F3MCNN: FUZZY MINIMUM MEAN MAXIMUM

CLUSTERING NEURAL NETWORK

By

LIANG-TSAN WU

Bachelor of Science

National Tsing Hua University

Hsinchu, Taiwan, R. O. C.

1982

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
July, 1995

OKLAHOMA STATE UNIVERSITY

F3MCNN: FUZZY MINIMUM MEAN MAXIMUM

CLUSTERING NEURAL NETWORK

Thesis Approved:

Huizhu Lu
Thesis Adviser

D. E. Hedrick

J Blomey

Thomas C. Collins
Dean of the Graduate College

PREFACE

In the real world, pattern clustering problems deal with perturbed and imperfect data. A machine pattern clustering system is expected to have real-time and unsupervised learning ability and be able to represent and manipulate inexact information just like human beings. Pattern clustering techniques, based on statistic theory, artificial neural network theory, and fuzzy set theory, have been developed in the last few decades. A group of artificial neural networks exhibits remarkable properties of self-organization that benefit techniques of pattern clustering. The advent of fuzzy set theory has had a positive impact on techniques of pattern clustering. We are concerned in this dissertation with pattern clustering systems having the adaptive learning ability from clustering neural networks and the ability of representing and manipulating imprecise data from fuzzy sets.

In this study, we develop a new pattern clustering model called the fuzzy minimum mean maximum clustering neural network (F3MCNN) that is a synergetic combination of a neural network with the fuzzy set theory. Its system architecture and clustering algorithm are presented. A famous data set is used in our experiments to compare clustering results accuracy and stability with other similar models. From the experimental results, the F3MCNN model is superior to the other models.

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere appreciation to my Ph.D. dissertation advisor, Dr. H. Lu, for her consistent and continuous advice, guidance, kindness, and valuable instruction through my graduate work.

Special thanks are due to my advisory committee members, Dr. K. M. George, Dr. G. E. Hedrick, and Dr. J. B. Conrey, for their encouragement, advice, support, and critical review of my research. I would like also to thank Computer Science Department at Oklahoma State University for providing financial support to me during my graduate studies.

I would like to take this opportunity to thank my parents, Tong-Kuei Wu and Su-Kuan Lee, who have provided consistent support and endless love in my whole life. Also, I would like to thank my sister and brother-in-law, Muh-Shien Wu and Wai Hung Wan, and my brother, Liang-Shian Wu, for their encouragement. Finally and most importantly, my deepest love and thanks go to my family, my dear wife, Yin-Sheng Amanda Yang, for her patience and moral support that gave me the motivation and inspiration to complete my graduate studies, and my lovely daughter, Rachel Shander Wu, who has brought joy and happiness into my family.

TABLE OF CONTENTS

Chapter Page

Chapter Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

NOMENCLATURE

$\Re^n$	n-dimensional data space
I^n	scaled unit n-dimensional data space
P	input pattern $P = (p_1, p_2, \ldots, p_n)$, where $0 \leq p_i \leq 1$ for $i = 1, \ldots, n$
X	universe discourse
F^1	the input representation layer in the bottom of the attentional subsystem of the F3MCNN system
F^2_j	the jth sub-layer of the membership calculating layer in the middle of the attentional subsystem of the F3MCNN system
F^3	the cluster representation layer in the top of the attentional subsystem of the F3MCNN system
C_j	fuzzy central point of the jth cluster where $C_j = (c_{j1}, c_{j2}, \ldots, c_{jn})$
A_j	fuzzy variation vector of the jth cluster where $A_j = (a_{j1}, a_{j2}, \ldots, a_{jn})$
U_j	minimum point of the jth cluster's hyper-box where $U_j = (u_{j1}, u_{j2}, \ldots, u_{jn})$
V_j	maximum point of the jth cluster's hyper-box where $V_j = (v_{j1}, v_{j2}, \ldots, v_{jn})$
N	total number of patterns presented
N_j	fuzzy number of patterns of the jth committed cluster
$H_j(P)$	fuzzy hyper-box similarity degree of the input pattern P to the jth cluster
$S_j(P)$	fuzzy statistical similarity degree of the input pattern P to the jth cluster
ρ	user-defined vigilance for the maximum size of the hyper-boxes

CHAPTER I

INTRODUCTION

The combination of neural networks and fuzzy set theory develops a synergetic system that can handle pattern clustering problems more efficiently and accurately. A pattern clustering system, which deals with perturbed and imperfect elements in the real world, should have real-time and unsupervised learning ability and be able to represent and manipulate inexact information just like human beings. Pattern clustering is an unsupervised pattern classification process that assigns a pattern to its proper place. It is one of the inexact problems for which mathematical methods have not been defined. Human beings handle it subconsciously and without knowing how they really solve it. There have been considerable interest and rapid advances in both the development and research of pattern clustering in the last decade, but many questions are still open. In this study, we develop a new real-time pattern clustering model that can represent and manipulate inexact information.

In the following of this Chapter, we provide the background information related to our study for our readers in Section 1.1. Section 1.2 describes problems in neural network clustering systems and fuzzy neural network systems. Section 1.3 explains the motivation of our study. Section 1.4 provides the objectives of our study. Section 1.5 outlines the remainder of this dissertation.

1.1 Background Information

Although the computing power of digital computers have been improved tremendously in the past few decades, a human brain is still more efficient than advanced computers in performing complex tasks such as image and speech recognition. A human brain performs computation in a different manner from a conventional digital computer does. Engineers and scientists have tried to develop intelligent machines that function similarly to the human brain. Although the understanding of biological neural systems is not developed enough to exploit the function similarity between biological and artificial neural systems, there are many artificial neural systems developed to enhance the quality of our lives and make many difficult tasks easier to accomplish [McCulloch 43] [Hebb 49] [Rosenbalatt 58] [Widrow 60] [Widrow 62] [Nilsson 65] [Fukushima 80] [Kohonen 77] [Kohonen 82] [Kohonen 84] [Kohonen 88] [Anderson 77] [Amari 72] [Amari 77] [Grossberg 77] [Grossberg 82] [Hopfield 82] [Hopfield 84] [Rumelhart 86]. Artificial neural networks can enhance the enormous processing power of the digital computer with the ability to make sensible decisions and learn by ordinary experience, as humans do. In the meantime, the simple and regular architecture of artificial neural systems makes them easy to implement.

Most of human reasoning is approximate rather than exact. Humans have the ability to make decisions from uncertain and imprecise information. We can conceive distorted speech, recognize unclear pictures, and drive in tight traffic. In these cases, we manipulate approximate and uncertain information differently from how the conventional computer does. Usually, the objects of our reasoning are general and imprecise rather than precise and sharp. Fuzzy computing attempts to manipulate uncertain information in same the way human reasoning does. The rapidly growing number of applications that deal with complex and ill-defined problems suggests that it represents a natural development in the evolution of understanding how humans can reason without the use of numbers.

Pattern clustering is a sub-field of pattern recognition that has become a major area of research and development activity in the last few decades. The goal of pattern clustering is to classify complex patterns of information and to automate these functions using computers. Scientists from different areas including statistics, neural networks, and fuzzy systems have developed different pattern clustering techniques from each of their aspects. We provide the background information for the field of pattern clustering in the following subsections.

1.1.1 Pattern Clustering

The pattern clustering process can be generally presented in the following way. Consider a finite set of unlabeled patterns $P = \{p_1, p_2, \ldots, p_m\}$, where each pattern p_i is an n-dimensional feature vector. The clustering process partitions P into c-many classes, where c is either predetermined or dynamically determined during cluster learning. As a result, similar patterns are classified into the same cluster, and patterns that differ significantly are put into different clusters. How to determine the number of clusters and how to choose the proper grouping metric are two major concerns in the research field of pattern clustering. Different pattern clustering techniques have been developed to find the underlying structure of the patterns, based on statistic theory, fuzzy set theory, and neural network theory.

1.1.1.1. Statistical Clustering Statistical techniques have played a major part in pattern clustering in the last few decades. Researchers have developed many clustering algorithms, including the nearest mean clustering algorithm [Fukunaga 70] [Hall 65], the branch and bound algorithm [Koontz 75a], the maximum likelihood estimate algorithm [Day 69] [Wolfe 70], the estimation of density gradient algorithm [Fukunaga 75], the normality test algorithm [Fukunaga 86], and the graph theoretic approach [Koontz 75b]. Fukunaga classified these clustering algorithms into two approaches [Fukunaga 90]:

(1) Parametric Approach: In the parametric approach, a clustering criterion is defined to minimize an objective function. The given input patterns are classified into a number of clusters in order to optimize the clustering criteria. Normally, a parametric approach contains an iterative procedure. The iterative procedure stops when the objective function converges to a user defined criterion during successive presentation of input patterns. Parametric approaches suffer several major problems, such as the convergence being not guaranteed, the output being always dependent on the sequence of input patterns, and the termination being forced.

(2) Nonparametic Approach: In nonparametric approaches, neither clustering criteria nor assumed mathematical forms for the distribution are defined. The density function of the input patterns is used to separate patterns. The valley of the density function may be considered as the naturally boundary that separates the patterns.

These statistical clustering approaches classify patterns based on statistical properties such as mean vectors, covariance matrices, the density function, and the distribution function. There are many commercial clustering products available, for example, ISODATA, FIRGY, CLUSTER, and WISH [Dubes 76].

1.1.1.2 <u>Fuzzy Clustering</u> Fuzzy sets [Zadeh 65] provide a new direction to classical clustering systems by allowing a pattern to belong to several clusters in different degrees. Usually, people are more interested in the fuzzy representation than the statistical representation of a pattern to a class. Because in the fuzzy representation "a grade of membership of the pattern to the class ω_i is α ", α indicates how close the pattern is to the class ω_i. While in the statistical representation "a probability the pattern belongs to the class ω_i is α", α represents the frequency which that pattern belongs to the class ω_i in repeating experiments.

One of the widely used fuzzy clustering methods being a fuzzy relative of ISODATA and called fuzzy c-means has been introduced by [Dunn 74] and developed by [Bezdek 81] in detail. This fuzzy clustering method defines an object criterion function. Minimization of the object function is obtained by means of iterative procedure.

1.1.1.3 Neural Network Clustering In pattern clustering, neural networks provide the remarkable ability to determine the size, number, placement, and shape of pattern clusters. The tremendous advantages of a clustering neural network system are its adaptive learning and massive parallel processing abilities. The popular clustering neural networks can be classified by the learning algorithm as follows:

(1) Competitive Learning: This class of clustering neural networks uses the *c*-means similar algorithm to find the centers of clusters in the pattern space. This class of clustering networks uses an unsupervised scheme to find the best set of weight vectors for hard clusters in an iterative and sequential procedure. A learning rate is defined which decreases with time in order to force convergence of the iterative procedure. The Kohonen network [Kohonen 88] is one of the well-known network of this class of clustering network.

(2) Adaptive Resonance Theory: Adaptive resonance theory (ART) was introduced by [Grossberg 76] and [Grossberg 80]. Numerous extensions and refinements are found in [Carpenter 87], [Carpenter 90], [Carpenter 91a], [Carpenter 91b], and [Carpenter 91c]. The ART type neural networks create clusters from the input patterns by themselves. For some applications without a pre-specified number of clusters, the ART type neural networks are able to classify patterns depending on the natural properties of the applications. The ART type neural networks generate new clusters when the presented pattern can not be classified into any encoded cluster. The generation of the new cluster will not deconstruct any previous encoded cluster.

1.1.2 Neural Networks

The first formal definition of an artificial neuron model was formulated by [McCulloch 43]. Every neuron model consists of a processing element with synaptic input connections and a single out. The neural network can be defined as an interconnection of neurons such that output of neurons is connected to all other neurons including themselves. The synaptic connections between neurons are unidirectional. A neural network may contain several layers of neurons. Neurons in the same layer perform the same activation function.

There are many artificial neural networks developed in last few decades [McCulloch 43] [Hebb 49] [Rosenbalatt 58] [Widrow 60] [Widrow 62] [Nilsson 65] [Fukushima 80] [Kohonen 77] [Kohonen 82] [Kohonen 84] [Kohonen 88] [Anderson 77] [Amari 72] [Amari 77] [Grossberg 77] [Grossberg 82] [Hopfield 82] [Hopfield 84] [Rumelhart 86]. They can be classified as supervised learning, unsupervised learning, and recording learning neural networks according to their learning mode. Additionally, they can be classified as feedforward or recurrent neural networks. In our study, we are interested in the adaptive resonance theory 1 (ART-1) network that is an unsupervised clustering network [Carpenter 87] [Carpenter 88].

1.1.3 Fuzzy Set Theory

Fuzzy sets were introduced by [Zadeh 65] to represent and manipulate data that are imprecise and fuzzy. Researchers and theoreticians [Kandel 86] [Kosko 92] presented their definitions and representations of fuzzy sets. Zadeh extended the bivalent indicator function in traditional set theory to a multi-value indicator called a membership function. His extension provided a mechanism to present linguistic constructs such as "tall," "short," "many," and "few," and it provided a new tool that can be applied to pattern recognition. This new tool can be used to describe the degree to which a pattern is closer to a cluster. Traditional set theory uses probability theory to explain whether an event will occur. While probability indicates what is the chance an event will occur, fuzzy set theory measures the degree to which an event occurs.

The following definitions related to fuzzy set theory are abstracted from [Zimmermann 91]. If X is a collection of objects denoted generically by x, then a fuzzy set A in X is a set of ordered pairs:

$$A = \{(x, m_A(x)) \mid x \in X\}, \tag{1.1}$$

where the membership function $m_A(x)$ describes the degree to which the element x belongs to the set A and $m_A(x)$ is in the interval $[0,1]$. When $m_A(x)$ equals 0, it represents no membership; when it equals 1, it represents full membership. Elements with a zero degree of membership are normally not listed.

The operations on fuzzy sets are extensions of the basic classical set-theoretic operations. They constitute a consistent framework for the theory of fuzzy sets. Some of the common operations include intersection, union, and complement. With A and B being fuzzy sets in X, these operations are defined as follows.

Intersection: The membership function $m_C(x)$ of the intersection $C = A \cap B$ is pointwise defined by

$$m_C(x) = min\ \{m_A(x), m_B(x)\}, \quad \forall\ x \in X. \tag{1.2}$$

Union: : The membership function $m_C(x)$ of the intersection $C = A \cup B$ is pointwise defined by

$$m_C(x) = max\{m_A(x), m_B(x)\}, \quad \forall\ x \in X. \tag{1.3}$$

Complement: The membership function of the complement of the fuzzy set A, $\overline{A}$, is defined by

$$m_{\overline{A}}(x) = 1 - m_A(x), \qquad \forall\ x \in X. \tag{1.4}$$

1.2 The Problems

In neural network clustering, unsupervised and competitive learning algorithms adjust the weight vectors to classify input patterns into a number of clusters. There are some generic disadvantages in this class of neural networks [Caudill 89]:

(1) There is no hierarchical knowledge representation between patterns. Two input patterns are either classified to be in the same cluster or not. There is no representation of the information among clusters.

(2) Once an output neuron fails, we lose the representation of a whole category. This class of neural networks is not robust enough to prevent degradation or failure.

(3) For a given clustering problem, the choice of the proper clustering metric and the different number of clusters will decide the different clustering schemes. Which clustering scheme is more appropriate to the clustering problem? The answer is not clear; it is fuzzy.

(4). The crisp membership that represents the belongings of a pattern to a hard cluster limits the ability to describe the relationship between two clusters.

Fuzzy clustering neural networks are introduced to classify patterns into clusters that are represented by fuzzy sets. In fuzzy clustering representation, an input pattern belongs to numerous fuzzy set clusters in different degrees. This fuzzy membership function enables the fuzzy clustering system to describe the relationship among clusters for a pattern. There are two categories of fuzzy clustering neural networks; one is based on the competitive *c*-means algorithm and the other is developed from the adaptive resonance theory network.

Our study is focused on the fuzzy ART-type clustering neural networks that include the fuzzy ART clustering neural network (fuzzy ART) [Carpenter 91] and the fuzzy min-max clustering neural network (fuzzy Min-Max) [Simpson 93]. In the ART-type fuzzy clustering neural networks, a hyper-box defines a region of the *n*-dimensional pattern space and represent a cluster. There are some flaws in these fuzzy clustering neural systems:

(1) In fuzzy ART, hyper-boxes may overlap with each other. This results in an input pattern that may belong to different clusters in full membership. This in turn results in the problem of full membership ambiguity and also weakens the use of fuzzy sets.

(2) In fuzzy ART, although the intersection and union operations are replaced by the min and max operations from fuzzy set theory, the relationship to fuzzy sets is not suitably identified.

(3) In fuzzy ART, the cluster choice function does not present any information about the degree of an input pattern belonging to a cluster. It is used only to select the candidate resonant cluster.

(4) In fuzzy Min-max, hyper-box overlapping is not allowed. Once a hyper-box extension occurs, the tedious work for hyper-box overlap checking and hyper-box contraction must be performed.

1.3 Motivation of the Study

In pattern clustering (unsupervised classification) problems, unlabeled patterns are split into a number of classes with respect to a suitable similarity measure. Similar patterns are classified to the same cluster, while the patterns that differ significantly are put in different clusters. There are many classical clustering techniques developed based on the theory of mathematical statistics [Fukunaga 90]. In the past few decades, artificial neural networks have become a very popular field of research in cognitive science, computer science, signal processing, and neurobiology. Among these artificial neural networks, an important group of neural networks can be used to detect clusters of data. The most widely used clustering neural networks are the Kohonen network [Kohonen 88] and the adaptive resonance theory 1 network [Carpenter 87]. The Kohonen network classifies input patterns into one of the specified number of *c* categories based on an unsupervised and winner-take-all learning technique. In our study, we are interested in the ART-type network that creates clusters dynamically and learns new patterns without having to be retrained with previously learned patterns.

Since the introduction of fuzzy set theory [Zadeh 65], it has been obvious that it has a strong impact on techniques of pattern recognition. The fuzzy set representation of clusters provides partitioning results with additional information supplied by the cluster membership values. Many researchers applied fuzzy sets to pattern clustering and yielded an amount of interesting and useful results [Backer 76] [Bezdek 81] [Bezdek 92] [Carpenter 91] [Choe 92] [Ruspini 72] [Pedrycz 86] [Simpson 93]. Two categories of fuzzy clustering neural networks are very popular in the field of research:

(1) The fuzzy c-means type clustering neural networks (FCM-type): The first fuzzy *c*-means clustering neural network was developed by Dunn [Dunn 74] and later generalized by Bezdek [Bezdek 73] [Bezdek 74] [Bezdek 75]. The FCM-type neural networks find the optimal cluster centers and membership functions that minimize the sum of the weighted squares Euclidean distances between patterns and cluster centers. An iterative algorithm of FCM-type clustering neural networks is given in [Bezdek 81]. This type of fuzzy clustering networks suffers from several major problems, for example, the convergence is slow and not guaranteed, the assumption of the number of clusters is necessary, and the termination is forced. Recently, several studies related to the convergence and optimality of solutions generated by FCM-type

clustering neural networks are reported in [Bezdek 92] [Choe 92] [Ismail 86] [Hathaway 86] [Selim 86].

(2) The fuzzy ART type clustering neural networks (fuzzy ART-type): The first work that introduced fuzzy sets into the ART neural network was presented by Simpson [Simpson 90]. Since the introduction of the first fuzzy ART neural network, there have been other fuzzy ART-type neural networks introduced to the field of pattern clustering (e.g., the fuzzy min-max clustering network [Simpson 93] and the fuzzy ART network [Carpenter 91]). The fuzzy ART-type clustering neural networks use hyper-boxes to represent clusters in the pattern space. They use the minimum and maximum points of a hyper-box and the hyper-box membership function to define a fuzzy cluster. When input patterns are presented, the fuzzy ART-type clustering neural networks create and adjust hyper-boxes in pattern feature space to adapt patterns. Although these fuzzy ART-type clustering neural networks demonstrate good performance [Carpenter 91] [Simpson 93], they still suffer some disadvantages. The fuzzy Min-Max system requires a tedious work for hyper-box overlap checking and contraction when the number of hyper-boxes increases. In the fuzzy ART system, overlapped hyper-boxes lead to the confusion of pattern belongings.

In this study, we intend to develop a novel fuzzy clustering neural network system that is expected to perform pattern clustering without the shortcomings of the FCM-type and the fuzzy-ART-type clustering neural networks. However, we are not trying to develop an optimal clustering technique that will out-performed other techniques in every application problem, since a caution has been pointed out in [Pedrycz 90] that the structure of the data set obtained might be superimposed rather than detected. Usually the distance metric used to calculate the similarity degree between patterns determines the representation of the structure of clusters, for example, hyper-spherical and hyper-rectangular representations. In [Dubes 79], they suggest that users of clustering algorithms apply several clustering approaches and check for common clusters instead of searching the cluster validity in an individual clustering. The details of objectives of the study are described next.

1.4 Objectives of the Study

In this study, we develop a new pattern clustering model called the fuzzy minimum mean maximum clustering neural network (F3MCNN) that is a synergetic combination of a neural network with the fuzzy set theory. An F3MCNN system has the neural network system architecture that creates pattern clusters in a fashion similar to the ART neural network. Moreover, it has the fuzzy set information representation that can present and manipulate the inexact information. We expect that the F3MCNN model performs pattern clustering without the shortcomings of the FCM-type and the fuzzy-ART-type clustering neural networks mentioned in the previous section. The proposed model will provide the following special features:

(1) It is a self-organized pattern clustering neural network. An F3MCNN system should be able to create and adapt pattern clusters from input patterns by adjusting the size and the shape for clusters that are encoded in the synaptic connection weights of F3MCNN.

(2) It is a real-time pattern clustering system. An F3MCNN system does not assume a pre-defined number of clusters and is adaptive to dynamically changing conditions. A user-defined vigilance limits the maximum size for fuzzy hyper-boxes and will decide the clustering results. An F3MCNN should be able to learn new patterns without having to be retrained with previously learned patterns.

(3) It is able to represent and manipulate inexact information. The F3MCNN model should be able to classify both linear separable and non-linear separable clusters.

(4) It is a self-stabilized pattern clustering system. We define cluster stability as there being no change in the size of the hyper-boxes during successive presentation of the same patterns. An F3MCNN system should achieve the cluster stability after one pass of patterns.

(5) It does not require time-consuming work for hyper-box overlap checking and contraction. By providing the extra fuzzy statistical similarity degree, an F3MCNN system should be able to handle hyper-box overlapping without the tedious work in [Simpson 93].

(6) It does not suffer the problem of full membership ambiguity. Our solution for the problem of full hyper-box membership ambiguity should be superior to that of fuzzy ART [Grossberg 91].

1.5 Organization of the Dissertation

The remainder of this dissertation is organized as follows. Chapter II provides the literature review for the fuzzy set theory, the neural network, and different pattern clustering techniques. Chapter III describes the fuzzy similarity used in the F3MCNN model in detail, including (1) the hyper-box fuzzy set function and the statistical fuzzy set function defined in F3MCNN used to calculate the fuzzy hyper-box similarity degree and the fuzzy statistical similarity degree respectively, (2) the pattern similarity selection procedure, and (3) an example to show the differences in pattern similarity selection among fuzzy ART, fuzzy Min-Max, and F3MCNN. Chapter IV introduces the system architecture and the clustering algorithm of the F3MCNN model. Chapter V presents the experimental results from the iris flower data by using F3MCNN and compares the F3MCNN model with other fuzzy clustering neural networks, including (1) fuzzy *c*-means, (2) fuzzy ART, and (3) fuzzy Min-Max. Finally, Chapter VI provides the summary and conclusions of our work.

