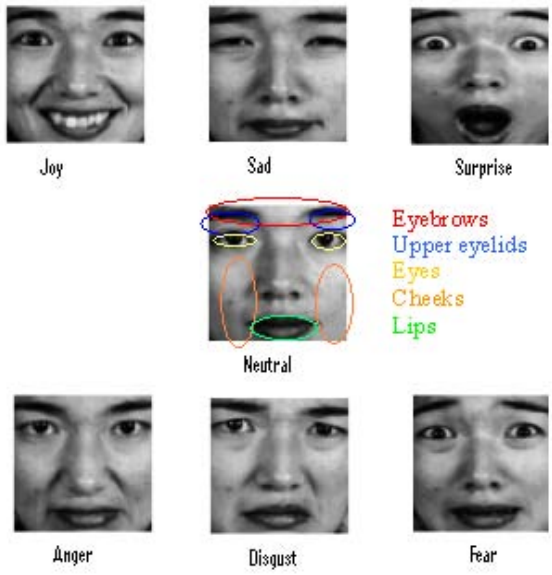


Figure 4-11 Facial Expressions Evolved from Neutral Face

Table 4-4 Facial Cues and Emotions

Emotions	Observed Facial Cues
Surprise	Brows raised
	Upper eyelid stretched
	Horizontal Wrinkles across forehead
	Eyelids Opened
	Jaw drops open without tension
Fear	Brows raised
	Forehead wrinkles drawn to the centre
	Upper eyelid raised
	Mouth open
	Lips are slightly tensed
Disgust	Upper lip raised
	Nose is wrinkled
	Cheeks are raised
	Brows are lowered
Anger	Brows lowered
	Vertical lines appear between brows
	Lower lid is tensed
	Lips are pressed firmly together
	Nostrils may be dilated
Happiness	Corners of lips are drawned back and up
	Mouth may not parted with teeth exposed
	Cheeks are raised
	Lower eyelids show wrinkles below it
Sadness	Inner corners of eyebrows are drawn up
	Skin below the eyebrow is triangulated
	Upper lid inner corner is raised
	Corners of lips are drawn or lip is trembling

Most people are able to interpret emotion expressed by others all the times, but there are people who lack this ability, such as people diagnosed with the autism spectrum (Baron-Cohen, 1995). The first known facial expression analysis was presented by Darwin in 1872 (Darwin, 1872). He presented the universality of human face expressions and the continuity in man and animals. He pointed out that there are specific inborn emotions, which originated in serviceable associated habits. After about a century, Ekman and Friesen (Ekman & Friesen, 1971) postulated six primary emotions that possess each a distinctive content together with a unique facial expression. These prototypic emotional displays are also referred to as basic emotions in many of

the later literature. They seem to be universal across human cultures and are namely happiness, sadness, fear, disgust, surprise and anger. They developed the Facial Action Coding System (FACS) for describing facial expressions. It is an appearance-based approach. FACS uses 44 action units (Bauer & Pawelzik) for the description of facial actions with regard to their location as well as their intensity. Individual expressions may be modeled by single action units or action unit combinations. FACS codes expression from static pictures. FACS is an anatomically oriented coding system, which is based on the definition of Action Units (Bauer & Pawelzik) of a face causing facial movements. Each AU may correspond to several muscles that generate a certain facial action. Forty-six AUs were considered responsible for expression control and twelve AUs were assigned for gaze direction and orientation. The AU codes were assigned to the action of muscles specific to certain portion of the face. Samples of AU codes are presented in Figure 4-10. Different combinations of AU codes result in different emotion expression. For example, the code AU4 of FACS was assigned to the action of lowering the eyebrows and pulling them together. We can find the muscle motion code AU4 in the expressions of anger, sadness, fear or disgust. Another example, the AU combination 1+2 was assigned to the action of lifting the eyebrows up. This AU combination results in the wrinkling in the forehead which indicates emotion surprise. The AU5 can be obtained by widening the eye aperture and raising the upper eyelid so that some or the entire upper eyelid disappears from view. As a result, more of the upper portion of the eyeball is exposed.

Following Ekman, more works evolved during the nineties which include the use of Multi-Layer Perceptron Neural Networks (Cottrell & Fleming, 1990), optical flow estimation (Mase & Pentland, 1991), 2D Potential nets (Matsuno, Iee, & Tsuji, 1994), Radial basis function network (Rosenblum, Yacoob, & Davis, 1996). Essa and Pentland (Essa & Pentland, 1997) further extend the FACS and developed a FACS+ model to represent facial expression. More recent works include Ye et. al. 2004, who use Gabor transformation to form elastic facial graph,

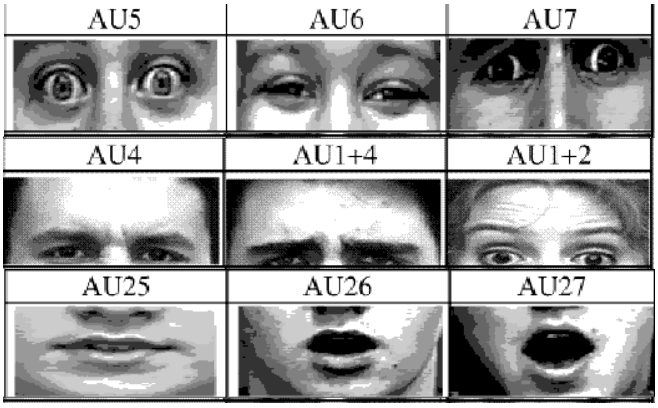


Figure 4-12 Example of AU coding to upper, middle and lower parts of face

Dynamic Bayesian Networks (DBM) with FACS by Ji (Ji, 2005), Xiang et. al. (T. Xiang, 2007) utilized fourier transform, fuzzy C means to generate a spatio-temporal model for each expression type. Their recognition rates are in the range of 80~93%.

The performance of existing techniques is still inconsistent and often results in vastly different results as perceived by human visual perception. Many researchers have explored geometrical features based method for face recognition. Kanade (Kanade, 1973) presented an automatic feature extraction method based on ratios of distance and reported a recognition rate of between 45-75% with a database of 20 people. Brunelli and Poggio extracted a set of geometrical features (Brunelli & Poggio, 1993) using independently matching templates in the 3 key fiducial points such as nose width and length, mouth position, and chin shape in which this method could achieve 90% recognition rate on a database of 47 people. Although many facial recognition techniques have been able to deliver promising results, the task of robust face recognition remains very difficult (Ning et al., 1999). Indeed, there are two major problems in the current approaches: the problem of illumination variation and the pose variation (Han & Yang, 2004). Either of these two problems may cause significant degradation in the performance of facial recognition systems. The illumination problem is basically illustrated by the same face appearing differently due to a

variation in lighting. The differences introduced by varied illuminations often caused systems to misclassify input images that have been theoretically proved by Adini et al. (Casti, 1989)for systems based on Eigenface projection. Another problem is pose variation. The performance of face recognition may also drop significantly when pose variations are presented. Some works have been proposed to handle the pose problem, such as including multiple images of each person by using a template-based correlation matching scheme , or using hybrid methods (Anderberg, 1973) when multiple images are available during training but only one database image per person is available during recognition. All these methods are using holistic features of the entire face images for recognition but may not be practical as large amount of computational cost and storage are required for large database applications.

Most facial recognition systems use geometrical features or fixing area of interest to represent facial images. They attempt to recognize facial expression in the general circumstances where the training and testing data do not involve imbalanced inclined data. There is lacking of literatures that attempt to address this imbalanced issue.

4.2.5.1 Framework for Emotion Recognition

A computer based cognitive system to recognize emotions portrayed by human face has been implemented. This work attempts to emulate or model the cognitive processes of the human brain that are employed during emotion recognition. The system investigated gray-scaled images.

Figure 4-12 shows the architecture of the entire system and how it maps to human visual system. The system comprises of 2 parts, feature extraction part and SOM adaptation part. Part 1 models the cognitive processes of feature extraction. It extracts essential information using Gabor wavelets and passes the information to the next level. Part 2 models the storing of information through the use of biologically similar map achieved through competitive learning.

Four procedures are involved in the whole process. Image features are first extracted using Gabor Wavelets. Secondly, the feature vector undergoes usupervised adaptation using TtEsom. The third process involve knowledge representation, different emotion category is labeled according to the input marked neurons. The last step is the recognition of image through K-Nearest Neighbour (Knn).

The face image can be captured from video or static image. The first step in the whole process is to use face detection technique to track the location of the face and crop the face as the area of interest. Bradski (Bradski, 1998) has developed a face tracking model base on the color probability mean shift. The face detection step provides the face boundary suitable for further investigation. If the quality of the image is poor which renders face detection difficult, image processing techniques like smoothing filter can be used to enhance the quality of image. The next step would be to crop the face image from the background.

Meaningful information is hidden underneath the image. Proper selection of features optimizes the performance of classification. Feature extraction forms the basic building block of recognition problem. We choose to use Gabor wavelets to convolve the cropped face image.

Gabor wavelets has the capability to capture the properties of spatial localization, orientation selectivity, spatial frequency selectivity, and quadrature phase relationship, which makes it a good approximation to filter response profiles encountered experimentally in cortical neurons (C. Liu & Wechsler, 2002). It is these properties that make it a natural choice for modelling the receptive part of human visual cortex. The receptive field of cortical cell consists of a central ON region surrounded by 2 OFF regions, each region elongated along a preferred orientation.

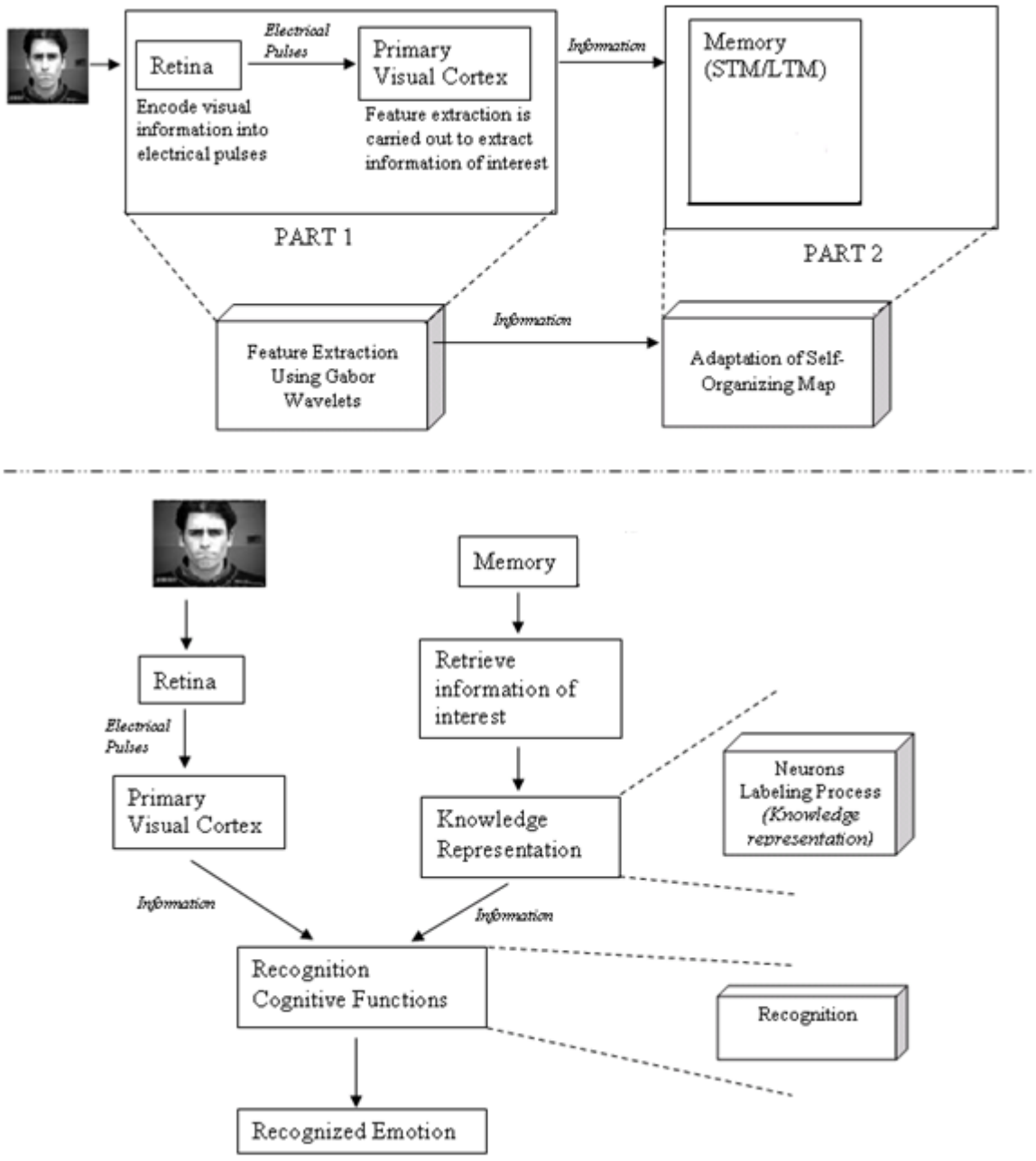


Figure 4-13 Mapping of Emotion Recognition System Design and Cognitive Processes

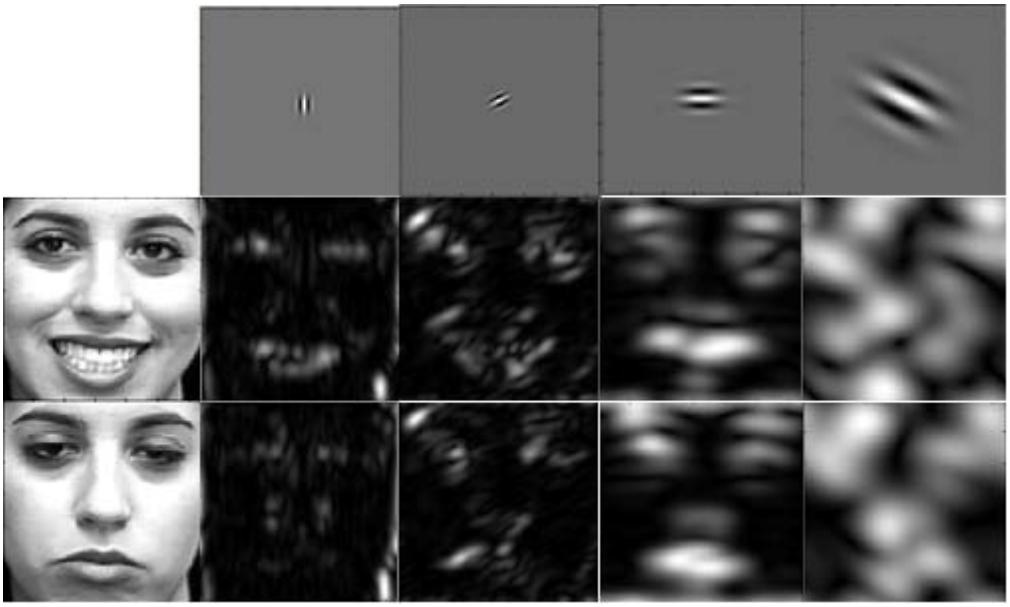


Figure 4-14 Gabor wavelets representations

We then use an image $I(x,y)$ convolute with the Gabor wavelet which is defined as follows (Bernhard E. Boser):

$$W_{mn}(x,y) = \int I(x_1,y_1) g_{mn} * (x - x_1, y - y_1)\, dx_1\, dy_1 \tag{4.9}$$

The subscript m denotes the size of the filter bank in terms of number of iteration. The localized Gabor features are then formed. The feature set F_i for C_i is the resultant of convoluting the mean and standard deviation of original image. They are represented by:

$$\mu_{mn} = \iint |F_{mn}(x_F, y_F)|\, dx\, dy \tag{4.10}$$

$$\sigma_{mn} = \sqrt{\iint (|F_{mn}(x_F, y_F)| - \mu_{mn})^2}\, dx\, dy \tag{4.11}$$

And the features are described as:

$$F_i = [\mu_0, \sigma_0, \mu_1, \sigma_1, \dots, \mu_i, \sigma_i] \tag{4.12}$$

We convolute the face image with the Gabor filters so as to extract the facial features. We have chosen 6 orientations and 4 spatial frequencies; generating a total of 24 Gabor filtered images. The lower bound frequency is chosen as 0.05 while the upper bound frequency is chosen to be 0.4. Orientations are in multiples of π/6 from 0 to π. Figure 4-13 shows the responses of two different

facial images for four of the selected Gabor filters. All the convoluted images model the data received by the primary visual cortex (area V1). Since the processing at the retina and LGN cortical layers are used by Gaussian kernels applying to some spectral bands to simulate local inhibition, most stimuli resemble Gabor wavelets. Since the processing at the retina and LGN cortical layers are used by Gaussian kernels applying to some spectral bands to simulate local inhibition, most stimuli resemble Gabor wavelets. The mean Gabor image is then generated and form the major feature vector for the next stage to process.

4.2.5.2 Emotion Map Formation

After the feature vectors are obtained from the face images, face and emotion recognition classifiers are required to fully utilise the selected features. ESOM compresses the topological information of facial expression images using narrow mapping space and perform classification based on features. Several studies (Jabbi, Swart, & Keysers, 2007; Wittling & Roschmann, 1991) have shown the presence of positive and negative affections map in the cognition level. Jabbi et al. (2007) have shown the correlation between the positive and negative emotion in the hemispheres as displayed in Figure 4-14. Figure 4-14 shows that positive correlation exist between the composite Interpersonal Reactivity Index (IRI) (Yan et al., 2003) and the parameter estimates obtained during the vision of disgusted facial expressions, in both the right and left gustatory Empathy for positive and negative emotions in the gustatory cortex (IFO) (Jabbi et al. 2007). In the left IFO, similar correlations were observed between the vision of pleased facial expressions and the composite IRI. Correlations with neutral facial expressions were smaller, and restricted to the right hemisphere. More information can be referred to from Jabbi et al. 2007.

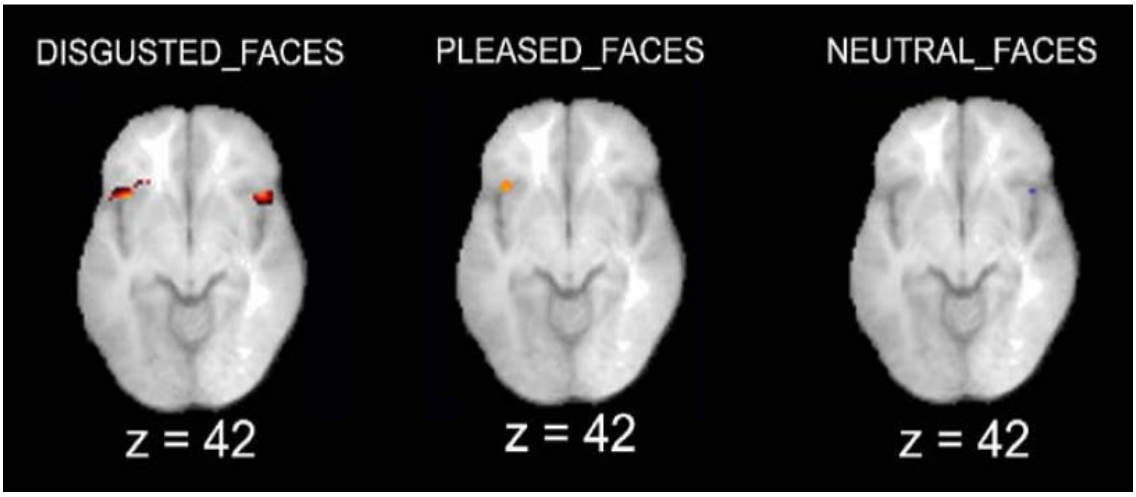


Figure 4-15 Scores for positive (pleased_faces) and negative (disgusted_faces) affects (Jong et al., 2004)

We employ 2 maps that are interconnected to classify emotions, the affection map (A-Map) and emotion map (E-Map). The input vector $\{f_1, f_2, \dots, f_p\}$ represents the mean Gabor image feature vector of the entire face of length *P* (i.e. the size of an image). The input is first presented to the F-Map. The procedures are similar to Algorithm 4.1:

The proposed method used the representative image from A-Map and carry out learning based on patterns between each emotion category. The proposed method was used in an attempt to extract facial expression category hierarchically using emergent self organizing map with a narrow mapping space. The approach classifies given facial expression images based on their topological characteristics. Figure 4-15 depicts the visualization of A-Map and E-Map. The representative face images obtained from A-Map were used as teaching signal and input data which form the training data for E-Map. The facial expression topological characteristics are learned using Algorithm 3.2. The Gabored values of the representative images were used as input data. The weight in each unit of E-Map is compared upon learning is completed. An emotion category of the greatest value was used as the label of the unit. The proposed method has a decreasing neighhborhood region that gives rise to winner node once the neighbourhood region is 1. The

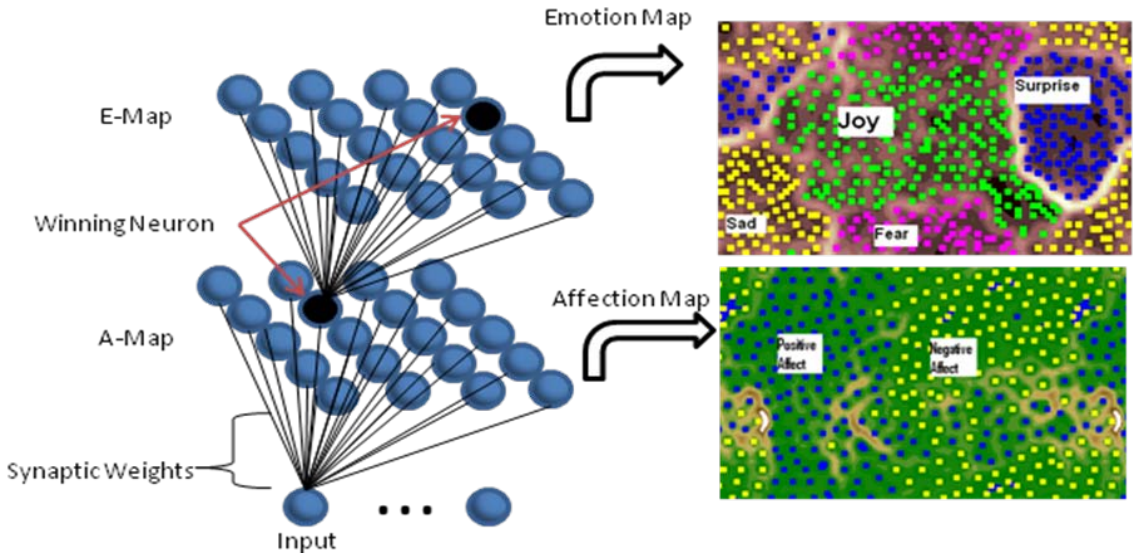


Figure 4-16 Structure of Two-tier ESOM

learning coefficients $\alpha(t)$ and $\beta(t)$ were defined to slowly decrease from the initial value of 0.5 to 0.01.

4.2.4.3 Experimental Results for Emotion Recognition

We have adopted datasets from two sources: CMU's Cohn-Kanade AU-coded Facial Expression database (Kanade, Cohn, & Tian, 2000; The_face_research_group)and JAFFE database (Lyons, Akamatsu, Kamachi, & Gyoba, 1998). They are described in more details in the following paragraphs.

CMU database consists of images of approximately 100 subjects. Facial images are of size 640x490 pixels, 8-bit precision grayscale in png format. Subjects were 100 university students enrolled in introductory psychology class. Age ranges from 18 to 30. Sixty-five percent were female, 15 percent were African-American, and three percent were Asian or Latino. Subjects were instructed by experimenter to perform a series of facial displays. Subjects began each display from a neutral face. Before performing each display, the experimenter described and modeled the desired display. Six of the displays were based on descriptions of basic emotions (joy, surprise,

anger, fear, disgust, and sadness). There are 2 different datasets derived from CMU. One set contains 6 different emotions, and the other contains 5 different emotions excluding the surprise emotion.

The JAFFE database contains 213 images of 7 facial expressions (6 basic facial expressions + 1 neutral) posed by 10 Japanese female models. Each image has been rated on 6 emotion adjectives by 60 Japanese subjects. The database was planned and assembled by Miyuki Kamachi, Michael Lyons, and Jiro Gyoba. The photos were taken at the Psychology Department in Kyushu University.

As shown in Figure 4-16, the first row CMU5 shows the dataset of 5 facial expressions and the second row CMU6 shows the dataset with 6 facial expressions from CMU database. The snapshots of the JAFFE dataset are shown in the last row of Figure 4-16. A subset of the datasets is selected to be used in our experiments as shown in Table 4-5. CMU5 does not provide surprise emotion. The number of subjects and images are shown for each category.

The experiments were carried out using four emotions, namely fear, joy, sad, and surprise. Two emotions are positive emotions, joy and surprise; fear and sad are negative emotions. The experiments are carried using the cognitive mapping model described previously. It starts with grey property of the raw images, and goes through pre-processing. After pre-processing, the input vectors are fed into the network and projected into the labelled neurons. Neuron with the most similar weight as the input vector would be the best-match or winning neuron and its label would tell the category of emotions each input vector belongs to. The performance of the network is measured by comparing the result from identification with the predefined label.

Two types of experiments are carried out. The first type uses one layer of Emergent Self-Organizing. The emotion is categorised into one of the four classes as shown in Figure 4-17 after pre-processing. As shown in Figure 4-18 is the visualizations of 1-tier emotion map. The clusters are well separated. The second type of experiment uses two tier maps. The first tier represents the

affection map that separates out emotion into either positive or negative affections. If it is a positive affection, it will enter the second layer of positive map that further categorises it into joy or surprise. On the other hand, if it is a negative affection then it will enter the second layer of negative map that further categorises it into sad or fear. They are shown in Figure 4-19.

The results of ESOVPM are shown in Table 4-6 is the experimental result using 1-tier map on the three test sets, CMU5, CMU6 and JAFFE. One-tier categorises the four emotions directly, so the positive and negative results are not available. The recalling and recognition abilities of the model are assessed. The test obtains good recalling rate and average result on recognition rate. Table 4-7 shows the experimental result using 2-tier maps. The CMU5 dataset does not come with surprise emotion, so the results of positive map are not available. The first layer is trained with three emotions, joy, fear, and sad excluding surprise. The recognition results are seen improving. Table 4-8 shows the benchmarking result. Majority of the models achieve recognition rates of about 80%. Our approach of using two-tier emotion maps achieves average recognition rates of 85%.

