30891

P2305

Projek Penyelidikan Jangka Pendek

Fingerprint Recognition System Using NeuroFuzzy

Ketua Projek: Dr. Rosalina Abdul Salam
Pusat Pengajian Sains Komputer

Tempoh Projek
16 Mac 2005 – 14 Mac 2007

Kandungan

1. Borang Laporan Akhir projek yang telah disahkan
2. Laporan Komprehensif Projek
3. Penerbitan (X2)
4. Penyata Kewangan daripada Jabatan Bendahari

Borang Laporan Akhir projek yang telah disahkan

3) **Abstrak untuk penyelidikan anda**

(Perlu disediakan di antara 100 – 200 perkataan di dalam **Bahasa Malaysia dan Bahasa Inggeris**. Ini kemudiannya akan dimuatkan ke dalam Laporan Tahunan Bahagian Penyelidikan & Inovasi sebagai satu cara untuk menyampaikan dapatan projek tuan/puan kepada pihak Universiti & luar).

Abstract of Research

(Must be prepared in 100 – 200 words in Bahasa Malaysia as well as in English. This abstract will later be included in the Annual Report of the Research and Innovation Section as a means of presenting the project findings of the researcher/s to the university and the outside community)

The main objective of this project is to build a Fingerprint Clustering System using Fuzzy Logic (FL) and Neural Networks (NN). The hybrid of these two methods that is NeuroFuzzy was implemented and compared to both existing methods. Clustering of fingerprints can help to reduce the complexity of the search process in a database. This can be done by grouping fingerprints with the same characteristic in the same group. The matching algorithm can compare stored fingerprint codes with only one cluster instead of the entire database. In this research, we classify fingerprints into five categories which are arch, left loop, right loop, whorl, and others. The last category is use to categorize fingerprint pattern other then the four categories. Finally, experiments were carried out to show that clustering can reduce the recognition time. Experiments were carried out using neural network classifier, fuzzy logic and Neurofuzzy. Results showed that NeuroFuzzy classifier is the best among the three.

Objektif utama projek ini adalah mambina satu Sistem Kluster Cap Jari dengan menggunakan teknologi Rangkaian Neural (NN) dan teknologi Logik Kabur (LK). Hybrid kedua-dua teknik iaitu NueroKabur telah dibina dan dibandingkan dengan kedua-dua teknik asal. Kluster capjari dapat mengurangkan masalah yang kompleks yang dihadapi dalam proses gelintaran pangkalan data. Ini dapat diatasi dengan mengumpulkan capjari yang mempunyai cirr-i-ciri yang sama dalam satu kumpulan. Algoritma pemadanan dapat mebezakan capjari yang di simpan dalam satu kluster berbanding dengan maklumat yang terdapat dalam keseluruhan pangkalan data. Di dalam penyelidikan ini capjari diklasifikasi dalam lima kategori, lengkungan, gelung kiri, gelung kanan, lingkaran dan lain-lain. Kategori yang akhir digunakan bagi capjari yang lain dari yang terdapat dalam kempat-empat kategori yang dinyatakan. Akhirnya, eksperimen dijalankan bagi membuktikan bahawa kluster dapat mengurangkan masa pemadanan. Eksperimen dijalankan dengan menggunakan rangkaian neural, logik kabur dan neurokabur. Keputusan yang diperolehi menunjukkan bahawa kaedah Neurofuzzy adalah yang terbaik.

Laporan Komprehensif Projek

CONTENT

Contents ***Page***

Chapter 1: Introduction

Upon the completion of the first stage of our fingerprint recognition system, we encountered two problems. The first problem is that it only able to perform recognition on computer generated fingerprint images and not good with real fingerprint images (from fingerprint scanner) and our early hypothesis; this problem is cause by the real fingerprint image consist more noise and image enhancement techniques must be applied during the image preprocessing level. The second problem that we encounter is the system matching efficiency is low. The system with 150 data in database the system need an average of 80 seconds to recognize a fingerprint and almost 140 seconds to realize that the fingerprint does not exist in our database. This problem can be solved with several ways such as optimize the feature vector representative, optimize the algorithm or apply clustering methods on the database.

Classification can be applied on the fingerprint recognition system to reduce the complexity of the process of database search. Classification can helps to reduce complexity of the fingerprint database by grouping fingerprints with the same characteristic in the same group and eventually the matching algorithm only needs to compare stored fingerprint codes that belong to the same class instead of the entire database. [10] In this paper we propose and developed our classification which can be perform with fingerprints into five categories which are arch, left loop, right loop, whorl, and others. Tented arch will be categorize in the arch category, while the others category is use to categorize fingerprint pattern other then the four categories.

There are several pattern recognition algorithms have been proposed for solving fingerprint classification problems, according to [48] such as early syntactic approaches, methods which based on detection of singular points, connectionist algorithms such as self-organizing feature maps, neural networks, structural methods based on graph matching, and multi-space principal component analysis but did not solve it completely.

From our survey, fuzzy neural network seems to be able to provide us a better solution to perform classification. According to [1] there are many ways to combine neural networks models and fuzzy logic. Neural network models are able to provide algorithms for numeric classification, optimization, and associative storage and recall while fuzzy logic able to work at the semantic level and provide a solution to process inexact or approximate data. Fuzzy neural is the combination of neural network with fuzzy logic, this combination will provide us even greater representation power, higher processing speed, and are more robust than conventional neural network. There are many other researches proposed and claim that fuzzy neural is good.

Therefore we proposed and developed fuzzy neural classifier on our fingerprint classification system. Our objective is to build a classifier with 80% accuracy and able to perform classification within one second time. Besides testing the accuracy and efficiency of fuzzy neural classifier, we also implemented neural network classifier and fuzzy logic classifier to make a comparison with fuzzy neural classifier. The comparison will cover areas such as the accuracy, efficiency, ease of training / learning, and robustness.

1.2 Objectives and Research Scope

The main objective of this research is to build three classifiers which uses neural network, fuzzy logic and fuzzy neural methodologies to speed up the matching time of our fingerprint recognition system which we build earlier and figure 1 show the time that we intended to achieve.

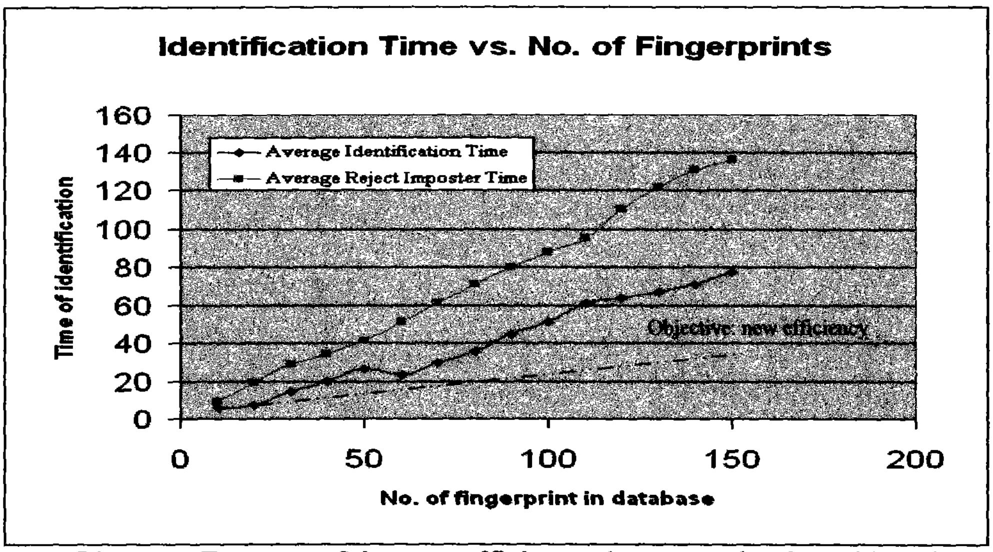


Figure 1: Forecast of the new efficiency that wanted to be achieved

In this process we will study and learn to apply neural-network and fuzzy logic methodologies to solve our classification problem. After we succeed develop the neural network and fuzzy logic classifier, we will proceed to combine fuzzy logic and neural network together to create and build a fuzzy neural classifier. Our objective includes developing a way to adapt the three methodologies in to our problem domain. After the three methodologies are being developed, we will perform test on them and make conclusion on their performance and find the best classifier.

The scope of this research can be divided into 3 sections, the first section is the individual scope which is I am going to develop the neural network classifier and my partner will develop the fuzzy logic classifier. After we finish performing our own scope, we will proceed to the second section we will join together to build fuzzy neural classifier. Last section, we will compare the three methodologies through several experiments (accuracy and efficiency). Before we begin the first section we will have to develop a way to segment and extract features from fingerprint and convert the features into the input for the three classifiers. Therefore, we proposed to build a direction reader program that able to read and generate a series of code that able to represents the pattern of fingerprints and also can be used as input for our fingerprint classifiers.

Chapter 2: Literature Review

2.1 Previous work

In the previous work [47], we had developed a fingerprint recognition system which is minutiae based and uses Euclidean distance for the fingerprint matching. The system is able to perform verification and recognition. The system will extract feature from the provided fingerprint image and then the extracted feature will be use to create a finger code which according to the arrangement the fingerprint's minutiae and it is different for every fingerprint. The finger code is then stored in the database to perform recognition and verification later. Generally, a fingerprint recognition system has to tolerate with 3 problems and that are transition, rotation and scale. In our matching algorithm we had applied finger code to solve the transition and rotation problem. Although, we did not solve the scale problem but theoretically our finger code is able to solve the scale problem. Figure 2 is the frame work of our system.

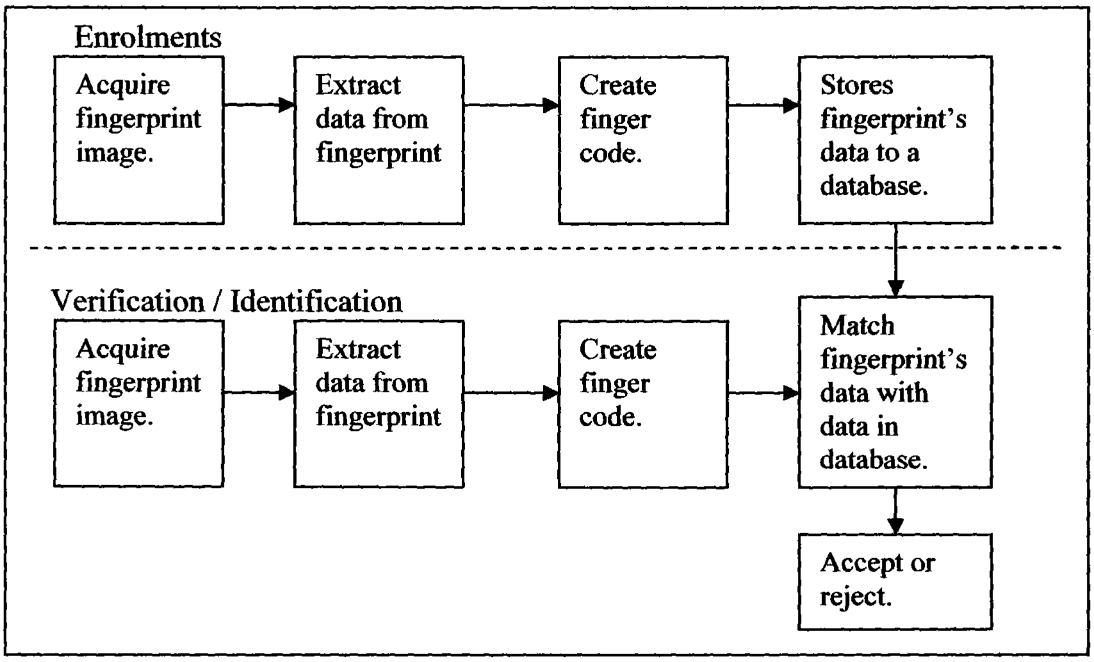


Figure 2: Our fingerprint recognition system frame work [47]

After the completion of this fingerprint recognition system, we encounter two new problems. The first problem is fingerprint's image with noise, our system perform good on computer generated fingerprint images but not with real fingerprint images. This problem can be solved by embedding better image enhancement techniques at the preprocessing level. The second problem is the matching efficiency and Figure 3 shows our algorithm efficiency. Figure 3 shows that with 150 data in our database we need an average of 80 seconds to recognize a fingerprint and almost 140 seconds to realize that the fingerprint does not exist in our database. This results that the efficiency of our algorithm does not reach an acceptable stage.

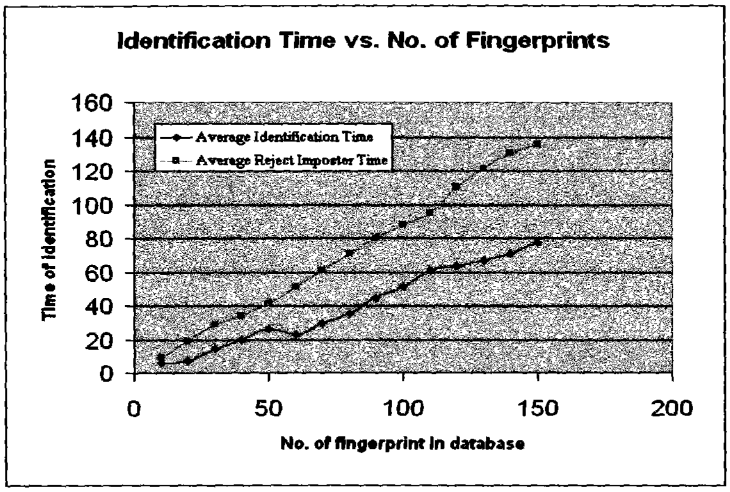


Figure 3: Our fingerprint recognition system efficiency [47]

There are a few approaches that we can take to solve this efficiency problem, such as modify the finger code and optimize the program code or apply clustering methods on our database. This is because we are confident that if we modify our finger code, optimize our program and apply clustering methods on our database and matching algorithm we will be able to enhance our algorithm efficiency.

In this paper, we choose to enhance our fingerprint recognition system by solving the second problem with the second alternative; implement a clustering method. We did not choose to solve the first problem (image noise) because we decided to concentrate in automated fingerprint classification method instead of image enhancement and partially also because we have the constraint of time and the scope of image enhancement are much larger compare to fingerprint classification and clustering.

2.2 What is clustering?

Clustering is a process of decomposition data into groups, each groups are form according to the similarity available of the data characteristic. The characteristic of these groups can be used to find similar groups of data items. According to [9], if clustering is performed with classification which made with abstract algorithmic space; the data may be group into subsets with similar characteristic such as goals and abilities. This can reduce the problem space by having a representative from each subset.

Clustering is a very important process in data mining; clustering can be very effective if the problem domain has huge number of separated patterns. Biology is also an area which needs many clustering applications; examples from [9] are phylogenetic tree construction, taxonomy generation, and genome analysis. This is because the clustering process to reduce enormous amount of data to manageable amount. To perform clustering for the purpose of cognitive and computational simplification, the data in a same cluster must share some similarity. There is no metric to measure similarity, it depends on assumptions and the desired on how to represent the data. For example, the same data can be differently cluster because of the usage of different similarity measures and clustering with different similarity measures may be equally valid [9]. The general clustering algorithm according to [9]:

Preprocessing and feature extraction – Data items usually can be represented with one or more feature vectors. Therefore in this first step of clustering, it is important to choose the appropriate feature vector according to the data usage so that appropriate preprocessing and feature extraction can be chose. This step usually requires a good domain knowledge and data analysis.

Similarity measure – This step takes two sets of data items as inputs and returns the output a similarity measure between the inputs. The item similarity measure can be perform with method such as weights from a fuzzy logic, Hamming distance, Mahalanobis distance, and edit distance. Then define the cluster representative for example max / min / average distance or any representation that are able represent the data separately in groups.

Clustering algorithm – it is a process that use a particular similarity measures to perform classification and clustering. The choice of which clustering algorithm to use are usually depends on the desired properties of the final clustering. The examples of the desired properties could be compactness, parsimony and inclusiveness of data, time and space complexity.

Result validation – this step is to make sure that the classification results make sense. If not the then the clustering process needed to be repeated again. There might be chances that some data do not fit into any cluster at all.

Result interpretation and application – The application that uses clustering are usually include data compression, hypothesis generation, hypothesis testing and prediction.

According to [10] clustering algorithms can be classified as Figure 4, although the classification of clustering algorithm may be overlapping. In this paper, we will use [10] classification of clustering algorithm to perform a brief study the available and common clustering algorithms. According to [10] traditional clustering algorithm can be categorized as hierarchical clustering algorithm or partitioning clustering algorithm. The different between hierarchical clustering and partitioning clustering is that the hierarchical clustering build clusters gradually while partitioning clustering learn all the clusters immediately.

The hierarchical clustering algorithm builds a hierarchy of clusters which also known as dendrogram (a tree of clusters). This dendrogram have the properties like a tree such as every cluster node has child clusters and sibling clusters to partition the cluster with their common parent. Hierarchical clustering methods can be categorized into agglomerative and divisive. The agglomerative clustering starts with one point clusters and iteratively merges suitable clusters together while the divisive starts with one cluster and iteratively split the appropriate cluster. These iterative processes usually stop after the requested number of clusters being achieved.

Clustering Algorithm

- Hierarchical Methods
  - Agglomerative Algorithms
  - Divisive Algorithms
- Partitioning Methods
  - Relocation Algorithms
  - Probabilistic Clustering
  - *K*-medoids Methods
  - *K*-means Methods
- Density-Based Algorithms
  - Density-Based Connectivity Clustering
  - Density Functions Clustering
- Grid-Based Methods
- Clustering Algorithms Used in Machine Learning
  - Artificial Neural Networks
  - Fuzzy Logic
  - Fuzzy Neural

Figure 4: The classification of clustering algorithms [10]

According to [10] the partitioning clustering algorithm divides data into several subsets to perform clustering. The process of checking all possible subset will need a lot of computational power and not practical, therefore certain heuristics algorithms are used. This means that different relocation schemes that iteratively reassign points between the k clusters are used. The are other approach to data partitioning is by taking a conceptual point of view that able to identifies clusters with certain models which unknown parameters have to be found. Example of such approach is the probabilistic clustering techniques. The advantage of probabilistic clustering is the interpretability of the constructed clusters, which means that by creating cluster that concise it will reduce the cost of computing the intra-clusters measures.

Another approach that was used to solve the problem of computing pair-wise distance or compute similarities by measures the inter-cluster and intra-cluster relations is by using unique cluster representatives and computes the objective function. This is known as iterative optimization partitioning algorithms which can be subdivided into k-means and k-medoids methods. The advantages of representation by using k-medoids are no limitations on attributes types and the choice of medoids is assigned by the location of a predominant fraction of points inside a cluster which result it less sensitive to outliers. In k-means a cluster is represented by its centroid (points within a cluster with average weight usually). The usage of centroid to represent cluster will have the advantage such as convenient to use with numerical attributes data but its disadvantage is sensitive to outlier [10].

Density-Based Algorithms [10] uses the idea that, an open set in the Euclidean space can be divided into a set of connected components. The implementation of this idea for partitioning requires concepts of density, connectivity and boundary. A cluster is defined as a connected dense component and it is able to grow in any direction that the density leads and because of this behavior density-based algorithms are able to discover clusters with random shapes and it is protected against outliers. According to [10] there are some problems which unable to handle by partitioning relocation clustering, but the problem are well solved by this density based algorithm. The advantage of density-based algorithms comes with certain inconveniencies such as a single dense cluster may consist of two near areas with obvious different densities and another drawback is lack of interpretability between clusters. According to [10] there are two major approaches for density-based methods which are Density-Based Connectivity Clustering and Density Functions Clustering.

Grid-based methods [10] is another method of clustering which also uses the concepts of density, connectivity and boundary but manipulate in different way compare to the Density-Based Algorithms. The way that Grid-based methods to manipulate is to inherit the topology from the underlying attribute space. The different between grid-based method and partitioning-based method is that the partitioning-based method uses relocation method while grid-based method uses space partitioning instead of data (relocation method). One advantage of space partitioning is it makes grid-based clustering techniques independent from data ordering.

Clustering algorithms which are used in machine learning can be group into two categories of pattern recognition system which is supervised and unsupervised method. The supervised methods required a number of training samples for each class for training while for unsupervised method; training samples are not available. [1]

According to [1] neural network models are powerful and reasonable alternative for conventional classifier because of the massive parallel processing ability poses by neural network. Neural network classifiers perform better then conventional classifier because neural network classifier offer a higher degree of robustness and fault tolerance. Various learning algorithms for neural network models can be used in supervised classifier. Pattern recognition is a process that maps an input feature vector to the output class membership space, this process is a nonlinear process therefore neural networks is suitable to perform pattern recognition [1]. Figure 5 presented a block diagram of a neural network classifier according to [1].

Neural network classifier could be implemented in several ways such as Single Layer Network (which works well if the classes are linearly separable), Multilayer Network (are used more complex decision where the classes are not linearly separable), Radial Basis Function Networks (introduces a set of basis functions that can be apply for interpolation), Probabilistic Networks (neural network models based on Bayesian classification), Hopfield Network and Hamming Network. All this neural network classifier is under the category of supervised classifier [1].

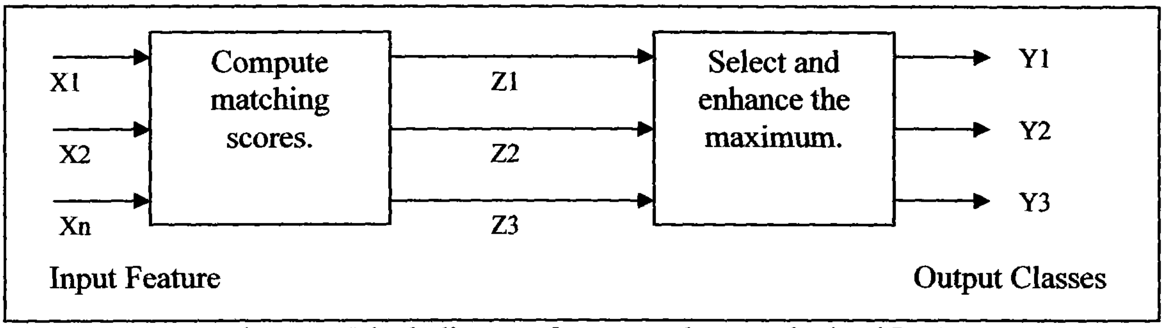


Figure 5: Block diagram for a neural network classifier [1]

Fuzzy c-means clustering is under the unsupervised classifier. Fuzzy logic is another type of machine learning method, it allows partial memberships. The different between conventional clustering algorithm and this fuzzy logic clustering method is that conventional clustering only allow one input assigned to one cluster while fuzzy logic clustering method allow a input to be categorize in to more then one category. For examples a fraction of an image consist a group of pixels might have some pixels belong to one category and other pixels belong to another category. With this algorithm, the fraction of that image will have the membership of the both category [1].

Neural network models are able to perform task such as numeric classification, optimization, and associative storage and recall while fuzzy logic models are able to process inexact or approximate data. There are many ways to combine neural network models with fuzzy logic. This combination are called fuzzy neural. Fuzzy neural network model have greater representation power, higher processing speed, and more robust then conventional neural networks [1]. The examples of fuzzy neural network models are such as Fuzzy Neural Network with Fixed Membership Functions, Fuzzy Neural Network Model with Adaptive Membership Functions, Adaptive Neuro-Fuzzy Inference System, Fuzzy Adaptive Learning Control Network, and Fuzzy Neurons [1]. All of these combinations have their advantages and difficulty to implement.

2.3 Neural Networks Models

2.3.1 Introduction

The human brain has the ability of learning by experience; generalize previous experience from previous problems to make decisions on new problems. This is because the human brain consists of cells called neurons. There are hundreds billions of neurons interconnected with each other and capable to receive, process and transmit electrochemical signals over the brain by using the neural pathways that make up the human brains. According to [2], from the Hebb theory, repeated firings across a synapse will eventually increase its sensitivity and the likelihood of firing in the future will also increased. Therefore if there is a particular stimulus that repeatedly stimulates a group of cells, those cells will be associated strongly together. If in the future the same stimuli are being encounter again, it will likely to trigger the same neurons to fire so it will result the stimuli being recognized.

Although the human brain have a lower computing speed compare to electronic circuits but the ability of human brain to understand and solve vision and language problems are much faster compare to computers. Therefore, the neural network models are biologically inspired by the nature of neurons in the brain. The neural network models are created by trying to mimic the human brains. According to [1], Hebb provided the learning law for artificial neural network that became the beginning point for the early success of neural network models in 1949. But after Minsky and Papert analyzed the earliest single layer neural network models in 1969 and pointed out that such model were incapable of solving many simple problems (theoretically), they discouraged many researchers. Although there was discouragement, there were a few researchers that continued their researched on neural network models such as Teuvo Kohonen and John Hopfield. After several years, there were improvements include the network architecture and learning algorithms and these improvements were then implemented in applications and attracted corporation to commercialize the applications.

The advantages of using neural network models according to [1] are that neural network models can be implemented in many ways such as electrically, optically, or electro-optically or even can be implemented on a personal computer. Besides that, neural network models are fault tolerant and robust, work in parallel and many training algorithms are available. These training algorithms are such as backpropagation, kohonen feature maps, competitive learning, hopfield network and counterpropagation network.