CHAPTER II

REVIEW OF THE LITERATURE

2.1 Introduction

Many scientists and engineers have dedicated their effort to pattern recognition problems, especially clustering procedures. Clustering of a group of patterns forms a basic problem in many areas of human activity. It deals with the task of classifying a set of patterns into a number of more or less homogenous clusters with respect to a suitable similarity measure. As a result, patterns that are similar are allocated to the same cluster, while patterns that are differ significantly are put in different clusters.

Scientists and engineers from different fields are concerned with the idea of designing and making automata that can carry pattern clustering as we human beings do. Statistical approaches to pattern clustering have a highly developed theory and are regarded as fundamental to the study of pattern clustering [Fukunaga 90]. An important group of neural networks is developed to detect clusters of data [Zurada 92]. The self-organizing and self-adjusting properties of clustering neural networks have a strong impact on techniques of pattern clustering. The application of fuzzy sets to pattern clustering has yielded a reasonable amount of interesting and useful results [Pedrycz 90]. Moreover, the combination of clustering neural networks and fuzzy set theory has developed an evolutionary process to develop a fast, efficient, and reliable pattern clustering neural network system [Simpson 92] [Simpson 93] [Carpenter 91a] [Carpenter 91b].

The following sections provide a review of pattern clustering techniques developed from statistical theory, neural network theory, fuzzy set theory, and fuzzy neural network theory. Even though the primary focus of this dissertation is on the fuzzy neural network clustering, this review includes statistical pattern clustering, neural network pattern clustering, fuzzy pattern clustering, and fuzzy neural network pattern clustering for completeness.

2.2 Statistical Pattern Clustering

The work on statistical pattern recognition which was started in the late fifties made use of the statistical decision theory to classify patterns. The list of the most commonly used statistical pattern clustering techniques includes: ISODATA [Duda 73], the nearest mean clustering algorithm [Fukunaga 70] [Hall 65], the branch and bound algorithm [Koontz 75a], the maximum likelihood estimate algorithm [Day 69] [Wolfe 70], the estimation of density gradient algorithm [Fukunaga 75], the normality test algorithm [Fukunaga 86], and the graph theoretic approach [Koontz 75b].

ISODATA, the nearest mean clustering algorithm, the branch and bound algorithm, and the maximum likelihood estimate algorithm are classified as parametric approaches [Fukunaga 90]. In parametric approaches, clustering criteria are defined to classify patterns to optimize the criteria. The commonly used criteria are the class separation measures, such as within-class scatter matrix, between-class scatter matrix, and mixture scatter matrix in [Fukunaga 90]. Clustering criteria are defined to determine the structure of the classification boundary. A clustering algorithm contains an iterative procedure to optimize the clustering criteria. In another parametric approach, the assumption of distribution of patterns is expressed by a mathematical form. The clustering algorithm consists of finding the parameter values to match the distribution of the data. This group of parametric approaches suffer several flaws such as there is no guarantee of the convergence of the iterative process and the process might stop at a local minimum point and fail to find the global minimum point.

The estimation of density gradient algorithm, the normality test algorithm, and the graph theoretic approach are classified as nonparametric approaches [Fukunaga 90]. In nonparametric approaches, neither clustering criteria nor assumed mathematical forms for the pattern distribution are used. Instead, the valley of the pattern density function is used to separate patterns. We may consider the valley as the natural boundary that separates the modes of the distributions. Usually, the boundary is complex and not expressible by any parametric form. Therefore, most nonparametric approaches are implicitly or explicitly based on the estimate of the density gradient to characterize the local structure of the valley. Each pattern is moved toward the direction of its gradient in a repeating process. The valley becomes wider at each iteration and patterns form compact clusters.

2.3 Neural Network Pattern Clustering

A number of neural architectures and theories are introduced and developed to classify input patterns without a priori information from the teacher [Grossberg 74] [Grossberg 82] [Kohonen 88]. This class of neural networks is called clustering (unsupervised recognition) neural networks. The objective of clustering neural networks is to categorize or cluster patterns. Clustering neural networks usually have a simple architecture and exhibit remarkable properties of self-organization. The learning in clustering neural networks is based on clustering of input patterns. Since no a priori information is available regarding a pattern's membership in a particular cluster, the similarity of incoming patterns is used as the criterion for clustering. During the training, dissimilar clusters are rejected and the most similar one is accepted for cluster learning. The similarity measure may be defined by the scalar product or distance between the pattern and cluster's weights. Most scientists and engineers developed clustering neural networks based on two typical clustering neural networks: the Kohonen network [Kohonen 88] and the ART-1 network [Carpenter 87].

The Kohonen network classifies input patterns into one of the specified number of p categories detected in the training set $\{ x_1, x_2, \ldots, x_N \}$. The Kohonen network is a two-layer feedforward neural network. It uses the winner-take-all strategy for learning algorithm. The learning algorithm treats the set of p random initial weight vectors as variable vectors that need to be learned. Before the learning, all random initial weight vectors are normalized. The scalar products of the pattern and weight vectors are used as the similarity measures to select the winning cluster. The winning cluster is rewarded with a weight adjustment, while the weights of other clusters remain unaffected. In adjusting the weight of winning cluster, a learning rate is defined which decreases during learning to avoid divergence. The Kohonen network suffers from some limitations. One obvious deficiency is that linearly non-separable patterns cannot be handled by this network. The second limitation is that initial weights may become stuck in isolated regions without forming adequate clusters even for linearly separable patterns. One of weight selection methods that improves the chances for successful training is presented in [Hecht-Nielsen 87].

The ART-1 network serves the propose of cluster discovery. The ART-1 network is a two-layer network and there are bi-direction connection weights between these two layers. The bottom layer is the input fan-out layer and the top layer is the pattern cluster layer. The bottom-up connection weights are used to compute the matching score that reflects the degree of similarity of the input pattern to previously encoded clusters. The winning neuron with the maximum degree of similarity will be the candidate cluster to learn the input pattern. The top-down connection weights are used to check the similarity of the candidate cluster with the stored cluster reference pattern and compare with a user-define vigilance. The vigilance is the threshold that sets the degree of required similarity. If the degree of similarity is less than the threshold, the input pattern resonates with the candidate cluster and the weight vector of the candidate cluster will be adjusted. Otherwise, a search will perform along those encoded clusters to find next candidate. If no encoded cluster is found, the ART-1 network will create a new cluster for the input pattern. The steps of the learning algorithm for cluster discovery can be found in [Lippmann 87] [Pao 89].

2.4 Fuzzy Pattern Clustering

Since the development of fuzzy set theory [Zadeh 65] it has been obvious that it has a strong impact on techniques on pattern recognition. The question of pattern clustering is itself a fuzzy one and the representation of clusters by fuzzy sets may be more appropriate. Many researchers have developed different clustering techniques by applying fuzzy sets to pattern clustering [Dunn 74] [Bezdek 74] [Bezdek 75] [Bezdek 77] [Bezdek 78] [Bezdek 80] [Ruspini 69] [Ruspini 70] [Ruspini 72] [Pedrycz 85] [Pedrycz 86]. The main advantage of all fuzzy clustering techniques lies in the fact that they provide partitioning results with additional information supplied by the cluster membership value.

One of the widely used fuzzy pattern clustering methods is a fuzzy relative of ISODATA and called the fuzzy c-means clustering algorithm. The fuzzy c-means clustering algorithm was initially developed in [Dunn 74] and has been further studied and developed in [Bezdek 81]. Dunn extended the original objective function in the classical c-means clustering network to a fuzzy clustering criterion and developed the fuzzy c-means algorithm to minimize the fuzzy objective function through an iterative process. Bezdek extended the fuzzy objective function proposed by Dunn to a more general form by introducing the weighting exponent into the fuzzy objective function. In [Choe 92], a method for determining the optimal weighting exponent was proposed.

A general fuzzy c-means algorithm is described as follows. The number of clusters c, weighting exponent m, and error tolerance ε are specified for a clustering problem in fuzzy c-means methods. A set of randomly initialized weight vectors is used to represent prototype of clusters in the fuzzy c-means clustering method. In each iteration, the membership degree to every cluster of the input pattern is calculated and then the clusters' prototypes are updated from the new memberships. The iteration continues until the changing of clusters' prototypes is smaller than the pre-defined parameter ε.

2.5 Fuzzy Neural Network Pattern Clustering

Fuzzy clustering neural networks were developed by combining fuzzy set theory and clustering neural networks into an integrated model [Carpenter 91] [Simpson 93] [Bezdek 92]. They inherit the self-organization property from clustering neural networks and are able to represent imprecise information as fuzzy sets. The hard cluster membership value which is in { 0, 1} in clustering neural systems is replaced by the fuzzy membership value which is in the interval [0, 1]. There are two types of fuzzy clustering neural networks; the c-means type fuzzy clustering neural system and the ART type clustering neural system. The c-means type fuzzy clustering neural system assumes a fix number of clusters and used cluster prototype to represent fuzzy clusters. The mean vector and the covariance matrix are used to define a fuzzy cluster. While in the ART-type fuzzy clustering system, the number of clusters is dynamically determined during pattern learning. A fuzzy cluster in the ART-type fuzzy clustering system is represented by a hyper-rectangle, which is presented by its minimum and maximum points. The fuzzy cluster's hyper-rectangle is adjusted to include a new pattern when the pattern is classified into the fuzzy cluster. A user-defined parameter will limit the maximum size for clusters' hyper-rectangle and also decide clustering result.

In [Bezdek 92], Bezdek, Tsao, and Pal propose the fuzzy Kohonen clustering network that integrates the fuzzy *c*-means algorithm into the learning rate and updating strategies of the Kohonen network. The fuzzy Kohonen network modifies cluster prototypes update rule in [Huntsberger 89] by introducing the learning rate which is computed from membership values in the fuzzy c-means algorithm. The fuzzy Kohonen network is proved to address some problems of the classical Kohonen clustering network.

Carpenter and Grossberg incorporated fuzzy set theory into the ART network to generate the fuzzy ART neural network (fuzzy ART) [Carpenter 91]. Fuzzy ART is able to learn both binary and analog input patterns. Input patterns are scaled into the interval [0, 1] and are normalized to prevent cluster proliferation. Intersection and union operators in the ART network are replaced by the maximum and minimum operators in the fuzzy set theory. Clusters are represented as hyper-rectangles in the pattern space. The information about the minimum and maximum points of a hyper-rectangle is encoded into the connection weights from the input nodes to the cluster node.

Simpson proposed a fuzzy min-max clustering network (fuzzy Min-Max) in which pattern clusters are implemented as fuzzy sets [Simpson 93]. The membership function for a fuzzy cluster is constructed from the minimum and maximum points of a hyper-box which represents the cluster. Patterns inside the cluster hyper-box have the full membership degree. The membership degree decreases when an input pattern is separated from the hyper-box core. In order to avoid the problem of full membership ambiguity, the hyper-box overlap is not allowed. A hyper-box overlap checking and hyper-box contraction procedure is performed whenever a hyper-box expands. Fuzzy Min-Max has the capability to incorporate new data and create new clusters without losing encoded clusters. The degree of membership information is useful in the high level decision making and information processing.

CHAPTER III

FUZZY SIMILARITY

3.1 Introduction

In this Chapter, we review some background information of fuzzy set functions for measuring pattern similarity and define two new fuzzy set functions and a pattern similarity selection procedure for the F3MCNN model. These two defined fuzzy set functions will be utilized as activation function of neurons in the network of F3MCNN. Also, the defined pattern similarity selection procedure will play a major role in our F3MCNN clustering algorithm. The F3MCNN system architecture and clustering algorithm are introduced in the next Chapter.

The relation between the theory of fuzzy sets [Zadeh 65] and the theory of pattern clustering rests on the fact that most real-world classes are fuzzy in nature. Therefore, given a pattern *P* and a cluster *C*, the basic question in most problems related to pattern clustering is the degree of *P* belonging to *C* instead of whether *P* is an element in *C* or not. Most practical problems in pattern clustering do not lend themselves to a precise formulation. Generally, practical problems in pattern clustering are not linearly separable. Consequently, less precise techniques might have solutions to the intrinsic imprecision in pattern clustering cases. The representation for clusters using fuzzy sets provides solutions to present and manipulate the intrinsic imprecision in pattern clustering.

Two fuzzy set functions are defined in the F3MCNN model to measure similarity degree of input patterns to clusters, which might be presented in an inexact way. They are the hyper-box fuzzy set function and the statistical fuzzy set function used to measure the fuzzy Hyper-box Similarity Degree (fuzzy HSD) and the fuzzy Statistical Similarity Degree (fuzzy SSD) respectively. In the F3MCNN model, we use both the fuzzy HSD and the fuzzy SSD to represent the relationship between patterns and clusters. A pattern's fuzzy HSD to a cluster exhibits its geometrical relationship to the cluster's learned concept. Meanwhile, its fuzzy SSD to a cluster indicates its statistical relationship to the cluster's fuzzy prototype. Since there are two fuzzy similarity degrees utilized to measure similarity between input patterns and clusters, we introduce a pattern similarity selection procedure into the F3MCNN model to select a cluster for matching the input pattern.

Basically, an F3MCNN system classifies patterns located in the same hyper-box to the same cluster. However, a problem of full membership ambiguity occurs when an input pattern is located inside a hyper-box overlapped area. We introduce the statistical fuzzy set function to classify patterns located inside the hyper-box overlapped area. Although a time consuming work for hyper-box overlap checking and contraction is introduced in fuzzy Min-Max [Simpson 93] to solve the problem of full membership ambiguity. It does not totally solve the ambiguity problem. In contrast, our model is able to completely overcome the problem of full membership ambiguity.

The contents of this Chapter are organized as follows. First, we review hyper-box fuzzy set functions used in previously introduced pattern clustering systems that are similar to our model and then present our new hyper-box fuzzy set function defined for the F3MCNN model in section 3.2. Secondly, in section 3.3 we review related fuzzy membership functions and define a new fuzzy set function to measure the fuzzy SSD of input patterns. Finally, in section 3.4, we explain the pattern similarity selection procedure defined in the F3MCNN model. Then, we provide an example for pattern similarity selection to show the different clustering results of the fuzzy ART, fuzzy Min-Max, and F3MCNN models, respectively.

3.2 Hyper-box Fuzzy Set Functions

In fuzzy ART-type clustering neural networks, a hyper-box is a geometric representation for a cluster in the feature space. A two-dimensional cluster representation is illustrated in Figure 3.1, where R_j is a two-dimensional hyper-box and U_j and V_j are its minimum and maximum points respectively. Patterns in the n-dimensional data space R^n are normalized or scaled to be in the n-dimensional unit cube I^n. The minimum and maximum points define the hyper-box of a cluster. Patterns inside the hyper-box have full hyper-box membership degree. In addition, a fuzzy set function is defined to associate with the hyper-box for determining the degree to which a pattern is contained within the hyper-box. Its hyper-box membership degree decreases to zero when the distance from the pattern to the hyper-box increases to a certain value. Patterns inside the same hyper-box belong to the same cluster. A cluster learns new patterns by adjusting its minimum

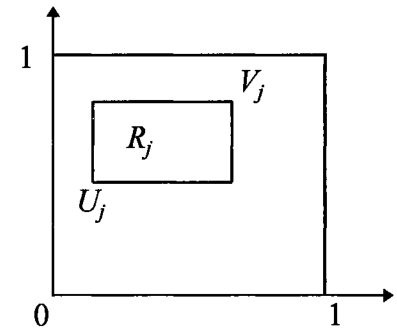


Figure 3.1 Two-dimensional cluster geometric representation in fuzzy ART-type clustering neural networks

and maximum points of the hyper-box. In the following sections, we review hyper-box fuzzy set functions defined in the previously introduced systems, fuzzy ART and fuzzy Min-Max, and reveal their potential problems. Then, we introduce the new fuzzy set function defined in the F3MCNN model to measure the fuzzy HSD.

3.2.1 Review of Hyper-box Fuzzy Set Functions

The Fuzzy ART [Carpenter 91] and fuzzy Min-Max [Simpson 93] neural networks are two typical fuzzy ART-type pattern clustering neural networks. These two models use the same hyper-box representation for clusters and assign full membership degree to patterns located inside the cluster's hyper-box. However, the fuzzy set functions that they use for measuring hyper-box similarity degrees are different. Also, they use different strategies to solve the hyper-box overlap problem.

3.2.1.1 Fuzzy Set Function in Fuzzy ART In the fuzzy ART model, a geometric interpretation of fuzzy ART with complement coding was developed. Thus the preprocessed complement coding form for an input pattern P is presented as

$$B = (P, P^c), \tag{3.1}$$

where $P^c = 1 - P$. A cluster, j, in fuzzy ART has a geometric representation as a rectangle R_j, as shown in Figure 3.1. The weight vector associated with the cluster j can be written in complement coding form as:

$$W_j = (U_j, V_j^c), \tag{3.2}$$

where U_j and V_j are vectors defining two corners of the rectangle R_j. The fuzzy set function defined in the fuzzy ART system to measure similarity degree is called the category choice function T_j. For the fast learning fuzzy ART [Carpenter 91], T_j is defined as

$$T_j = \frac{|B \wedge W_j|}{|W_j|}, \tag{3.3}$$

where the fuzzy AND [Zadeh 65] operator $\wedge$ is defined by

$$(x \wedge y)_i \equiv \min(x_i, y_i), \tag{3.4}$$

and where the norm $|.|$ is defined by

$$|x| \equiv \sum_{i=1}^{M} |x_i|, \tag{3.5}$$

where M is the dimension of the vector x. The candidate cluster selected from committed clusters to best match the input pattern P is the one with the maximum degree calculated by Equation (3.3). The fuzzy ART model uses the term

$$\frac{|x \wedge y|}{|y|} \tag{3.6}$$

in [Kosko 86] to define the category choice function T_j in Equation (3.3). It represents the degree to which y is a fuzzy subset of x. Therefore, the cluster selected with the maximum degree calculated by Equation (3.3) will always be the cluster whose weight vector w_j is the largest subset of the input pattern P. When the input pattern P is located inside the jth cluster's hyper-box, the degree of the cluster choice function T_j equals one. The degree will decrease when an input pattern is separated from the hyper-box. It is possible that an input pattern has the same maximum degree to more than one cluster. The fuzzy ART system allows clusters' hyper-boxes to overlap, which results in the problem of full membership ambiguity. To solve this ambiguity problem, the fuzzy ART system selects the cluster with the smallest index to learn the input pattern. In a fuzzy ART system, the smaller the cluster's index, the earlier it was created.

3.2.1.2 Fuzzy Set Function in Fuzzy Min-Max Simpson introduced a new fuzzy set hyper-box membership function in his fuzzy min-max neural network to measure the degree to which an input pattern falls within a hyper-box [Simpson 93]. To calculate the fuzzy hyper-box similarity degree for the input pattern P to the jth cluster, he defined the fuzzy set hyper-box membership function as

$$b_j(P, V_j, W_j) = \frac{1}{n}\sum_{i=1}^{n}\left[1 - f(p_i - w_{ji}, \gamma) - f(v_{ji} - p_i, \gamma)\right], \tag{3.7}$$

where $f(\,)$ is the two-parameter ramp threshold function

$$f(x, \gamma) = 1 \quad \text{if } x\gamma > 1, \qquad (3.8)$$
$$= x\gamma \quad \text{if } 0 \leq x\gamma \leq 1,$$
$$= 0 \quad \text{if } x\gamma < 0,$$

and where $P = (p_1, p_2, \ldots, p_n)$ is the input pattern, $V_j = (v_{j1}, v_{j2}, \ldots, v_{jn})$ is the minimum point for the jth hyper-box, and $W_j = (w_{j1}, w_{j2}, \ldots, w_{jn})$ is the maximum point for the jth hyper-box. The parameter γ is the sensitivity parameter that is used to regulate how fast the membership degree decreases when the input pattern is separated from the hyper-box. When the input pattern is located inside the hyper-box core, the value calculated by the fuzzy set hyper-box membership function equals one. The value decreases when the input pattern is separated from the hyper-box.

Simpson introduced the hyper-box overlap checking and hyper-box contraction processes into his learning algorithm in order to avoid the problem of full membership ambiguity. However, the hyper-box overlap checking and hyper-box contraction processes are exhaustive overhead. Besides, the hyper-box contraction process eliminates the stable learning advantage in ART-type neural networks, i.e., successive presentations of the data set is necessary in the learning process for fuzzy Min-Max neural networks to achieve clustering stability.

3.2.2 Hyper-box Fuzzy Set Function Defined in F3MCNN

The F3MCNN model is a variant of the fuzzy ART-type pattern clustering neural networks. We not only use the hyper-box geometric representation but also introduce the statistical characteristic representation to present clusters. We define a new hyper-box fuzzy set function to calculate the fuzzy HSD for an input pattern to a committed cluster. This calculated fuzzy HSD represents the geometric characteristic of the input pattern to the learned concept of a cluster. In F3MCNN, the hyper-box fuzzy set function to calculate the fuzzy HSD of the input pattern P to the jth cluster is defined as

$$H_j(P) = 1 - \frac{1}{n}\sum_{i=1}^{n} d(u_{ji}, v_{ji}, p_i), \tag{3.9}$$

where $d(\,)$ is the three-parameter distance function

$$d(u_{ji}, v_{ji}, p_i) = \begin{cases} u_{ji} - p_i & \text{if } p_i < u_{ji}, \\ 0 & \text{if } u_{ji} \le p_i \le v_{ji}, \\ p_i - v_{ji} & \text{if } v_{ji} < p_i, \end{cases} \tag{3.10}$$

and where $P = (p_1, p_2, \ldots, p_n)$ is the input vector, $U_j = (u_{j1}, u_{j2}, \ldots, u_{jn})$ is the minimum point vector of the jth cluster's hyper-box, and $V_j = (v_{j1}, v_{j2}, \ldots, v_{jn})$ is the maximum point vector of the jth cluster's hyper-box.

The summation part in the hyper-box fuzzy set function defined for F3MCNN calculates the d-function distance between the input pattern and the cluster's hyper-box. When the input pattern is located inside the cluster's hyper-box, the distance is zero. Therefore the value of the hyper-box fuzzy set function equals one. The distance increases but the fuzzy HSD of the pattern decreases when the input pattern is separated from the cluster's hyper-box.

The hyper-box fuzzy set function H defined in the F3MCNN model has the advantage of simplicity and is a more appropriate interpretation compared with the hyper-box membership functions defined in the other two models. The fuzzy ART model defines the hyper-box membership function based upon an interpretation of fuzzy sets as points in the unit hyper-cube [Kosko 92]. It utilizes the fuzzy subset-hood measure and is not appropriate for setting the membership function. Because even though the pattern is normalized or scaled to be a point inside the unit cube, it does not make the pattern fuzzy. In the fuzzy hyper-box membership function of the fuzzy Min-Max model, a parameter is introduced to regulate how fast the membership values decrease when an input pattern is separated from a hyper-box. This parameter is defined by users and will decide one decreasing rate for all clusters represented by fuzzy sets. It is not reasonable to assume that all fuzzy sets representing clusters have the same rate of decrease.

3.3 Statistical Fuzzy Set Function

In this section, we introduce two standard fuzzy set functions defined in [Zadeh 75] as the background information for defining the statistical fuzzy set function in the F3MCNN model. Then, we define a new fuzzy set function to calculate the fuzzy SSD of input patterns to clusters.