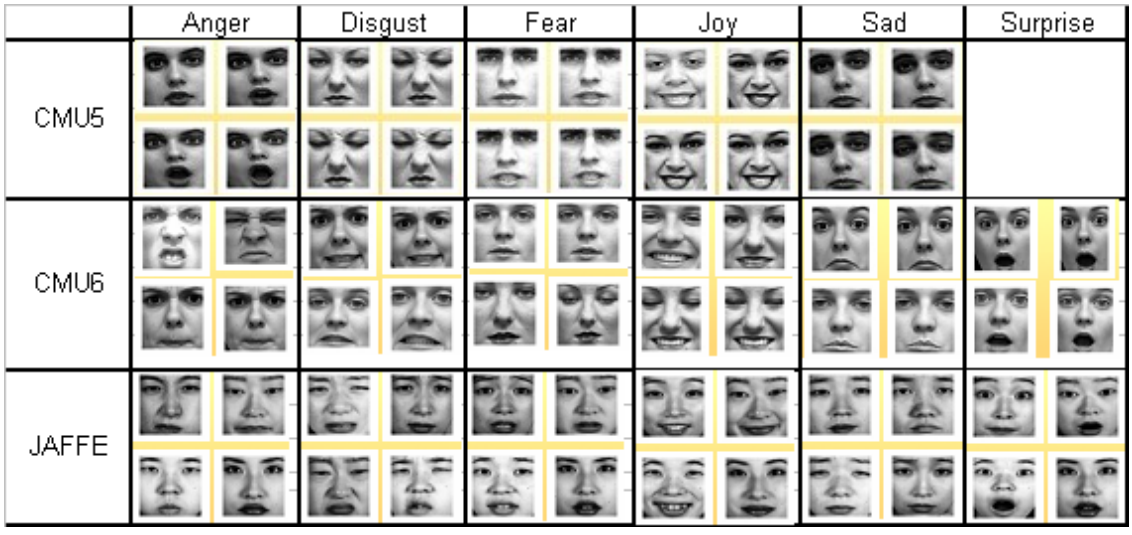


Figure 4-17 Images of different facial expressions taken from the three datasets

Table 4-5 Training and testing samples in use for the experiments

Database	Emotion	Training		Testing	
		Subject #	Image #	Subject #	Image #
CMU 5	Joy	6	71	6	6
	Surprise	-	-	-	-
	Fear	7	62	7	15
	Sad	8	61	8	16
CMU 6	Joy	6	11	6	11
	Surprise	7	7	7	7
	Fear	4	11	4	8
	Sad	4	16	4	10
JAFFE	Joy	10	21	10	10
	Surprise	10	20	10	10
	Fear	10	22	10	10
	Sad	10	21	10	10

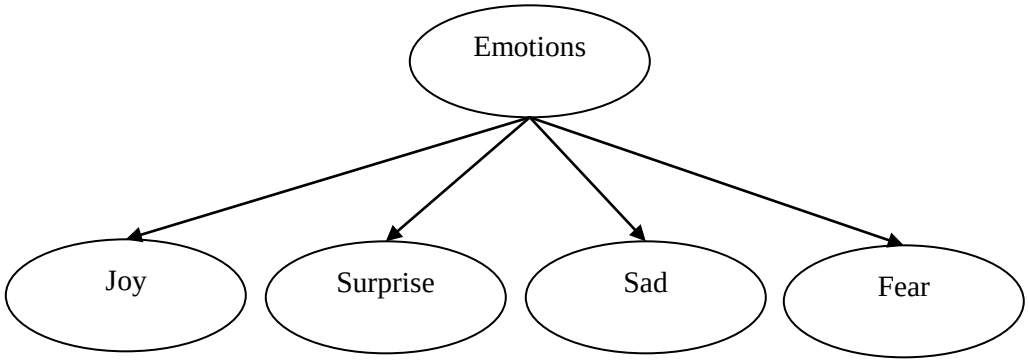


Figure 4-18 One level categorisation of four classes emotions

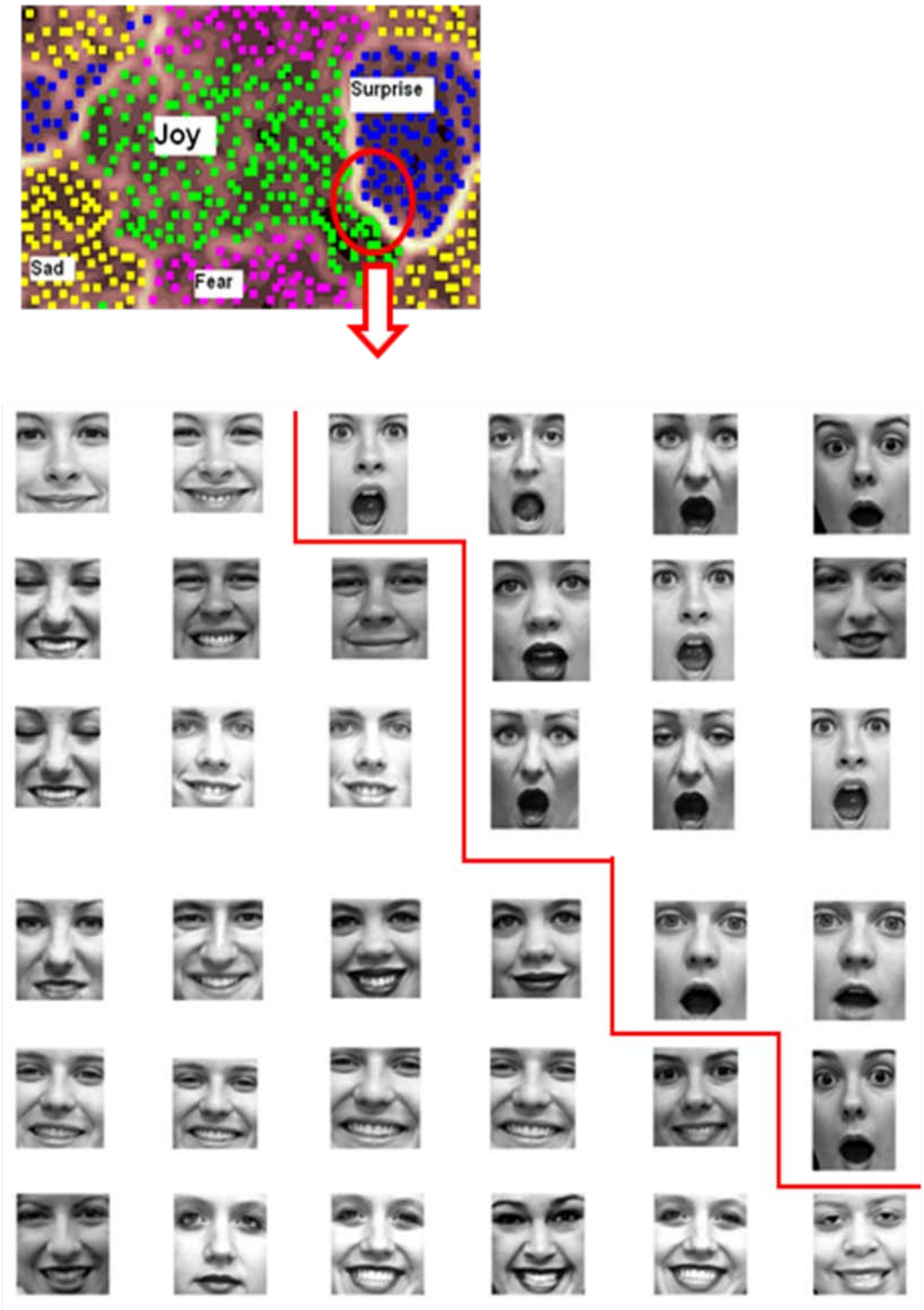


Figure 4-19 Visualization of 1-tier E-Map with CMU6 data

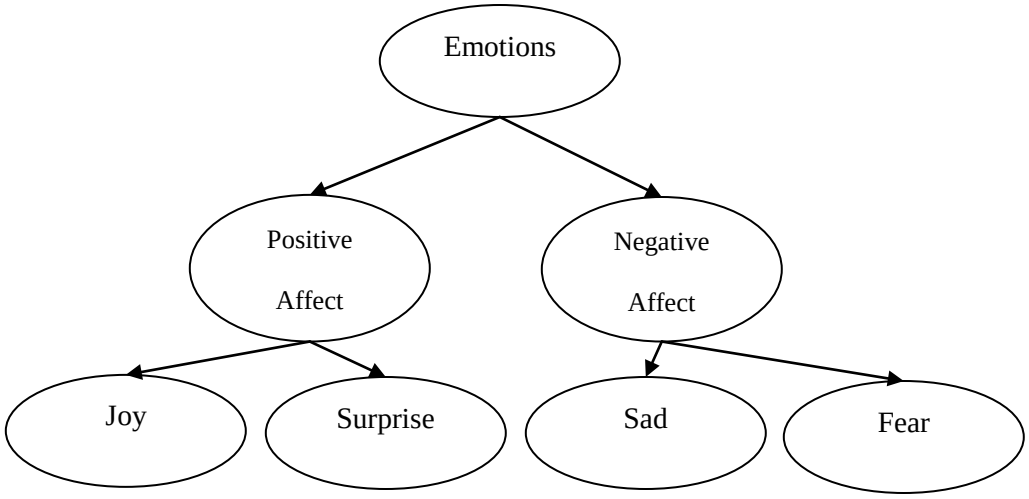


Figure 4-20 Two level categorisation of four classes emotions

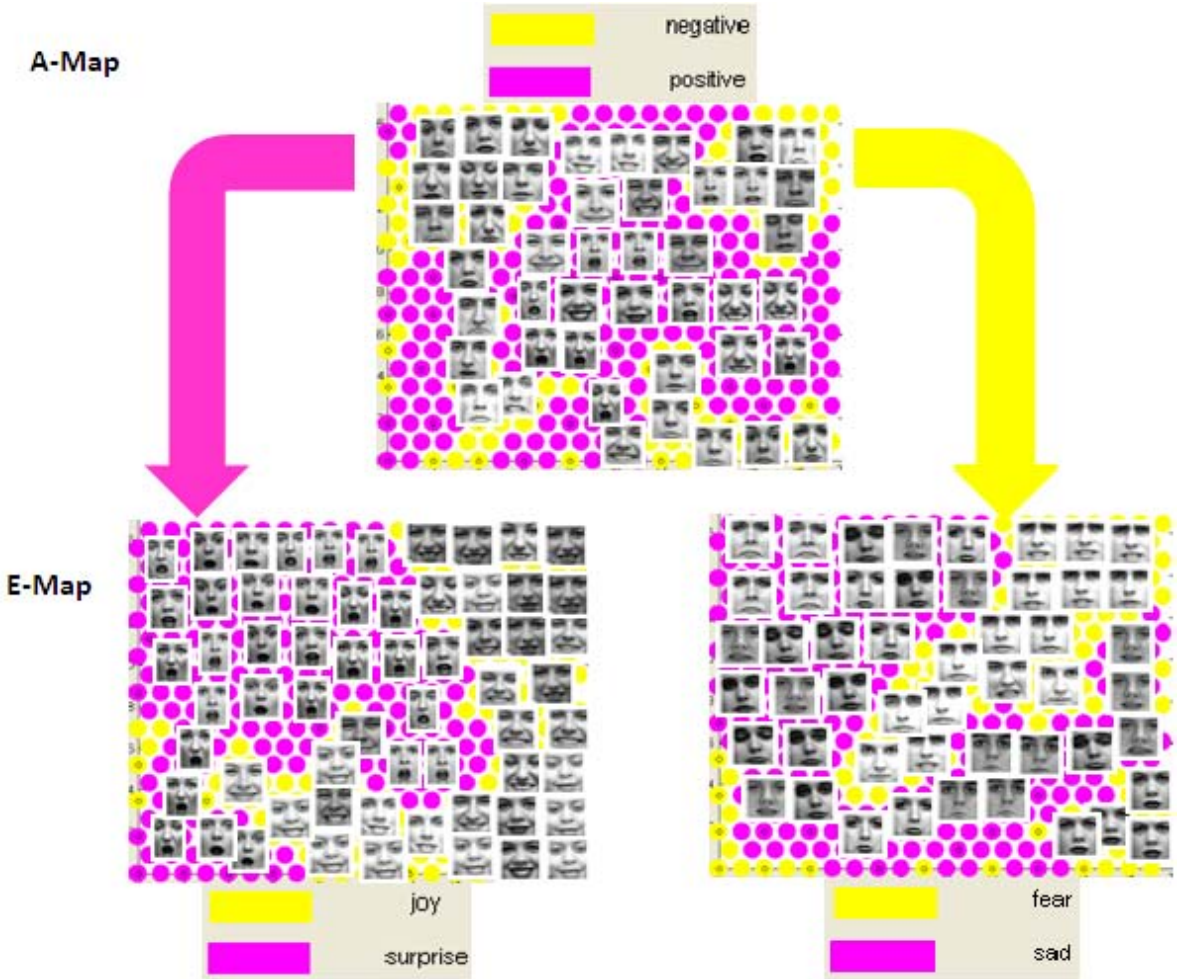


Figure 4-21 Visualization of A-Map + E-Map with CMU6 data

Table 4-6 Recalling and recognition results with Emergent Self-Organizing Map

Dataset	Emotion	Recalling			Recognition		
		Hit	Total	Hit Rate	Hit	Total	Hit Rate
CMU5	Negative	-	-	-	-	-	-
	Positive	-	-	-	-	-	-
	Joy	71	71	100.00%	6	6	100.00%
	Surprise	-	-	-	-	-	-
	Fear	62	62	100.00%	14	15	93.33%
	Sad	60	61	98.36%	15	16	93.75%
CMU6	Negative	-	-	-	-	-	-
	Positive	-	-	-	-	-	-
	Joy	17	17	100.00%	9	11	81.82%
	Surprise	13	13	100.00%	6	7	85.71%
	Fear	11	11	100.00%	6	8	75.00%
	Sad	15	16	93.75%	9	10	90.00%
JAFFE	Negative	-	-	-	-	-	-
	Positive	-	-	-	-	-	-
	Joy	19	21	90.48%	4	10	40.00%
	Surprise	17	20	85.00%	6	10	60.00%
	Fear	22	22	100.00%	7	10	70.00%
	Sad	20	21	95.24%	4	10	40.00%

Table 4-7 Recalling and recognition results with Self-Organizing Map

Dataset	Emotion	Recalling			Recognition		
		Hit	Total		Hit	Total	
CMU5	Negative	-	-	-	-	-	-
	Positive	-	-	-	-	-	-
	Joy	71	71	100.00%	5	6	83.33%
	Surprise	-	-	-	-	-	-
	Fear	62	62	100.00%	6	15	40.00%
	Sad	60	61	98.36%	10	16	62.50%
CMU6	Negative	-	-	-	-	-	-
	Positive	-	-	-	-	-	-
	Joy	17	17	100.00%	2	11	18.18%
	Surprise	13	13	100.00%	6	7	85.71%
	Fear	11	11	100.00%	5	6	83.33%
	Sad	16	16	100.00%	4	10	40.00%
JAFFE	Negative	-	-	-	-	-	-
	Positive	-	-	-	-	-	-
	Joy	21	21	100.00%	1	10	10.00%
	Surprise	20	20	100.00%	4	10	40.00%
	Fear	22	22	100.00%	0	10	0.00%
	Sad	21	21	100.00%	3	10	30.00%

Table 4-8 Recalling and recognition results with 2-tier A-Map + E-Maps

Dataset	Emotion	Recalling			Recognition		
		Hit	Total	Hit Rate	Hit	Total	Hit Rate
CMU5	Negative	123	123	100.00%	31	31	100.00%
	Positive	65	71	91.55%	3	6	50.00%
	Joy	-	-	-	-	-	-
	Surprise	-	-	-	-	-	-
	Fear	62	62	100.00%	14	15	93.33%
	Sad	61	61	100.00%	16	16	100.00%
CMU6	Negative	27	27	100.00%	16	18	88.89%
	Positive	29	30	96.67%	16	18	88.89%
	Joy	11	11	100.00%	17	17	100.00%
	Surprise	7	7	100.00%	13	13	100.00%
	Fear	11	11	100.00%	8	8	100.00%
	Sad	16	16	100.00%	10	10	100.00%
JAFFE	Negative	43	43	100.00%	14	20	70.00%
	Positive	36	41	87.80%	15	75	20.00%
	Joy	21	21	100.00%	7	10	70.00%
	Surprise	19	20	95.00%	7	10	70.00%
	Fear	22	22	100.00%	7	10	70.00%
	Sad	21	21	100.00%	7	10	70.00%

Table 4-9 Emotion Recognition Benchmarking

Facial Expression Recognition	Database Nature	Numbers of testing images	Recognition rates
CNRS, France (Wittling & Roschmann, 1991)	10 subjects and 7 recognized emotions	70	83.3%
Concordia University, Canada (Jabbi et al., 2007)	60 subjects and 4 recognized emotions	80	83.8%
Southeast University Nanjing, China (Zheng et al., 2006)	JAFFE (10 subjects and 6 emotions)	183	77.1%
	EKMAN (14 subjects and 6 emotions)	96	79.2%
University of Wisconsin-Madison (Hecht-Nielsen, 1988)	JAFFE (10 subjects and 6 emotions)	213	81.0%
Peking University, China (Wu et al., 2005)	10 subjects and 6 emotions recognized	183	83.2%
Our model (A+E Map)	**CMU + JAFFEE (20 subjects and 4 emotions)**	**141**	**85%**

4.3 Conclusion

This chapter demonstrated the use of Gabor wavelet to model the pre-cortical process. As we have mentioned earlier, feature selection methods based on convolution are effective in translating problems into another meaningful domain, but there is a lack of effective method to prioritise these convoluted features according to its discriminative ability. The next chapter introduces the use of Support Vector Machine to develop the prototype ranking algorithm. Firstly, statistical learning theory is briefly described. Statistical learning theory forms the basis for support vector machine. The framework of prototype ranking based on Support Vector Machine is then presented.

Chapter 5 Prototype Ranking Based for Feature Selection

5.1 Statistical Learning Theory

SVM is based on statistical learning theory (SLT) (V. Vapnik, 1995; V. N. Vapnik, 1998), which has its roots in the 1960s. SLT is a mathematical framework for estimating dependencies from finite samples which can be used to solve pattern recognition problems. This theory combines fundamental concepts and principles related to learning, well-defined problem formulation, and self-consistent mathematical theory. The SLT explains learning process from statistical point of view. The general setting of the learning problem can be formulated as (X. Wang & Zhong, 2003): Variant x and y exist an unknown dependent relationship with probability $P(x, y)$. The goal is to find the most optimised function $f(x, w_0)$ in a given set of functions $\{f(x, w_0)\}$, which minimizes the risk function $R(w)$.

$$R(w) = \int L\big(y, f(x, w)\big) dP(x, y) \tag{5.1}$$

where w denotes the generalized parameter of functions, $L(y, f(x, w))$ is the loss caused by using $f(x, w)$ to predict y. The risk function $R(w)$ can be replaced by empirical risk function as based on ERM principle:

$$R_{emp}(w) = \frac{1}{t}\sum_{i=1}^{t} L(y_i, f(x_i, w)) \quad (5.2)$$

The ERM is intended for large sample size which can be justified by considering the inequalities bounds (V. Vapnik, 1995):

$$R(w) \leq R_{emp}(w) + \sqrt{\frac{h\left(\ln\left(\frac{2l}{h}\right)+1\right)-\ln\left(\frac{\eta}{4}\right)}{l}} \quad (5.3)$$

The actual risk $R(w)$ is close to $R_{emp}(w)$ when l/h is large.

To avoid over-fitting, we require a new principle, based on the simultaneous minimization of the empirical risk and the VC-dimension of the set of functions. This is termed Structural Risk Minimization (SRM). To achieve SRM, one can design a structure for the function set, which makes up each subset and get the minimized experiential risk and select the appropriate subset to make the confidence interval minimal (X. Wang & Zhong, 2003). SVM is one such method aims at achieving SRM. Figure 5-1 shows the SVM separation of two classes to achieve SRM. In the graph, w denotes the weight vector; b is the bias and ξ_i is the slack variables that measures the misclassifications..

5.2 Support Vector Machine (SVM)

The Support Vector Machine was originally a linear classifier based on optimal hyperplane algorithm developed by Vapnik in 1963 (V. Vapnik & Lerner, 1963). In 1992, Bernhard Boser, Isabelle Guyon and Vapnik successfully apply kernel method to a maximum-margin hyperplane and build a non-linear classifier (B. E. Boser, I. M. Guyon, & V. N. Vapnik, 1992). In 1995, Cortes and Vapnik (Cortes & Vapnik, 1995) suggested a soft margin classifier which is a modified

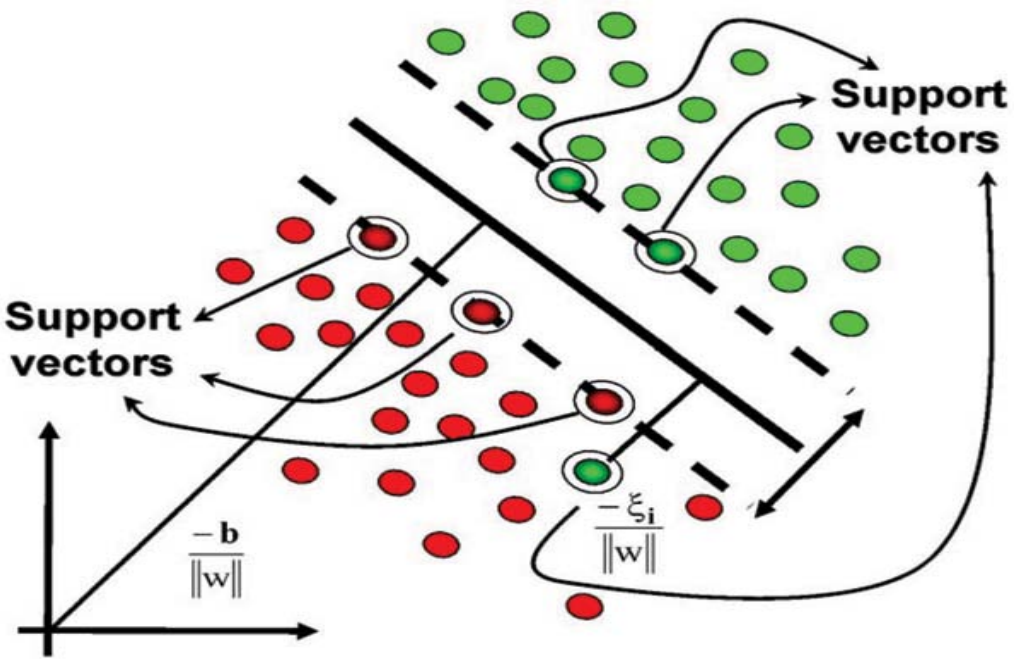


Figure 5-1 Support Vector Machine uses Structural Risk Minimization to compare various separation models and choose the model with the largest margin of separation (Winters-Hilt & Merat, 2007)

maximum margin classifier that allow for misclassified data. If there is no hyperplane that can separate the data into two classes, the soft margin classifier selects a hyperplane that separates the data as cleanly as possible with maximum margin.

SVM has been widely used in feature ranking and development of applications due to its robustness and remarkable performance in dealing with sparse as well as noisy data. One of the biggest issues in feature selection is the ability to deal with correlations between attributes as well as processing their non linear properties (Y. Liu & Zheng, 2006). The commonly adapted techniques in feature ranking such as multivariate analysis and multiple regression analysis do not handle nonlinear relationships. Support Vector Machines theory is widely used due to its elegance in solving such problems through application of kernel techniques for automatic mapping of feature space non linearity. One of the most commonly used kernel functions called the Gaussian

kernel is used for feature ranking in many applications due to its outstanding features (W. Wang, Xu, Lu, & Zhang, 2003).

In most real world applications, input samples cannot be exactly assigned to one class. In addition, sampling can affect the significance of a feature. It is therefore critical for some samples to be assigned to one class for effective separation using SVM. Noisy samples are not meaningful in and should therefore be discarded. There exist other problems like over-fitting. Subsequently, the fuzzy SVM (Chen & Wang, 2003) concept emerges, which uses a combination of fuzzy logic and SVM to allow different samples of to have different contributions to their own classes. Their research was however limited to binary SVM and did not consider multiclass SVM.

Support virtual machine theory was originally developed for binary classification and for it to be extended to multiclass SVM; various methods based on binary SVM were proposed. The various approaches for feature selection can be grouped into two broad categories: the wrapper approach and the filter approach (Trujillo-Ortiz & Hernandez-Walls., 2003).The filter approach is classifier independent while the wrapper approach is classifier dependent. The wrapper approach examines the effectiveness as well as the goodness of the selected feature and its subsets for development of a better performance. This is done based on the classification accuracy. Most experimental results have been found to favour the wrapper approach (Farrar & Glauber, 1967; Nakayama & Shimojo, 1992).

Support vector machines belong to a family of generalized linear classifiers. It is a classification and regression prediction tool that maximises the predictive accuracy using machine learning theory. Support vector machines can be defined as systems which use hypothesis space of a linear functions in a high dimensional feature space, trained with a learning algorithm from optimization theory that implements a learning bias derived from statistical learning theory (SLT) (Vikramaditya, 2006). SVM becomes famous when, using pixel maps as input; it gives accuracy comparable to sophisticated neural networks with elaborated features in a handwriting recognition

task (Ayat, Cheriet, Remaki, & Suen, 2001). SVM embodies the Structural Risk Minimization (SRM) principle, which has been shown to be superior (Burges, 1998) to traditional Empirical Risk Minimization (ERM) principle used by conventional neural networks. SRM minimizes an upper bound on the expected risk, where as ERM minimizes the error on the training data. It is this difference which equips SVM with a greater ability to generalize, which is the goal in statistical learning.

Supposing that we had a set of data classes given $(x_i, y_i), x_i \in \{1, -1\}, i = 1, \dots, l$, the linear soft margin SVM attempts to solve the constrained quadratic optimization problem:

$$\arg\min \frac{1}{2}\|w\|^2 + C\sum_i \xi_i \tag{5.4}$$

Subject to the loss function:

$$c_i(w^* x_i - b) \geq 1 - \xi_i \tag{5.5}$$

$$\xi_i > 0, i = 1, \dots, l \tag{5.6}$$

The resulting hyperplane has the following function:

$$f(x) = sign\left[\sum_{i=1}^{sv} \alpha_i y_i (x^* x_i) + b\right] \tag{5.7}$$

α_i is the Lagrange multiplier for each training data points, $\alpha_i > 0$ lies on the boundary of the hyperplane is called support vectors. Non-linear SVM first map data into high dimensional feature space via a kernel function and constructs the optimal separating plane using linear algorithm. The kernel can vary from polynomial, RBF, or sigmoid functions. The constructed hyperplane equation (5.8) that separates the data points corresponding to a decision rule:

$$\prod_{w,b} = wx + b = 0, x \in R^n \tag{5.8}$$

$$g(x) = sign(wx + b) \tag{5.9}$$

where *w* denotes the weight vector; *b* is the bias. The SVM choose the separating hyperplane $w.x+b=0$ which gives the largest margin. The central idea of SVM is that obtaining the hyperplane with the maximal margin minimise the risk of wrong decisions when classifying new data. The distance of this hyperplane to the closest data points should be maximised by solving the following equation:

$$\max_{(w,b)}(\min_i d(\Pi_{w,b}, x_i)) \tag{5.10}$$

where $d(\Pi_{w,b}, x_i) = \frac{|wx_i + b|}{\|w\|}$ is the distance between data point i and the plane $\prod_w$. The plane $\prod_w$ that solves the above equation is called the optimal separating hyperplane (V. N. Vapnik, 1998). Support vector machine (V. N. Vapnik, 1998) has the following features:

1. Avoid over-fitting through the use of Structural Risk Minimisation.
2. The formulation can be simplified to a convex quadratic programming problem; the training will converge to a global optimum which is the best solution for a given kernel and training data sets.

For the given data set, information can be condensed while training without losing useful information (Hua & Sun, 2001; Nakayama & Shimojo, 1992).

