UNIVERSITÉ PARIS 13 - INSTITUT GALILÉE
Laboratoire d'Informatique de Paris Nord (L.I.P.N.)

THESE

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ PARIS 13

Discipline: **INFORMATIQUE**

présentée et soutenue publiquement

par

Vladimir RADEVSKI

21 Juin 2000

Titre:

Fusion de Caractéristiques et de Décisions dans les Systèmes d'Apprentissage Connexionnistes

Directeur de thèse:
Younès BENNANI

JURY

M^{me} Jacqueline VAUZEILLES,	Examinateur
M^{me} Bernadette DORIZZI,	Rapporteur
M. Srdjan STANKOVIC,	Rapporteur
M. Fouad BADRAN,	Examinateur
M.Younès BENNANI,	Examinateur
M. Dusan CAKMAKOV,	Examinateur
M^{me} Jacqueline CASTAING,	Examinateur

Université Paris 13 - Institut Galilée - Avenue Jean-Baptiste Clément 93430 Villetaneuse (France)

Fusion de Caractéristiques et de Décisions dans les Systèmes d'Apprentissage Connexionnistes

Pattern Features and Decision Fusion in Neural Network Based Pattern Recognition

Vladimir RADEVSKI

B.Sc. University St. Cyril and Methodius, Skopje, Republic of Macedonia
M.Sc. University of Belgrade, F.R. of Yugoslavia
Ph.D. candidate University Paris 13, France

Abstract

The subject of this thesis is the aspect of feature extraction and selection techniques in the pattern recognition systems based on a simple or multiple neural network classifiers.

The problem of the definition and implementing the feature extraction and selection phases as the main pre-processing phases in patter recognition system (PR) is emphasized. A review of the existing techniques is given for both of them, and an original classification of different possible approaches is given. The two main problems to be treated for the PR systems are the parameterization (effective reliable information extraction and selection) and the discrimination (classification and recognition issues). An original approach is proposed for the case of structural features definition and extraction, and an effective data fusion is shown for the case of two sets of features of different nature.

We propose a set of modular architectures for classifier combination in the pattern recognition environment. The classifiers combination is studied for the case of two neural network based classifiers performing on the two different representations of the same input samples, and the linear, non-linear and rule based techniques are examined. On the decision fusion level, a modular multi-level architecture is proposed based on the rule-based reasoning and classical combining techniques.

The validation of the proposed approaches and techniques is performed on the set of hand-printed Cyrillic alphabet characters, and the handwritten digits from the international NIST data base.

key-words:

pattern recognition, handwritten character recognition, pattern features, feature extraction, feature selection, structural features, data fusion, decision fusion, committee classifiers, combining classifiers, neural networks.

Résumé

Cette thèse est consacrée, d'une part, à l'étude d'approches d'extraction de caractéristiques structurelles et statistiques et leur fusion dans les systèmes d'apprentissage connexionnistes, et d'autre part, à la combinaison de classificateurs et la fusion de leurs décisions. L'étude se focalise essentiellement sur deux problèmes de la reconnaissance des formes (RdF): la parametrisation qui a pour objectif d'extraire et de sélectionner l'information pertinente de la forme dans le but d'en fournir une description aussi compacte et informative que possible, et la discrimination qui consiste essentiellement à classer des formes en vue de la reconnaissance.

Pour ces deux directions de recherche nous proposons un ensemble de contributions qui commence par une classification des techniques de sélection et d'extraction de caractéristiques en RdF. Nous proposons une nouvelle méthode d'extraction de caractéristiques structurelles ainsi qu'une méthodologie connexionniste de fusion de ces caractéristiques avec d'autres caractéristiques de nature statistique.

Ensuite, pour améliorer les performances en discrimination d'un système de reconnaissance, nous proposons quelques approches modulaires basées sur la combinaison d'un ensemble de classificateurs. Cette combinaison s'établit grâce à des techniques de fusion de décisions linéaires, non-linéaires ou à base de règles. Nous nous intéressons plus particulièrement dans cette thèse au Perceptron Multicouche qui constitue le modèle connexionniste le plus populaire et le plus utilisé du fait de sa simplicité et de son efficacité. Tout le long de la thèse, nous nous appuyons sur la reconnaissance de caractères (cyrilliques) et de chiffres manuscrits, comme domaine d'application et de validation de nos propositions.

Mots-Cles :

reconnaissance des formes, reconnaissance de caractères manuscrits, caractéristiques de formes, extraction de caractéristiques, sélection de caractéristiques, caractéristiques structurelles, fusion de données, fusion de décisions, réseaux connexionnistes, classificateurs, combinaison de classificateurs.

List of Figures

Page

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Chapter 6

List of Tables

Page

Chapter 2

Chapter 3

Chapter 5

Chapter 6

Table of Contents

Page

Avant propos

Contexte et but de la thèse

La Reconnaissance des Formes (RdF) représente un défi majeur dans le domaine de l'intelligence artificielle. Nombreuses sont les difficultés à résoudre avant de pouvoir mettre au point des systèmes capables d'égaler les performances humaines. Les applications dans ce domaine sont nombreuses dans la communication homme-machine, écrite ou parlée, en robotique, en diagnostic ou en télédétection.

La RdF a pour but de percevoir, d'interpréter et de reconnaître automatiquement les formes présentes dans des signaux. Ces signaux sont captés dans le monde physique par des instruments spécifiques (caméra, microphone, tablette à numériser, ...). Les instruments jouent un rôle très important pour la suite de la chaîne de traitement RdF. Ils doivent " dialoguer " avec les différents éléments de la chaîne de traitement. Des études montrent que la chauve-souris, volant au ras des herbes, modifie en une fraction de seconde la forme de son oreille pour l'adapter aux cibles ultra-sonores intéressantes, en particulier à ses proies, ceci confirme bien l'utilité de l'interaction capteur-prétraitement-reconnaissance. La RdF se doit alors résoudre des problèmes liés au codage des formes, à leur paramétrisation et à leur discrimination. Dans cette thèse nous nous intéressons essentiellement aux deux derniers problèmes. La paramètrisation qui a pour objectif d'extraire et de sélectionner l'information pertinente de la forme dans le but d'en fournir une description aussi compacte et informative que possible. La discrimination qui consiste essentiellement à classer des formes en vue de la reconnaissance. Généralement le problème d'extraction/sélection de caractéristiques et le problème de classement (discrimination) sont traités l'un après l'autre dans un système traditionnel de RdF. On peut s'interroger sur l'optimalité, en terme de minimisation de l'erreur de classement, des caractéristiques dérivées d'un autre critère. En d'autres termes, les méthodes traditionnelles ne garantissent pas l'optimalité du système global de la reconnaissance. Dans ce cadre, les systèmes d'apprentissage connexionniste sont très attrayants car ils sont capables de traiter les deux problèmes en parallèle : les caractéristiques sélectionnées optimisent le critère d'apprentissage en classement. Pour ces deux directions de recherche nous avons proposé un ensemble de contributions qui commence par une classification des techniques de sélection et d'extraction de caractéristiques en RdF (cf. chapitre 3). Nous avons proposé une nouvelle méthode d'extraction de caractéristiques structurelles (cf. chapitre 4) et une méthodologie connexionniste de fusion de ces caractéristiques avec d'autres caractéristiques de nature statistique (cf. chapitre 5). Ensuite, pour améliorer les performances en discrimination du système de reconnaissance, nous avons proposé quelques approches modulaires basées sur la coopération d'un ensemble de classificateurs. Cette coopération s'établit grâce à des techniques de fusion de décisions linéaires, non-linéaires ou à base de règles (cf. chapitre 6). Nous nous sommes intéressés plus particulièrement dans cette thèse au Perceptron Multicouche qui constitue le modèle connexionniste le plus populaire et le plus utilisé du fait de sa simplicité et de son efficacité. Tout le long de la thèse, nous nous sommes appuyés sur la reconnaissance de caractères (cyrilliques) et de chiffres manuscrits, comme domaine d'application et de validation de nos propositions.

Rapide parcours de la thèse

La thèse est structurée autour de sept chapitres.

Le premier chapitre sert d'introduction et présente le contexte de l'étude ainsi que les objectifs de la thèse, et se termine par le plan du manuscrit.

Le **chapitre 2** concerne la " **Reconnaissance des Formes et les Traits** ". Il sert d'introduction au domaine de la reconnaissance des formes en insistant sur la définition du terme " traits " ou " caractéristiques ". Il fournit un résumé des principales méthodes utilisées dans le domaine et donne un schéma général de conception d'un système de reconnaissance des formes. Les différentes étapes de la conception d'un système de reconnaissance y sont développées, avec un accent particulier sur l'importance de la phase des pré-traitements. Une présentation et une analyse des modèles d'apprentissage connexionniste ainsi que la détermination de leur place dans les systèmes de reconnaissance des formes sont décrites et soutenues par une série d'exemples d'implémentations publiées. Cette présentation débouche sur les particularités de la reconnaissance des caractères manuscrits. Le chapitre s'achève avec une présentation de la problématique de la représentation des caractéristiques des formes. Leur place et leur importance dans un système de reconnaissance y sont mises en évidence. De nombreux exemples illustrent la diversité des idées et des choix concernant la caractérisation des formes.

Le chapitre 3 s'intitule " **Extraction et Sélection de Caractéristiques** ". Il présente le problème de l'extraction et la sélection des caractéristiques d'une forme. Ces deux termes ont été souvent ambigus et confus dans la littérature. Nous donnons, dans ce chapitre, une définition plus claire permettant de faire une nette distinction entre ces deux termes. L'objectif principal de l'extraction et de la sélection des caractéristiques est de réduire d'une façon pertinente la dimension des formes, nous établissons alors une typologie des principales méthodes de réduction de dimension. Nous décrivons ensuite la phase de transformation de caractéristiques (i.e.

l'extraction et la découverte des caractéristiques), s'en suit une présentation des techniques de sélection des caractéristiques. Celles-ci se divisent en deux grandes classes: " classificateur-dépendante " et " classificateur-indépendante " avec une attention toute particulière aux procédures de recherche de sous-ensembles de caractéristiques. Nous proposons alors une stratégie de parcours de l'ensemble des caractéristiques et un critère de sélection des caractéristiques les plus pertinentes. Le chapitre conclut par un ensemble de résultats issus d'études expérimentales.

Le chapitre 4, " Caractéristiques Structurelles pour la Reconnaissance de Caractères manuscrits Cyrilliques ", propose une nouvelle technique d'extraction de caractéristiques de nature structurelle, avec une application à la reconnaissance de caractères cyrilliques manuscrits. La description détaillée de cette technique d'extraction de caractéristiques est donnée et un critère original de comparaison de lignes est proposé pour la définition de l'ensemble des caractéristiques structurelles. On obtient la présentation des caractéristiques des lettres entrées et un groupement de celles-ci est donné. Les principes d'un système de reconnaissance hiérarchique sont établis à travers l'estimation de l'erreur bayesienne à tous les niveaux du groupement. La structure obtenue permet de découvrir des règles de reconnaissance multi-niveaux parcourant l'arbre hiérarchique, et engendrant des décisions, d'une complexité variable par niveaux.

Le chapitre 5 traite le problème de la " **Fusion de Données Structurelles et Statistiques dans un Système Connexionniste de Reconnaissance** ". L'étude se situe dans le cas de deux ensembles de caractéristiques de nature distincte (structurelle et statistique) caractérisant des chiffres manuscrits. Ces chiffres sont manuscrits et issus de la base internationale NIST. La fusion est opérée à travers un classificateur basé sur un système d'apprentissage connexionniste. Après une description de l'architecture générale du système, les deux ensembles de caractéristiques structurelles et statistiques sont définis. La fusion des données est effectuée par un classificateur connexionniste lors de la phase d'apprentissage. Une phase de sélection de caractéristiques est ensuite mise en œuvre. La présentation des résultats expérimentaux est accompagnée d'une comparaison de notre approche avec des systèmes de référence dans le domaine. Cette comparaison met plus l'accent sur l'aspect complexité sans négliger les performances des systèmes de reconnaissance.

Le chapitre 6 traite le problème de la " **Fusion de Décisions** ". Dans une première partie, les critères de combinaison des classificateurs sont décrits, ainsi que quelques résultats d'implémentation de ces techniques. Une étude comparative et une analyse des performances, de deux classificateurs proposés, donne une idée sur l'apport et l'intérêt de ces techniques de fusion.

Dans une deuxième partie, nous proposons une fusion plus sophistiquée. Dans une première étape, après avoir défini les stratégies de fusion de décisions, les classificateurs individuels sont examinés au niveau d'informations exploitables pour la fusion. Ensuite, est traité le problème de l'amélioration de la **fiabilité** en introduisant un critère de rejet. Les ressources sur lesquelles la fusion de décision agira étant définies, nous proposerons une fusion de décisions en deux étapes basée sur des règles de raisonnement. Les résultats expérimentaux montrent les avantages et les intérêts de cette nouvelle approche de fusion de décisions.

Le chapitre 7 sert de " **Conclusion et Perspectives** " de la thèse. Il contient une discussion finale sur l'apport de la fusion de caractéristiques et de la fusion de décisions, aux systèmes de reconnaissance. On y conclut que notre nouvelle technique d'extraction de caractéristiques structurelles et statistiques permet d'exhiber des traits pertinents pour caractériser les formes. La combinaison de plusieurs types de caractéristiques de différentes natures (différents capteurs) représente un plus pour le problème de la reconnaissance de caractères manuscrits. Notre approche de fusion de décisions, à base de règles, confirme le fait que la fusion de décisions de plusieurs classificateurs augmente les performances et la robustesse des systèmes de reconnaissance. Ces deux types de fusions (caractéristiques et décisions) permettent de concevoir des systèmes compacts et de performance accrue décisions) permettent de concevoir des systèmes compacts et de performance accrue tout en gardant une complexité moindre.

Comme perspectives, nous continuons à explorer les deux axes de recherche étudiés dans cette thèse: l'extraction et la sélection de caractéristiques et la fusion de données et de décisions.

Concernant le première axe, nous pensons incorporer des connaissances à priori du domaine lors de l'extraction et la sélection de caractéristiques. En effet, cette introduction de connaissances permettra d'enrichir les caractéristiques sélectionnées et d'augmenter leur pertinence. D'autres mesures de similarité de traits seront étudiées et adaptées à notre problème.