3.3.1 Background Information

The fuzzy set theory deals with a subset A of the universe discourse X, where the transition between full membership and no membership is gradual rather than abrupt. Let $X = \{x\}$ denote a space of objects. Then a fuzzy set A in X is a set of ordered pairs

$$A = \{(x, \mu_A(x)\}, x \in X \tag{3.11}$$

where $\mu_A(x)$ represents the degree of membership of x in A. For the reason of simplicity, we normalize $\mu_A(x)$ to be a number in the interval [0, 1], with the degrees of one and zero representing full membership and no membership respectively. There are two standard functions defined in [Zadeh 75] to express the membership function of a fuzzy set of the real line. They are defined as

$$S(x; \alpha, \beta, \gamma) = 0 \qquad \text{for } x \leq \alpha \qquad (3.12)$$

$$= 2\left(\frac{x-\alpha}{\gamma-\alpha}\right)^2 \qquad \text{for } \alpha \leq x \leq \beta$$

$$= 1-2\left(\frac{x-\gamma}{\gamma-\alpha}\right)^2 \qquad \text{for } \beta \leq x \leq \gamma$$

$$= 1 \qquad \text{for } x \geq \gamma$$

with $\beta = (\alpha + \gamma)/2$

and $$\pi(x; \beta, \gamma) = S(x; \beta - \gamma, \beta - \gamma/2, \gamma) \qquad \text{for } x \leq \gamma \qquad (3.13)$$

$$= 1 - S(x; \gamma, \gamma + \beta/2, \gamma + \beta) \qquad \text{for } x \geq \gamma.$$

In $S(x; \alpha, \beta, \gamma)$, the parameter β is the crossover point, i.e., $S(\beta; \alpha, \beta, \gamma) = 0.5$. In the π-

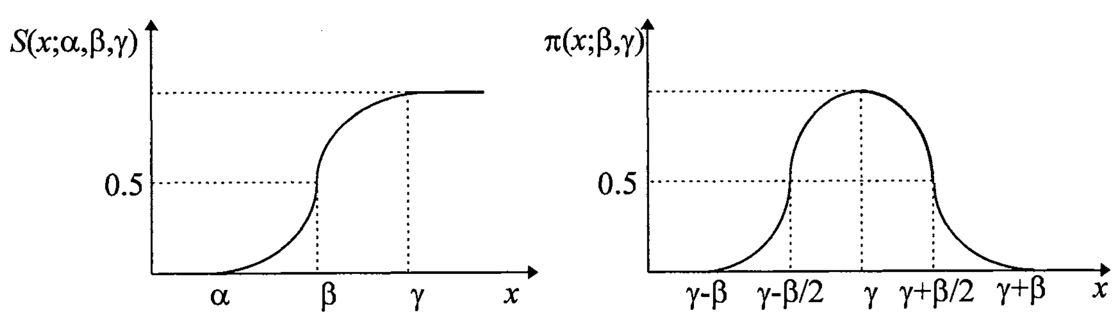


Figure 3.2 Fuzzy S-function and π-function

function, β is its bandwidth, i.e., the separation between the crossover points of a π-function, 2β is its support, i.e., the interval that the π-function has non-zero value, and γ is the central point at which π is unity. These two fuzzy set functions are illustrated in Figure 3.2. Equations (3.12) and (3.13) define the membership function corresponding to fuzzy sets 'x is large' and 'x is γ' respectively. Let γ represent the fuzzy central point for a fuzzy cluster, the π-function has a basic primitive which is useful in clustering patterns.

3.3.2 Statistical Fuzzy Set Function Defined in F3MCNN

We define the statistical fuzzy set function for the F3MCNN model to measure the fuzzy SSD of patterns to clusters. The fuzzy SSD of a pattern to a cluster exhibits its relation to the statistical characteristic of a fuzzy cluster. The statistical characteristic of a fuzzy cluster includes its fuzzy number of patterns, fuzzy central point and fuzzy variation vectors, as defined below. We utilize the π-function to define our statistical fuzzy set function as shown in Figure 3.3. The statistical fuzzy set function is a symmetric function. The fuzzy central point is defined as the central point of the statistical fuzzy set function. We define the fuzzy central point plus and minus the fuzzy variation as the crossover points for the statistical fuzzy set function. Therefore, each cluster has its own rate of decrease for its statistical fuzzy set membership function. In order to cover the hyper-box of a cluster, the support of the statistical fuzzy set function of a cluster is define at least twice the difference of the maximum point minus the minimum point. Therefore, the parameters used in the statistical fuzzy set function should include the fuzzy central point, the fuzzy variation vector, the fuzzy number of

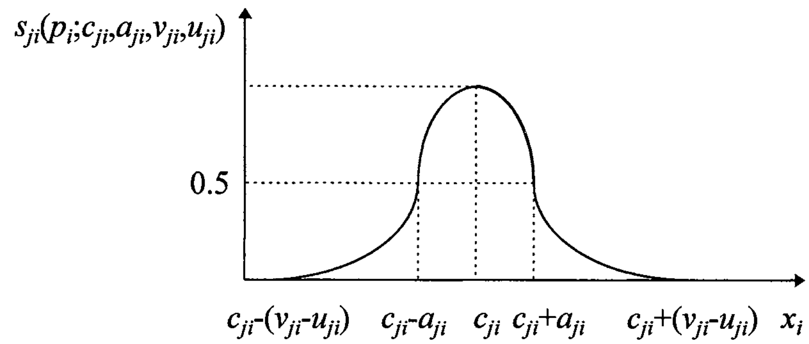


Figure 3.3 Statistical fuzzy set function in the *i*th dimension in F3MCNN

patterns, and the minimum and maximum points used in the hyper-box fuzzy set function. Before introducing the fuzzy set function to measure fuzzy SSD, we need to define the parameters used in the statistical fuzzy set function except the minimum and maximum points.

Definition 1: $C_j = (c_{j1}, c_{j2}, \ldots, c_{jn})$ is the fuzzy central point for the *j*th fuzzy cluster, where C_j is the average vector of those patterns that have been classified into the *j*th cluster. We define C_j as follows

$$C_j = \frac{1}{N_j} \sum_k P_k \tag{3.14}$$

where each input pattern P_k has full fuzzy HSD to the *j*th cluster and N_j is the fuzzy number of patterns in the *j*th cluster as defined next.

Definition 2: N_j is the fuzzy number of patterns in the jth cluster, that is, the number of patterns that have been classified into the jth cluster. Note that a pattern located in the hyper-box of a cluster is not necessarily classified as a pattern in this particular cluster in an F3MCNN system.

Definition 3: The deviation vector of the pattern P from the cluster j is defined as

$$D_j(P) = \left(\left|p_1 - c_{j1}\right|, \left|p_2 - c_{j2}\right|, \ldots, \left|p_n - c_{jn}\right|\right) \tag{3.15}$$

where $C_j = (c_{j1}, c_{j2}, \ldots, c_{jn})$ is the fuzzy central point for the jth fuzzy cluster. The fuzzy variation vector $A_j = (a_{j1}, a_{j2}, \ldots, a_{jn})$ for the jth cluster is defined as

$$A_j = \frac{1}{N_j} \sum_k D_j(P_k) \tag{3.16}$$

where each input pattern P_k is located in the hyper-box of the jth cluster, i.e., P_k has full fuzzy HSD to the jth cluster.

The fuzzy variation vector of a cluster is defined as the average of the deviation vector of those patterns that have been classified into the cluster. However, because F3MCNN is a real-time learning system, we are not able to calculate the deviation vector that runs from each of the previously presented patterns to the most recently updated fuzzy central point of a cluster. Therefore, we develop a modified learning rule to allow for updating of the fuzzy variation vector.

For a fuzzy cluster j with $C_j = (c_{j1}, c_{j2}, \ldots, c_{jn})$ as its fuzzy central point, $A_j = (a_{j1}, a_{j2}, \ldots, a_{jn})$ as its fuzzy variation vector, and $U_j = (u_{j1}, u_{j2}, \ldots, u_{jn})$ and

$V_j = (v_{j1}, v_{j2}, \ldots, v_{jn})$ as its minimum and maximum point vectors for its hyper-box, we define the five-parameter fuzzy set function s_{ji} that calculates the fuzzy SSD of the ith element of the input pattern $P = (p_1, p_2, \ldots, p_n)$ to the jth cluster along the ith feature dimension as

$$s_{ji}(p_i; c_{ji}, a_{ji}, u_{ji}, v_{ji}) = \begin{cases} 0 & \text{for } p_i \leq c_{ji} - (v_{ji} - u_{ji}) \\ \frac{1}{2}\left(\frac{p_i - (c_{ji} - (v_{ji} - u_{ji}))}{(v_{ji} - u_{ji}) - a_{ji}}\right)^2 & \text{for } c_{ji} - (v_{ji} - u_{ji}) \leq p_i \leq c_{ji} - a_{ji} \\ 1 - \frac{1}{2}\left(\frac{p_i - c_{ji}}{a_{ji}}\right)^2 & \text{for } c_{ji} - a_{ji} \leq p_i \leq c_{ji} + a_{ji} \\ \frac{1}{2}\left(\frac{p_i - (c_{ji} + (v_{ji} - u_{ji}))}{(v_{ji} - u_{ji}) - a_{ji}}\right)^2 & \text{for } c_{ji} + a_{ji} \leq p_i \leq c_{ji} + v_{ji} - u_{ji} \\ 0. & \text{for } p_i \geq c_{ji} + (v_{ji} - u_{ji}) \end{cases} \quad (3.17)$$

$s_{ji}(p_i; c_{ji}, a_{ji}, u_{ji}, v_{ji})$ is a convex fuzzy set function as shown in Figure 3.3. The defined statistical fuzzy set function is symmetrical, and c_{ji} is the fuzzy central point for the jth fuzzy cluster along ith feature dimension at which $s_{ji}(c_{ji})$ is unity. As explained earlier, we define $2a_{ji}$ as the bandwidth of the fuzzy set function, i.e., the points $c_{ji} - a_{ji}$ and $c_{ji} + a_{ji}$ are cross-over points of the fuzzy set function where $s_{ji}(c_{ji} - a_{ji}) = 0.5$ and $s_{ji}(c_{ji} + a_{ji}) = 0.5$. We define the support of the fuzzy set function s_{ji} as twice the size of the interval between the maximum point and the minimum point of the hyper-box for the fuzzy cluster. Therefore, $2(v_{ji} - u_{ji})$ is the width of the fuzzy set function s_{ji} and the closed interval $[c_{ji} - (v_{ji} - u_{ji}), c_{ji} + (v_{ji} - u_{ji})]$ is its support, i.e., a non-fuzzy subset of x_i such that $s_{ji}(p_i) > 0$ for every p_i in the open interval $(c_{ji} - (v_{ji} - u_{ji}), c_{ji} + (v_{ji} - u_{ji}))$.

We define the statistical fuzzy set function to calculate the fuzzy SSD to the jth cluster for the input pattern P as

$$S_j(P) = 0 \qquad \text{if for some } i,\ s_{ji} = 0 \tag{3.18}$$

$$= \frac{1}{n}\sum_{i=1}^{n} s_{ji}(p_i) \qquad s_{ji} \neq 0, \text{ for all } i.$$

Because there are distinct parameter values for the statistical fuzzy set function along each feature dimension, an input pattern has a different fuzzy SSD along each feature dimension. The fuzzy SSD to the jth fuzzy cluster for an input pattern is a function of all its degrees of fuzzy set function s_{ji} along each feature dimension. The fuzzy SSD equals to zero if any s_{ji} equals zero in the ith dimension. Otherwise, the fuzzy SSD will be the average of all its s_{ji} degrees along every feature dimension.

The statistical fuzzy set function of a cluster represents the statistical characteristic of patterns located inside the cluster's hyper-box core. Its output value, the fuzzy SSD, exhibits the relation between an input pattern and the cluster's prototype. From the definition of statistical fuzzy set function, only the cluster's central point has a full fuzzy SSD. The fuzzy SSDs for other patterns decrease when they are separated from the central point of the cluster.

3.4 Pattern Similarity Selection

The objective for defining the statistical fuzzy set function in the F3MCNN model is to solve the problem of full membership ambiguity that occurs in overlapped hyper-boxes. The fuzzy ART and fuzzy Min-Max models only use the fuzzy hyper-box similarity degree in pattern matching and learning. Fuzzy ART applies the order of committed clusters as the solution to the problem of full membership ambiguity. Exhaustive work in the hyper-box overlap checking and contraction is employed in fuzzy Min-Max to eliminate any overlapped hyper-boxes. The statistical fuzzy set function is introduced into the F3MCNN model to solve the problem of full membership ambiguity in overlapped hyper-boxes. The statistical fuzzy set function calculates fuzzy SSDs for the input pattern to those overlapped hyper-boxes when the input pattern is located inside the overlapped hyper-boxes area. As a result, the cluster with the maximum fuzzy SSD is selected as the candidate cluster to match and learn the input pattern. In this section, we explain the pattern similarity selection procedure for selecting a candidate cluster to learn an input pattern in detail. We then use an example to illustrate the different clustering results of the fuzzy ART, fuzzy Min-Max neural network, and F3MCNN model, respectively.

3.4.1 Pattern Similarity Selection Procedure

The pattern similarity selection procedure of the F3MCNN model is a two-step process for selecting a candidate cluster to match and learn the input pattern. When an input pattern is presented, the first step is to calculate its fuzzy HSD to every cluster. If there is only one cluster with the maximum fuzzy HSD then the pattern similarity selection procedure stops and this cluster will be the candidate cluster. Otherwise, if there are two or more clusters that have the same maximum fuzzy HSD, the problem of full membership ambiguity occurs. The second step of the pattern similarity selection procedure is carried out to calculate the fuzzy SSD of the input pattern to those clusters having the same maximum fuzzy HSD. The cluster with the maximum fuzzy SSD is selected as the candidate cluster to learn the input pattern; finally, the pattern similarity selection procedure stops.

3.4.2 Pattern Similarity Selection Example

In this section, we present an example to illustrate the different clustering results among fuzzy ART, fuzzy Min-Max, and F3MCNN. A set of two-dimensional input patterns contains five input patterns including $P_1 = (0.2, 0.7)$, $P_2 = (0.6, 0.3)$, $P_3 = (0.8, 0.2)$, $P_4 = (0.4, 0.2)$, and $P_5 = (0.8, 0.5)$ is presented to each model for clustering in the order of their subscripts.

Let us define the maximum hyper-box size allowed as 0.4. All these three models have the same hyper-box learning process in their clustering algorithms (see Figure 3.4).

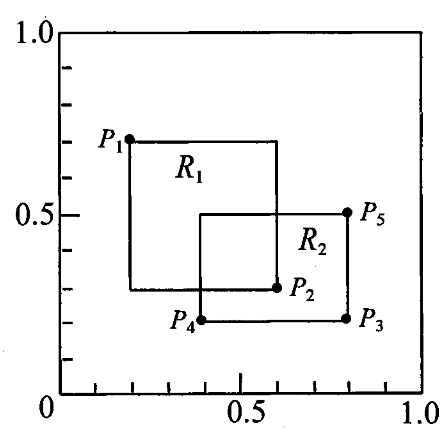


Figure 3.4 Clustering results after one pass of input patterns of fuzzy ART, fuzzy Min-Max, and F3MCNN

When the first pattern P_1 is presented, the first cluster R_1 is created and it contains only the pattern P_1. Then, the second pattern P_2 is presented and R_1 is expanded to include P_2 since the size of R_1 is not great than 0.4 after expansion. When the third pattern P_3 is presented, the second cluster R_2 is created to include P_3. Because, the cluster R_1 cannot be expanded to include P_3 due to the limitation for the maximum size of hyper-box. When the fourth pattern P_4 is presented, R_2 is expanded to include P_4, since the size of R_2 is not great than 0.4 after expansion. Finally, the last pattern P_5 is presented and R_2 is expanded to include P_5 because the size of R_2 is still not greater than 0.4 after expansion. Therefore, after one pass of the ordered input patterns, all three models learn input patterns and classify them into two clusters: R_1 and R_2.

For these three models, the clustering stability is defined as there being no hyper-box change in successive presentations of the same data set. Therefore, only one pass of the input patterns is needed to achieve clustering stability in fuzzy ART and F3MCNN.

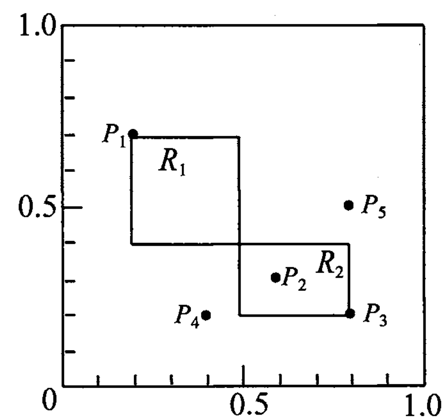


Figure 3.5 Clustering result after one pass of input patterns and overlapped hyper-boxes contraction of fuzzy Min-Max

The hyper-box overlapping is not allowed in fuzzy Min-Max. Since hyper-boxes R_1 and R_2 are overlapped in the clustering result of fuzzy Min-Max clustering, the hyper-box contraction process needs to be performed to eliminate the hyper-box overlapped area. The hyper-box overlapped area has its lower left-hand and higher right-hand corner points as (0.4, 0.3) and (0.6, 0.5) respectively (see Figure 3.4). The hyper-box contraction process takes the middle point (0.5, 0.4) of the hyper-box overlapped area as the place to separate clusters R_1 and R_2. The clustering result of fuzzy Min-Max becomes different from clustering results of the fuzzy ART and F3MCNN models after hyper-boxes contraction (see Figure 3.5).

Moreover, successive presentations of input patterns in the same order are necessary for the fuzzy Min-Max system to achieve clustering stability. This is because the clustering stability in the fuzzy Min-Max model is defined as there being no hyper-box change during successive presentations of input patterns in the same order. In this

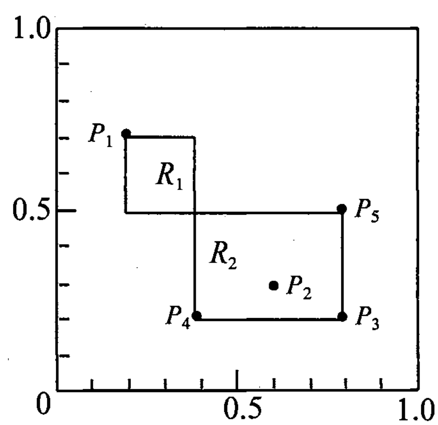


Figure 3.6 Clustering result after successively presenting input patterns to achieve cluster stability in fuzzy Min-Max

example, patterns P_4 and P_5 are not included in any hyper-boxes after one pass of patterns in fuzzy Min-Max clustering (see Figure 3.5). After the second pass of patterns, there are changes in the size of hyper-boxes. The hyper-box of cluster R_1 shrinks and the hyper-box of cluster R_2 expands. We apply successive passes of the same input until there is no hyper-box change in the clustering results of fuzzy Min-Max. Then, the fuzzy Min-Max system achieves clustering stability and its clustering result is shown in Figure 3.6.

The problem of full membership ambiguity is not solved in fuzzy ART nor in fuzzy Min-Max. Let us use an example to show the unsolved ambiguity problem. When a new input pattern $P_6 = (0.4, 0.5)$ is presented (see Figures 3.7, 3.8, and 3.9) to the stable clustering result in each model, its membership degree is determined in different ways in the fuzzy ART and fuzzy Min-Max systems.

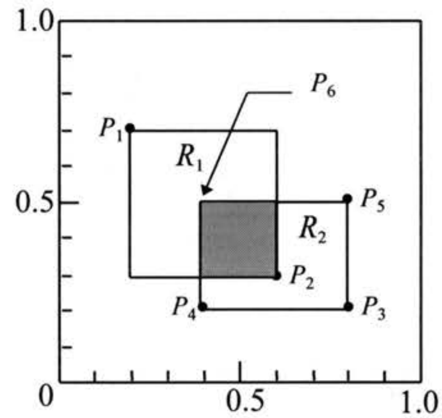


Figure 3.7 Clustering result after the pattern P_6 is presented for the clustering example in fuzzy ART

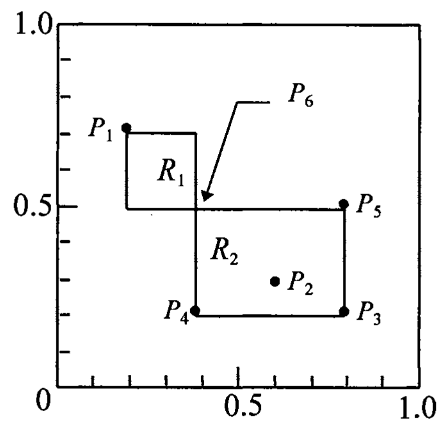


Figure 3.8 Clustering result after the pattern P_6 is presented for the clustering example in fuzzy Min-Max

First, we perform fuzzy ART pattern clustering and use the category choice function in Equation (3.3) to calculate the hyper-box similarity degrees for the input pattern P_6 to clusters R_1 and R_2 (see Figure 3.7) respectively. The fuzzy HSDs of the input pattern P_6 to the clusters R_1 and R_2 are equal, $T_1(P_6) = T_2(P_6) = 1$. As shown in Figure 3.7, the input pattern P_6 is located inside the hyper-box overlapped area (the gray rectangle). The new input pattern P_6 is located in the corner of the gray rectangle. As shown in this example, the input pattern P_6 has full membership degrees to both clusters R_1 and R_2 in the clustering results of the fuzzy ART model. To solve the problem of full membership ambiguity, the index associated with the created order of clusters is used in fuzzy ART to select the cluster with the smallest index as the candidate to match the input pattern. Therefore in this example, the input pattern P_6 is classified as a pattern in the cluster R_1. However, we cannot conceive of any reason for this nor find any explanation in [Carpenter 91].

Next, we perform fuzzy Min-Max pattern clustering and use the fuzzy set hyper-box membership function from Equation (3.7) to calculate the hyper-box membership values for the input pattern P_6 to clusters R_1 and R_2 (see Figure 3.8) respectively. The hyper-box membership values for the input pattern P_6 to clusters R_1 and R_2 are equal, $b_1(P_6) = b_2(P_6) = 1$. Figure 3.8 shows the relative position of the input pattern P_6 to clusters R_1 and R_2 in the fuzzy Min-Max clustering result. P_6 is exactly the abutting point between the cluster R_1 and the cluster R_2. This is the point that fuzzy Min-Max systems allow to have full membership in more than one cluster. Although the fuzzy Min-Max neural network utilizes a contraction process to eliminate any hyper-box overlap, the full membership ambiguity still occurs along the boundary between two abutting hyper-boxes. Therefore, the problem of full membership ambiguity is still not solved in the fuzzy Min-Max model.

In F3MCNN, the problem of full membership ambiguity is completely solved by introducing the fuzzy SSD into the pattern selection procedure. In this example, not only the hyper-boxes learned concept but also the statistical characteristic information are encoded into the F3MCNN system. The minimum and maximum points for each cluster are adjusted during hyper-box learning. They are $U_1 = (0.2, 0.3)$ and $V_1 = (0.6, 0.7)$ for hyper-box R_1 and $U_2 = (0.4, 0.2)$ and $V_2 = (0.8, 0.5)$ for hyper-box R_2. In the mean time, the fuzzy central point, the fuzzy variation vector, and the fuzzy number of patterns of each cluster are adapted during fuzzy statistical learning. They are $C_1 = (0.4, 0.5)$,

$A_1 = (0.2, 0.2)$, and $N_1 = 2$ for hyper-box R_1 and $C_2 = (0.67, 0.3)$, $A_2 = (0.2, 0.1)$, and $N_2 = 3$ for hyper-box R_2. We show the hyper-boxes and fuzzy set functions along each dimension of the feature space for clusters R_1 and R_2 in Figure 3.9.