2.3.2 Survey of available neural networks applications

According to [3], neural network models are usually implemented in a few areas such as computer vision, speech recognition, signal analysis, robotics, expert systems and scheduling. As more improvement are made to neural networks models architecture and learning algorithms, neural networks models will be able to be implemented into more areas. Generally neural networks model are used in data mining, matching and clustering. Below are the examples of applications that use neural network models from [3].

Computer Vision: One of the neural network applications is character recognition, The Nestor Learning System™ claim that their recognizer has the ability to recognize approximately 2,500 Japanese handwritten characters with 92% accuracy, recognize handwritten zip code with 98% accuracy and verify signature recognition with 4% false reject rate. Besides implemented in character recognition, neural network was also implemented in face recognition system, image compression, object recognition, edge detection, data classification and biometric recognition.

Speech Recognition: Neural network were implemented in speech recognition for several purposes such as to convert text to speech and speech to text application, and also use speech as a biometric. The Phonetic Typewriter and Nettalk are two of a few successful applications that able to convert text to speech (aid the keyboard to enter words) and speech to text (to enhance the accessibility of computers). A speech recognition system will automatically extracts speeches and store in the memory and to be used again for matching. This speech recognition process can be implemented with Recurrent Neural Networks (RNN), Radial Basis Functions (RBF) and Vector Learning Quantization (LVQ).

Signal Analysis: Signal analysis is one of the largest neural network models research areas. Neural network is used because radar is required to perform tracking, recognizing, and classifying on an object by analyzing signal that receive by a receiver. Besides that, neural network are used in radar technology because a radar system must handle a lot data such as image angles and targets which required long computational time by conventional algorithms and neural network are able to solve the problems with its generalization and parallel processing characteristic. Helicopter recognition for smart weapons, radar target tracking, classification and recognition, and sonar classifier are the examples of radar that use neural network technology.

Robotic: A robotic movement can be divided in to Autonomous Vehicles and Manipulator Trajectory Control. The Autonomous Vehicles work as the robot decision module, it make decision of the robot movement based to the input provided by sensors. Manipulator Trajectory Control is used to control the robot's kinematics, to design a manipulator control is difficult and time consuming with conventional programming. The adaptability and generalization characteristic of a neural network is able to solve the problem.

Expert System: The different between a neural expert system with other expert system is that the neural expert system does not require a knowledge engineer to formulate rules. The neural expert system is applied in medical services and financial services. One of the example of medical services expert system is the Saito's Medical Diagnostic Expert System which able to diagnose 23 diseases from 216 symptoms with the 67% accuracy after 300 examples training. Nestor's Mortgage Origination Underwriter is one of the financial services expert system is a system that determines a mortgage loan application based upon the applicant's information.

2.3.3 Survey of available applications that perform clustering with neural networks

In the data mining field on of the biggest problem it to organize and retrieve information from storage. Clustering is one of the methods that able to help organize information, reduce time and ease information retrieve process. In the process of clustering similar information has to be grouped together, the major problem is to obtain and clearly define the similarity of the information (so that similar information can be grouped together). This clustering problem can be solved by using neural network because of its ability to learn. Therefore, many applications had used neural network to perform classification and clustering [8].

Neural network models are able to perform recognition, besides that neural network models are also used to perform classification especially on classifying images such as classifying patterns and characters. This is because the advantages of learning and parallel computing. The ability of classifying data and images makes neural network models appropriate to perform clustering process. Examples of applications that use neural network to perform clustering are such as fingerprint classification and optical pattern classifier system.

Fingerprint classification will be elaborate in the next section. The optical pattern recognition system which include the optical character recognition system could be consider as application that perform classification because these systems perform recognition on images and make decision on which categories that the image belongs, such as square or round. For the optical character recognition system is the same, it need to classify which category (alphabet) that an image of alphabet belongs to.

Other than that, neural network clustering is also being applied in other fields such as biology; biological sequencing [7]. There are enormous amount of data needed to be processed by scientists, therefore automated techniques need to help scientists to analyze, understanding, and clustering. The ability of neural network of being able to learn and can perform task without predetermine rules (expert's knowledge) makes neural network suitable for the process of analyzing and sequencing. For examples, neural network had been used to identify the relevant genes that responsible for certain distinctiveness from DNA. [7]

2.3.4 Survey of available applications that perform fingerprints clustering with neural networks models

The architecture of neural network model that study by [6] uses edge detection to create eigenvector from a given fingerprint image, then Kohonen Loeve transform (KLT) are then being applied to reduce the dimension of the input from the eigenvector, before feeding it to a multi layer perceptron. All weight of the multi layer perceptron is fully connected. The neural network model was studied by [6] with different number of nodes at the input and the hidden layer, Figure 6 shows the architecture of the neural network model. Table 1 is the result obtained by them:

Table 1: Testing Percentages Correct for Various Architectures [6]

inputs	*hiddens*				
	32	48	64	80	96
32	90.29	89.82	90.96	90.24	90.99
48	90.54	90.24	91.34	91.12	90.92
64	90.66	90.48	91.20	91.19	91.76
80	90.60	91.34	90.60	90.27	91.18
96	91.07	90.65	90.40	91.28	90.37

The conclusion obtained by [6] was that better results can be achieved by implement prior probabilistic of different class to the neural network model and expand the training set and testing set.

According to [5] Probabilistic Neural Network (PNN) was proved to perform better to solve fingerprint classification problem compare to other neural network such as

Radial Basis Function and Multi Layer Perceptron (MLP) methods. The proposal of [5] is to enhance the performance of neural network performance for fingerprint classification by altering the network dynamics. The altering are a changes to the activation function, Boltzmann pruning was applied and prior class probability was used.

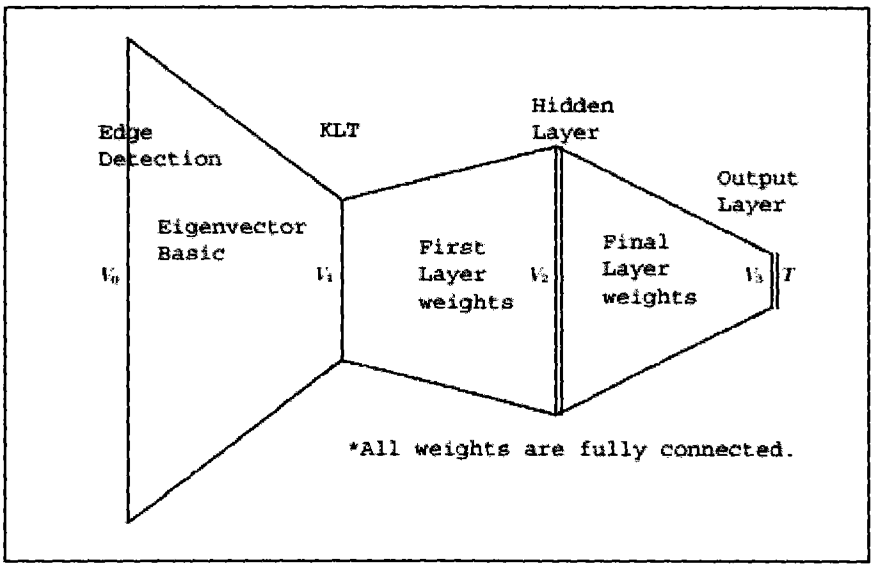


Figure 6: Classification Architecture of Neural Network Model [6]

According to [5] survey, there are research on combining PNN with MLP together but it performance was only equally to PNN and are very expensive to train, slower and less efficient than conventional MLP systems. Comparisons were made between PNN, MLP with sinusoidal activation function and MLP with sigmoidal activation function by [5]. The result of the comparison according to [5] is that MLP with sinusoidal and PNN are able to out perform each other at certain case, but MLP with sinusoidal was prefer because of its simplicity.

2.4 Fuzzy Logic Models

2.4.1 Introduction

Fuzzy Logic (FL) is a powerful problem solving methodology in the past few years. It shows a rapid growth in the applications of FL, especially in the image-understanding applications, such as edges detection, feature extraction, classification and clustering [1]. FL provides a simple and easy way to draw a definite conclusion from ambiguous, imprecise or vague information. The adaptation of FL is mimicking the human decision making, which provide a precise solution from approximate data.

The FL technology give a strong impact to conventional classical logic applications which requires a deep understanding of whole system, the exact equations of formula and precise numeric values. All the precision and certainty in the classical logic might carry a cost. The FL gives tolerance in decision making process, which allows construction of a complex human decision making system with only using a higher level of abstraction originating from our knowledge and experience with subjective concepts [11]. The subjective concepts are the description of conditions or concrete things such as very, little, long, many and others.

With FL, we can apply rule based decision making in terms of words rather than numeric figure. It is more similar with the way of human thinking which generate solution from the expertise experiences (represent in rules), knowledge and even skill. Although we know that the rule-based systems have a long history in the field of Artificial Intelligence (AI), but the conventional rule-based system have difficulty to tackle the fuzzy consequents and fuzzy antecedents' problems [1]. The if-then rules in FL will combine with an inference engine to be more flexible and accurately to solute the problems mentioned above.

Furthermore, the past research show that FL in practical applications able to give best performances due to its simplicity in the design of algorithm, lower cost and high productivity. Because of the advantages just mentioned, we will provide some examples of FL applications, clustering in specific and also fingerprint clustering as well at the below. It helps us to more understand the practical applications of FL in the fingerprint clustering compare to other fields. And it might provide us a framework of fingerprint clustering from the modification of others successfully implemented FL engine as well.

2.4.2 Survey of available fuzzy logic applications

From the official web page of fuzzyTECH [12], we find that the FL technology has very good results in two main application fields, which consists of industrial applications and business and finance applications. Those FL applications can categorize under automated control or decision-making support. Some of the common applications of FL will describe below.

The common used of FL in automated control industrial applications is due to three main factors: complex design of control systems and involve multiple parameters, the optimization of most system based on engineering expertise rather than mathematical methods, and the competitive automotive engineering on an international scale [13]. In another hand, FL has been widely used in risk assessment and can be used to help investors evaluate data in the field of business and finance applications [14]. Just like automated engineering, in market investment, a manager may use his knowledge, which consists of a lot of rules, and also his experiences and skill to analyze the investment situation. FL will provide a mechanism to users, especially when rules and experiences are important in problem solving.

Adaptations of FL in multiple application fields show a good inspire result. Just take an example of FL application in anti-sway control of cranes. The benefit was a capacity gain of about 20% due to the faster transportation and an increase in safety [12]. In fire zone control in waste incineration plants, The FL controller was capable reduce the fluctuation less than ±1 Mg/h [15]. Another example is dosing control in waste water treatment plants, the adaptation of FL resulted in savings of about 50% of the chemistry resources compared to the manual control before [12]. It also showed very good results in a very short engineering time at less than 10% of the costs of a conventional solution for control of tunnel inspection robots [12], design time of positioning in presses only require a third of conventional approach [16], temperature control in plastic molding machines using FL gives faster response time and a significantly smaller overshoot combined with extreme robustness [16], saves about 25% on electrical energy, equivalent to the amount of $50,000 per year in climate control and building automation [12, 17] and many more.

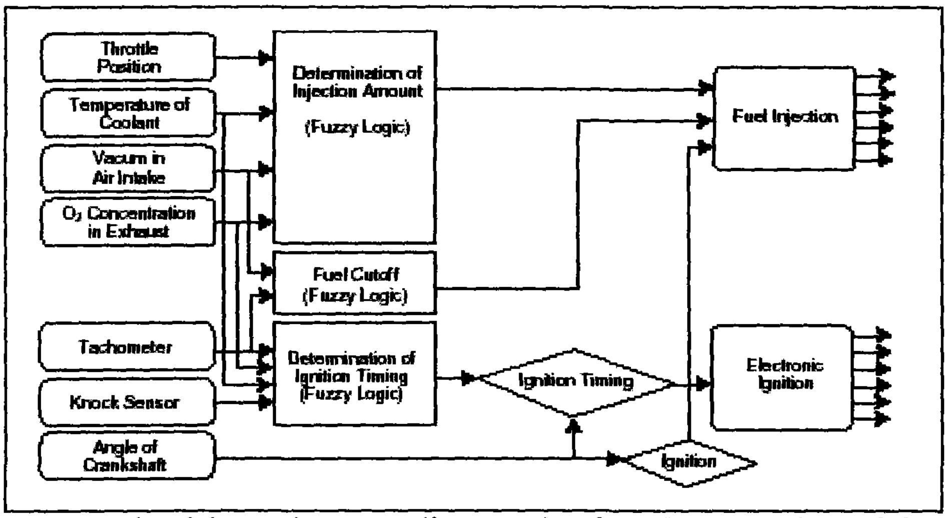


Figure 7: An example of the engine controller contains three Fuzzy Logic modules [13]

Other currently available applications of FL in automated control consists of automatic control of dam gates for hydroelectric-power plants, wind energy converter control, camera aiming for the telecast of sporting events, efficient and stable control of car-engines, cruise-control for automobiles, positioning of wafer-steppers in the production of semiconductors, back light control for camcorders, automatic motor-control for vacuum cleaners with recognition of surface condition and degree of soiling, single button control for washing-machines, flight aid for helicopters, software-design for industrial processes, controlling of machinery speed and temperature for steel-works, controlling of subway systems in order to improve driving comfort, precision of halting and power economy, improved fuel-consumption for automobiles, improved sensitiveness and efficiency for elevator control, and improved safety for nuclear reactors [18].

Furthermore, FL applications in decision-making support systems included substitution of an expert for the assessment of stock exchange activities, optimized planning of bus time-tables, archiving system for documents, prediction system for early recognition of earthquakes, medicine technology like cancer diagnosis, recognition of handwritten symbols with pocket computers, recognition of motives in pictures with video cameras, compensation against vibrations in camcorders, simulation for legal proceedings, and recognition of handwriting, objects, voice [18].

2.4.3 Survey of available applications that perform clustering with fuzzy logic

"...One important limitation of classification of statistical approaches to land cover mapping is that the output derived consists only of the code of allocated class. This type of output is often referred to as being 'hard' or 'crisp' and is wasteful of information of the strength of class membership generated in the classification..." [19]. Conventional classification approaches therefore may not provide a realistic or accurate representation of land cover [20]. From the two statements above, we find that the conventional clustering algorithms use crisp memberships for allocating samples to clusters. In this section, we will introduce FL technology which allows the advantages of using partial memberships for clustering in variable application fields.

The concept of multiple and partial class membership is fundamental to fuzzy sets techniques and has been well adapted in many practical applications such as river water quality classification, classification of gene expression data, land cover representation and others. The fuzzy classification is mainly categorized to unsupervised fuzzy classification and supervised fuzzy classification.

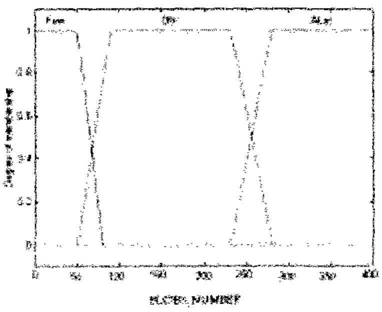
One of the most well-known unsupervised fuzzy clustering algorithms that allow fuzzy memberships is the fuzzy-c-means (FCM) clustering algorithm. FCM will represent membership values from the range 0 to 1, which indicate relative strength of class membership a pixel has to each class may be derived [21]. The FCM has been applied in many practical applications, such as magnetic resonance imaging (MRI) data analysis [1].

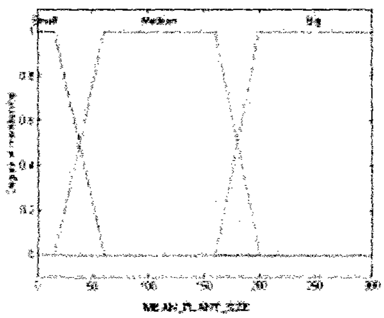
Normally, there are two major steps in applying an algorithm for fuzzy supervised classification: estimation of fuzzy parameters from fuzzy training data which generated by applying statistical methods, and fuzzy partition of spectral space which can be recorded in a fuzzy partition matrix [21]. From the paper [19], training information and classification results are represented in a one-pixel-one-class method for the conventional remote sensing supervised classification and a very encouraging result have been obtained for the overall classification accuracy.

Besides MRI data analysis, there are many available fuzzy clustering paper had been published such as unsupervised fuzzy video content characterization and shot classification [22], FCM for image segmentation in the presence intensity in homogeneities [23], fuzzy classification of gene expression data [24], gammas fuzzy clustering [25], database schema and query language fuzzy classification [26], generic edge features fuzzy clustering [27], fuzzy decision system for threshold selection to cluster cauliflower plant blobs from fields visual images [28], fuzzy image classification of land use map accuracy assessment [29], river water quality fuzzy classification [30], ecological impact fuzzy sets classification [31] and many more.

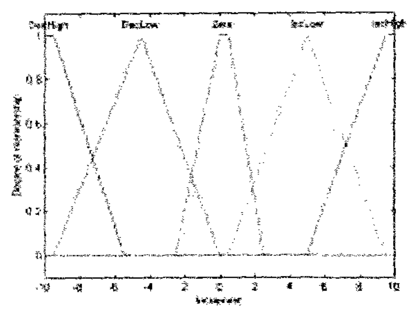
Just take the paper [28] as an example of fuzzy clustering in a reasoning system. It shows that a FL clustering algorithm able to rejoin the fragmented blobs of the plants and recover the cauliflower image to identify the plants position. This technique is very valuable in autonomous agricultural tasks which implemented for real time outdoor scenarios. And the performance in successfully identify the location of cauliflowers is as high as 94.8%. We find that all image processing has been performed on gray level in this paper. So we hope the implementation of FL clustering in fingerprint will give a roughly similar result although the fingerprint image is known more complicated than cauliflower images. Below are the detail of the input variables, output variable and production rules for the fuzzy inference engine using in the reasoning system.

The first input variable is BLOBS_NUMBER, which correspond to the number of blobs in the image. Very low or very high values indicate too high or too low thresholds intensity values, respectively. Thus this variable has been modeled as a fuzzy variable with three linguistic labels: {Few, OK, A lot}, where each of them is represented by a trapezoidal membership function. The second input variable is MEAN_PLANT_SIZE. To represent this parameter the mean value of the ten biggest blobs in the image has been used. The three linguistic labels for this variable are: {Small, Medium, and High}.





Whereas for the out variable is the increment or decrement that must be applied to the previous threshold. There are five trapezoidal membership functions as shown below.



In the production rules model, the classical choices for the conjunction and disjunction operations with logic expressions, the maximum and minimum respectively is implemented. The sample of knowledge base is shown in Figure 8. Two antecedents (BLOBS_NUMBER, MEAN_PLANT_SIZE) must be satisfied to fire the consequent (increment or decrement). For the defuzzification process, the gravity centre algorithm is performed. The fuzzy rules computing time is slightly over the second, as the algorithm is implemented using Matlab [28].

1. IF BLOBS_NUMBER is Few & MEAN_PLANT_SIZE is Small THEN INCREMENT is DecrementHigh
2. IF BLOBS_NUMBER is Few & MEAN_PLANT_SIZE is Medium THEN INCREMENT is DecrementLow
3. IF BLOBS_NUMBER is Few & MEAN_PLANT_SIZE is High THEN INCREMENT is DecrementLow
4. IF BLOBS_NUMBER is OK THEN INCREMENT is Zero
5. IF BLOBS_NUMBER is ALot & MEAN_PLANT_SIZE is Small THEN INCREMENT is IncrementHigh
6. IF BLOBS_NUMBER is ALot & MEAN_PLANT_SIZE is Medium THEN INCREMENT is IncrementHigh
7. IF BLOBS_NUMBER is ALot & MEAN_PLANT_SIZE is High THEN INCREMENT is IncrementLow

Figure 8: An example of Rules set [28]

Sometimes, we found that the FL clustering will not provide a better result than conventional approaches such as [24]. This situation is believed normal because the implementation of FL sometimes unable correctly classify the giving cases in both the logistic regression and the fuzzy models. It tended to make the same errors or non-commitments as the logistic regression model. We will face difficulty in representation the multiple conditions with rules set which is translated directly from linguistic form. So the implementation of FL clustering is not totally guarantee return an acceptable result.

2.4.4 Survey of available applications that perform fingerprints clustering with fuzzy logic models

After analyze the fuzzy application and fuzzy clustering in variable fields, we might wonder how far the performance of fuzzy on fingerprint clustering as well. The application of FL in biometric especially fingerprint recognition is not a new invention but it still carry out lots of problems in the performances. Normally, FL technology is applied for features extraction only in fingerprint recognition system. The example of feature extraction techniques is shown in Fuzzy Feature Selection for Fingerprint Identification [32] and Fingerprint Feature Extraction by Fuzzy Logic and Neural Networks [33].

The adaptation of FL in fingerprint features extraction gives a reasonable and acceptable result, but in other hand, we face limitation references for the fingerprint clustering using FL. The problem may due to the technology itself not suitable for fingerprint clustering or there are others more suitable techniques to apply. So, we hope to implement a pure fuzzy clustering for fingerprint to evaluate the performance in the time consuming for clustering and the accuracy in the clustering as well.

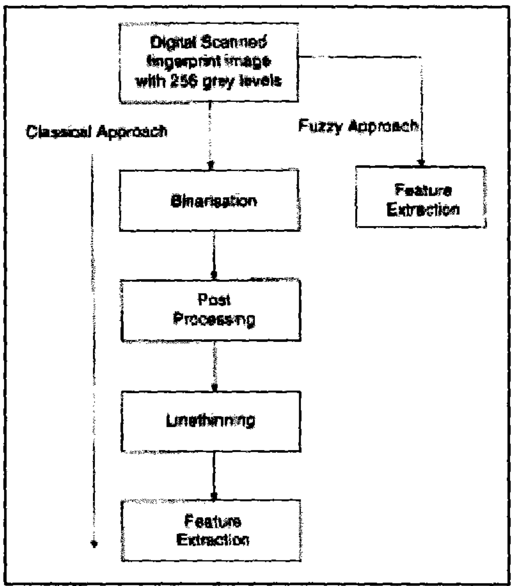


Figure 9: Comparison of fingerprint feature extraction between classical approach and fuzzy approach [32]

2.5 Neuro Fuzzy Models

2.5.1 Introduction

From our understanding after survey, we found that there are many ways to synthesize Fuzzy Logic (FL) and Neural Networks (NN). The final product of the combination of fuzzy logic and neural networks often recognized as "Neuro Fuzzy" (NF) technology, which combines the advantages of the two technologies. In the knowledge representation part, NN is more implicit because the NN system cannot be easy interpreted or modified whereas FL more on explicit due to the easiness and efficiency in verification and optimization [12]. For trainability, FL shows difficulty in learning because we have to define everything explicitly while the NN is more "clever" because NN trains itself by learning from data sets [12].

The NF systems often provide greater representation power, have higher processing speeds and are more robust than conventional pure FL or NN systems [1]. From the book [1], it shows four examples on how to synthesize FL and NN as a practical NF engine. We know that it only a part of the possibility of the combination of FL and NN because it still remains many ways to do that so.

The first approach of synthesizing is to use input-output signals or weights in a NN as fuzzy sets along with fuzzy neurons. The fuzzy neurons models had been proposed by many authors. The second approach is to use fuzzy membership functions to preprocess or post process data with NN models. Meanwhile, the third approach will build a multiple stages classifier and each technology (FL and NN) have some stages to implement. Another possible approach is to use fuzzy associative memories (FAM) which refer to FL rules with some associated weight. Then a mathematical framework will map FAM to NN [1].

Year:	Computing:	Neural Networks:	Fuzzy Logic:
1940	Relay/Valve Based		
1945		Neuron Model (McCulloch Pitts)	
1950	Transistors	Training Rules (Hepp)	
1955			
1960	Small Scale Integration	Delta Rule (Widrow Hoff)	
1965			Seminal Paper (Zadeh)
1970	Large Scale Integration	Multilayer Perception, XOR	
1975			Fuzzy Control (Mamdani)
1980		Hopfield Model (Hopfield Tank)	
1985	Artificial Intelligence	Backpropagation (Rumelhart)	Broad Application in Japan
1990		Bidir. Assoc. Mem. (Kosko)	Broad Application in Europe
1995			Broad Application in the U.S.
2000			
		Soft Computing, NeuroFuzzy	

Figure 10: Convergence of fuzzy logic and neural networks technologies [34]

There are many NF models use fuzzy membership functions for prepocessing have been developed [1]. For example, Lin and Lee (1991) suggested a five layers NF model, which consists of input/linguistic nodes, two layers of term nodes and also nodes for represent fuzzy rules. Another author, Horikawa et al. (1992) also proposed some NF models. The proposed models used sigmoid fuzzy membership functions; middle part consists of five layers corresponding to the premise fuzzy rule and three types of consequent part (constant, first-order equation or fuzzy variable). Besides, Takagi and Hayashi (1991) proposed a model for NN-driven fuzzy reasoning, Pal and Mitra (1992) proposed NF model using back-propagation learning algorithm, Jang and Sun (1995) presented an adaptive network model for a fuzzy inference system and many more.