Nous pensons orienter notre recherche vers d'autres types de données, comme les données dynamiques et les données évolutives. Pour ce type de données nous pensons que l'extension des approches proposées dans cette thèse peut être très fructueuses.

Chapter 1

Introduction

1.1 The Problem and the Approach

The subject of the thesis is concentrated around three main problems of multiple classifiers environment pattern recognition (PR). Firstly, the problem of the definition of feature extraction and selection phases is stressed, and a consistent determination of the possible approaches to both of them is given. A set of structural features is proposed for handwritten digit description, and is tested on the digits from the NIST data base. Secondly, the aspects of data fusion are discussed on the example of the fusion of structural and statistical features for handwritten digit recognition. The third problem treated is the problem of effective decision fusion for a high reliability multistage recognition system.

Pattern recognition is a major challenge in the domain of artificial intelligence. The complexity of the recognition of the humans is still too unknown to be implemented in the world of machines. Yet the possible applications of this domain are increasing in communication man-machine, written or spoken, in roboics, diagnostics and tele-detection.

The PR aim is to consider, interpret and automatically recognize patterns in signals. The signals can be captured from the physical world by specific instruments (scanners, cameras, microphones, etc.) and the PR techniques should tackle the problem of pattern coding, parameterization and discrimination. This thesis traits the problem of the pattern parameterization and discrimination. The parameterization having the aim of the extraction and the selection of the reliable pattern information for best possible compact and informative description, and the discrimination being essential for the classification in perspective of reliable recognition.

The problem of combining different kind of features as a pattern descriptors, and the use of the neural networks as a powerful classification tool in PR have been the starting points for the investigations undertaken in the work presented in this thesis. Among the raisons of the popularity of the neural networks in the PR research area, we emphasize their ability to tackle two problems in parallel: feature selection and optimization of the classification.

For these two research directions we have proposed some approaches in regard to the study of the techniques of the feature extraction and selection in PR (Chapter 3.), an original method of structural features definition and extraction (Chapter 4.), the cooperation of the features of different nature (structural and statistical) through effective data fusion (Chapter 5.) and a neural network based multistage decision fusion in cooperation with rule-based reasoning.

The validation of the propositions is performed on the data bases of Cyrillic hand-printed characters and handwritten digits extracted form the international NIST data base.

1.2 Guide to the Dissertation

The dissertation has seven chapters. After the introduction given in the first chapter, **Chapter 2**. tackles the problem of pattern recognition and features. The definition of the scientific are of pattern recognition is given, some aspects of the pattern recognition performed in the nature are emphasized. The problems of implementing the observed characteristics in the nature are located, and the basis of the phases of learning and classification are stated. Thus, a general scheme for constructing a pattern recognition is given, and the importance of the features is emphasized. the first part of the Chapter 2. gives the relations between the areas of pattern recognition and classification and pattern recognition and artificial neural networks. The phases of learning and recognition in a neural network environment are presented in regard to the application in the pattern recognition area. The first part of Chapter 2. ends with a brief presentation of the area of optical character recognition. The second part of Chapter 2. deals with features, their definition, and the intrinsic characteristics of the term. This Chapter two ends with a state of the art presentation of the main ideas and examples of the implementations of the features in pattern recognition applications.

Chapter 3. is about the feature extraction and selection. After remarking a strong ambiguity in the use of those terms and the mixture of the phases of extraction and selection in the literature, a definition of the phases is proposed. The recent works in the area, and the appearance of the IEEE Special issue on feature transformation, extraction and selection (1998), are showing the actuality of the problem, and the main ideas proposed to make a distinction and a definition of those phases are presented. The first part of Chapter 3. ends with a list of a dimensionality reduction methods list. The second part of the Chapter 3. is about the feature transformation phase, i.e. the feature discovery and extraction, to be followed with a part on a feature selection techniques. The feature selection techniques are divided in two groups: classifier dependent and classifier independent, and a special attention is paid to the feature subset searching procedures. In the last part of Chapter 3. a strategy for feature selection is proposed and the experimental results are shown.

The **Chapter 4.** presents a real world character recognition application based on the two sets of features of different nature. The implementation is tested on a Cyrillic hand-printed characters base. The detailed description of the phase of feature acquisition is given for the both sets of features; structural and statistical. An original similarity criterion for comparing the line primitives is proposed, and implemented in the definition of the set of structural features. The feature presentation of the input letters is obtained, and a clustering of so presented letters is given. The basis of multi-level recognition system are established, through estimating the Bayes error on all levels of the hierarchical clustering of the samples base. The obtained structure permits establishing rule-based multilevel recognition going through the passing the hierarchical dendogram and implementing the more or less sophisticated decision rules by levels and corresponding to the estimated Bayes error by levels.

Chapter 5. tackle the problem of data fusion, where data are the two sets of features of different nature for presenting the handwritten digits. The experiments are done on the NIST segmented handwritten digits, and the fusion is performed through a neural network based classifier. After giving the general architecture of the system, and a description of the data base, the two sets of structural and statistical features are defined. The phases of feature acquisition and extraction are described in details for both set of features. The data fusion is performed by neural network based classifiers, and the phase of feature selection being classifier dependent is performed and the results are shown. Last part of Chapter 5 situate the proposed system among the systems published previously, and compares in terms of complexity and performances.

The **Chapter 6.** is about the decision fusion problem. In the first part of Chapter 6 the classifier combing criteria are given, and the results are shown after the implementation of the main combining techniques known in the literature. The comparative study of the performances gives an idea of adaptability of the known combing criteria for the problem of fusion of the two proposed neural network based classifiers each of which is acting on the separate set of the previously defined features. In the second part of the Chapter 6 a improved fusion approach is proposed. Firstly, the decision fusion strategies are defined and the member classifiers are examined in the terms of the level of information that they can provide to the fusion instance. Further on we tackle the problem of the reliability improving introducing the rejection criteria. After defining sources on which to be defined the decision fusion, in the last part we propose a rule-based decision fusion on two stages. The results are given and the advantages of the approach are emphasized.

In the **chapter 7.** a conclusion of the thesis is given. A brief summary of the results is followed by the possible perspectives for the future research.

The thesis has a 46 figures and 28 tables and a list of 180 bibliography units.

Chapter 2

Pattern Recognition and Features

A wide variety of work has been done under the label of “pattern recognition”. In this chapter we will try to cover some aspects of the pattern recognition research field relative to the implemented approaches and the techniques that were developed for the purposes of this thesis.

It seems that nature decreed the information in pattern format, and humans (and not only the humans) adapted well to that circumstance. Part of the motivation for the study of pattern recognition stems from the desire to understand the basis of pattern recognition powers in humans. In general, patterns are *the means* by which we interpret the world.

2.1 Pattern Recognition

Pattern recognition is a scientific discipline whose goal is the *classification* of *objects* into a number of categories or *classes*.

As defined in [Merriam-Webster, 2000] a "pattern" is: a form or model proposed for imitation; exemplar; or a discernable coherent system based on the intended interrelationship of component parts. The generic term *pattern* we will be used here to refer to the objects to be classified into a number of categories, *classes*. In general, these objects can be images, or signal waveforms or any type of measurements that need to be classified.

The two main aspects of the *pattern recognition* are the development of a decision rule and using it. The *recognition* is the term which we use to describe the use of this rule, and the *pattern* is something that is defined in the learning process by the labeled samples.

There is a strong engineering based pragmatic motivation for the research in this field. As the society evolves into its postindustrial phase, automation and the need for handling and retrieval of the informationare becoming increasingly important. Some of the principal application area, and the area where the research has given the most important results include: character recognition, fingerprint classification, general photo recognition, natural resources identification on the basis of satellite images, cell tissue analysis etc. In general, any system which can be considered as a "black box" described by input-output data, where the output is a class label and exemplar data are available, is a candidate for the pattern recognition algorithms. Many of the concepts and techniques which were proposed in this particular field, are used in more general data analysis applications.

The main constraints which could appear and may dissolve an apparent applications are: a) too few labeled samples, b) the lack of a distinguishable pattern in the data available or c) the availability of a better way to solve the problem.

We can think of pattern recognition in terms of mapping a pattern correctly from pattern space into class-membership space [Zadeh, 1977]. In nature, this process is carried out in terms of opaque mapping. The opaque nature of this mapping lies in the fact that although being able to recognize, we can not describe adequately the class of "A"s, or the class of "apples". The task of implementing computer-based pattern recognition is to replace the opaque mapping with a transparent mapping that we can describe precisely to a computer [Pao, 1989]. It is worth of making the distinction here between the *classification* and *pattern recognition*. Namely, the problem of *classification* is to find a way to decide whether two given objects are equivalent (given the relation of equivalence), while the problem of *recognition* is to find a way to decide whether a given object is equivalent to a prototype (canonical form, pattern) in a given collection of structures.

A pattern can be presented as a pair:

$$\text{pattern} = (s, c)$$

where s is *a sensory information*, *collection of observations* or more technically - *measurements* and c is *a concept*, *a class*, *meaning* or *a name* which corresponds to the sensory information.

From the technical point of view, simulation of the living being's behavior can be made by the following order of activities:

$$\text{measurements} \rightarrow \text{classifier} \rightarrow \text{object (concept) class}$$

The measurements can be seen as a vector $\mathbf{v}$ of input information for the classifier. Using these information, the classifier should put considered thing in one of the classes, where the number of classes-k can be given, or not (*classification*) or, give a meaning to corresponding measurements (*recognition*). Thus, the process of pattern recognition is a mapping V → Ω, where V is a space of vector measurements and Ω is a set of classes or more generally, mapping from the measurements (observations) to the meanings.

The representation of the differences between the artificial pattern recognition system and the pattern recognition realized in nature and are presented in Fig. 2.1.

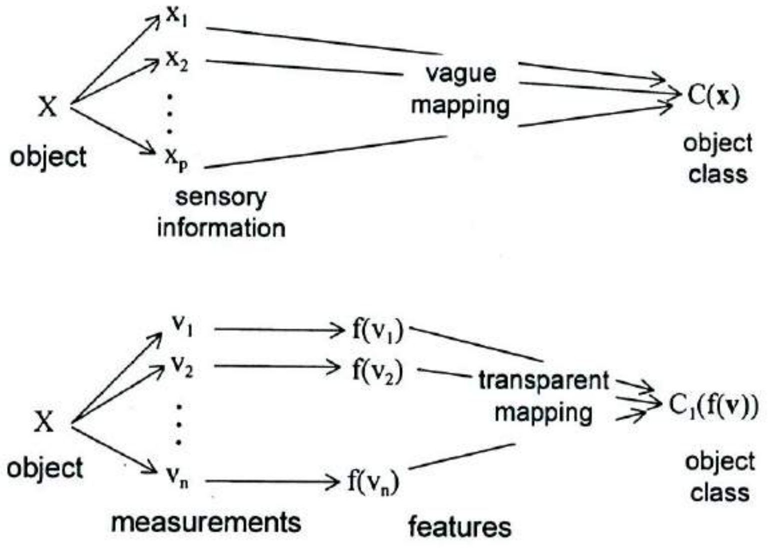


Figure 2.1 *The process of pattern recognition in nature and computer designed*

For the computer-implemented pattern recognition we need to be concerned not only about the transparent mapping $C_1(.)$, but also about the appropriate choice of the function f(.). The choice of features for the description of objects is a difficult but essential preprocessing task in the implementation of computer-based pattern recognition.

In general, although in restraint form, the objectives of Pattern recognition as a classification model are:

1. Performing *feature extraction*; deciding how the manifestation **x** of the object X should be described symbolically in the form of f(**x**);
2. Learning the transparent mapping $C_1(.)$; that is, using a set of labeled training-set patterns to infer decision rules;
3. Exercising the mapping $C_1(f(\mathbf{x}))$ from the representation space to the interpretation space to carry out the actual classification act.

The general scheme of the phases in a pattern recognition system development is given in Fig. 2.2.

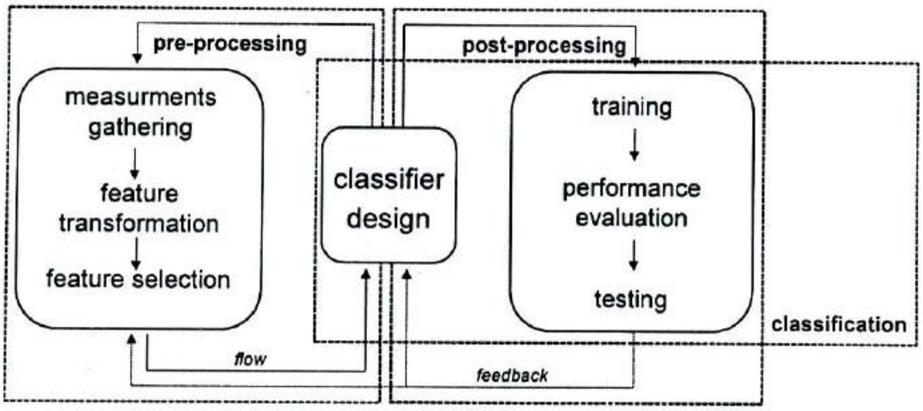


Figure 2.2 *The phases in the developing a pattern recognition system*

The term *measurement* means low-level data (row data, collection of observations) describing the system. A sample of pattern is represented by specific values of all measurements, corresponding to a point in the measurement space.

The term *feature* is used for higher level data prepared for classification. Thus, features are the threads connecting the three basic elements of the general model of pattern recognition: observing element, learning element and performance element. The pattern space is rarely identical (although it can be) with the measurement space. Usually, a several stages of intermediate processing may be necessary. The future extraction (the feature transformation, as it will be explained in Chapter 3.) is the processes by which a sample in the measurement space is described by a finite and usually smaller set of numbers called features.

Generally, we can divide the phases into three groups: *pre-processing, classifier design* and *post-processing*. In the first group, we can include the phases: measurements gathering, feature transformation, and feature selection, and in the third group: classifier training, performance evaluation (often a validation process) and testing. The classifier design is presented as a phase between these two groups. The post-processing phases together with classifier design we can simply call *classification*. We recall the difference we made between the classification and recognition; when the equivalency between objects is defined along with a label-named classes we consider it as a recognition problem, otherwise a classification problem. If we do not precise otherwise, by classification, here on, we will mean also the specific classification case where the classes are label-named, which is in fact a recognition.