The pattern similarity selection procedure for classifying the new input pattern P_6 in the example is explained as follows. In the first step, we calculate the fuzzy HSDs of the new input pattern P_6 to clusters R_1 and R_2 from Equation (3.9). Because P_6 is located in the hyper-box overlapped area, it has same fuzzy HSD to clusters R_1 and R_2, i.e.,

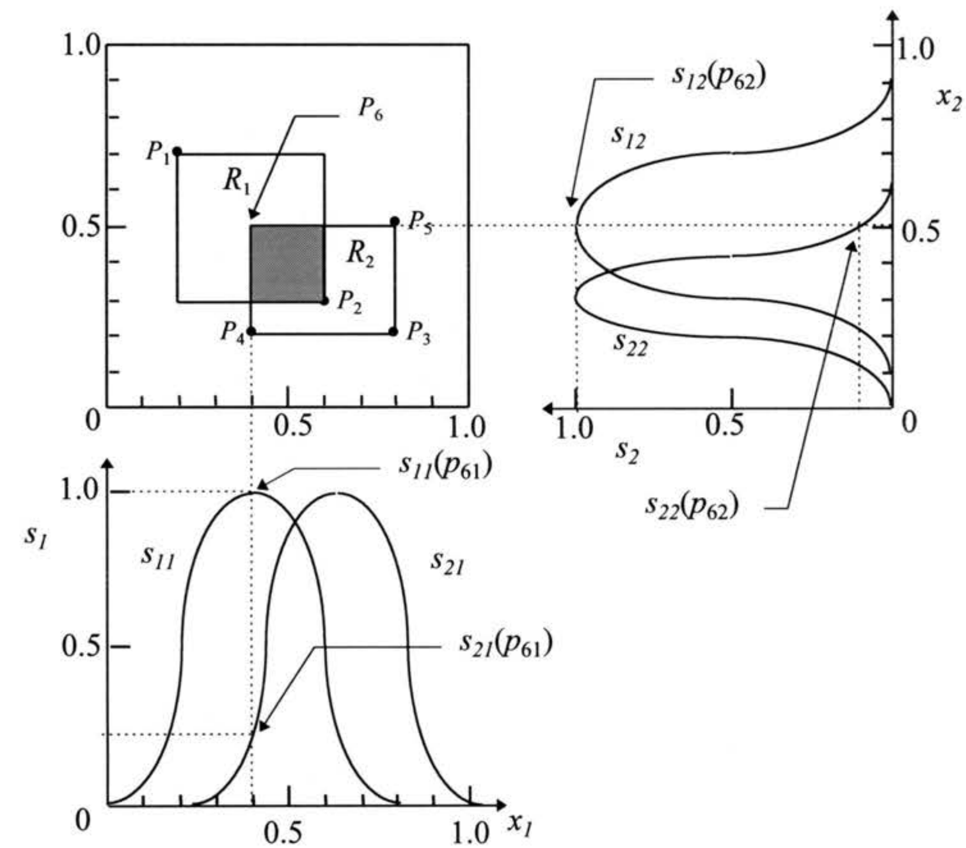


Figure 3.9 Clustering result after the pattern P_6 is presented for the clustering example in F3MCNN. P_6 has the same fuzzy hyper-box similarity degree to R_1 and R_2; but, P_6 has different fuzzy statistical similarity degree to R_1 and R_2.

$H_1(P_6) = H_1(P_6) = 1$. As a result, the problem of full hyper-box membership ambiguity is faced. Therefore, we need to perform the second step to solve the problem of full membership ambiguity in hyper-box clustering. We use the statistical fuzzy set function defined in Equation (3.18) to calculate the fuzzy SSDs of P_6 to clusters R_1 and R_2. From Equation (3.17), the fuzzy SSDs of P_6 to cluster R_1 along feature dimensions x_1 and x_2 are $s_{11}(p_{61}) = 1.0$ and $s_{12}(p_{62}) = 1.0$, respectively. The overall fuzzy SSD calculated from Equation (3.18) of P_6 to the cluster R_1 is $S_1(P_6) = 1$. The fuzzy SSDs of P_6 to cluster R_2 along feature dimensions x_1 and x_2 are $s_{21}(p_{61}) = 0.222$ and $s_{22}(p_{62}) = 0.125$, respectively. The overall fuzzy SSD calculated from Equation (3.18) of P_6 to the cluster R_2 is $S_2(P_6) = 0.173$. Since $S_1(P_6)$ is greater than $S_2(P_6)$, we classify the input pattern P_6 as a pattern of the cluster R_1.

3.4.3 Summary

From the pattern similarity selecting example demonstrated above, we summarize the differences among fuzzy ART, fuzzy Min-Max, and F3MCNN. We are concerned with two major points: learning stability and full membership ambiguity.

Both the fuzzy ART model and the F3MCNN model achieve clustering stability in one pass of patterns. From the stable category learning defined in [Carpenter 91], a fuzzy ART system with complement coding, conservative limits, and fast learning forms stable hyper-box categories in one pass of input patterns. In subsequent presentations of any input, no reset or additional learning occurs. In hyper-box learning, the F3MCNN system is identical to a fast learning fuzzy ART system. The stable hyper-box clusters are constructed within one pass of input patterns. In these two models, hyper-boxes are only allowed to expand. While in a fuzzy Min-Max system, hyper-boxes are expanded to learn input patterns and are contracted to eliminate overlaps. Therefore, a fuzzy Min-Max system needs more than one pass of input patterns in order to achieve clustering stability. Also, an error parameter must be defined for the fuzzy Min-Max model to achieve clustering stability.

Only the F3MCNN model completely solves the problem of full membership ambiguity. In the F3MCNN model, patterns may have the same partial membership to more than one cluster but they will never have full membership to more than one cluster. We introduce the statistical fuzzy set function into the F3MCNN model to solve the problem of full hyper-box membership ambiguity. The pattern's membership degree to a cluster is determined by the pattern similarity selection procedure defined in section 3.4.1. Both fuzzy HSD and fuzzy SSD are considered in the pattern similarity selection procedure. In the fuzzy ART and fuzzy Min-Max models, only the fuzzy HSD is used to select a candidate cluster. The order of committed clusters is utilized to resolve the problem of full membership ambiguity in the fuzzy ART system. In the example shown in Figure 3.7, patterns located inside the hyper-box overlapped area are classified as patterns in the cluster R_1 because R_1 has the smaller index. This regulation rule does solve the problem of full membership ambiguity but results in inaccuracy in its clustering result. In the fuzzy Min-Max system, hyper-boxes overlap checking and contraction processes are used to try to solve the problem of full membership ambiguity.

Nevertheless, patterns along the boundary between two abutting hyper-boxes still have full membership in more than one cluster.

CHAPTER IV

SYSTEM ARCHITECTURE AND CLUSTERING ALGORITHM

4.1 Introduction

In this Chapter, we introduce the system architecture and the clustering algorithm of the F3MCNN model. F3MCNN is a synergetic model of fuzzy set theory [Zadeh 65] and a modified adaptive resonance theory neural network [Grossberg 87]. We define two fuzzy set functions based on the theory of fuzzy sets to calculate the degree of similarity of input patterns to clusters. Meanwhile, the F3MCNN model has the ART-like architecture to learn input patterns and create clusters dynamically. The F3MCNN model resembles both fuzzy ART [Carpenter 91] and fuzzy Min-Max [Simpson 93], since the minimum and maximum points hyper-box methodology also is used as part of the cluster representation in our model. However, the F3MCNN model has a significant difference from those two systems. The clustering algorithm in F3MCNN is a two-phase clustering and learning procedure. We utilize both the learned concept and statistical characteristics of pattern clusters in F3MCNN for pattern clustering. While in the fuzzy ART and fuzzy Min-Max models, only the learned concept of pattern clusters is used for clustering. We expect that the introduction of the statistical characteristics of pattern clusters will improve the performance and clustering results of our model.

The contents of this Chapter are organized as follows. In section 4.2, we introduce the system architecture of the F3MCNN model, which contains the attentional subsystem and the selection control subsystem. We will explain the functional purpose for each neural node layer and synapse connection between neural node layers in the attentional subsystem. Also, the cluster selecting and matching control performed by the selection control subsystem is introduced. In section 4.3, we introduce the clustering algorithm that performs a two-phase pattern clustering in F3MCNN: hyper-box clustering in the first phase and statistical clustering in the second phase. Finally, choice functions, expansion criteria, and learning rules in both phases are explained.

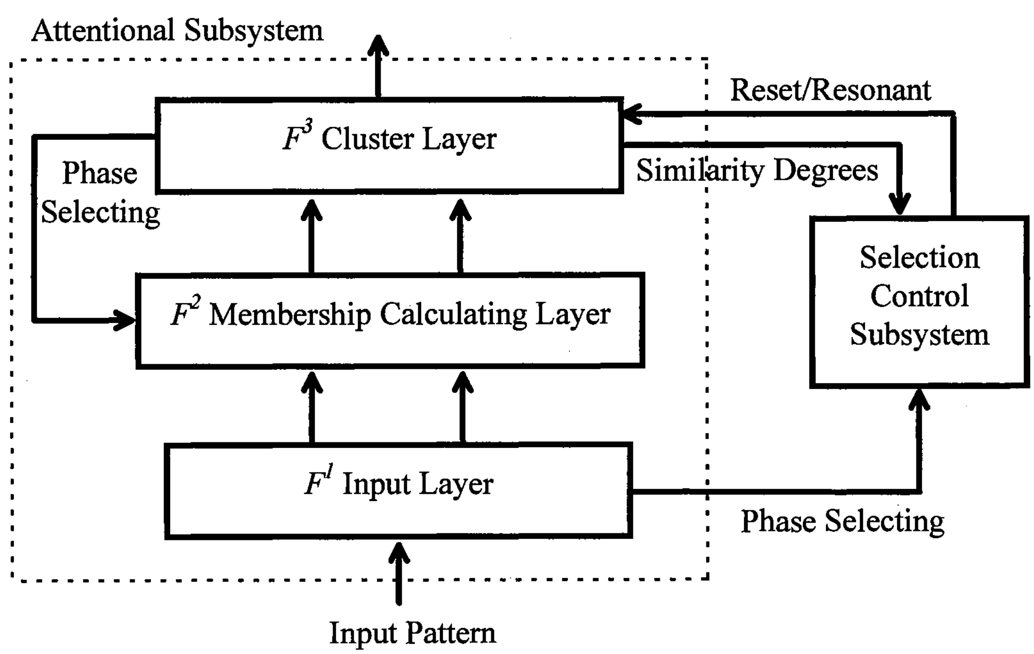


Figure 4.1 System architecture of F3MCNN

4.2 System Architecture

The F3MCNN system architecture is illustrated in Figure 4.1. The F3MCNN module includes two major parts: the attentional subsystem and the selection control subsystem. The attentional subsystem is a modified ART-type neural network. It is a three-layer network and is responsible for: (1) encoding clusters' attributes, (2) calculating the fuzzy similarity degrees to clusters of an input pattern including the fuzzy HSD and the fuzzy SSD, (3) creating new cluster nodes and new sub-layer membership calculation nodes, and (4) interacting with the selection control subsystem to reset or resonant cluster nodes. The selection control subsystem interacts with the attentional subsystem to carry out internal cluster searching and matching processes and issues a control signal to change the operation phase of the F3MCNN system.

The network of an F3MCNN system operates in two different phases. In the first phase, the attentional subsystem calculates the fuzzy HSDs of an input pattern to every cluster. Then, the selection control subsystem searches for the cluster with the maximum fuzzy HSD to learn the input pattern. When there are two or more clusters that have the same maximum fuzzy HSD, the selection control subsystem issues a control signal to change the F3MCNN system to operate in the second phase. In the second phase, the attentional subsystem calculates the fuzzy SSDs of an input pattern to those clusters that have the same maximum fuzzy HSD. Then, the selection control subsystem searches the cluster with the maximum fuzzy SSD to learn the input pattern. A detailed description of

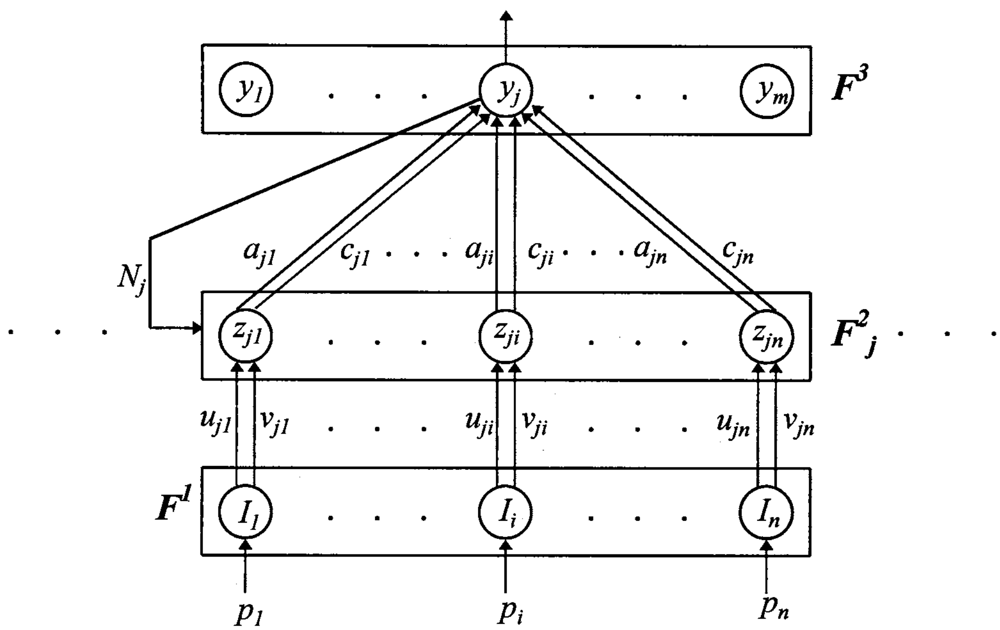


Figure 4.2 The attentional subsystem of F3MCNN

the system architecture and the different function corresponding to each phase of these two subsystems will be discussed next.

4.2.1 Attentional Subsystem

The attentional subsystem of F3MCNN is a modified ART-type neural network. It is a three-layer neural network (see Figure 4.2). Neurons in each layer receive information transmitted from neurons in other layers, perform a function, and then generate an output information. The neuron layers contain: (1) the input representation layer F^1 at the bottom, (2) the membership calculating layer F^2 in the middle, and (3) the cluster representation layer F^3 on the top. Between neuron layers, there are synaptic connections to take care of information transmission and encoding. The synaptic connections between neurons include: (1) the bottom-up connection vectors from F^1 to F^2 to encode the hyper-box learned concept and transmit input patterns, (2) the bottom-up connection vectors from F^2 to F^3 to encode clusters' statistical characteristics and transmit calculated similarity degrees, and (3) the top-down connection vectors to encode the number of patterns inside the hyper-box and initiate calculation of the fuzzy SSD.

4.2.1.1 Activation Function of Neuron Layers The neurons in F^1 perform a linear activation function; in contrast, the neurons in F^2 and F^3 perform two different activation functions when the F3MCNN system operates in different phases. In our model, we scale n-dimensional input patterns from $\Re^n$ to I^n where n is the dimension of the feature vector space. We denote the input pattern as $P = (p_1, p_2, \ldots, p_n)$, where $0 \leq p_i \leq 1$ for $i = 1, \ldots, n$. The input layer F^1 contains n nodes; each node encodes a feature of the input pattern. The activation function of nodes in the input layer F^1 is a linear function; the output of the activation function for the input layer is the scaled input pattern in both phases.

A neuron in a sub-layer of F^2 receives only one feature of the input pattern from F^1 and calculates its similarity degree in this particular feature dimension. A cluster node in F^3 receives the similarity degrees from its corresponding sub-layer in F^2 and calculates the overall similarity degree of the input pattern. When the F3MCNN system operates in the first phase, neurons in F^2_j and F^3 calculate the fuzzy HSDs of an input pattern to clusters. When the selection control subsystem issues a control signal to make the F3MCNN system to operate in the second phase, neurons in F^2 and F^3 calculate the fuzzy SSDs of an input pattern to clusters in the second phase.

The membership calculating layer F^2 contains one n-node sub-layer for each committed cluster; each node inside the n-node sub-layer calculates the fuzzy HSD or fuzzy SSD of an input pattern in each feature dimension. Therefore, the activation function of nodes in a sub-layer in F^2 will be the d-function defined in Equation (3.10) when the F3MCNN system operates in the first phase, or the s-function defined in Equation (3.17) when the F3MCNN system operates in the second phase. We denote the output vector of the jth sub-layer in membership calculating layer F^2_j as $Z_j = (z_{j1}, z_{j2}, \ldots, z_{jn})$. When the F3MCNN system is operating in the first phase, z_{ji} represents the output of the distance function d_{ji}. Otherwise, z_{ji} represents the output of the statistical fuzzy set function s_{ji}.

The cluster representation layer F^3 contains one node for each committed cluster; and each node calculates the fuzzy HSD or the fuzzy SSD of an input pattern to the corresponding cluster. When the F3MCNN system is operating in the first phase, the S-function defined in Equation (3.18) will be the activation function for the cluster node in F^3. Otherwise, the H-function defined in Equation (3.9) will be the activation function for the cluster node in F^3 in the second phase. We denote the output vector of the cluster layer F^3 as $Y = (y_1, y_2, \ldots, y_m)$, where m is the number of committed clusters. When the network is operating in the first phase, y_j represents the output of the hyper-box fuzzy set function H_j. Otherwise, y_j represents the output of the statistical fuzzy set function S_j.

4.2.1.2 Synaptic Connection Vectors The synaptic connection vectors encode the attribute of clusters and transmit the output of the activation functions from one neuron layer to another. Two bottom-up synaptic connection vectors, U_j and V_j, connect from nodes in F^1 to nodes in the jth sub-layer in F^2. We denote the bottom-up synaptic connection vectors as $U_j = (u_{j1}, u_{j2}, \ldots, u_{jn})$ and $V_j = (v_{j1}, v_{j2}, \ldots, v_{jn})$. U_j and V_j are used to transmit the scaled input pattern from F^1 to F^2 in both operation phases of an F3MCNN system. We use the synaptic connection vectors U_j and V_j to encode the minimum point and the maximum point of the hyper-box of the jth cluster respectively. Elements of the synaptic connection vectors U_j and V_j, u_{ji} and v_{ji}, emit from the ith node in F^1 to the ith node in the jth sub-layer in F^2. When a new cluster j is created during pattern learning, it contains only the new input pattern. Therefore, its minimum point vector (U_j) and the maximum point vector (V_j) are the same as the new input pattern. During pattern clustering, the bottom-up synaptic connection vectors, U_j and V_j, are updated according to the fuzzy hyper-box learning rule introduced in Section 4.3.

There are two bottom-up connection vectors, C_j and A_j, from the jth sub-layer in F^2 to the jth node in F^3. We denote the bottom-up connection vectors from the jth sub-layer F^2 to the jth node in F^3 as $C_j = (c_{j1}, c_{j2}, \ldots, c_{jn})$ and $A_j = (a_{j1}, a_{j2}, \ldots, a_{jn})$. When an F3MCNN system is operating in the first phase, these bottom-up connections are used to transmit the fuzzy HSDs of an input pattern to the jth cluster along each feature dimension. When the F3MCNN system is operating in the second phase, they are used to transmit the fuzzy SSDs of an input pattern to the jth cluster along each feature dimension. We also use the synaptic connection vectors, C_j and A_j, to encode the fuzzy central point and the fuzzy variation vector for the jth cluster. When a new cluster j is created during pattern learning, it contains only the new input pattern. Therefore, its fuzzy central point C_j is same as the new input pattern and its fuzzy variation vector A_j is a zero vector. During pattern clustering, the bottom-up synaptic connection vectors, C_j and A_j, are updated according to the fuzzy statistical learning rule introduced in Section 4.3.

The top-down connection N_j from the jth node in F^3 to the jth sub-layer in F^2 encodes the fuzzy number of pattern in the jth cluster, i.e., the number of patterns located inside the hyper-box of the jth cluster. It is also used to transmit an activation signal whenever an F3MCNN system is operating in the second phase. When a new cluster j is created during pattern learning, it contains only the new input pattern. Therefore, its fuzzy number of patterns N_j is initialized as one. It is updated during learning.

4.2.2 Selection Control Subsystem

The selection control subsystem is responsible for selecting a candidate cluster for an input pattern and checking the expansion criterion for the selected cluster during pattern learning. It works differently when an F3MCNN system is operating in different phases. The selection control subsystem receives the similarity degrees from the attentional subsystem then invokes a search procedure to discover the cluster with the maximum similarity degree. Then, a matching function is also invoked to check if the selected cluster satisfies the expansion criterion.

When the F3MCNN system is operating in the first phase, the selection control subsystem selects the cluster to which the input pattern has the maximum fuzzy HSD. If there is only one cluster with the maximum fuzzy HSD, then it invokes the matching function to check whether the expanded hyper-box size of the selected cluster exceeds the maximum size or not. If the selected cluster satisfies the expansion criterion, the selection control subsystem issues a resonant command to adapt the input pattern to the selected cluster. Otherwise, the selection control subsystem resets the selected cluster.

If there are two or more clusters have the same maximum fuzzy HSD, the selection control subsystem issues a control signal to change the operation phase of the F3MCNN system. The F3MCNN system changes from the first phase to the second phase in order to calculate fuzzy SSDs of the input pattern to solve the tie. The selection control subsystem selects the cluster with the maximum fuzzy SSD of the input pattern calculated by the attentional subsystem. After the cluster selection, it returns the index of the selected cluster for fuzzy statistical learning.

4.3 F3MCNN Clustering Algorithm

The clustering algorithm of the F3MCNN model is a two-phase procedure that includes fuzzy hyper-box clustering and fuzzy statistical clustering. The learning process in the F3MCNN model includes fuzzy hyper-box learning and fuzzy statistical learning. Fuzzy hyper-box learning adapts a pattern to the learned concept of the selected cluster, and fuzzy statistical learning adapts a pattern into the statistical characteristic of the selected cluster. When an input pattern is presented, the F3MCNN system operates in the first phase to invoke fuzzy hyper-box clustering. The fuzzy hyper-box clustering process begins by selecting the cluster with the maximum fuzzy HSD to the input pattern that can be expanded (if necessary) to include the input pattern. When there are two or more clusters have the same maximum fuzzy HSD, the F3MCNN system changes its operation phase to the second phase to invoke fuzzy statistical clustering in order to solve the tie. The fuzzy statistical clustering process selects the cluster with the maximum fuzzy SSD from those clusters with the same maximum fuzzy HSD. If the expansion is necessary and the selected cluster meets the expansion criterion, the F3MCNN system invokes fuzzy statistical learning and fuzzy hyper-box learning. Otherwise, only fuzzy statistical learning takes place since the input pattern is already inside the hyper-box. If there is no cluster can be found to meet the expansion criterion, a new cluster is formed and added to the F3MCNN system.

A user-defined parameter is used to limit the maximum size of hyper-boxes. During fuzzy hyper-box learning, hyper-boxes may grow to the user-defined maximum size. In an F3MCNN system, we use the fast learning strategy in fuzzy hyper-box learning, i.e., a hyper-box is expanded to include the input pattern during fuzzy hyper-box learning. Therefore, the vigilance parameter $\rho \in [0, 1]$ determines the maximum size for the hyper-box of every cluster. Users are allowed to decide the vigilance parameter for their applications. However, a large ρ will result in fewer clusters but bigger sizes of clusters in the clustering result.