2.5.2 Survey of available fuzzy-neuro applications

NF provides a powerful tool to design intelligent systems. As mentioned above, there are many ways to combine FL and NN to adapt in many practical applications. NN is useful in tuning the fuzzy rules of a fuzzy inference system. From our survey, we found that majority NF models have been used in computer vision applications. Of course, NF technology itself is more powerful than that and below we will describe some of the practical applications of NF and also the results.

A hierarchical NF structure able to adapt for a speaker-independent speech recognition system [35]. A great number of different individual speech templates of Chinese digits 0-9 are collected as the testing samples. Although there are heavier noises around but the system still remain a high recognition rate of 92.2% which increases by 5-6% of the previous system. It shows the powerful of NF techniques in acoustic applications.

Some author also shows that NF gives a good result in medical diagnostics [36, 37]. An example of medical application of NF techniques is used to diagnose Heart Rate

Signals [38]. The methodologies of NF and evolving fuzzy neural networks (EFuNN) are applied to heart rate variability. The NF have several advantages, included robust to catastrophic forgetting, both statistical and knowledge engineering tools, interpolate and extrapolate well in regions where data is sparse, and accept both real input data and fuzzy input data. The final result show NF model produces a good classification rate for the experimental classes of heart disease status if there is sufficient data for training the model [38].

In decision making application, someone tries to apply NF to check authorization from incomplete information [39]. A simple benchmark case was established and compares with others techniques like multilayer perceptron, polynomial neural networks, and fuzzy decision model. An overall improvement of at least 10% was obtained from the application of NF method.

Data redundancy is a central issue in image compression which consists of coding redundancy, interpixel redundancy and psycho visual redundancy. We find a paper of application by Evolving Fuzzy Neural Network (EFuNN) for Compressed Video Parsing [40]. The EfuNN is used to compress MPEG video parsing. The EFuNN model learns from pre-classified examples in the form of motion vector patterns in order to distinguish between six classes: static, panning, zooming, object motion, tracking and dissolve. The result shows the EFuNN model has a high classification accuracy and fast training.

Besides the example of applications just mentioned, NF technology also commonly apply in remote sensing, control application, data mining and computer vision, biometric applications, character recognition, knowledge-based pattern recognition, stereo vision, image data compression and medical image processing [1]. Some authors also apply it in prediction of concrete fatigue durability [41], real-time adaptive control of musical processes [42] and also behavioral representation in computer Generated forces [43]. As conclusion, NF model commonly shows an acceptable result but some suggested an algorithm is believed can be better with several adjustments in the future. For example, apply back-propagation in NF model can make the engine more "clever" in the learning process.

2.5.3 Survey of available applications that perform clustering with neuro fuzzy

Commonly, NF models include fuzzy competitive learning, fuzzy ARTMAP and fuzzy linear vector quantization (LVQ) [1]. In these models, preprocessors will be used to fuzzy the input values. Also, we can assign the input vector to multiple categories, and updated the weights by using partial membership values. With the ability of learning, the NF model has been used in many practical applications.

For an example, a NF approach is applied to context-sensitive feature selection in aerobic fitness classification [44]. A fuzzy preclassifier and multistructure feed forward neural network is used. The same feature sets are used in the different structures, but the neurocalculation is different in each structure due to different synaptic weights. The final classification output is selected from the multi-network structure using the fuzzy preclassifier's output as a decision criterion. The proposed method improved over 10% of the classification accuracy.

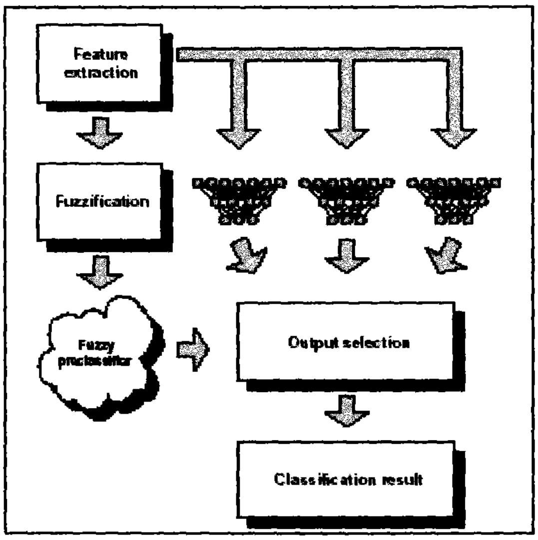


Figure 11: The principle of fuzzy preclassifier [44]

Another practical application has been carried out to use NF approach to classify, evaluate and forecast the agriculture condition [45]. The system is developed to against the agriculture infestation and the NF engine will suggest suitable treatment for each monitors farm. The neuro-fuzzy methodology is applied because it allows a large use of infestation dates with a good flexibility degree. After validating with experimental data, the classes of infestation and treatments is reduce in clustering, which make the monitoring of farm condition become easier.

Although NF clustering shown applicable in different practical application fields. But we still wonder it capabilities in the complicated tasks, especially biometric applications. Fingerprint analysis, clustering and recognition itself is a complex task, and the computer vision systems have not been able to completely solve the problems related with them [1]. After all the survey have been gone through, a modified NF model hope to be proposed to enhance the currently available engine, which might helps us in the fingerprint clustering system later.

Chapter 3: Problem analysis

In order to enhance our fingerprint recognition system, we proposed to implement and compare the accuracy and efficiency of fuzzy neural clustering algorithm, pure neural network clustering algorithm and pure fuzzy logic algorithm to simplify the process of matching in our fingerprint recognition system process. There are three major problems of clustering have to take into account before we are able to start design and implement the three clustering methods. The three problems are to figure out number of clusters for the fingerprint clustering process, what are the standard (accuracy and efficiency) that required to be achieved by the clustering method, and the third problem is how this classification is going to solve the transition, scale and rotation problems. We had proposed the solutions for the three problems, but before we elaborate on our solutions, we have to figure out the input for these clustering methods.

We propose to build a module that responsible to convert the fingerprint pattern from a 256 x 256 pixels grayscale image into a 256 columns array. The grayscale image will first be divided into 256 blocks. Each of this block's direction will be read and store into the 256 columns array. These directions are obtained by reading the fingerprint pattern according to the ridges and valleys of the fingerprint. These directions are divided into six categories (90°-270°, 0°-180°, 30°-210°, 60°-240°, 120°-300° and 150°-330°) and every direction will be represented with the numbers from 1 to 6. As the end result the 256 columns array will be the input for the three clustering methods.

The first problem is the number of clusters that should our fingerprint clustering system has. From our study we know that fingerprint can be classified into approximately seven different types such as arch, tent arch, loop, double loop, pocked loop, whorl, and mixed figure [46]. We proposed that our fingerprint clustering system should have five and these five categories are whorl, right loop, left loop, arch and others. We proposed these five categories because according to our study the whorl, left loop, right loop and arch are the majority of fingerprint pattern and the "others" category is to serve the purpose of classifying fingerprint pattern that are different from the four categories. For examples, a fingerprint image has mixture feature of a right loop and a tented arch.

The second problem is to define the standard that we expect from our fingerprint clustering system. From our fingerprint recognition system without clustering, we only able to handle 150 fingerprint images because of low matching efficiency. Therefore we proposed to apply clustering methods in our fingerprint recognition problem. We expect that the clustering methods will able to double the current matching efficiency and able manage more fingerprint images approximately 250 images. This enhance efficiency will able to achieve only if the clustering algorithm are able to have 90% accuracy and clustering can be perform within one second time. Therefore neural network, fuzzy logic and fuzzy neural classifier will take this as a standard to be achieved.

The third problem is the problem of transition, rotation and scale. These problems is not a new problems, previous fingerprint recognition system had solve these except scale problem by use finger code for matching. For the moment we need more experiments to have a clear idea on solving these problems. We still do not know how good the direction reader (the module that responsible to convert the fingerprint pattern from a 256 x 256 pixels grayscale image into a 256 columns array) performs on a scaled image. These problems will also make the training process for neural network classifier become more difficult because more testing and training are required. These problems also make fuzzy logic more difficult to be developed because more rules are needed in fuzzy inference engine to support the classifier decision.

After all this is only the clustering problems, there are also equally problems in designing and implementation of neural network, fuzzy logic and fuzzy neural. I will be building the neural network classification and my teammate will be responsible in developing a fuzzy logic classifier. After both of us succeeded building the neural network and fuzzy logic classifier, we will be working together to develop a fuzzy neural classifier. The objective of developing these three methodologies is to make comparison among them so that we can know which on is the best.

The problem of developing the neural network classifier will be the design of the architecture of the neural network structure. From my current understanding, if the structure have too many hidden nodes there will be enormous weights that needed to be handled and this might slow down the classifying process and if the hidden layer nodes is too little, this might cause the neural network models fails to perform classification, therefore in order to find the balance between these two constraints, several experiment must be performed. The major problem of developing the fuzzy neural is how to combine fuzzy logic classifier and neural network classifier. Currently we have two conceptual ideas on combining them and these ideas will be discuss in the methodology chapter, we also need to perform more experiments on these idea to confirm whether these ideas can works and make conclusions on them.

Chapter 4: System Overview

4.1 As is fingerprint recognition system

Figure 12 shows the framework of our previous fingerprint recognition system. The process of the system can be divided into two, for enrolment process the system begins with acquire fingerprint image from either computer generated fingerprint or fingerprint scanner, later then the image passes through a series of preprocessing filter so that feature extraction can be performed to obtain the finger code from the features. The finger code is then store in database.

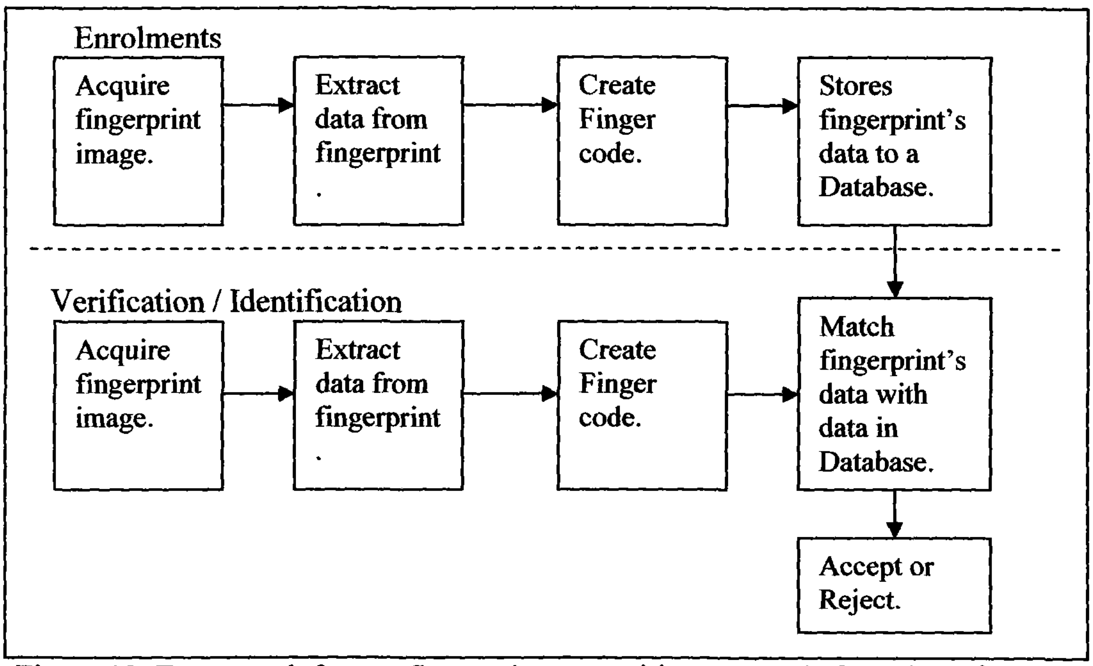


Figure 12: Framework for our fingerprint recognition system before clustering was implemented (as is system)

For the verification and identification process, an acquired image will passes through the process same like enrolment just that instead of storing the finger code into the database, the finger code is then use to perform matching with data (finger code) in the database.

4.2 To be fingerprint recognition system with clustering algorithm

Figure 13 shows that the *to be* fingerprint recognition system will have some modification compare to the previous *as is* system. A direction reader and a clustering module are added to the system. The function of direction reader is to convert the fingerprint pattern from a 256 x 256 pixels grayscale image into a 256 columns array. Then the 256 array is fed to the clustering module to perform classification on the fingerprint image. The result of this clustering module is which class the fingerprint image belongs. The classification result is then store into database with the finger code.

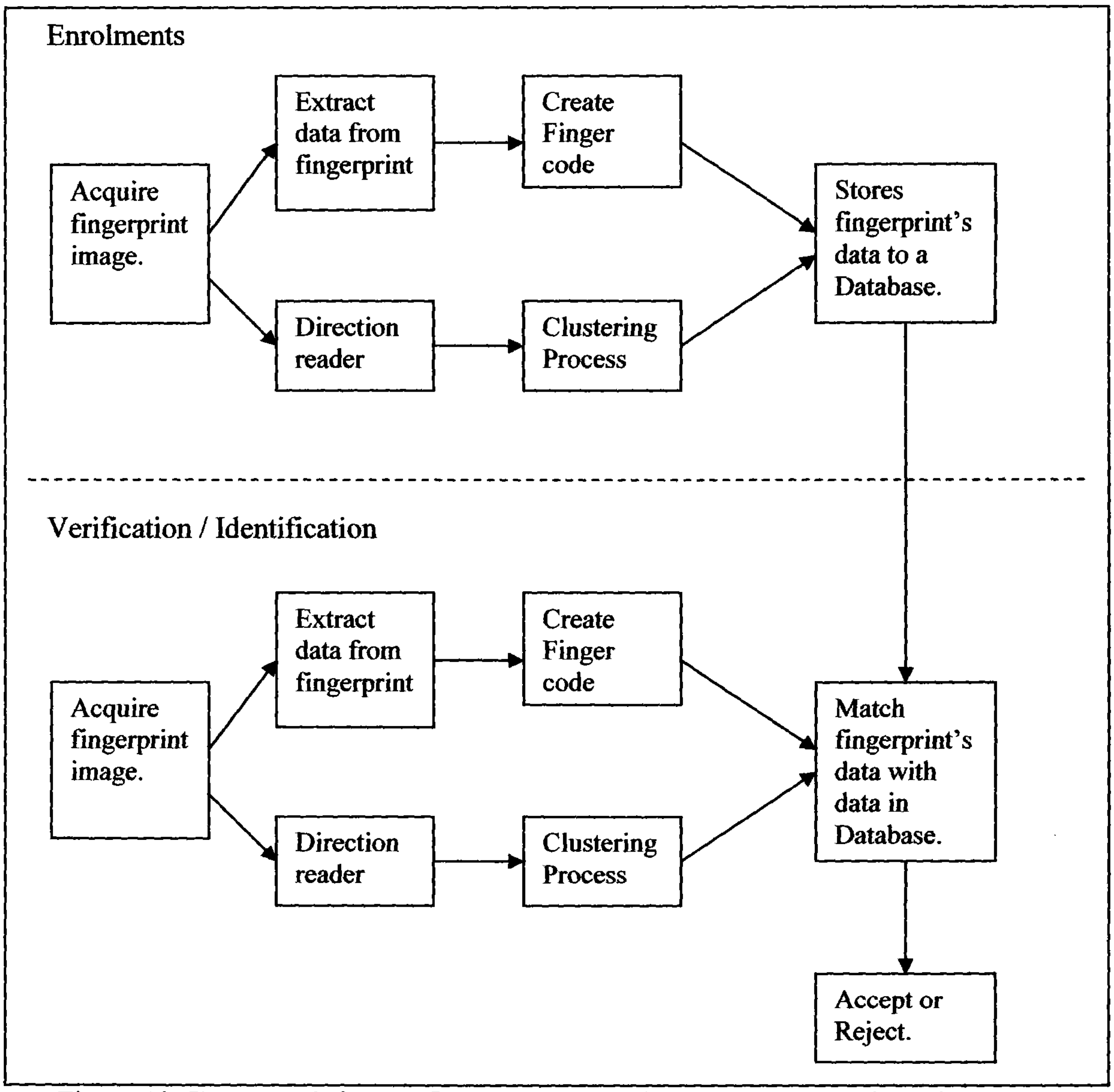


Figure 13: Framework for our fingerprint recognition system after clustering was implemented (to be system)

For the identification process with this *to be* system is slightly different compare to *as is* system. After the fingerprint image being processed for the finger code and classification result, instead of using the finger code to search the database immediately like the *as is* system, the classification result is first use to identify the class that the finger code belongs and then the matching will be perform within the class. This means that the matching does not need to search through the entire process and this will speed up the matching duration.

4.2.1 The clustering process

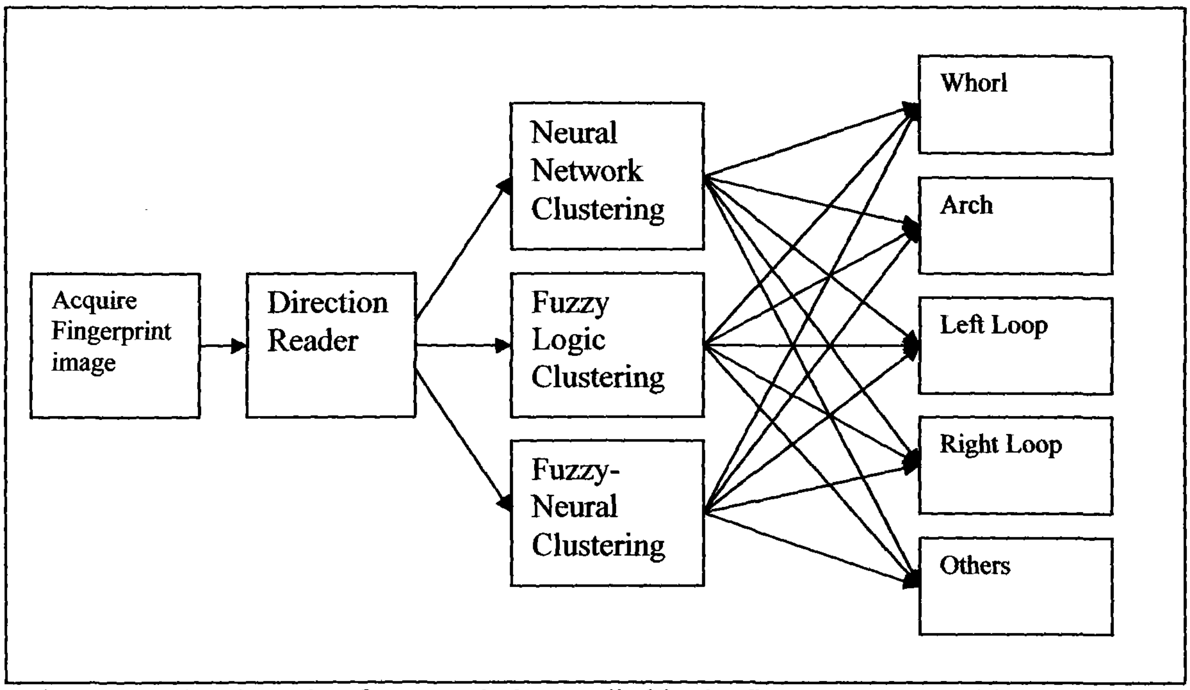


Figure 14: The clustering framework that applied in the fingerprint recognition system.

This clustering framework shows that after direction reading, there are going to be three different classifier methods to classify the fingerprint image (can be chose by user) and after classifying the system will categorize the image into five classes such as whorl, right loop, left loop, arch and others.

4.2.2 Software and Hardware Overview

This fingerprint recognition system with clustering was build with *Borland JBuilder8 Java1.4.1* and *Microsoft Access Database* running under W*indow XP* operating system. The database is used to store fingercode (for matching purposes), neural network's weights, fuzzy logic's rules and fuzzy neural's weights and rules. Besides that, *Fingerprint Generator OPTEL* was used to generate fingerprint images for the system to identify and classifier training. Other software that was also being used during the development period is ACDsee, Adobe PhotoShop 6.0 and Window Magnifier.

Hardware that use by this fingerprint recognition system with clustering development is an IBM personal computer with 1.7 GHz Intel processor and 512 Mb DDRAM. Testing and experiments results of all the classifier in the fingerprint recognition system are also being done on the same computer.

Chapter 5: Methodology

The methods that are going to be applied to solve the fingerprint classification and clustering problem will be the neural network models, fuzzy logic and fuzzy neural methods. I am responsible of developing the neural networking and my partner will be developing in the fuzzy logic clustering. For the neural fuzzy classifier, we worked together to build it with the experience and knowledge that we both gain from developing the neural network and fuzzy logic classifier earlier. Since I am concentrating on developing neural network and fuzzy neural classifier, in this chapter I will allocate more details on neural network and fuzzy neural classifiers.

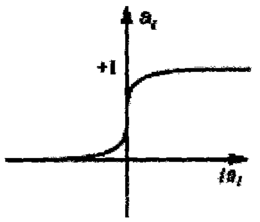
5.1.1 Neural Network Fundamentals

Artificial Neural Network has been chose to solve this problem because of its success in several fields that require pattern recognition for the pass 3 decades. Neural network models are preferred for image processing and recognition because of their parallel processing capabilities as well as decision making and learning abilities. According to [1], neural network models are a good alternative to conventional classifiers. There are studies between neural network classifiers and conventional classifiers. The potential of neural network models is because of the ability of extending beyond the high computation rates which provided by massive parallelism. Besides that neural network classifier are able to provide a higher degree of robustness and fault tolerance compared to conventional classifiers, and currently there are various of learning algorithms can be used to train the neural network classifier.

There are many types of neural network models for classification and every neural net model has own advantages and limitations. The fingerprint clustering system needed to classify images which are non-linearly separable, therefore I proposed to use multilayer networks. According to [1], a multilayer network can be trained with backpropagation learning algorithm. Training with backpropagation requires pairs of inputs and desired outputs. The actual outputs that calculated by the multilayer network will be compared with the desired outputs. If the desired outputs is the same compare with the actual outputs then do nothing else if there are difference then weights are adjusted to reduce the difference. According to [1], this learning method uses a gradient search technique to minimize the cost function. The network updates its weights with iteration to minimize errors gradually, where the initial weight of the network begins by setting all the weights and threshold randomly. The modal with three layers is guaranteed to find the best set of weights but there is a risk of getting stuck in a local minimum.

Figure 16 shows structure of three layer network that are going to be use to perform fingerprint clustering system. I have used 256 input nodes, 10 hidden nodes and 5 output nodes and all nodes are fully connected. Therefore, there are 2610 (256 x 10 + 10 x 5) weights to compute for every clustering process. Initialize value for all the weights randomly between -1 to +1.

There are several activation function such as step activation function, sign activation function and sigmoid activation function. Sigmoid activation function is being used because it has a very well defined derivative function.



$$sig(x) = \frac{1}{1+e^{-x}}$$

$$\frac{dsig}{dx}(x) = sig'(x) = \frac{e^{-x}}{(1+e^{-x})^2}$$

Figure 15: Left is the Sigmoid Function and right is its derivative function.

The directions generated from the direction reader model will be the input for this neural network. Then after setting up the structure of neural network, I used 120 different fingerprint images to train this neural network. This 120 fingerprint images have the mixture of whorl type, arch type, left loop and right loop. Those fingerprint images will be train according to table 2.

Table 2: Desire output of the neural network and predefined cluster representation

If the fingerprint image is a *Whorl* type then the output of the five nodes should be *00001.*
If the fingerprint image is an *Arch* type then the output of the five nodes should be *10000.*
If the fingerprint image is a *Left Loop* type then the output of the five nodes should be *01000.*
If the fingerprint image is a *Right Loop* type then the output of the five nodes will be *00010.*

After the training process the neural network will be able to perform classification on fingerprint images. On any give fingerprint images, the output of the neural network will then be converted to text according to the production rules table 3.

Table 3: Output of the neural network will then be converted to text according to this production rules.

If the output is *00001* then the fingerprint image is a *Whorl* type.
If the output is *10000* then the fingerprint image is an *Arch* type.
If the output is *01000* then the fingerprint image is a *Left Loop* type.
If the output is *00010* then the fingerprint image is a *Right Loop* type.
Else the fingerprint image belongs to *Other* type.