In this thesis the pre-processing phase addresses the work on the handwritten digits from the International NIST data base [NIST, 1992] and on a basis of handprinted capitals of the Cyrillic alphabet. The classification phases are performed on the Neural Networks based classifiers as well as on designed rule based meta-classifications including Committee classifiers performing and reasoning.

In most practical applications the pre-processing is the most important factor in determining the performance of the final system. If we perform good pre-processing the classification phases could become trivial. On the other hand, good classification phases are not able to "correct" weak pre-processing. Thus, the pre-processing offers larger maneuver space in designing a pattern recognition system.

Unfortunately, although on first sight it seems that the classification phases are more difficult to be realized than the pre-processing ones, it is far from the truth. There are relatively well-developed theories that guide the development of classification techniques while that is not the case when searching for optimal measurements and features.

Therefore, in practice, we are forced to compromise in order to balance between more promising but more uncertain pre-processing and less promising but less uncertain classification.

The main areas of pattern recognition are the *machine vision*, *computer-aided diagnosis*, *speech recognition* and *character recognition* [Theodoridis and Koutroumbas, 1999]. A *machine vision* system captures images via a camera and analyzes them to produce descriptions of what is imaged. A typical application of a machine vision system is in the manufacturing industry for automated visual inspection or for automation in the assembly line. The systems have been proposed in radar image analysis, military target and object detection as well as in the applications of space, satellite and ocean data analysis. *Computer-aided diagnosis* has been applied to and it is of interest for a variety of medical data, such as X-rays, computed tomographic images, ultrasound images, electrocardiograms and electroencephalograms. The need for a computer-aided diagnosis stems from the fact that medical data are often not easily interpretable, and the interpretation can depend very much on the skill of the doctor. Speech recognition is another area in which a great deal of research and development effort has been invested. The goal of building intelligent machines that recognize spoken information has been a long-standing one for scientists and engineers. Others areas of pattern recognition include fingerprint identification, signature authentication text retrieval, and face and gesture recognition.

2.1.1 Pattern Recognition and Classification

Depending on the assumptions of the existence or non-existence of sets of training data we can distinguish a *supervised* or an *unsupervised* recognition and /or classification. A major issue of the later is that of defining the "similarity" between two feature vectors and choosing an appropriate measure for it. Another topic of this kind of recognition and / or classification is choosing an algorithmic scheme that will cluster the vectors on the basis of the adopted similarity measure.

For the case of *supervised* classification (recognition, or even learning) we will outline the main approaches and techniques. For a given classification task of M classes $\omega_1, \omega_2, \ldots \omega_M$, and an unknown pattern, which is represented by a feature vector $\boldsymbol{x}$, we form the M conditional probabilities $P(\omega_i \mid \boldsymbol{x})$, i=1,2,...$M$, we refer to them as *a posteriori probabilities*. Each of them represents the probability that the unknown pattern belongs to the respective class ω_i, given that the corresponding feature vector takes the value $\boldsymbol{x}$. There is a class of classifiers that compute either the maximum of these M values, or equivalently, the maximum of an appropriately defined function of them. The unknown pattern is assigned to the class corresponding to that maximum, and we refer to this class of classifiers as *Classifiers Based on Bayes Decision Theory* [Theodoridis, Koutroumbas, 1999].

A reasonable assumption to made is the one of the availability of $P(\omega_i)$, i=1,...,M, even if they are not known, they can easily be estimated from the available learning feature vectors. If N is the total number of available training patterns, and N_i of them belong to ω_i respectively, then $P(\omega_i) \approx N_i / N$. The class-conditional probability density functions $p(x|\omega_i)$, i=1,...,N describing the distribution of the feature vectors in each of the classes can also be estimated from the training data. In the case that feature vectors can take only discrete values, density functions $p(x|\omega_i)$, i=1,...,N become probabilities and will be denoted as $P(x|\omega_i)$, i=1,...,N. Now, to compute the conditional probabilities we use the Bayes rule:

$$P(\omega_i)|x) = \frac{p(x|\omega_i)P(\omega_i)}{p(x)}$$

where $p(x) = \sum_{i=1}^{M} p(x|\omega_i)P(\omega_i)$.

The *Bayes classification rule* acts according to the maximum of $P(\omega_i|x)$, or which is equal to the maximum of $p(x|\omega_i)$.

Bayesian classifier is optimal with respect to minimizing the classification error probability [Theodoridis, Koutroumbas, 1999]. Minimizing either the risk or the error probability is equivalent to partitioning the feature space into M regions, for a task with M classes. If we pass from probabilities (or risk functions) to an equivalent function of them, for example $g_i(x) \equiv f(P(\omega_i|x))$, where $f(.)$ is a monotonocally increasing function, and $g_i(x)$ will be refereed as a *discriminant function*. So, the decision procedure now follows:

classify x in ω_i if $g_i(x) > g_j(x)\ \forall\ j \neq i$

The decision surfaces, separating contiguous regions are described by

$g_{ij}(x) \equiv g_i(x) - g_j(x) = 0,\ i,j = 1,2,\ldots,M,\ i\neq j$

However, not all problems are well suited to such approach; in many cases the involved probability density functions are complicated and their estimations is not an easy task. Before stating some approaches of the classification where the discriminant functions and decision surfaces are with no (necessary) relation to Bayesian classification, which are, in general, suboptimal with respect to Bayesian classifiers, we state some decision surfaces associated with the Bayesian classification for the case of Gaussian density functions. In the case of the most commonly encountered probability density function in practice, the Gaussian, or normal density function, the Baysian classifier is quadratic classifier, in the sense that the partition of the feature space is performed via quadratic (or hyperquadratic) decision surfaces. Under the assumption of equiprobability of the classes with the same covariance matrix, the maximum $g_i(x)$ implies minimum of the Euclidean distance $d_e = \| x - \mu_i \|$, or in the case of nondiagonal Σ, minimizing the Σ^{-1} norm known as Mahalonobis distance

$$d_m = ((x - \mu_i))^T \Sigma^{-1} (x - \mu_i))^{1/2}.$$

Depending of the nature of the information which is not available (probability density functions or certain parameters, such as mean value or the variance) different approaches can be adopted for their estimation, like maximum likelihood parameter estimation, maximum a posteriori probability estimation, Bayseain inference, non-parametric estimation etc. Concerning the density estimation, the *k* Nearest Neighbor density estimation, in one variant can be seen as popular in practice nonlinear classifier. The basis of the *k* NN classification for an unknown feature vector x, and a given distance measure is:

1. Out of *N* training vectors, identify the k nearest neighbors, irrespective of the class label (with *k* odd);
2. Out of these *k* samples, identify the number of vectors, k_i, that belong to class ω_i, i=1,2,...,*M*.
3. Assign x to the class ω_i with the maximum number k_i of samples.

This makes of the *k* NN popular in practice, nonlinear classifier, specially having on mind the results of Devroye and colleagues [Devroye et al., 1996] stating that for a large enough number of training samples this simple rule exhibits good performance, and that as $N \rightarrow \infty$ the classification error probability for this approach is bounded by two times the optimal Bayesian error.

When the involved probability density functions are complicated and their estimation is not an easy task, it is preferable to compute the decision surfaces directly by means of alternative costs. In this group we will outline the *Linear* and *Non-linear classifiers*. Under an assumption that all feature vectors from the available classes can be classified correctly using a linear classifier, the respective decision hypersurface in the l-dimensional feature space is given by

$$g(x) = \boldsymbol{w}^T\mathbf{x} + w_0 = 0$$

where $\boldsymbol{w} = [\ w_1, w_2, \ldots, w_l]^T$ is the weight vector, and w_0 is the threshold, where $\boldsymbol{w}$ is orthogonal to the decision hyperplane, and the major concern for the definition of the decision hyperplane will be to compute the unknown parameters w_i, i=0,…, l.

The problem can be approached as a typical optimization task if we adopt an appropriate cost function (continuous and piecewise linear) and an algorithmic scheme to optimize it. The perceptron algorithm can be defined by a definition of an iterative scheme for the minimization of the cost function, in the spirit of the gradient descent method, with an assurance of the convergence of the algorithm to a solution in a finite number of steps [Theodoridis, Koutroumbas, 1999]. Once the perceptron algorithm has converged to a weight vector w, and a threshold w0, the classification of an unknown feature vector, for the case of two classes, to either of the two classes is achieved via the simple rule:

$$\text{if } \boldsymbol{w}^T\boldsymbol{x} + w_0 > 0 \text{ assign } \mathbf{x} \text{ to } \omega_1$$

$$\text{if } \boldsymbol{w}^T\boldsymbol{x} + w_0 < 0 \text{ assign } \mathbf{x} \text{ to } \omega_2$$

The basic perceptron model is shown in Fig. 2.3.

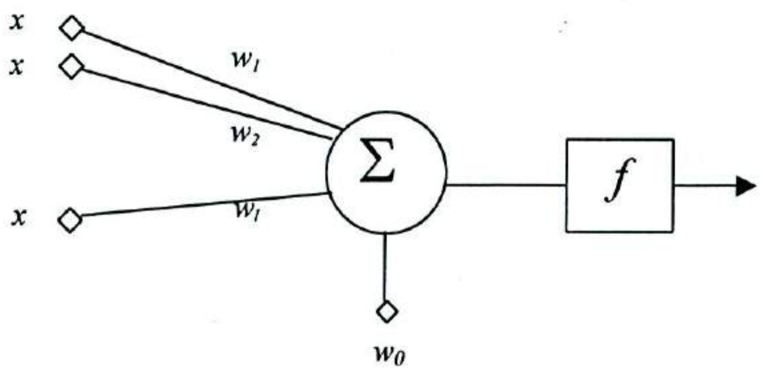


Figure 2.3 *The basic perceptron model*

For non-linearly separable classes linear classifiers are optimally designed, for example, by minimizing the squared error. For the problems which are not linearly separable, and for which the design of a linear classifier, even in an optimal way, does not lead to satisfactory performance a nonlinear classifiers must be considered. In he next section (2.1.3) we will discuss some aspects of multilayered neural networks, specially the multilayered perceptron. Another large class of nonlinear classifiers is the class of decision trees. They are multistage decision systems in which classes are sequentially rejected until we reach a finally accepted class.

2.1.2 Pattern Recognition and Artificial Neural Networks

A important place in our research is hold by the artificial Neural Networks (NN) as being tool in pattern recognition. Digital computers, with their speed and high accuracy, were easily overwhelmed by algorithmic tasks which are of exponential or greater complexity. Most of the real-world perception and cognition tasks when approached in a direct manner are of such nature. In contrast, the nature of biological systems is that of distributed parallel processing systems, made up of large numbers of interconnected elemental processors of rather slow processing speed [Pao, 1993]. In addition, information processing seems to depend on the ability to discern what is cogent and relevant, and to focus on that, while sustaining a minimal degree of maintenance on other matters.

In contrast to the systematic, frontal-attack approach, biological systems seem to rely more on experience and learning so that any good path or even a segment of a good path is remembered, and that knowledge is transmitted through generations, either genetically or through education. Of a central importance in the educational process are patterns, both spatial and temporal. This significance is established by associations between a pattern (or a set of patterns) and other patterns (or sets of patterns). So, the formation of such associations and the activation of such linkages are matters of critical importance and is in relation with the intrinsic objectives of the practical pattern recognition applications: to design and implement machine systems, which are able to perform perception tasks competently to degrees of proficiency comparable to that of biological systems.

So, specially in the pattern recognition research community, there is not only interest in basic matters in artificial neural net computing, but also an extensive activity in the application of this technology to practical tasks. Thus, regardless of whether the artificial neural net computing paradigm models biology or not, it is of intrinsic value to information processing researches and especially to pattern recognition researches, the researches who focus to "pattern-ness" of matters and in rapid distributed parallel processing of associated nets of such patterns.

Historically, the field of Pattern recognition started with the early efforts in neural networks (perceptrons, adalines and backpropagation) [Beale and Jackson, 1990, Haykin, 1994, Bishop, 1995]. Properly designed multilayered neural networks can learn complex mappings in high-dimensional spaces, without requiring complicated handcrafted feature extractors. This relies on the fact that gradient-based minimization techniques can be used to learn very complex nonlinear mappings [LeCun and Bengio, 1995].

We present the list given by Pao Y.S. [Pao, 1993] of a topic areas and typical architectures, algorithms and functionalities supported by the NN algorithms and corresponding activities and results in traditional pattern recognition research in Table 2.1.

Artificial NN computing area	Representation Algorithms	Functionality	Traditional Pattern recognition Issues
Unsupervised Learning	•ART 1 & 2 •LVQ •Self Organizing Maps	Disconserning regularities in data Classification	•Data reduction •Forming clusters •K-means •Classification
Supervised Learning	•Generalized delta rule / error back propagation •Functional link-net	Learning a functional mapping from a set of examples	Non-parametric estimation (usually limited to the estimation of density distribution functions)
Associative memory	•Hopfield net •ART 1, 2 or 3	•Restoration of corrupted patterns •Associative recall •Classification	Distributed matrix associative memories
Optimization	•Hopfield and Tank approach	•Optimal activation to complex problems gradient research	no direct correspondence
System Level Issues	•Concept formation	•Inductive learning •Feature extraction •Concept formation	•Feature extraction •Learning discriminant •Associative memory

Table 2.1 *Neural networks computing and Pattern Recognition*

The importance of the place of the area of neural networks to pattern recognition is emphasized in many works [Leon et al., Yan, 1995].

The pattern recognition task of learning a discriminant function for the purposes of classification corresponds to the Supervised learning of a functional mapping from observations space of examples to the space of classes. For the case of functional mapping of $R^n \rightarrow R$ we can schematize the work of a feedforward network as in Fig. 2.4.