5.3 Prototype Ranking using Support Vector Machine Recursive Feature Elimination (SVM-RFE)

SVM-RFE (Support vector machine recursive feature elimination) (Guyon et al., 2002a) was initially developed to help in gene selection during cancer classification. It is a wrapper based approach. Various features that are selected by SVM-RFE provide a better classification performance as opposed to other methods such as rough sets (Guyon et al., 2002a). If we let

$z_i = \varphi(x_i)$ denote the respective feature space vector with a mapping function φ from R^N to a feature space Z, a hyperplane can be defined as:

$$w.z + b = 0 \tag{5.11}$$

The given set S is said to be linearly separable if and only if there exists (w,b) so that the following inequalities:

$$\begin{aligned} w.z_i + b \geq +1 \Rightarrow y_i = +1 \\ w.z_i + b \geq -1 \Rightarrow y_i = -1 \end{aligned} \tag{5.12}$$

hold true for all data samples of the given set S. To work on data that is not linearly separable, the previous analysis can be generalized through introduction of some non-negative variables $\xi_i \geq 0$ so that equation (5.12) is modified to

$$y_i(w^* z_i + b) \geq 1 - \xi_i \ , i = 1,...,l \tag{5.13}$$

The optimal hyperplane problem is then regarded as the solution to

$$\text{minimize } \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l} \mu_i \xi_i \tag{5.14}$$

$$\xi_i \geq 0 \ \ i = 1,...,l \tag{5.15}$$

The parameter C is a constant which can be treated as a regulation parameter. Manipulating this parameter can make a good balance between minimization of the error function and maximization of the optimal hyperplane margin. A smaller μ_i reduces the effect of the parameter ξ_i such that the impact of the respective point x_i is lesser. Equation (5.15) can be solved through introduction of a Lagrange multiplier α transforming the equation to:

$$\text{minimize } W(\alpha) = \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j (z_i \cdot z_j) - \sum_{i=1}^{l} \alpha_i \tag{5.16}$$

With the Kuhn-Tucker conditions defined as shown below:

$$\alpha_i(y_i(w\cdot z_i+b)-1+\xi_i)=0,\ i=1,...,l \tag{5.17}$$

$$(\mu_i C-\alpha_i)\xi_i=0,\ i=1,...,l \tag{5.18}$$

The given data sample denoted as x_i with the respective $\alpha_i>0$ is known as a support vector. Two types of support vectors can be defined. The support vector with corresponding $0<\alpha_i<\mu_i C$ lies on the hyperplane margin. When $\alpha_i=\mu_i C$, the point is said to be misclassified. One important distinction between SVM and fuzzy SVM is that the point having the same value of α_i may show a different type of support vector in fuzzy SVM because of the μ_i (Roberts, 1976). The mapping function, ϕ , is in most cases nonlinear and unknown. As opposed to calculating the function ϕ , the kernel function K is used to calculate the inner product of the two vectors within the feature space Z simplifying the mapping function as

$$K(x_i,x_j)=\phi(x_i)\cdot\phi(x_j)=z_i\cdot z_j \tag{5.19}$$

The most commonly used kernel functions are shown below (C.-F. Lin & Wang, 2002)

$$\text{linear kernel: } K(x_i,x_j)=x_i\cdot x_j \tag{5.20}$$

$$\text{polynomial kernel: } K(x_i,x_j)=(1+x_i\cdot x_j)^p \tag{5.21}$$

$$\text{Gaussian kernel: } K(x_i,x_j)=\exp(-\left\|x_i-x_j\right\|^2/2\sigma^2) \tag{5.22}$$

The order P of the polynomial kernel for equation (5.20) and the spread in the Gaussian function σ in equation (5.21) are kernel parameters that can be manipulated. The parameter w is the weight vector and the decision function which can be expressed using the Lagrange multiplier α_i as shown in equation (5.22) and (5.23) below.

$$w=\sum_{i=1}^{l}\alpha_i y_i z_i \tag{5.23}$$

$$D(x)=sign(w\cdot z+b)=sign(\sum_{i=1}^{l}\alpha_i y_i K(x_i,x)+b) \tag{5.24}$$

One-versus-one SVM (OVO-SVM) (A. Rakotomamonjy, 2003b) can be used for binary classification in multiclass problem. This method constructs $\binom{n}{2} = \frac{n(n-1)}{2}$ binary Support vector Machines for an n-class problem, where each $\frac{n(n-1)}{2}$ Support vector machine is trained on data samples from the two classes. Data samples are separated by a series of optimal hyperplanes which implies that the training data for the optimal hyperplane is maximally distanced from the other hyperplanes. In addition, the lowest classification error rate can be achieved when using the hyperplane for classification of the current training set. The hyper lanes can be modified from Equation (5.10) as shown below.

$$w_{st} \cdot z + b_{st} = 0 \tag{5.25}$$

The decision functions can be defined as $D_{st}(x_i) = w_{st} \cdot z + b_{st}$, where s and t denote two arbitrary classes partitioned by an optimal hyperplane within n classes; w_{st} represents the weight vector while b_{st} represents the bias term. Once all $\frac{n(n-1)}{2}$ classifiers have been constructed, a voting strategy called max-win can be used to test all data samples (Krebel, 1999). Every $\frac{n(n-1)}{2}$ one versus one support vector machine gives one vote. Supposing $D_{st}(x_i)$ gives x_i in the *s*-th class, the vote of x_i for the given *s-th* class is incremented by one or else, the *t*-th is incremented by one. x_i can therefore be predicted in the class with the highest vote. Because fuzzy SVM is a natural extension of SVM, OVO (One Versus One) scheme can be used to tackle multiclass problems with ease (Krebel, 1999).

SVM-RFE is an efficient and effective wrapper approach, can be used to conduct feature ranking in various design tasks (Duan et al., 2005). SVM-RFE is a backward elimination method which is executed sequentially based on the binary SVM. This method is most commonly used for selection of relevant set of features from complex and numerous features. The selection criterion

used in SVM-RFE was developed according to OBD (Optimal Brain Damage) and it has proved to be better than earlier methods such as rough sets and fuzzy sets (Alain Rakotomamonjy, 2003). The Optimal Brain Damage technique utilizes the change of cost function for feature selections and ranking. This can be defined as order two terms in the Taylor series of the give cost function as shown in the equation below:

$$c_f = \frac{1}{2}\frac{\partial^2 L}{\partial(\omega^f)^2}(D\omega^f)^2 \tag{5.26}$$

In the above equation, L is the cost function of any selected learning machine while ω is the weight of features. Optimal Brain Damage technique uses c_f for approximation of the change in the cost function which is caused by removing a given feature f through expansion of the cost function in the Taylor series. For binary SVMs, the OBD measure can be considered as the removed feature that has the smallest influence on the weight vector norm denoted as $\|w\|^2$ in Equation (5.15). The squared coefficients $w_j^2 (j = 1,..., p)$ of the weight vector *w* are employed as feature ranking criteria. The prototypes in this context represent the grey intensity of the pixel. Prototypes are selected using ranking criterion to rank variables. The ranking criteria w_j^2 for all features are computed, and the prototype with the smallest ranking criterion is discarded. This is an extension to bounds on L error, margin bound and other bounds of the generalization error. The criterion being investigated is C_t which is either weight vector $\|w\|^2$, the radius/margin bound $R^2\|w\|^2$ or the span estimate. It gives either an estimation of the generalization performance or an estimation of the dataset separability. It was initially proposed by Guyon *et al.* (2000) for selecting genes that is relevant for a cancer classification problem. The goal is to find a subset of size r among d features where $r < d$ which maximizes the performance of classifier (A. Rakotomamonjy, 2003a). The method is based on backward sequential selection. The features are

removed one at a time until r features remain. The criteria for selection are derived from Support Vector Machines (SVM) and are based on weight vector sensitivity with respect to a variable (Guyon, Weston, Barnhill, & Vapnik, 2002b). Variables are selected using ranking criterion to rank variables. This is an extension to bounds on L error, margin bound and other bounds of the generalization error (A. Rakotomamonjy, 2003a). The criteria being investigated is C_t which is either weight vector $\|w\|^2$, the radius/margin bound $R^2\|w\|^2$ or the span estimate. It gives either an estimation of the generalization performance or an estimation of the dataset separability. The removed variable is the one whose removal minimizes the variation of $\|w\|^2$.Feature ranking criterion can therefore be represented as shown in the equation below.

$$\begin{aligned} c_f &= \left| \|w\|^2 - \|w^{(-f)}\|^2 \right| \\ &= \frac{1}{2}\left| \sum_{i=1}^{l}\sum_{j=1}^{l} \alpha_i^* \alpha_j^* y_i y_j K(x_i, x_j) - \sum_{i=1}^{l}\sum_{j=1}^{l} \alpha_i^{*(-f)} \alpha_j^{*(-f)} y_i y_j K^{(-f)}(x_i, x_j) \right| \end{aligned} \tag{5.27}$$

In the above equation, α_i^* is the solution for equation (5.15) while $-f$ implies that the feature denoted as f has been removed. K is the kernel function which is calculated using x_i and x_j (Y. Mao, Zhou, Pi, Sun, & Wong, 2005). In order to calculate the change in an objective through removal of feature f, $\alpha_i^{*(-f)}$ must be equal to α_i^* (Alain Rakotomamonjy, 2003). The reduction of computational complexity requires re-computation of the kernel function $K^{(-f)}(x_i, x_j)$. SVM-RFE iterates through all the features. At every step of elimination, the weights of various features are obtained through comparison of the training samples with the existing features. The feature with the minimum c_f is eliminated. This procedure is repeated until all features are ranked according to the order of removal.

This approach of criteria ranking derived from SVM is different from the popularly used soft margin SVM. In traditional SVM where given a set of labelled instances $\mathbf{X}_{train} = \{\mathbf{x}_i, y_i\}_{i=1}^{n}$ and a kernel function k, SVM finds the optimal α_t for each x_i to maximize the margin γ between the hyperplane and the closest instances to it. The prediction for a new sample x is made through:

$$sign\left(f(\mathbf{x}) = \sum_{i=1}^{n} y_i \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b \right) \tag{5.28}$$

where *b* is the threshold. One norm soft-margin SVM minimize the primal Lagranian:

$$L_p = \frac{\|\mathbf{w}\|^2}{2} + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i \left[y_i (\mathbf{w} \cdot \mathbf{x}_i + b) - 1 + \xi_i \right] - \sum_{i=1}^{n} r_i \xi_i \tag{5.29}$$

where $\alpha_i \geq 0$ and $r_i \geq 0$. The penalty constant C represents the trade-off between the empirical error ξ and the margin. Two approaches (A. Rakotomamonjy, 2003a) are proposed for each criterion:

Zero-order method: The criterion C_t is directly used for variable ranking, and it identifies the variable that produces the smallest value of C_t when removed. The ranking criterion becomes $R_c(i) = C_t^{(i)}$ with $C_t^{(i)}$ being the criterion value when variable *i* has been removed.

First-order method: This uses the derivatives of the criterion C_t with regards to a variable. This approach differs from the zero-order method because a variable is ranked according to its influence on the criterion which is measured with the absolute value of the derivative.

The zero-order criteria based on bounds have been used for feature selection associated with different search space algorithm whereas the first-order ones are rather new for the purpose of feature selection. In the zero-order method, one suppresses the feature whose removal minimizes the criterion whereas in the first order methods, one removes the variable to which the criterion is less sensitive. For instance, in the zero-order $\|\mathbf{w}\|^2$ case, the ranking term is:

$$R_c(i) = \left\| \mathbf{w}^{(i)} \right\|^2 = \sum_{k,j} \alpha_k^{*(i)} \alpha_j^{*(i)} y_k y_j K^i \left(\mathbf{x}_k, \mathbf{x}_j \right) \tag{5.30}$$

where $K^{(i)}$ is the Gram matrix of the training data when variable *i* is removed.

In the first order case, the sensitivity of a given criterion with respect to a variable is measured. A possible approach is to introduce a virtual scaling factor and to compute the gradient of a criterion with respect to that scaling factor ρ. The latter acts as a componentwise multiplicative term whose value is 1 on the input variables and thus $k(\mathbf{x}, \mathbf{x}')$ becomes:

$$k(\rho \cdot \mathbf{x}, \rho \cdot \mathbf{x}'), \tag{5.31}$$

where "·" denotes the componentwise vector product. Consequently, one obtains the following derivatives for a Gaussian Kernel $k(\rho \cdot \mathbf{x}, \rho \cdot \mathbf{x}') = \exp\left(-\frac{\left\| \rho \cdot \mathbf{x} - \rho \cdot \mathbf{x}' \right\|}{2\sigma^2} \right)$:

$$\begin{aligned} \frac{\partial k}{\partial \rho_i} &= -\frac{1}{\sigma^2} \left(\rho_i x_i - \rho_i x_i' \right)^2 k(\mathbf{x}, \mathbf{x}') \\ &= -\frac{1}{\sigma^2} \left(x_i - x_i' \right)^2 k(\mathbf{x}, \mathbf{x}') \end{aligned}, \tag{5.32}$$

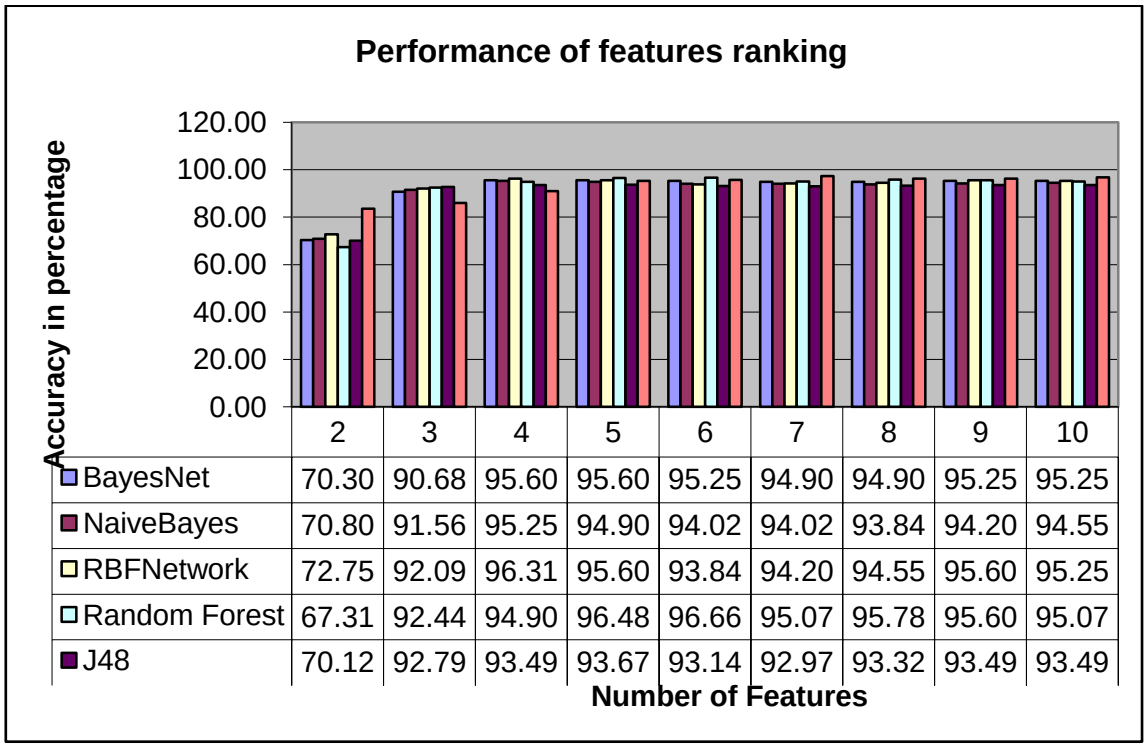
where we used $\rho_i = 1$. Then one needs to evaluate the gradient of the bounds with regards to a variable ρ_i and for a given criterion *C*, the ranking term would be:

$$R_c(i) = \left| \frac{\partial C(\alpha, b)}{\partial \rho_i} \right| = \left| \nabla \left\| \mathbf{w}^{(i)} \right\|^2 \right|. \tag{5.33}$$

The problem of searching the best *r* variables is solved by means of greedy algorithm based on backward selection. A backward sequential selection is used because of its lower computational complexity compared to randomized or exponential algorithms and its optimality in the subset selection problem. The algorithm starts with all features and repeatedly removes a feature until *r* features are left for all variables have been ranked.

Figure 5-2 shows the simulation results of features ranking using UCI heart data (Kurgan, Cios, Tadeusiewicz, Ogiela, & Goodenday, 2001). The data set uses SVM-RFE approach selects

top ten features and uses different classifiers for testing. The chart shows that only as few as three features are sufficient to obtain generalization of above 80%. Using different classifiers like BayesNet, NaiveBayes, RBF, Random Forest, J48 and our approach Support Vector Emergent Self-Organizing Map (SVESOM) of self-organizing classification (more details on SVESOM can be found in section 7.2 case studies), the results are comparable. The use of four features onwards does not significantly improve the performance further. It is very efficient to use this approach because the classifier only needs to process a few top ranked variables.



	2	3	4	5	6	7	8	9	10
BayesNet	70.30	90.68	95.60	95.60	95.25	94.90	94.90	95.25	95.25
NaiveBayes	70.80	91.56	95.25	94.90	94.02	94.02	93.84	94.20	94.55
RBFNetwork	72.75	92.09	96.31	95.60	93.84	94.20	94.55	95.60	95.25
Random Forest	67.31	92.44	94.90	96.48	96.66	95.07	95.78	95.60	95.07
J48	70.12	92.79	93.49	93.67	93.14	92.97	93.32	93.49	93.49

Figure 5-2 Performance of feature ranking

Algorithm 5.1: SVM based Prototype Ranking

Given Rank=[] ; Var=[1,...,N]

Repeat the following until Var is not empty:

1. Train a SVM classifier with all the training data and variable Var.
2. For all Var, evaluate ranking criterion $R_c(i)$ of variable *i*
3. $\text{Best} = \arg\min R_c$
4. Rank the variable that minimizes R_c :
   Rank=[bestRank]
5. Remove the variable that minimizes R_c from the selected variables set
   Var=[1, …, best-1, best+1, …N]

5.3.1 Multi-class SVM-RFE

SVM-RFE can be extended to multiclass SVM-RFE and each feature ranking criterion can be denoted as c_{st} for two arbitrary classes s and t, with $\frac{n(n-1)}{2}$ classifiers. There are two basic approaches for implementing multi-class SVM: consider all data in a single optimization function (S. H. Han & Kim, 2003) or decompose the multiple classes into a series of binary SVM. The second approach of implementing SVM includes One-Versus-All SVM (OVASVM) (V. N. Vapnik, 1998) and One-Versus-One SVM (OVOSVM) (Guyon et al., 2008). For a n-class problem, n binary SVM are constructed. The SVM is trained using class samples against the rest of the samples. Given a sample to classify, SVM are evaluated with the result being the label of the most significant decision function value. OVOSVM works by training binary SVM between pair-wise classes. OVO model consists of $\frac{n(n-1)}{2}$ binary SVM for n-class problem. Each of the $\frac{n(n-1)}{2}$ casts one vote for its favoured class, the class with the maximum votes wins (Guyon et al., 2008). As shown in Figure 5-3 is an example of the OVOSVM with four-class problem, with $n=4$ establishes 6 binary SVM. OVOSVMs can also be used to calculate feature ranking in multiclass fuzzy SVM models. Research has shown that multiclass feature selection is not widely

used in bioinformatics field due to the computational overheads of data calculation, especially in the field of gene selection which involves thousands of gene data. Apart from gene selection problems, the number of features of other problem domains like medical datasets is relatively smaller and it can still be computed efficiently. Binary SVM-RFE for gene selection applications usually uses the linear kernel function for shorter training times. Nonlinear kernel function is preferable due to its ability to tackle nonlinear relationships between products form features (Sung H. Han, Kim, Yun, Hong, & Kim, 2004) . The computational cost for accelerating the SVM-FRE process can sometimes lead to degradation in the ranking accuracy if not carefully executed.

Multiclass SVM-RFE process can be done on the basis of optimal one versus one multiclass fuzzy SVM model. The parameters obtained from cross-validation can be used to enable critical features to be selected according to n-class labels. The relative significance of various features can be identified by analyzing the distribution of weights in every iterative step of the multiclass SVM-RFE process. Every OVO SVM model can be used to separate the two arbitrary classes' i.e. s and t from each other. The ranking criterion denoted as c_{st} of $\frac{n(n-1)}{2}$ OVO SVMs can then be computed using equation 16 (Y. W. Chen & C. J. Lin, 2006).

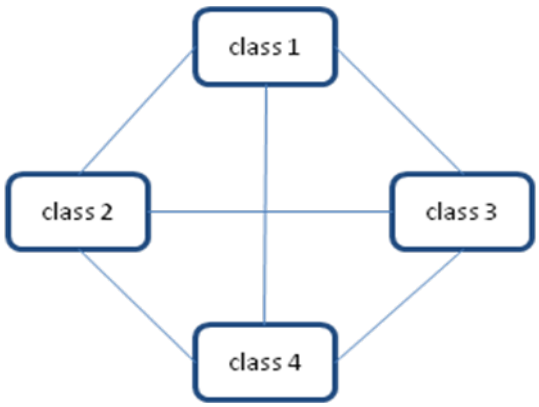


Figure 5-3 The OVO SVM for a four-class problem, six binary SVMs are required to perform the task

Two strategies can be used for selecting features: the overall ranking as well as class-specific ranking. The first strategy involves summation of c_{st} of $\frac{n(n-1)}{2}$ One Versus One SVMs to get the overall ranking and examine the common features of the n-class labels. This method is commonly used in SVM-RFE applications like selecting important genes by considering several kinds of cancer at the same time. The class specific feature ranking strategy can be used to allocate a class specific ranking to sum c_{st} of $n-1$ One Versus One Support vector Machines for a specific class denoted as S. The multi-class SVM-RFE procedure can be summarised as follows (Shieh & Yang, 2008):

Algorithm 5.2: Multi-class SVM-RFE

1. Begin with an empty ranked features list say $R=[]$ and the selected product feature list denoted as $F=[1,...,d]$;
2. Repeat the following until all features are ranked:
   - Train $n(n-1)/2$ or $n-1$ fuzzy SVMs using all the training samples and all features in F ;.
   - Calculate and sum the ranking criterion denoted as c_{st} of $n(n-1)/2$ or $n-1$ Support vector machines for features in F .
   - Get the feature with the smallest ranking criterion denoted as $e=\arg\min_f c_f$;
   - Add the identified feature e into the ranked feature list denoted as R: $R=[e,R]$;
   - Eliminate the feature e from the selected feature list F : $F=F-[e]$;
3. Display the ranked feature list R.

5.4 Conclusion

This chapter illustrates the concept of Prototype Ranking based on Support Vector Machine to solve the problem of imbalanced dataset. Prototype ranking results in the best feature(s) which can greatly enhance the learning and significantly reduce the computation time. This chapter walks through some background of the statistical learning theory and support vector machine. The algorithm of prototype ranking is then presented.

We will discuss the problem of imbalanced dataset which presents an important challenge to the machine learning community (Guo & Viktor, 2004) in next chapter. As traditional machine learning algorithm may be biased towards the majority class, thus producing poor predictive accuracy over the minority class. The purpose of this chapter is to analyse the imbalanced datasets and demonstrate how the model works well even under skewed condition. In this chapter, we first introduce the problem domain and problem formulation. Section 3 performs the analysis to discover the problems underlying imbalanced data. Lastly, the results are discussed.

Chapter 6 Analysis on Imbalanced Data

6.1 Imbalanced Dataset (IDS)

Typical dataset is constructed with equal or close to equal number of instances from each class. Most classifiers perform well in balanced dataset but not imbalanced dataset. According to Rehan (Akbani, Kwek, & Japkowicz, 2004), classifiers generally do not perform well on imbalanced datasets because they are designed to generalize from sample data and output the simplest hypothesis that best fits the data, based on the principle of Occam's razor. Imbalanced data set (IDS) is a phenomenon occurs where the number of instances in one class significantly outnumbers the instances from other classes. That is the training data is dominated by the instances belonging to one class. This type of datasets is observed in worlds of business, industry, scientific research and many real-world applications like vision recognition (Maloof, 2003), bioinformatics (Guo-ping, Li-xiu, & Jie, 2005), credit card fraud detection (Chan & Stolfo, 1998), detection of oil spills (Kubat, Holte, & Matwin, 1998), medical data (Blake & Merz, 1998) and

risk management data (Technology) where certain classes of the diagnosis are not as easily available as the other classes.

The class imbalance problem is one of the (relatively) new problems that emerged when machine learning matured from an embryonic science to an applied technology. Although practitioners might already have known about this problem early, it made its appearance in the machine learning and data mining research circles about a decade ago. Its importance grew as more and more researchers realized that their data sets were imbalanced and that this imbalance caused suboptimal classification performance (N. V. Chawla, Japkowicz, & A. Kolcz, 2003). This increase in interest gave rise to two workshops held in year 2000 and 2003 at the AAAI (Japkowicz, 2000) and ICML (N. V. Chawla et al., 2003) conferences, respectively. A follow up workshop, PAKDD, of the previous two workshops will be conducted on 2009 (N. Chawla, Japkowicz, & Zhou, 2009). Despite the fact that the workshops have already been held to discuss about the topic, a large number of practitioners plagued by the problem are still working in isolation.

In AAAI workshop, several issues were highlighted (N. Chawla et al., 2009; N. V. Chawla et al., 2003). Firstly, they found that large number of applications suffer from class imbalance problem. Secondly, smart sampling is commonly used to solve the problem but it is sometimes not useful (N. V. Chawla et al., 2003). Thirdly, the use of common evaluation measures often draws misleading conclusions. More accurate measures are desired like ROC curves and Cost Curves (Drummond & Holte, 2000; Provost & Fawcett, 2001). Fourthly, it was shown that concept-learning methods can use one-sided approach focusing on either the majority or the minority class. Next, it was discussed that there is close connection between the class imbalance problem and cost-sensitive learning. Cost-sensitive learning was reported to outperform random re-sampling (Domingos., 1999; Forman., 2003). Lastly, it was agreed that creating a

classifier that performs well across a range of costs is important. The evaluation measures and the relationship between class imbalances and cost-sensitive learning were vastly discussed.

The second workshop (N. V. Chawla et al., 2003) was a follow up to the first workshop (Japkowicz, 2000), it focused on different types of sampling approach for imbalance data. They discussed the works on random-sampling, uncertainty-sampling (selective-sampling) (Juszczak & Duin, 2003), under-sampling (down-sampling) the majority class (Drummond & Holte, 2000), over-sampling (up-sampling) minority class, progressive sampling and etc. On top of sampling, methods like probabilistic estimates (Nitesh Chawla, 2003), pruning, threshold adjusting, and cost-matrix adjusting are used to solve imbalanced problems as well. Another direction for imbalanced problem is to balance out the class distribution in data set, and extend the scenario to be one-class learning(Raskutti & Kowalczyk, 2003). In this workshop, Charles Elkan (Elkan., 2003) pointed out that ROC curves are unable to deal with within-class imbalances and different within-class misclassification costs. This workshop centers on different sampling approaches and C4.5 classifier to overcome the imbalance problem. It was argued that (N. V. Chawla et al., 2003), C4.5 is not the best classifier to deal with imbalance issue.

Imbalanced dataset exhibit skewed class distribution in which almost all instances belong to one or larger classes and far fewer instances belong to a smaller class. There are two types of imbalance, intrinsic and non-intrinsic imbalance. Intrinsic imbalance problems occur in situation where it happens by nature. Take fraudulent data for example, there are usually very few cases of fraud as compared to the large number of honest cases because it is the guideline or by nature to be honest rather than fraud. On the other hand, non-intrinsic imbalance happens when there is a constraint during data collection process that causes significant differences in the classes of samples collected. There are two inner factors affecting imbalance data, namely imbalance ratio and lack of information that makes the problem challenging. Imbalance ratio is obtained by dividing the number of positive samples by the number of negative samples. Lack of information

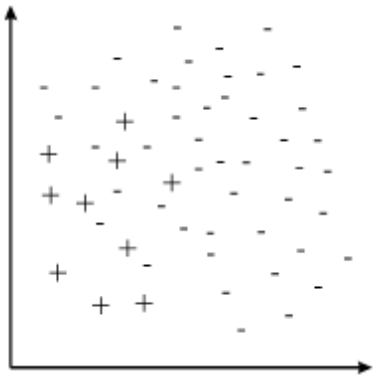


Figure 6-1 Sample plot of imbalanced data

occurs when there are too few samples available for training and testing. However, the lack of information does not necessarily qualify a problem to be imbalanced. A simple illustration can be observed from Figure 6-1 which shows the imbalanced data distribution with IR=0.1. The majority instances are represented by (-), the majority instances are represented by (+). We can see that there are some degrees of class overlapping and the class boundary may not be easily defined.

Imbalanced data is not commonly dealt with specifically in conventional classifiers; they do not make special allowance concerning the class imbalance. The IDS usually results in poor performance of standard classification algorithms (Chan & Stolfo, 1998; Guo-ping et al., 2005; Kubat et al., 1998; Maloof, 2003) like decision tree, nearest neighbor and naive-bayesian. When the data is imbalanced, it causes problem in proper feature selection and clustering. Traditional machine learning algorithms can be biased towards majority class due to over-prevalence. In the case of decision tree, cases from majority class tend to dominate the tree structure. In k-nearest neighbor, the nearest neighbors are mostly from the majority class. The same goes to naïve-bayesian because majority class is much more probable than minority class. The minority class is usually the class of interest and the errors coming from this class is more important and thus higher penalty errors in cost-sensitive learning. If a classifier can make correct prediction on the

minority class efficiently, it will be useful in many real life applications like the credit risk in banking industry.

Weiss (Weiss, 2004) pointed out six major problems that arise when solving imbalanced classes:

- *Improper evaluation metrics*: Metrics are used to guide the data mining algorithm and to evaluate the results of data mining. The metric ought to value rarity of the problem domain. Accuracy is insufficient as it places more weight on the majority class. Metrics like ROC analysis, AUC(Area Under Curve), Precision, Recall, Geometric Mean and F-Measure should be considered.
- *Lack of data*: The lack of data is the most fundamental problem underlying IDS as it makes the detection of patterns in minority difficult. The rare case causes small disjuncts (Bosch, Weijters, Herik, & Daelemans, 1997; K. Ali & Pazzani, 1995) in the learned classifier which have a higher error rate than large disjuncts.
- *Relative rarity*: Some objects are not rare in absolute sense but are rare relative to other objects. It is difficult to identify the rare object as it is not easily located using greedy search heuristics and global methods.
- *Data fragmentation*: This problem occurs when the original problem is decomposed into smaller pieces and regularities can only be found within each individual partition which contains less data.
- *Inappropriate inductive bias*: Extra bias is usually introduced to generalize from specific examples. The bias is critical to its performance; it can adversely impact the ability to learn rare class.
- *Noise*: Weiss (Weiss, 2004) argued that noise has greater impact on rare cases than on common cases. Because rare cases have fewer examples to begin with, it will take fewer noisy examples to impact the learned sub-concept.