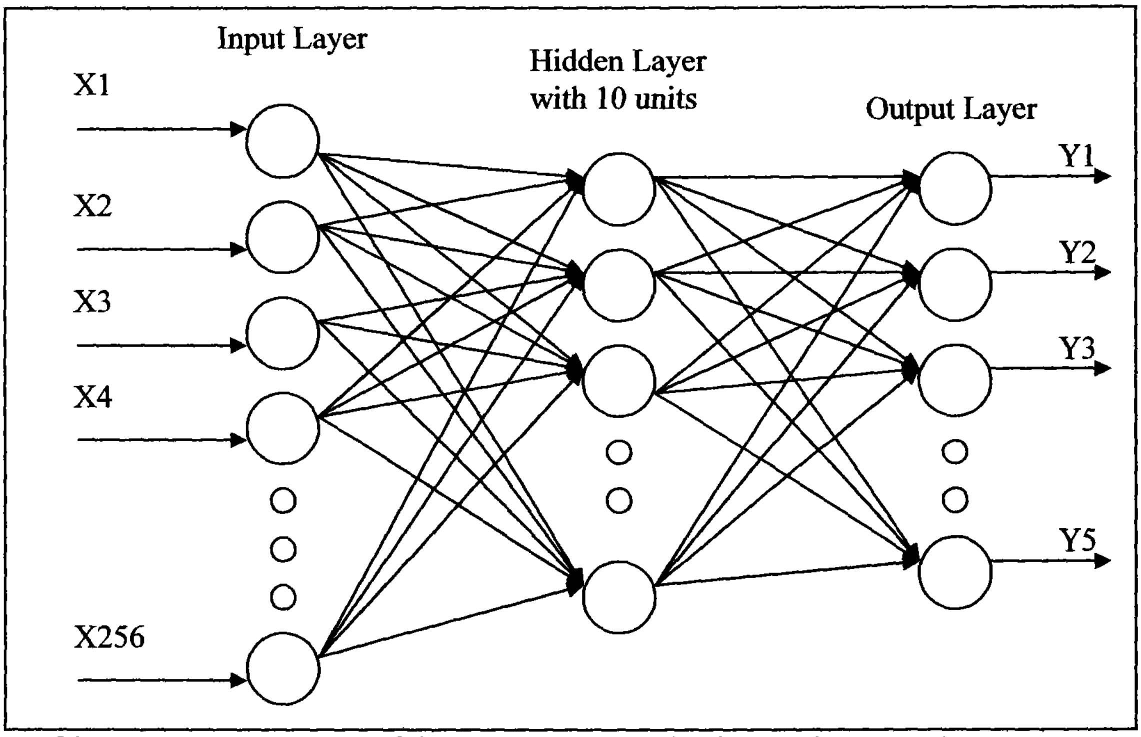


Figure 16: The structure of three layer network for fingerprint clustering system.

5.1.2 Implementation Neural Network

The implementation of developing this neural network classifier begins with developing a neural network class which enables user to build neural network modularly by calling this class functions and user also able to determine and configure the number of nodes with minimum modification on the code (change the number of nodes at the neural network declaration). This class assumed that all nodes are fully connected. This class consists of nine functions:

FeedForward ( ): This function that responsible to take input from direction reader and compute the output of the neural network.

BackPropagationForANode ( ): This function will compare the output of the neural network with the desired output; if the output isn't the desired one backpropagation will be performed. Every comparison is performed on one node only; if there are five output nodes this function has to be iterated five times.

CreateWeightToDatabase ( ): This function will initialize value for all the weights randomly between -1 to +1 and save all the initial weights to database.

Dsig ( ): This function responsible to calculate the dsig(x) for every node. The dsig function is $\frac{disg}{dx}(x) = sig'(x) = \frac{e^{-x}}{(1+e^{-x})^2}$. This is the derivative function of the sigmoid activation function.

Sig ( ): This function responsible to calculate the sig(x) for every node. The dsig function is $sig(X) = \frac{1}{1+e^{-x}}$. This is the sigmoid activation function formula.

Reverse_sig ( ): This function is called to compute the reverse of Sig ( ). The purpose of this function is to reduce the needs of keeping two sets of value; before and after Sig ( ).

SaveWeightToDatabase ( ): This function will save all value of all the weights to database.

LoadWeightToDatabase ( ): This function is called to load all the weight from database.

SetLearningRate ( ): This function allow user to set the learning rate of the Neural Network, default of this value is 1.

After successful developed this neural network class, several tests were made on this class to validate and verify its functions. Then by using all this available functions, several neural network classifiers' functions can be built modularly. Functions of the neural network classifier are:-

NN-classifier: This function will take the input from direction reader and then use the input to feed the neural network by using the FeedForward ( ) function. Then the output of the neural network will determine the cluster the fingerprint belongs. Table 3 shows the conversion of the output from the neural network to text according to production rules.

Training: This function will take the input from direction reader and then use the input to feed the neural network by using the FeedForward ( ) function, the output of each node will be compared, if the output is different from the desired output then BackPropagationForANode ( ) will be called to adjust the weights. This process will repeat until the desired output is achieved.

AutoTraining: This function is the same like Training but more automated. It will automatically feed the neural network will all available images, and train until the neural network can perform classification correctly.

Other functions such as **SaveWeight**, **LoadWeight**, **SetLearningRate** and **ResetDatabase** are implemented by calling function from the neural-network class.

The data structure that being applied to represent weights is a single dimension array. The process of generating input by the direction reader is shown by Figure 17. The fingerprint will first be segmented into 8 x 8 segments. Then each of these segments will consist of 16 x 16 pixels and will be read by the direction reader. There are total of six directions that this direction reader can differentiate and these directions are (90°-270°, 0°-180°, 30°-210°, 60°-240°, 120°-300° and 150°-330°) and every direction will be represented with the numbers from 1 to 6. A series of direction will be generated by the direction reader and will be the input for the neural network.

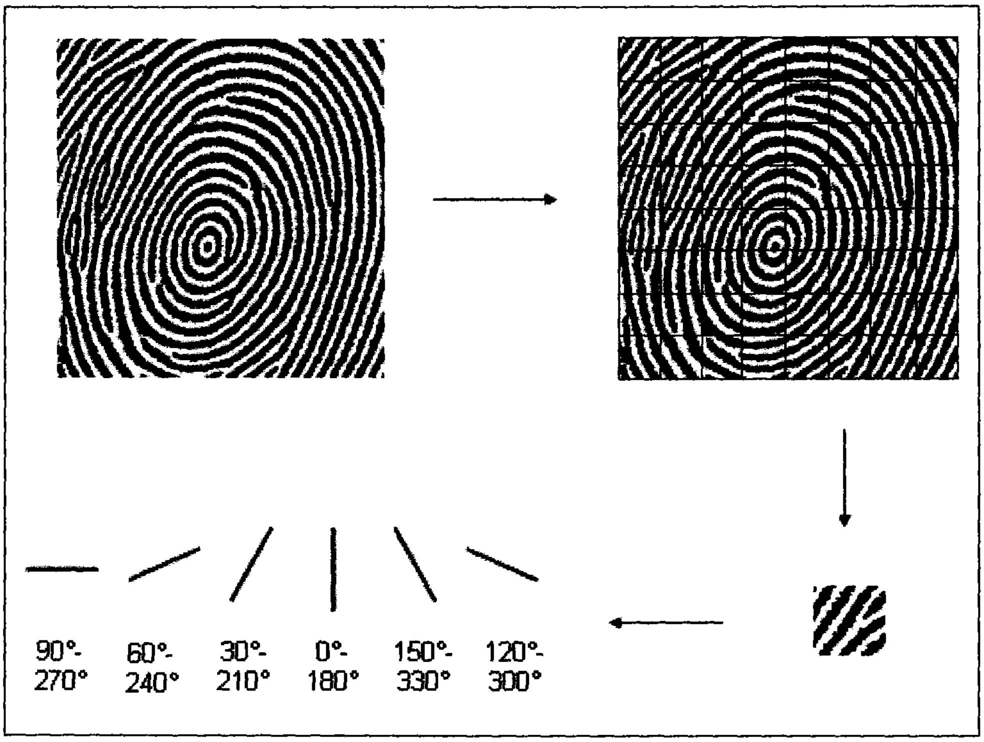


Figure 17: The process flow of Direction Reader.

This neural-network model was implemented and trained with 120 different fingerprints with +10 and −10 degree of rotation. The result of this neural-network model is located at **Chapter 6: Experimental Results** of this report.

5.1.3 Comparison of the Neural Network Methodology

This neural network structure was inspired by the neural network using a quick propagation training algorithm build by [4] and neural network by [6] which uses edge detection to create eigenvector from a given fingerprint image, then Kohonen Loeve transform (KLT) are then being applied to reduce the dimension of the input from the eigenvector, before feeding it to a multi layer perceptron.

The similarity of our neural network compare to the neural network of [4] and [6] is that all are Multi Layer Perceptron (MLP) with 3 layer network and fully connected.

The different between our neural network classifier with [4] is mainly the structure of the neural network and the training algorithm. The structure of the [4] has one layer of hidden nodes with 20 units, 192 input nodes and has 5 output nodes and the algorithm that they use to train is quick propagation training algorithm. Our neural network uses one layer of hidden nodes with 10 units, 256 input nodes correspond to the 256 features provided form the direction reader and 5 output nodes and the algorithm that we apply to train is backpropagation training algorithm.

Another difference is that the number of weights that neural network of [4] have to compute is 3940 (192 x 20 + 20 x 5) weights while our neural network only have to compute 2610 (256 x 10 + 10 x 5) weights.

If our neural network compare with neural network of [6] which uses edge detection to create eigenvector from a given fingerprint image, then Kohonen Loeve transform (KLT) are then being applied to reduce the dimension of the input from the eigenvector, before feeding it to a multi layer perceptron. The major difference is the neural network structure. Our neural network takes input from direction reader instead of eigenvector and never go through and transformation to reduce the dimension of input.

5.2.1 Fuzzy logic and Fuzzy Neural Fundamentals

Besides using neural network models we also proposed to use Fuzzy Neural Networks Models. The Fuzzy Neural Networks Models approach was being introduced about a decade ago. Fuzzy neural models are the combination of neural networks and fuzzy logic. According to [1], neural networks offer algorithms for numeric classification, optimization, and associative storage and recall while fuzzy logic offer tools to process inexact or approximate data at semantic level. Therefore fuzzy neural gives us more flexibility, greater representation power, higher processing ability and more robust than conventional neural networks.

By understanding and implementing multilayer neural network and fuzzy inference system; we know that it will enable us to develop a fuzzy neural classifier easier and also will have higher chance of success to build a fuzzy neural classifier. After succeeded develop the three classifier, all the three classifier were compared and studied on the efficiency and characteristic of each classifier.

From my study on fuzzy logic and according to [1] fuzzy logic is a technique to mimic human mind to have to ability of reasoning approximately instead of exact. This means that it tries to compute a reason or a decision with the ability to tolerate of imprecision. For examples understand sloppy handwriting, recognize and classify images. In fuzzy logic there is a fuzzy inference system which able to solve a nonlinear mapping of the input data vector into a scalar output by using fuzzy rules. Figure 20 and 21 shows the structure of a fuzzy inference system. The fuzzy logic inference system has four components which are the fuzzifier, inference engine, rule base, and defuzzier. The fuzzy inference system needs rules for a specified domain which usually need the knowledge of expert in that field. Besides that rules are possible to be extracted from numeric data. Therefore, to build the rules for fuzzy logic we performed a study on a number of fingerprints images and then generated a graph shown in figure 18 to show the frequency of direction distribution through different type of fingerprint images. Figure 19 shows the sample of fuzzy logic production rules set.

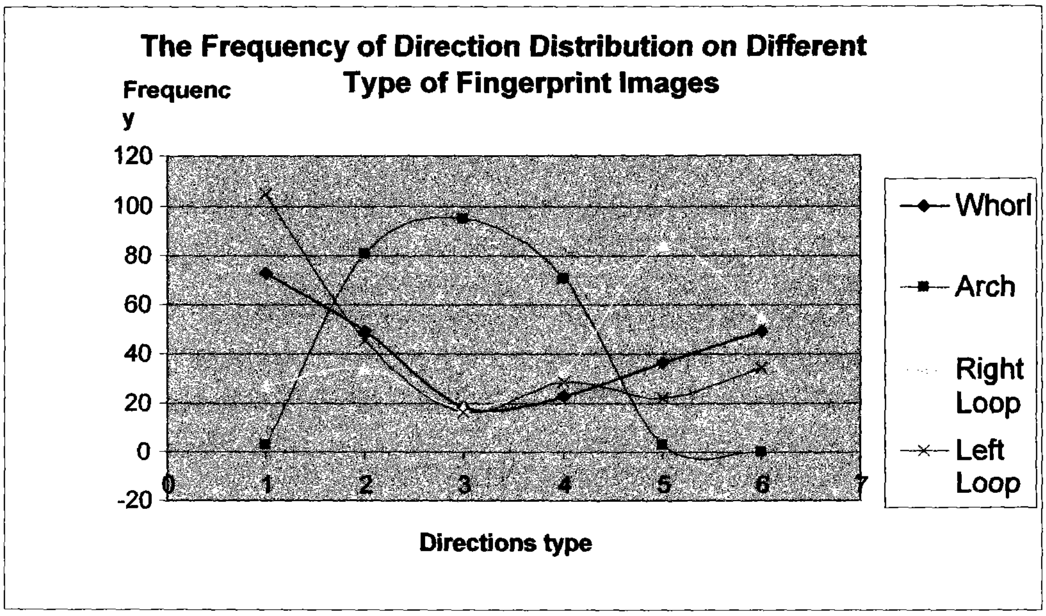


Figure 18: Frequency of Directions Distribution on Different Type of Fingerprint Images

From figure 18 we could determine the fuzzy logic's production rules and fuzzy input sets values. For examples, we can set the production rule for the Arch type as "if directions type 1 is low and directions type 2 is high and directions type 3 is high and directions type 4 is high and directions type 5 is low and directions type 6 is low then it is a Arch type fingerprint image." The value of the input fuzzy set can also be determine from these graph, for instance we have set that value below frequency 30 is very few, value above 25 is few, value above 45 is considered as average and lower then 55 is few, value above 65 is considered much and value below 75 is average, and if value above 80 is considered very much and below 90 is much. These values are shown in the figure 23 and figure 24 shows fuzzy logic output sets.

These rules enables the fuzzy system to maps an input vector to an output vector. The function of fuzzifier is to maps input number into corresponding fuzzy membership in order to activate rules that are in the form of linguistic variables. It takes the input value and determines the degree of belonging to the fuzzy sets along membership functions. Then the inference engine which responsible to map the fuzzy input to fuzzy output by determining the degree to which the antecedent is satisfied for each rules and if then the rules have more then one clause, the fuzzy operators will be applied to obtain one number that represents the result of the antecedent for that rules. There are also possibilities that more then one rules are being fired at the same time. Therefore the outputs for all these rules are then aggregated by combining the fuzzy sets that represent the output into a single fuzzy set. Lastly the defuzzier maps the output fuzzy sets in to a crisp number. There are several methods of defuzzification such as centroid, maximum, mean of maxima, height, and modified height defuzzifier. [1]

R_1: if d1 is average and d2 is very few and d3 is average and d4 is very few and d5 is average and d6 is average, then output is whorl.
R_2: if d1 is few and d2 is very few and d3 is very much and d4 is average and d5 is average and d6 is much, then output is left loop.
: :
: :

R_{15625}: if d1 is average and d2 is few and d3 is average and d4 is average and d5 is average and d6 is much, then output is others.

Figure 19: Sample of fuzzy production rules set used in fuzzy logic classifier.

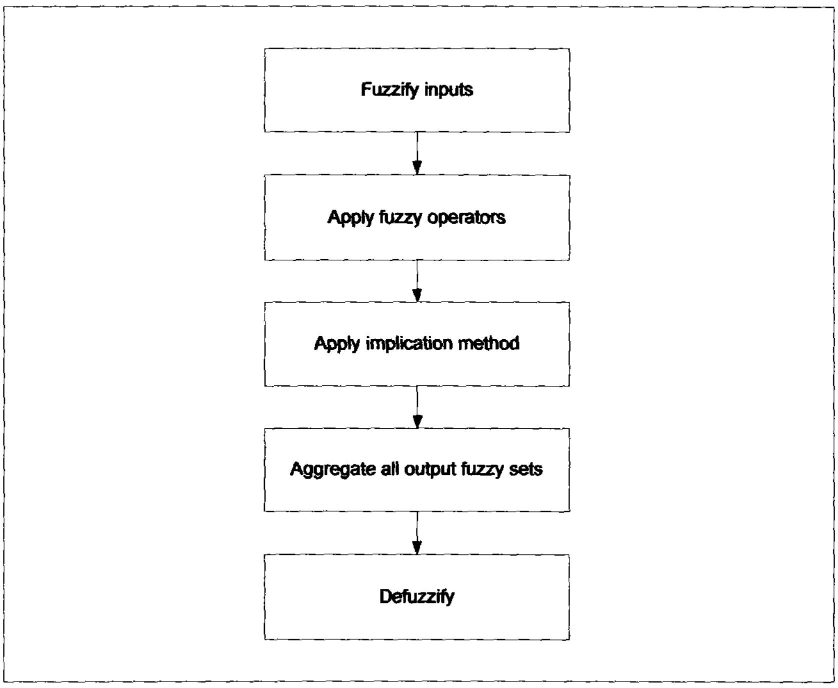


Figure 20: Fuzzy inference process [1]

Previously, we proposed two ideas on combining neural network model and fuzzy logic to perform classification. The first idea of combination is to integrate a fuzzy inference system in the middle of a multilayer neural network (inspired by [1]) and the second idea is to use a fuzzy inference system to determine which neural networks to perform classification (inspired by [4]). However, the first idea was not implemented because due to its complexity and the second idea was successfully being implemented, tested and compared with neural network and fuzzy logic classifier. The results of these comparisons are located at **Chapter 6: Experimental Results** of this report.

Figure 22 shows the structure of our first idea; three layer fuzzy neural network model which is a model for Fuzzy Neural Network Model with Fixed Membership Functions. From the diagram H, the **x** is going to be the input and the **o** is the output. There are 3 levels in the diagram L1, L2 and L3. The L1 represent the input layer, the L2 perform the function of the inference engine and L3 is the output layer. Therefore to applied this structure of three layer fuzzy neural network model, the input will be value provided by the direction reader, then the value will be pass to L2 the inference engine to be process with numerical data generated rules. The confident value that obtains from the L2 layer will be transfer to L3 to perform a typical neural network feed forward process and if the output value different from the desired output then backpropagation learning algorithm will be use. According to [1], it is also possible to use a feed forward neural network to work as an inference engine.

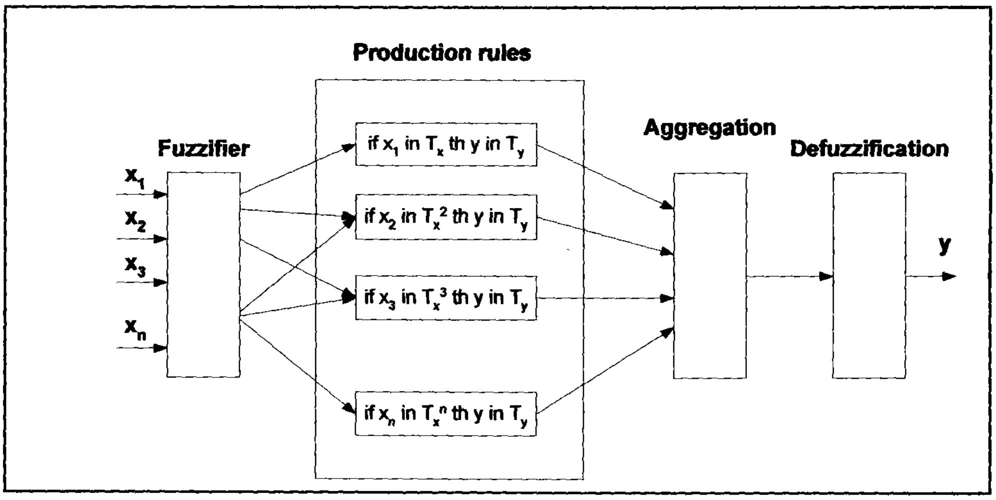


Figure 21: Schematic diagram of a fuzzy inference system [1]

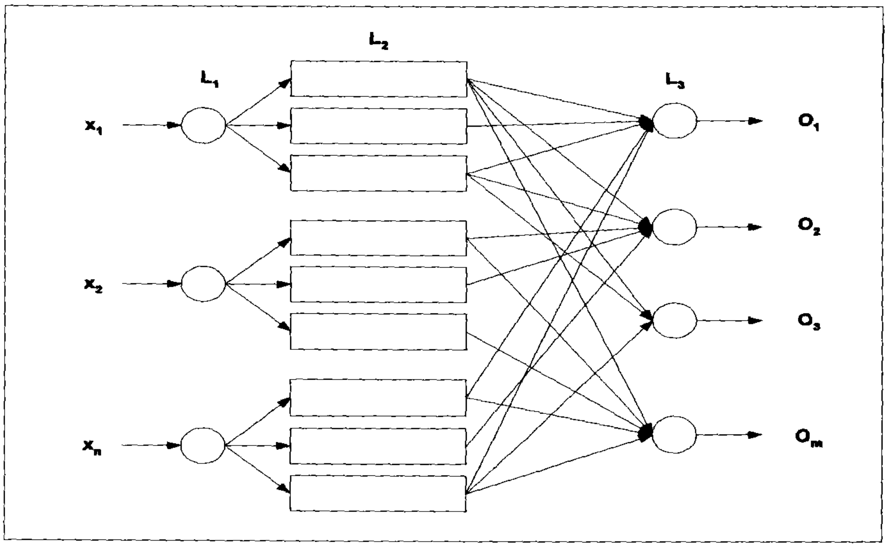


Figure 22: The first idea structure of three layer fuzzy neural network models extracted from [1]

Figure 25 shows our second idea of combining neural network with fuzzy logic. The input of this system will be from the direction reader a 256 array. Then the input will first be processed by the fuzzy inference system and it will make decision on which neural network classifier will be used. There will be one out of the six multilayer neural network models to perform classification (chosen earlier by the fuzzy inference system) and each of this neural network models will responsible on differentiating only two types of fingerprints and one unidentified fingerprint type which will be cluster to the *Others* class.

We need to have six multilayer neural networks because our system has five classes of fingerprint needed to be classified. The six neural network classifier are the classifier that only classify between Whorl and Right Loop (WR), Right and Arch (RA), Left Loop and Right Loop (LR), Left Loop and Arch (LA), Whorl and Left Loop (WL), and Whorl and Arch (WA). If the chosen neural network failed to classify the fingerprint image then it will classify that image to the *Others* class. Currently, we have successfully implemented this model; therefore we are able to give all the exact details on this model. Implementation and experiments on this model was written at the coming sections.

The structure of neural network and fuzzy logic in this fuzzy neural classifier is slightly different from the neural network classifier and the fuzzy logic classifier that we built to compare with this fuzzy neural classifier. The structure for the six neural networks are the same, instead of using the same structure like the neural network classifier which uses 256 input nodes correspond to the 256 features provided form the direction reader, one hidden layer with 10 nodes and 5 output nodes, this fuzzy neural uses 256 input nodes, one hidden layer with 5 nodes and 2 output nodes and backpropagation training algorithm is being applied.

The differences between fuzzy logic classifier that we built and this fuzzy neural is that the number of membership functions in the fuzzy input sets and fuzzy output sets, instead of using 5 membership functions in the fuzzy input sets it uses only 2 membership functions in the fuzzy neural input sets and instead of using 6 membership functions in the fuzzy output sets it uses 7 membership functions in the fuzzy neural output sets. See figure 26 for fuzzy neural input sets and see figure 27 for fuzzy neural output sets and figure 28 shows the samples of fuzzy neural's production rules.