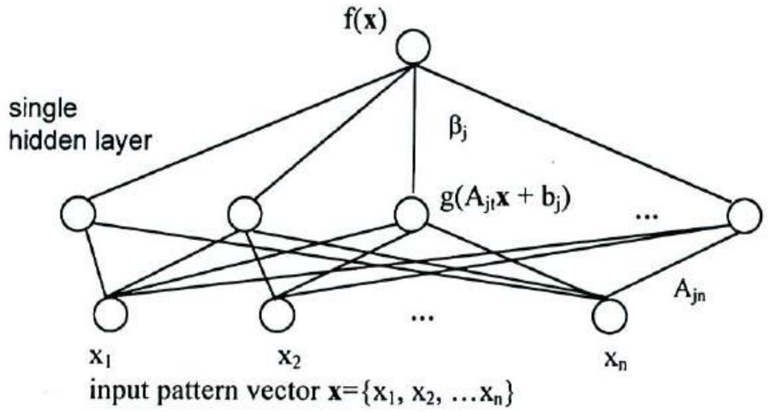


Figure 2.4 *A feedforward neural network shown for $R^n \rightarrow R$*

For an example purpose, we will consider a neural network of input vector from R^n and the single number as an output. It is assumed that there is a functional mapping $y= f(\mathbf{x})$, instances of which are known $\{y_p=f(x_p)\}$, and the learning task consists of determining the values of the weights $\{A_{ij}\}$ and $\{\beta_j\}$ and the thresholds $\{b_j\}$ so that the mean of the squares of the error

$$\sum_p(\hat{f}(x_p - f(x_p))^2$$

is minimized. There is no loss of generality in omitting a nonlinear transform at the single output node. In the general case there would be more than a single output and there could be more than one hidden layer. The weights and thresholds are determined on the basis of minimizing the overall system error averaged over all the training sets. That is, the quantity

$$\sum_k\sum_p(\hat{O}_k(x_p)-O_k(x_p))^2$$

is minimized, where $O_k(x_p)$ is the desired or target output at the k^{th} node for the p^{th} pattern, and $\hat{O}_k(x_p)$ is the actual computed value of the k^{th} output for the same pattern.

In the learning process, the weights β_j (or β_{kj} in the multi-output case) are readily learned because we have a direct measure of the error $\hat{O}_k(x_p)$- $O_k(x_p)$ at each and all outputs, for all the training patterns. For the hidden nodes, there is no direct measurement of the relevant error ascribable to a particular hidden node and so the output pattern error has to be propagated backwards and interpreted appropriately to serve as a measure of guidance for improving the values of the weights leading into hidden-layer node.

Although the overall learning procedure of the backpropagation of the error rhythm is that of a gradient search in the weight space, there are many variations on the adaptation on how to improve the rate of convergence to the point of least error.

It was stated by Pao, that the neural network computing might turn out to be an essential tool for unifying our pieces of knowledge in the fragmented bastions of research endeavor known nowadays as an artificial intelligence, pattern recognition, fuzzy logic, computer vision and so on [Pao, 1993].

An attempt of characterization of all types of patterns is the goal of the Pattern Theory, but the principles are very close to those of the neural networks [Jean and Goel, 1994].

The implementation of the neural networks based techniques in the field of pattern recognition is expanded largely out of the borders of their use as a simple classifiers and can appear in a wide diversity of phases or ideas on which a pattern recognition is based: from the use of the hidden layers of a neural network as a feature extractors [LeCun et al., 1992] up to an integration part of a hybrid connectionist and symbolic image recognition systems [Roli et al., 1995]. The integration paradigm of the later is shown in Fig. 2.5.

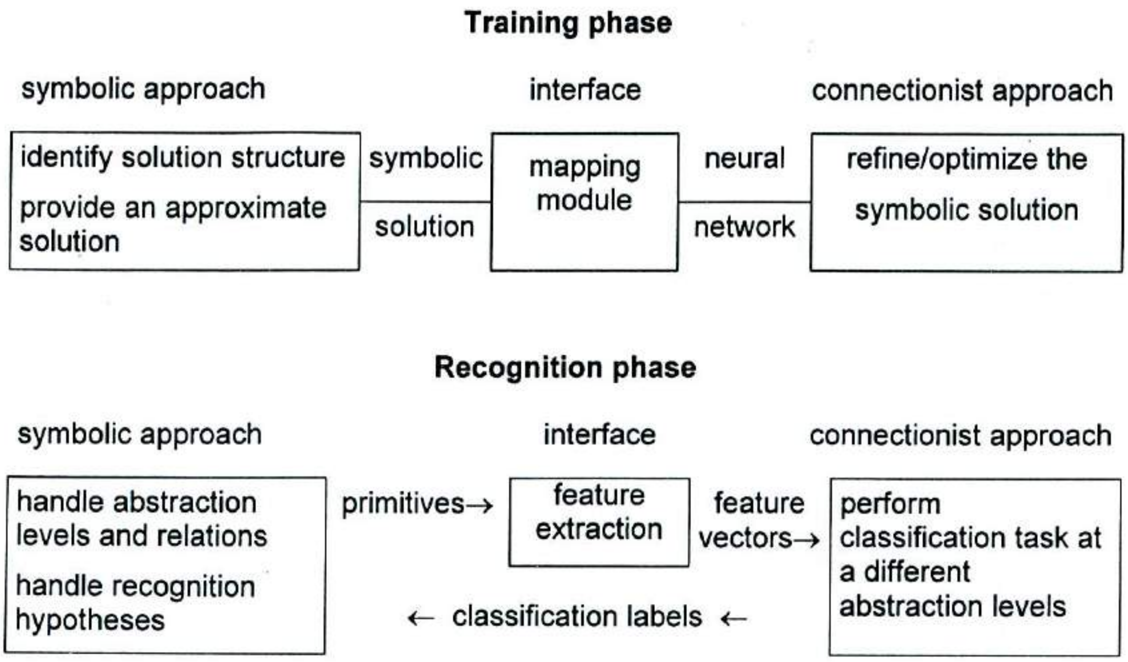

Training phase
symbolic approach
interface
connectionist approach
identify solution structure
provide an approximate solution
symbolic
solution
mapping module
neural
network
refine/optimize the symbolic solution
Recognition phase
symbolic approach
interface
connectionist approach
handle abstraction levels and relations
handle recognition hypotheses
primitives→
feature extraction
feature vectors→
perform classification task at a different abstraction levels
← classification labels ←

Figure 2.5 *Integration paradigm of connectionist and symbolic approaches*

There is another important direction of the pattern recognition research where neural networks issues represent an essential part of the recognition approach. A complex system for visual pattern recognition was proposed by Fukushima [Fukushima, 1988]. The approach is inspired by the visual area of the cerebrum, where neurons are found to respond selectively to local features of a visual patterns, such as lines and edges in particular orientations. In the area higher than the visual cortex, it has been found that exist cells which respond selectively to certain figures like circles, triangles, squares, or even to a human face. On the basis of this hierarchical structure, the same idea can be implemented in constructing the neural network architecture of the similar hierarchy in which simple features are first extracted from stimulus pattern, and integrated into more complicated ones. This is the main idea of the proposed recognition system - Neocognitron, a hierarchical multilayered network consisting of neuron-like cells [Fukushima, 1988]. The implemented system based on the similarity of shapes between patterns is not affected by deformations, nor by changes in size, nor by shifts in the position of the input patterns.

Deco and Blasig propose a handwritten digit recognition system based on the Radial Basis Functions (RBF), known by their local properties, and a preprocessing technique of Principal Component Analysis implementation for the input space reduction [Deco and Blasig, 1993].

Lee uses a cluster neural network where the nodes in the layer are clustered and each cluster is fully connected to a corresponding cluster in the following layer independently. Each subnetwork has started the training from different initial state, thus the network final activity is not influenced by the possible confusion of some of the subnetworks [Lee, 1996].

A similar segmentation of what one layer of the network "sees" by introducing the local connections and shared weights is proposed by Burel and colleagues in [Burel et al., 1992]. The local connections and the share weights follow the idea of the preclassification phase of feature definition and extraction so the specialization is respective to the introduced feature set.

A two stage NN classificator is proposed by Cao and colleagues where at the first stage *an incremental clustering* neural net clusters the gray scale features and inputs in a *subclass* neural nets [Cao et al., 1994]. Gazula and Kabuka use Boolean neural networks for treat the binary and continuous features [Gazula and Kabuka, 1995].

A model for describing the neural network complexity as a system based on linear threshold function through an abstract model of family of circuits is given Parberry [Parberry, 1995].

Chi and Yan use the prototypes created by the Kohonen self-organizing map instead of raw training patterns and generate a small set of fuzzy rules. The optimization of the defuzzification is made by three-layered feedforward network through training on all training parameters. On the digit data from the same data base, as the one used in this thesis, the recognition rate of 96.3% has been achieved [Chi and Yan, 1995].

Kojima and colleagues applied neural networks designed on approximate reasoning architecture consisting of plural functionally combined small-scale NNs to handwritten numeric character recognition [Kojima et al., 1993]. Firstly, the input data are classified by means of a fuzzy clustering. The sub-classification groups are constructed by the Learning Vector Quantization (LVQ). At the final step the classification is made by a combination of the main and the sub-classifiers. The system is tested on the local handwritten data base and the recognition of 95.41% with 4.38 rejection rate is obtained.

A comparison of multilayered neural network and the nearest neighbor classifiers for the case of handwritten digit recognition is given by Yan [Yan, 1995]. The author states that the same recognition rate can be obtained by the nearest neighbor classifier and the equivalent neural network classifier. Even more, as an advantage of the nearest neighbor classifier, the less training time is emphasized. He reported the recognition rate of 97.62% with 0% of rejection for the neural network on the set of 10000 training samples from the NIST base and 10000 test samples.

Wilson and colleagues propose a set of 45 binary decision machines (small neural networks) at the input stage of a single neural network based classifier and/or the majority voting committee [Wilson et al., 1996]. The neural network paradigms adopted in these input and output networks besides the multi-layered perceptron are the radial-basis function network, and the probabilistic neural network. On the basis of only 20 of the features extracted by the Karhunen-Loève method, the best combination of this multistage classification system is reported to be the combination of the single node output.

2.1.3 Character recognition

Character recognition (or Optical Character Recognition, OCR) is an operational step of a system that reads the text from paper, and translate the images of text into a form that can be manipulated character by character. This operational step goes after the Document analysis, and before the Contextual processing of the text image.

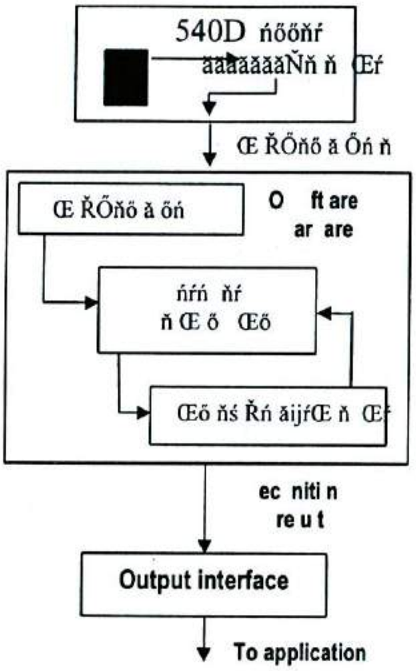


Figure 2.6 *Optical character recognition system components*

OCR has been a popular focus of Pattern recognition research since at least the 1960's. A review of the character recognition techniques is given by Govindan and Shivaprasad [Govinda and Shivaprasad, 1990]. Actually the classification of loosely constrained handwritten digits represents a well studied problem. On the basis of the available variety of the research results and a work that has been done in this field, we will try to emphasize the importance of the same steps of the Character recognition phase, especially the feature transformation and selection phase, the final decision phase and the rejection criteria.

One major problem, mainly in handwritten character recognition because of the pattern variabilities, is to derive an unique feature representation for each class of characters. Not only the combination of classifiers (combination of experts) has been shown as a promising way to extract more substantial information needed for the recognition, but it is specially the case when it is based on the mixture of the approaches on the feature extraction level as well as on classification level. The systems were proposed implementing one kind of features to "not by definition" appropriate classifier. Huette and colleagues introduce the structural and statistical features to statistical classifier [Huette et al., 1996]. We show the results of introducing structural and statistical features to a neural network based classifiers and their cooperation in the environment of the unique neural network classifier, as well as in the decision fusion environment of combining the specialized classifiers on these two feature sets in the next chapters of the thesis.

Bottou and colleagues have given a summary of the performances of some known classifiers for handwritten digit recognition [Bottou et al., 1994]. Several parameters are considered as a raw accuracy, training time, recognition time and memory requirements.

We situate the contribution of this work in the field of Off-line handwriting digits recognition and handprint Cyrillic letters recognition, where the recognition is on segmented characters.

2.2. Features, the Nature

The choice of the features for the representation basis for the representing, learning and classification task is based on their intrinsic nature; the features are [Liu et al., 1998]:

- **primitive** - they are the basic units for defining a problem, a domain, or a world to be observed, and do not require much effort from human experts to design them;
- **convenient** - they are easy to define, implement and use;
- **independent** - the use of features makes data acquisition decoupled from learning or classification;
- **widely used** - they are used in pattern recognition, machine learning, statistics, as well as databases;
- yet still **reasonably general** - so that they are sufficiently powerful for many applications in knowledge discovery and data mining.

We saw that we can consider a pattern as a data structure of features. This structure is characterized by implicit or explicit information on relationships among features so the pattern is generally expressed as a conjunction of statements of the form:

(<feature-name> <feature value, belief in value> <relationship with other values>)

This form of representation traditionally appears in two different modes, arrays of numbers or linear sequences of nonnumeric symbols, respectively to the two traditional areas of pattern recognition: the *decision-theoretic* and the *syntactic* or *structural*. If we fix the order of the feature values, and restraint the representation on numerical values, the patterns became points in N-dimensional space, described in terms of N real-number components. Such representation can benefit of all mathematical instruments about metric spaces, and we can easily reflect the notions of distances, similarities etc. This is not the case with the patterns described by nonnumeric feature values. In some cases the domains of the feature values can be seen as a fuzzy sets. In the cases where it is important whether the observed pattern of symbols could be generated with use of certain production or generation rules, the techniques of mathematical linguistic are used for recognition and classification.

Generally, in the traditional pattern recognition the decision-theoretic approach uses the results of statistical communication and estimation theory, whereas syntactic approach is based on the works of mathematical linguistic and on the research in computer languages.

The recent research in the field of feature examination, including feature transformation and feature selection, spreads over various real-world applications including: Feature structure discovering that are meaningful to a domain expert for allocating housing loans [Zupan et al., 1998], feature reduction in face recognition [Vafaie and De Jong, 1998], text-categorization and natural-scene interpretation [Bloedorn and Michalski, 1998], texture-discrimination and speech-recognition [Pudil and Novovicova, 1998]. In these recent works the data-driven approach is complemented with the emphasized necessity for the problem-oriented specificity of the task and the domain knowledge importance is emphasized as a general need for the feature examination task.