6.2 Problem Formulation

For a binary-class problem, we often assume we have collected datums of n samples given as:

$$D := ((x_1, y_1), ...(x_n, y_n)), x_i \in \Re^m, y_i \in \{-1, +1\} \quad (6.1)$$

The datums D are obtained independent identically distributed (i.i.d.) with unknown probability function (pdf) $P(x, y)$ of underlying dependency. The datums is presumed to have distribution of $P(Y \mid X)$.

The supervised learning attempts to train a classifier that maps the input to output through some objective functions or by adjusting neurons weight $F = X \to Y$. The goal is to maximise the likelihood of accurately predicting x_i.

For the case of imbalanced dataset $D = \left\{ D^+ \cup D^- \right\}$, the positive data D^+ is associated with $y = 1$ and negative data D^- is associated with $y = -1$. In an imbalanced dataset, the positive class instances are significantly lesser than instances from negative class:

$$\left| D^+ \right| > \left| D^- \right| \quad (6.2)$$

The probability estimation of the positive class is much smaller, i.e.

$$P(c = D^+ | x) < 0.5 . \quad (6.3)$$

The imbalance ratio is defined as the follows,

$$\text{IR} = \frac{\text{Positive Instance}}{\text{Negative Instance}} \quad (6.4)$$

To a specific dataset $\mathrm{D}(\mathrm{x_i}, y_i)$, with the distribution $\mathrm{p(Y|X)}$, for which each $\mathrm{x_i} \in X$ the features vector and $y_i \in Y$ is the associated class label. The objective of to solve the binary problem is to train a classifier $f : X \mapsto Y$ to estimate the probability for each x_i in the testing sample which belong to Y. The idea is to enhance the performance by minimizing loss or maximize accuracy.

Thus, the model should maximize some objective function, O, such that the $E_{(X,Y)}\left[O(f(X),Y)\right]$ is maximized. In the presence of high imbalance, this is practically difficult to achieve due to the limitation of the number of instances from minority, so we can only approximate the true function. The most straightforward way to balance the imbalance data is resample the data. Re-sampling attempts to re-draw the training data distribution to be D^*. By tuning the $S: D \mapsto D^*$ can improve the classifier's performance. Cieslak et. al. (Cieslak & Chawla, 2008) provides a study on improving the performance on this issue, they demonstrate examples of binary problem with IR=0.05 as shown in Figure 6-2. The pairs (ρ, δ) specify the level of density ρ and centroid separation δ for each. As the row descend, ρ decreases; δ decreases from the left to right columns. Each class is centered at a fixed point $\bar{x}_+$ and $\bar{x}_-$ with radius r_+ and r_-. Since the number of points for each class is fixed, the densities ρ_+ and ρ_- are product of radii, reducing radius length increases density. The higher the δ, the lower the class overlap. The experiment (Cieslak & Chawla, 2008) considers high, medium, and low for the pairs and observe that $\bar{x}_-$ and ρ_- remain fixed through the constructions.

6.3 Data Analysis

Prior to solving imbalance data problem, we ought to have an understanding about the data. This section examines the data and investigates the complexity underneath the data. We first introduce the dataset to be analyzed. We then explore the intrinsic properties underlying the imbalanced datasets with Radviz, Samples to Feature Ratio (SFR), Fisher's Discriminant Ratio (FDR), Pearson's Skewness Coefficient (PSC) and Kurtosis Skewness Risk (KSR).

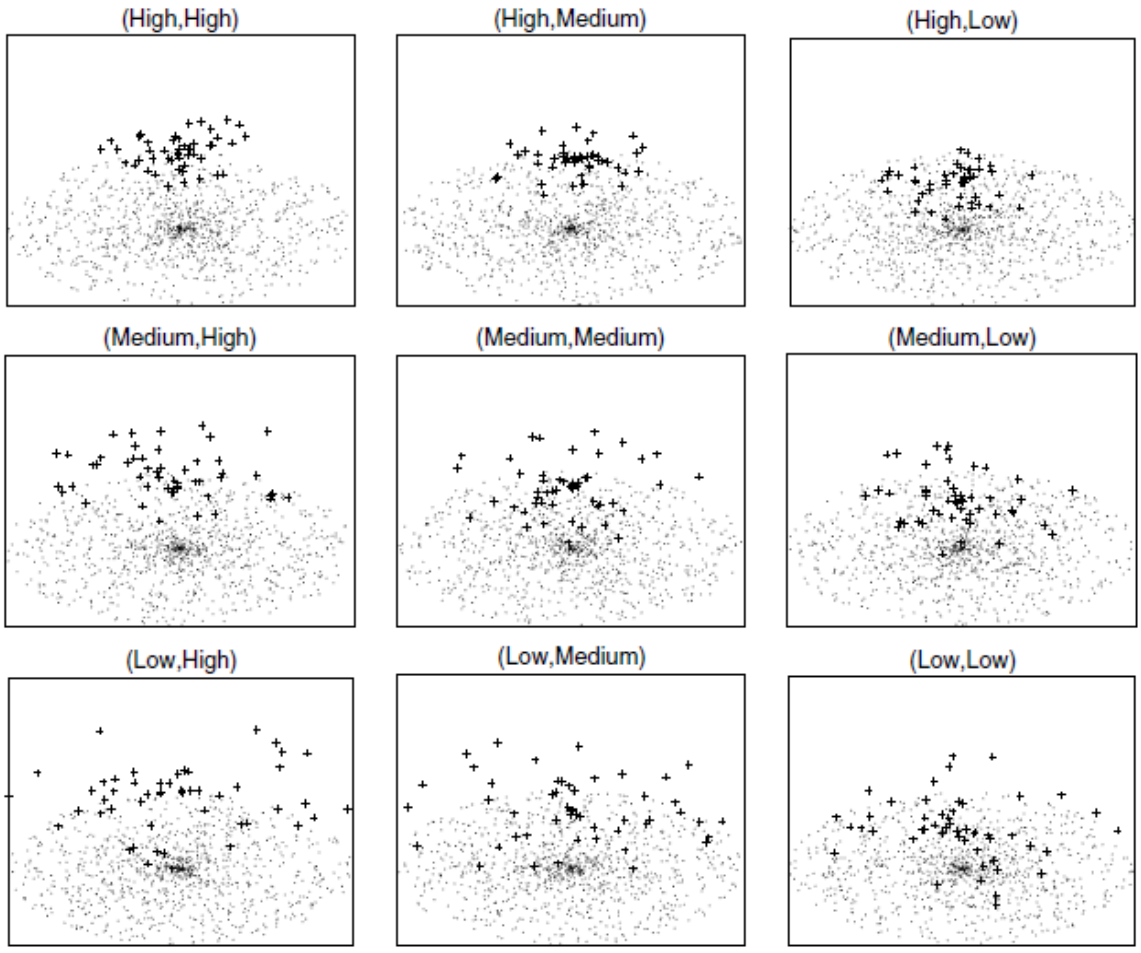


Figure 6-2 Demonstration of two-class radial distribution, with different level of density and centroid separation (Cieslak & Chawla, 2008)

6.3.1 Dataset

The data being inspected are medical datasets which were extracted from the UCI Machine Learning Repository(Blake & Merz, 1998). The IDS in used, namely Pima Indians Diabetes (PIMA) data and Wisconsin Breast Cancer Diagnostic (WBCD) data. The PIMA dataset contains the data from all female patients of at least 21 years old, and of Pima Indian heritage. The database consists of 768 instances, each with 8 attributes. A total of 268 patients were diagnosed as having diabetes and 500 patients are healthy person without diabetes. The WBCD data is to diagnose benign and malignant cancer. The features are computed from digitized image of a fine needle

aspirate (FNA) of a breast mass which described the characteristics of the cell nuclei. The real-valued features are computed for each cell nucleus like radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, fractal dimension and etc. WBCD data consist of 569 instances with 32 binary attributes. The heart dataset describes the diagnosis of cardiac Single Photon Emission Computed Tomography (SPECT) images. Each of the patients is classified into two categories: normal and abnormal. The database of 267 SPECT image sets was processed to extract features that summarize the original SPECT images. As a result, 44 continuous feature patterns were created for each patient. The pattern was further processed to obtain 22 binary feature patterns which was generated using CLIP3 algorithm. SPECT data contains 267 samples, which are represented by attributes summarizing the original SPECT images. The SPECT data has 22 binary attributes.

Table 6-1 summarizes the imbalanced datasets that we used in our experiments. The number of features, total instances, minimum class percentage and the imbalance ratio (IR) are listed. All are arranged into binary decision problem to predict instances as either positive or negative. The imbalance ratio (IR) is obtained by dividing the number of positive samples over the number of negative samples. A dataset is termed balance if the imbalance ratio is one. The IR of the datasets is imbalance with the ratio of ranging from 0.23 to 0.54. In most medical data, positive samples correspond to having infected the disease, whereas negative samples means the samples do not response to the test, hence not infected. Each of the dataset is divided into three stratified cross-validation sets for training and testing. Training set is made up of 2/3 of the overall data and testing set is made up of the remaining 1/3.

Table 6-1 Different datasets with imbalance ratio (IR)

Datasets	Features	Instances	Class(positive, negative)	Min %	IR
SPECT	22	267	(Abnormal, normal)	13.21	0.25
WBCD	30	569	(Malignant, benign)	20.6	0.23
PIMA	8	768	(Diabetic, healthy)	34.9	0.54

6.3.2 Radviz

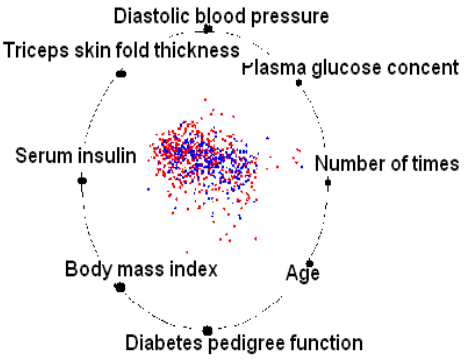
We first look at the distributions of data points under radial visualization (Radviz). Radviz (Hoffman et al., 1997) is a method where the examples are represented by points inside a circle. The visualized attributes correspond to points equidistantly distributed along the circumference of the circle. The most influential attributes wholly decides the position of a point. The data point is placed at the position where the sum of all forces from each attributes equals to zero. Previously we discussed that sampling is thus far the most popular approach in solving imbalanced data problem. However, as shown in Figure 6-3 are the visualizations of imbalanced and balanced datasets for PIMA, SPECT and WBCD medical data. The datasets are originally imbalanced with much larger numbers of negative instances. We randomly down sampled the positive instances to achieve a balance ratio between the two. Figure 6-3 suggests that the down sampling does not make the data separable under the presence of full features. The 536 data points in balanced PIMA appear to be centrally clustered together for both the classes. The SPECT dataset consists of 267 data points initially as shown in Figure 6-3(c) which is dominated by class 1. After applying the sampling, the data distribution remains the same with a balanced IR. The same goes with the WBCD data. Radviz showed that the intrinsic properties of imbalanced datasets remain the same despite randomly down-sample it to have a balanced number of samples as the data distributions are inseparable.

6.3.3 CHI-Plot

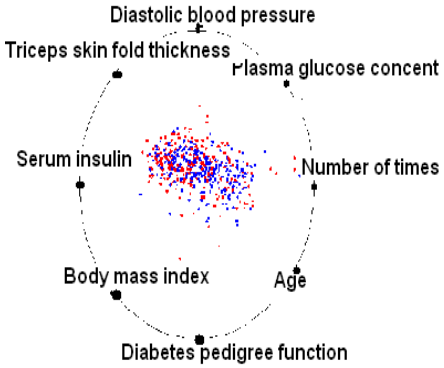
This section makes use of Chi-square quantile-quantile (Q-Q) plot to compare the effect of imbalanced data graphically. The results obtained are generated using the work from Trujillo-Ortiz (Trujillo-Ortiz & Hernandez-Walls., 2003). The Q-Q plot displays the squared Mahalanobis distances of the observations from the mean vector. The Q-Q plot for the normal data should lie closely along the red line which is $x = y$. The plots from Figure 6-4 show the multivariate Q-Q plots of the medical data. The Mahalanobis distance (y-axis) for each datum is plotted against the Chi-square (x-axis). All the three datasets have similar patterns with increasing gradients towards the right and are not normally distributed; the data distribution is skewed to the right. The balanced data are randomly down sampled to have an IR of 1.0. We can observe from the plots that sampling does not reduce the skewness of the medical data.

6.3.4 Samples to Features Ratio (SFR)

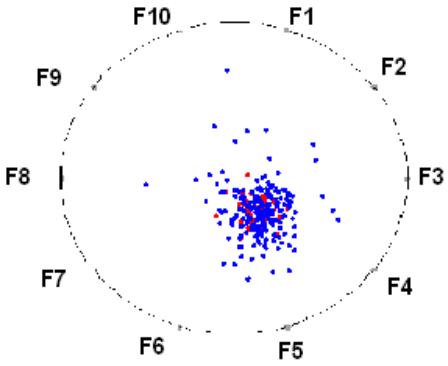
This number shows the distribution of the samples over the feature space it is in. SFR is defined as dividing the total number of features by the total number of training samples. This number should be greater than 0, provided that training data size and the feature dimensionality are both greater than zero. The lower the ratio denotes more data points available which is desirable. SPECT consists of 267 instances with 22 features which gives SFR of 8.2%. PIMA has 768 samples with 8 features that make its SFR very appealing at 1.0%. WBCD obtains SFR of 5.6% with 569 instances and 32 features. Breast tissue data (Farrar & Glauber) is made up of 106 instances with 9 features that make its SFR 8.5%.



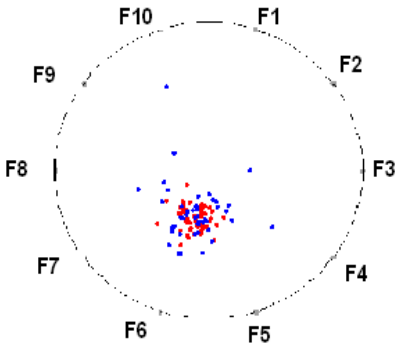
(a)Imbalanced PIMA



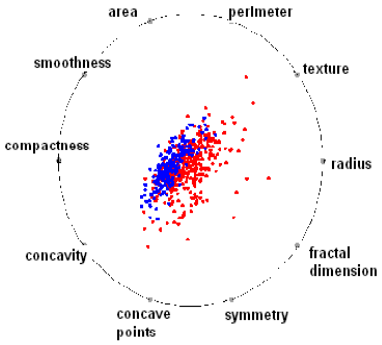
(b) Balanced PIMA



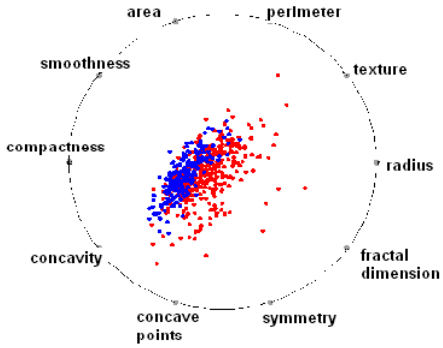
(c) Imbalanced SPECT



(d) Balanced SPECT

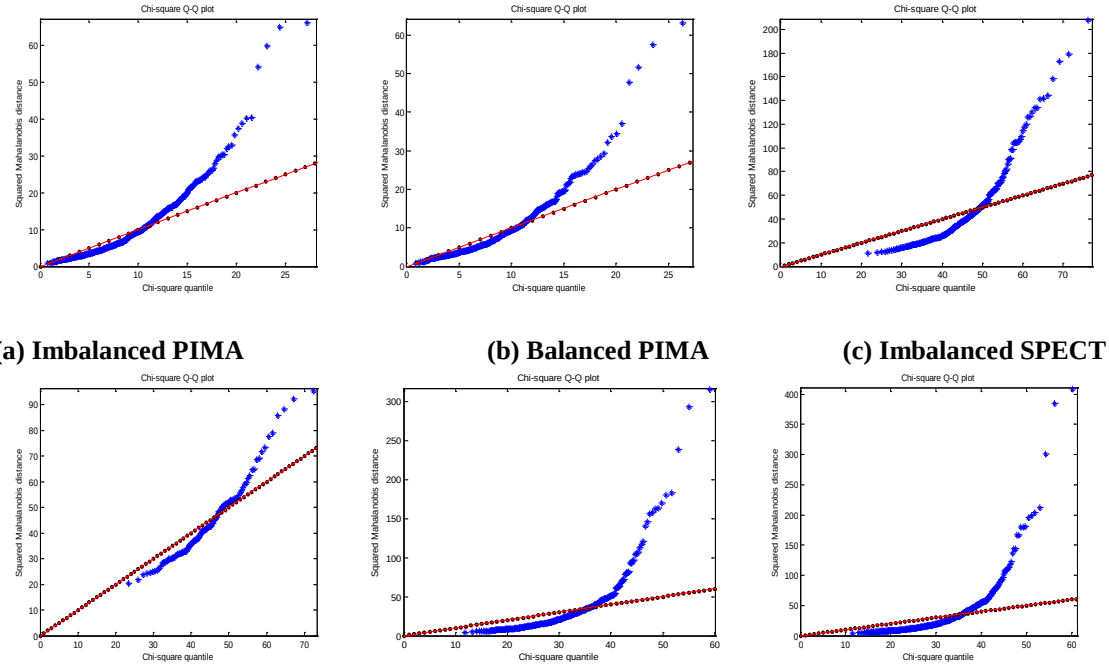


(e) Imbalanced WBCD



(f) Balanced WBCD

Figure 6-3 Multi-dimensional visualization of the distribution of balanced and imbalanced datasets



(a) Imbalanced PIMA **(b) Balanced PIMA** **(c) Imbalanced SPECT**

(d) Balanced SPECT **(e) Imbalanced WBCD** **(f) Balanced WBCD**

Figure 6-4 CHI-plots of balanced and imbalanced datasets

6.3.5 Pearson's Skewness Coefficient (PSC)

The skewness characterizes the degree of asymmetry of a distribution around its mean. The mean, standard deviation, and average deviation are dimensional quantities which have the same units as the measured quantities x_j, the skewness is conventionally defined in such a way as to make it non-dimensional. It is pure number that characterizes only the shape of the distribution. The usual definition is:

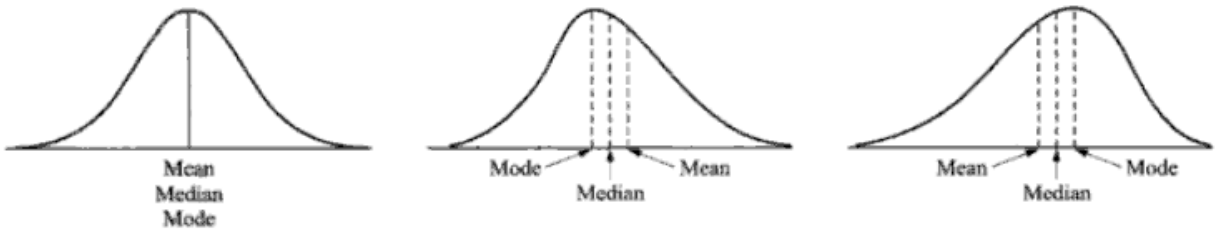
$$\text{Skew}(x_1 \ldots x_N) = \frac{1}{\text{N}} \sum_{j=1}^{N} \left[\frac{x_j - \overline{x}}{\sigma} \right]^3 \tag{6.5}$$

where $\sigma = \sigma(x_1 \ldots x_N)$ is the distribution's standard deviation. Positive value of skewness signifies a distribution with an asymmetric tail towards the positive side. Negatively skewed is when the left tail is longer. As shown in Figure 6-5 is the three categories of skewness distribution for two class

problem. Alternatively, we can measure the skewness by Pearson 's second skewness coefficient which is defined by:

$$PSC = \frac{3(\mu - med)}{\sigma} \tag{6.6}$$

The Figure 6-6 displays the degree of skewness of breast tissue and Pima Indian dataset. We can see that breast tissue dataset with imbalanced ratio of 0.18 is positively skewed for most of the variables. Whereas for PIMA dataset with imbalanced ratio of 0.54 the situation gets better as some variables are not skewed, a few of it is positively skewed.



(a) Symmetrical **(b) Positively skewed** **(c) Negatively skewed**

Figure 6-5 Skewness distribution of datasets

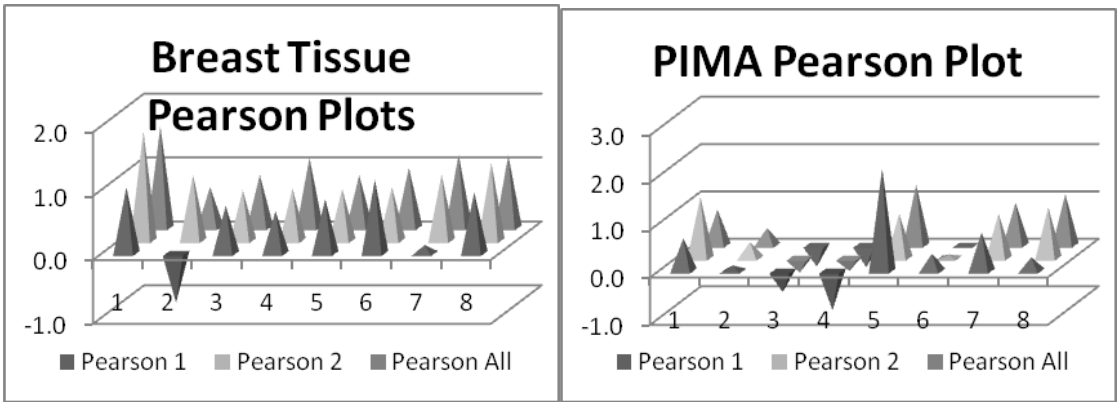


Figure 6-6 Pearson's second skewness coefficient plot

6.3.6 Kurtosis Skewness Risk (KSR)

The kurtosis (Girolami & Fyfe, 1997) is also a non-dimensional quantity. It measures the relative peakness or flatness of a distribution relative to normal distribution. A distribution with positive kurtosis is termed leptokurtic. A distribution with negative kurtosis is termed platykurtic. Higher kurtosis means more of the variance is due to infrequent extreme deviation, as opposed to frequent modestly-sized deviations. If $x_1 \ldots x_n$ are independent random variables all having the same variance, then

$$Kurt\left(\sum_{i=1}^{n} X_i\right) = \frac{1}{n^2}\sum_{i=1}^{n} Kurt(X_i), \tag{6.7}$$

As shown in Figure 6-7 is the Kurtosis value of breast tissue and pima data. Both datasets appear to have similar properties with positive Kurt which indicates a relatively peaked distribution than normal distribution.

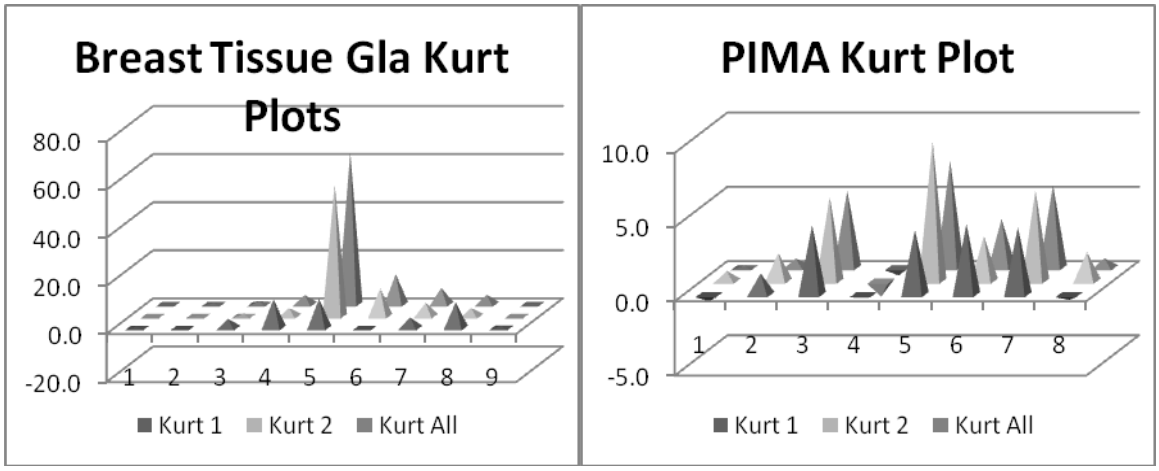


Figure 6-7 Kurtosis plot of breast tissue and pima dataset

6.4 Performance Measurements

Accuracy may not be an appropriate measure to evaluate the performance of classifiers for imbalanced datasets. A classifier may be able to obtain very high accuracy by classifying all

instances to majority class which outnumbers the minority class. Therefore, the use of accuracy and mean square error rate are inappropriate for IDS. So we adopted the use of other kinds of evaluation metrics to measure the classifiers' capability to differentiate the two-class problem under the case of imbalanced data. They are namely, ROC (Receiver Operating Characteristic) analysis, F-measure, and Geometric Means Measure. These performance measures are independent to prior probabilities.

Table 6-2 A Confusion Matrix for False Acceptance and False Rejection

		Classifier output	
		Misclassified	Classified
Desired output	Misclassified	True Negative (TN)	False Positive (FP)
	Classified	False Negative (FN)	True Positive (TP)

Sensitivity and specificity are statistical measures of the performance of a binary classification test to conduct the ROC analysis. The sensitivity measures the proportion of actual positives which are correctly identified. The specificity measures the proportion of negatives which are correctly identified. The sensitivity and specificity are calculated from true positive (TP), false negative (Christopher L), false positive (FP), and true negative (Guyon et al.) in which a confusion matrix of these four outcomes is tabulated as in Table 6-2. Thus, the sensitivity and specificity are defined. Recall or sensitivity is the percentage of the positive labeled instances which are predicted as positive. Specificity is the percentage of negative labeled instances that are predicted as negative. Their formulas are given by the followings:

$$\text{Recall/Sensitivity} = \frac{\text{True Positive}}{\text{True Positive + False Negative}}, \tag{6.8}$$

$$\text{Specificity} = \frac{\text{True Negative}}{\text{True Negative + False Positive}}. \tag{6.9}$$

Besides, in order to evaluate the generalization capability of the model, some metrics are necessary to be utilized. For instances, precision is the percentage of positive predictions that are correct. Accuracy is more general that it is the total of correct predictions over all data. F-measure is a metric derived from recall and precision. Some variants using different weighting make them equal weighted as we consider false positive and false negative equal likely to occur. According to the evaluations by Barandela *et al.* [43], geometric means measure (GMM) is a more appropriate metric to evaluate the classifier performance on IDS. Both ROC and GMM are good indicators as they try to maximize the accuracy on each of the two classes while keeping these accuracies balanced. The GMM is defined as the square root of the product of accuracy on positive samples and negative samples. Their formulas are shown below:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive + False Positive}}, \tag{6.10}$$

$$\text{Accuracy} = \frac{\text{True Positive + True Negative}}{\text{True Positive + True Negative + False Negative+ False Positive}}, \tag{6.11}$$

$$\text{F-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision + Recall}}, \tag{6.12}$$

$$\text{GMM} = \sqrt{\text{Accuracy}_{+ve} \times \text{Accuracy}_{-ve}}\ . \tag{6.13}$$

6.5 Experimental Results

Two major sets of medical data are used for assessing the performance of our proposed Support Vector Emergent Self Organizing Map model. The data being inspected are the breast tissue data (Silva, Sá, & Jossinet, 2000), and other medical datasets which were extracted from the UCI Machine Learning Repository. The breast tissue data was provided by Silva *et al.* (Silva et al., 2000). It contains the electrical impedance measurements performed on 106 excised breast tissue samples. The impedance measurements were taken at seven frequencies. A total of nine features

were computed from the impedance spectrum. There are five types of breast tissue classes in this dataset which represent different types of breast tissue, some are normal while others are pathological. The five types of breast tissue classes are namely Carcinoma Tissue (Car), Mastopathy Tissue (Mas), Glandular Tissue (Gla), Adipose Tissue (Adi), and Connective Tissue (Con).

The other IDS are being used, namely Pima Indians Diabetes (PIMA) data and Wisconsin Breast Cancer Diagnostic (WBCD) data. The PIMA dataset contains the data from all female patients of at least 21 years old, and of Pima Indian heritage. The database consists of 768 instances, each with 8 attributes. A total of 268 patients were diagnosed as having diabetes and 500 patients are healthy person without diabetes. The WBCD data is to diagnose benign and malignant cancer. The features are computed from digitized image of a fine needle aspirate (FNA) of a breast mass which described the characteristics of the cell nuclei. The real-valued features are computed for each cell nucleus like radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, fractal dimension and etc. WBCD data consist of 569 instances with 32 binary attributes.