We define cluster stability of the F3MCNN model as there being no change in the size of hyper-boxes for successive presentations of the same input patterns for clustering. The F3MCNN clustering procedure has the advantage of fast attainment of cluster stability. After one pass of all input patterns, the learning procedure achieves cluster stability. This clustering procedure allows existing clusters to grow in order to adapt new patterns, and it allows new clusters to be added without retraining. We explain the clustering algorithm in the following steps and show its flow chart in Figure 4.3.

Step 1: Set up the user-defined vigilance parameters ρ.

Step 2: Read in the first input pattern then set up the first cluster node.

Step 3: Read in next input pattern.

If no more input patterns, then stop. Otherwise, go to next step.

Step 4: Use the hyper-box fuzzy set function to calculate the fuzzy HSD of the input pattern to every committed cluster.

Step 5: Perform the fuzzy hyper-box choice function to select the candidate cluster with the maximum fuzzy HSD.

If there is no cluster with the fuzzy HSD > 0, then use the new input pattern to create a new cluster and go to Step 3.

Otherwise, go to Step 6.

Step 6: If there are more than one candidate clusters with the same maximum fuzzy HSD, then go to Step 9.

Otherwise, go to next step.

Step 7: If the maximum fuzzy HSD = 1, then perform fuzzy statistical learning for the selected cluster and go to Step 3.

Otherwise, go to next step.

Step 8: If the candidate cluster meets the fuzzy hyper-box expansion criterion, then perform fuzzy hyper-box learning and fuzzy statistical learning to adjust the candidate cluster, then go to Step 3.

Otherwise, use the new pattern to create a new cluster and go to Step 3.

Step 9: Use the statistical fuzzy set function to calculate the fuzzy SSD of the input pattern to those selected clusters with the same maximum fuzzy HSD.

Step 10: Perform fuzzy statistical choice function to select a candidate cluster with the maximum fuzzy SSD.

If the maximum fuzzy SSD = 0, then reset all candidate clusters with the same maximum HSD then go to Step 5.

Otherwise, go to next step.

Step 11: If the maximum fuzzy HSD = 1, then perform fuzzy statistical learning, then go to Step 3.

Otherwise, go to next step.

Step 12: If the candidate cluster meets the fuzzy hyper-box expansion criterion, then perform fuzzy hyper-box learning and fuzzy statistical learning to adjust the candidate cluster, then go to Step 3.

Otherwise, reset the selected cluster and go to Step 10.

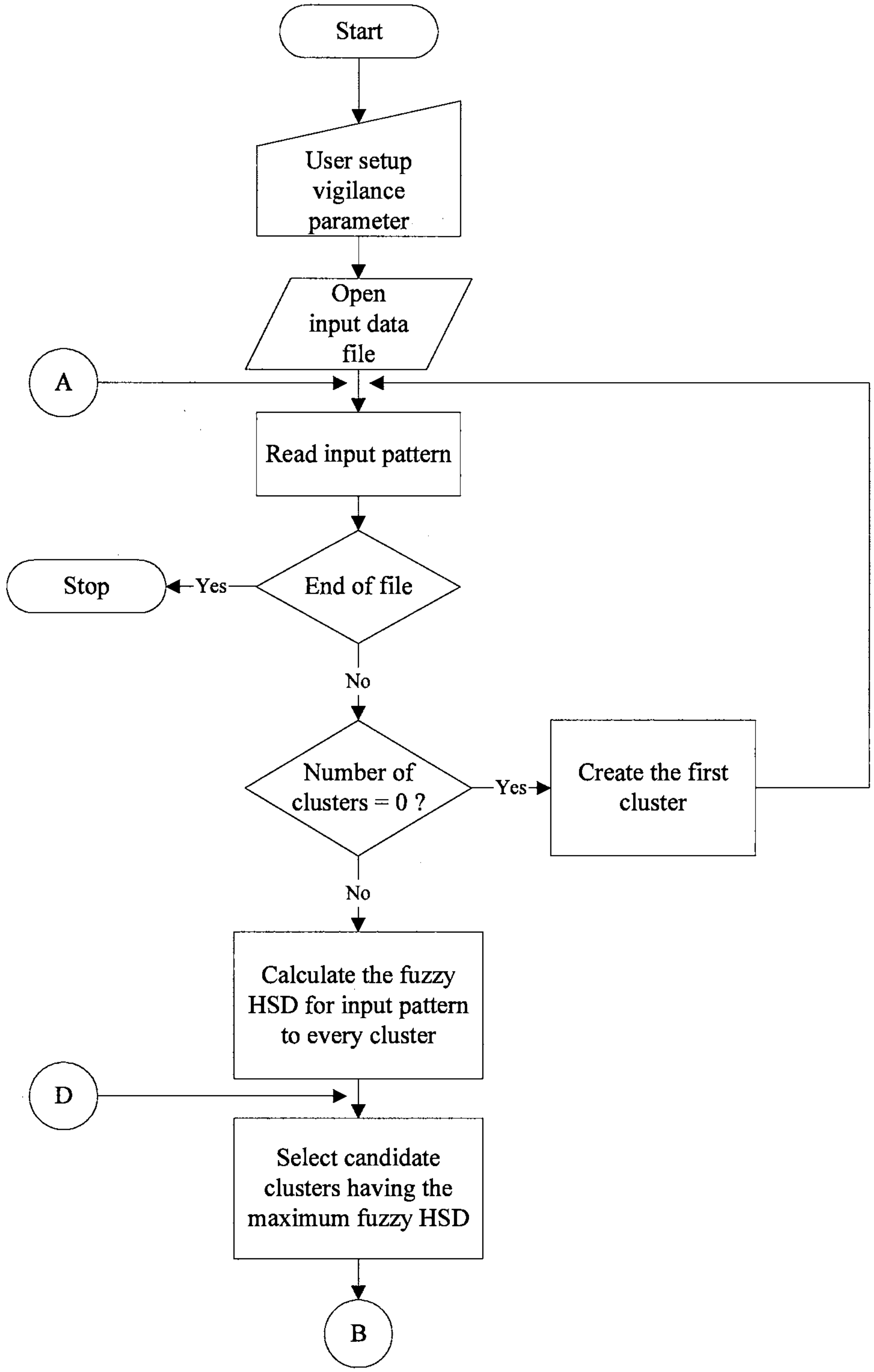


Figure 4.3 F3MCNN clustering algorithm flow chart

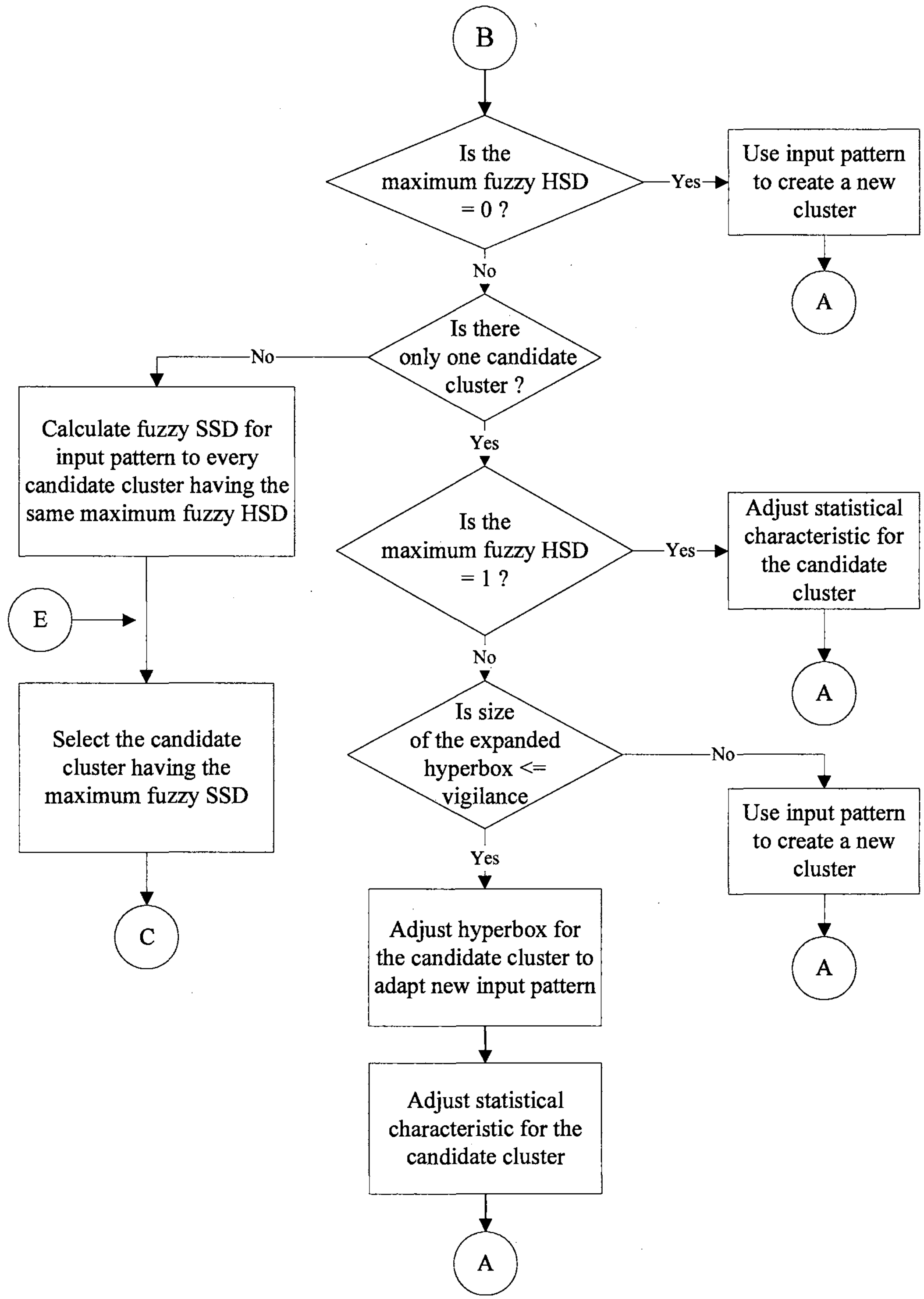


Figure 4.3 F3MCNN clustering algorithm flow chart (*continued*)

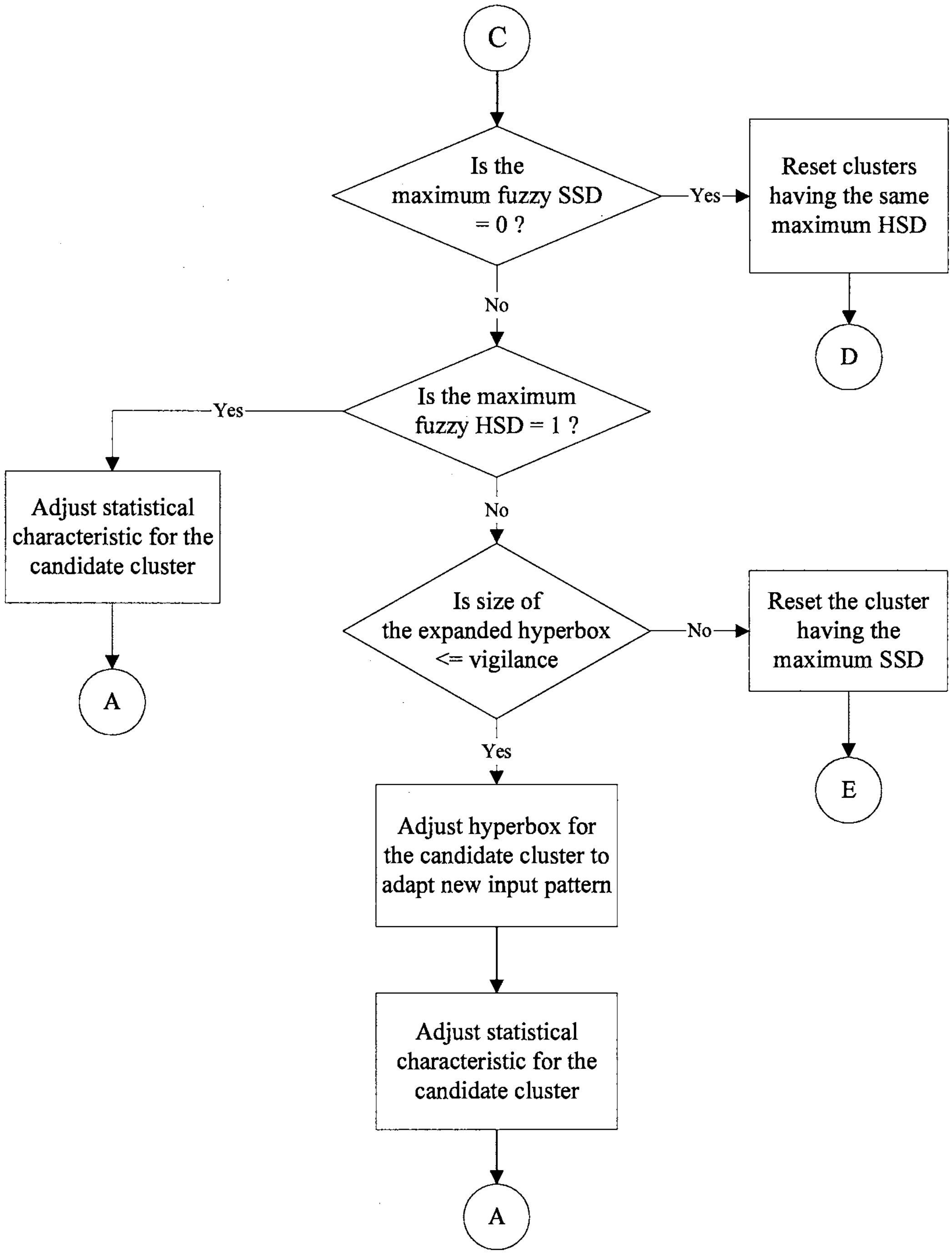


Figure 4.3 F3MCNN clustering algorithm flow chart (*continued*)

There are two sets of clusters in F^3 used in F3MCNN, the committed set and the uncommitted set. We also define two sets of sub-layers in F^2, the committed sub-layers set and the uncommitted sub-layers set. Before any learning has occurred, an F3MCNN system has an empty set of committed clusters and undetermined number of clusters in the uncommitted set. It also has an empty set of committed sub-layers and undetermined number of sub-layers in the uncommitted set. When the first input pattern is presented, we create the first sub-layer of nodes in F^2 and a new cluster node in F^3. The new cluster is represented by a single point in the feature vector space. Its fuzzy central point, maximum point, and minimum point are all equal to the input pattern and its fuzzy variation vector is a zero vector. Because the maximum point and the minimum point of the new cluster node are equal, the new cluster has zero support of its statistical fuzzy set function.

The processes in the F3MCNN clustering algorithm include fuzzy hyper-box clustering, fuzzy statistical clustering, fuzzy hyper-box learning, and fuzzy statistical learning are described in greater detail in the following subsections.

4.3.1 Fuzzy Hyper-box Clustering

Whenever an input pattern is presented, the fuzzy hyper-box clustering process takes place first in the F3MCNN clustering. In the fuzzy hyper-box clustering process, the attentional subsystem uses the hyper-box fuzzy set functions introduced in Equations (3.9) and (3.10) to calculate the fuzzy HSD of the input pattern to every cluster. Then the selection control subsystem invokes the defined fuzzy hyper-box choice function to select a candidate cluster with the maximum fuzzy HSD. In order to learn the new pattern, this selected cluster must meet the fuzzy hyper-box expansion criterion. We define the fuzzy hyper-box choice function and fuzzy hyper-box expansion criterion in the following subsections.

4.3.1.1 Fuzzy Hyper-box Choice Function For the input pattern P and cluster j, the fuzzy HSD is $H_j(P)$ calculated from Equation (3.9). We define the fuzzy hyper-box choice function as

$$H_J(P) = max\{H_j(P): j = 1, \ldots, m\}, \tag{4.1}$$

where m is the number of committed clusters. The fuzzy hyper-box choice function selects a candidate cluster with the maximum fuzzy HSD in fuzzy hyper-box clustering.

4.3.1.2 Fuzzy Hyper-box Expansion Criterion We set the user-defined vigilance parameter ρ as the threshold to guard the maximum hyper-box size for every cluster. The candidate cluster must satisfy the fuzzy hyper-box expansion criterion; that is,

$$H_J(P) \leq \rho \tag{4.2}$$

where the user-defined vigilance parameter ρ is in the interval [0, 1]. The hyper-box expansion criterion limits the maximum size of the expanded hyper-box. The selected cluster can be expanded to include the input pattern if it satisfies the fuzzy hyper-box expansion criterion. Otherwise, its fuzzy HSD is reset to zero and loses the competition for learning this particular pattern.

4.3.2 Fuzzy Statistical Clustering

The fuzzy statistical clustering in F3MCNN is used to solve the problem of hyper-box membership ambiguity. When this problem occurs, the selection control subsystem issues a control signal to change the operation phase of the F3MCNN system to the second phase. Soon afterward, the attentional subsystem uses the defined Equations (3.17) and (3.18) to calculate the fuzzy SSDs of the input pattern to those clusters selected in fuzzy hyper-box clustering. We define a fuzzy statistical choice function to select a cluster with the maximum fuzzy SSD to be the candidate cluster in fuzzy statistical clustering. If the input pattern is located inside the hyper-box of the selected cluster, only fuzzy statistical learning is performed without checking the fuzzy hyper-box expansion criterion. This is because the input pattern has full fuzzy HSD to the selected cluster and there is no need to expand the selected cluster's hyper-box. Otherwise, the selected cluster's hyper-box needs to be expanded to include the new input pattern. In this case, we need to check the fuzzy hyper-box expansion criterion for the selected cluster. If it satisfies the fuzzy hyper-box expansion criterion, fuzzy hyper-box learning is performed to expand the hyper-box of the selected cluster and fuzzy statistical learning also is performed to adjust the statistical characteristic of the selected cluster.

4.3.2.1 Fuzzy Statistical Choice Function In F3MCNN, the fuzzy SSD takes account of the statistical characteristic of patterns in the fuzzy clusters. The value of the statistical fuzzy set function of a cluster shows the degree in which the input pattern is related to the corresponding cluster's prototype. The statistical fuzzy set functions defined in Equations (3.17) and (3.18) are used as the fuzzy similarity function in fuzzy statistical clustering. For each input pattern, we calculate its fuzzy SSDs to candidate clusters in fuzzy hyper-box clustering. Then, we use the winner-take-all strategy to select the cluster with the largest fuzzy SSD as the candidate in fuzzy statistical clustering. Therefore, the fuzzy statistical choice function is defined as

$$S_J(P) = max\{S_j(P): j = 1, \ldots , m\} \tag{4.3}$$

and m is the number of the committed clusters. The input pattern will have the maximum fuzzy SSD to the selected cluster. At the same time, it will have the maximum fuzzy HSD because it is one of the clusters with the maximum degree from the fuzzy hyper-box clustering process.

4.3.2.2 <u>Fuzzy Statistical Expansion Criterion</u> We do not put any limit on fuzzy statistical clustering. However, the fuzzy hyper-box expansion criterion still needs to be contained in the fuzzy statistical clustering process. Fuzzy statistical clustering is only performed to solve the problem of hyper-box membership ambiguity that occurred in the fuzzy hyper-box clustering process.

4.3.3 <u>Fuzzy Hyper-box Learning</u>

When the selected cluster satisfies the fuzzy hyper-box expansion criterion, its hyper-box is allowed to expand to cover the input pattern if necessary. If the expansion is necessary, the new pattern is adapted to the learned concept of the selected cluster according to the fuzzy hyper-box learning rule defined as follows.

For the input pattern P and cluster j, the minimum point U_j of the hyper-box is adjusted using the equation

$$U_j = P \wedge U_j, \tag{4.4}$$

and the maximum point V_j is adjusted using the equation

$$V_j = P \vee V_j, \tag{4.5}$$

where the fuzzy AND operator $\wedge$ is defined as

$$(a \wedge b)_i = \min(a_i, b_i), \tag{4.6}$$

and the fuzzy OR operator $\vee$ is defined as

$$(a \vee b)_i = \max(a_i, b_i). \tag{4.7}$$

The fuzzy hyper-box learning rule, Equations (4.4) and (4.5), is fast learning; this means the selected cluster's hyper-box is expanded to include the input pattern at once (see Figure 4.4). The defined fuzzy hyper-box learning rule is based on the fast learning approach defined in [Carpenter 91].

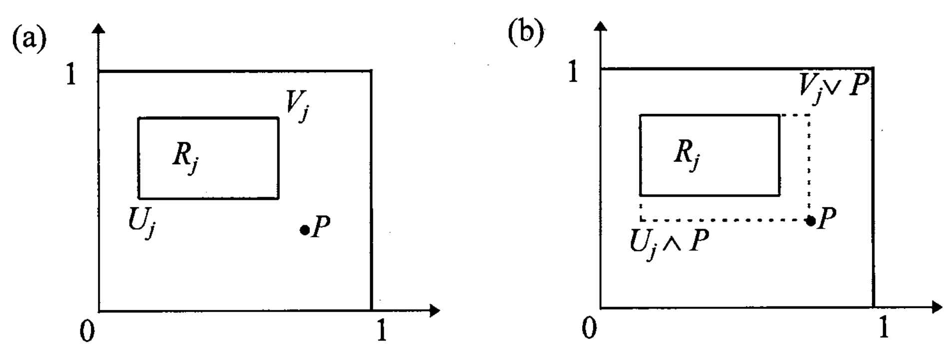


Figure 4.4 Fast hyper-box learning geometric representation in F3MCNN: (a) Before expansion, and (b) After expansion.

4.3.4 Fuzzy Statistical Learning

Fuzzy statistical learning takes place every time an input pattern is presented no matter whether the hyper-box of the selected cluster needs to be expanded or not. The rules of fuzzy statistical learning include learning for fuzzy central point, learning for fuzzy number of patterns, and learning for fuzzy variation vector. We describe them in greater detail as follows.

4.3.4.1 Learning for Fuzzy Central Point For the jth newly created cluster, we use the new input pattern to initialize its fuzzy central point. After the initiation, fuzzy cluster learning updates the fuzzy central point $C_j = (c_{j1}, c_{j2}, \ldots, c_{jn})$ according to the defined equation

$$c_{ji}^{(\text{new})} \equiv \frac{N_j^{(\text{old})} \cdot c_{ji}^{(\text{old})} + p_i}{N_j^{(\text{new})}}, \tag{4.8}$$

where N_j is the fuzzy number of patterns of the jth cluster, i denotes the ith feature dimension, $i \in \{1, 2, \ldots, n\}$, and $P = (p_1, p_2, \ldots, p_n)$ is the input pattern.

4.3.4.2 Learning for Fuzzy Number of Patterns In Equation (4.8), we use N_j as the denominator to update the fuzzy central point. We call N_j as fuzzy number of patterns and define it as the number of patterns that have been classified into the jth cluster. The initial value for N_j is one because a newly created cluster is exactly the same as the input pattern. When the input pattern is classified into the selected cluster j, the fuzzy number of patterns of cluster j is adjusted by using the defined equation

$$N_j^{(\text{new})} \equiv N_j^{(\text{old})} + 1. \tag{4.9}$$

4.3.4.3 Learning for Fuzzy Variation Vector The fuzzy variation vector reflects the spread of patterns that have been classified into a cluster. When the input pattern is classified into the jth cluster, we update the fuzzy variation vector $A_j = (a_{j1}, a_{j2}, \ldots, a_{jn})$ of the jth cluster according to the defined equation

$$a_{ji}^{(\text{new})} \equiv \frac{N_j^{(\text{old})} \cdot a_{ji}^{(\text{old})} + \left| p_i - c_{ji}^{(\text{new})} \right|}{N_j^{(\text{new})}}, \tag{4.10}$$

where N_j is the fuzzy number of patterns of the jth cluster, i denotes the ith feature dimension, $i \in \{1, 2, \ldots, n\}$, and $P = (p_1, p_2, \ldots, p_n)$ is the input pattern. The $|p_j - c_{ji}|$ represents the absolute distance between the point p_j and c_{ji}.