The choice of features to represent the patterns affects several aspects of pattern classification:

1. ***Accuracy.*** Regardless the learning and/or classification algorithm, the amount and the quality of the information given by the features limits the accuracy of the classification function.

2. ***Required learning time.*** The search space that the learning algorithm must explore is determined by the features describing the patterns. The important presence of the irrelevant features unnecessarily increase the size of the search space.

3. ***Necessary number of examples.*** All other components being equal, the larger the number of features describing the patterns, the larger is the

number of examples needed to train a classification function to the desired accuracy.

4. ***Cost.*** Often the feature representation of the patterns consist of features which can be obtained by more or less costly procedures. Eliminating the irrelevant features obviously reduces the costs of the classification task as a whole.

The basic questions arising in the feature definition and handling phases are:

- How are the features generated?, and
- What is the best number of features to use?

The answers of the first one are mainly problem dependent, and they concern the *feature generation* stage of the design of a pattern recognition / classification system.

A practical way to the solution of the second one is the generation of a larger than the necessary number of feature candidates, and then the adoption of "the best" amongst them. This is a very important task and it concerns the feature selection stage of the recognition / classification system. We will discuss more closely these aspects in Chapter 4.

2.2.1 Examples and Ideas

From the very nature of the term feature, the appearance of "things" referenced as features in the pattern recognition, and specially in handwritten or hand-printed character recognition is divers, and the class of "features" is a very rich one.

Implementing a classification tree technique and so separating the entries in a easily separable subset of input images, Meng and colleagues used a *topological* *features* to split the training sample set in such subsets [Meng et al., 1994]. The topological features here are constructed by the number of feature endpoints, number of circles, and number of points that do not belong to the circles. On the bases of those features a fuzzy feature extraction method is implemented to get the feature vectors of each subset.

Knerr and colleagues [Knerr et al., 1992] use a Kirsch masks [Pratt, 1978] for domain independent features extraction to be the inputs of the NN classifier. Kovacs use a combination of features representing *distance transform*, *chain code histograms*, and *bending points* [Kovacs, 1995]. In close relation with the chain code histograms is the *directional histogram* feature extractor [Cao et al., 1994]. Parizeau and Plamodon concentrate too on the morphological aspect of cursive script recognition [Parizeau and Plamodon, 1995]. However, the approach have been tested on the relatively small base of 600 digits, and the performances are from 84.4% to 91.6% the best one, obtained from not a single character recognized, but from a set of top selected characters. The recognition procedure includes the implementation of exclusively morphological and pragmatic knowledge.

[Halici and Erol, 1995] used a Principal Component Analysis for the feature extraction and a Self Organizing Feature Map for the preclassification.

A study from the topological point of view is given in the work of Marjanovic and colleagues [Marjanovic et al., 1994]. The spatial position of the digit arcs and their orientation is examined as the basis for the recognition. The authors look at a digit form as a graph of a multi-valued function and exploit its continuity. A sequence of numbers is attached to a form and the authors prove its relation with the Euler characteristics of the form.

Some concepts developed in the close area of research of word recognition give some perspectives for their implementation in the world of character recognition too. An example is the word recognition without segmentation based on the recognition of subgraphs homeomorphic to previously defined prototypes [Rocha and Pavlidis, 1995]. A positioning network was used for detecting the character appearances within the word, and than to recognize the character by a classification network [Shustorovich and Thrasher, 1996].

A local detection of the line segments in the character image is the basic idea for many proposed feature definitions. Our approach follows partly this idea too in the construction of our structural features set. We have developed an original technique for the local detection and analysis of the line segments in the digit image. However, a starightforward implementation of the first-order differential edge detectors as Chen, Kirsch, Prewitt, Sole and other edge detectors, is a common practice for a local detection of a line segments. Lee uses a Kirsch edge detector for extraction of features to be an input of a cluster neural network [Lee, 1996].

Takahashi uses two kinds of features *zonal-pattern features* (based on smoothed image and contour), and *geometrical features* (based on bending points) as an input to a NN classifier for Japanese phonetic characters [Takahashi, 1991]. A combination of statistical and geometrical features is used by Chen and colleagues for texture classification [Chen et al., 1995].

A combination of the binary and continuos features is proposed by Gazula and Kabuka [Gazula and Kabuka, 1995]. Lampinen and Smolander use the self-organizing maps in the feature extraction phase, so the later supervised learning in the final classifiers acts on reduced number of free parameters [Lampinen and Smolander, 1996].

For various non Ancient Greek based alphabets as Chinese or Arabic in the case of domain dependent features, specific set of features has to been proposed [Toraichi et al., 1990].

In the recognition architecture based on classification trees for pattern recognition Guo and Gelfand propose the idea of using small multilayered networks at the decision nodes to extract nonlinear features [Guo and Gelfand, 1992]. The same idea is implemented by Stromberg and colleagues in a system for speech recognition [Stromberg et al., 1991].

Nakanishi and Fukui propose a recognition based on hierarchical feature type and location [Nakanishi and Fukui, 1993]. In fact a hierarchical feature sets is proposed with the hierarchical sequence of feature detection by type and location.

Autret and Solaiman propose an interesting set of structural features based on the existence of closed circles in the digit image, and its position in the image [Solaiman and Autret, 1991]. Nishida propose a structural feature extraction technique for structural analysis and description of simple arcs or closed curves based on directional features [Nishida, 1995].

Heutte and colleagues propose two set of features called *structural* and *statistical* for their system of recognition of handwritten characters [Heutte et al., 1996]. With this general purpose structural/statistical feature vector, they show the effective representation for the recognition of digits, uppercase letters and graphemes. The classifier is a statistical classifier based on a linear discrimination technique. The 124-variable feature vector represents seven classes of features, ranging from pure structural to pure statistical, including both so called *local* and *global* ones. Having on mind the ideas of this approach which are close to ours (in the terms of feature set definition only), the results of this approach will be discussed more in details in the Chapters 5 and 6 of this thesis. Regarding the proposed feature set we describe here only the features classes which are defined: 1.) *intersections with straight lines*: features that describe the character in the terms of intersections with two horizontal and one vertical line; 2.) *invariant moments*: pure statistical measures of the pixel distribution around the center of gravity; 3.) *holes and concave arcs*; 4.) *extrema*: the top, bottom, left and right extrema of the character; 5.) *end points and junctions*: pure structural features extracted on the skeletonized representation of the character; 6.) *profiles*: the four basic profiles, left right, top and bottom introducing the smoothness of the corresponding character side; 7.) *projections*: the histograms of horizontal and vertical projections.

A classical feature extractor, the Karhunen-Loève transformation is proposed in a two stage binary decision - neural network / majority voting recognition system proposed by Wilson and colleagues [Wilson, 1996]. Karhunen-Loève (K-L) expansion is reported to be an effective dimensionality reductor, and ones which gives optimally compact representation for the input of the binary decision networks at the first stage of the recognition process. The implementation of the sequence K-L as a feature extractor and a simple small binary decision neural network is reported to be an effective and simple low-dimensional method that produce comparable results to those performed by complex networks. The 96 features have been extracted but there was no recognition improvement for more than 32 features.

Based on the shape model (class descriptions) in terms of the high-level features integrating several types of information, Nishida propose a method for synthesizing various patterns incorporating structural, statistical and geometrical deformations [Nishida, 1996].

For the handwritten recognition system by combination of multiple classifiers, Suzuki and colleagues propose a set of features divided in three groups: 1.) *quasi-topological features:* convexity, concavity, and loop; 2.) *directional features:* upward, downward, leftward and rightward; and 3.) *singularities:* branch points and crossings [Suzuki et al., 1996].

Cao and colleagues use a two stage neural network architecture, where, on the first stage a two-layered neural network acts as an feature extractor yielding the principal components [Cao et al., 1997].

Chapter 3

Feature Extraction and Selection

Feature extraction (FE) and Feature Selection (FS) are key phases in the designing of a successful pattern recognition system. Through the history of pattern recognition a significant ambiguity of the definition of these phases is present. The first clear distinction between these phases and an attempt of clear definition of each of them has been made very recently in a Special issue of IEEE Intelligent systems entitled Feature Transformation and Subset selection (1998).

The phases related to the feature creation and manipulation represent an important part of the work of the practitioners from research fields such as statistics, pattern recognition, data mining, knowledge discovery, and machine learning. In general, anywhere where is a need for effective use, processing and accumulating the large scale data, an appropriate preprocessing is inevitable. The research fields where the most of the concepts of feature examination techniques are developed are the *classification* and the *concept description.*

3.1 Definition and Terminology

The number of features that the designer of a classification system has in disposition is usually very large, and moreover, they can be of different nature.

The process through which a new set of features is created is called *feature transformation*. Two variants of the feature transformation are the *feature extraction* (FE) and *feature construction (or feature discovery)*. The difference between these two is: the *feature extraction* extracts a set of new features from the original features through some functional mapping for the dimensionality reduction, and the *feature construction* discovers the missing information about the relationships between features and augments the space of features by inferring or creating additional features. So, in feature transformation, *feature construction* often expands the feature space, while *feature extraction* usually reduces the feature space.

Feature selection (FS), or *subset selection* does not generate a new features; it reduces the feature space by selecting a subset of original features.

Feature transformation and feature selection are not totally independent issues, and this not only from the point of view of their definition. In some applications a combination of feature transformation and subset selection have been shown as a prospective direction of exploration. This combination can be performed systematically like it is in the works of Lavrac and colleagues [Lavrac et al., 1998], or sequentially, using a genetic algorithm to order the features in hierarchy, and than adding the best features to the previous set of features, proposed by Zupan and colleagues [Zupan et al., 1998].

If we consider features as a representation language, we can view these phases as two sides of the representation problem. The most of the research on the subject of the nature of these phases has been done in the areas of *concept description*, *classification* and *recognition* [Milgram, 1993]. In the concept description the aim is to preserve data's topological structure, whilst in the classification and recognition the aim is to enhance the predictive power. The importance of the techniques of feature transformation and subset selection goes beyond the limits of the machine learning, statistics or pattern recognition where those techniques are most developed, and can be interesting issue in many fields where they are performed implicitly, which is the case, for example in the switching-circuit design in electrical engineering.

Very often in pattern recognition research those phases were considered as something which was happening implicitly during classifier training and optimization, but also sometimes even the importance of those phases was underrated.

An exhaustive survey on the FE methods for off-line recognition of segmented characters is given in [Trier et al., 1995]. The authors group the various feature selection techniques on the basis of the feature nature: features extracted from gray-scale images, from binary images, from binary contours, from vector representation, and features obtained by neural network classifiers.

The problem of FE or FS can be simple defined in the following way: We need to reduce the number of features in the source feature set V, |V|=n by constructing new feature set S, |S|=m where m≤n, in order to achieve more efficient classification, i.e. to reduce the cost and/or to increase the performance of the system.

The simplest way to explain the differences between these two phases is to define the FE as a task of finding new features, and the FS as a task of making choice which of them to use in the classification process. A bit more sophisticated approach is to define the FS as a mapping which preserves feature individuality

$$f(\{x_1, x_2, \ldots, x_n\}) = \{f_{i_1}(x_{i_1}), f_{i_2}(x_{i_2}), \ldots, f_{i_m}(x_{i_m})\}, \ m < n$$

Looking at FS this way, we can say that the FE is any other mapping which combine the features.

Many other terms are used in relation with FE and FS as: *feature construction*, *feature discovery*, *feature pre-processing*, *feature transformation*, *dimensionality reduction*, *subset selection* etc.

For example, *feature construction* could be defined as an augmentation of the feature space by inferring features (length, width ⇒ 2·length + width), while *feature discovery* could be defined as an augmentation of the feature space by missing relations among features (length, width ⇒ area). According to this terminology, FE is the process of the reduction of the feature space by generating new features, while subset selection is the process of the reduction of the feature space without generating new features.

The following terminology, which uses two views on FE and FS seems to be closest to the most frequently used terminology in the literature [Devijver and Kittler, 1982]. The FE can be considered as a process of preparation of the measurements for classification, or as a feature selection in the transformed space. The FS can be considered either as a process of feature selection in the original space or in the transformed space.

According to this terminology FE and FS are overlapped and FS in transformed space is considered as a FE. The correspondence between the last and the former terminology can be made in the following way. By feature construction and discovery we consider the preparation of the measurements for the classification. FS in original space corresponds to a *subset selection*, while FS in transformed space corresponds to the term *feature extraction.*

Often we can discover and construct practically unlimited number of features. However, we perform an implicit FS during preparation of the row data for classification by limiting our attention to intuitively "good" features. Then, we approach to a real dimensionality reduction based on some criterion.

There are more reasons to include dimensionality reduction in a pattern recognition system:

- In practice, less features can increase the classifier accuracy,
- Less features need less learning samples,
- Less features result in less time consuming classifier,
- The chances to design a good classifier are better with a reduced number of features.

In addition, there are some more abstract reasons:

- It seems that the human use only few features to perform classification,
- Almost in all situations there are redundant or irrelevant features,

- By keeping the number of features under control we avoid the problem of "curse of dimensionality" (classifier complexity grows exponentially by the number of features).

All of the above reasons are logical, excluding the first one which say that the classifier performance can be increased by deleting a feature. Theoretically, it seems strange because we expect to improve the classifier performance by discarding some of the input information. But in practice, where we dispose of finite learning set and a non-ideal classifier which is not able to use input information in an optimal way, this phenomenon becomes natural.

For example, let us consider a NN classifier with hidden layers, as it is presented in Fig. 3.1. Each hidden NN layer can be considered as a feature extractor combining the input (or outgoing from the former hidden layer) features into new features. It is well-known that the NN performance can be improved by deleting the unit in the hidden layer (up to same limit however), that means by deleting feature.