Table 6-3 summarizes the imbalanced datasets that we used in our experiments. The first five rows represent the five types of breast tissue classes. The number of features, total instances, minimum class percentage and the imbalance ratio (IR) are listed. All are arranged into binary decision problem to predict instances as either positive or negative. The imbalance ratio (IR) is obtained by dividing the number of positive samples over the number of negative samples. A dataset is termed balance if the imbalance ratio is one. The IR of the datasets is imbalance with the ratio of ranging from 0.18 to 0.59. In most medical data, positive samples correspond to having infected the disease, whereas negative samples means the samples do not response to the test, hence not infected. Each of the dataset is divided into three stratified cross-validation sets for

Table 6-3 Different datasets with imbalance ratio (IR)

Datasets	Features	Instances	Class(minority, majority)	Min %	IR
CAR	9	106	(Carcinoma Pathological Tissue, others)	19.81	0.25
MAS	9	106	(Mastopathy Pathological Tissue, others)	16.98	0.20
GLA	9	106	(Glandular Normal Tissue, others)	15.09	0.18
ADI	9	106	(Adipose Normal Tissue, others)	20.75	0.24
CON	9	106	(Connective Normal Tissue, others)	13.21	0.25
SPECT	22	267	(Abnormal, normal)	13.21	0.25
WBCD	30	569	(malignant, benign)	37.26	0.59
PIMA	8	768	(Diabetic, healthy)	34.9	0.54

training and testing. Training set is made up of 2/3 of the overall data and testing set is made up of the remaining 1/3.

We first observe the learning capability of Support Vector Machine in selecting good features. The derivation of Support Vector Machine selects a subset of features that are representative for the whole data set as shown in Table 6-4. Figure 6-8 shows the accuracy of Support Vector Machine with respect to the number of features. We start with two features up to ten features. The breast cancer data (WBCD) achieves 83% of accuracy with two features. The accuracy increases gradually up to fourth features. The accuracy maintains at the level of $96.3 \pm 1.05\%$ subsequently. The heart data (SPECT) starts with 70% for 2 features. In contrast with breast cancer data, the heart features achieve consistent accuracy ($74.1 \pm 1.14\%$) starting from the third features. In the case of diabetes data (PID), the accuracy is around $63.1 \pm 8.4\%$. There are only 8 features available for diabetes data. The classification phase makes use of emergent self-organizing map to segregate data of two classes. The first three features of SPECT correspond to feature 2, 12, and 24. The first three features of WBCD correspond to feature 20, 13, and 1. Features 6, 7 and 2 are the three most important features of PID correspond to the body

mass index, diabetes pedigree function, and plasma glucose concentration as depicted in Table 6-4.

Table 6-5 depicts the result of the dataset using 3 attributes and full set of attributes. The recalling rates of various tests are studied. The first three recalling rates adopt only heart data's 3 features out of the 22 features. Row 4 to 6 displays the result using full set of features, i.e. without using SVM to select the features. The training time is reduced by around 5 seconds using reduced attributes and the performance ($93.0 \pm 0.28\%$) is very close to that of full features ($95.0 \pm 1.12\%$). The breast cancer data with selected features achieves recalling rate of $87.5 \pm 1.45\%$. The original features recalling rate is higher, $98.3\% \pm 0.4\%$. The generalization of SPECT dataset is $74.2 \pm 3.37\%$. The full features achieve $73.6 \pm 1.68\%$. This is very encouraging as the subset attributes contribute higher generalization rate than original features. The breast cancer data with selected features' generalization is $86.0 \pm 1.28\%$. The full features results are surprisingly good, $98.7 \pm 0.79\%$. This shows that the unsupervised learning approach of classification is superior over the medical data.

In a Receiver Operating Characteristic (Bernhard E. Boser) curve, the sensitivity (true positive rate) is plotted in function of specificity (false positive rate) for different cut-off points. A test with perfect discrimination (no overlap in the two distributions) has a ROC plot that passes through the upper left corner. Therefore the closer the ROC plot is to the upper left corner, the higher the overall accuracy of the test Figure 6-9. As shown in Figure 6-9 is the ROC curve of SPECT heart data and WCBD breast cancer data. It provides good curve for all data as shown. The other results are included in Appendix A1. For ROC analysis, the curve is plotted by the function of sensitivity against specificity for different cut-off points. This demonstrates a trade off between true positive and false positive rates provided with different classification criteria. A result with perfect classification, i.e., no overlapping in both distributions of sensitivity and

specificity, has a ROC curve that passes through the upper left corner. Therefore, the closer the ROC curve to the upper left corner, the higher the accuracy of the test. Figure A1- show different ROC curves performed in different IDS obtained by both zero-order and first-order SVESOM. They show that for all these imbalanced datasets, both of them are able to produce good results in this ROC analysis. Particularly, the first-order SVESOM can achieve better performance for CAR, MAS and PIMA datasets, whereas in WBCD dataset, the zero-order SVESOM achieve better results than the first-order SVESOM.

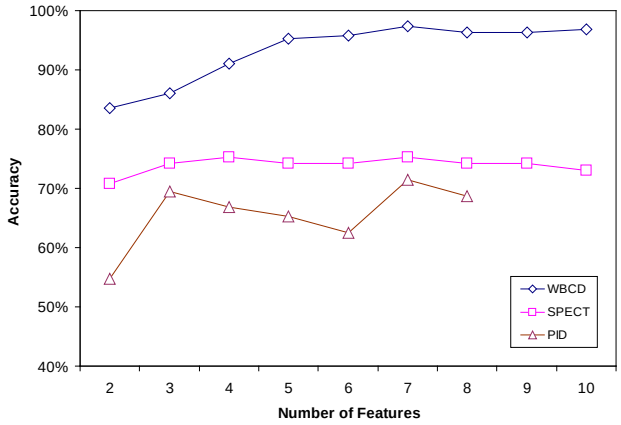


Figure 6-8 Support Vector Machine features selection performance

Table 6-4 Features ranking of PIMA indian diabetes data

Rank	Feature Index	Feature Names
1	6	Body mass index
2	7	Diabetes pedigree function
3	2	Plasma glucose concentration
4	3	Diastolic blood pressure
5	8	Age
6	4	Triceps skin fold thickness
7	1	Number of times pregnant
8	5	serum insulin

Table 6-6 and Table 6-7 depict the generalization results of zero-order and first-order SVESOM respectively in terms of F-measure, GMM (training and testing) and Accuracy. The results of other dataset can be found in Appendix A1. It can be observed that the performance of the first-order SVESOM is better than that of the zero-order SVESOM in most imbalanced datasets because of the absolute value of derivatives. It is also shown that the use of only three features ranked by the support vector ranking provides sufficient information as much as the full set of features. Such results show that using three or four features is sufficient to obtain good generalization performance.

Table 6-5 Performance of Medical Data

	Test	Recalling Rate	Generalization Rate
Selected features of SPECT	1	93.3%	73.0%
	2	92.7%	77.5%
	3	93.3%	70.8%
Full set of SPECT	1	93.8%	71.9%
	2	96.1%	74.2%
	3	95.5%	75.3%
Selected features of WBCD	1	86.0%	85.8%
	2	88.9%	84.7%
	3	88.4%	87.3%
Full set of WBCD	1	97.9%	99.5%
	2	98.7%	97.9%
	3	97.9%	98.9%
Selected features of PID	1	82.4%	67.2%
	2	83.6%	64.5%
	3	86.9%	68.0%
Full set of PID	1	81.1%	62.1%
	2	80.1%	64.1%
	3	82.0%	61.7%

Our approach is benchmarked against other approaches, such as, Bayes-Net, Naïve Bayes, RBF Network, Bagging and J48 decision tree as shown in Table 6-8. The breast cancer data requires four features to achieve accuracy of more than 91%. The average hit rate is highest for zero order (0.96) and followed by first order. The classification performance for zero order is constant and stable. The best performance for Bayes Net is 0.885 with top 6 ranked features, and

average accuracy of 93.1 $\pm$ 2.64%. The highest hit rate for NaiveBayesian is 0.914 and accuracy of Naïve Bayes for more than 3 features is 93.4 $\pm$ 1.85%. The best performance for RBF Network is 0.93 which is slightly weaker than zero order and average of 94.2 $\pm$ 2.11%. The random forest performance is in the range of 94.6 $\pm$ 2.11%. Bagging is performing as good as 0.911 and J48 at 0.911 as well. Our approach suggests good generalization capability with reduced training features in most cases.

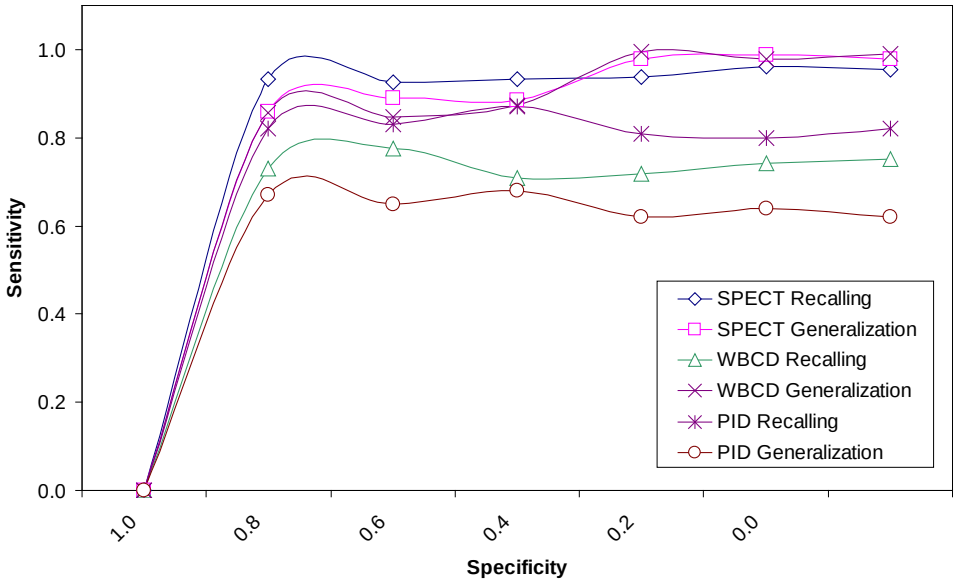


Figure 6-9 ROC curve of medical data

Table 6-6 Generalization results of zero-order SVESOM applied to MAS

	Number of Features for MAS IDS							
	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**
F-Measure	0.75	0.80	0.86	0.89	0.88	0.91	0.82	0.85
GMM(training)	0.91	1.00	1.00	1.00	1.00	0.98	0.97	0.97
GMM(testing)	0.35	0.46	0.67	0.73	0.69	0.70	0.66	0.69
Generalization/Accuracy	0.67	0.67	0.75	0.81	0.78	0.86	0.72	0.78

Table 6-7 Generalization results of first-order SVESOM applied to MAS

	Number of Features for MAS IDS							
	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**
F-Measure	0.87	0.80	0.80	0.85	0.80	0.91	0.89	0.86
GMM(training)	0.75	0.37	0.37	0.53	0.00	1.00	0.98	1.00
GMM(testing)	0.97	0.97	0.94	0.99	0.61	0.72	0.72	0.64
Generalization/Accuracy	0.96	0.96	0.96	0.99	0.80	0.83	0.83	0.76

Table 6-8 Benchmarking results of the zero-order and first-order SVESOM against other classifiers

Rank	Zero-order SVESOM		First-order SVESOM		Bayes Net		NaiveBayesian		RBF Network		Bagging		J48	
2	0.96	1	0.86	6	0.853	7	0.901	4	0.93	2	0.911	3	0.898	5
3	0.96	1	0.95	2	0.84	7	0.914	4	0.924	3	0.911	5	0.911	5
4	0.96	1	0.96	1	0.853	7	0.914	3	0.911	4	0.911	4	0.898	6
5	0.96	1	0.95	2	0.827	7	0.881	6	0.941	3	0.911	4	0.898	5
6	0.96	1	0.87	7	0.885	6	0.892	5	0.914	2	0.911	3	0.898	4
7	0.96	2	0.97	1	0.885	6	0.88	7	0.914	3	0.911	4	0.898	5
8	0.96	1	0.96	1	0.8615	7	0.911	4	0.927	3	0.88	6	0.911	4
9	0.96	1	0.96	1	0.8615	6	0.858	7	0.927	3	0.911	4	0.898	5
Avg Ranking	1.1		2.6		6.6		5.0		2.9		4.1		4.9	

6.6 Conclusion

The previous chapters presented the development of the model and the problem of imbalanced data. In this chapter, we introduce the problem domain and formulation. The analysis to discover the problems underlying imbalansed data will be performed. We then utilize the prototype ranking framework as described in Chapter 5 to solve the imbalanced data under the medical domain. In the next chapter, we propose to implement our algorithm of Prototype Ranking with Self-Organizing Learning into a case study with questionnaire based emotion profiling. Experimental results show that the top ranked features are in-line with the work of Wallbott and Scherer published in 1994 which approached the emotions physiologically.

Chapter 7 Case Study: Emotion Understanding and Interpretation

7.1 Background

Facial expression recognitions are laboratory approaches that mandate the use of databases with directed facial expressions or acted tones to provide training for the recognition systems. The emotions are usually exaggerated and very rarely appear in real world. It is difficult to study emotional experiences by using laboratory techniques of emotion induction or field observation. Emotion induction in the laboratory is often inefficient as it results in weak emotional experience due to the fact that strong emotions are difficult to undertake in real like situations outside the laboratory. This leads to another category of emotion recognitions - the questionnaire approach. In a questionnaire approach, the subjects are given a set of questions to help them recall occasions which they have experienced different emotions. This is a self-reported measure in measuring affect. Although it can be argued that questionnaires are only capable of measuring only conscious experience of emotion, and most of the affective process are non-conscious processes (Wallbott and Scherer, 1988). But this approach represents the most direct ways to measure sentiment and it

is not artificial because it is based on the recalling of personal encounters. Questionnaire approach to the study of emotional processes by asking subjects to describe emotional situations and the reactions experienced is a suitable way to study not only emotion-eliciting situations but also emotional reactions. They also (Scherer and Wallbott, 1994) argued that the justification of questionnaire approach to emotion study is that it is preferable to have access to real, intimate emotions through verbal report on recalled emotion experiences. Secondly, the two important components of the total emotion process, cognitive appraisal of emotion-antecedent situations and subjective feeling state are only accessible through self-report.

During the conjunction of nineteenth and twentieth century, a large group of psychologists directed by Scherer and Wallbott (Wallbott and Scherer, 1988; Scherer and Wallbott, 1994; Scherer, 1997) collected cross cultural differences in the reaction patterns for the seven emotions over 10 years in the Intercultural Study on Emotional Antecedent (ISEAR). The studies initially collected data in eight European countries, Japan, and the United States. They subsequently evolve to include close to 3000 respondents from 38 countries that cover the 5 continents. Student respondents, both psychologists and non psychologists, were asked to report situations in which they had experienced all of 7 major emotions (joy, fear, anger, sadness, disgust, shame, and guilt). In each case, the questions covered the way they had appraised the situation and how they reacted.

The response domains cover subjective feelings, physiological symptoms, and motor expression patterns of the seven emotions. The intensities of the measures are rated by the subjects. The published work in Scherer and Wallbott, 1994 extrapolated from reported predictions on the basis of individual expressive behavior variables as shown in Table 7-1 Predictions for Significant Emotion Differences Concerning Subjective Feeling, Physiological Symtoms, and Expressive Behaviours. The questionnaires consist of four parts: (a) situation description; (b) subjective feeling state; (c) physiological symptoms, expressive behavior, and other reactions; and (d) appraisal. For the subjective feeling domain, predictions were made for the

dimensions of intensity and duration of the affective state, attempts at controlling the state, and how long ago the event took place. The physiological symptoms report individual reactions or symptoms that they recall as having experienced. Cardiovascular, muscle symptoms, and perspiration predictions were translated into ergotropic arousal predictions. Stomach symptoms, lump in throat, and crying predictions were interpreted in terms of trophotropic arousal. Similarly, the expressive behavior variables were grouped into movement, nonverbal-nonvocal, paralinguistic, and speech behavior (Scherer and Wallbott, 1994).

The experiment sections will demonstrate how the model effectively identifies the important variables by giving discriminative features with high ranking. The classifier then performs the classification based on the selected features. Experimental results show that the top rank features are in-line with the work of Scherer and Wallbott (Scherer & Wallbott, 1994) which approached the emotions physiologically. Whilst the performance measures show that using the full features for classifications can degrade the performance, the selected features provide superior results in terms of accuracy and generalization.

7.2 Emotion Profiling

In our study, we refer to the study from the work in Scherer and Wallbott (1994) which extrapolated from reported predictions on the basis of individual expressive behaviour variables. The questionnaires consist of four parts: (a) situation description; (b) subjective feeling state; (c) physiological symptoms, expressive behaviour, and other reactions; and (d) appraisal. Table 7-1 illustrates the predictions of different emotions significantly concerned by subjective feeling, physiological symptoms and expressive behaviours. For the subjective feeling domain, predictions were made for the dimensions of intensity and duration of the affective state, which attempts at controlling the state, and how long ago the event took place. The physiological symptoms report individual reactions or symptoms that they recall. Cardiovascular, muscle symptoms, and

perspiration predictions were translated into ergotropic arousal predictions. Stomach symptoms, lump in throat, and crying predictions were interpreted in terms of trophotropic arousal. Similarly, the expressive behaviour variables were grouped into movement, nonverbal-nonvocal, paralinguistic, and speech behaviour (Scherer and Wallbott, 1994).

Table 7-1 Predictions for significant emotion differences concerning subjective feeling, physiological symtoms, and expressive behaviours

Measure	Prediction
Subjective feeling	
Time distance (long ago-recent)	Sad=Fear<Joy=anger
Intensity (weak-strong)	Anger=Fear<Sad=Joy
Duration (short-long)	Fear<Anger<Joy=Sad
Control attempts (weak-strong)	Joy<Fear=Sad=Anger
Physiological symptoms	
Ergotropic arousal (weak-strong)	Sad=Joy<Anger<Fear
Trophotropic arousal (weak-strong)	Joy<Fear=Anger<Sad
Felt Temperature (cold-warm)	Fear<Sad<Joy<Anger
Expressive behaviours	
Approach/withdrawal (away-towards)	Fear=Sad=Anger<Joy
Nonverbal behaviour (little-much)	Fear<Sad<Joy=Anger
Paralinguistic behavious (little-much)	No predictions available
Verbal behaviour (little-much	Fear=Sad<Joy=Anger

Note: < indicates significant difference will be found between groups of emotions at either side of the sign; = indicates no significant differences are expected

Forty variables are extracted from the questionnaires in which they are interpreted by different questions as shown in Table 7-2. With the huge database (i.e. over 10,000 records), it is difficult to classify as the classifiers may be confused by the overwhelming data. It is due to the fact that the distributions of dataset are imbalanced. Based on the principle of Occam's razor, classifiers generally do not perform well on imbalanced datasets because they are designed to generalize from the limited numbers of sampling data and output the simplest hypothesis that best fits to the data. Imbalanced data set (IDS) is a phenomenon occurs where the number of instances in one class significantly outnumbered the instances from other classes. Therefore, the training data is dominated by the instances belonging to one class. We thereby introduce an approach that

is derived from support vector machine that can actively select the appropriate variables according to the different emotions.

Our proposed methodology to realize the questionnaire approach for emotion profiling is similar to the algorithmic approach which comprises of the Support Vector Machine (SVM) based criterion ranking feature selection and Emergent Self-Organizing Mapping (ESOM) for unsupervised classification. Figure 7-1 shows the flow diagram that illustrates the Support Vector ESOM (SVESOM). The input data is first trained by SVM classifier and the ranking criterion are evaluated for feature ranking. The data is then clustered by the ESOM algorithm and such clusters are assigned for classification. The input space consists of the full attributes data. R_c is the ranking criteria and Var provides the vector that ranks the best features in ascending order.

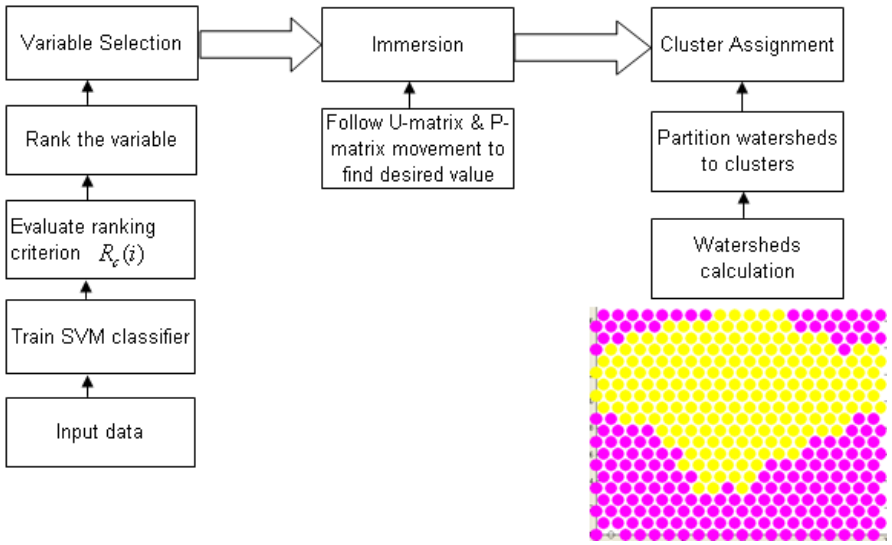


Figure 7-1 The flow structure of the Emergent Self-Organizing Learning with Support Vector feature ranking. The input data are first trained by SVM classifier and the ranking criterion are evaluated for feature ranking. The data are then clustered by the ESOM algorithm and such clusters are assigned for classification

Table 7-2 Interpretations of the variables reading

Var	Interpretations
RELI	Subject's religion
PRAC	Subject practicing religion
FOCC	Father's occupation
MOCC	Mother's occupation
FIEL	Subject's field of study
WHEN	When did this happen?
LONG	How long did you feel the emotion?
INTS	How intense was this feeling?
ERGO	Ergotropic arousal
TROPHO	Trophotropic arousal
TEMPER	Felt temperature
EXPRES	Non-verbal activity (laughing/crying and etc)
MOVE	Movement behaviour (move towards people?)
EXP1	Laughing, smiling
EXP2	Crying, sobbing
EXP10	Moving against people or things
PARAL	Paralinguistic activity
CON	Did subject try to control the feeling?
EXPC	Did subject expect the situation to occur?
PLEA	The event is pleasant or unpleasant?
PLAN	Did the event help in subject's plans or aims?
FAIR	The event is fair?
CAUS	Who is responsible for causing the event?
COPING	Can subject cope with the event?
MORL	The event is improper or immoral?
SELF	Did the event affect subject's self-esteem?
RELA	Did event change subject's relationship with people?
VERBAL	Verbal activity (silence /lengthy utterance)

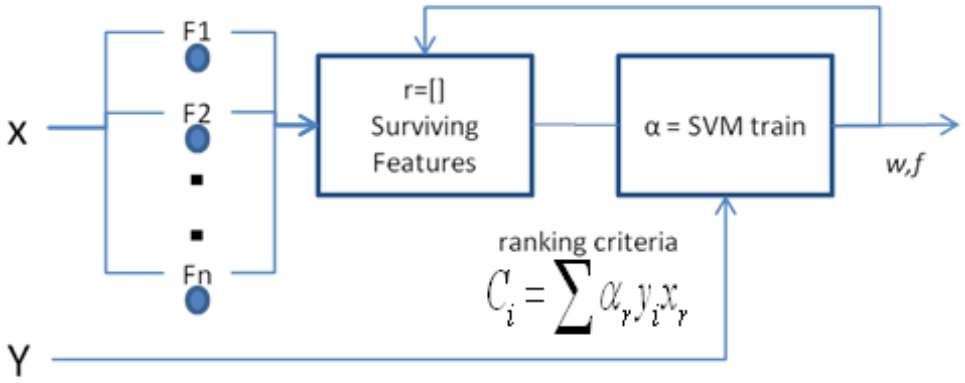


Figure 7-2 Overview of prototype ranking

The next step in prototype ranking is SVM training as shown in Figure 7-2. The prototypes are batch trained together with the criterion. The remaining values of surviving features continued to be recursively ranked till all criterions are met and all rankings are finalized. The prototype rankings are derived from Support Vector Machines and are based on weight vector sensitivity with respect to a prototype. It was initially proposed by Guyon *et al.* (Guyon et al., 2002b) for selecting genes that is relevant for a cancer classification problem. The squared coefficients $w_j^2 (j = 1,..., p)$ of the weight vector *w*are employed as feature ranking criteria. The prototypes in this context represent the grey intensity of the pixel. Prototypes are selected using ranking criterion to rank variables. The ranking criteria w_j^2 for all features are computed, and the prototype with the smallest ranking criterion is discarded. This is an extension to bounds on L error, margin bound and other bounds of the generalization error. The criterion being investigated is C_t which is either weight vector $\|w\|^2$, the radius/margin bound $R^2\|w\|^2$ or the span estimate. It gives either an estimation of the generalization performance or an estimation of the dataset separability. It was initially proposed by Guyon *et al.* (2000) for selecting genes that is relevant for a cancer classification problem. The goal is to find a subset of size r among d features where $r < d$ which maximizes the performance of classifier (A. Rakotomamonjy, 2003a). The method is based on backward sequential selection. The features are removed one at a time until r features. The criteria for selection are derived from Support Vector Machines (SVM) and are based on weight vector sensitivity with respect to a variable (Guyon et al., 2002b). Variables are selected using ranking criterion to rank variables. This is an extension to bounds on L error, margin bound and other bounds of the generalization error (A. Rakotomamonjy, 2003a). The criteria being investigated is C_t which is either weight vector $\|w\|^2$, the radius/margin bound $R^2\|w\|^2$ or the span estimate. It gives either an estimation of the generalization performance or an estimation of

the dataset separability. The removed variable is the one whose removal minimizes the variation of $\|w\|^2$.

After finding the top ranked features, we then use the features for Emergent Self-Organizing Mapping (ESOM). The Emergent Self-Organizing Map is a non-linear projection technique using neurons arranged on a map. It is an extension of Self-Organizing Map (SOM) where the neurons go through competitive learning. The goal of SOM is to transform an incoming signal pattern of arbitrary dimension into a one- or two- dimensional discrete map, and to perform this transformation adaptively in a topologically ordered fashion.

7.3 Experimental Results

We shall now assess the experimental results on ISEAR emotion questionnaire. Forty variables are extracted from the questionnaires using different questions. With the huge database (i.e. over 10,000 records), it is difficult to classify as the classifiers may not be able to handle the overwhelming amount of data. The difficulty in classification arises from the fact that the distributions of dataset are imbalanced. Based on the principle of Occam's razor (Thorburn 1915), classifiers generally do not perform well on imbalanced datasets because they are designed to generalize from the limited numbers of sampling data and provide the simplest hypothesis that best fits to the data. Imbalanced data set (IDS) is a phenomenon occurs where the number of instances in one class significantly outnumbered the instances from other classes. Therefore, the training data is dominated by the instances belonging to one class. We thereby introduce an approach which is derived from support vector machine that can actively select the appropriate variables according to the different emotions.

The experiments were carried out using three stratified cross-validation training and test sets. Two third of the dataset constitute testing and the remaining one third is for training. The

questionnaire data is arranged into two class problem, for instance, joy classification consists of joy emotion versus non-joy emotion and similarly for the other emotions. The ratios of the emotions are maintained in the cross validation experiments. The International Survey on Emotion Antecedents and Reactions (ISEAR) dataset was constructed by a group of psychologists during the 1990. Student respondents, both psychologists and non-psychologists, were asked to report situation in which they had experienced the 7 major emotions (joy, fear, anger, sadness, disgust, shame, and guilt). In each case, the questions covered the way they had appraised the situation and how they reacted. The ISEAR contains 40 variables. This section starts with analyzing the important features selected and how these features link to psychological states and behaviour. The ROC analysis and performance measurements are discussed in the following sections. Lastly the data visualizations are shown to demonstrate the distribution of data after clustering.