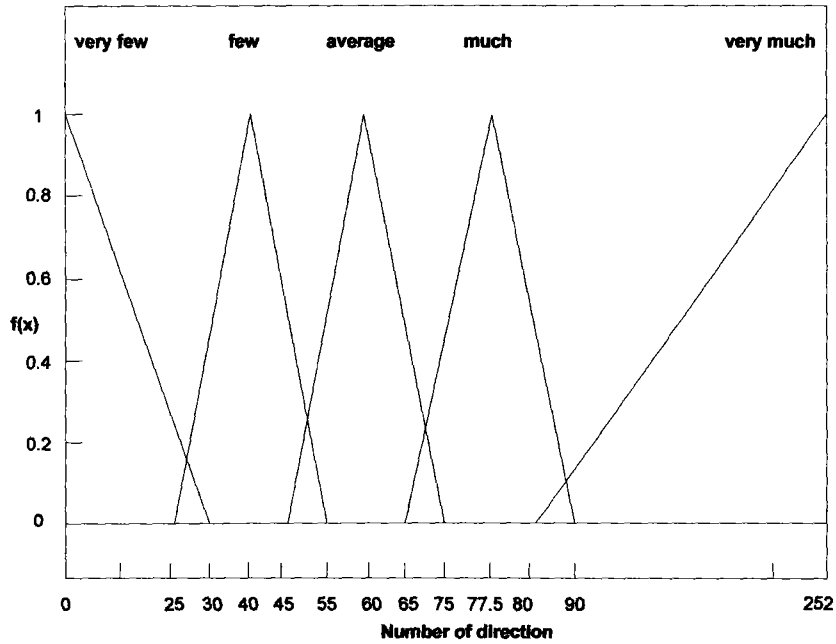


Figure 23: Fuzzy Logic's Input fuzzy sets with 5 membership functions

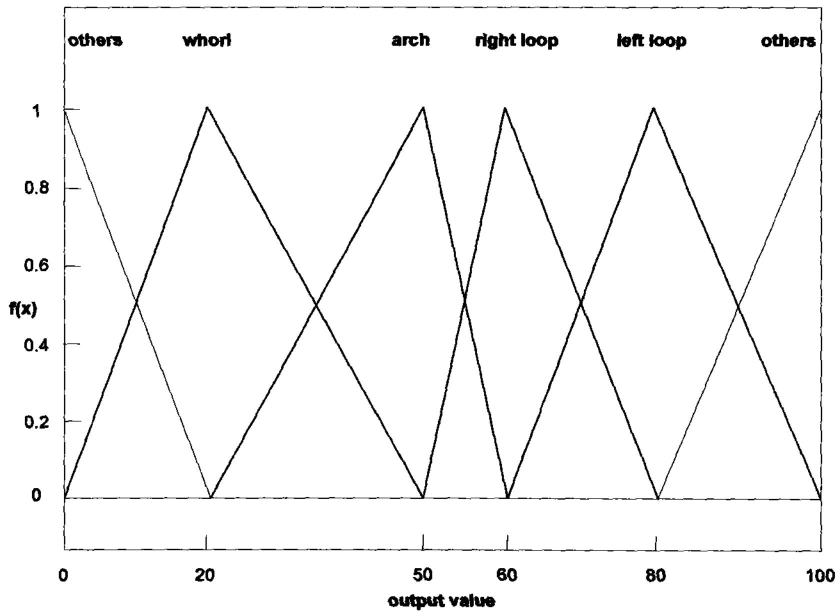


Figure 24: Fuzzy Logic's Output fuzzy sets with 6 membership functions

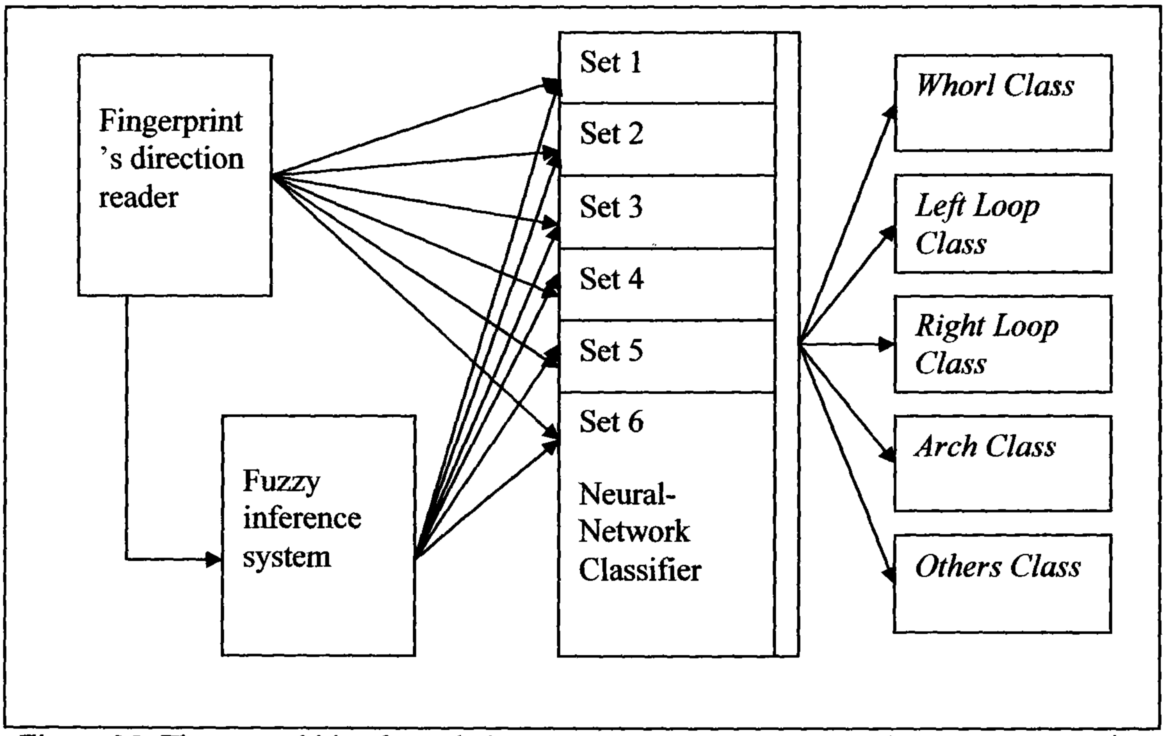


Figure 25: The second idea fuzzy inference system to determine which neural networks to perform classification

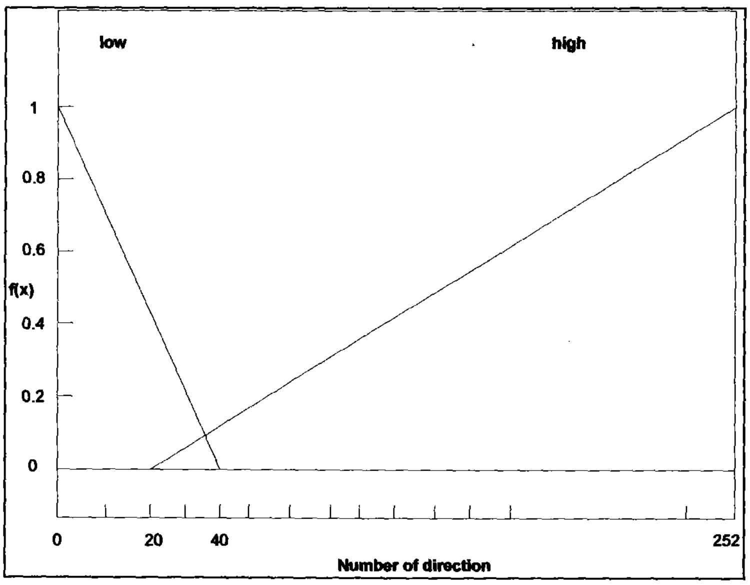


Figure 26: Fuzzy Neural input fuzzy sets with 2 membership functions.

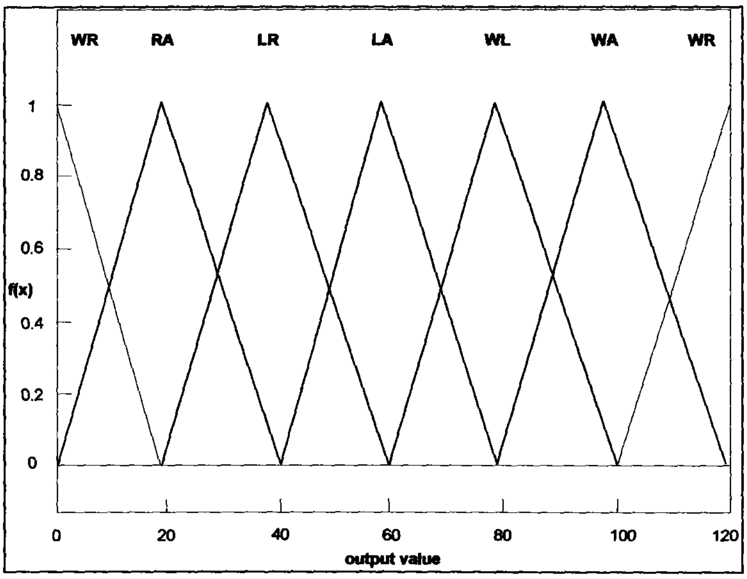


Figure 27: Fuzzy Neural output fuzzy sets with 6 membership functions.

R_1:	if d1 low and d2 high, then output is whorl.
R_2:	if d1 low and d2 low, then output is arch.
:	:
:	:
R_{25}:	if d1 high and d2 low, then output is left loop.

Figure 28: Sample of fuzzy production rules set used in fuzzy neural classifier.

5.2.2 Implementation

The implementation of this fuzzy neural model begins by developing a fuzzy neural class by using functions from neural network class (written at section 5.1.3 Implementation Neural Network) and fuzzy logic class. This fuzzy logic class consists of functions such as:-

> **GraphReader ( )**: This function performing fuzzification, takes input and determine the degree to which they belong to each of the appropriate fuzzy sets via membership functions. [1]
>
> **Defuzzification ( )**: This function responsible to perform two task, first, once the input have been fuzzified by GraphReader ( ) it will check which rules to fire and their degree of firing all the fired rules will be aggregated. Then the second task is

defuzzification by mapping the output fuzzy sets in to a crisp number by using centroid as method of defuzzification.

ProductionRulesCreator ( ): This function is being developed to generate all possible rules. Although this function are being used once in the entire process of developing fuzzy logic, it is essential to have this function because this fuzzy logic have six input fuzzy sets and each input fuzzy set only have 5 membership functions therefore there are 5^6 (15625) rules. All the rules are store in database.

FuzzyLearning ( ): This is a special function; it is responsible to adjust the condition of the production rules according the training set of image.

Statistic ( ): This function is responsible to count the possibility rules of firing for a set of fingerprint images. After the statistic of the rules are gather, then FuzzyLearning ( ) is being applied to set all the condition of all the fired rules according to its statistic.

Functions from the neural network class and the fuzzy logic class contribute to build a fuzzy neural class which consist all the function from neural network class and fuzzy logic class. Several functions are modified to adapt to the needs of fuzzy neural classifier. For examples,

FNProductionRulesCreator ( ): This function is the same like ProductionRulesCreator ( ), instead of generating 5^6 (15625) rules, this function was being modified to generate 2^6 (64) rules. This is because the fuzzy neural have six input fuzzy sets and each input fuzzy set only have 2 membership functions.

Other modified functions such as **SaveWeight** and **LoadWeight** are implemented by modifying functions from the neural-network class to adapt to the needs of fuzzy neural classifier.

The data structure that being applied to represent the graphs for the fuzzy input and output sets in the code are several 3 dimensional arrays where the first subscript represents the number of fuzzy input set, the second subscript represents the number of membership functions and the third subscript represents all the details of the membership function in the fuzzy input sets.

This fuzzy neural classifier structure is shown in Figure 25. Inputs generated from the directional reader will then be send to fuzzy inference system to determine which set of neural network should be use to perform classification on the inputs. During at the fuzzy inference system, GraphReader ( ) were called to perform fuzzification on the inputs. Then Defuzzification ( ) were invoked and it will first check all the rules in the database and fire related rules, each fired rules will be assigned with a confident value (the degree of the rules being fired). Aggregation will be performed to combine all the output of the fuzzy set into one composite fuzzy set.

Finally, defuzzification process will be applied on this composite fuzzy set to obtain the output decision. Then the output decision will determine which neural network to be used and the outputs from the selected neural network determine the cluster that the fingerprint belongs.

5.2.3 Comparison of the Fuzzy-Neural Methodology

This fuzzy-neural classifier is being inspired by [4], in this section we will compare our fuzzy-neural classifier with [4]'s classifier. The [4]'s classifier is not a fuzzy-neural classifier but a classifier that uses two stage of classifier to perform a fingerprint classification. The two methodologies that are being used are K-nearest neighbor classifier and 10 sets of neural networks. It begins clustering by using K-nearest neighbor classifier to find the most probable classes for a given input, the top two categories that the K-nearest neighbor (tendency to assign the input to the class; based on the classes which have the highest and the second highest possibility) is being use to choose a specific neural network that has been trained to classify the input. The neural network classifier is the second stage classification of this classifier.

The similarity of our fuzzy-neural classifier with [4]'s classifier is that both of classifier uses two stage classification. Instead of using K-nearest neighbor classifier, fuzzy-neural classifier uses fuzzy logic classifier as the first stage of classification and then the output of this fuzzy logic will choose a specific neural network that have been trained to classify as the second stage of the classification.

The differences between our fuzzy-neural classifier with [4]'s classifier is that our fuzzy-neural uses 256 features as input while [4]'s classifier uses 192 features as input. The inputs to these two systems are also different, we uses fingerprint's ridge and valley orientation while [4]'s classifier uses feature that generated by Gabor filter. Instead of using 10 sets of neural network classifiers, our fuzzy-neural classifier uses only 6 sets because every classifier responsible to classify two type of fingerprints and for all unidentified type of fingerprints to a specific class. We consider the unidentified type as a cluster. Figure 29 is the structure of [4]'s classifier.

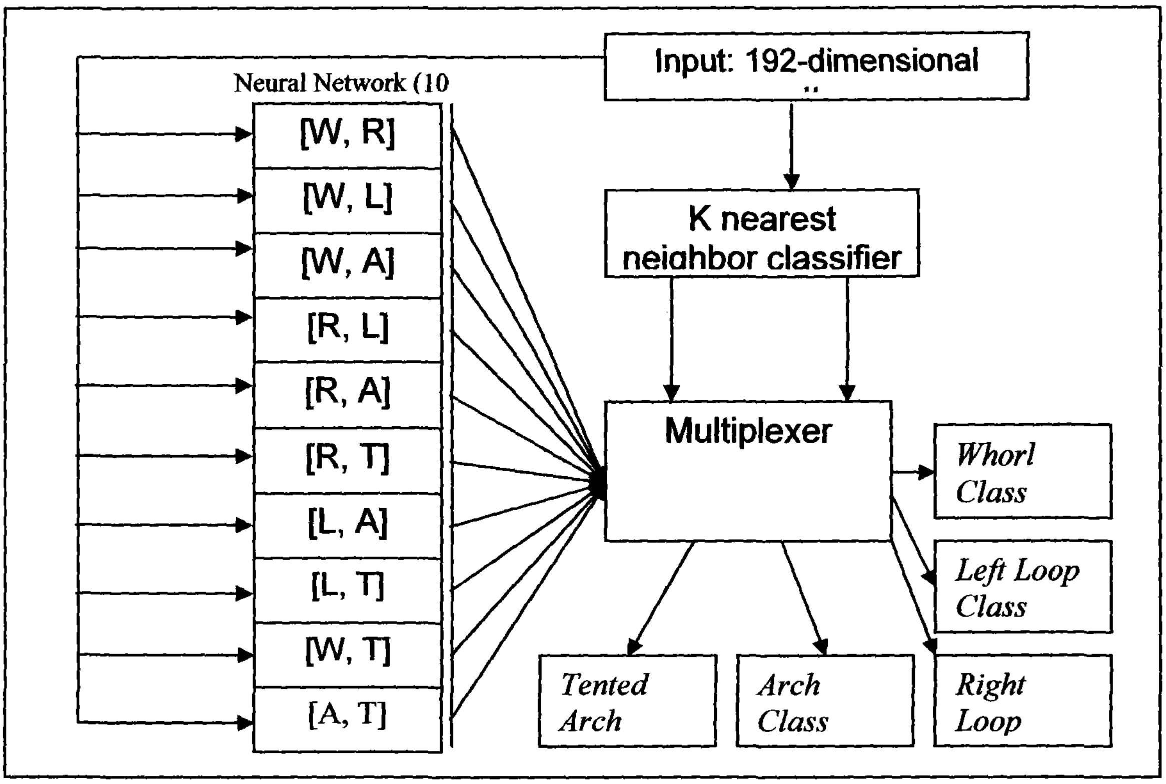


Figure 29: Structure of [4]'s classifier

Chapter 6: Experimental Results

This section we will compare the performance of the three methodologies in fingerprints classification. These three methodologies are all being trained with a same set of 120 images with +10 and -10 degree of rotations. There are total of 5 comparisons being made, such as Accuracy of Classification for Trained Fingerprint Images, Accuracy of Classification for Untrained Fingerprint Images, Efficiency of Neural Network, Fuzzy Logic and Fuzzy Neural. The purpose of having accuracy test is to check how accurate the three classifiers perform classification on trained images and untrained images and the purpose of having efficiency test is to check how efficient (speed) the three classifiers perform classification on different number of fingerprints, being from 10 images to 150 images.

6.1 Accuracy Test for Neural Network (NN), Fuzzy Logic (FL) and Fuzzy Neural (FN) classifier on Classifying Trained and Untrained Fingerprint Images.

The accuracy test for trained fingerprint images uses a set of 120 images with +10 and -10 degree of rotations. The 120 images consist of 4 types of fingerprint where each type has 30 images (30 whorls, 30 arches, 30 left loops, and 30 right loops). After the process of training these classifiers, each image of the same sets of the training images is again feed into these classifiers to perform validation. If the submitted fingerprint is classified correctly then we will consider it as TRUE while if the classifier classified it wrongly then we will consider it as FALSE.

Table 4 shows the accuracy of classification for trained fingerprint images and figure 30 shows the accuracy in the form of bar graph, from table 4 and figure 30 we can make conclusion that neural network classifier have a very outstanding results. It is able to achieve 100% accuracy, while fuzzy neural have the worst performance which manage to score 94.17% accuracy. The error of fuzzy neural are concentrated on the classifying Arch type fingerprints, while fuzzy logic have errors classifying whorl type and right loop cluster but not as serious as error in fuzzy neural, therefore fuzzy logic achieved 95.83% accuracy.

The accuracy test for untrained fingerprint images uses a set of 260 images with +10 and -10 degree of rotations, these 260 images never been exposed to the three classifiers. These untrained images also like the 120 images set, it consist of 4 types of fingerprint where each type has 65 images (65 whorls, 65 arches, 65 left loops, and 65 right loops). Then all these fingerprints will be submitted and classified by each classifier, if the classifier correctly then we will consider it as TRUE while if the classifier classified it wrongly then we will consider it as FALSE.

Table 5 shows the result of accuracy test of classification on untrained fingerprint images and figure 31 show the accuracy in the form of bar graph. As the results, neural network classifier has the highest accuracy 96.92% but some error when classifying Whorl and Left Loop type of fingerprint. Fuzzy logic has the lowest accuracy with 76.54% which dropped 19.29% because it misclassified a lot of Whorl type fingerprints while in this experiment fuzzy neural accuracy decreases 6.66% lesser compare to fuzzy logic classifier and decrease 12.63% compare to classifying trained images accuracy test. Overall fuzzy neural classified untrained images better then fuzzy logic classifier and neural network is still the best of the all three classifier.

Table 4: The accuracy of classification for trained fingerprint images

NN Finger Types	Whorl	Arch	Right Loop	Left Loop	SUM
Total Images	30	30	30	30	120
TRUE	30	30	30	30	120
FALSE	0	0	0	0	0
Accuracy (%)	100	100	100	100	100

FL Finger Types	Whorl	Arch	Right Loop	Left Loop	SUM
Total Images	30	30	30	30	120
TRUE	27	30	28	30	115
FALSE	3	0	2	0	5
Accuracy (%)	90	100	93.33	100	95.83

FN Finger Types	Whorl	Arch	Right Loop	Left Loop	SUM
Total Images	30	30	30	30	120
TRUE	30	23	30	30	113
FALSE	0	7	0	0	7
Accuracy (%)	100	76.67	100	100	94.17

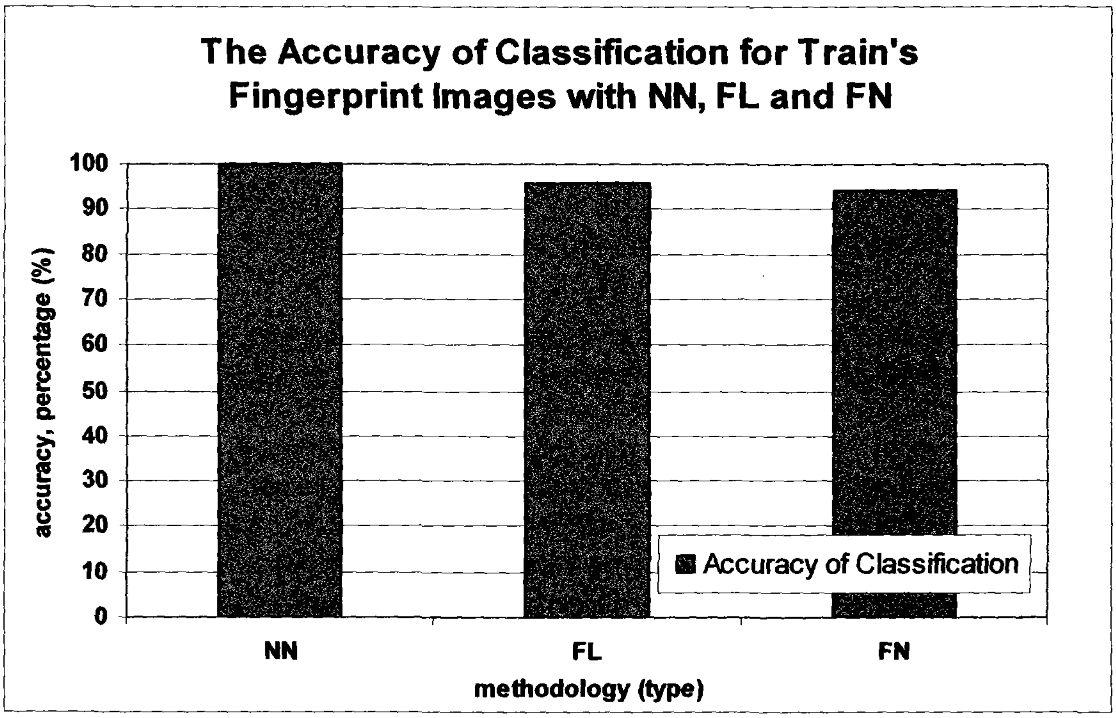


Figure 30: The accuracy of classification for trained fingerprint images

Table 5: The accuracy of classification for untrained fingerprint images

NN Finger Types	Whorl	Arch	Right Loop	Left Loop	SUM
Total Images	65	65	65	65	260
TRUE	60	65	65	62	252
FALSE	5	0	0	3	8
Accuracy (%)	92.31	100	100	95.38	96.92

FL Finger Types	Whorl	Arch	Right Loop	Left Loop	SUM
Total Images	65	65	65	65	260
TRUE	38	56	48	57	199
FALSE	27	9	17	8	61
Accuracy (%)	58.46	86.15	73.85	87.69	76.54

FN Finger Types	Whorl	Arch	Right Loop	Left Loop	SUM
Total Images	65	65	65	65	260
TRUE	39	49	64	60	212
FALSE	26	16	1	5	48
Accuracy (%)	60	75.38	98.46	92.31	81.54

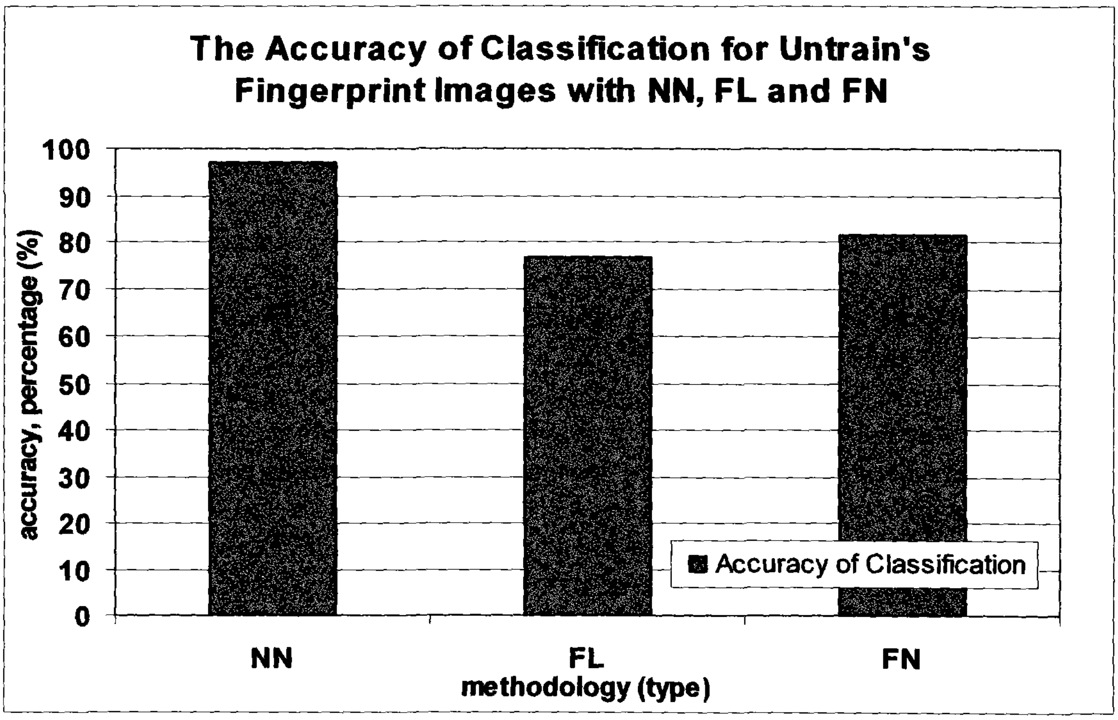


Figure 31: The accuracy of classification for untrained fingerprint images

6.2 Efficiency of Neural Network, Fuzzy Logic and Fuzzy Neural

The test of efficiency is being performed after we embedded the classifier into the fingerprint recognition system. This test uses 150 fingerprint images; these images are being registered into the system by 10 by 10 basics and for every 10 images being register, the matching process will be performed to get the matching time (duration of identifying a fingerprint images with different amount of fingerprints data in the database). This is also the same test that we had performed on the previous fingerprint recognition system without classification. The previous system efficiency is also shown in each of the graph Figure A, B, C.

This experiment is being performed on the three methodologies, in order to reduce the variance of time taking process we take the average matching time; for every recognition with different methodologies are being perform several times and the average time of the matching time are being recorded.