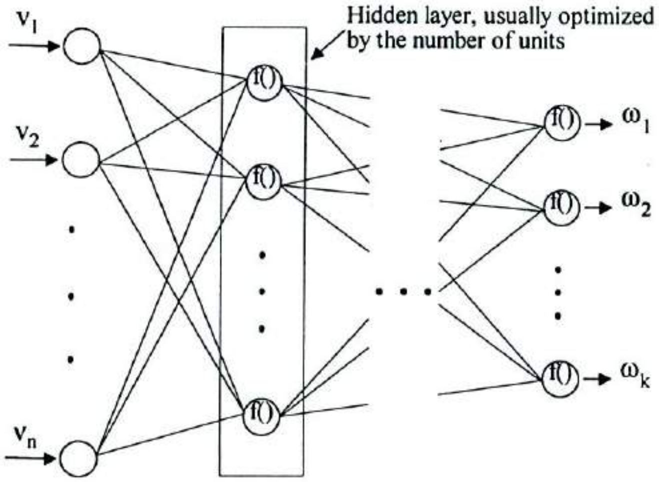


Figure 3.1 *A neural network classifier*

It is interesting to discuss this phenomenon in the case of classification by humans. If we agree that the human classification can be improved by non-taking into consideration some input information, then that improvement is a result of the uncompleted prior knowledge. It corresponds to the finites of the learning set in the case of artificial classifiers. Rui-Ping and colleagues give the study about the relation of the feature weights and feature selection in pattern recognition neural networks [Rui-Ping et al., 1996].

In practical applications, the system parameters and the need for samples grow rapidly with dimensionality which results in decreasing of the generalization ability. So, although the dimensionality reduction is often performed implicitly in the classifier (for example, by optimizing the number of hidden layers nodes in a NN), indeed, it is still beneficially to perform it separately.

It is worth mentioning that an increasing attention is being given to the data preprocessing as an essential step for knowledge discovery in many real-world applications in various area of research. Not only in pattern recognition area, the practitioners from fields such as statistics, data mining, knowledge discovery and machine learning have more and more interest in feature transformation and subset selection.

The finding of feature sets that are "optimal" in terms of size and performance represent a significant opportunity for improving the recognition techniques. Recently, the application of genetic algorithms has been shown as a promising technique for performing an efficient search the large space of feature candidates.

It has been stated [Trier, et al., 1995] that often a single feature extraction method alone is not sufficient to obtain good discrimination power. We will show further on in this work, specially in Chapter 4. a combination of features obtained from different feature extraction methods. Moreover, we will show a combination of some aspects of the statistical, structural, and neural network approaches in the phases of FE and FS, as well as during the classification phase.

3.2 Dimensionality Reduction: Methods

As it was stated that the feature transformation techniques are in fact the techniques for dimensionality we can categorize them using different criteria:

- The place where they are performed: in *original,* or in *transformed space*

We have already discussed some aspects of this categorization as distinction between FS and FE. In the case of FS, we completely eliminate the features, while in the case of FE, we usually use information from all features. Thus, FE seems to be a more promising because the chance to lose relevant information is reduced, and moreover, it is usually a faster approach to dimensionality reduction. On the other hand, in the case of FE it is more difficult to reject irrelevant information. The feature extraction causes a loss of interpretability, while, feature selection preserves data interpretability. FS has lower discriminative power than FE does.

- Individual - Collective

Individual approach considers and estimates the importance of the features one by one, while the collective acts on the subset of features. The individual FS can be seen as a preparation for collective FS enabling this process to be "easier".

- Supervised - Unsupervised

If we already dispose of pattern classes, it is reasonable to use a supervised dimensionality reduction. Unsupervised methods are methods used on a non-labeled entry data - the parallel can be made between recognition and classification applications.

- Classifier dependent - Classifier independent

Indeed, it is an advantage to design general, classifier independent techniques, but unfortunately they are less promising because the quality of the feature set depends on the classifier being used.

- Feature dependent - Feature independent

Feature dependent dimensionality reduction could be extremely useful because it includes prior knowledge about the concrete problem.

- NN implemented - Non NN implemented

Having on mind the increasing implementation of Neural Networks techniques in pattern recognition, and generally in classification applications it is worth of considering a group of approaches around the NN implemented dimensionality reduction.

3.3 Feature transformation: Discovery and Extraction

After a long history of ambiguity of the utilization of the terms of FE and FS in the literature, an attempt for clear distinction between those and other terms has been made in a Special issue of IEEE Intelligent systems entitled Feature Transformation and Subset selection (1998). So, the process of the *feature transformation* is defined as a process through which a new set of features is created. Variants of feature transformation are *feature construction* and *feature extraction*, both called sometimes *feature discovery*.

Feature construction discovers a missing information about the relationships between features and augments the space of features by inferring or creating additional features. For example, after a feature construction, we might have additional m features A_{n+1}, A_{n+2}, ..., A_{n+m}, where a new feature A_k ($n < k \leq n+m$)

could be constructed by performing a logical operation on A_i and A_j from the original set.

Feature extraction is a mapping from the feature space with higher dimensionality to a feature space with lower dimensionality. In other words, we are looking for new features which will improve the performance of the classifier being used. The FE is a process of extracting from the raw data the information which is most relevant for classification purposes, in the sense of minimizing within-class pattern variability, while enhancing the between-class pattern variability [Devivjer and Kittler, 1982]. The FE is a process of search among all possible singular transformations, for the best subspace which preserves class separability as much as possible in the lowest possible dimensional space [Fukunaga, 1990]. Ramdas and colleagues considered this "separability" by extracting features by an effective clustering [Ramdas et al., 1994].

A very complete survey of the state of the art in the area of FE methods for the character recognition task can be find in [Trier et al., 1996]. The FE method is seen in a firm relation with the pattern recognition system as a whole. The directions of choosing a FE method for character recognition include the information about the characters to be recognized and the classifier to be used for the recognition. Concerning the input character information, the main points are: the type of the input characters (single-font, or multi-font typed or machine printed, neatly hand-printed or unconstrained handwritten), the variability of the characters belonging to the same class, the type of the image (gray-level, binary or vector), the scanner resolution. The classifier's influence on the FE method to be chosen is concentrated around the nature of the classifier: statistical or structural, and the throughput and the recognition requirements (specially for the rejection criteria) .

The problem of *feature extraction* could be defined as a search for the transformation of the feature set $S=W(V)$, where $W: R_n \rightarrow R_m$ $(m \leq n)$, which maximize a criterion of optimality J(.):

$$J(\mathrm{S}) = \max \left\{J(\mathrm{T}) \mid \mathrm{T} \in \{W(\mathrm{V})\}\right\}$$

The classical problem of function approximation is similar. However, in the case of FE the desired outputs are usually not known while in the case of approximation the unknown function is estimated using the set of pairs (input, desired output).

It is interesting that the process can be repeated many times, always trying to improve the already improved feature set, shifting the solution into some far distant point. In practice this sequence must be cut (usually immediately).

The phase of FE has important practical advantages in comparison with FS. It is usually implemented more efficiently than the search techniques for subset selection. In addition, one can find advantage in the fact that these techniques do not discard the features completely like the techniques for FS. This possibility is theoretical because, indeed, it is not known how to construct a transformation that will keep all relevant and discard all irrelevant information. Thus, this advantage seems relative and can be consider as a given larger maneuver space for construction of improved feature set.

We have already discussed that FS is a very difficult problem from both, theoretical and practical point of view. On the other hand, it is obvious that such FS is a special case of FE if we consider trivial projection transformation. So, it is practically impossible to discuss optimality of the transformation W(.), i.e. of the produced subset S, except in the case of very limited domain of transformation like linear [Fukunaga, 1990]. In many applications of pattern recognition, there are important features which are not linear functions of the original measurements, but are highly nonlinear functions. The basic problem is to find a proper nonlinear mapping function for the given data. Since we do not have any general algorithm to generate nonlinear mapping functions systematically, the selection of features in the transformed space becomes very much problem-oriented. In the case of features for signal representation the central problem is the estimation of the eigenvalues and eigenvectors.

The representatives of FE in this area are the methods of factor analysis or principle component analysis. All features considered are second-order statistics of the distribution, the covariance or autocorrelation matrix. The methods in this regard are:

- Minimum Mean-Square Error by Discrete Karhunen-Loéve expansion (the expansion of a random vector in the eigenvectors of the covariance matrix),
- Scatter measure (the expected value of the squared between-sample distance),
- Population entropy (as a measure of diversity of a distribution).

A number of networks and learning algorithms providing tools for feature extraction and data projection are given by Mao and Jain in [Mao and Jain, 1995]. The common attribute of these networks is that they employ adaptive learning algorithms which makes them suitable in the environments where the distribution of patterns in feature space changes in respect to time. The linear discriminate analysis (LDA) network, SAMANN network for Sammon's projection, nonlinear projection based on Kohonen's Self Organizing Maps (NP-SOM), and nonlinear discriminate analysis network (NDA) using a feedforward network classifier are presented and compared.

FE is potentially a very powerful approach for a dimensionality reduction. The basic idea in classifiers like NNs is based on FE. Hidden NN units learn during the training how to construct new significant features from the input, performing FE. In addition, all classifier could be considered as a feature extractor transforming input features into new features which make the classification to be trivial. Usually, these final features are an estimation of Bayes' posterior probabilities. The difference between FE in a hidden layer and FE in the output layer is that in the last case the desired output is known. Thus, in an extreme case, the FE problem is equivalent with the classification problem.

3.4 Feature Selection

Feature selection is often refereed as a process that chooses an *optimal subset of features* according to a certain criterion [Liu et al., 1998]. Feature (or attribute, property, characteristic as it's refereed sometimes) selection, in contrast to feature extraction, is a form of representation space destruction. It is an important phase of any learning-based algorithm. The better the representation space, the easier it is for the program to learn. Thus, the feature selection is the process of determining the relevant features, but also the process of removing the irrelevant ones. It is an important if not crucial phase in the applications of pattern recognition, machine learning, data mining and related area [Blum and Langley, 1997, Jain and Zongker, 1997, Liu et al., 1998]. Feature selection can be seen as a special case of *feature weighting* with binary weights. In feature weighting, the weight associated with a feature measures its relevance or significance in the classification task, consequently by implementing the binary weights, we obtain a feature selection or a subset selection in the above defined sense.

The quality of a feature can be evaluated based on two considerations: pattern generation knowledge (if the feature succeeds in characterizing the pattern class) and statistics (the ability of the feature to provide large inter-class separation and small intra-class spread [Duda and Hart, 1973]. Krishnan and colleagues propose a generalized definition of the Fisher ratio (the measure for ranking features based on the cluster means and cluster variances) and show the usefulness of the discrimination criterion based on mixing proportions of the component densities [Krishnan et al., 1996].

If V is a source feature set, in the task of feature (subset) selection we are looking for a set $S \subset V$ such that

$$J(S) = \max \{ J(T) \mid T \subset V \}$$

where J(.) measures the quality of a feature set. However, in practice, it is often enough to find a feature subset $S \subset V$ such that $J(S) \geq J(V)$ or even more, sometimes reducing the number of the features can be paid by loosing some of the feature set quality, $J(S) \geq J(V) - \varepsilon$.

Any FS procedure has to be based on the following two components:

- An optimality criterion must be defined to make judgment whether one feature subset is more appropriate than another, and
- An appropriate procedure for moving through the feature subset must be designed.

The first component is connected with conceptual difficulties which could be expressed in the following way:

What does it mean that a feature is good or irrelevant, i.e. what criteria should be used to evaluate features?

The second component is connected with practical difficulties. Very often the number of the features that could be extracted from the measurements can be practically unlimited (each transformation of the features is a new feature) and the number of subsets that have to be examined to find the optimal is 2^n.

Good choice of the optimality criterion is extremely important to make a reliable FS. Perhaps, the most important categorization of the optimality criteria is on those that are classifier independent and those that are based on the classifier being used.

3.4.1 Classifier Independent Optimality Criteria

Classifier independent criteria are the criteria based on the direct analysis of the sample set. In this group we include most of the criteria based on probabilistic and statistical analysis of the sample set. An example of the approach is the criteria based on evaluation of the Bayes' error given by Ben-Bassat [Ben-Bassat, 1980]. Siddiqui and colleagues used an optimization model based on the proximity index to select features [Siddiqui et al., 1994]. An early study of the selection of the variables in multiple regression is given by Thompson [Thompson, 1978].

Another criteria used in the same direction is the estimation of class densities [Pudil et al., 1994, 1995], distance measures of class separability [Devijver and Kittler, 1982, Fukanaga, 1992], correlation [Schurmann, 1996], mutual information [Battiti, 1994, Bollacker and Ghosh, 1996, Bonnlander and Weigend, 1994].

Novovicova and colleagues propose a feature selection procedure based on Kullback J-divergence between two class conditional density functions approximated by a finite mixture of parametrized densities, which enables to find a feature subset of any cardinalty without involving search procedure [Novovicova et al., 1996].

Theoretically, some of these criteria are ideal to evaluate features (as Bayes error). Also, they are often relatively easy for calculation. Unfortunately, frequently there are no enough reliable ways to be directly calculated from the samples. The common characteristic of these criteria is that they are indirect, i.e. independent of the classifier being used. Additionally, many of these criteria do not address non-monotonocity as we will discuss later.

The classifier (the learning algorithm) independent approach in machine learning community is refereed as a *filter* approach. Generally they are computationally more efficient, but the major drawback of this approach is that an optimal selection of features may not be independent of the inductive and representational biases of the learning algorithm that constructs the classifier.

3.4.2 Classifier Dependent Optimality Criteria

Here, we include the criteria which directly use the performance of the classifier being used or estimate the feature contributions in the classification process. The FS techniques that use the classifier performance are usually deduced to suboptimal search procedures through the feature subsets [Siedlecki and Sklansky, 1988]. The feature individual contributions in the classification process can be calculated during classifier training [Cibas et al., 1996] or on trained classifier [Cibas et al. 1996, Bennani 1996, 1997]. These criteria are very important because of the possibility to be very efficiently computed. Moreover, some authors develop techniques that integrate the feature selection with the architecture selection process [Steppe et al., 1996].

This approach is addressed as a *wrapper* approach [Battiti, 1994] in machine learning community. The induction algorithm is used as a part of a evaluation function to perform the feature subset selection. In general, the implication of this criteria incurs the computational overhead of evaluating candidate feature subsets by executing a used learning algorithm on the data set using each feature subset under consideration. Often the search over all possible combinations of features is not computationally feasible. Most of current approaches assume monotonicity of some measure of the classification performance and then use branch-and-bound search. The problem is that the techniques that make the monotonicity assumption work reasonably well with linear classifiers. There is not known an approach that will exhibit a reasonable performance with nonlinear classifiers such as neural networks.