4.3.5 Learning in an F3MCNN System

An F3MCNN system learns new patterns by adapting them into the learned concept and the statistical characteristics of clusters. Whenever an input pattern is presented, a cluster is selected from either the fuzzy hyper-box clustering process or the fuzzy statistical clustering process to adapt the new pattern. There are two cases where the selected cluster satisfies the fuzzy clustering expansion criterion. In the first case, the input pattern, x, is located inside the hyper-box of the selected cluster (see Figure 4.5(a)). While in the second case, the input pattern, x, is located outside the hyper-box of the selected cluster (see Figure 4.5(b)). The fuzzy cluster learning in Figure 4.5(a) is simple and straightforward. Because the input pattern is located inside the cluster's hyper-box, it will satisfy the fuzzy hyper-box expansion criterion automatically. Therefore, learning for the selected cluster includes only fuzzy statistical learning to adjust its fuzzy central point, fuzzy variation vector, and fuzzy number of patterns.

However, fuzzy cluster learning in Figure 4.5(b) is much more complicated. We need to introduce the hyper-box expansion criterion to check if the size of the expanded hyper-box is smaller than or equal to the user-defined vigilance parameter ρ. If it meets the hyper-box expansion criterion, we need to perform both fuzzy hyper-box learning and fuzzy statistical learning. Otherwise, we reset the candidate cluster and search the next candidate cluster from the rest of the committed clusters.

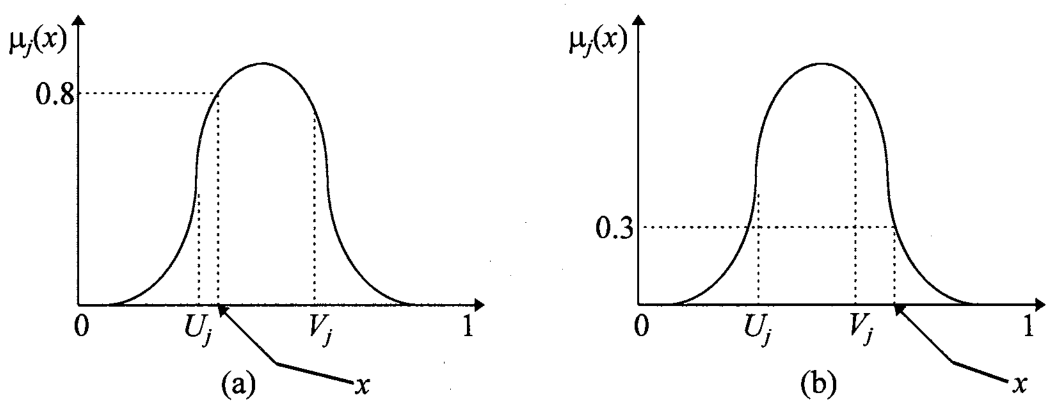


Figure 4.5. Relative positions for a pattern x to a hyper-box in F3MCNN: (a) The input pattern, x, locates inside the cluster's hyper-box. (b) The input pattern, x, locates outside the cluster's hyper-box.

4.3.6 Resonance or Reset

In the F3MCNN learning process, patterns are adapted into the learned concept and the statistical characteristics of clusters. The learned concept is represented by the hyper-box and the statistical characteristic is represented by the statistical fuzzy set function of each cluster. In the F3MCNN learning process, we proceed with fuzzy hyper-box clustering first. The attentional subsystem calculates the fuzzy HSD to every cluster of the input pattern. Then, the selection control subsystem searches and selects the cluster with the maximum fuzzy HSD. If there is only one cluster with the maximum fuzzy HSD, resonance occurs when the selected cluster satisfies the fuzzy hyper-box expansion criterion. Fuzzy statistical learning then ensues, as described above. Mismatch reset occurs if the candidate cluster cannot pass the fuzzy hyper-box expansion criterion. Then the value of the fuzzy choice function H_J is reset to zero for the duration of the particular input pattern presentation to avoid unnecessary processing during search. The search process continues until the selected cluster satisfies the fuzzy hyper-box expansion criterion or there are no more committed clusters available.

Obviously, in the previous discussion about resonance or reset we only considered the case where there is only one cluster with the maximum fuzzy HSD in fuzzy hyper-box clustering. However, it may happen that there are two or more clusters with the same maximum fuzzy HSD. In this case, the problem of hyper-box membership ambiguity happens. The selection control subsystem detects this ambiguity problem and issues a phase-change-signal to the F3MCNN system. Then, the attentional subsystem calculates the fuzzy SSDs of the input pattern to those clusters selected in fuzzy hyper-box clustering. Fuzzy statistical clustering invokes the fuzzy statistical choice function to select a cluster with the maximum fuzzy SSD. If the input pattern has the full fuzzy HSD to the selected cluster, only fuzzy statistical learning is needed to adjust the statistical characteristic of the selected cluster and learning stops. If the input pattern is located outside the hyper-box of the selected cluster, the fuzzy hyper-box expansion criterion is applied to check the size of the expanded hyper-box. Resonance occurs if the selected cluster passes the fuzzy hyper-box expansion criterion. Both fuzzy hyper-box learning and fuzzy statistical learning are performed. Otherwise, mismatch reset occurs if the selected cluster cannot pass the fuzzy hyper-box expansion criterion. Then the value of the fuzzy statistical choice function S_J is reset to zero for the duration of the input pattern presentation to avoid unnecessary processing during search. The search process continues until one selected cluster satisfies the hyper-box expansion criterion or there are no more selected clusters in fuzzy hyper-box clustering available. If there are no more selected clusters in fuzzy hyper-box clustering are available, the F3MCNN system performs the fuzzy hyper-box choice function again to select another cluster with the maximum fuzzy HSD and repeats the process described above.

CHAPTER V

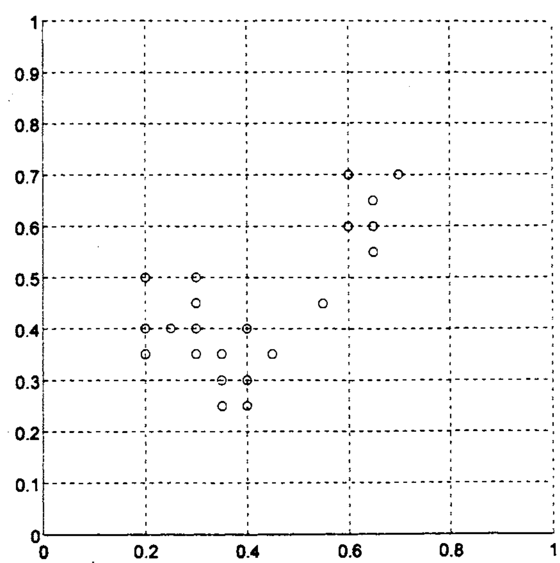
EXPERIMENTAL RESULTS AND COMPARISONS

5.1 Software Simulation and Purposes

Software simulation is used in our experiments to demonstrate the expected improvement in the clustering results of the F3MCNN model. The simulation programs were coded under the MATLAB software environment and ran on a 486-33MHz PC. Two data sets are utilized in our experiments. The first data set is a two-dimensional pattern set example uses to show the different clustering results of the F3MCNN model with defined various hyper-box sizes. The second data set is Fisher iris data [Fisher 36]. The experiment of the iris data is used to compare the accuracy of clustering results of the F3MCNN model with the fuzzy ART and fuzzy Min-Max models. Also, we analyze and compare model properties, clustering stability, and the number of epochs to achieve clustering stability among these three models.

5.2 Experimental Results

In our experiments, we use a two-dimensional pattern set to show how an F3MCNN system performs with defined various hyper-box sizes and the iris data to demonstrate the improvement in the F3MCNN clustering results. The reason the two-dimensional data set was chosen for the first experiment is that it's easier to present data points in two-dimensional data space than in higher dimensional data space. From clustering results with defined different maximum sizes of hyper-boxes, we show that it is important to have a feel for the appropriate hyper-box size for a specific application. We choose the Iris data set for the second experiment because it's a very famous data sample to the pattern recognition community that allows us to compare the F3MCNN clustering results with other similar techniques.



$P_1 = (0.20, 0.35)$ $P_2 = (0.20, 0.40)$ $P_3 = (0.30, 0.50)$ $P_4 = (0.20, 0.50)$
$P_5 = (0.25, 0.40)$ $P_6 = (0.30, 0.35)$ $P_7 = (0.30, 0.40)$ $P_8 = (0.30, 0.45)$
$P_9 = (0.40, 0.40)$ $P_{10} = (0.35, 0.25)$ $P_{11} = (0.35, 0.30)$ $P_{12} = (0.35, 0.35)$
$P_{13} = (0.40, 0.25)$ $P_{14} = (0.40, 0.30)$ $P_{15} = (0.45, 0.35)$ $P_{16} = (0.55, 0.45)$
$P_{17} = (0.60, 0.60)$ $P_{18} = (0.70, 0.70)$ $P_{19} = (0.60, 0.70)$ $P_{20} = (0.65, 0.55)$
$P_{21} = (0.65, 0.60)$ $P_{22} = (0.65, 0.65)$

Figure 5.1 The data used for the two-dimensional example in F3MCNN clustering

5.2.1 Two-dimensional Pattern Clustering

We construct a two-dimensional data set to show how an F3MCNN system performs with different user-defined vigilance parameters. This two-dimensional data set contains 22 patterns divided into two primary groups that are linearly separable. However, patterns in one of the primary groups could be considered either one cluster or two. We show the pattern values used for this experiment and their scatter plot in Figure 5.1.

This two-dimensional pattern set is classified four times with four different maximum hyper-box size parameters ranging from 0.5 to 0.15 as shown in Figure 5.2. When the maximum hyper-box size parameter is 0.5, the number of clusters created is one because all patterns can be included in one cluster as shown in Figure 5.2(a). There are two clusters created when the maximum hyper-box parameter is defined as 0.3. Two primary groups of patterns are separated as shown in Figure 5.2(b). However, when we define the maximum hyper-box parameter as 0.2, the patterns in the primary group on the left hand side are divided into two smaller clusters. In Figure 5.2(c), these two clusters' hyper-boxes overlap and three patterns, (0.35, 0.35), (0.35, 0.30), and (0.40, 0.30), are located in this overlapped area. We apply fuzzy statistical clustering to classify (0.35, 0.35) to the bigger cluster and (0.35, 0.30) and (0.40, 0.30) to the smaller cluster. When the maximum hyper-box size parameter is defined as 0.15, there are four clusters created as in Figure 5.2(d). Each primary group is divided into two clusters. As this example shows, the clustering results with the maximum hyper-box size defined as 0.3 and 0.15

seem to be able to find the underlying structure of the data set. However, the cluster arrangement that seems most appropriate is application-oriented.

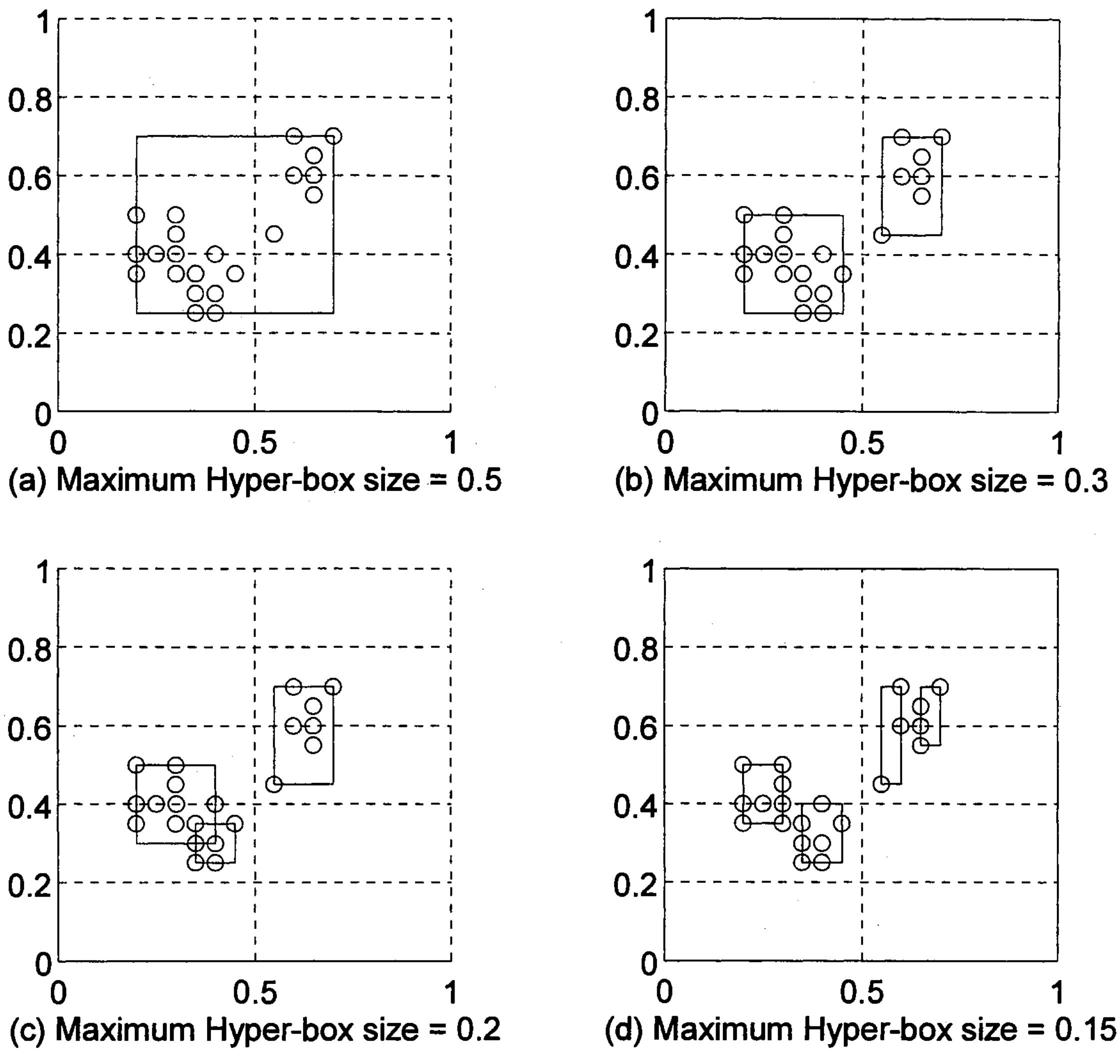


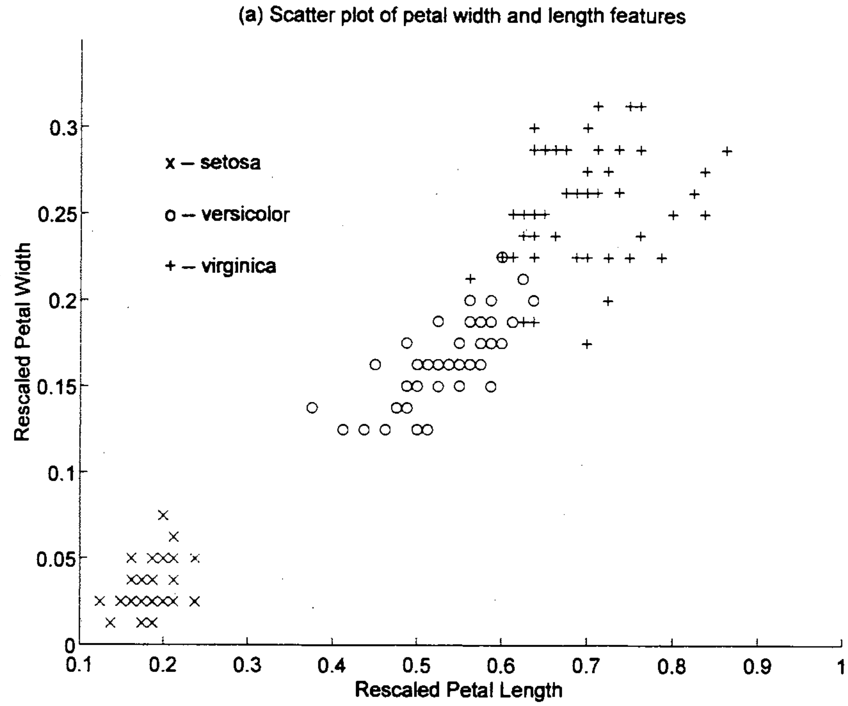
Figure 5.2 Two-dimensional clustering example with four different maximum hyper-box sizes

5.2.2 Iris Flowers Clustering

The iris data is used for the second experiment in order to compare the F3MCNN clustering results with clustering results from the fuzzy ART and fuzzy Min-Max models. The original iris flower data are given in [Fisher 36]. There are three classes in the iris flower data set: setosa, versicolor, and virginica. The data set consists of 150 four-dimensional feature vectors, 50 for each class. The four features that describe the shape and size of the flower are sepal length, sepal width, petal length, and petal width. In the iris data set, the setosa class is linearly separable from the other two, but the other two, versicolor and virginica, are not linearly separable from each other. Therefore, the iris data set is suitable for evaluating the performance of the F3MCNN model and allows performance comparisons between different models.

In Figure 5.3(a), the iris data are plotted for the petal features. We scale each feature vector to lie between 0 and 1 by dividing each component of each pattern by 8. Note that the first class (setosa) appears to be well separated from the other two. Also notice that there is overlapping between versicolor and virginica classes. Patterns in the overlapping area are the ones that will be difficult to classify unless they are separated on other features. We plot the iris data for the sepal features in Figure 5.3(b). Note that there is still no apparent separation between versicolor and virginica.

We use the evaluation method described in [Simpson 93] to evaluate the clustering performance of the F3MCNN model. Our experiments and results analysis consist of the following steps. In the first step, we classify the iris data into clusters



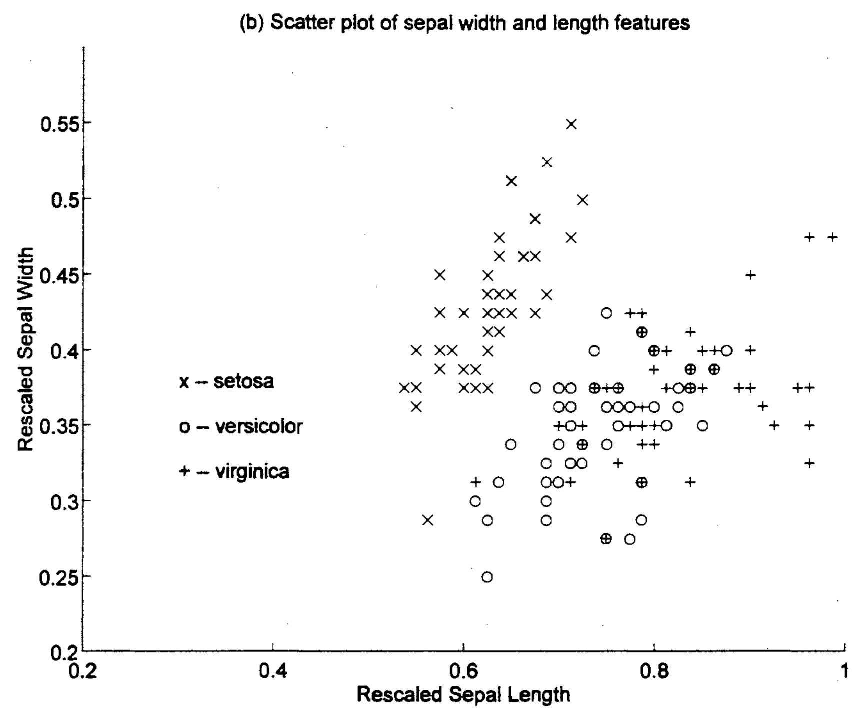


Figure 5.3 Scaled data plots for the three iris classes: setosa, versicolor, and virginica. (a) Plot of petal width vs. petal length. (b) Plot of sepal width vs. sepal length.

using six differently defined maximum hyper-box size parameters. Secondly, we utilized the class information provided in the original iris data set to identify which clusters are associated with each assigned class. Class information is not provided until all clusters are formed. In the third step, we calculate the accuracy rate that is defined as the ratio of the number of the correctly classified patterns to the total number of patterns. Note the cluster validity is another research topic that is beyond our current research objectives. Class information in the iris data set helps us identify clusters with classes in our experiments. However, since pattern clustering is an unsupervised pattern classification process, class information is not available for every pattern in practical applications. Therefore, there will not be an accuracy rate in pattern clustering defined as the same as in the supervised pattern classification.

TABLE 5.1

IRIS DATA CLUSTERING RESULT OF F3MCNN
WITH VIGILANCE = 0.21

Accuracy rate = 91.3%		Actual class			
		1	2	3	Assigned Clusters
Assigned class	1	50	0	0	1
	2	0	50	13	2
	3	0	0	37	3

The results of this experiment are shown in Table 5.1 to Table 5.6 for the vigilance parameters range from 0.21 to 0.10. The accuracy rate for the clustering results range from 91.3% to 97.3 %. When the vigilance is defined as 0.21, three clusters are created in F3MCNN clustering result corresponding to three classes, see Table 5.1.

Patterns in setosa and versicolor are classified into exactly correct classes; while, 13 patterns from virginica are misclassified into versicolor. The overall accuracy rate of the clustering result is 91.3%.

TABLE 5.2

IRIS DATA CLUSTERING RESULT OF F3MCNN
WITH VIGILANCE = 0.20

Accuracy rate = 95.3%		Actual class 1	2	3	Assigned Clusters
Assigned class	1	50	0	0	1
	2	0	49	6	2
	3	0	1	44	3, 4

When the maximum hyper-box size is defined as 0.2, there are four clusters created in F3MCNN clustering result. We identify cluster 1 as the first class (setosa), cluster 2 as the second class (versicolor), and cluster 3 and 4 as the third class (virginica), see TABLE 5.2. Fifty patterns in setosa are classified into the correct class; however, misclassification occurs in the patterns from both versicolor and virginica. One pattern from versicolor is misclassified as a pattern in virginica and six patterns from virginica are misclassified as patterns in versicolor. The overall accuracy rate of clustering result is 95.3%.

From Tables 5.3 to 5.6, the interpretation of the number of clusters, overall accuracy rate, class and clusters assignments, and numbers inside the table are the same as we explained previously. The best performance in this experiment is 97.3% with the maximum hyper-box size parameter defined as 0.10, which produces 13 clusters. We need only one pass of patterns to achieve the clustering stability in each instance.