The results of this experiment on the three classifiers are almost the same, the identification and reject imposter time for fingerprint recognition are successfully reduce more then 5 times, the old fingerprint recognition system from 80 seconds to approximately 12 seconds and from 140 seconds to approximately 25 seconds to identified a fingerprint that does not exist in the database which registered with 150 fingerprint images.

The identification and reject imposter time for fingerprint recognition with Fuzzy Logic classifier is showed in table 6 and figure 32; from the graph we can see that fuzzy logic classifier successfully reduce the matching time of the old fingerprint recognition system from 80 seconds to approximately 12 seconds and from 140 seconds to approximately 24 seconds to identified a fingerprint that does not exist in the database which registered with 150 fingerprint images.

For fingerprint recognition with Neural Network classifier is shown in table 7 and figure 33, it also shows that it is able to reduce the identification time from 80 seconds to approximately 12 seconds and able to realize that the submitted fingerprint does not exist in the database within approximately 27 seconds.

Fingerprint recognition with Fuzzy Neural classifier is shown in table 8 and figure 34 also provide almost same results; it is able to recognize a fingerprint approximately 13 seconds in a database with 150 images, and reject an unregistered fingerprint within approximately 26 seconds.

Although fuzzy logic has the shortest time of rejecting an imposter time, but conclusion cannot be made; that fuzzy logic perform classification faster because there are times that neural network and fuzzy neural classifier are able to perform classification and matching faster then fuzzy logic. This is because that the matching speed also dependent to the database query speeds.

Table 6: The identification and reject imposter time for Fuzzy Logic

No. of Fingerprints	10	20	30	40	50	60	70
Identification Time	2"20	2"22	3"22	3"69	4"47	4"82	4"96
Reject Imposter Time	1"75	4"63	5"15	7"92	8"76	10"15	9"47

No. of Fingerprints	80	90	100	110	120	130	140
Identification Time	5"38	6"35	8"81	9"55	9"67	10"21	10"85
Reject Imposter Time	13"70	15"64	18"25	15"85	19"52	21"58	21"96

No. of Fingerprints	150
Identification Time	12"25
Reject Imposter Time	24"20

**** Time read as minute'second"milisecond**

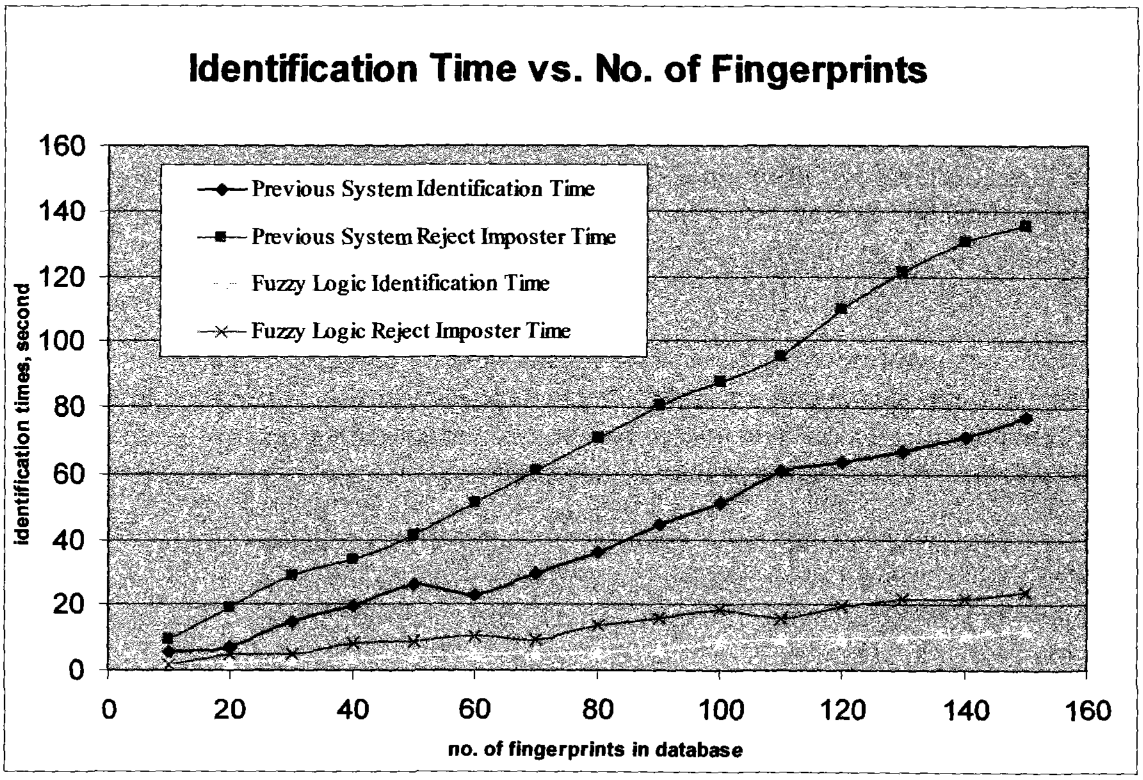


Figure 32: The identification and reject imposter time for Fuzzy Logic

Table 8: The identification and reject imposter time for Fuzzy Neural

No. of Fingerprints	10	20	30	40	50	60	70
Identification Time	2"25	2"38	3"68	3"89	5"88	5"70	4"90
Reject Imposter Time	2"00	5"25	4"70	8"43	10"07	10"63	9"48

No. of Fingerprints	80	90	100	110	120	130	140
Identification Time	5"53	6"57	9"19	9"82	9"29	9"92	11"00
Reject Imposter Time	13"52	15"48	19"06	16"91	19"21	22"53	21"88

No. of Fingerprints	150
Identification Time	12"69
Reject Imposter Time	25"33

**** Time read as minute'second"milisecond**

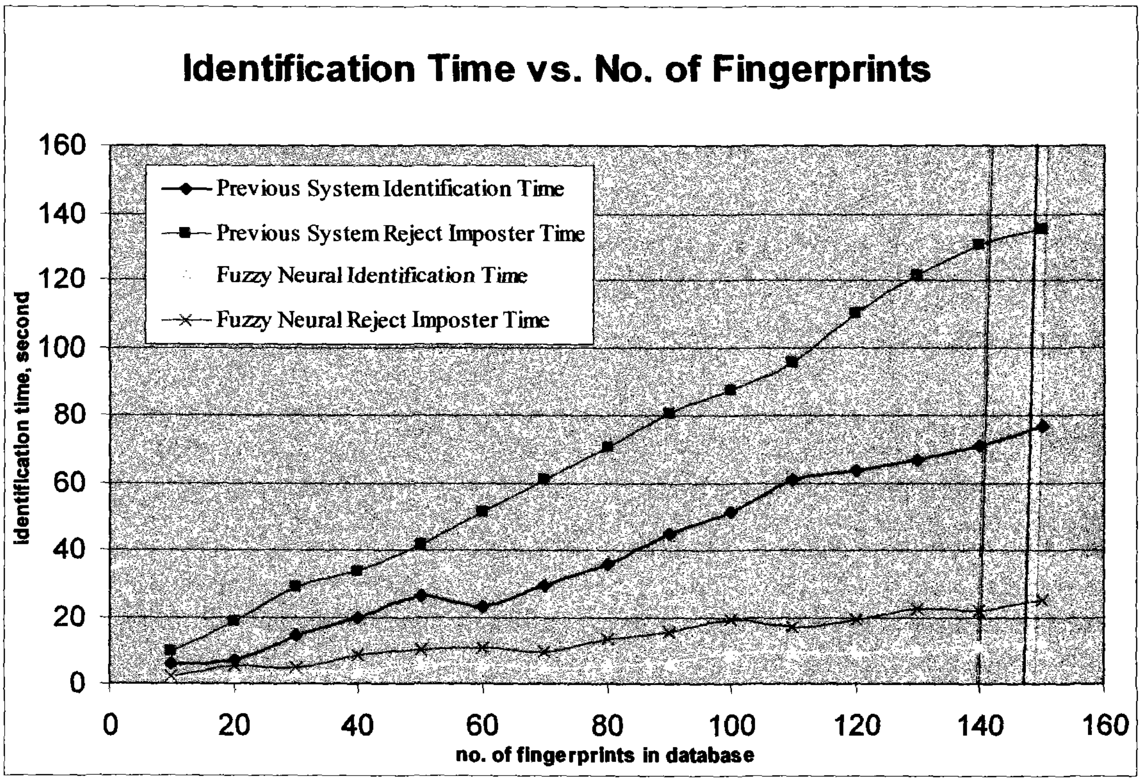


Figure 34: The identification and reject imposter time for Fuzzy Neural

Table 7: The identification and reject imposter time for Neural Network

No. of Fingerprints	10	20	30	40	50	60	70
Identification Time	1"86	2"43	3"10	3"28	4"75	4"86	4"77
Reject Imposter Time	2"59	4"69	4"62	7"47	8"60	10"53	9"50

No. of Fingerprints	80	90	100	110	120	130	140
Identification Time	5"41	6"67	9"20	9"37	8"96	9"78	10"25
Reject Imposter Time	13"61	14"80	18"25	16"82	20"59	22"15	21"24

No. of Fingerprints	150
Identification Time	12"03
Reject Imposter Time	26"43

**** Time read as minute'second"milisecond**

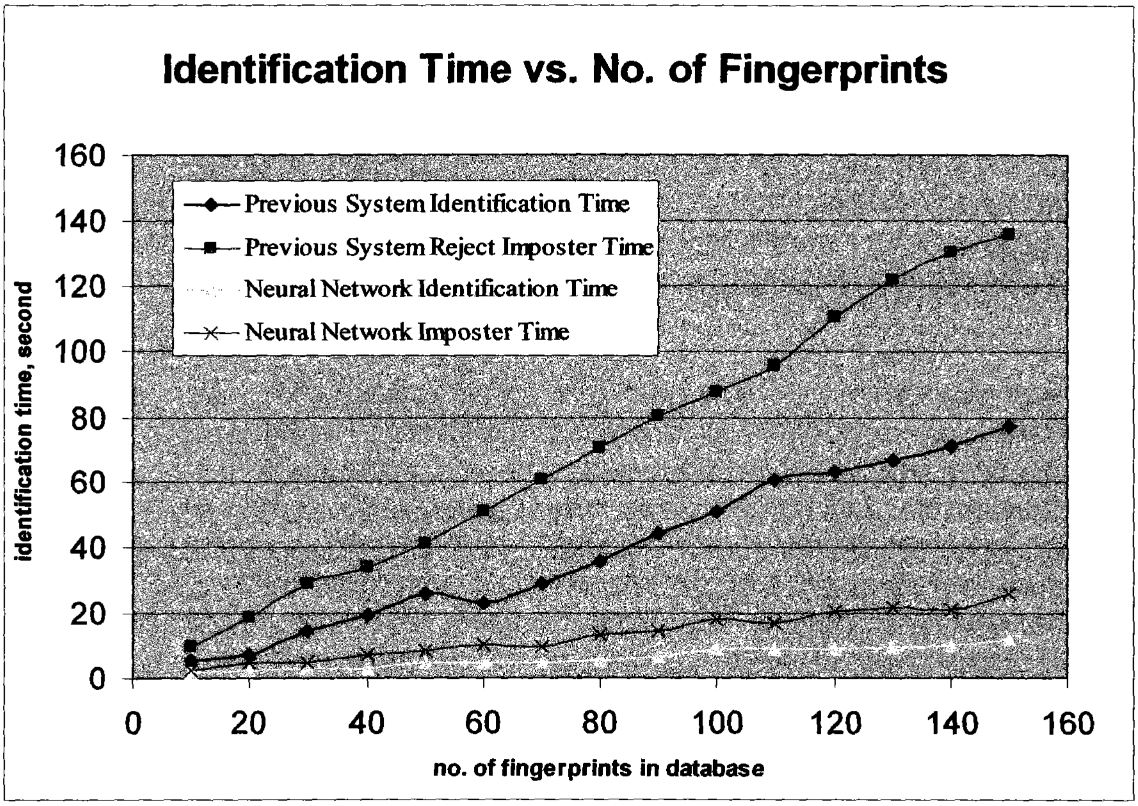


Figure 33: The identification and reject imposter time for Neural Network

Chapter 7: Discussions

The results from chapter 6 can be explained, for the trained images accuracy test the neural network classifier have an outstanding results because it have a good training algorithm and it is forced to train until all the images in the training sets are correctly classified. The fuzzy neural and fuzzy logic performance was not good as what we predicted earlier, this is because the imperfection of production rules of the fuzzy neural and fuzzy logic, moreover the production rules of these two classifiers are created manually base on statistic that we collected from the 120 images. We had tried to adjust the fuzzy output set of fuzzy neural and fuzzy logic to achieve better accuracy and this is the best result that we able to achieve.

For the untrained images accuracy test, the error that cause by neural network classifier is still at an acceptable state; the neural network classifier is still able to classify all arch and right loop type fingerprints without any error while the fuzzy logic classifier get a lot of errors at classifying whorl type fingerprints. We believe that neural network classifier problem can be solved by using even more fingerprint images but the error rate for fuzzy logic and fuzzy neural become worst compare with the results of trained images accuracy test. The causes that contribute to this error could be insufficient images during training these classifier; those training images do not able generally represent all other fingerprints and the second cause is the fault of incompleteness and inaccurate representative for the fingerprints images with the current production rules.

From the efficiency test of neural network, fuzzy logic and fuzzy neural classifier we can conclude that overall of these three classifiers have almost the same effects on speeding up the matching speed of the fingerprint recognition systems. The new fingerprint recognition systems with these classifiers perform 5 times more better then the previous fingerprint recognition system.

Although overall result shows that neural network classifier is the best among the three methodologies, instead of concluding that fuzzy logic and fuzzy neural classifier is weaker then neural network classifier, our conclusion on this experiments and results it is just that fuzzy logic and fuzzy neural classifier do not fit to this problem domain (classifying fingerprints) and there are several reasons that cause fuzzy logic and fuzzy neural cannot perform well in this problem domain.

The limitation of direction reader is one of the reasons that cause fuzzy logic and fuzzy neural cannot perform well. The directional reader did not generate enough directions type to serve the needs of fuzzy logic. This is because there are only six directions that this direction reader can differentiate and therefore there are only total of six inputs can be provided to fuzzy logic to perform classification; the frequency of the six directions distribution on the fingerprint image is being taken as input. Using frequency of the six directions distribution the fingerprint image will cause a lot of information losses especially on the vector (location) of those directions.

Statements from [1] stated that "The nature of fuzzy logic; fuzzy logic is a technique to mimic human mind to have to ability of reasoning approximately instead of exact. This means that it tries to compute a reason or a decision with the ability to tolerate of imprecision." Therefore by using only six inputs (six frequency directions distribution on fingerprint image) it is difficult to achieve high accuracy of classification with fuzzy logic.

This is very different from neural network classifier which takes the location of the direction in to consideration. Every direction on a different location of a fingerprint image is corresponded to a unique node of the neural network classifier. Neural network uses all the available information to perform classification therefore it have a better results then both other classifier. Fuzzy neural was expected to have the best result of the three classifiers but it did not achieve to be the best classifier and this can be explained through the reason that deteriorates fuzzy logic. Since the fuzzy neural that we built structured the fuzzy logic in front of the neural network, the inaccuracy of fuzzy logic also affect the choice of choosing the correct sets of neural network classifier.

Another reason that deteriorates fuzzy logic and fuzzy neural classifier performance is the incompleteness and inaccurate representative of the current production rules. This problem domain does not have expert to determine those production rules therefore it is difficult to build complete and accurate representative production rules. Beside the production rules, it is also difficult to determine the input and output of the fuzzy sets value. Therefore, we use a special function to gather all information of a certain number of fingerprints and try to generalize the information so that we could perform classification on all fingerprint with a set of generalize production rules and fuzzy sets. See chapter 5.2.1 Fuzzy logic and Fuzzy Neural Fundamentals for more details.

From the perspective of the ease of developing these classifiers, we strongly feel that neural network is the easiest to be built and followed by fuzzy logic and fuzzy neural. The ease of developing neural network classifier is because there are plenty of existing and excellent training algorithms and the difficulty of developing neural network classifier is that there are many weights to be managed and computed, it required more mathematical calculation compare to the other two classifier and neural network performs classification in a black box manner which cause it very difficult to debug during developing period.

The beginning of developing a neural network classifier, we have an assumption that number of nodes in the hidden layer must be larger then number of nodes in the input layer so that the neural network classifier will have higher capabilities to classify more type of fingerprint images with lesser training time. However, upon finishing developing the first version of neural network classifier with 256 input nodes, 512 hidden nodes and 5 output nodes have a very slow computational speed and it also takes a very long time to load and save all the weights to database. Since it has a very slow computational speed, it is useless to train it. Then we begin to alter the neural network classifier structure by changing the number of node in the hidden layer from 512 to 256 and we still encounter same the problem, and we continue to reduce the number of the hidden layer from 256 to

64 and until we reaches 10 nodes. The idea of reducing number of nodes it from analyzing the sigmoid functions, we realized that if there are too many nodes; the sum of all the value of weight x input value for any hidden nodes will be very large. The sig(x) value will approach 0 if the summation value of the weight x input because the activation function for this neural network classifier is $sig(X) = \frac{1}{1+e^{-x}}$. Therefore we conclude that our early assumption on neural network is not accurate.

Comparing fuzzy logic and fuzzy neural with neural network classifier, fuzzy logic and fuzzy neural are more difficult to be develop because of the difficulty of creating a good sets of production rules and determines the fuzzy sets value. The ease of developing a fuzzy logic classifier is that fuzzy logic provides reasons on every decision (classification) the fuzzy logic classifier made and this ease the debugging process during the developing period. In the earlier development of fuzzy logic, we first decided to set the fuzzy input set to have only 3 membership functions. After developing the fuzzy logic classifier with only 3 membership functions we realized that there are a lot of clashes between rules (a same rule that can be use by two different conclusions) due to limited number of rules 5^3 (125). There to reduce the number of rules that clashes, we decide to add the number of membership function from 3 membership functions to 5 membership functions that allow us to have 5^6 (15625) rules. Although the number of rules is large, there are only a partial of it is being used, and this also success fully reduce the number of clashes.

Fuzzy neural classifier is the hybrid of the two methodologies, therefore it has the advantages and disadvantage of the both classifier. In the earlier development of fuzzy neural we thought that fuzzy neural will have the best performance of all the three classifier but it does not achieve the performance we expected. Our expectation on fuzzy neural is not only to achieve high performance classification but also the ability of reducing the burden of fuzzy logic classifier and neural network classifier. Therefore we decided to reduce the complexity of fuzzy logic and neural network which being applied in the fuzzy neural classifier. The fuzzy logic in this fuzzy neural classifier has only 2 membership functions at the fuzzy input set and has 7 membership functions at the fuzzy output set. The neural network in this fuzzy neural classifier also has a simpler structure which has 256 input nodes, 5 hidden nodes and 2 output nodes. Although the expected performance is not being achieve but it definitely shows the ability to reduce the burden of fuzzy logic classifier and neural network classifier when combining them together.

Lastly, there are several constraints of the direction reader and also the three classifiers. The constraint of direction reader is that is does not conform to scale invariance; it will fail to generate direction if the size of the fingerprint image is different. The constraints for the three classifier is that they do not conform to translation and scale invariance and only able to support ±10 degree of rotation.

Chapter 8: Conclusions

8.1 Summary of Research

In this research, we wanted to search for the best classifier from three different methodologies (fuzzy logic, fuzzy neural, neural network) and enhance our previous fingerprint recognition system by 75% faster. We have succeeded develop the three classifier and perform accuracy and efficiency test on them and the fingerprint recognition process of our previous system was enhanced more then 5 times by the three classifier that we built. We have also develop a direction reader that have the ability to extract information (directions) from a given fingerprint images.

8.2 Analysis of the Work

Earlier of this research we are determine to search for the best methodology to perform fingerprint classification out of the three that we build. Although neural network classifier overall perform better then fuzzy neural and fuzzy logic, but we do not concluded that neural network classifier is the best this is because from this research we realized that there are more aspects which have to be considered rather then just the methodology's performance (efficiency and accuracy of classifying) itself. Examples of other aspects that need to be considered are such as the suitability of the methodology in a given problem domain, the choice of feature representation and the degree of the feature representation discrimination. All these are important factors in determining a good classifier methodology.

Therefore the conclusion that we wanted to make from the efficiency and accuracy experiments is that performance of a classifier methodology can be enhanced if we select the correct methodology that fit a problem domain and type of feature representation that being used.

8.3 Practical Contributions of the Work

We successfully implemented fuzzy logic classifier, neural network classifier and combine them to become a fuzzy neural classification for fingerprint classification. We also created a way to extract features from a given fingerprint images and convert the extracted features in to feature vectors that can be the input of the three classifiers. Accuracy and efficiency test were applied on them and we draw conclusions from the results. We also developed a neural network class, a fuzzy logic class and a fuzzy neural class which can be reused in other problem domain.

8.4 Recommendations for the Future Work

The process of enhancing our fingerprint recognition system is like a spiral model, after we have enhanced our fingerprint recognition with classification. The next step to enhance our fingerprint recognition with classification is to enhance the image preprocessing of our fingerprint recognition so that this system can be applied on real fingerprint images and followed by enhancing the matching algorithm by developing a better feature representation. After if we succeeded to develop a better matching algorithm, we will again come back and enhance current classifier with higher expectation such as higher accuracy, higher efficiency and enhance the number of the fingerprints that this system can support.

References

[1] Arun D. Kulkarni, Computer Vision and Fuzzy-Neural Systems, Prentice Hall, New Jersey, 2001.

[2] George F Luger, Artificial Intelligence 4th edition, Addison Wesley, United States of America, 2002.

[3] David H. Kemsley, Tony R. Martinez, Douglas M. Campbell, "A Survey of Neural Network Research and Fielded Applications", In International Journal of Neural Networks: Research and Applications, vol. 2, No. 2/3/4, pp.123-133, 1992.

[4] Salil Prabhakar, "Fingerprint Classification", Fingerprint Classification and Matching Using a Filterbank, Michigan State University, pp. 119-150, 2001.

[5] Charles L. Wilson, James L. Blue and Omid M. Omidvar, "Improving Neural Network Performance for Character and Fingerprint Classification by Altering Network Dynamics", World Congress on Neural Networks Proceeding, II, Washington DC, pp. 151-158, July, 20 1995.

[6] C.L Wilson, G.T. Candela and C. I. Watson, "Neural Network Fingerprint Classification", J. Artificial Neural Networks,1, No.2, 1993;

[7] Faramarz Valafar, "Neural Network Applications in Biological Sequencing", San Diego State University, California, USA.

[8] Juha Vesanto and Esa Alhoniemi, "Clustering of the Self-Organizing Map", IEEE Transaction on Neural Networks, Vol. 11, No. 3, May 2000, pp. 586-600.

[9] Hassan Masum's Homepage, http://www.carleton.ca/~hmasum, September 2003.

[10] Pavel Berkhin, "Survey of Clustering Data Mining Techniques", Accrue Software Inc.

[11] fide TM, "What is Fuzzy Logic ?", http://www.aptronix.com/index-fuzzy.htm, September 2003.

[12] fuzzyTECH Home Page, http://www.fuzzytech.com, September 2003.

[13] von Altrock, C., "Fuzzy Logic in Automotive Engineering", Circuit Cellar INK, The Computer Application Journal, Issue 88, November 1997.

[14] Gary Neill, "Applying Fuzzy Logic to Business and Finance", http://www.risksys.com/media/pdf_files/Optimus_UBS_E.pdf, 2002.

[15] Gierend, Ch., "Fuzzy Logic Control of a Waste Incineration Plant", 5. Aachen Fuzzy-Symposium, 1995.

[16] Gebhardt, J., "New Industrial Applications of the Fuzzy-PLC", Proceedings of the 3. European Congress on Fuzzy and Intelligent Technologies (EUFIT 95), Aachen, 1995.

[17] von Altrock, C., "Fuzzy Logic and Neuro Fuzzy Applications Explained", Prentice Hall, ISBN 0-13-368465-2, 1995.

[18] "Applications for fuzzy Logic", http://www.mines.u-nancy.fr/~gueniffe/CoursEMN/I31/fcourse/fuzzy_appl.10.html, September 2003.

[19] Wang, F., "Fuzzy supervised classification of remote sensing images", IEEE Transactions on Geosciences and Remote Sensing, 28, 194-201, 1990.

fingerprint images and followed by enhancing the matching algorithm by developing a better feature representation. After if we succeeded to develop a better matching algorithm, we will again come back and enhance current classifier with higher expectation such as higher accuracy, higher efficiency and enhance the number of the fingerprints that this system can support.

[20] Giles M. Foody, Doreen S. Boyd, "Fuzzy Mapping of Tropical Land Cover from Remotely Sensed Data: the Use of an Artificial Neural Network for Sub-Pixel Land Cover Estimation", Department of Geography, University of Salford and University of Southampton, United Kingdom, 1996.

[21] V.Raviprakash Emani, "Image Classification using Fuzzy Logic", http://b.domaindlx.com/evrprakash/Fuzzy.html, September 2003.