However, many authors agree that the criteria based on the performance of the classifier being used is the only way to perform an optimal FS. Classifier design is based on finite sample set and should refer to any particular classifier performance rather than some other characteristic of the sample set.

Unfortunately, in this case, FS includes any structural error caused by the classifier. In addition, there is no known optimal training procedure for finite samples using classifier performance as a criterion of optimality [Siedlecki and Sklansky, 1988].

The main problem in choosing criterion for FS could be simple summarized in the following way:

If we use as a optimality criterion something different than the classifier's performance, we can not predict the real usefulness of the selected set for the classifier being used. Simple speaking, the quality of the selected feature subsets depends on the classifier, and consequently, its performance is a most reliable way to measure their usefulness.

If we use as a optimality criterion a classifier performance, it is known that such an information is very biased, and very often it is computationally impossible to be obtained.

So, in each case we are confronted with serious problems.

Unlike this discussion, some authors [Pudil et al., 1995] consider the criteria based on classifier performance as indirect because the quality of the feature sets are not directly provided from the sample set. Their view on the problem seems to be more "idealistic" avoiding to address the fact that the sample set is finite and biased, that the classifier is non-ideal, and consequently, that the quality of the feature set depends on the classifier being used.

In relation with the optimality criteria is the term monotonicity, which means that the criterion function grows monotonically over any nested feature subsets S_1, S_2, ... ,S_k:

$$S_1 \subseteq S_2 \subseteq \ \dots \ \subseteq S_k \ \Rightarrow \ J(S_1) \leq J(S_2) \leq \ \dots \ \leq J(S_k)$$

Such criteria make the FS problem to be very simplified because the optimal set is the source feature set V. So, we should consider only the case of making compromise between some degradation of criterion function and reduction of the number of features. The branch and bound algorithm solves this problem in an optimal way [Narendra and Fukunaga, 1977]. Thus, the criteria which keep monotonicity like Bayes error or distance function of the scatter matrices have an additional drawback. As we already discussed, in practice, the classifier performance can be increased by deleting a feature that is in contradiction with the monotonocity.

After the optimality criterion is defined, the search procedure for moving through the feature subsets has to be designed.

3.4.3 Feature Subset Searching Procedures

The searching procedures for detecting the optimal feature subset has become a main-stream of the research in the area. In fact, it is based on the developing computationally feasible procedures designed to avoid the exhaustive search, even though the obtained feature set may be sub-optimal. The classical approaches in this direction include a "bottom up" and the "top down" method of subset searching. The former has an empty feature set as a starting point, whilst the later starts with a feature set containing the complete set of measurements. The "top down" approach is introduced as *backward selection* in 1963 by Marill and Green, and its counterpart known as *sequential forward selection* by Whitney (1971).

An overview of major classical feature selection methods is presented in [Devivjer and Kittler, 1982]. A schematic representation of the feature subset search procedures is given in Fig. 3.2.

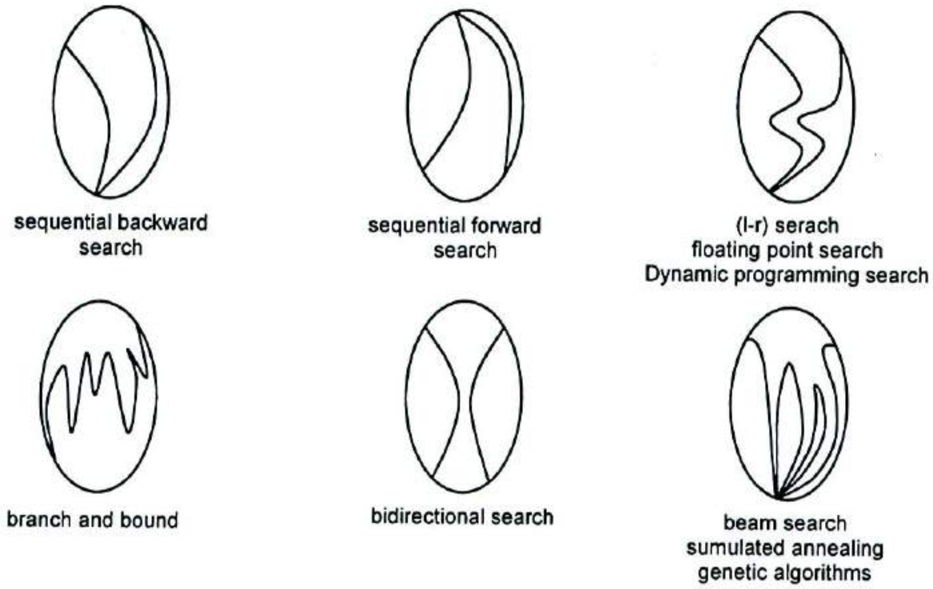


Figure 3.2 *A feature subset searching procedures*

It is useful to divide the search procedures in two groups, according to the way in which they approach the problem [Schurmann, 1996]. In the first group are the techniques that evaluate features individually and than construct the feature subset. The simple way to do it, is to check the performance of a single-feature classifier and to collect best *m* features into the new feature subset. As in this case the input space is one-dimensional, it is relatively easy to perform such a check.

Individual feature evaluation can be low time consuming and it gives the usefulness of individual features that could be important for classifier design, and generally, for better understanding of the problem.

The main drawback of the individual feature evaluation is that it does not take into consideration the combination among features that can drastically change their usefulness (two individually best can produce one weak and vice versa). In fact, after calculation of the usefulness of individual features we select m best or the features whose usefulness exceeds some threshold value. If we have already spent efforts to obtain the usefulness of individual features, intuitively, it is reasonable to use that simple variant of subset construction, although a bit more sophisticated approaches can give better results. For example, in the approach proposed by Battiti [Battiti, 1994] the feature with best mutual information with the output and worst with the already selected features is included in the optimal feature subset.

Gurong and colleagues propose a recursive algorithm called Bhattacharyya distance feature selection for selecting a real-optimum feature under normal multi-distribution. The problem of minimizing the criterion of the sum of the upper bound of error probability of every two class pairs is changed into problem of solving a nonlinear matrix equation in a multi-class problem under an orthonormal coordinate system [Guorong et al., 1996]. Koller and Sahami examine a method for feature subset selection based on information theory. The method consists in eliminating the feature if it gives a little or no additional information beyond that subsumed by the remaining features [Koller and Sahami, 1996].

The selection of best individual features is likely to be unreliable and could be appropriate in the case of highly independent features. In other case, we again need search procedure for feature subset selection.

The representatives of individual feature evaluation are the following techniques:

- Direct evaluation of the Bayes posterior probabilities or performance of some single-feature classifier [Bishop, 1995],
- Mutual information [Battiti, 1994],
- Correlation [Schurmann, 1996],

Individual feature contribution in the classification process (NN-based techniques: OCD, weight pruning, regularization, [Cibas et al. 1994, 1996, Bennani 1996]).

Since combination of features can provide significant information which is not available in any of the individual features, the logical alternative is to consider feature subsets, i.e. to evaluate feature collectively. This approach completely addresses the key issue: contribution of the single feature after a set of features is already selected. For example, the significance of feature x_i for the subset S can be defined as:

$$Sig(x_i,\mathrm{S}) = \begin{cases} J(\mathrm{S}) - J(\mathrm{S}-\{x_i\}), & \text{if } x_i \in \mathrm{S} \\ J(\mathrm{S}\cup\{x_i\}) - J(\mathrm{S}), & \text{if } x_i \in \mathrm{V}-\mathrm{S} \end{cases}$$

Further, we can define best and worst features for the set S,

$$\text{best(S):} \qquad x_b = \max_{x_i \in \mathrm{V}-\mathrm{S}} \{Sig(x_i,\mathrm{S})\}$$

$$\text{worst(S):} \qquad x_w = \min_{x_i \in \mathrm{S}} \{Sig(x_i,\mathrm{S})\}$$

and design appropriate search procedure which will permanently include best features into S and exclude worst features from S until a terminate criterion is satisfied [Pudil et al., 1994].

In general case, the only way to find the optimal feature subset is to evaluate the performance of all feature subsets that is NP hard problem. Here, the term optimality means optimality according to some criterion.

Unfortunately, the question of the trade-off between optimality and efficiency of algorithms for NP hard problems was recognized early and the main efforts were directed toward sub-optimal search procedures. In addition, this kind of FS combined with the classifier performance as a criterion of optimality is very problematic. In such case, it is necessary to train and optimize the classifier for each input feature subset which is known to be very biased and computationally too expensive process.

The biasdeness of a classifier performance could be decreased by techniques as cross-validation, but that increases already discouraging complexity of the procedure.

The representatives of the techniques for collective evaluation of feature sets, often called classical approaches, are the various sub-optimal search procedures as:

- Sequential forward and backward searches and (l-r) search [Devijver and Kittler, 1982],
- Bi-directional search and Beam search [Siedlecki and Sklansky, 1988],
- Branch and bound [Narendra and Fukunaga, 1977]
- Genetic algorithms & simulated annealing [Siedlecki and Sklansky, 1988, Brill et al., 1992],
- Floating point forward & backward search [Pudil et al., 1994].

In [Siedlecki and Sklansky, 1988] an exhaustive review on classical FS techniques is given. Zongker and Jain give a parallel evaluation of various feature selection techniques on the set of synthetic data [Zongker and Jain, 1996].

It is interesting to note, that most of the search methods work backward. That is due to the fact that it is a faster way to achieve acceptable result (increased performance with reduced number of features).

Recently the genetic algorithms took an important place in the field of feature selection. This is due to their capacity to efficiently search large spaces about which little is known and have proved to provide robustness. Vafaie and colleagues have developed the genetic algorithm based representation transformation that can select and create appropriate features to suitably represent a problem [Vafaie et al., 1998]. Pudil and Novovicova are developing the feature selection guide, an environment that will integrate a family of methods with an expert system. They give a substantial flowchart of a subset selection guide and propose one computationally effective floating-search method, and a method wich trades off the requirement for a priori information for the requirement of sufficient data to represent the distributions involved [Pudil and Novovicova, 1998].

3.5 Individual Feature Importance Strategy for Feature Selection

The search procedures can be based on individual or collective evaluation of the features [Schurmann, 1996]. Individual feature evaluation is low time consuming and gives the importance of the individual features.

The main drawback of the individual feature evaluation is that it does not take into consideration the combination among features that can drastically change their importance.

Since the combination of the features can provide significant information which is not available in any of the individual features, the logical alternative is to consider feature subsets, i.e. to evaluate the features collectively. Unfortunately, if we use classifier performance as a criterion of optimality, it is necessary to train and optimize the classifier for each input feature subset .that is very biased and computationally too expensive process [Siedlecki and Sklansky, 1988]. An excellent overview and comparison among various optimality criteria and search strategies for collective feature evaluation is given in [Kohavi and John, 1997].

Above discussion implies that the combination of individual-collective feature evaluation could be a promising research direction, based on a trade-off between complexity and reliability of the FS procedure.

Hereon, we discuss and illustrate potential usefulness of some search strategies for FS based on individual feature importance. We present a novel method based on $Sat(\cdot)$ (satisfy) function that tries to address the key issue in FS procedures:

the contribution of the feature after a set of features is already selected [Cakmakov and Radevski, 1998]. The search based on the *Sat*(·) function is an alternative to the search procedures such as branch and bound approach [Narendra and Fukunaga, 1977] or floating point forward and backward searches [Pudil et al., 1994] where it is computationally impossible to use more complex optimality criterion (like classifier performance), even for small number of features.

For difference of many authors, we use real sample set (about 30000 samples and 116 features) to present potential usefulness of the proposed techniques. Indeed, we are aware of the advantages of using cross-validation techniques to increase accuracy of estimates. However, the number of features and samples makes such examination very difficult (time consuming) and it is left for future work.

3.5.1 Procedures for FS Based on Individual Feature Importance

When we dispose of individual feature importance, it is possible to design various efficient search strategies for reduction of the input feature set F.

All presented strategies will be based on the function obtained by the points of estimated qualities of each feature subset F_j, j=1,2,...,n, containing j individually best features. We call this function Nested-Set Performance Function (NSPF). If the features in the feature set F=$\{x_1, x_2, ..., x_n\}$ are ordered in decreasing order of their individual importance, then $F_j=\{x_1, x_2, ..., x_j\}$. The qualities of the subsets S_k could be estimated using any optimality criterion $J(\cdot)$. It is possible to use various search procedures for feature subset selection based on the NSPF:

- Selection 1:

  - *the best m* ($m < n$); Actually, for this selection we do not need the NSPF.

- Subset whose performance exceeds some threshold value;

- Subset which corresponds to some local (including global) maximums in the NSPF.

- Selection 2:

  - Exclusion of "non-monotonic features", S = F-{*features in which the NSPF decreases*};

  - Partitioning of the feature set based on local characteristics of the NSPF: good, suspicious, weak and bad features.

- Selection 3:

  - A heuristic search based on $Sat(\cdot)$ *(satisfy) function.*

Let us F={x_1, x_2, ..., x_n} be set of features in decreasing order of their importance, and F_j={x_1, x_2, ..., x_j}, j=1,2,…,n corresponding nested sets of features. If the current selection is $S = \{x_{i_1}, x_{i_2}, \ldots, x_{i_k}\}$, we define the function $Sat(\cdot)$ as follows:

$$Sat(x_{i_{k+1}}) = \begin{cases} 0 \ \ \textit{if} \ \ J(S \cup \{x_{i_{k+1}}\}) < J(F_{|s|+1}) \\ 1 \ \ \textit{if} \ \ J(S \cup \{x_{i_{k+1}}\}) > J(F_{max\{i_j\}, j=1,2,\ldots,k+1}) \\ [J(S \cup \{x_{i_{k+1}}\}) - J(F_{|s|+1})] \cdot \delta \end{cases}$$

where $\delta = \delta(k$, local situation in the NSPF).

The $Sat(\cdot)$ function is an attempt to address the key issue in FS procedures: the contribution of the single feature after a set of features is already selected. It takes value 0 if the current order of FS is worse then the source order, and value 1 if the current order of FS is better then the source order until the highest feature number in S (we have better performance with smaller number of features). In other cases, the $Sat(\cdot)$ function measures our "satisfaction" of inclusion of the new feature through the difference between the current order of FS and the source order of FS (the same number of individually best features). Parameter δ tries to correct our "satisfaction" and it increases by growing of the number of selected features. In other words, we should be satisfied by less and less improvement of the performance of the current selection as the number of selected features grows. The Parameter δ uses the natural proposition that a new feature contributes more in feature subsets, i.e.:

$$\mathrm{S} \subseteq \mathrm{T} \Rightarrow J(\mathrm{S} \cup \{x\}) - J(\mathrm{S}) \geq J(\mathrm{T} \cup \{x\}) - J(\mathrm{T})$$

where x is a feature and S, T are two feature subsets.