TABLE 5.3

IRIS DATA CLUSTERING RESULT OF F3MCNN
WITH VIGILANCE = 0.18

Accuracy rate = 94%		Actual class 1	2	3	Assigned Clusters
Assigned class	1	50	0	0	1
	2	0	47	6	2, 3
	3	0	3	44	4, 5, 6

TABLE 5.4

IRIS DATA CLUSTERING RESULT OF F3MCNN
WITH VIGILANCE = 0.15

Accuracy rate = 94.7%		Actual class 1	2	3	Assigned Clusters
Assigned class	1	49	0	0	1
	2	1	43	0	2, 3
	3	0	7	50	4, 5, 6

TABLE 5.5

IRIS DATA CLUSTERING RESULT OF F3MCNN
WITH VIGILANCE = 0.12

Accuracy rate = 95.3%		Actual class 1	2	3	Assigned Clusters
Assigned class	1	49	0	0	1,2
	2	1	49	5	3, 4, 5, 6, 7
	3	0	1	45	8, 9, 10, 11, 12

TABLE 5.6

IRIS DATA CLUSTERING RESULT OF F3MCNN
WITH VIGILANCE = 0.10

Accuracy rate = 97.3%		Actual class 1	2	3	Assigned Clusters
Assigned class	1	50	0	0	1, 2, 3, 4
	2	0	47	1	5, 6, 7
	3	0	3	49	8, 9, 10, 11, 12, 13

5.3 Comparisons

There are many pattern clustering algorithms presented in the pattern recognition, neural network, and fuzzy set literature. The proposed F3MCNN is a synergetic model of fuzzy set theory and modified ART neural networks. Other interested clustering techniques include (1) fuzzy ART, (2) fuzzy Min-Max, and (3) fuzzy c-means clustering algorithm. In the comparison of these clustering techniques, we concentrate on the model properties, the clustering stability, the number of epochs to achieve clustering stability, and the accuracy of clustering results.

5.3.1 Model Properties Comparison

The fuzzy c-means clustering technique assumes the number of clusters in the data set and then finds the optimal cluster centers and membership functions which minimize an objective function. The first fuzzy c-means clustering neural network was developed by Dunn [Dunn 74] and generalized by Bezdek [Bezdek 73] [Bezdek 74] [Bezdek 75]. The fuzzy ART model closely resembles the F3MCNN model because they both grew out of the fuzzification of the ART-1 neural network. The fuzzy Min-Max model is also a fuzzy ART-type pattern clustering technique. Therefore, it closely resembles the F3MCNN model in many ways. We compare their model properties based on the cluster representation, the way the number of clusters is decided, the learning type, and the solution method for hyper-box overlap.

5.3.1.1 Cluster Representation Both prototype and hyper-box representations are used in the F3MCNN model. It utilizes a hyper-box to represent learned the concept of a cluster and a statistical fuzzy set function to represent the statistical characteristic of a cluster. The fuzzy *c*-means clustering algorithm utilizes a single point to represent the prototype of each cluster. In the fuzzy ART and fuzzy Min-Max models, only the minimum and maximum points of a hyper-box are used to represent a cluster. There is no information about how patterns behave inside a hyper-box.

5.3.1.2 Method of Determining the Number of Clusters The fuzzy *c*-means clustering model assumes the number of clusters is known in advance; while, the F3MCNN, fuzzy ART, and fuzzy Min-Max models determine the number of clusters dynamically. To determine the best clustering result, both algorithms need to cluster with different parameters (e.g., different numbers of clusters for fuzzy *c*-means clustering algorithm and different hyper-box sizes for F3MCNN, fuzzy ART, and fuzzy Min-Max).

5.3.1.3 Learning Type The F3MCNN, fuzzy ART, and fuzzy Min-Max models are real-time on-line pattern clustering systems. They learn new patterns without having to retrain or refer to any of the previous patterns.

In contrast, the fuzzy *c*-means clustering model is an off-line pattern clustering system. It needs all the patterns before learning. To learn a new pattern, a fuzzy *c*-means clustering system needs to use the previous data set and the new pattern for training. However, the on-line F3MCNN, fuzzy ART, and fuzzy Min-Max models are pattern-order-dependent. It is possible that the same data set will create different clustering results if the data are processed in a different order during learning.

5.3.1.4 Method to Handle Hyper-box Overlap Since the hyper-box representation for clusters is not used in the fuzzy *c*-means clustering model, we compare methods to handle hyper-box overlap in the F3MCNN, fuzzy ART, and fuzzy Min-Max models only. In the F3MCNN model, we allow hyper-box overlapping and utilize the statistical characteristic in hyper-boxes to solve the problem of full hyper-box membership ambiguity. Because it is impossible for two clusters to have the same central point, the problem of full statistical membership ambiguity never occurs in the clustering results of F3MCNN. For the input pattern located inside the hyper-box overlapping area, it is reasonable and natural to utilize the statistical characteristic in the hyper-box to select a candidate cluster. The results from the iris data clustering example show the improvement of F3MCNN clustering.

Hyper-box overlapping is allowed in fuzzy ART clustering and it results in the problem of full hyper-box membership ambiguity. To solve this problem, fuzzy ART utilizes the order of the committed clusters to choose the cluster with the smallest index as the candidate for learning. There is no explanation why the cluster with the smallest index is chosen in [Grossberg 91].

In fuzzy Min-Max, hyper-box overlapping is not allowed. A hyper-box overlap checking and contraction procedure is performed after each hyper-box expansion to make sure there are no overlapped hyper-boxes. Therefore, hyper-boxes may expand or contract during pattern learning. This hyper-box overlap checking and contraction procedure is tedious when the number of hyper-boxes increases. After all is done, there is still a major problem remaining in fuzzy Min-Max clustering. It is still possible that a pattern has full hyper-box membership in more than one cluster when it is a point along the boundary between two abutting hyper-boxes.

5.3.1.5 Similarity Measure Two fuzzy set functions are defined to measure the fuzzy HSD and fuzzy SSD of an input pattern to clusters respectively in the F3MCNN model. The hyper-box fuzzy set function, defined in Equation (3.9), calculates the fuzzy HSD that is a complement of the distance between an input pattern and a cluster's hyper-box. The statistical fuzzy set function, defined in Equation (3.18) is a π-function to calculate the fuzzy SSD of an input pattern to a cluster's fuzzy prototype.

There is only one fuzzy set function used in fuzzy ART to measure the similarity degree of an input pattern to hyper-box clusters, see Equation (3.3). The fuzzy ART model introduced the fuzzy subset [Kosko 86] into its category choice function. The category choice function calculates the degree to which a cluster is a fuzzy subset of an input pattern. Also, only one fuzzy set function is defined in the fuzzy Min-Max model to measure the hyper-box similarity degree for an input pattern to clusters, see Equation (3.7).

In the fuzzy *c*-means clustering algorithm, the grade of a pattern associated with a cluster is used to represent the fuzzy membership degree. The grade is a fuzzy function of the Euclidean distances between the pattern and clusters.

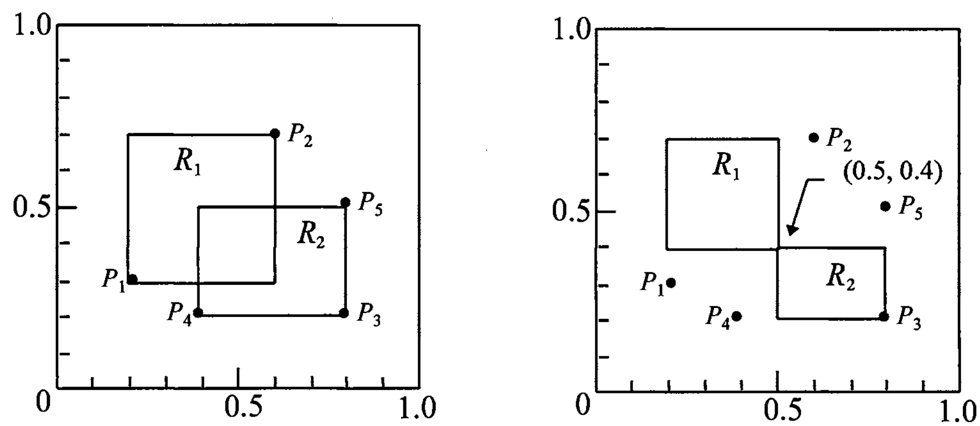
5.3.2 Clustering Stability Comparison

The definitions of clustering stability in F3MCNN and fuzzy ART are same. They are defined as when there is no change in hyper-boxes during successive presentations of the same input patterns in fast learning. Because every pattern is covered by one of the clusters' hyper-boxes after clustering, we need only one pass of the data patterns to achieve clustering stability. Therefore, there will be no change in hyper-boxes after one pass of the input patterns.

The clustering stability in the fuzzy *c*-means clustering algorithm is defined as the convergence of an iterative procedure in which a defined objective function is minimized. A fuzzy *c*-means clustering system always needs a vast number of iterations to minimize an objective function in order to reach stable cluster centers. In addition, the fuzzy *c*-means clustering algorithm iteration sequence may converge to either a local minimum or stable point of the objective function [Bezdek 87] [Sabin 87].

The clustering stability in the fuzzy Min-Max model is defined as when there is no change in hyper-boxes during successive presentations of the same ordered input patterns. The fuzzy Min-Max system needs more than one pass of a data set to achieve clustering stability in fast learning. Because the hyper-box overlap checking and contraction procedure is performed after each hyper-box expansion, hyper-boxes may be expanded or contracted during pattern learning. If there is no hyper-box overlap during learning, hyper-boxes are expanded to include input patterns. The fuzzy Min-Max clustering achieves learning stability in one epoch because there will not be any change in the size of the hyper-boxes. Otherwise, the fuzzy Min-Max clustering needs more than one epoch to achieve clustering stability.

In some cases, the hyper-boxes keep on changing very slightly after a large number of epochs in fuzzy Min-Max clustering. A stability criterion that is not



(a) Two overlapped hyper-boxes were created.

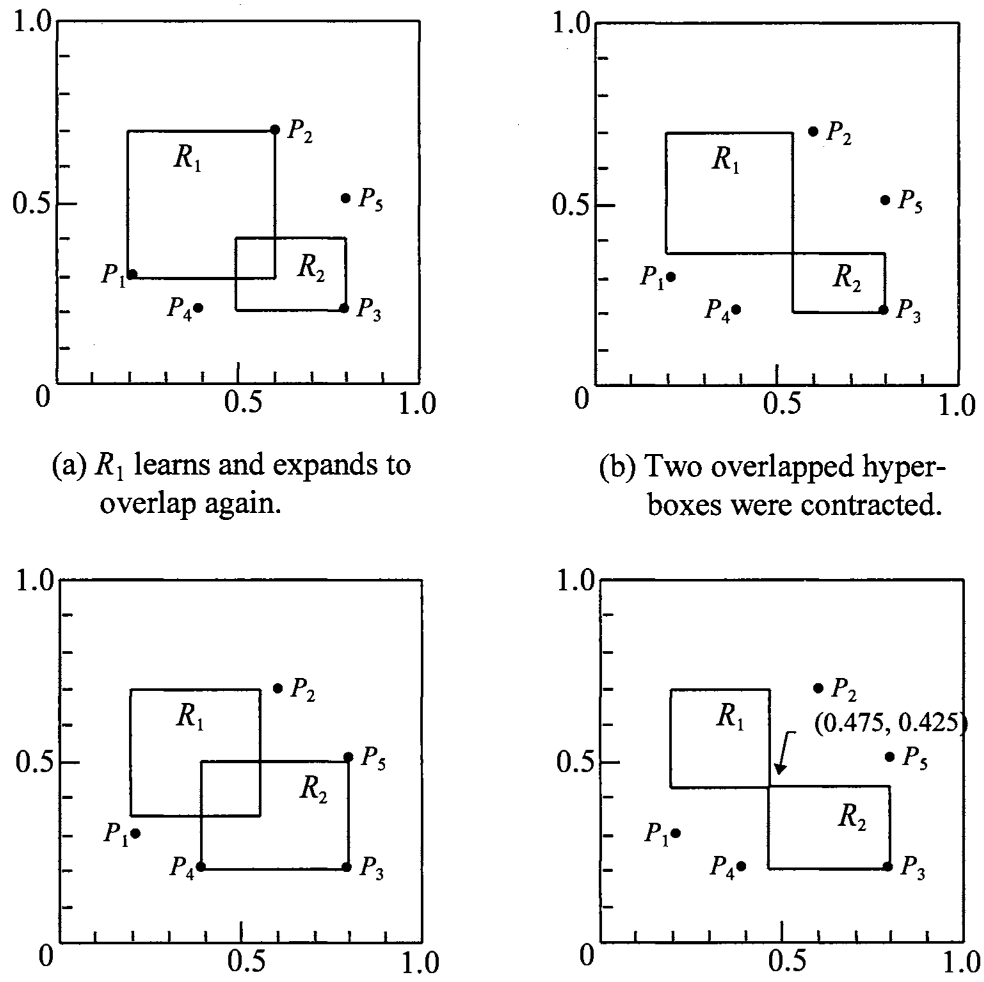
(b) Two overlapped hyper-boxes were contracted.

Figure 5.4 Iterative clustering example after the first epoch by using fuzzy Min-Max mentioned in [Simpson 93] needs to be defined in order to decide when to stop pattern training.

We use an example to illustrate the necessity. In this example, we use fuzzy Min-Max to classify five two-dimensional input patterns. The maximum hyper-box size is defined as 0.4. These five input patterns are represented as five points, P_1 to P_5, and are presented in their index order. In the first epoch. they are clustered into two clusters, R_1 and R_2, as shown in Figure 5.4(a). Because these two hyper-boxes overlap, we apply hyper-box contraction process defined in [Simpson 93] to contract both hyper-boxes as shown in Figure 5.4(b). After the first epoch, the abutting point between these two hyper-boxes is (0.5, 0.4).

We need to apply the second epoch to check whether the system achieves clustering stability or not. After P_1 and P_2 are presented again, the hyper-box R_1 is expanded to include theses two points. Again, there is hyper-box overlapping as shown in Figure 5.5(a). We need to apply the hyper-box contraction again to eliminate the hyper-box overlapping, see Figure 5.5(b). In the second epoch, three patterns, P_3, P_4, and P_5, are still not presented for clustering. After these three patterns are presented, the hyper-box R_2 is expanded again to include these three points as shown in Figure 5.5(c). Hyper-box overlapping happens after the expansion of the hyper-box R_2. We apply hyper-box contraction again to eliminate hyper-box overlapping as shown in Figure 5.5(d). After the second epoch, the abutting point between these two hyper-boxes is (0.475, 0.425). The changing of the abutting point means there is change in the size of the hyper-boxes. The difference in these two points is around 10^{-2}. We need to apply the input patterns again.

After the fifth epoch, the change of the abutting point is around 10^{-4}. The change of the abutting point is around 10^{-5} after the seventh epoch. The abutting point keeps changing after 24 epochs although the difference is then only around 10^{-15}.



(a) R_1 learns and expands to overlap again.

(b) Two overlapped hyper-boxes were contracted.

(c) R_2 learns and expands to overlap again.

(d) Two overlapped hyper-boxes were contracted.

Figure 5.5 Iterative clustering example after the second epoch by using fuzzy Min-Max

Obviously, we need to define a small number to stop pattern clustering in this example before the change in the size of the hyper-boxes approaches zero. This topic was not mentioned in [Simpson 93]. However, the hyper-box overlap checking and contraction procedure is time consuming when the number of hyper-boxes increases or when in high dimensional feature space.

5.3.3 Training Epochs Comparison

The F3MCNN and fuzzy ART models are superior to the fuzzy Min-Max and fuzzy *c*-means clustering models in respect to the need for training epochs. Hyper-boxes in F3MCNN and fuzzy ART are expanded to include a new input pattern and never contract. After all input patterns are presented, every pattern should be covered by a hyper-box. Therefore, only one epoch is needed in the F3MCNN and fuzzy ART systems to achieve clustering stability in fast learning.

In contrast, even an improved fuzzy *c*-means clustering algorithm with the reduced objective function needs 12 to 83 iterations to converge on experiments of different test data sets [Selim 92]. The fuzzy Min-Max model needs more than one pass of the same ordered input data to achieve clustering stability in fast learning. In the iris data experiments in [Simpson 93], less than 10 passes of the same ordered input data in each experiment are required to achieve clustering stability.

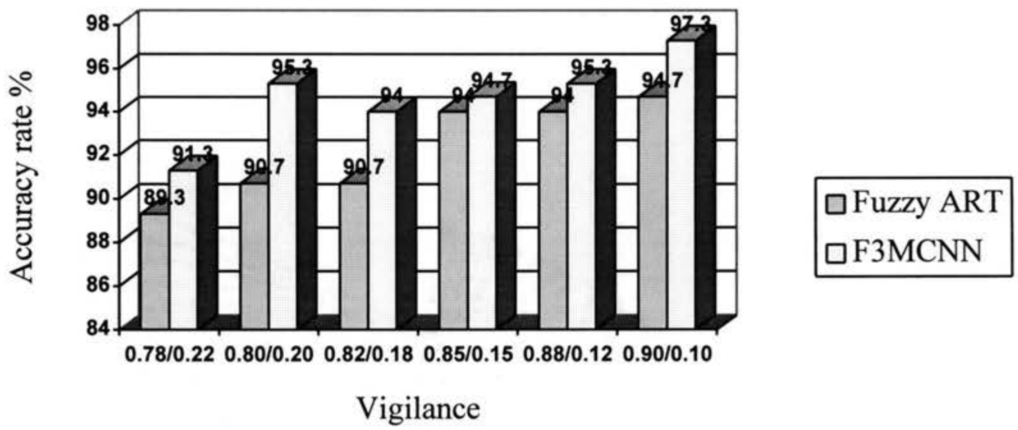


Figure 5.6 Iris data clustering results comparison between fuzzy ART and F3MCNN

5.3.4 Clustering Results Comparison

Comparing the iris data clustering results, the F3MCNN model is superior to the fuzzy ART and fuzzy Min-Max models in terms of the evaluation method discussed in [Simpson 93] to evaluate the clustering performance among F3MCNN, fuzzy Art, and fuzzy Min-Max. The clustering results of the iris data set of the F3MCNN model are presented in Section 5.2.2. The experiment consists of six different vigilance parameters defined as 0.22, 0.20, 0.18, 0.15, 0.12, and 0.10 and their resulting accuracy rates are 91.3%, 95.3%, 94.0%, 94.7%, 95.3%, and 97.3% respectively.

We also wrote a simulation program for the fuzzy ART model under the MATLAB software environment and ran it on a i486-33MHz PC. The same evaluation method is applied to evaluate the clustering performance for fuzzy ART.

The same six defined sizes of hyper-boxes are also used in the iris data experiment with fuzzy ART. Because the resonance is defined as the match function that is greater than or equals to the vigilance criterion, high vigilance results in smaller hyper-boxes and low vigilance results in bigger hyper-boxes in fuzzy ART clustering result. These six vigilance parameters are defined as 0.78, 0.80, 0.82, 0.85, 0.88, and 0.90 to map the hyper-box size limited by the six vigilance parameters in the iris data experiment with F3MCNN. Their accuracy rates are 89.3%, 90.7%, 90.7%, 94.0%, 94.0%, and 94.7%, respectively. Also, the number of clusters in the clustering results are 3, 4, 6, 8, 11, and 14, respectively. The clustering results comparison between fuzzy ART and F3MCNN is illustrated in Figure 5.6. The best performance is found with the hyper-box size of 0.10 for both fuzzy ART and F3MCNN. The accuracy rates are 94.7% and 97.3% in the clustering results of fuzzy ART and F3MCNN respectively. There is a notable improvement in the clustering result of F3MCNN over the clustering result of the fuzzy ART system for the iris flower data.

The clustering results of the iris data of the fuzzy Min-Max model with four different hyper-box sizes are presented in [Simpson 93]. The accuracy rates of experimental results are 86.7%, 88.0%, 90.2% and 92.7% with hyper-box sizes defined as 0.25, 0,20, 0.15, and 0.10. The number of clusters is 3, 3, 7, and 14 with respect to the defined hyper-box sizes. We use the F3MCNN clustering results of the iris data experiment with to compare with those of fuzzy Min-Max as shown in Figure 5.7. The best performance is found with the hyper-box size of 0.10 for both fuzzy Min-Max and F3MCNN. The accuracy rates are 92.7% and 97.3% in the clustering result of fuzzy

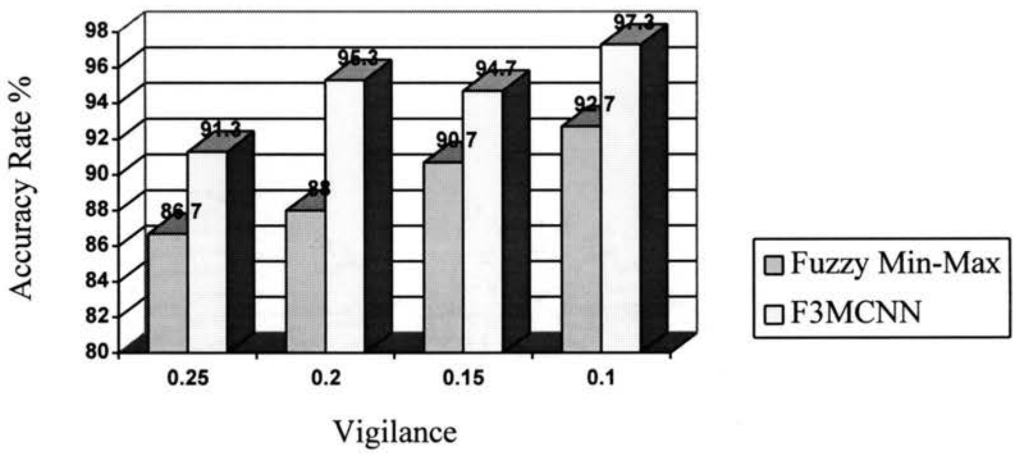


Figure 5.7 Iris data clustering results comparison
between F3MCNN and fuzzy Min-Max

Min-Max and F3MCNN respectively. Note that there also is an obvious improvement in the clustering result of F3MCNN over the clustering result of the fuzzy Min-Max system for the iris flower data.

5.3.5 Summary of Comparisons

In Section 5.3, we compare the model properties, the clustering stability, the number of epochs to achieve clustering stability, and the accuracy of the clustering results of fuzzy c-means, fuzzy ART, fuzzy Min-Max, and F3MCNN in detail. We summarize our comparisons briefly in TABLE 5.7.

TABLE 5.7

A SUMMARY OF COMPARISONS AMONG FUZZY C-MEANS, FUZZY ART, FUZZY MIN-MAN, AND F3MCNN

Compared Aspects	Fuzzy c-means	Fuzzy ART	Fuzzy Min-Max	F3MCNN
Learning Type	Off-line	Real-time	Off-line	Real-time
Decides number of clusters before clustering	Yes	No	No	No
Dynamically determines number of clusters during clustering	No	Yes	Yes	Yes
Cluster representation	Prototype	Hyper-box	Hyper-box	Hyper-box and Prototype
Hyper-box overlap	Not applied	Allowed	Not allowed	Allowed
Full membership ambiguity	No	Yes	Yes	No
Result is pattern order dependent	No	Yes	Yes	Yes
Minimize an objective function to achieve clustering stability	Yes	No	No	No
Stability defined as no change after successive passes of patterns	No	Yes	Yes	Yes
Number of epochs to achieve clustering stability	Vast number	One epoch	More than 10 epochs	One epoch
Clustering performance in iris data experiments	Not available	Better	Good	Best

CHAPTER VI

SUMMARY, CONCLUSIONS, AND FUTURE WORK

6.1 Epilogue

This research was undertaken to develop a real-time and unsupervised pattern classification system called the fuzzy minimum mean maximum clustering neural network (F3MCNN) to solve inexact pattern clustering problems. Inexact pattern clustering problems cannot be fully stated by means of mathematics. Human beings handle them subconsciously and without knowing how they really solve them. The F3MCNN model is designed to have human-like learning abilities: (1) it learns new patterns without needing to be retrained with previously learned patterns, (2) it learns pattern clusters without a supervisor, and (3) it can handle ill-separated cluster problems.