[22] A. Mufit Ferman, A Murat Tekalp, "A Fuzzy Framework for Unsupervised Video Content Characterization and Classification", Sharp Labs of America Ins, and Dept. of Electrical and Computer Engineering, University of Rochester, 2001.

[23] Dzung L. Pham, Jerry L. Prince, "An Adaptive Fuzzy C-Means Algorithm for Image Segmentation in the Presence of Intensity Inhomogeneities" , Dept. of Electrical and Computer Engineering, The Johns Hopkins University, 1997.

[24] Lucila Ohno-Machado, Staal Vinterbo, Griffin Weber, "Classification of Gene Expression Data Using Fuzzy Logic", Decision Systems Group, Brigham and Women's Hospital, 2001.

[25] "Clustering Gammas by Using Fuzzy Logic", http://agata.pd.infn.it/documents/glp5152003/CarlosRossiAlvarez.pdf, September 2003.

[26] A. Meier, C. Savary, G. Schindler, Y. Veryha, "Database Schema with Fuzzy Classification and Classification Query Language", University of Fribourg, 2001.

[27] Qigang Gao, Dan Qing, Siwei Lu, "Fuzzy Classification of Generic Edge Features", Dalhousie University, and Memorial University of Newfoundland, 2000.

[28] L. García-Pérez, M. C. García-Alegre, J. Marchant, T. Hague, "Fuzzy decision system for threshold selection to cluster cauliflower plant blobs from fields visual images", Instituto de Automática Industrial, and Silsoe Research Institute, 2000.

[29] "Land Use Map Accuracy Assessment from Fuzzy Logic Image Classification", http://www.geomatica.ufpr.br/docentes/ felippe/pessoal/accufuzz.pdf, September 2003.

[30] Dr. Ashok W. Deshpande, "Application of Fuzzy Logic to River Water Quality Classification", Former Deputy Director, National Environmental Engineering Research Institute, India, 2001.

[31] William Silvert, "Ecological Impact Classification with Fuzzy Sets", Marine Environmental Science Division, Bedford Institute of Oceanography, 1997.

[32] V.K. Sagar, D.B.L. Ngo, K.C.K. Foo, "Fuzzy Features Selection for Fingerprint Identification", Dept. of ESE, University of Essex, Colchester, 1995.

[33] V.K. Sagar, J.B. Koh, Alex, "Fingerprint Features Extraction by Fuzzy Logic and Neural Networks", School of Applied Science, Nanyang Technological University, 1999.

[34] von Altrock, C. ,"Introduction to Neuro Fuzzy Technologies", http://www.fuzzytech.com/binaries/e_p_nf1.ppt, Prentice Hall, 1998.

[35] Liqing Zhou, "The Application of Neural Networks and Fuzzy Technique in Speaker-Independent Speech Recognition", Dept. of Radio Engineering, Beijing University of Posts and Telecommunications, 1997.

[36] Patrik Eklund, Robert Fuller, "A Neuro-Fuzzy Approach to Medical Diagnostics", Dept. of Computer Science, Abo Akademi University, and Eotvos Lor and University, 1993.

[37] Horia-Nicolai Teodorescu, "An Overview of the Bio-Medical Applications of Fuzzy and Neuro-fuzzy Systems", http://agdec2.technion.ac.il/~granot/softcomp/An%20Overview%20of%20the%20Medical%20Applications%20of%20Fuzzy%20and%20Neuro.doc, September 2003.

[38] J.A. Swope, N.K. Kasabov, M.J.A. Williams,"Neuro-Fuzzy Modelling of Heart Rate Signals and Application to Diagnostics", Dept. of Computer and Information Science, and Dept, of Medicine, University of Otago, 2000.

[39] G. J. Vachtsevanos, S. S. Kim, J. R. Echauz, and V. K. Ramani, "Neuro-Fuzzy Approaches to Decision Making: Application to Check Authorization from Incomplete Information", School of Electrical anc Computer Engineering, Georgia Institute of Technology, 1995.

[40] Irena Koprinska and Nikola Kasabov, "An Application of Evolving Fuzzy Neural Network for Compressed Video Parsing", Dept. of Information Science, University of Otago, 2000.

[41] Marek Slonski, "Application Of ANFIS Neuro-Fuzzy System for Prediction of Concrete Fatigue Durability", Institute of Computer Methods in Civil Engineering, Cracow University of Technology, 2000.

[42] Michael A. Lee, David Wessel, "Real-Time Neuro-Fuzzy Systems for Adaptive Control of Musical Processes", International Computer Music Association, 1994.

[43] Gary R., George P. E., Frank Cardullo, "Application of Neuro-Fuzzy Systems to Behavioral Representation in Computer Generated Forces", State University of New York at Binghamton, 1999.

[44] K. Väinämö, T. Mäkikallio, M. Tulppo, J. Röning, "A Neuro-Fuzzy Approach to Aerobic Fitness Classification: a multistructure solution to the context-sensitive feature selection problem", University of Oulu, and Merikoski Rehabilitation and Research Center, Nahkatehtaankatu, 1998.

[45] Bellei E., Guidotti D., Petacchi R., Reyneri L., Rizzi I., "Applications of Neuro-Fuzzy Classification Evaluation and Forecasting Techniques in Agriculture", Politecnico of Torino, and Scuola Superiore S.Anna, 2001.

[46] "A brief Tour in the World of Fingerprints", http://www.xs4all.nl/~dacty/schedule.htm, Fingerprints TK, August 2003.

[47] W.K. Cheng, B.Y. Ooi, "Fingerprints Recognition System", Undergraduate Research Training Report, School of Computer Science, Universiti Sains Malaysia, 2003.

[48] Yuan Yao, Paolo Frasconi, and Massimiliano Pontil. Fingerprint Classification with Combinations of Support Vector Machines. Technical report. Department of Mathematics, City University of Hong Kong, Hong Kong.

Penerbitan (X2)

:osalina Abdul Salam

, rom: <iajit@iajit.org>
To: <iajit@iajit.org>; <rosalina@cs.usm.my>
ent: Friday, April 20, 2007 11:34 AM
ubject: Paper Submission #245 for IAJIT

T ınk you for your submission to IAJIT. Below is a copy of the information you submitted for your records.

Paper ID: 245

1 ;e: Fingerprint Image Enhancement using Enhanced Gabor Filter (EnGB)

Author1
ıme: Rosalina Abdul Salam
ffiliation: School of Computer Sciences
Country: Malaysia
nail:rosalina@cs.usm.my
ıthor2
ame: Pan Fong Pei
Affiliation: School of Computer Sciences
untry: Malaysia
ıail:panfong36@yahoo.com
uthor3
Name:
filiation:
ɔuntry:
Email:
ıthor4
ıme:
filiation:
Country:
nail:
ıthor5
ame:
Affiliation:
ɔuntry:
nail:
Other Authors:
C ıtact Author #: 1
(ıtact Alt Email:
C ıtact Phone: 604-6532486
Keywords: Fingerprint, pre-processing, Wiener Filter, Gabor Filter, Enhanced Gabor Filter
/ tract: Fingerprints are the most regular used biometric features for personal identification. A fingerprint is made of series of ridges and furrow
c he surface of the finger. The uniqueness of a fingerprint can be determined by the pattern of ridges and furrows as well as the minutiae points. Pre-processing for fingerprint is very important before any further processing can be applied. This paper focuses on using different filtering techniques in pre-processing. Differences of Gabor filters and Wiener filter were compared. A hybrid of Gabor Filter and Wiener Filter, that is the E ıanced Gabor Filter (EnGF) were implemented. Results showed that the enhanced filter is the best.
(nments:
Paper: included

--

Fingerprint Image Enhancement using Enhanced Gabor Filter (EnGB)

Rosalina Abdul Salam, Pan Fong Pei
School of Computer Sciences, Universiti Sains Malaysia, 11800 USM Malaysia

Abstract: *Fingerprints are the most regular used biometric features for personal identification. A fingerprint is made of series of ridges and furrows on the surface of the finger. The uniqueness of a fingerprint can be determined by the pattern of ridges and furrows as well as the minutiae points. Pre-processing for fingerprint is very important before any further processing can be applied. This paper focuses on using different filtering techniques in pre-processing. Differences of Gabor filters and Wiener filter were compared. A hybrid of Gabor Filter and Wiener Filter, that is the Enhanced Gabor Filter (EnGF) were implemented. Results showed that the enhanced filter is the best.*

Keywords: Fingerprint, pre-processing, Wiener Filter, Gabor Filter, Enhanced Gabor Filter..

1. Introduction

Fingerprint of an individual is unique and unchanged over a lifetime. Fingerprint images have been applied over a century and become the most tremendously used in biometric identification. Among all the biometric techniques, fingerprint identification is the oldest method, which has been successfully used in numerous applications. Fingerprint identification is emerging computer-based scientific discipline, which can help police and other law enforcement officials in criminal investigation, and in biometric systems such as civilian and commercial identification devices. Attorneys, scientific organizations and the government also benefit from the extraction of objective information from fingerprint identification.

In particular, fingerprints are the most regular used biometric features for personal identification. Fingerprint is made of series of ridges and furrows on the surface of a finger. The uniqueness of a fingerprint can be determined by the pattern of ridges and furrows as well as the minutiae points. A ridge is a single curved segment, whereas a valley is the region between two adjacent ridges. Minutiae characteristics are local discontinuities in the fingerprint pattern that represent terminations and bifurcations [4]. A ridge termination is the point where a ridge ends. Meanwhile, a ridge bifurcation is the point where a ridge forks or diverges into branch ridges. Many researchers investigate on different methods for fingerprint enhancement. These can be found in [1-11].

However, the ridge structures in fingerprint images are rarely of perfect quality and not always well defined. The quality of most fingerprints may be degraded or corrupted with some elements of noise that due to several factors including impression conditions and variations in skin such as the injured part on the finger skin and the surroundings in which fingerprint was taken. Neither those poor quality ridges structure nor the not well-defined fingerprint images cannot be accurately detected. An essential step in studying the statistic of fingerprint minutiae is to accurately extract minutiae from fingerprint images. In order to do this, the fingerprint image will have to be filtered and enhanced.

In this paper, we proposed a method of fingerprint image enhancement using the combination of Hong [5], Gabor filtering technique and Wiener Filtering technique [4]. Gabor filtering technique consists of both frequency selective and orientation selective properties and it have optimal joint resolution in both spatial and frequency domains. Thus, it is the benefit of using Gabor filtering as band pass filters to remove the noise and preserve the true ridge and valley structures and Weiner filter is good for noise reduction. In addition to these approaches, we suggest to add image segmentation process as an additional stage. The purpose of image segmentation process is to segregate the foreground regions in the image from the background regions.

2. Methodology

The region of interest area in fingerprint image can be divided into thee categories: well-defined region, recoverable corrupted region and unrecoverable corrupted region. In most cases, acquired fingerprints are degraded, influence of noise or incomplete. Thus, to reduce the rejection rates during the matching stages, fingerprint images have to be enhanced prior to matching. Therefore, a reasonable enhancement algorithm is necessary to improve the clarity of ridge and valley structures of fingerprint images in recoverable regions and to mask out the unrecoverable regions.

The goal of fingerprint images enhancement is to remove noise and any irrelevant information. Moreover, to improve the clarity of fingerprint ridge structures images to facilitate the extraction of ridges and minutiae in recoverable regions and mask out the unrecoverable regions. This corruption on a certain ridges and valleys will cause invisible or blur ridges and valleys that do not provide sufficient information about the actual ridge and valley structures. Our main objectives are to improve visual appearance by removing noise, irrelevant information and to improve the ridge structures.

We have implemented an algorithm and it is built on the techniques developed by Hong [5] and Greenberg [4]. It consists of two main stages, namely image pre-processing and binarization. In our proposed methodology in the image pre-processing stage, there are three stages: segmentation, orientation estimation and filtering. This can be seen in Figure 1.

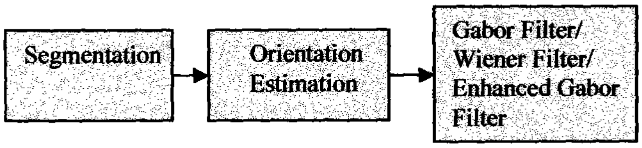


Figure 1. Image Pre-Processing

Segmentation is a process where the input image will be divided into blocks and grayscale variance is calculated for each block in the image to determined foreground regions and background regions.

Orientation estimation is where it partition an image into square blocks and gradient is calculated for every pixel, in x and y directions. Then, the orientation vector for each pixel can be derived by performing an averaging operation on all the vectors orthogonal to the gradient pixels in the block. Due to the possibility on the ridge orientation varies slowly in a local neighborhood then the orientation image will be smoothed using a low pass filter to reduce the effect of outliers.

Filtering is the process of enhancing the fingerprint image. This can be seen on ridges. Gabor Filtering is a band pass filter that has both frequency-selective and orientation-selective properties. Therefore, Gabor filter enhanced the ridge oriented in the direction of the local orientation, and decrease anything oriented differently. Eventually, Gabor filter increase the contrast between the foreground ridges and the background, whilst effectively suppress noise. In Wiener Filtering the main purpose for a pixel-wise adaptive Wiener method is noise reduction. The filter used in this Wiener filtering (WF) method is based on local statistics estimated from a local neighborhood (η) of size 3x3 of each pixel. The proposed method uses the hybrid of both Gabor Filtering and Wiener Filtering.

According to Greenberg [4], the advantage of WF is effectively removes noise while preserving important image features like edges. On the other hand, the existing Gabor filtering (GF) approached by Hong *et al.* takes into account of the ridge frequency information. Both ridge orientation and ridge frequency information have to be employed together to allow accurate tuning of the Gabor filter parameters, consequently leads to optimum enhancement results. In our research we are using the hybrid methodology of WF and GF, called as enhanced Gabor filter (EnGF). Comparison between Gabor Filter and Wiener Filtering is shown in Table 1.

Table 1. Comparison of Wiener Filter and Gabor Filter

Wiener Filtering
- is the optimal linear estimator - achieve best performance when the original signal is known - attenuate the noisy data coefficients in a Karhunen-Loeve basis - straightforward use of rank-reduction without the need of eight-decomposition of the input covariance matrix - input signal is projected in a lower-dimension subspace, and filter optimization occurs within this subspace - reduce the number of filter coefficients that need to be estimated, leading to faster convergence speed
Gabor Filtering
- capacity in handling nonlinear data - a set of band pass filters can efficiently remove the undesired noise and preserve the true/valley structures - have both frequency-selective and orientation-selective properties - have optimal joint resolution in both spatial an frequency domains - have similar characteristics to those of the human visual system - directly extracted from gray-level images - applied locally to extract local image features - applied to the whole image through a convolution/filtering process - can break down image content to different scales, locations, and orientations that can be extracted effectively for recognition

3. Experimental Results

Experiments were conducted on a set of real fingerprint images. The first stage is the segmentation process. The result of segmentation on a fingerprint image based on variance thresholding is shown in Figure 2. Variance image in Figure 2(b) illustrates the central fingerprint area exhibits a very high variance value, whereas the regions outside this area have a very low variance. Therefore, a variance

Filter (EnGF) that are the hybrid of Weiner Filter and Gabor filter are the best.

Future work such as integrating more enhancement techniques such as histogram equalization, ridge frequency estimation and thinning process will be very useful. All the grayscale input images can illustrates their stretches contrast for gray-level and the transformation and improves the ability of many image features detection. Besides that, wavelength estimation canbe employed for ridge frequency estimation. Although wavelength estimation is much difficult to assess especially for the real fingerprint image, the average ridge wavelength values between the two images are significantly to be used as ridge frequency estimation. Thinning process is to preserves the connectivity of the ridge structures while forming a skeletonised version of the binary image. This skeleton image is then used in the subsequent extraction of minutiae. As a results higher accuracy of fingerprint images will be produced.

References

[1] Ammar, H. H., Zhen, X., Nassar, D. *et al.* Web-Based Test Bed for Fingerprint Image Comparison. Department of Computer Science and Electrical Engineering, West Viginia University, Morgantown.

[2] Bamberger, R. H. and Smith, M. J. T. Narrow Band Analysis of A Filter Bank for the Directional Decomposition of Images. IEEE Trans. on Signal Processing, pp. 1739-1742. (1990).

[3] Bernard, P. and Michel, B. Constrained Wiener Filtering. IEEE Transactions on Information Theory, vol. IT-33, no. 1, pp. 160-166. (January, 1987).

[4] Greenberg, S., Aladjem, M., Kogan, D. *et al.* Fingerprint Image Enhancement Using Filtering Techniques. Electrical and Computer Engineering Department, Ben-Gurion University of Negev, Israel.

[5] Hong, L., Wan, Y. and Jain, A. Fingerprint Image Enhancement Algorithm and Performance Evaluation. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 20, no. 8, pp. 777-789. (1998).

[6] Klimanee, C. and Nguyen, D. T. On the Design of 2-Gabor Filtering of Fingerprint Image. IEEE. pp. 430-468. (2004).

[7] Mario, D., and Maltoni, D. Direct Gray-Scale Minutiae Detection In Fingerprints. IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no. 1, pp. 27-40. (1997). [23] Kamei, T. and Mizoguchi, M. Image Filter Design for Fingerprint Enhancement. Information Technology Research

[8] Sang-il, P. and Smith, M. J. T. Fingerprint Enhancement Based on the Directional Filter Bank. Center for Signal and Image Processing Georgia Institute of Technology: Atlanta, Georgia. (2000).

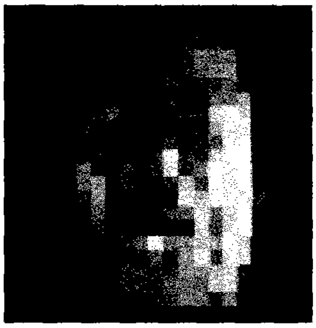
[9] Sherlock, B. G., Monro, D. M. and Milard, K. M. Fingerprint Enhancement by Directional Fourier Filtering. IEE Proc. Vis. Image Signal Process, vol. 141, no. 2, pp. 87-94. (1994).

[10] Xudong, J. A study of fingerprint image filtering. Centre of Signal Processing, Nanyang Technological University, Singapore. IEEE. pp. 238-241. (2001).

[11] Wang, S. and Wnag, Y. Fingerprint Enhancement in the Singular Point Area. IEEE Signal Processing Letters, vol. 11, no. 1, pp. 16-19. (2004).

threshold is used to separate the fingerprint area from the background regions. The final segmented image is shown in Figure 2(c), which formed by assigning the regions with a variance value below the threshold to a grey-level value of zero. These experiments show that the foreground regions segmented by variance thresholding method comprise only for containing the fingerprint ridge structures. Thus, the variance thresholding method is effective for discriminating the foreground area from the background regions.

(a)



(b)

(c)

Figure 2. Real fingerprint image (a) Original image, (b) Variance image, (c) Segmented image

The second stage is the orientation estimation. The Gabor Filtering stage is for filtering along the local ridge orientation to enhance the ridge structure and reduce noise. Therefore, it is important to obtain an accurate estimation of the orientation field. Since the orientation estimation stage plays a central role in the enhancement process, we have applied the default set of parameters, which have been specified by Hong [5] throughout the experiments: there are an averaging block size of 16x16, and a Gaussian filter size of 3x3.

The Gabor filtering stage is the central part of the enhancement algorithm because this stage performs the actual enhancement of the fingerprint image. The aim of the filtering stage is to enhance the clarity of the ridge structures while reducing noise in the image. In order to assess the good performance of the Gabor filtering, parameter selection is an essential part that will affect the enhancement artifacts and blurring the ridge structures.

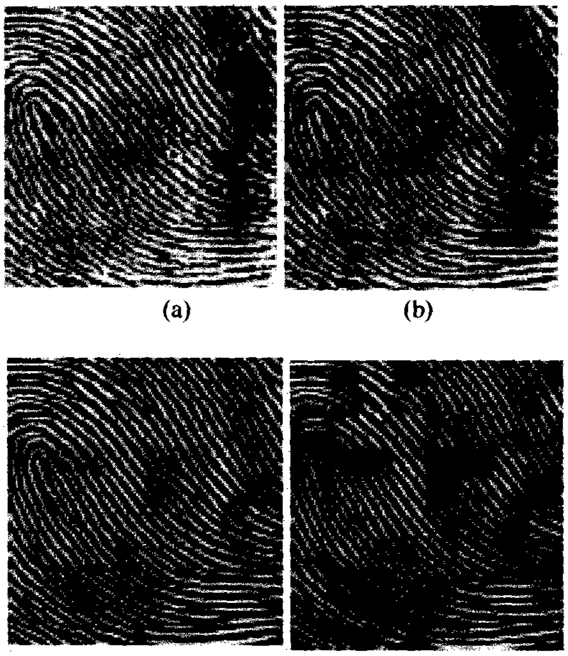
The Gabor filter parameter σ_x and σ_y are controlling the bandwidth of the filter and have significant effect on the enhancement results, therefore the parameter values must be chosen carefully. The value of σ_x is to determine the degree of contrast enhancement between ridges and valleys, and σ_y is to determine the amount of smoothing applied to the ridges along the local orientation. Figure 3 illustrates the result of using different values of σ_x and σ_y to apply the Gabor filter to a fingerprint image

Figure 3(b) illustrates a smoothed version of the original image with the Gabor filter evolving into the shape of a pure low pass filter. But, if values of σ_x and σ_y, are too small the filter is not effective. On the other hand, Figure 3(d) shows a large values of σ_x and σ_y that lead to enhancement artifacts and a significant amount of blurring of the ridge structures.

The experiment conducted the Gabor filter with parameter, $k_x = 0.5$ and $k_y = 0.5$ provides a reasonable enhancement shown in Figure 3(c). These experiments indicate an improved contrast between the ridge and valley structures.

Figure 4 illustrates a medium quality fingerprint image as an input for the Gabor filter. The enhancement results show that the filter preserve the continuous of the ridge flow pattern and enhances the clarity of the ridge and valley structures. Figure 4(b) successfully generate a result, which is not only reducing noise in the image, the filter is able to fill in small breaks that occur within the ridges. But, there are slightly blur among the minutiae points in Figure 3(b). The reason of those undesired blurring among the minutiae is due to the Gabor filter is designed to enhance along the ridges line that are parallel to one another and have a consistent orientation. In other words, minutiae points occur as local discontinuities in the ridge flow pattern, it may cause the local orientation and frequency to be in accurately estimated. Eventually, the results of applying the filter to regions that have minutiae points are less effective in enhancing the image, when compare to minutiae-free region.

In contrast, by applying the enhancement algorithm on a low quality image as shown in Figure 5, the results are not very good. It can be concluded that the filter has difficulty with regions of the image that are seriously corrupted and degraded.



(a) (b)

(c) (d)

Figure 3: Enhanced fingerprint image using Gabor filter with different parameter (a) Original image, (b) $k_x = 0.2$ and $k_y = 0.2$, (c) $k_x = 0.5$ and $k_y = 0.5$, (d) $k_x = 0.9$ and $k_y = 0.9$

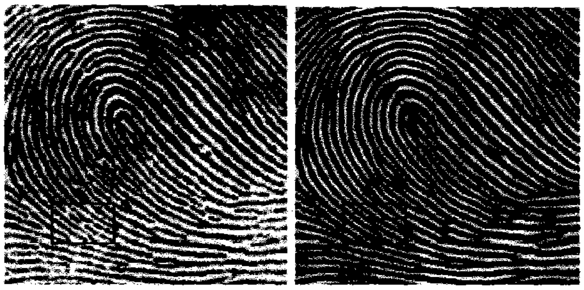


Figure 4: Gabor filter reduces noise and fill in small break with parameter, $k_x = 0.5$ and $k_y = 0.5$

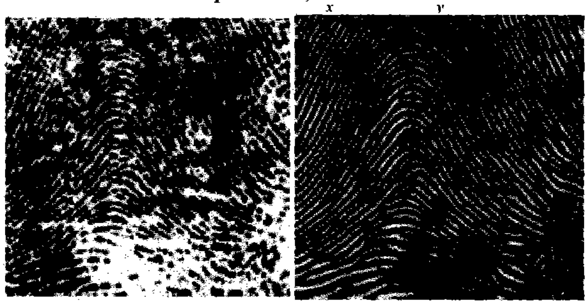
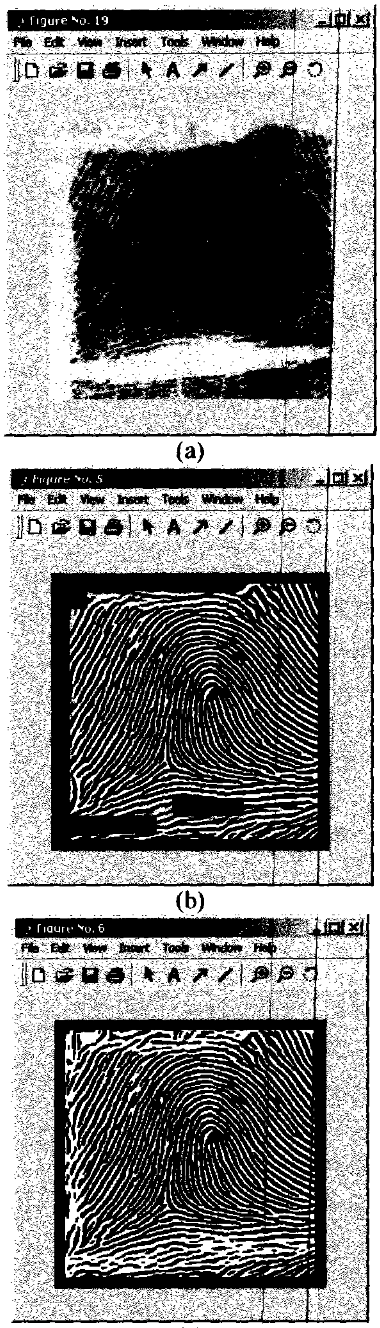


Figure 5: Gabor filter on corrupted image with parameter, $k_x = 0.5$ and $k_y = 0.5$

Results with the use of Wiener Filter and Gabor Filter and Enhanced Gabor Filter (EnGF) that is the hybrid of Weiner Filter and Gabor filter are shown in Figure 6 .