7.3.1 Feature Ranking Analysis

The priority of the variables represented by feature ranking corresponding to different emotions is tabulated in Table 7-4. The corresponding measures are shown in Table 7-3. The ten selected features are meant to describe an emotion collectively. Joy is a happy emotion and it usually last long (Pepe et al.). When a subject experiences joy, he/she is less likely to control (CON) or hide the emotion as it is felt intensely (INTS). Laughter is the result of expressing (EXP1) joyfulness. Fear is an emotion that is difficult to cope with (COPING) as it is an emotion with great obstacles to work with. The first natural reaction is to feel afraid and too intense to be controlled. It takes some times (Pepe et al.) for this feeling to be subsided. Anger is another emotion which is difficult to control (CON) (Wilkowski & Robinson, 2008). The subject expresses the anger through facial expression directly (EXPRES) or uttering (VERBAL) about the anger felt. The expression of anger differs from individuals; Figure 7-3 (third picture from left) shows an anger face with lowered eyebrows, straightened lower eyelids, and tensed lips. Sad is a difficult feeling to cope

with (COPING) and it persists for a long period of time (Pepe et al.). If the sadness continues to linger for a longer period of time, it could lead to depression that affect subject's relationship with the others (RELA). Disgust, shame and guilt are emotions that peoples find it difficult to cope with (COPING) as they belong to the negative emotion which is unpleasant. Disgust typically

Figure 7-3 Six basic emotions and neutral expression from JAFEE database (MJ, J, & S, 1999)

Table 7-3 Predictions of emotions in accordance to different subjective feeling, physiological symptoms and expressive behaviours together with the corresponding variables and interpretation

Measure	Prediction	Variable	Interpretation
Subjective feeling			
Time distance (long ago-recent)	Sad=Fear<Joy=anger	WHEN	When did this happen?
Intensity (weak-strong)	Anger=Fear<Sad=Joy	INTS	How intense was this feeling?
Duration (short-long)	Fear<Anger<Joy=Sad	LONG	How long did you feel the emotion?
Control attempts (weak-strong)	Joy<Fear=Sad=Anger	CON	Did subject try to control the feeling?
Physiological symptoms			
Ergotropic arousal (weak-strong)	Sad=Joy<Anger<Fear	ERGO	Ergotropic arousal
Trophotropic arousal (weak-strong)	Joy<Fear=Anger<Sad	TROPHO	Trophotropic arousal
Felt Temperature (cold-warm)	Fear<Sad<Joy<Anger	TEMPER	Felt temperature
Expressive behaviours			
Approach/withdrawal (away-towards)	Fear=Sad=Anger<Joy	MOVE	Movement behaviour (move towards people?)
		EXP10	Moving against people or things
Nonverbal behaviour (little-much)	Fear<Sad<Joy=Anger	EXPRES	Non-verbal activity (laughing/crying and etc)
		EXP1	Laughing, smiling
		EXP2	Crying, sobbing
Paralinguistic behavious (little-much)	No predictions available	PARAL	Paralinguistic activity
Verbal behaviour (little-much	Fear=Sad<Joy=Anger	VERBAL	Verbal activity (silence /lengthy utterance)

Note: < indicates significant difference will be found between groups of emotions at either side of the sign; = indicates no significant differences are expected

comes with facial expression (EXPRES) like wrinkled nose with eyebrows pulled down, upper lip drawn up, lower eyelid is tensed and eye opening narrowed as shown in Figure 7-3 (second picture from right). Guilt is an intense (INTS) emotion which can be difficult to control (CON) and cope with (COPING). Some peoples deal with guilt by turning to self- destructive behaviour such as excessive drinking. The ten features selected for each emotion as tabulated in Table 7-4 collectively explain the emotion and the distinguishing attributes to differentiate the different emotions.

Table 7-4 Feature Ranking by SVESOM for seven different emotions

Feature Ranking	1	2	3	4	5	6	7	8	9	10
Joy	PLEA	LONG	CON	ERGO	TEMPER	PLAN	EXPC	EXP1	MORL	INTS
Fear	COPING	CAUS	INTS	ERGO	FAIR	MORL	EXPRES	VERBAL	LONG	PLAN
Anger	WHEN	CAUS	LONG	VERBAL	PLAN	MOCC	FOCC	EXPRES	CON	ERGO
Sad	RELA	LONG	CAUS	FOCC	COPING	MORL	WHEN	NEUTRO	EXPC	PLAN
Disgust	COPING	PLAN	WHEN	MORL	PLAN	FAIR	EXPRES	ERGO	SEX	PRAC
Shame	COPING	EXPC	PRAC	WHEN	LONG	INTS	PARAL	MORL	CAUS	VERBAL
Guilt	AGE	COPING	FAIR	WHEN	LONG	CON	RELA	INTS	MORL	PLAN

7.3.2 Sensitivity and Specificity

Sensitivity and specificity measures are adopted in our experiments. Both sensitivity and specificity are statistical performance measures of a binary classification test. The sensitivity measures the proportion of actual positives which are correctly identified. The specificity measures the proportion of negatives which are correctly identified. The sensitivity and specificity are calculated between the values of true positive (TP), false negative (FN), false positive (FP), and true negative (TN).

Recall or sensitivity is the percentage of the positive labelled instances which are predicted as positive. Specificity is the percentage of negative labelled instances that are predicted as negative. The sensitivity and specificity are used for plotting the ROC (Receiver Operating Characteristic) curve to illustrate their performances. The ROC curves of the training and test sets are shown in Figure 7-4 and Figure 7-5 respectively. In this ROC analysis, the curve is plotted by

the function of sensitivity against specificity for different cut-off points. This demonstrates a trade off between the rates of true positive and false positive provided with different classification criteria. The perfect classification, i.e., no overlapping in both distributions of sensitivity and specificity, means that the ROC curve would pass through the upper left corner. As shown in

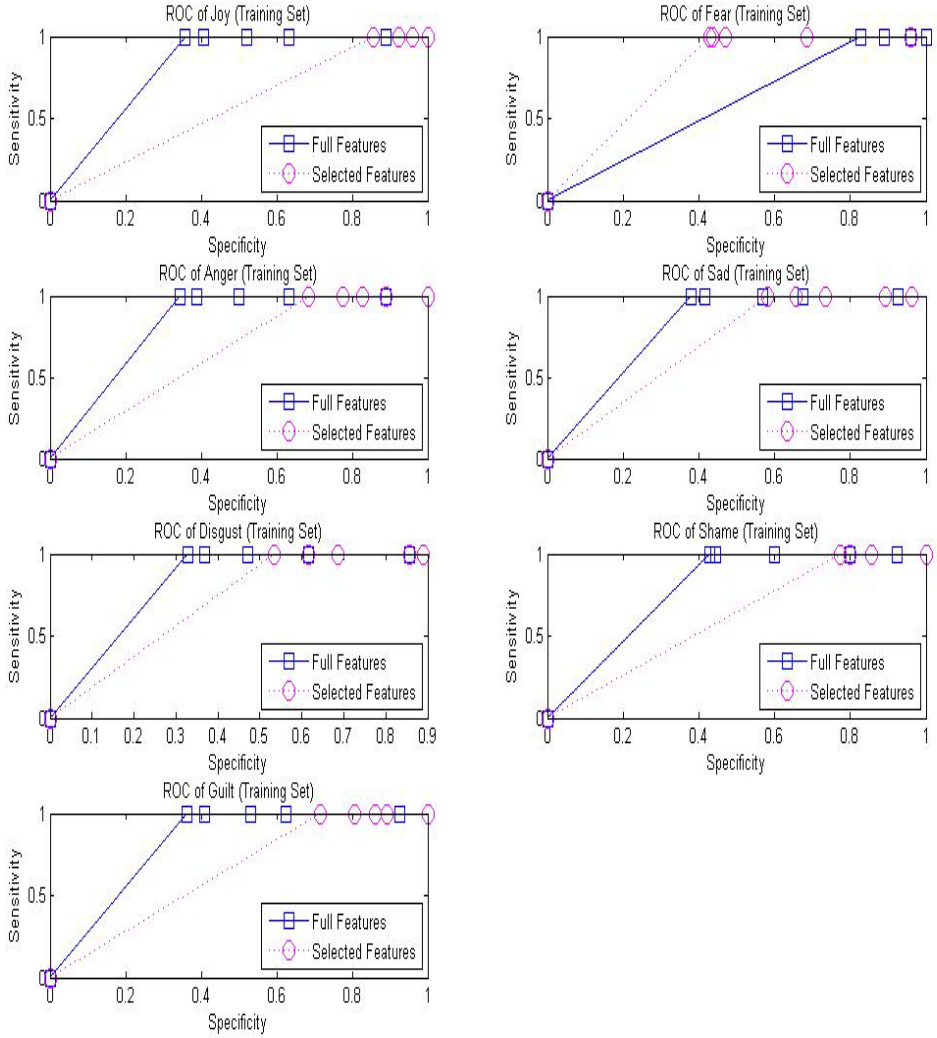


Figure 7-4 The ROC analysis of using top four most discriminative features versus full features in training set of questionnaires emotion data

Figure 7-4 and Figure 7-5, by using both four highest ranking features and full features respectively good ROC performances can be achieved for all the emotions considered. For consistency purpose, we took four most discriminative features in this ROC analysis for all the seven types of emotions. The four features vary with different profiling of emotions. By taking the joy emotion for example, the four selected features correspond to pleasant, duration, control and ergotropic. The analysis for the feature ranking will be discussed in the next sub-section.

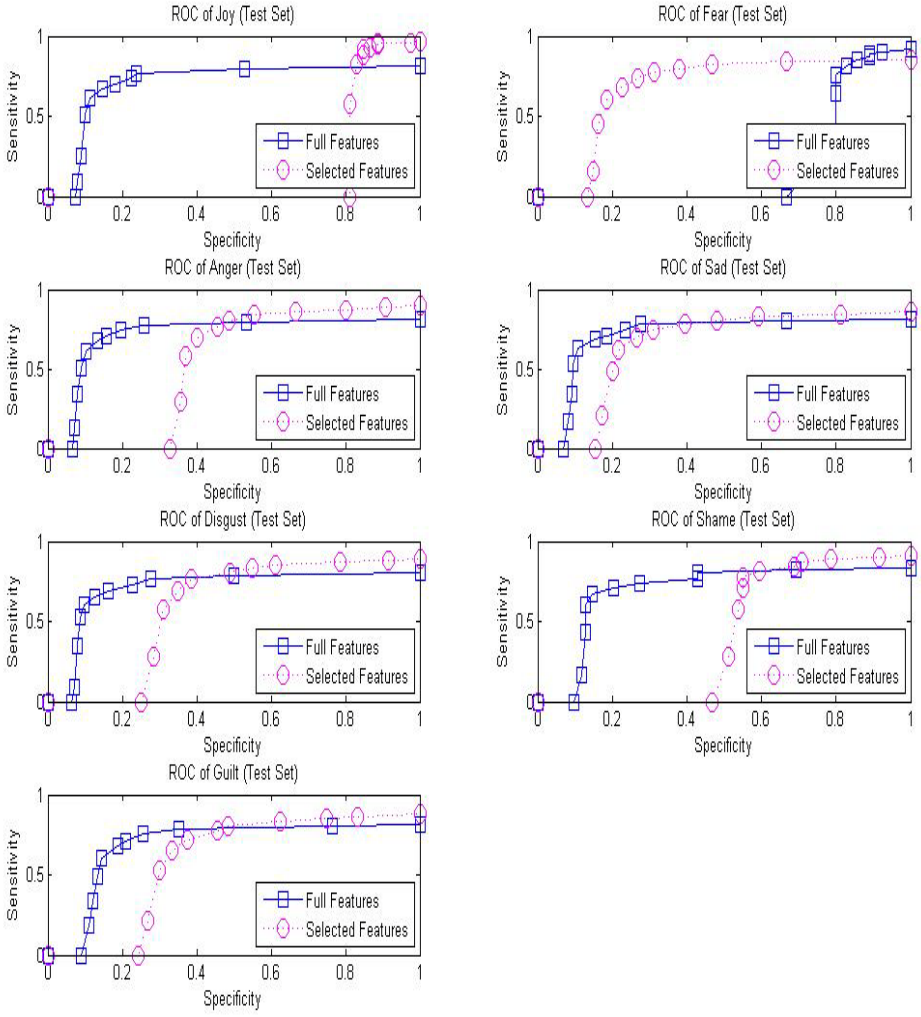


Figure 7-5 The ROC analysis of using top four most discriminative features versus full features in test set of questionnaires emotion data

7.3.3 Performance Evaluations

The experimental results of joy emotion are shown in Table 7-5. Three sets of results are displayed to show the best, moderate and the worst performances among the seven emotions of joy, fear, anger, sadness, disgust, and guilt. In these tables, each row represents different numbers of selected features as the input to perform the classifiers. The corresponding results in terms of generalization, sensitivity, F-measure, geometric means measure for both training and test datasets are shown in each column of the tables. Joy is the most easily distinguished emotion with F-measure constantly above 90%. The first three features are insufficient to allow the competitive learning of ESOM to converge. However, with the use of 4 features onwards, the convergence ensures. It demonstrates that the use of more features or full features does not guarantee a higher performance in terms of F-measure, generalization and geometric means. The result is consistent with the other emotions like anger, shame and etc. Most of the experiments in different emotional profiling are converged if more than 3 features were used. The fear emotion in Table 7-5 shows the median results among the emotions, the average results of the common emotions are around 72%. It is also noted that there is no significant improvement by increasing the number of features more than 4. It can be concluded that using 4 features are sufficient to profile the emotion by this questionnaire dataset in most cases. Moreover, the results of F-measure, which is the main performance metrics for imbalance datasets, often maintain at a relative high level of 0.7 or above. Note that disgust is a difficult emotion to be classified because the data collected from questionnaires do not have significant variations between the variables (Scherer, 1997).

In addition, the result obtained by our approach was benchmarked with the work of Danisman & Alpkocak (2008) in which the so-called Vector Space Model (VSM) was adopted to classify the ISEAR dataset and achieve an overall F-measure of 0.68 for 5 types of emotions (see the results detail in (Danisman & Alpkocak, 2008)). The overall benchmarking results are tabulated in Table 7-6. The results use 3 measures for comparison, Kappa measure, F-measure,

and accuracy. Kappa measure is a metric to measure the degree of agreement or disagreement of two or more people observing the same phenomenon. It can be implemented into any classifier that it is preferable to just counting the number of misses – even for those cases where all errors can rightfully be treated as being of similar importance (Arie, 2008). The results show that our SVESOM model outperforms the Vector Space Model (VSM) model since all the 7 types of emotions were analyzed in this ISEAR dataset and an overall F-measure of 0.82 was achieved by our proposed model. Moreover, our analysis gave individual results of the different emotions as opposed to the Danisman & Alpkocak approach which only showed the overall classification result for this ISEAR dataset.

Table 7-5 Performance result of shame emotion by SVESOM

Features	Generalization	Sensitivity	F-Measure	GMts	GMtr
2	0.55	0.63	0.59	0.48	0.75
3	0.93	0.94	0.94	0.96	0.96
4	0.95	0.96	0.96	0.97	0.98
5	0.97	0.96	0.97	0.98	0.98
6	0.98	0.98	0.98	0.97	0.99
7	0.96	0.98	0.97	0.98	0.99
8	0.96	0.96	0.96	0.98	0.98
All	0.89	0.96	0.92	0.59	0.84

Table 7-6 ISEAR Benchmarking results between Danisman & Alpkocak (2008) vs our approach

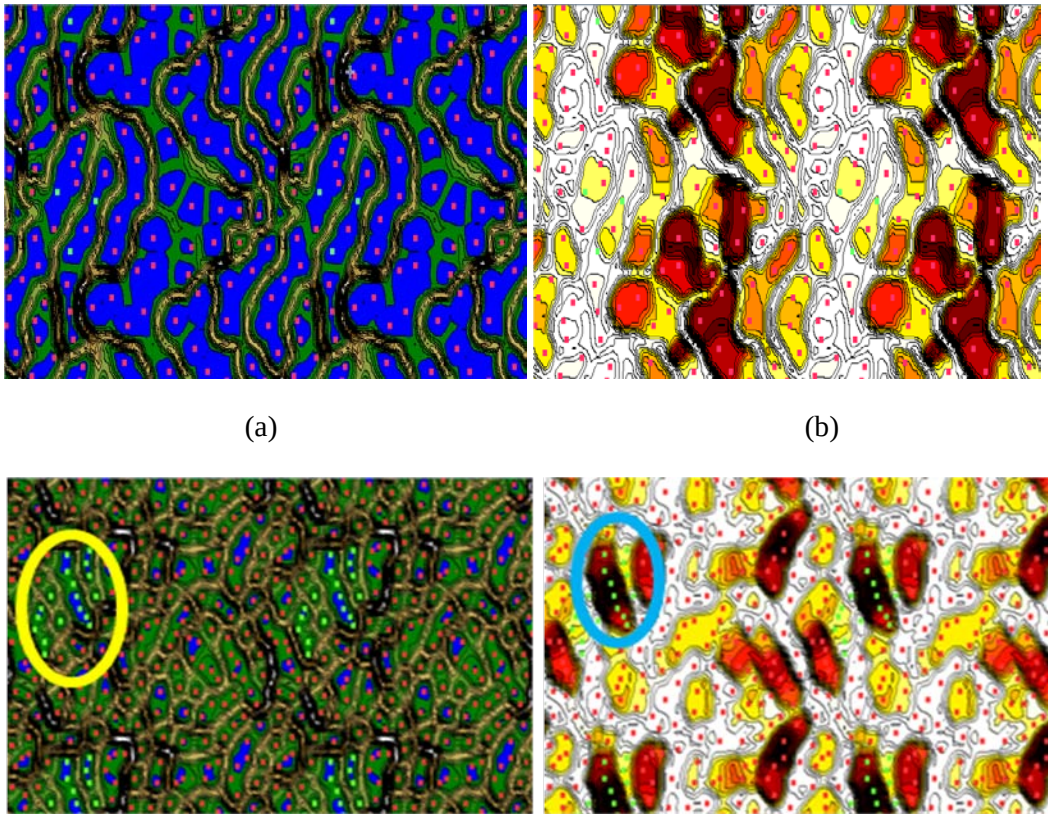
Measure	VSM 5 class emotions (Danisman & Alpkocak (2008))	SVESOM 7 class emotions
Kappa[1]	0.59	0.68
F-Measure	0.675	0.82
Accuracy	0.674	0.8

[1] **Kappa measure is a metric to measure the degree of agreement or disagreement of two or more people observing the same phenomenon. It can be implemented into any classifier that it is preferable to just counting the number of misses – even for those cases where all errors can rightfully be treated as being of similar importance (Arie, 2008).**

7.3.4 Data Visualization

This section presents the data visualization results generated by the ESOM with selected features as shown in Figure 7-6. More results can be found in Appendix A2. ESOM maps the original data

features onto two-dimensional axis. The locations of the data points represent the trained neurons after the competitive learning. The U-Matrix realizes the distances of the emergence of structural features within the data space. 2D toroid structure with Euclidean distance was adopted for the ESOM mapping. The adjacent U-matrix can be combined together to form a 'boundless' picture for such data distribution, for example the majority samples in red (see Figure 7-6 (c) and Figure 7-6 (d)) are distributed around the edges of the map. The structure of high dimensional data would be extracted from major blocks of the visualized results in the centre of U-Map. The questionnaire dataset can be visualized in the map space. The cluster of the minority class (i.e., the green points) is formed amidst the majority class (i.e. the red points). The maps show the visualization of shame emotion denoted as green points whereas red points denote non-shame emotion. The darker colour of the background in U-map implies larger separation between the neighbouring data points. Similarly, darker background colour in the P-Map implies that the area is denser. As shown in the maps of Figure 7-6(c) and (d), with the use of 4 features selected, the clusters of minority class started to appear ,as circled in the figures, The map is a tiled display of multiple maps and hence the cluster is repeated in the other regions.. The ESOM provides a low dimensional projection preserving the topology of the input space, thus the high dimensional distances can be visualized with the canonical U-Matrix, P-Matrix and U*-Matrix together so that the cluster boundaries can be distinguished in more sharpen. In addition, the visualization by the ESOM feature map can be interpreted as height values on top of the usually two dimensional grid of the SOM, leading to an intuitive paradigm of a landscape. In summary, the visualization results obtained by the ESOM can help in recognizing and classifying consistent emotions in imbalanced datasets.



(a) (b)

(c) (d)

Figure 7-6 Visualisation of questionnaire emotion data (shame) generated by SVESOM under different numbers of features selected. (a) U-Map with 2 features selected; (b) P-Map with 2 features selected; (c) U-Map with 4 features selected; (d) P-Map with 4 features selected.

7.4 Conclusion

This study discusses the computational analysis of general emotion understanding from questionnaires methodology. The questionnaires method approaches a subject by dwelling on the real experience accompanied the emotions, whereas the other laboratory approaches are generally associated with exaggerated elements. We adopted a connectionist model called Support-Vector based Emergent Self Organizing Map (SVESOM) to analyze the emotion profiling from the questionnaires method. The SVESOM first identifies the important variables by giving discriminative features with high ranking. The classifier then performs the classification based on

the selected features. Experimental results showed that the top rank features are in-line with the work of Wallbott and Scherer published in 1994 which approached the emotions physiologically. Whilst the performance measures show that using the full features for classifications can degrade the performance, the selected features provide more convincing results in terms of accuracy and generalization.

Chapter 8 Conclusion and Future Direction

This chapter summarizes and concludes the research work that was carried out, the research issues as well as the achievements of this thesis. Finally, the possible solutions are suggested for further studies along with the research in the future.

8.1 Conclusion

We have presented a thesis that started with literature review on Self-Organizing Maps and Features Ranking. We then presented three models that have synergy effect when they are combined:

1. **Two-Tier Emergent Self-Organizing Map**

   A Two-Tier Emergent Self-Organizing Map that have properties like incremental learning, discovery capabilities and can adapt to the changing structures of data has been proposed. The model is capable of handling large data. The Two-Tier Emergent Self-Organizing Map

achieves improvements of accuracies over Self-Organizing Map. Most recognition problems are solved through supervised learning, whereas this work utilizes the topographic preserving, self-organizing, and emergent properties to resolve the recognition problem effectively.

2. Self-Organizing Cortical Visual Processing Model

This model attempts to model human brain's cognitive process at the primary visual cortex to comprehend the region of interest in an image. The model consists of pre-cortical processing and cortical processing. There are six types of retinal and six types of LGN cells which are modelled using different Gabor wavelets, where each Gabor wavelet is applied to a particular spectral band. The outputs of these LGN cells are a series of features that produce similar images as perceived by the visual pathway. The second stage (cortical processing) consists of two types of unit representing two broad categories of cells in the primate striate cortex. The adaptation process is achieved by the Two-Tier Emergent Self-Organizing Map. We demonstrated how this model can fit into simulations like road sign recognition and emotion recognition.

3. Prototype Ranking for Visual Feature Selection

The prototype rankings are derived from Support Vector Machines and are based on weight vector sensitivity with respect to a prototype. It was initially proposed by (Guyon et al., 2002a) for selecting genes that is relevant for a cancer classification problem. This is to complement the learning of classifiers to shrink down the number of prototypes that need to be processed and hence reducing the computation time and alleviates the skewness effect that is observed in imbalanced data. We have extended this into medical imbalanced dataset and have successfully pinpointed the important prototypes that help in diagnosing illness.

The above models have been applied successfully in pattern recognition data, imbalanced data and difficult emotion problem like questionnaire study. The results have demonstrated that the models

are suitable for the investigated problem domains and it is general enough to fit into wider range of problems that require a framework that is dynamic and intelligent with minimal supervision.

8.2 Future Direction

The research discussed thus far can serve as a starting point for a wide variety of future computational investigations. First, the self-organizing model architecture can be extended in several ways to match biological systems in more detail. Second, self-organizing model can be built to understand several new visual phenomena, both in V1 and in other visual areas. Third, self-organizing model can be used as a starting point for new research directions, including modelling other cortical areas and building artificial vision and other information processing systems.

More complex mechanisms can be included in the model. These extensions should be in tandem gets more similar to human visual processing. The extensions include the following.

1. **Neurons connections**

   Neurons in adult V1 retain their selectivity over wide variations in contrast; how strongly they respond depends primarily on how well the input matches their preferred features such as orientation, ocularity, and direction. In future work, this may be incorporated into the self-organizing model. Such an extension should allow self-organizing model to self-organize as before, but would represent the input more accurately, and also lead to reliable responses to a wider variety of input patterns.

2. **Scaling up to large networks**

   Developing techniques to simulate large networks accurately is a major issue in computational neuroscience research. Large maps are necessary for many other important visual phenomena, such as visual attention, saccades between stimulus features, the interaction between the

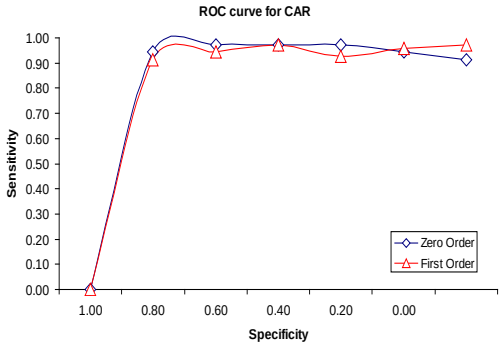
foveal and peripheral representations of the visual field, and the self-organization based on large scale patterns of optic flow due to head movement.

3. Feedback from higher level

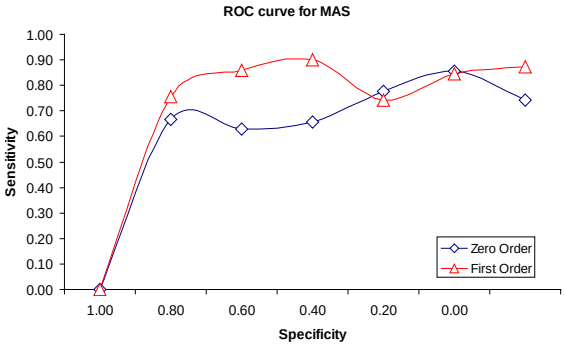
Current self-organizing model proposed here is feedforward only. Activation propagates from the first-tier to second-tier. In the cortex, a large proportion of connections propagate in the reverse direction, connecting from higher levels to V1 and the LGN. The role of these feedback connections is not yet clear, but they may be involved in top-down pattern completion, attention, visual imagery, and large-scale object grouping. During self-organization, the feedback connections may also encourage different areas to develop synergistically, mediating competition and cooperation between multiple areas.

Appendix

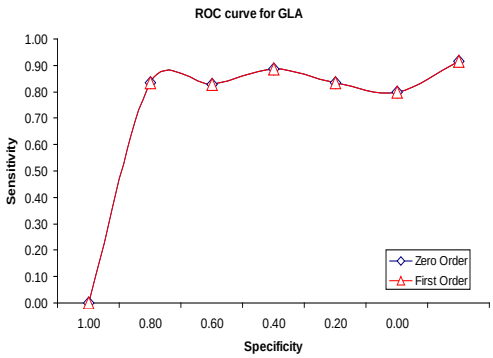
A1. Roc Analysis and Results on Imbalanced Data



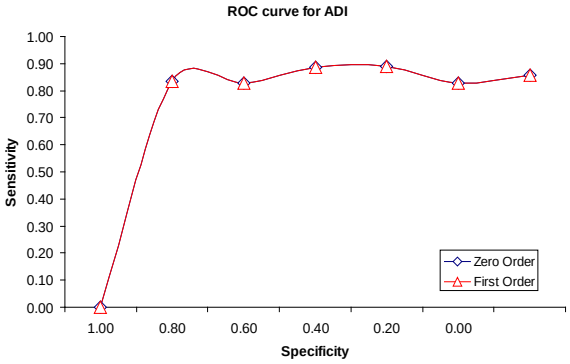
(a) CAR



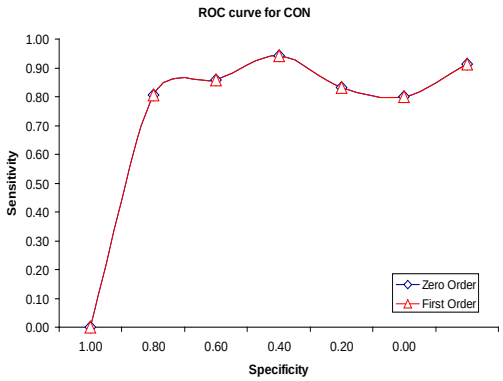
(b) MAS



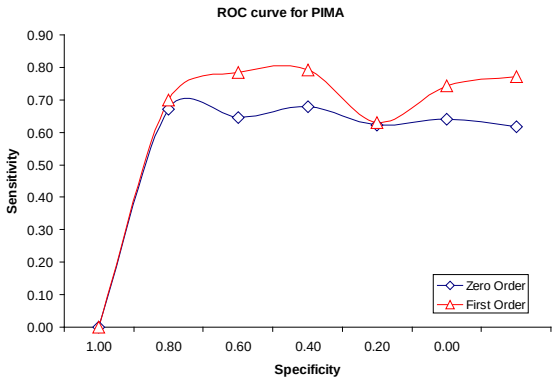
(c) GLA



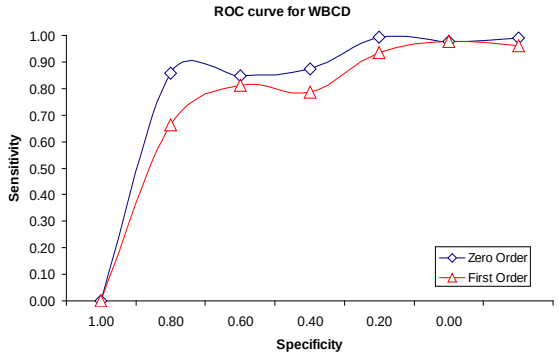
(d) ADI



(e) CON



(f) PIMA



(g) WBCD

Figure A1-1 ROC analysis between zero-order SVESOM and first-order SVESOM for different imbalanced datasets.