The F3MCNN model is a synergetic combination of a modified ART neural network with the fuzzy set theory; therefore, it has learning ability and can represent and manipulate inexact information. An F3MCNN system operates in two different phases: fuzzy hyper-box clustering and fuzzy statistical clustering. The system architecture of F3MCNN contains two subsystems: the attentional subsystem and the selection control subsystem. The attentional subsystem is responsible for pattern cluster learning. It is a modified version of the fuzzy adaptive resonance theory (ART) neural network

[Carpenter 87]. The selection control subsystem issues phase-change-signal and cooperates with the attentional subsystem in cluster selection and criterion matching. Both the fuzzy hyper-box similarity degree (fuzzy HSD) and the fuzzy statistical similarity degree (fuzzy SSD) are utilized in F3MCNN for pattern clustering. When a new pattern is presented, the F3MCNN system operates in the fuzzy hyper-box clustering phase to perform fuzzy hyper-box learning. If the problem of full hyper-box membership ambiguity occurs during hyper-box learning, the selection control subsystem makes the F3MCNN system operate in the fuzzy statistical clustering phase. Fuzzy statistical clustering eliminates the ambiguity problem and guarantees that no pattern has full membership belonging to more than one cluster in F3MCNN clustering results.

A very famous data set, Fisher's iris data [Fisher 36], is used as an example in our experiment to demonstrate the superiority of the F3MCNN model. We use the same evaluation method that was applied in the fuzzy min-max clustering neural network (fuzzy Min-Max) [Simpson 93] to evaluate the clustering performance of the F3MCNN model. We cluster data into clusters and identify each cluster with a class to determine how well the F3MCNN model was able to find the underlying structure of the data as the method mentioned in [Simpson 93]. The accuracy rates from our iris data experiment range from 91.3% to 97.3% with the defined vigilance parameters ranging from 0.21 to 0.10. Compared to the accuracy rate ranges of 86.7% to 92.7% presented in [Simpson 93] and 89.3% to 94.7% in fuzzy ART from our simulation program, the F3MCNN model does improve the clustering results as we expected. Moreover, the F3MCNN system achieves cluster stability just after one pass of patterns, but the Fuzzy Min-Max system needs about ten repeated passes of the input patterns in the same order to achieve clustering stability.

6.2 Contribution and Further Research

The main contribution of the F3MCNN model developed by this work is to provide a faster and more accurate pattern clustering technique for inexact pattern clustering problems. It provides the fuzzy similarity degree information that is extremely useful in higher level decision making applications. Moreover, its system architecture can be utilized as the subsystem for a supervised multi-layer pattern classification system. We outline the advantages in the F3MCNN model as follows:

(1) F3MCNN can handle a variety of real-world dynamic pattern clustering problems. The F3MCNN model creates clusters as needed without assuming the number of clusters in advance.

(2) F3MCNN can handle inexact pattern clustering problems. The F3MCNN model can classify both linearly separable and non-linearly separable clusters.

(3) Learning in F3MCNN is accumulated. The F3MCNN model is able to learn new patterns without having to be retrained with the previously learned patterns.

(4) Learning in F3MCNN is self-stabilized. Once the F3MCNN system learns a new pattern, it achieves clustering stability since the new pattern is covered by one of the hyper-boxes.

(5) Learning in F3MCNN is fast. An F3MCNN system achieves clustering stability in just one pass of input patterns. The F3MCNN model does not require time-consuming work for cluster overlap checking and contraction.

(6) Learning in F3MCNN is free of membership ambiguity. In an F3MCNN system, the fuzzy statistical clustering is activated to solve the problem of full membership ambiguity.

Future research directions include: developing a technique to measure the cluster validity for the F3MCNN model and exploring a method to alleviate the input order dependent problem in the F3MCNN model.

GLOSSARY

Attentional subsystem. A subsystem of the F3MCNN model, which is a three-layer neural network. It is responsible for calculating the fuzzy HSD and fuzzy SSD of patterns to clusters and encoding attributes of committed clusters.

***d*-function.** A three-parameter function, which calculates the distance along one feature dimension between an input pattern and the interval of the minimum and maximum points of a hyper-box.

F3MCNN. *See* Fuzzy minimum mean maximum clustering neural network.

Fuzzy central point. A fuzzy central point of a cluster is the average of those patterns that have been classified into the cluster. It is used as the center of the statistical fuzzy set function defined in the F3MCNN model.

Fuzzy HSD. *See* Fuzzy hyper-box similarity degree.

Fuzzy hyper-box choice function. A function used in the fuzzy hyper-box clustering process to select a candidate cluster having the maximum fuzzy HSD to the input pattern for learning.

Fuzzy hyper-box clustering. This is the first phase of the F3MCNN clustering process. In this clustering phase, the fuzzy hyper-box choice function is applied to select a candidate cluster. Then it checks whether the selected cluster meets the fuzzy hyper-box expansion criterion.

Fuzzy hyper-box expansion criterion. A criterion used in fuzzy hyper-box clustering to make sure the size of the expanded hyper-box of the selected cluster is not greater than the user-defined vigilance.

Fuzzy hyper-box learning. This is one of the pattern learning processes in the F3MCNN model, which adjusts the minimum and maximum points of the hyper-box when the selected cluster meets the fuzzy hyper-box expansion criterion. In fuzzy hyper-box learning, the hyper-box of the selected cluster is expanded to cover the input pattern.

Fuzzy hyper-box similarity degree (Fuzzy HSD). This is the similarity measure between the input pattern and the hyper-box of a cluster. It is a value between 0 and 1 calculated by the hyper-box fuzzy set function.

Fuzzy hyper-box. This is one of the representations for a cluster in the F3MCNN system. A fuzzy hyper-box of a cluster covers the patterns that have been classified into the cluster and is represented by its minimum and maximum points.

Fuzzy minimum mean maximum clustering neural network (F3MCNN). This is the model we proposed in this study for pattern clustering. It is a synergetic model of the fuzzy set theory and a modified ART neural network. In this model, the relation between patterns and clusters is represented by the fuzzy HSD and fuzzy SSD. Its clustering process contains fuzzy hyper-box clustering and fuzzy statistical clustering.

Fuzzy number of patterns. A fuzzy number of patterns for a cluster is the number of patterns that have been classified into the cluster. It is not the number of patterns that are located inside a cluster's hyper-box. A pattern does not have to be classified into a cluster even if it is located inside the cluster's hyper-box.

Fuzzy prototype. This is one of the representations for a cluster in the F3MCNN system. A fuzzy prototype of a cluster presents the statistical characteristic of patterns that have been classified into the cluster. It is defined by the fuzzy central point, fuzzy variation vector, minimum point and maximum point of a cluster.

Fuzzy SSD. *See* Fuzzy statistical similarity degree.

Fuzzy statistical choice function. A function used in the fuzzy statistical clustering process to select a candidate cluster having the maximum fuzzy SSD to the input pattern from the clusters selected in fuzzy hyper-box clustering process.

Fuzzy statistical clustering. This is the second phase of the clustering process of the F3MCNN system. In this clustering phase, a cluster with the maximum fuzzy SSD to the input pattern is selected among those clusters selected in the fuzzy hyper-box clustering. This clustering process is performed only when the problem of full hyper-box membership ambiguity occurs in fuzzy hyper-box clustering.

Fuzzy statistical expansion criterion. A criterion used in the fuzzy statistical clustering process, which performs exactly the same function as the fuzzy hyper-box expansion criterion.

Fuzzy statistical learning. This is one of the pattern learning processes in the F3MCNN model, which adjusts the fuzzy central point, fuzzy variation vector, and fuzzy number of patterns of a cluster when the cluster meets the fuzzy statistical expansion criterion.

Fuzzy statistical similarity degree (Fuzzy SSD). This is the similarity measure between the input pattern and the fuzzy prototype of a cluster. It is a value between 0 and 1 calculated by the statistical fuzzy set function.

Fuzzy variation vector. The fuzzy variation vector of a cluster is the average of the deviation vectors for patterns that have been classified into the cluster.

***H*-function.** *See* Hyper-box fuzzy set function.

Hyper-box fuzzy set function. The hyper-box fuzzy set function calculates the fuzzy HSD of a pattern to clusters. Its output value is the complement of the average of the *d*-function distance values.

***s*-function.** A fuzzy set function, which calculates the fuzzy SSD of an input pattern to a cluster along one feature dimension.

***S*-function.** *See* Statistical fuzzy set function.

Selection control subsystem. A subsystem of the F3MCNN model, which is responsible for selecting the candidate cluster, checking expansion criteria, and changing clustering process phases of the F3MCNN system.

Statistical fuzzy set function. The statistical fuzzy set function calculates the fuzzy SSD of patterns to clusters. Its output value is the average of the *s*-function output values. But its output value will equal 0 if there is any *s*-function output value equal to 0.

Vigilance. A user-defined parameter, which limits the maximum hyper-box size of clusters in fuzzy hyper-box learning.

BIBLIOGRAPHY

[Amari 72] Amari, S. I. "Learning Patterns and Pattern Sequences by Self-Organizing Nets of Threshold Elements," *IEEE Trans. Computers* C-21, pp. 1197-1206, 1972.

[Amari 77] Amari, S. I. "Neural Theory of Association and Concept Formation," *Biological Cybernetics* Vol. 26, pp. 175-185, 1977.

[Anderson 77] Anderson, J. A., Silveratein, J. W., Rite, S. A., and Jones, R. S. "Distinctive Features, Categorical Perception, and Probability Learning: Some Applications of a Neural Model," *Psych. Rev.*, Vol. 84, pp. 413-451, 1977.

[Bezdek 78] Backer, E. *Cluster Analysis by Optimal Decomposition of Induced Fuzzy Sets*, Delft: Delft University Press; 1978.

[Bezdek 73] Bezdek, J. C. "Fuzzy Mathematics in Pattern Classification," Ph.D. Thesis, Cornell University, Ithaca, NY, 1973.

[Bezdek 74] Bezdek, J. C. "Cluster Validity with Fuzzy Sets," *Journal of Cybernetics*, Vol. 3, No. 3, pp. 58-73, 1974.

[Bezdek 75] Bezdek, J. C. and Dunn, J. C. "Optimal Fuzzy Partitions: A Heuristic for Estimating the Parameters in a Mixture of Normal Distributions," *IEEE Trans. on Computers*, Vol. 24, pp. 835-838, 1975.

[Bezdek 81] Bezdek, J. C. *Pattern Recognition with Fuzzy Objective Function Algorithms*, New York: Plenum Press; 1981.

[Bezdek 87] Bezdek, J. C., Hathaway, R. J., Sabin, M. J., and Tucker, W. T. "Convergence Theory for Fuzzy C-Means: Counterexamples and Repairs," *IEEE Trans. Syst., Man, Cybern.*, Vol. SMC-17, No. 5, pp. 873-877, September/October 1987.

[Bezdek 92] Bezdek, J. C., Tsao, E. C., and Pal, N. R. "Fuzzy Kohonen Clustering Networks," *Proceedings of the 1st IEEE International Conference on Fuzzy Systems*, pp. 1035-1043, March 1992.

[Carpenter 87] Carpenter, G. A., and Grossberg, S. "A Massively Parallel Architecture for a Self-organizing Neural Pattern Recognition Machine," *Computer Vision, Graphics, and Image Understanding*, Vol. 37, pp. 54-115, 1987.

[Carpenter 88] Carpenter, G. A., and Grossberg, S. "Neural Dynamics of Category Learning and Recognition: Attention, Memory Consolidation and Amnesia," in *Brain Structure, Learning and Memory*, ed. I. Davis, R. Newburgh, I. Wegman. AAAS Symp. Series. Westview Press, Boulder, Co;o. 1988.

[Carpenter 90] Carpenter, G. A., and Grossberg, S. "ART 3: Hierarchical Search Using Chemical Transmitters in Self-Organizing Pattern Recognition Architectures," *Neural Networks*, Vol. 3, pp. 129-152, 1990.

[Carpenter 91a] Carpenter, G. A., Grossberg, S., and Reynolds, J. H. "ARTMAP: Supervised Real-Time Learning and Classification of Nonstationary Data by a Self-Organizing Neural Network," *Neural Networks*, Vol. 4, pp. 565-588, 1991.

[Carpenter 91b] Carpenter, G. A., Grossberg, S., and Rosen, D. B. "Fuzzy ART: Fast Stable Learning and Categorization of Analog Patterns by an Adaptive Resonance System," *Neural Networks*, Vol. 4, pp. 759-771, 1991.

[Carpenter 92] Carpenter, G. A., Grossberg, S., Markuzon, N., Reynolds, J. H., and Rosen, D. B. "Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps," *IEEE Transactions on Neural Networks*, Vol. 3, No. 5, pp. 700-712, September 1992.

[Caudill 89] Caudill, M. *Neural Networks Primer*, San Francisco: Miller Freeman, 1989.

[Choe 92] Choe, H. and Jordan, J. B. "On the Optimal Choice of Parameters in a Fuzzy C-Means Algorithm," *Proceedings of the 1st IEEE International Conference on Fuzzy Systems*, pp. 349-354, March 1992.

[Dubes 76] Dubes, R. and Jain, A. "Clustering Techniques: The User's Dilemma," *Pattern Recognition*, Vol. 8, pp. 247-260, 1976.

[Dunn 74] Dunn, J. "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters," *J. Cybern.*, Vol. 3, pp. 32-57, 1974.

[Fisher 36] Fisher, R. A. "The Use of Multiple Measurement in Taxonomic problems," *Annals of Eugenics*, Vol. 7, pp. 179-188, 1936.

[Fu 93] Fu, L. "A Neural Network Model for Real-Time Adaptive Clustering," *Proceedings of the IEEE International Conference on Neural Networks*, pp. 413-416, 1993.

[Fukunaga 70] Fukunaga, K. and Koontz W. L. G. "A Criterion and an Algorithm for Grouping Data," *Trans. IEEE Computers*, C-19, pp. 917-923, 1970.

[Fukunaga 90] Fukunaga, K. *Introduction to Statistical Pattern Recognition.* New York: Academic Press, 2nd Edition 1990.

[Fukushima 80] Fukushima, K. and Miyaka, S. "Neocognitron: A Self-Organizing Neural Network Model for s Mechanism of Pattern Recognition Unaffected by Shift in Position," *Biological Cybernetics,* Vol. 36, No. 4, pp. 193-202, 1980.

[Grossberg 76] Grossberg, S. "Adaptive Pattern Classification and Universal Recording. II: Feedback, Expectation, Olfaction, and Illusions," *Biological Cybernetics*, Vol. 23, pp. 187-202, 1976.

[Grossberg 77] Grossberg, S. "Classical and Instrumental Learning by Neural Networks," in *Progress in Theoretical Biology*, Vol. 3, New York: Academic Press, pp. 51-141, 1977.

[Grossberg 80] Grossberg, S. "How Does a Brain Build a Cognitive Code?" *Psychological Review*, Vol. 1, pp. 1-51, 1980.

[Grossberg 82] Grossberg, S. *Studies of Mind and Brain: Neural Principles of Learning Perception, Development, Cognition, and Motor Control.* Boston: Reidel Press, 1982.

[Hall 65] Hall, D. J. and Ball, G. B. "ISODATA: A Novel Method of Data Analysis and Pattern Classification," Technical report, Standard Research Institute, Menlo Park, CA, 1965.

[Hathaway 86] Hathaway, R. J. and Bezdek, J. C. "Local Convergence of the Fuzzy c-means Algorithms," *Pattern Recognition*, Vol. 19, No. 6, pp. 477-480, 1986.

[Hebb 49] Hebb, D. O. *The Organization of Behavior, A neuropsychological Theory.* New York: Jone Wiley, 1949.

[Hecht-Nielsen 87] Hecht-Neilsen, R. "Counterpropagation Networks," *Appl. Opt.*, Vol. 26, No. 23, pp. 4979-4984, 1987.

[Hopfield 82] Hopfield, J. J. "Neural Networks and Physical System with Emergent Collective Computational Abilities," *Proc. Natl. Acad. Sci.*, Vol. 79, pp. 2554-2558, 1982.

[Hopfield 84] Hopfield, J. J. "Neurons with Graded Response Have Collective Computational Properties Like Those of Two State Neurons," *Proc. Natl. Acad. Sci.*, Vol. 81, pp. 3088-3092, 1984.

[Huntsberger 89] Huntsberger, T. and Ajjimarangsee, P. "Parallel Self-Organizing Feature Maps for Unsupervised Pattern Recognition," *Int'l. Jo. General Systems*, Vol. 16, pp. 357-372, 1989.

[Ismail 86] Ismail, M. A. and Selim, S. Z. "Fuzzy c-means: Optimality of Solutions and Effective Termination of the Algorithm," *Pattern Recognition*, Vol. 19, No. 6, pp. 481-485, 1986.

[Kandel 86] Kandel, A. *Fuzzy Mathematical Techniques With Applications*. Reading, MA: Addison-Wesley, 1986.

[Kohonen 77] Kohonen, T. *Associative Memory: A System-Theoretical Approach*, Berlin: Springer-Verlag, 1977.

[Kohonen 82] Kohonen, T. "A Simple Paradigm for the Self-Organized Formation of Structured Feature Maps," in *Competition and Cooperation in Neural Nets*. ed. S. Amari, M. Arbib, Vol. 45, Berlin: Springer-Verlag, 1982.

[Kohonen 84] Kohonen, T. *Self-Organization and Associative Memory*, Berlin: Springer-Verlag, 1984.

[Kohonen 88] Kohonen, T. "The 'Neural' Phonetic Typewriter," *IEEE Computer*, Vol. 27, No. 3, pp. 11-22, 1988.

[Kosko 92a] Kosko, B. "Fuzzy Systems As Universal Approximators," *Proceedings of the 1st IEEE International Conference on Fuzzy Systems*, pp. 1153-1162, March 1992.

[Kosko 92b] Kosko, B. *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*. Englewood Cliffs, NJ: Prentice-Hall, 1992.

[Lippmann 87] Lippmann, R. P. "An Introduction to Computing with Neural Nets," *IEEE Magazine on Acoustics, Signal and Speech Processing*, pp. 4-22, April 1987.

[McCulloch 43] McCulloch, W. S. and Pitts, W. H. " A Logical Calculus of the Ideas Imminent in Nervous Activity," *Bull. Math. Biophy.*, Vol. 5, pp. 115-133, 1943.

[Nilsson 90] Nilsson, N. J. *Learning Machines: Foundations of Trainable Pattern Classifiers*, New York: McGraw Hill, 1965; also republished as *The Mathematical Foundations of Learning Machines,* Morgan-Kaufmann Publishers, San Mateo, Calif., 1990.

[Pao 89] Pao, Y. H. *Adaptive Pattern Recognition and Neural Networks*, Reading, Mass.: Addison-Wesley Publishing Co., 1989.

[Pedrycz 85] Pedrycz, W. "Algorithm of Fuzzy Clustering with Partial Supervision," *Pattern Recognition Lett.*, Vol. 3, pp. 13-20, 1985.

[Pedrycz 86] Pedrycz, W. "Techniques of Supervised and Unsupervised Pattern Recognition with the Aid of Fuzzy Set Theory," *Pattern Recognition in Practice* II, E.S. Gelsema and L.N. Kanal, Eds, pp. 439-448, Amsterdam: North Holand; 1986.

[Pedrycz 90] Pedrycz, W. "Fuzzy Set in Pattern Recognition: Methodology and Methods," *Pattern Recognition*, Vol. 23, No. 1/2, pp. 121-146, 1990.

[Rosenblatt 58] Rosenblatt, F. "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain," *Psych. Rev.*, Vol. 65, pp. 386-408, 1958.

[Rumelhart 86] Rumelhart, D. and McClelland, J. , Eds., *Parallel Distributed Processing: Explorations in the Microstructure of cognition.* Cambridge, MA: MIT Press, 1986.

[Ruspini 69] Ruspini, E. H. "A New Approach to Clustering," *Inf. Control*, Vol. 15, pp. 22-32, 1969.

[Ruspini 70] Ruspini, E. H. "Numerical Methods for Fuzzy Clustering," *Inf. Sci.*, Vol. 2, pp. 319-350, 1970.

[Ruspini 72] Ruspini, E. H. "Optimization in Sample Descriptions -- Data reduction and Pattern Recognition Using Fuzzy Clustering," *IEEE Transactions on Syst. Man Cybernetics*, Vol. 2, pp. 541, 1972.

[Sabin 87] Sabin, M. J. "Convergence and Consistency of Fuzzy C-Means/ISODATA Algorithms," *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. PAMI-9,, No. 5, pp. 661-668, September 1987.

[Selim 86] Selim, S. Z. and Ismail, M. A. "On the Local Optimality of the Fuzzy ISODATA Clustering Algorithm," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAM1-8, No. 2, pp. 284-288, 1986.

[Selim 92] Selim, S. Z., and Kamel, M. S. "On the Mathematical and Numerical Properties of the Fuzzy C-Means Algorithm," *Fuzzy Set and System*, Vol. 49, pp. 181-191, 1992.

[Simpson 90] Simpson, P. K. "Fuzzy Adaptive Resonance Theory," *South Illinois University Neuroengineering Workship*, Carbondale, IL, Sept. 6-7, 1990.

[Simpson 93] Simpson,P. K. "Fuzzy Min-Max Neural Networks-Part 2: Clustering," *IEEE Transactions on Fuzzy Systems*, Vol. 1, No. 1, February 1993.

[Wang 92] Wang, L. "Fuzzy Systems Are Universal Approximators," *Proceedings of the 1st IEEE International Conference on Fuzzy Systems*, pp. 1163-1169, March 1992.

[Widrow 60] Widrow, B. and Hoff, M. E. Jr. "Adaptive Switching Circuits," 1960 IRE Western Electric Show and Convention Record, part 4, pp. 96-104, Aug. 23, 1960.

[Widrow 62] Widrow, B. "Generalization and Information Storage in Networks of Adaline 'Neurons'," in *Self-organizing System*, M. C. Jovitz, g. T. Jacobi, and G. Goldstein. eds., Washington, D. C.: Spartan Books, pp. 435-461, 1962.

[Zadeh 65] Zadeh, L. "Fuzzy Sets," *Inform. Contr.*, vol. 8, pp. 338-353, 1965.

2

VITA

Liang-Tsan Wu

Candidate for the Degree of

Doctor of Philosophy

Thesis: F3MCNN: FUZZY MINIMUM MEAN MAXIMUM CLUSTERING NEURAL NETWORK

Major Field: Computer Science

Biographical:

Personal Data: Born in Taiwan, R.O.C., on November 21, 1959, the son of Tong-Kuei Wu and Su-Kuan Lee.

Education: Graduated from Kaohsiung Senior High School, Kaohsiung, Taiwan in July 1977; received the Bachelor of Science degree in Mathematics from National Tsing Hua University, Hsinchu, Taiwan in July 1982; completed the requirements for the Doctor of Philosophy degree at Oklahoma State University in July 1995.

Professional Experience: Foreign Sales Engineer, Sunshine Electronic Industry Co., Ltd., Taipei, Taiwan, August 1983, to November 1985; System Analyst and Programmer, Data Processing Center, Sin Kong Life Insurance Co., Ltd., Taipei, Taiwan, December 1986, to July 1988; Database System developer and DP specialist, Department of Environmental Quality, State of Oklahoma, Oklahoma City, OK, May 1992 to August 1992, and May 1994 to August 1994; employed by Oklahoma State University, Department of Computer Science as graduate assistant; Oklahoma State University, Department of Computer Science, August 1989 to present.