(a)

(b)

(c)

Figure 6. Output of (a) Wiener Filter Figure, (b) Gabor Filter and (c) Enhanced Gabor Filter (EnGF)

4. Conclusion and Future Works

The primary focus of the work in this paper is on the fingerprint images enhancement. Firstly, we have implemented a series of techniques for fingerprint images enhancement. Experiments were then conducted using real fingerprint images in order to provide a good evaluation on the performance of the implemented algorithm. The use of real fingerprint images rely on qualitative measures of inspection, but also can provide a more realistic evaluation as they provide a natural representation of fingerprint imperfections like noise and corrupted elements. Results shows that the output of the Enhanced Gabor

Fingerprint Recognition System using Neurofuzzy

Abstract

Clustering of fingerprints can help to reduce the complexity of the search process in a database. This can be done by grouping fingerprints with the same characteristic in the same group. The matching algorithm can compare stored fingerprint codes with only one cluster instead of the entire database. In this research, we classify fingerprints into five categories which are arch, left loop, right loop, whorl, and others. The last category is use to categorize fingerprint pattern other then the four categories. Finally, experiments were carried out to show that clustering can reduce the recognition time. Experiments were carried out using neural network classifier, fuzzy logic and neuro-fuzz. Results showed that neural network classifier is the best among the three.

Keywords: **Fuzzy Logic, Neural Networks, Fingerprint, Clustering**

1. Introduction

There are many different ways of acquiring fingerprints, such as frustrated total internal reflection(FTIR) and optical methods [1], CMOS capacitance [1],[2], thermal method [1],[2], ultrasound [1] and re-imaging [2]. Once fingerprints images were captured there are a number of different methods that can be used to extract important information. There are two possible details that can be identified in a fingerprint. The first one is the directional field [3]. This method describes the coarse structure or the basic shape of a fingerprint and defines as the local orientation of the ridge-valley structures at each position in the fingerprint. The directional field is normally used for fingerprints classification. The second is the minutiae [3]. This method provides details of the ridge-valley structures such as ridge-endings and bifurcations. The minutiae will be used for one-to-one comparison of two fingerprints. In this research the minutiae extraction were used for the recognition purpose.

The second stage is the recognition process, which is the main focus of the paper. The recognition time was extremely reduced with the used of clustering technique. There are a number of methods available for the recognition stage such as Neural-Network [4], a Correlation-Based Fingerprint Verification System [5], fingerprint matching using feature space correlation [4], combination of flat and structural approaches [6], fuzzy logic, neuro fuzzy and computational intelligence in fingerprints identification [7].

A simple matching algorithm can recognize a fingerprint image easily. However, with a huge database, the system will be very slow. There are a few different clustering algorithms such as hierarchical methods, partitioning methods, density-based algorithms, grid-based methods, clustering algorithms used in machine learning such as neural networks, fuzzy logic and fuzzy neural [8]. In our research, neural network, fuzzy logic and neuro fuzzy clustering system was developed and tested against a simple matching algorithm. Results were compared between all these techniques.

2. Motivation

In our previous work [9], we had developed a fingerprint recognition system which is minutiae based and uses Euclidean distance for the fingerprint matching. The system is able to perform verification and recognition. The system will extract features from the provided fingerprint image and then the extracted feature will be use to create a finger code. This is based on the arrangement of the fingerprint's minutiae and it is different for every fingerprint. The finger code is then stored in the database to perform recognition and verification. A fingerprint recognition system has to tolerate three problems such as, transition, rotation and scale. In our matching algorithm we had applied finger code to solve the transition and rotation problem. Figure 1 shows our earlier work.

In the earlier work, the recognition time was very long. In order to overcome the problem, clustering method can be applied. This was inspired by fact that fingerprints can be grouped together with the same characteristics [8].

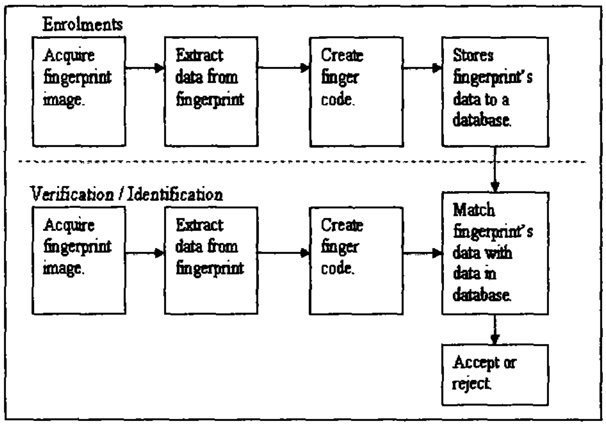


Figure 1. Fingerprint recognition system

Classification can be done with many different methods. The most common are neural networks, fuzzy logic, simulated annealing, graph matching and neuro fuzzy or fuzzy neuro [10]. Combination of neural network and fuzzy logic can be done in many different ways. Neural network models are able to provide algorithms for numeric classification, optimization, and associative storage and recall while fuzzy logic able to work at the semantic level and provide a solution to process inexact or approximate data. Fuzzy neural is the combination of neural network with fuzzy logic, this combination will provide us even greater representation power, higher processing speed, and are more robust than conventional neural network. There are many other researches proposed and claim that fuzzy neural is good.

In our research we proposed and developed fuzzy neural classifier for our fingerprint classification system. Besides testing the accuracy and efficiency of fuzzy neural classifier, we also implemented neural network classifier and fuzzy logic classifier to make a comparison with fuzzy neural classifier. The comparison will cover areas such as the accuracy and efficiency.

3. Methodology

A module that is responsible to convert the fingerprint pattern from a 256 x 256 pixels grayscale image into a 256 columns array was build. The grayscale image will first be divided into 256 blocks. Each of this block's direction will be read and store into the 256 columns array. These directions are obtained by reading the fingerprint pattern according to the ridges and valleys of the fingerprint. These directions are divided into six categories (90°-270°, 0°-180°, 30°-210°, 60°-240°, 120°-300° and 150°-330°) and every direction will be represented with the numbers from 1 to 6. As the end result the 256 columns array will be the input for the three clustering methods.

Our study showed that fingerprint can be classified into approximately seven different types such as arch, tent arch, loop, double loop, pocked loop, whorl, and mixed figure [11]. We proposed five categories that are whorl, right loop, left loop, arch and others. The main four categories are the most common type of fingerprints. The fifth category that is the 'others' can cater for all other types which are not too common.

Figure 2 shows that the newly developed system. A direction reader and a clustering module are added to the system. The function of direction reader is to convert the fingerprint pattern from a 256 x 256 pixels grayscale image into a 256 columns array. The 256 array is fed to the clustering module to perform classification on the fingerprint image. This will decide on which class the fingerprint image belongs to. The classification result is then store into database with the finger code

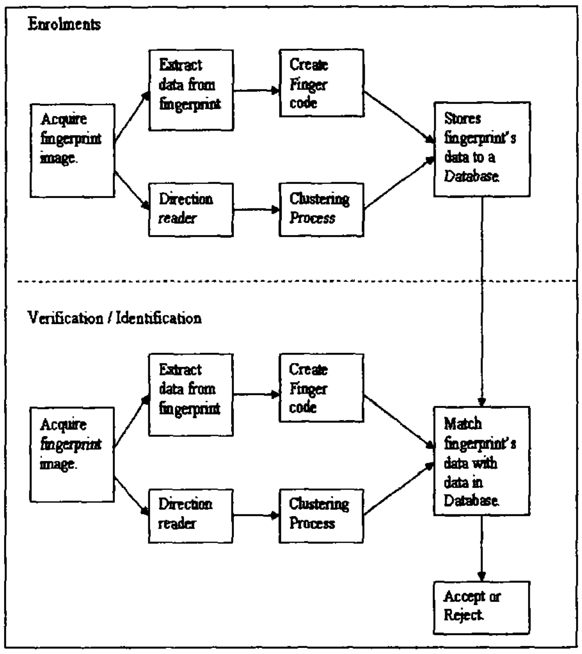


Figure 2. Fingerprint recognition system with the clustering approach

Figure 3 shows the clustering framework. After the direction reading, there are going to be three different classifier methods to classify the fingerprint image and after classifying the system will categorize the image into five classes such as whorl, right loop, left loop, arch and others.

3.1. Neural Network

In our neural network mode we have used 256 input nodes, 10 hidden nodes and 5 output nodes and all nodes are fully connected. Therefore, there are 2610 (256 x 10 + 10 x 5) weights to compute for every clustering process. The directions generated from the direction reader model will be the input for this neural network. Then after setting up the structure of neural network, 120 different fingerprint images were used to train this neural network. These 120 fingerprint images have the mixture of whorl type, arch type, left loop and right loop.

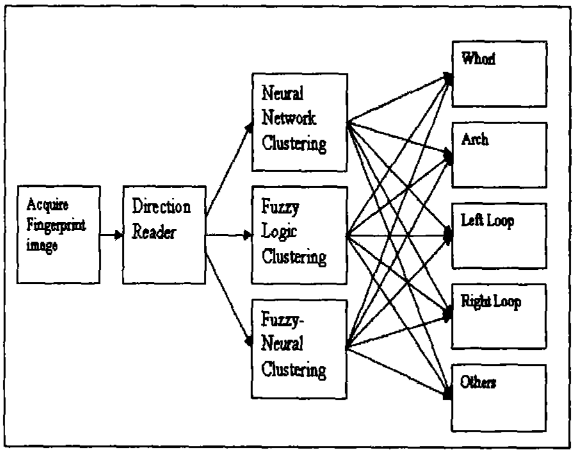


Figure 3. Clustering framework for fingerprint recognition system

The fingerprint will first be segmented into 8 x 8 segments. Then each of these segments will consist of 16 x 16 pixels and will be read by the direction reader. There are total of six directions that this direction reader can differentiate and these directions are (90°-270°, 0°-180°, 30°-210°, 60°-240°, 120°-300° and 150°-330°) and every direction will be represented with the numbers from 1 to 6. A series of direction will be generated by the direction reader and will be the input to the neural network.

This neural network structure was inspired by the neural network using a quick propagation training algorithm build by Prabhakar [12] and neural network by Wilson [6] which uses edge detection to create eigenvector from a given fingerprint image, then Kohonen Loeve transform (KLT) are then being applied to reduce the dimension of the input from the eigenvector, before feeding it to a multi layer perceptron.

The similarity of our neural network compare to the neural network of Prabhakar [12] and Wilson [6] is that all are Multi Layer Perceptron (MLP) with 3 layer network and fully connected. The different between our neural network classifier with Prabhakar [12] is mainly the structure of the neural network and the training algorithm. The structure of the Prabhakar's [12] has one layer of hidden nodes with 20 units, 192 input nodes and has 5 output nodes and the algorithm that they use to train is quick propagation training algorithm. Our neural network uses one layer of hidden nodes with 10 units, 256 input nodes correspond to the 256 features provided form the direction reader and 5 output nodes and the algorithm that we apply to train is backpropagation training algorithm.

Another difference is that the number of weights that neural network of Prabhakar's [12] have to compute is 3940 (192 x 20 + 20 x 5) weights while our neural network only have to compute 2610 (256 x 10 + 10 x 5) weights. The difference between Wilson's [6] and our work are our neural network takes input from direction reader instead of eigenvector and never go through and transformation to reduce the dimension of input.

3.2. Fuzzy Logic

From our study on fuzzy logic and according to [10] fuzzy logic is a technique to mimic human mind to have to ability of reasoning approximately instead of exact. This means that it tries to compute a reason or a decision with the ability to tolerate of imprecision. For examples understand sloppy handwriting, recognize and classify images. In fuzzy logic there is a fuzzy inference system which able to solve a nonlinear mapping of the input data vector into a scalar output by using fuzzy rules. Therefore, to build the rules for fuzzy logic we performed a study on a number of fingerprints images and then generated a graph shown in figure 4, to show the frequency of direction distribution through different type of fingerprint images.

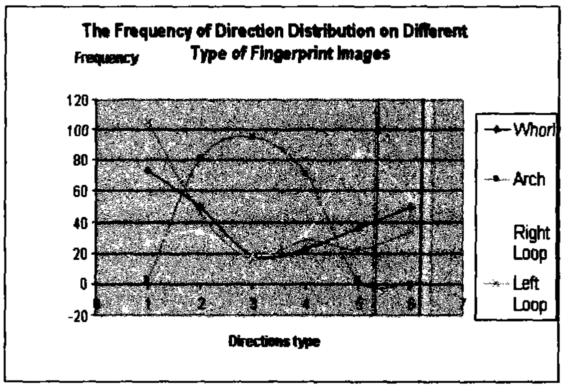


Figure 4. Frequency of Directions Distribution on Different Type of Fingerprint Images

Fuzzy logic's production rules and fuzzy input sets values can be set from the frequency from Figure 4. For examples, we can set the production rule for the Arch type as "if directions type 1 is low and directions type 2 is high and directions type 3 is high and directions type 4 is high and directions type 5 is low and directions type 6 is low then it is a Arch type fingerprint image." The value of the input fuzzy set can also be determine from these graph, for instance we have set that value below frequency 30 is very few, value above 25 is few, value above 45 is considered as average and lower then 55 is few, value above 65 is considered much and value below 75 is average, and if value above 80 is considered very much and below 90 is much.

These rules enables the fuzzy system to maps an input vector to an output vector. The function of fuzzifier is to maps input number into corresponding fuzzy membership in order to activate rules that are in the form of linguistic variables. It takes the input value and determines the degree of belonging to the fuzzy sets along membership functions. Then the inference engine which responsible to map the fuzzy input to fuzzy output by determining the degree to which the antecedent is satisfied for each rules and if then the rules have more then one clause, the fuzzy operators will be applied to obtain one number that represents the result of the antecedent for that rules. There are also possibilities that more then one rules are being fired at the same time. Therefore the outputs for all these rules are then aggregated by combining the fuzzy sets that represent the output into a single fuzzy set. Lastly the defuzzier maps the output fuzzy sets in to a crisp number. There are several methods of defuzzification such as centroid, maximum, mean of maxima, height, and modified height defuzzifier [10].

3.3. Neuro Fuzzy

Figure 5, shows our neuro fuzzy system. The input of this system will be from the direction reader a 256 array. Then the input wills firstly processed by the fuzzy inference system and it will make decision on which neural network classifier will be used. There will be one out of the six multilayer neural network models to perform classification and each of this neural network models will responsible on differentiating only two types of fingerprints and one unidentified fingerprint type which will be cluster to the 'others' class.

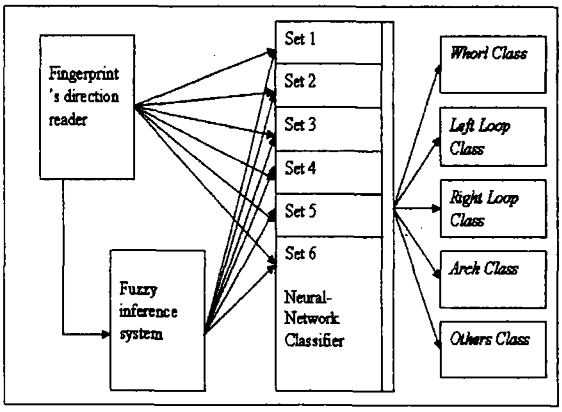


Figure 5. Neuro Fuzzy System

We need to have six multilayer neural networks because our system has five classes of fingerprint needed to be classified. The six neural network classifier are the classifier that only classify between Whorl and Right Loop (WR), Right and Arch (RA), Left Loop and Right Loop (LR), Left Loop and Arch (LA), Whorl and Left Loop (WL), and Whorl and Arch (WA). If the chosen neural network failed to classify the fingerprint image then it will classify that image to the *Others* class. Currently, we have successfully implemented this model; therefore we are able to give all the exact details on this model. Implementation and experiments on this model was written at the coming sections.

The structure of neural network and fuzzy logic in this fuzzy neural classifier is slightly different from the neural network classifier and the fuzzy logic classifier that we built to compare with this fuzzy neural classifier. The structure for the six neural networks are the same, instead of using the same structure like the neural network classifier which uses 256 input nodes correspond to the 256 features provided form the direction reader, one hidden layer with 10 nodes and 5 output nodes, this fuzzy neural uses 256 input nodes, one hidden layer with 5 nodes and 2 output nodes and backpropagation training algorithm was applied.

The differences between fuzzy logic classifier that we built and this fuzzy neural is that the number of membership functions in the fuzzy input sets and fuzzy output sets, instead of using 5 membership functions in the fuzzy input sets it uses only 2 membership functions in the fuzzy neural input sets and instead of using 6 membership functions in the fuzzy output sets it uses 7 membership functions in the fuzzy neural output sets.

The differences between our fuzzy-neural classifier with Prabhakar's [12] classifier is that our fuzzy-neural uses 256 features as input while his classifier uses 192 features as input. The inputs to these two systems are also different, we uses fingerprint's ridge and valley orientation while Prabhakar's [12] classifier uses feature that generated by Gabor filter. Instead of using 10 sets of neural network classifiers, our fuzzy-neural classifier uses only 6 sets because every classifier responsible to classify two type of fingerprints and for all unidentified type of fingerprints to a specific class.

4. Experimental Results

There are three different clustering approaches that were taken in our work. Same set of trained data were used. The trained data consists of 120 images with +10 and -10 degree of rotations. There are total of 5 comparisons being made, such as Accuracy of Classification for Trained Fingerprint Images, Accuracy of Classification for Untrained Fingerprint Images, Efficiency of Neural Network, Fuzzy Logic and Fuzzy Neural. The 120 images consist of 4 types of fingerprint where each type has 30 images (30 whorls, 30 arches, 30 left loops, and 30 right loops). After the process of training, image of the same sets of the training images is again feed into these classifiers to perform validation. If the submitted fingerprint is classified correctly then we will consider it as TRUE while if the classifier classified it wrongly then we will consider it as FALSE.

Figure 6 and 7 shows the accuracy results. Neural networks accuracy rate is the best. It is able to achieve 100% accuracy, while fuzzy neural have the worst performance which manage to score 94.17% accuracy. The error of fuzzy neural are concentrated on the classifying Arch type fingerprints, while fuzzy logic have errors classifying whorl type and right loop cluster but not as serious as error in fuzzy neural, therefore fuzzy logic achieved 95.83% accuracy.

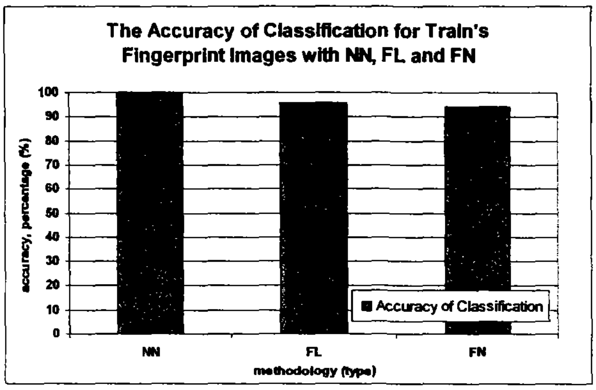


Figure 6. The accuracy of classification for trained fingerprint images

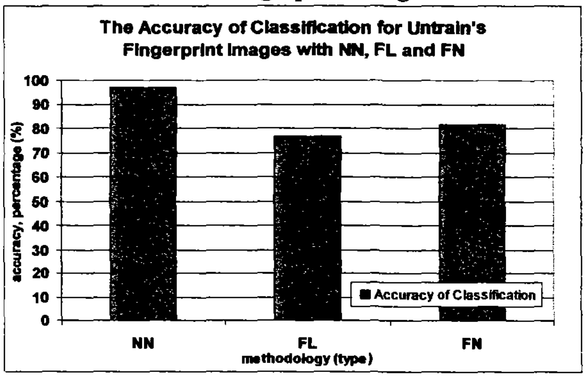


Figure 7. The accuracy of classification for untrained fingerprint images

The accuracy test for untrained fingerprint images uses a set of 260 images with +10 and -10 degree of rotations, these 260 images never been exposed to the three classifiers. These untrained images also like the 120 images set, it consist of 4 types of fingerprint where each type has 65 images. Then all these fingerprints will be submitted and classified by each classifier, if the classifier correctly then we will consider it as TRUE while if the classifier classified it wrongly then we will consider it as FALSE.

The test of efficiency is being performed after we embedded the classifier into the fingerprint recognition system. This test uses 150 fingerprint images; these images are being registered into the system by 10 by 10 basics and for every 10 images being register, the matching process will be performed to get the matching time (duration of identifying a fingerprint images with different amount of fingerprints data in the database). This is also the same test that we had performed on the previous fingerprint recognition system without classification.

The results of this experiment on the three classifiers are almost the same, the identification and reject imposter time for fingerprint recognition are successfully reduce more then 5 times, the old fingerprint recognition system from 80 seconds to approximately 12 seconds and from 140 seconds to approximately 25 seconds to identified a fingerprint that does not exist in the database which registered with 150 fingerprint images.

5. Conclusion and Future Work

Although neural network classifier perform better then fuzzy neural and fuzzy logic, but we cannot conclude that neural network classifier is the best. Other aspects that need to be considered are such as the suitability of the methodology in a given problem domain, the choice of feature representation and the degree of the feature representation discrimination.

Experiments showed that neural network classifier is the best classifier followed by fuzzy logic and fuzzy neural. Neural network classifier have good training algorithm. The limitation of direction reader deteriorates fuzzy logic and fuzzy neural performance but not in neural network. Fuzzy logic performance also deteriorates because of incompleteness and inaccurate representative of productions rule. Fingerprint image enhancement is also very crucial to produce better results

Problems with neural network classifier are that it performs classifying task in a black box manner, it is difficult to predict its behavior and to be enhanced later. In fuzzy logic input and output values are difficult to be defined. The performance of neuro fuzzy classifier is very dependent on the structure of the model. Different combination can produce different sets of results. Therefore, different model of neuro fuzzy classifier can be developed to see which can produce better results.

After several investigations, we believe a better neuro fuzzy classifier can be developed by rearranging the structure of fuzzy neural and enhance the direction reader to have the ability to distinguish more directions

References

[1] Nalini K. Ratha, Andrew Senior and Ruud M. Bolle. Automated Biometrics. IBM Thomas J. Watson Research Center, www.informatik.uni-trier.de/~ley/db/indices/a-tree/b/Bolle:Ruud_M=.html, 2001.

[2] EDN Access The Design Source for Engineers and Managers Worldwide – www.EDNAccess.com, 2003.

[3] Asker M.Bazen, Gerben T.B Verwaaijen, Sabih H.Gerez, Leo P.J. Veelenturf and Berend Jan van der Zwaag. A Correlation-Based Fingerprint Verification System. University of Twente, Department of Electrical Engineering, www.sas.el.utwente.nl/publications/download/71.pdf, 2002.

[4] Charles L. Wilson, James L. Blue and Omid M. Omidvar, "Improving Neural Network Performance for Character and Fingerprint Classification by Altering Network Dynamics", World Congress on Neural Networks Proceeding, II, Washington DC, pp. 151-158, July, 20 1995.

[5] David H. Kemsley, Tony R. Martinez, Douglas M. Campbell, "A Survey of Neural Network Research and Fielded Applications", In International Journal of Neural Networks: Research and Applications, vol. 2, No. 2/3/4, pp.123-133, 1992.

[6] C.L Wilson, G.T. Candela and C. I. Watson, "Neural Network Fingerprint Classification", J. Artificial Neural Networks,1, No.2, 1993;

[7] Faramarz Valafar, "Neural Network Applications in Biological Sequencing", San Diego State University, California, USA.

[8] Pavel Berkhin, "Survey of Clustering Data Mining Techniques", Accrue Software Inc.

[9] W.K. Cheng, B.Y. Ooi, "Fingerprints Recognition System", Undergraduate Research Training Report, School of Computer Science, Universiti Sains Malaysia, 2003.

[10] Arun D. Kulkarni, Computer Vision and Fuzzy-Neural Systems, Prentice Hall, New Jersey, 2001.

[11] "A brief Tour in the World of Fingerprints", http://www.xs4all.nl/~dacty/schedule.htm, Fingerprints TK, August 2003.

[12] Salil Prabhakar, "Fingerprint Classification", Fingerprint Classification and Matching Using a Filterbank, Michigan State University, pp. 119-150, 2001.