Table A1-1 Generalization results of zero-order SVESOM applied to different imbalanced datasets

Number of Features	2	3	4	5	6	7	8	9
CAR Results								
F-Measure	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96
GMM(training)	1.00	0.98	1.00	1.00	1.00	1.00	1.00	1.00
GMM(testing)	0.91	0.95	0.79	0.83	0.83	0.83	0.90	0.90
Generalization/Accuracy	0.92	0.94	0.92	0.94	0.94	0.92	0.92	0.97
MAS Results								
F-Measure	0.75	0.80	0.86	0.89	0.88	0.91	0.82	0.85
GMM(training)	0.91	1.00	1.00	1.00	1.00	0.98	0.97	0.97
GMM(testing)	0.35	0.46	0.67	0.73	0.69	0.70	0.66	0.69
Generalization/Accuracy	0.67	0.67	0.75	0.81	0.78	0.86	0.72	0.78
GLA Results								
F-Measure	0.91	0.91	0.88	0.91	0.91	0.90	0.90	0.89
GMM(training)	1.00	1.00	0.98	1.00	1.00	0.98	0.97	0.97
GMM(testing)	0.80	0.67	0.76	0.73	0.88	0.71	0.70	0.79
Generalization/Accuracy	0.83	0.83	0.81	0.83	0.83	0.83	0.86	0.83
ADI Results								
F-Measure	0.91	0.91	0.89	0.89	0.91	0.93	0.91	0.94
GMM(training)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
GMM(testing)	0.73	0.95	0.96	0.96	0.96	0.96	0.95	0.95
Generalization/Accuracy	0.83	0.83	0.81	0.81	0.83	0.86	0.83	0.89
CON Results								
F-Measure	0.89	0.91	0.89	0.91	0.94	0.93	0.94	0.91
GMM(training)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
GMM(testing)	0.73	0.87	0.87	0.87	0.87	0.93	0.94	0.81
Generalization/Accuracy	0.81	0.83	0.81	0.83	0.89	0.86	0.89	0.83
PIMA Results								
F-Measure	0.74	0.75	0.80	0.79	0.78	0.99	0.97	
GMM(training)	1.00	1.00	1.00	1.00	1.00	1.00	0.99	
GMM(testing)	0.67	0.78	0.69	0.77	0.78	0.76	0.69	
Generalization/Accuracy	0.59	0.60	0.67	0.66	0.63	0.98	0.95	
WBCD Results								
F-Measure	0.70	0.75	0.80	0.79	0.78	0.98	0.97	0.95
GMM(training)	0.57	0.74	0.70	0.76	0.74	0.74	0.71	0.93
GMM(testing)	0.90	0.98	0.98	1.00	1.00	0.99	1.00	1.00
Generalization/Accuracy	0.59	0.60	0.67	0.66	0.63	0.98	0.95	0.91

Table A1-2 Generalization results of first-order SVESOM applied to different imbalanced datasets

Number of Features	2	3	4	5	6	7	8	9
CAR Results								
F-Measure	0.86	0.95	0.96	0.95	0.87	0.97	0.96	0.96
GMM(training)	1.00	1.00	1.00	1.00	0.63	0.91	0.93	0.91
GMM(testing)	0.60	0.82	0.79	0.85	0.86	1.00	0.97	0.99
Generalization/Accuracy	0.76	0.90	0.91	0.90	0.91	1.00	0.96	0.99
MAS Results								
F-Measure	0.87	0.80	0.80	0.85	0.80	0.91	0.89	0.86
GMM(training)	0.75	0.37	0.37	0.53	0.00	1.00	0.98	1.00
GMM(testing)	0.97	0.97	0.94	0.99	0.61	0.72	0.72	0.64
Generalization/Accuracy	0.96	0.96	0.96	0.99	0.80	0.83	0.83	0.76
GLA Results								
F-Measure	0.91	0.91	0.88	0.91	0.91	0.90	0.90	0.89
GMM(training)	1.00	1.00	0.98	1.00	1.00	0.98	0.97	0.97
GMM(testing)	0.80	0.67	0.76	0.73	0.88	0.71	0.70	0.79
Generalization/Accuracy	0.83	0.83	0.81	0.83	0.83	0.83	0.86	0.83
ADI Results								
F-Measure	0.92	0.99	0.99	0.99	0.99	0.99	0.99	1.00
GMM(training)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
GMM(testing)	0.73	0.95	0.96	0.96	0.96	0.95	0.99	1.00
Generalization/Accuracy	0.86	0.97	0.99	0.99	0.99	0.97	0.99	1.00
CON Results								
F-Measure	0.95	0.97	0.97	0.98	0.98	0.99	0.97	0.98
GMM(training)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
GMM(testing)	0.73	0.87	0.87	0.87	0.87	0.88	0.75	0.82
Generalization/Accuracy	0.90	0.94	0.94	0.96	0.96	0.97	0.94	0.96
PIMA Results								
F-Measure	0.72	0.74	0.75	0.75	0.73	0.78	0.75	
GMM(training)	0.88	0.91	0.90	0.92	0.94	0.93	0.92	
GMM(testing)	0.61	0.63	0.65	0.64	0.59	0.67	0.62	
Generalization/Accuracy	0.63	0.63	0.66	0.64	0.61	0.67	0.64	
WBCD Results								
F-Measure	0.78	0.78	0.81	0.85	0.83	0.83	0.93	0.96
GMM(training)	0.93	0.95	0.96	0.97	0.97	0.96	0.97	1.00
GMM(testing)	0.67	0.64	0.68	0.73	0.71	0.71	0.89	0.92
Generalization/Accuracy	0.68	0.66	0.71	0.76	0.73	0.73	0.90	0.94

A2. Questionnaire Results and Visualizations

Table A2-1 Performance result of joy emotion by SVESOM

Features	Generalization	Sensitivity	F-Measure	GMts	GMtr
2	Cannot converge				
3					
4	0.96	0.97	0.96	0.94	0.98
5	0.96	1.00	0.98	0.91	1.00
6	0.94	0.99	0.96	0.87	0.99
7	0.95	0.99	0.97	0.87	0.99
8	0.96	0.99	0.98	0.93	1.00
All	0.95	1.00	0.97	0.91	1.00

Table A2-2 Performance result of fear emotion by SVESOM

Features	Generalization	Sensitivity	F-Measure	GMts	GMtr
2	Cannot converge				
3	0.65	0.75	0.70	0.54	0.84
4	0.63	0.80	0.70	0.44	0.88
5	0.65	0.86	0.74	0.47	0.91
6	0.70	0.84	0.77	0.49	0.90
7	0.67	0.86	0.75	0.36	0.91
8	0.63	0.84	0.72	0.45	0.90
All	0.81	0.96	0.88	0.53	0.84

Table A2-3 Performance result of anger emotion by SVESOM

Features	Generalization	Sensitivity	F-Measure	GMts	GMtr
2	Cannot converge				
3	0.64	0.72	0.68	0.50	0.82
4	0.78	0.94	0.85	0.54	0.96
5	0.71	0.93	0.81	0.29	0.96
6	0.78	0.96	0.86	0.45	0.98
7	0.78	0.94	0.85	0.52	0.96
8	0.82	0.93	0.87	0.60	0.96
All	0.71	0.92	0.8	0.43	0.94

Table A2-4 Performance result of sad emotion by SVESOM

Features	Generalization	Sensitivity	F-Measure	GMts	GMtr
2	0.47	0.50	0.48	0.56	0.64
3	0.56	0.66	0.61	0.55	0.77
4	0.69	0.90	0.78	0.38	0.94
5	0.72	0.93	0.81	0.51	0.96
6	0.68	0.90	0.77	0.45	0.94
7	0.66	0.86	0.75	0.46	0.92
8	0.72	0.92	0.81	0.37	0.94
All	0.86	0.88	0.87	0.49	0.41

Table A2-5 Performance result of disgust emotion by SVESOM

Features	Generalization	Sensitivity	F-Measure	GMts	GMtr
2	Cannot converge				
3	0.60	0.74	0.66	0.54	0.84
4	0.71	0.89	0.79	0.52	0.93
5	0.73	0.90	0.81	0.53	0.94
6	0.70	0.90	0.79	0.47	0.94
7	0.77	0.90	0.83	0.56	0.94
8	0.72	0.93	0.82	0.43	0.96
All	0.85	0.87	0.86	0	0.35

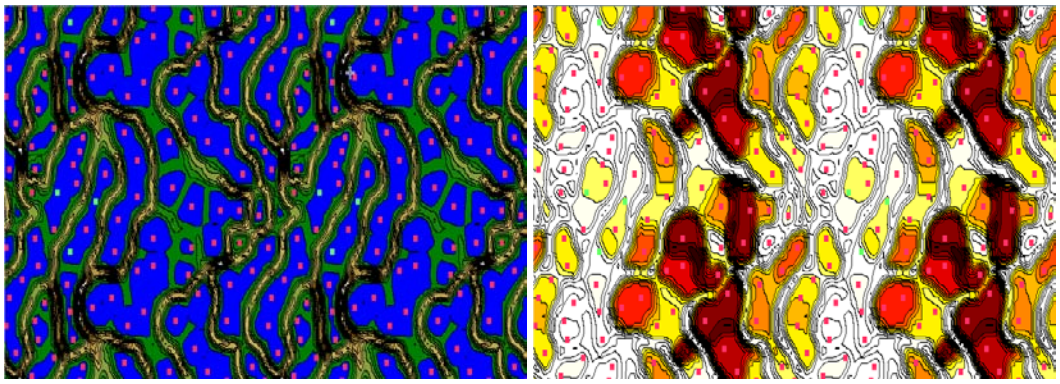
Table A2-6 Performance result of shame emotion by SVESOM

Features	Generalization	Sensitivity	F-Measure	GMts	GMtr
2	0.55	0.63	0.59	0.48	0.75
3	0.93	0.94	0.94	0.96	0.96
4	0.95	0.96	0.96	0.97	0.98
5	0.97	0.96	0.97	0.98	0.98
6	0.98	0.98	0.98	0.97	0.99
7	0.96	0.98	0.97	0.98	0.99
8	0.96	0.96	0.96	0.98	0.98
All	0.89	0.96	0.92	0.59	0.84

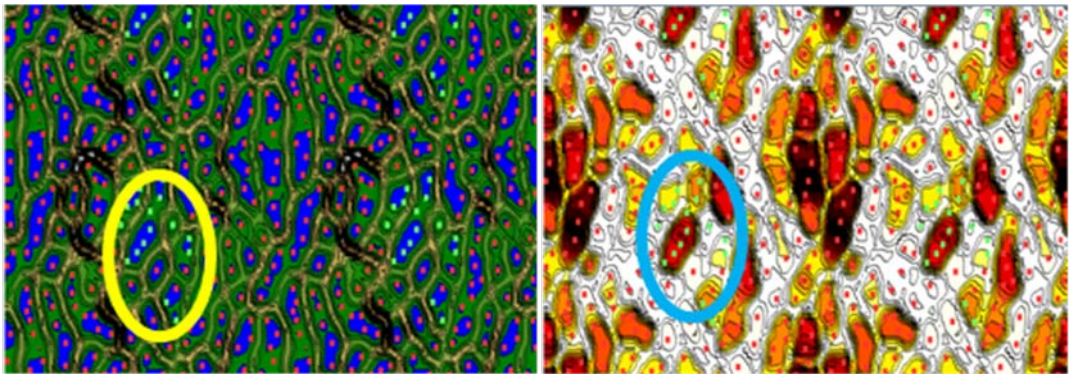
Table A2-7 Performance result of the guilt emotion by SVESOM

Features	Generalization	Sensitivity	F-Measure	GMts	GMtr
2	0.41	0.50	0.45	0.48	0.64
3	0.51	0.63	0.57	0.56	0.74
4	0.60	0.78	0.68	0.54	0.86
5	0.71	0.90	0.79	0.54	0.94
6	0.70	0.88	0.78	0.49	0.93
7	0.71	0.88	0.79	0.55	0.93
8	0.68	0.91	0.78	0.51	0.95
All	0.84	0.87	0.85	0.14	0.29

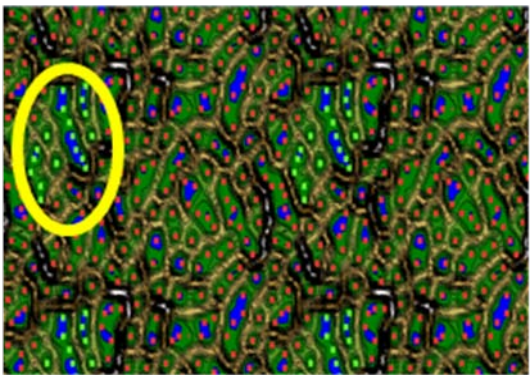
Visualisation of questionnaire emotion data generated by SVESOM under different numbers of features selected. (a) U-Map under 2 features selected; (b) P-Map under 2 features selected; (c) U-Map under 3 features selected; (d) P-Map under 3 features selected; (e) U-Map under 4 features selected; (f) P-Map under 4 features selected; (g) U-Map under 5 features selected; (h) P-Map under 5 features selected; (i) U-Map under 6 features selected; (j) P-Map under 7 features selected; (k) U-Map under full features selected; (l) P-Map under full features selected. (Note that the yellow circles represent the clusters of positive emotion at the U-Maps, the blue circles represent the clusters of positive emotion at the P-Maps)



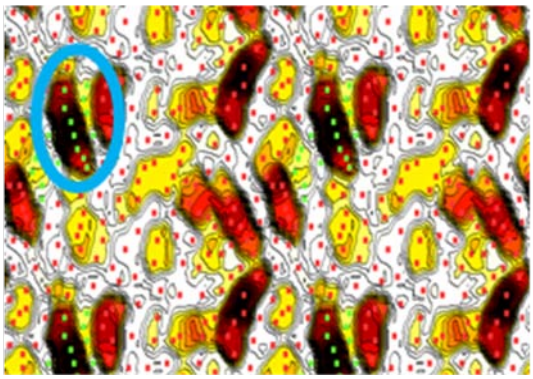
(a) (b)



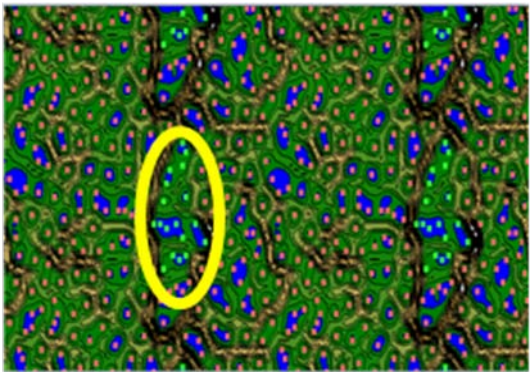
(c) (d)



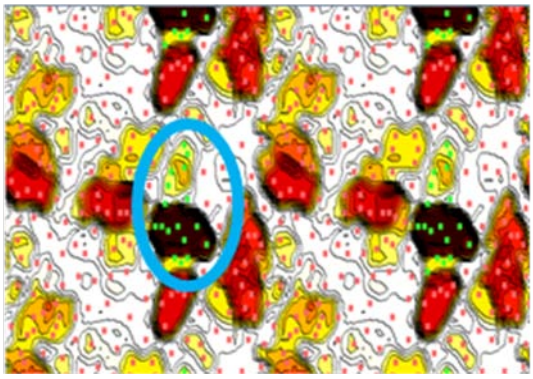
(e)



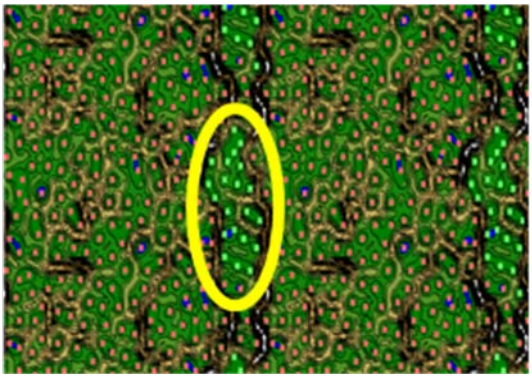
(f)



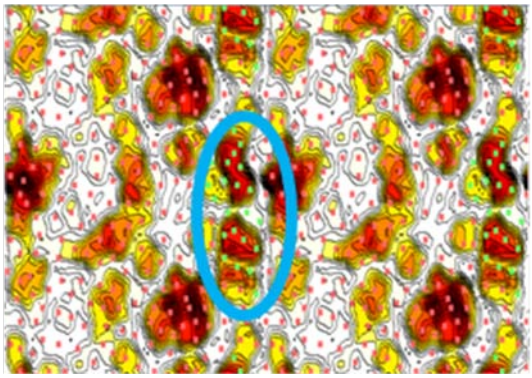
(g)



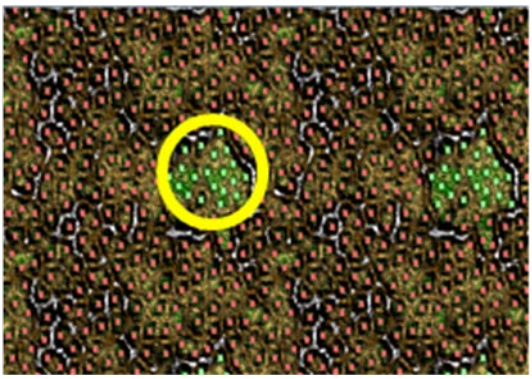
(h)



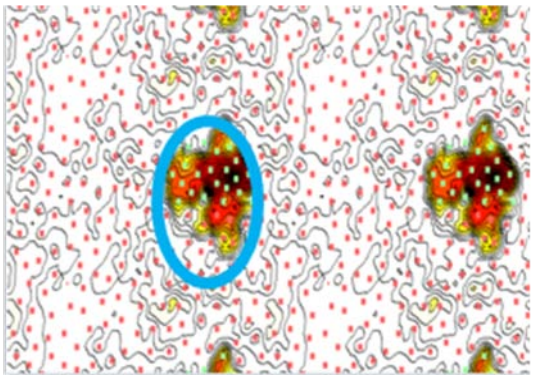
(i)



(j)



(k)



(l)

Publications

This research has in many parts been shaped by reviewers' comments and suggestions. During the course of this research, a number of publications have been made which are listed below:

Refereed journals (JCR impact factor @ 2009)

1. Yok-Yen Nguwi and Siu-Yeung Cho, "An Emergent Self-Organizing Learning with Support Vector Ranking for Imbalanced Datasets", *Expert Systems and Application* (ISI impact factor: 2.908), 37(12), pp. 8303-8312, 2010.
2. Yok-Yen Nguwi and Siu-Yeung Cho, "Emergent Self-Organizing Feature Map for Recognizing Road Sign Images", *Neural Computing and Applications* (ISI impact factor: 0.812), 19 (4), pp. 601-615, 2010.
3. Yok-Yen Nguwi and Siu-Yeung Cho, "Support Vector based Emergent Self-Organizing Approach for Emotional Understanding", *Connection Science* (ISI impact factor: 0.806), 22(4), pp.355-371, 2010.
4. Yok-Yen Nguwi, Teik-Toe Teoh and Siu-Yeung Cho, "Developing an Assistive Technology to Help Children with Autism for Recognising Human Emotion ", *HKIE Transactions*, 17(4), pp.61-68 2010 (shortlisted paper for The HKIE Outstanding Paper Award for Young Engineers/Researchers 2010).
5. Teik-Toe Teoh, Yok-Yen Nguwi and Siu-Yeung Cho, "Towards a Portable Intelligent Facial Expression Recognizer", *Intelligent Decision Technologies*, vol. 3:3, pp. 181-191, 2009. (featured by IOS press) (This project was featured by CNET and Science Daily on 19 Oct 2009)

6. Yok-Yen Nguwi and Abbas Kouzani, "Detection and Classification of Road Signs in Natural Environments", *Neural Computation and Applications* (ISI impact factor: 0.767), 17(3), 265-289, 2008.
7. Yok-Yen Nguwi, Teik-Toe Teoh, Insu Song, Siu-Yeung Cho, "Real-Time Road Sign Recognition", *Australian Journal of Intelligent Information Processing Systems*, 2010, in-press.
8. Yok-Yen Nguwi and Siu-Yeung Cho, "TtEsom: Two-tier Emergent self-organizing map for Cortical Visual Processing ", *International Journal of Neural Systems*, under review.
9. Yok-Yen Nguwi and Siu-Yeung Cho, "Prototype Ranking for Solving Imbalanced Conditions of Facial Emotion Recognition ", *International Journal of Innovative Computing, Information and Control*, rejected pending re-submission.

Invited books chapters

1. Siu-Yeung Cho, Teik-Toe Teoh and Yok-Yen Nguwi, "Development of an Intelligent Facial Expression Recognizer for Mobile Applications", Studies in Computational Intelligence: New Advances in Intelligent Decision Technologies, Springer Berlin / Heidelberg, pp. 21-29, 2009
2. Siu-Yeung Cho, Teik-Toe Teoh and Yok-Yen Nguwi, "Advanced Feature Selection and Classification methods", In Yu-Jin Zhang (eds.), Advances in Face Image Analysis: Techniques and Technologies: Tsing Hua University, IGI Global Publishing, pp. 239-258 2009.

Conference Proceedings (Refereed Papers)

1. Yok Yen Nguwi and Teik-Toe Teoh, “Two-tier Emergent Self-Organizing (TtEsom) Approach of Understanding Emotions”, International Conference on Control, Automation, Robotics and Vision 2010. Singapore. pp 654 - 8
2. Teik-Toe Teoh, Yok-Yen Nguwi, “Emotion Indexing using Hidden Markov Expert Rule Model (HMER) for autism children”, International Conference on Control, Automation, Robotics and Vision Dec 2010. Singapore. pp 668 - 72
3. Teik-Toe Teoh, Yok-Yen Nguwi and Siu-Yeung Cho, “Intelligent Face Locator for Smartphone”, in IEEE International Symposium on Industrial Electronics 2009, July 2009, Seoul, Korea. pp 1662 – 7.
4. Yok Yen Nguwi and Siu Yeung Cho, “Support Vector Self-Organizing Learning for Imbalanced Medical Data”, in 2009 International Joint Conference on Neural Networks, June 2009, Atlanta, Georgia, USA. pp 2250-5
5. Yok-Yen Nguwi and Abbas Kouzani, "A Study on Automatic Recognition of Road Signs. IEEE International Conference on Cybernetics and Intelligent Systems; June 2006; Bangkok, Thailand; pp 335-40.
6. Yok-Yen Nguwi and Abbas Kouzani, "Automatic Road Sign Recognition Using Neural Networks”, in IEEE World Congress on Computational Intelligence, July 2006; Vancouver, 2006. pp 7686-93.
7. Yok-Yen Nguwi and Siu-Yeung Cho, “Two-tier Self-Organizing Visual Model for Road Sign Recognition”, in IEEE World Congress on Computational Intelligence, June, 2008, Hong Kong, China. pp 794-9
8. Yok-Yen Nguwi and Siu-Yeung Cho, “Self-Organizing Adaptation for Facial Emotion Mapping. International Conference on Artificial Intelligence; June 2007; Las Vegas, USA; 2007. pp 132-7.

References

Ahumada, H., Grinblat, G. L., Uzal, L. C., Granitto, P. M., & Ceccatto, A. (2008). *REPMAC: A new hybrid approach to highly imbalanced classification problems.* Paper presented at the Proceedings - 8th International Conference on Hybrid Intelligent Systems, HIS 2008.

Akbani, R., Kwek, S., & Japkowicz, N. (2004). *Applying support vector machines to imbalanced datasets.* Paper presented at the Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science).

Amari, S. (1977). Dynamics of pattern formation in lateral inhibition type neural fields. *Biological Cybernetics, 27*(2), 77-87.

Anderberg, M. R. (1973). *Cluster analysis for applications: probability and mathematical statistics*: New York : Academic Press, .

Aoyagi, Y., & Asakura, T. (1996). *A study on traffic sign recognition in scene image using genetic algorithms and neural networks.* Paper presented at the Proc. of the 22nd Int. Conf. On Industrial Electronics, Control, and Instrumentation, Taipeis.

Atick, J. J., & Redlich., A. N. (1990). Mathematical model of the simple cells in the visual cortex. *Biological Cybernetics, 63*, 99-109.

Ayat, N. E., Cheriet, M., Remaki, L., & Suen, C. Y. (2001). *KMOD - a new support vector machine kernel with moderate decreasing for pattern recognition. Application to digit image recognition.* Paper presented at the Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on.

Bailing, Z., Minyue, F., Hong, Y., & Jabri, M. A. (1999). Handwritten digit recognition by adaptive-subspace self-organizing map (ASSOM). *Neural Networks, IEEE Transactions on, 10*(4), 939-945.

Baron-Cohen, S. (1995). *Mindblindness: An essay on autism and theory of mind*. Cambridge: MIT Press.

Barrow, H. G., Bray, A. J., & Budd, J. M. L. (1996). A Self-Organizing Model of "Color Blob" Formation. *Neural Computation, 8*(7), 1427-1448.

Bauer, H.-U., & Pawelzik, K. R. (1992). Quantifying the neighborhood preservation of self-organizing feature maps. *IEEE Transactions on Neural Networks, 3*(4), 570-579.

Bechara, A., Damasio, H., & Damasio, A. R. (2000). Emotion decision making and the orbitofrontal cortex. *Cerebral Cortex, 10*(3), 295–307.

Bednar, J. A., & Miikkulainen, R. (2003). Self-organization of spatiotemporal receptive fields and laterally connected direction and orientation maps. *Neurocomputing*(52-54), 473-480.

Benallal, M., & Meunier, J. (2003). *Real-Time Colour Segmentation of Road Signs.* Paper presented at the Proc. of the Canadian Conf. on Electrical and Computer Engineering.

Berkes, P., & Wiskott, L. (2002a). Applying Slow Feature Analysis to Image Sequences Yields a Rich Repertoire of Complex Cell Properties. *Artificial Neural Networks ICANN* 81-86.

Berkes, P., & Wiskott, L. (2002b). Applying Slow Feature Analysis to Image Sequences Yields a Rich Repertoire of Complex Cell Properties. *Artificial Neural Networks ICANN*, 81-86.

Bernhard E. Boser, I. G., and Vladimir Vapnik. A training algorithm for opti-Bradley, P. S. and O. L. Mangasarian (1998). Feature selection via concave minimization and support vector machines. Proceedings of International Conference on Machine Learning.

Berson, A., & Smith, S. J. (1997). *Data Warehousing, Data Mining, and Olap*: McGraw-Hill, Inc.

Blake, C. L., & Merz, C. J. (1998). UCI Repository of machine learning databases. Irvine, CA: University of California, Department of Information and Computer Science.

Blasdel, G. G., & Salama, G. (1986). Voltage-sensitive dyes reveal a modular organization in monkey striate cortex. *Nature, 321*(6070), 579-585.

Bosch, A. v. d., Weijters, T., Herik, H. J. v. d., & Daelemans, W. (1997). *When small disjuncts abound, try lazy learning:A case study.* Paper presented at the In Proceedings of the Seventh Belgian-Dutch Conference on Machine Learning.

Boser, B., Guyon, I., & Vapnik, V. (1992). *A training algorithm for optimal margin classifiers.* Paper presented at the COLT '92: Proceedings of the fifth annual workshop on Computational learning theory, Pittsburgh, Pennsylvania, United States.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). *A training algorithm for optimal margin classifiers.* Paper presented at the Proc. Fifth Annual Workshop on Computational Learning Theory.

Bradski, G. (1998). Computer Vision Face Tracking For Use in a Perceptual User Interface. from citeulike-article-id:3781379 http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.14.7673

Brunelli, R., & Poggio, T. (1993). Face Recognition: Features versus templates. *IEEE Trans. Pattern Anal. Machine Intell., 15*, 1042-1052.

Burges, C. (1998). A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery, 2*, 121-167.

C.D. Gilbert, T. N. W. (1989). Columnar speci city of intrinsic horizontal and corticocortical connections in cat visual cortex. *J. Neurosci., 9*, 2432–2442.

Casti, J. L. (1989). *Alternate Realities: Mathematical Models of Nature and Man* Wiley-Interscience

Chan, P., & Stolfo, S. (1998). *Toward scalable learning with nonuniform class and cost distributions: A case study in credit card fraud detection.* Paper presented at the

Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining,.

Chang, Y. W., & Lin, C.-J. (2008). *Feature Ranking Using Linear SVM,.* Paper presented at the JMLR: Workshop and Conference Proceedings.

Chawla, N. (2003). *C4.5 and Imbalanced Datasets: Investigating the effect of sampling method, probalistic estimate, and decision tree structure.* Paper presented at the Proceedings of the ICML'03 Workshop on Learning from Imbalanced Data Sets.

Chawla, N., Japkowicz, N., & Zhou, Z. H. (2009). *Data Mining When Classes are Imbalanced and Errors Have Costs*. Paper presented at the PAKDD'2009 Workshop.

Chawla, N. V., Japkowicz, N., & A. Kolcz, e. (2003). *Special Issue on Learning from Imbalanced Data Sets.* Paper presented at the Proceedings of the ICML'2003 Workshop on Learning from Imbalanced Data Sets.

Chen, Y.-W., & Lin, C.-J. (2006). Combining SVMs with Various Feature Selection Strategies. In I. Guyon, M. Nikravesh, S. Gunn & L. Zadeh (Eds.), *Feature Extraction* (Vol. 207, pp. 315-324): Springer Berlin Heidelberg.

Chen, Y., & Wang, J. Z. (2003). Support vector learning for fuzzy rule-based classification systems. *Fuzzy Systems, IEEE Transactions on, 11*(6), 716-728.

Chen, Y. W., & Lin, C. J. (2006). Combining SVMs with various feature selection strategies *Studies in Fuzziness and Soft Computing, 207*, 315-324.

Christopher L, H. Psychology 101 [Electronic Version]. *The virtual psychology classroom. 2001*, from http://allpsych.com/psychology101/emotion.html

Churchland, P. M., & Churchland, P. S. (1990, Jan 1990). Could Machines Think? *Scientific American,* 32-37.

Cieslak, D. A., & Chawla, N. V. (2008). *Start globally, optimize locally, predict globally: Improving performance on imbalanced data.* Paper presented at the Proceedings - IEEE International Conference on Data Mining, ICDM.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*(3), 273-297.

Cottrell, G. W., & Fleming, M. K. (1990). *Categorisation of Faces Using Unsupervised Feature Extraction.* Paper presented at the Int'l Conf. Neural Networks, San Diego.

Darwin, C. (1872). *The Expression of the Emotions in Man and Animals*: J. Murray, London.

Daugman, J. (1985). Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *J. Opt. Soc. Amer., 2*(7), 1160-1161 1169.

Daugman, J. G. (1985). Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America, 2*, 1160-1169.

David, H. H., & Torsten, N. W. (1974). Sequence regularity and geometry of orientation columns in the monkey striate cortex. *The Journal of Comparative Neurology, 158*(3), 267-293.

de Sá, J., Alexandre, L., Duch, W., Mandic, D., Lira, M., de Aquino, R., et al. (2007). Boosting Algorithm to Improve a Voltage Waveform Classifier Based on Artificial Neural Network. In *Artificial Neural Networks – ICANN 2007* (Vol. 4669, pp. 455-464): Springer Berlin / Heidelberg.

DeAngelis, G. C., Ohzawa, I., & Freeman, R. D. (1995). Receptive- eld dynamics in the central visual pathways. *Trends Neurosci., 18*, 451-458.

Domingos., P. (1999). *Metacost: A general method for making classifiers cost-sensitive.* Paper presented at the In Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego,CA.

Drummond, C., & Holte, R. C. (2000). *Explicitly representing expected cost: An alternative to ROC representation.* Paper presented at the Proceeding of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

Duan, K. B., Rajapakse, J. C., Wang, H., & Azuaje, F. (2005). Multiple SVM-RFE for gene selection in cancer classification with expression data. *IEEE Transactions on Nanobioscience, 4*(3), 228-233.

Ekman, P., & Friesen, W. V. (1971). Constants across cultures in the face and emotion. *J. Personality Social Psychol, 17*(2), 124-129.

Elkan., C. (2003). Invited talk: The real challenges in data mining: A contrarian view. . from http://www.site.uottawa.ca/~nat/Workshop2003/realchallenges2.ppt

Erwin, E., Obermayer, K., & Schulten, K. (1992). Self-organizing maps: ordering, convergence properties and energy functions. *Biological Cybernetics, 67*(1), 47-55.

Escalera, A. d. l., & Radeva, P. (2004). Fast greyscale road sign model matching and recognition. *Recent Advances in Artificial Intelligence Research and Development*, 69–76.

Essa, I. A., & Pentland, A. P. (1997). Coding, analysis, interpretation, and recognition of facial expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 19*(7), 757-763.

Evgeniou, T., Pontil, M., Papageorgiou, C., & Poggio, T. (2003). Image Representations and Feature Selection for Multimedia Database Search. *IEEE Trans. on Knowl. and Data Eng., 15*(4), 911-920.

Fairhall, S. L., & Ishai, A. (2007). Effective connectivity within the distributed cortical network for face perception. *Cerebral Cortex, 17*(10), 2400-2406.

Farkas, I., & Miikkulainen, R. (1999). *Modeling the self-organization of directional selectivity in the primary visual cortex.* Paper presented at the Proceedings of ICANN, Berlin.

Farrar, D., & Glauber, R. (1967). Multicollinearity in Regression Analysis: The Problem Revisited. *The Review of Economics and Statistics, 49*(1), 92-107.

Forman., G. (2003). An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research, 3*, 1289-1305.

G.E. La Cara, M. U., M. Bettini (2003). *Extraction of Salient Contours in Primary Visual Cortex: A Neural Network Model Based on Physiological Knowledge*. Paper presented at the Engineering in Medicine and Biology Society.

Gao, X., Shevtsova, N., Hong, K., Batty, S., Podladchikova, L., Golovan, A., et al. (2002). *Vision models based identification of traffic signs.* Paper presented at the Proc. of the 1st Europ. Conf. on Color in Graphics, Image and Vision, France.

Gavrila, D. M. (1999). *Traffic Sign Recognition Revisited.* Paper presented at the Proceedings of the 21st DAGM Symposium für Mustererkennung, Bonn, Germany.

Gavrila, D. M., & Philomin, V. (1999). *Real-time object detection for smart vehicles.* Paper presented at the Proc. of IEEE Int. Conf. on Computer Vision, Greece.

Girolami, M., & Fyfe, C. (1997). *Kurtosis extrema and identification of independent components: A neural network approach*.

Goodale, & Milner. (1992). Separate pathways for perception and action. *Trends in Neuroscience 15*, 20-25.

Guo-ping, L., Li-xiu, Y., & Jie, Y. (2005). Solving the Problem of Imbalanced Dataset in the Pre-diction of Membrane Protein Types Based on Weighted SVM. *Journal of Shanghai Jiaotong University*, 1676-1684.

Guo, H., & Viktor, H. L. (2004). Learning from imbalanced data sets with boosting and data generation: the DataBoost-IM approach. *SIGKDD Explor. Newsl., 6*(1), 30-39.

Guyon, I., Aliferis, C., Cooper, G., Pellet, A. E. J.-P., Spirtes, P., & Statnikov, A. (2008). *Design and analysis of the causation and prediction challenge*. Paper presented at the JMLR: Workshop and Conference Proceedings.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002a). Gene Selection for Cancer Classification using Support Vector Machines. *Machine Learning, 46*(1), 389-422.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002b). Gene Selection for Cancer Classification using Support Vector Machines *Machine Learning, 46*(1-3), 389-422.

H. Shouno, K. K. (2001). Formation of a direction map by projection learning using Kohonen's self-organization map. *Biol. Cybernetics, 85*(4), 241–246.

Hagenbuchner, M., Sperduti, A., & Ah Chung, T. (2003). A self-organizing map for adaptive processing of structured data. *Neural Networks, IEEE Transactions on, 14*(3), 491-505.

Han, & Yang. (2004). Screening important design variables for building a usability model: genetic algorithm-based partial least-squares approach. *International Journal of Industrial Ergonomics, 33*(2), 159-171.

Han, S. H., & Kim, J. (2003). A comparison of screening methods: selecting important design variables for modeling product usability. *International Journal of Industrial Ergonomics*(32), 189-198.

Han, S. H., Kim, K. J., Yun, M. H., Hong, S. W., & Kim, J. (2004). Identifying mobile phone design features critical to user satisfaction. *Hum. Factor. Ergon. Manuf., 14*(1), 15-29.

Han, S. H., & Yang, H. (2004). Screening important design variables for building a usability model. *International Journal of Industrial Ergonomics, 33*, 159-171.

Hawken, M. J., & Parker, A. J. (1991). Spatial receptive field organization in monkey V1 and its relationship to the cone mosaic. In M. S. Landy & J. A. Movshon (Eds.), *Computational Models of Visual Processing* (pp. 83-93): MIT.

Hecht-Nielsen, R. (1988). Applications of counterpropagation networks. *Neural Networks, 1*(2), 131-139.

Hermes, L., & Buhmann, J. M. (2000). *Feature selection for support vector machines.* Paper presented at the Pattern Recognition, 2000. Proceedings. 15th International Conference on.

Hindley, J. R. (2008). *Basic Simple Type Theory*: Cambridge University Press.

Hoffman, P., Grinstein, G., Marx, K., Grosse, I., & Stanley, E. (1997). *DNA visual and analytic data mining.* Paper presented at the Visualization '97., Proceedings.

Hsu, C.-W., & Lin, C.-J. (2002). A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on, 13*(2), 415-425.

Hua, S., & Sun, Z. (2001). A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach. *Journal of Molecular Biology, 308*(2), 397-407.

Hubel, D., & Wiesel, T. (1962a). Receptive fields, binocular interaction, and functional architecture in the cat's visual cortex. *J. Physiol, 160*, 106-154.

Hubel, D., & Wiesel, T. (1962b). Receptive fields, binocular interaction, and functional architecture in the cat's visual cortex. *J. Physiol., 160*, 106-154.

Informatique, C., Dépôts, C., Gérard, D., Rémi, D., Yacine, O., Isabelle, G., et al. (2008). Ranking a Random Feature for Variable and Feature Selection.

Isen, A. M. (2000). *Positive affect and decision making*. New York: Guilford Press.

Jabbi, M., Swart, M., & Keysers, C. (2007). Empathy for positive and negative emotions in the gustatory cortex. *NeuroImage, 34*(4), 1744-1753.

Japkowicz. (2000). *Proceedings of the AAAI'2000 Workshop on Learning from Imbalanced Data Sets* o. Document Number)

Ji, Y. Z. Q. (2005). Active and dynamic information fusion for facial expression understanding from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 27*(5), 699-714.

Jones, J. P., & Palmer, L. A. (1987). An evaluation of the Two-Dimensional Gabor Filter model of simple Receptive fields in cat striate cortex. *J.Neurophysiol, 58* (6), 1233-1258.

Jong, K., Mary, J., Cornuejols, A., Marchiori, E., & Sebag, M. (2004). *Ensemble feature ranking*. Paper presented at the Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases.

Juszczak, P., & Duin, R. P. W. (2003). *Uncertainty sampling methods for one-class classifiers.* Paper presented at the Proceedings of the ICML'03 Workshop on Learning from Imbalanced Data Sets.

K. Ali, & Pazzani, M. (1995). HYDRA-MM: learning multiple descriptions to improve classification accuracy. *International Journal of Artificial Intelligence Tools, 4*.

Kaas, J. H., Merzenich, M. M., & Killackey, H. P. (1983). The reorganization of somatosensory cortex following peripheral nerve damage in adult and developing mammals. *Annual review of Neurosciences, 6*, 325-356.

Kanade, T. (1973). Picture processing by computer complex and recognition of human faces. Kyoto University.

Kanade, T., Cohn, J. F., & Tian, Y. (2000). *Comprehensive database for facial expression analysis.* Paper presented at the Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on.

Kanndel, E. R., Martinetz, T., & Schulten, K. (2000). *Principles of Neural Science* (4th ed.). New York: McGraw Hill.

Kiviluoto, K. (1996). *Topology preservation in Self-Organizing Maps.* Paper presented at the International Conference on Neural Networks, Piscataway.

Knudsen, E. I., du Lac, S., & Esterly, S. D. (1987). Computational maps in the brain. *Annu Rev Neurosci, 10*, 41-65.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics, 43*, 59-69.

Kohonen, T. (1989). *Self-organization and associative memory* (3rd ed.). Berlin: Springer-Verlag.

Kohonen, T. (1990, sept 1990). *Self-organizing Map.* Paper presented at the Proc. Of IEEE.

Kohonen, T. (1995). *Self-Organizing Maps*. Berlin, Heidelberg: Springer.

Kohonen, T. (1996). Emergence of invariant-feature detectors in the adaptive-subspace self-organizing map. *Biological Cybernetics, 75*(4), 281-291.

Kohonen, T., Kaski, S., & Lappalainen, H. (1997). Self-Organized Formation of Various Invariant-Feature Filters in the Adaptive-Subspace SOM. *Neural Computation, 9*(6), 1321-1344.

Krebel, U. (1999). Pairwise classification and support vector machines. In B. Scholkopf, J. C. Burges & A. J. Smola (Eds.), *Advances in Kernel Methods - Support Vector Learning* (pp. 255-268). Cambridge, MA: MIT Press.

Kubat, M., Holte, R., & Matwin, S. (1998). Machine learning for the detection of oil spills in satellite radar images. *Machine Learning, 30*(2-3), 195-215.

Kurgan, L., Cios, K., Tadeusiewicz, R., Ogiela, M., & Goodenday, L. (2001). Knowledge discovery approach to automated cardiac SPECT diagnosis. *Artificial Intelligence in Medicine, 23*(2), 149-169.

Lauziere, Y. B., Gingras, D., & Ferrie, F. P. (2001). A Model-Based Road Sign Identification System. *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, 1*, 1163 –1170.

Lin, C.-F., & Wang, S.-D. (2002). Fuzzy support vector machines. *IEEE Transactions on Neural Networks, 13*(2), 464-471.

Lin, S. (1997). *The Self-Organizing Feature Map: Learning Characteristics, Applications, and Dynamic Topology Representing Networks.*

Liu, C., & Wechsler, H. (2002). Gabor Feature Based Classification using the Enhanced Fisher Linear Discriminant Model for Face Recognition. *IEEE Trans. on Image Processing, 11*(4), 467-472.

Liu, Y., & Zheng, Y. F. (2006). FS_SFS: A novel feature selection method for support vector machines. *Pattern Recogn., 39*(7), 1333-1345.

Lo, Z. P., & Bavarian, B. (1991). On the rate of convergence in topology preserving neural networks. *Biological Cybernetics, 65*(1), 55-63.

Lyons, M. J., Akamatsu, S., Kamachi, M., & Gyoba, J. (1998, April 14-16 1998). *Coding Facial Expressions with Gabor Wavelets.* Paper presented at the Third IEEE International Conference on Automatic Face and Gesture Recognition, Nara Japan.

MacQueen. (1967). *Methods for Classification and Analysis of Multivariate Observations* (5th Edition ed.). N.Y: Neyman Publications.

Maloof, M. (2003). *Learning when data sets are imbalanced and when costs are unequal and unknown.* Paper presented at the Proceedings of the ICML'03 Workshop on Learning from Imbalanced Data Sets.

Mao, K. Z. (2004). Feature subset selection for support vector machines through discriminative function pruning analysis. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 34*(1), 60-67.

Mao, Y., Zhou, X., Pi, D., Sun, Y., & Wong, S. T. C. (2005). Multiclass cancer classification by using fuzzy support vector machine and binary decision tree with gene selection. *Journal of Biomedicine and Biotechnology, 2005*(2), 160-171.

Marshall, J. A., & Alley, R. (1996). A self-organizing neural network that learns to detect and represent visual depth from occlusion events. In J. Sirosh, R. Miikkulainen & Y. Choe (Eds.), *Lateral Interactions in the Cortex: Structure and Function*. Austin, TX: The UTCS Neural Networks Research Group.

Martinetz, T. (1993). *Competitive Hebbian learning rule forms perfectly topology preserving maps.* Paper presented at the Proceedings of the International Conference on Artificial Neural Networks

Martinetz, T., & Schulten, K. (1994). Topology representing networks. *Neural Networks, 7*(3), 507-522.

Martinetz, T. M., Berkovich, S. G., & Schulten, K. J. (1993). Neural-gas' network for vector quantization and its application to time-series prediction. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council, 4*(4), 558-569.

Mase, K., & Pentland, A. (1991). Recognition of facial expression from optical flow. *IEICE Trans., 74*(10), 3474-3483.

Matsuno, K., Iee, C.-W., & Tsuji, S. (1994). *Recognition of Human Facial Expressions Without Feature Extraction.* Paper presented at the ECCV.

Miikkulainen, R., & Sirosh, J. (1996). *Lateral Interactions in the Cortex: Structure and Function*. Austin, TX: The UTCS Neural Networks Research Group.

Nakayama, K., & Shimojo, S. (1992). Experiencing and perceiving visual surfaces. *Science, 257*(5075), 1357-1363.

Ning, Z., Andrzej, S., & Setsuo, O. (1999 November 9-11). *New Directions in Rough Sets, Data Mining, and Granular Soft Computing.* Paper presented at the 7th International Workshop, RSFDGrC'99, Yamaguchi, Japan.

Nowlan. (1990). *Advances in Neural Information Processing Systems* (2nd Edition ed.). N.Y.: Morgan Kauffman Publishers.

Paclik, P., & Novovicova, J. (2000). *Road Sign Classification without Colour Information.* Paper presented at the Proc. of 6th Conf. of Advaced School of Imaging and Computing ASCI, Lommel, Belgium.

Paclik, P., Novovicova, J., Pudil, P., & Somol, P. (2000). Road sign classification using the Laplace kernel classifier. *Pattern Recognition Letters, 21*(13-14), 1165-1173.

Park, J., & Han, S. H. (2004). A fuzzy rule-based approach to modeling affective user satisfaction towards office chair design. *International Journal of Industrial Ergonomics, 34*(1), 31-47.

Pepe, M. S., Longton, G., Anderson, G. L., & Schummer, M. (2003). Selecting differentially expressed genes from microarray experiments. *Biometrics, 59*, 133-142.

Polat, U., M.Norcia, A., & Sagi, D. (1996). The pattern and functional significance of long-range interactions in human visual cortex. In R. M. Joseph Sirosh, and Yoonsuck Choe (Ed.), *Lateral Interactions in the Cortex: Structure and Function*.

Provost, F., & Fawcett, T. (2001). Robust classification for imprecise environments. *Machine Learning, 42*, 203-231.

Rakotomamonjy, A. (2003). Variable selection using SVM-based criteria. *Journal of Machine Learning Research, 3*, 1357-1370.

Rakotomamonjy, A. (2003a). Variable selection using svm based criteria. *The Journal of Machine Learning Research*, 1357-1370.

Rakotomamonjy, A. (2003b). Variable selection using SVM based criteria. *J. Mach. Learning*, 1357–1370.

Raskutti, B., & Kowalczyk, A. (2003). *Extreme re-balancing for SVM's: a case study.* Paper presented at the Proceedings of the ICML'03 Workshop on Learning from Imbalanced Data Sets.

Rauber, A., Merkl, D., & Dittenbach, M. (2002). The growing hierarchical self-organizing map: exploratory analysis of high-dimensional data. *Neural Networks, IEEE Transactions on, 13*(6), 1331-1341.

Roberts, F. S. (1976). *Discrete mathematical models with applications to social, biological, and environmental problems*: Prentice-Hall (Englewood Cliffs, N.J)

Rosenblum, M., Yacoob, Y., & Davis, L. S. (1996). Human Expression Recognition from Motion Using a Radial Basis Function Network Architecture. *IEEE Transactions on Neural Networks, 7*(5), 1121-1138.

Russell, J. A., & Fernández Dols, J. M. (1997). *The psychology of facial expression*. Cambridge ; New York: Cambridge University Press.

S.Vitabile, G.Pollaccia, G.Pilato, & F.Sorbello. (2001). *Road signs recognition using a dynamic pixel aggregation technique in the HSV color space.* Paper presented at the Proc. of the 11 th Int. Conf. on Image Analysis and Processing, Palermo, Italy.

Sabatini, S. P. (1996). Recurrent inhibition and clustered connectivity as a basis for Gabor-like receptive fields in the visual cortex. In R. M. Joseph Sirosh, and Yoonsuck Choe (Ed.), *Lateral Interactions in the Cortex: Structure and Function*. Austin, TX: The UTCS Neural Networks Research Group.

Sammon, J. W. (1969). A Nonlinear Mapping for Data Structure Analysis. *IEEE Trans. Comput., 18*(5), 401-409.

Scherer, K. R., & Wallbott, H. G. (1994). Evidence for universality and cultural variation of differential emotion response patterning. *Journal of Personality and Social Psychology, 66*, 310-328.

Shapiro, L. G., & Stockman, G. C. (2001). *Computer Vision*: Prentice Hall.

Shieh, M. D., & Yang, C. C. (2008). Multiclass SVM-RFE for product form feature selection. *Expert Systems with Applications, 35*(1-2), 531-541.

Shimizu, Y., & Jindo, T. (1995). A fuzzy logic analysis method for evaluating human sensitivities. *International Journal of Industrial Ergonomics, 15*(39-47).

Silva, J., Sá, J. M. d., & Jossinet, J. (2000). Classification of Breast Tissue by Electrical Impedance Spectroscopy. *Med & Bio Eng & Computing, 38*, 26-30.

Sirosh, J., Miikkulainen, R., & Bednar., J. A. (1996). Self-organization of orientation maps, lateral connections, and dynamic receptive fields in the primary visual cortex. . In R. M. J. Sirosh, and Y. Choe (Ed.), *Lateral Interactions in the Cortex: Structure and Function*. Austin, TX: The UTCS Neural Networks Research Group.

Smith, L. (2002). A Tutorial on Principal Component Analysis. from http://kybele.psych.cornell.edu/~edelman/Psych-465-Spring-2003/PCA-tutorial.pdf

Soetedjo, A., & Yamada, K. (2005). Traffic sign classification using ring partitioned method. *IEICE Trans. Fundamentals, E88A*(9), 166–178.

Somers, D., Toth, L. J., Todorov, E., Rao, S. C., Kim, D.-S., Nelson, S. B., et al. (1996). Variable gain control in local cortical circuitry supports context-dependent modulation by long-range connections. In J. Sirosh, R. Miikkulainen & Y. Choe (Eds.), *Lateral Interactions in the Cortex: Structure and Function*. Austin, TX: The UTCS Neural Networks Research Group.

Suga, N. (1985). The extent to which bisonar information is represented in the bat auditory cortex. *Dynamic Aspects of Neocortical Function*, 653-695.

T. Xiang, M. K. H. L. a. S. Y. C. (2007). Expression recognition using fuzzy spatio-temporal modeling. *Pattern Recognition, 41*(1), 204-216.

Technology, C. o. I. Customer foreign exchange data for currency risk management. from http://www.sis.uncc.edu/~mirsad/itcs6265/resource.htm

The_face_research_group. CMU Image Data Base [Electronic Version], from http://vasc.ri.cmu.edu/idb/html/face/

Thomas, V., Ralf, D., Herrmann, M., & Thomas, M. (1994). *Topology Preservation in Self-Organizing Feature Maps: General Definition and Efficient Measurement*. Paper presented at the Proceedings of Fuzzy Logik, Theorie und Praxis, 4. Dortmunder Fuzzy-Tage.

Torresen, J., Bakke, J., & Sekanina, L. (2004). *Efficient Recognition of Speed Limit Signs.* Paper presented at the Proc. of the 7th Int. IEEE Conf. on Intelligent Transportation Systems.

Trujillo-Ortiz, A., & Hernandez-Walls., R. (2003). Mskekur: Mardia's multivariate skewness and kurtosis coefficients and its hypotheses testing. A MATLAB file. from

http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=3519
Ultsch, A. (2005). *Clustering with SOM: U*C*. Paper presented at the WSOM.
Ultsch, A., & Herrmann, L. (2005). The architecture of emergent self-organizing maps to reduce projection errors. *ESANN* 1-6.
Ultsch, A., & Mörchen, F. (2005). *ESOM-Maps: tools for clustering, visualization, and classification with Emergent SOM*. Germany: University of Marburg
Dept. of Mathematics and Computer Scienceo. Document Number)
Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. New York: Springer.
Vapnik, V., & Lerner, A. (1963). Pattern Recognition using Generalized Portrait Method. *Automation and Remote Control, 24*.
Vapnik, V. N. (Ed.). (1998). *Statistical Learning Theory*. New York: John Wiley & Sons.
Vikramaditya, J. (2006). Tutorial on Support Vector Machine (SVM). from http://eecs.wsu.edu/~vjakkula/SVMTutorial.doc
Villmann, T., Der, R., Herrmann, M., & Martinetz, T. M. (1997). Topology preservation in self-organizing feature maps: exact definition and measurement. *Neural Networks, IEEE Transactions on, 8*(2), 256-266.
Villmann, T., Der, R., & Martinetz, T. (1994). *A new quantitative measure of topology preservation in Kohonen's feature maps.* Paper presented at the International Conference on Neural Networks, Piscataway NJ.
Vitabile, S., Gentile, A., & Sorbello, F. (2002). *A neural network based automatic road signs recognizer.* Paper presented at the Proc. of the 2002 International Joint Conference on Neural Networks.
Von Der Malsburg, C. (1973). Self organization of orientation sensitive cells in the striate cortex. *KYBERNETIK, 14*(2), 85-100.
Wang, W., Xu, Z., Lu, W., & Zhang, X. (2003). Determination of the spread parameter in the Gaussian kernel for classification and regression. *Neurocomputing, 55*(3-4), 643-663.
Wang, X., & Zhong, Y. (2003). *Statistical learning theory and state of the art in SVM.* Paper presented at the Cognitive Informatics, 2003. Proceedings. The Second IEEE International Conference on.
Weiss, G. M. (2004). Mining with rarity: A unifying framework. *SIGKDD Explorations, 6*(1), 7–19.
Wikipedia. from http://en.wikipedia.org/wiki/Eigenvalue
Willshaw, D. J., & Von Der Malsburg, C. (1976). How patterned neural connections can be set up by self organization. *Proceedings of the Royal Society of London - Biological Sciences, 194*(1117), 431-445.
Wilson, H. R., & Cowan, J. D. (1972). Excitatory and inhibitory interactions in localized populations of model neurons. *Biophysical Journal, 12*(1), 1-24.
Wilson, H. R., & Cowan, J. D. (1973). A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue. *KYBERNETIK, 13*(2), 55-80.

Wimbauer, S., Wenisch, O. G., Hemmen, J. L. v., & Miller, K. D. (1997). Development of spatiotemporal receptive felds of simple cells: II. *Biol. Cybernetics, 77*(6), 463–477.

Winters-Hilt, S., & Merat, S. (2007). SVM clustering. *BMC Bioinformatics 8*(18).

Wiskott, L., & Malsburg, C. v. d. (1996). Face recognition by dynamic link matching. In R. M. J. Sirosh, and Y. Choe (Ed.), *Lateral Interactions in the Cortex: Structure and Function*. Austin, TX: The UTCS Neural Networks Research Group.

Wittling, W., & Roschmann, R. (1991). Topographic brain mapping of emotion-related hemisphere asymmetries. *International Journal of Psychophysiology, 11*(1), 89-89.

Yan, L., Dodier, R. H., Mozer, M., & Wolniewicz, R. H. (2003). *Optimizing classifier performance via an approximation to the Wilcoxon-Mann-Whitney statistic.* Paper presented at the Proc. of ICML.

Yin, H. (2002). ViSOM: A novel method for multivariate data projection and structure visualization. *IEEE Trans. on Neural Networks, 13*(1), 237-243.