The following simple algorithm uses the $Sat(\cdot)$ function to build the feature subset S:

```
$\mathrm{Q} \leftarrow \mathrm{S} \leftarrow \varnothing$
$k \leftarrow 0$
while $J(\mathrm{S}) < perf$ and $|\mathrm{S}| < m$ do
    / The feature $x_{k+1}$ is examined /
    if $Sat(x_{k+1}) > 0.5$ then $\mathrm{S} \leftarrow \mathrm{S} \cup \{x_{k+1}\}$
                         else $\mathrm{Q} \leftarrow \mathrm{Q} \cup \{x_{k+1}\}$
    $\mathrm{F} \leftarrow \mathrm{F} - \{x_{k+1}\}$
    if $\mathrm{F} = \varnothing$ then
        $\mathrm{F} \leftarrow \mathrm{Q}$
        $\mathrm{Q} \leftarrow \varnothing$
        $k \leftarrow |\mathrm{S}|$
    $k \leftarrow k + 1$
```

We will briefly explain the idea behind this search. Let us suppose that we have already selected k features into S and F_k are nested sets of features in decreasing order of their individual importance. Then, if the feature x_{k+1} obtains good value of $Sat(\cdot)$ it is included in S, otherwise the feature x_{k+1} goes in the queue Q to be eventually later included. The *perf* and *m* are bounds for desired performance and the number of features.

3.5.2 Experimental Results

To illustrate the usefulness of the proposed approaches we used an application of hand-written digit recognition.

The data base for the experiment contains 23898 digits extracted from the segmented handwritten character base NIST [NIST, 1992].

The feature set is composed of 116 features that could be classified as 54 structural and 62 statistical [Radevski and Bennani, 1997].

To obtain individual feature importance the Optimal Cell Damage technique [Cibas et al., 1994] is applied on a trained two-layer NN with 116 cells in the input layer, 35 in the hidden layer and 10 - one for each of the ten digit classes in the output layer. This technique is based on the saliency of the units in the input layer.

The NSPF is obtained using the performance of NNs (k-14-10) as a criterion for feature subset quality ($1 \leq k \leq 116$).

In Fig. 3.2, the results of experiments are presented. All results are presented as a percent of the digit recognition for the learning and the test set obtained by dividing the training set (23989 samples) in relation 3:1. The NSPF gives the results for any subset of k ($4 \leq k \leq 116$) individually best features. Those values are basis for comparison.

The Tab 3.1 shows some performance results at the points where the NSPF has peaks (selection 1). The first column always gives the performance on learning set and the second the performance on the test set. The first two approaches in selection 1 are given by the NSPF alone.

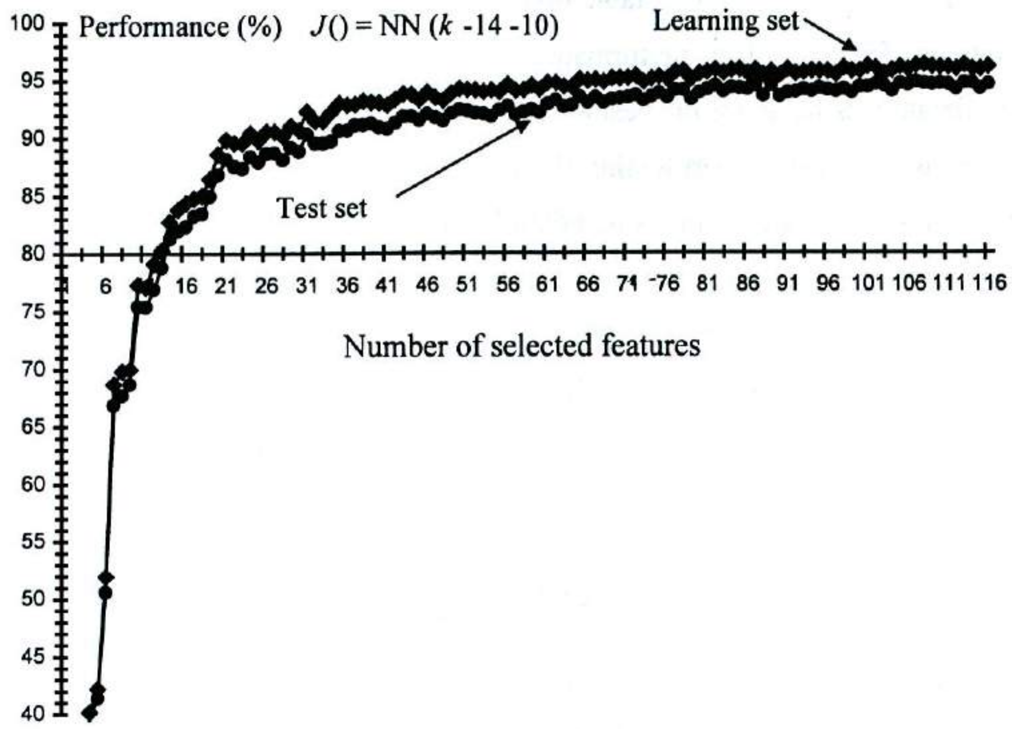


Figure 3.3 *Nested set performance function*

In the Tab 3.2 and Tab 3.3, the first two columns give the performance of the selection 2 and 3 respectively. The columns 3 and 4 give the corresponding NSPF values for the same number of features. These values enable easy estimation of the potential usefulness of the corresponding selection.

selection	performance %	
60 (individually best)	94.23	92.23
31 (local maximum)	92.28	90.33
56 (local maximum)	94.67	92.77
78 (local maximum)	95.85	94.16
107 (local maximum)	96.31	94.89

Table 3.1 *NSPF feature selection - 1.*

The first row in second table gives performances obtained by excluding non-monotonic features. The performance of such 84 selected features remained approximately same as for the best 84 features. The other rows give performances of 4 fractions obtained by partitioning the feature set on: good, suspicious, weak and bed features. This partitioning is based on local characteristic of the NSPF (degree of monotonocity). The results show that this local NSPF information is useful. Thus, 38 features selected as good give 93.06%, 91.42% performance for the learning and the test set respectively, that is significantly better then the results obtained by 38 individually best features which give 91.46%, 89.56% performance.

selection	performance %		performance (NSPF) %	
84 (F-{non-monotonic})	95.58	94.11	95.90	94.40
38 (good)	93.06	91.42	91.46	89.56
20 (suspicious)	83.67	81.38	88.60	86.81
26 (weak)	85.69	82.58	90.55	88.65
32 (bed)	87.48	85.37	91.46	89.56
58 (good+suspicious)	94.49	92.92	94.17	92.26
58 (weak+bed)	92.40	90.18	94.17	92.26

Table 3.2 *NSPF feature selection - 2.*

The third table shows the results obtained by the *Sat*(·) function where the goal was to select 60 features. The control results on each 10 selected features and final results are encouraging and illustrate potential usefulness of the search procedure based on *Sat*(·) function. Indeed, by growing of the number of the features, the performance become closer to the corresponding set of individually best features.

selection	performance %		performance (NSPF) %	
10 (Sat(·) function)	82.44	80.76	77.32	75.46
20 (Sat(·) function)	91.23	89.51	88.60	86.81
30 (Sat(·) function)	93.05	91.36	90.68	88.85
40 (Sat(·) function)	93.75	92.36	93.06	90.92
50 (Sat(·) function)	94.66	93.10	94.29	92.48
60 (Sat(·) function)	95.27	93.58	94.59	92.90

Table 3.3 *NSPF feature selection -3.*

The dimensionality reduction is a very difficult problem. Frequently, the discriminatory information is encoded in a very complex manner that it is practically impossible to reject all irrelevant and to keep all relevant information.

We presented some strategies for FS based on individual feature importance, considering the individual feature evaluation as a preparation for collective FS. The proposed strategies are classifier independent and can be used in all situation where we dispose of individual feature importance. The search based on the *Sat*(·) function can be considered as an efficient sub-optimal search procedure. It is an alternative to the search procedures such as branch and bound approach [Narendra and Fukunaga, 1977] or more sophisticated floating point forward and backward searches [Pudil et al., 1994] where it is computationally impossible to use more complex optimality criterion (like classifier performance), even for small number of features.

The proposed methods for FS based on the NSPF and obtained results impose two important questions:

- How much is the ordering of the features according to individual importance really useful in different search procedures?

- How to reduce the biasdness of the NSPF?

It is obvious that the NSPF is very biased that makes its usage very difficult. So, the future work can go into two general directions, finding other search procedures and reducing the biasdness by examination of various optimality criteria for construction of the NSPF, including different NN architectures and some statistical criteria.

Chapter 4

Structural Features for Hand-printed Cyrillic Character Recognition

For the task of handwritten recognition in this Chapter we show a structural features set definition acquisition and extraction on the implementation of the set of Cyrillic hand-printed capital characters. The base consist of segmented characters and contains 300 samples.This means that we will not be interested in details of the possible phases of word contextual analysis, or other word based techniques. The approach that will be developed in this thesis is an off-line recognition approach. The task of recognition is performed on the scanned image off-line, and the proposed phases and procedures act only on the scanned digit image without any additional information. It means that no on-line information is available about the way the character has been drawn, and no time parameters of the evolution of the character line is available. It is clear that in the on-line mode, various information of completely different nature can be used for the recognition process [Bengio et al., 1995]. The disadvantage of the on-line approach is the necessity of the supplement material like light pen or similar for the task of tracing the line generation.

The implementation of our ideas for the cooperation between different approaches in the pattern recognition starts with an application for Cyrillic Character Recognition. Namely, on the basis of structural approach for defining and manipulating the features, we build a Bayes error estimator for determining the discriminative power between the characters treated individually and in corresponding clusters. In this chapter the main aspects of the structural features definition and acquisition will be treated (4.1). Further on, the implemented clustering techniques will be showed (4.2), and we will close our discussion on this part of the implementation with the results obtained by the Bayesian error estimator.

4.1 Definition and Acquisition

The basic idea of the approach, which was taken for choosing this set of features, lies in the concept of syntactic pattern recognition. Namely, the principle that complex forms could be recursively described by the interrelation of simpler ones [Fu, 1982, Pavlidis, 1980], is the basis of syntactic pattern recognition techniques. However, the pure Syntactic pattern recognition, based only the relation between patterns and grammars, the semantic of the grammar, the expressive power of the grammar and grammatical inference has many drawbacks when implemented alone [Tanaka, 1995]. Moreover, it has been shown that a promising approach could be the one which introduces the features of nature (structural or statistical) to a classifier which is of the "opposite" nature. Heutte and colleagues introduce a set of structural and statistical features in a statistical classifier for general character recognition [Heutte et al., 1996].

Besides the aspects of the classification procedure which will be established later in this work, we consider that even on the feature generation (construction) level, for the handwritten recognition task an appropriate choice should include the syntactic, structural oriented type of features. We claim on the particularity of syntactic approach, even on the level of feature construction, to enhance not only the characteristics that will lead to correct recognition of the object, but also the characteristics that will distinct the object from the objects belonging to other classes. This approach is very popular especially in the fields of image analysis, where a hierarchical construction of the pattern can be established on the bases of interrelations of simpler sub-patterns. This is specially interesting, where the occurrences of the objects belonging to a same class can have a rich variety of appearances, which is obviously the case in handwritten character recognition. The general schema of syntactic pattern recognition system can be presented as shown in Fig. 4.1. The pure syntactic phases that are "borrowed" in our approach are double-bordered.

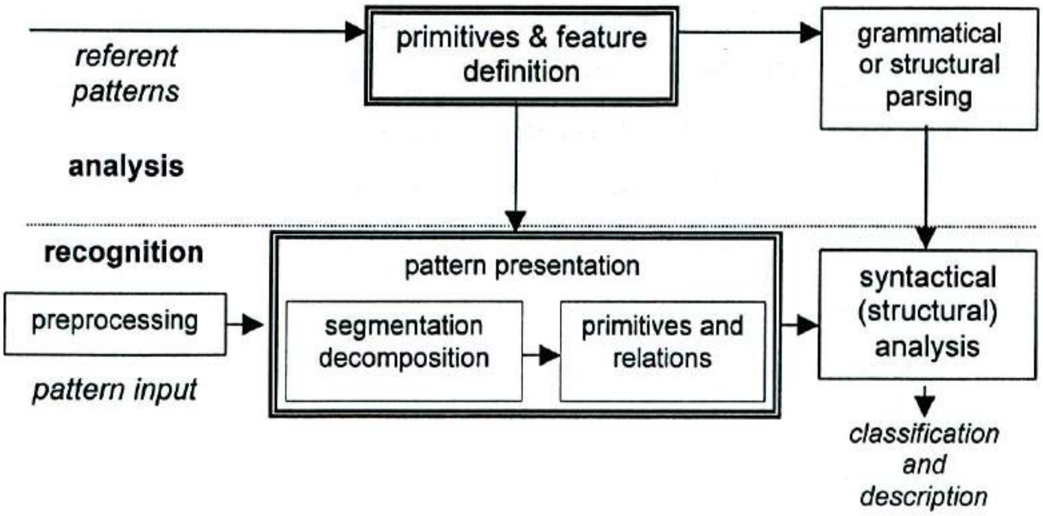

referent patterns
primitives & feature definition
grammatical or structural parsing
analysis
recognition
pattern presentation
preprocessing
segmentation decomposition
primitives and relations
syntactical (structural) analysis
pattern input
classification and description

Figure 4.1 *Syntactic pattern recognition system*

Hand-printed characters, as well as handwritten digits are essentially line drawings, i.e. one-dimensional structures in a two-dimensional space. We consider that the local detection of line segments is an adequate preprocessing and in our case it is the base of a structural features set.

In the classical syntactical approach, many authors have proposed different specialized grammars, for the description of different classes of patterns [Fu, 1982]. As an example we give the historical, but largely exploited approach is the work of Freeman about the encoding of digits with segments with constant length and declivity (Fig.4.2).

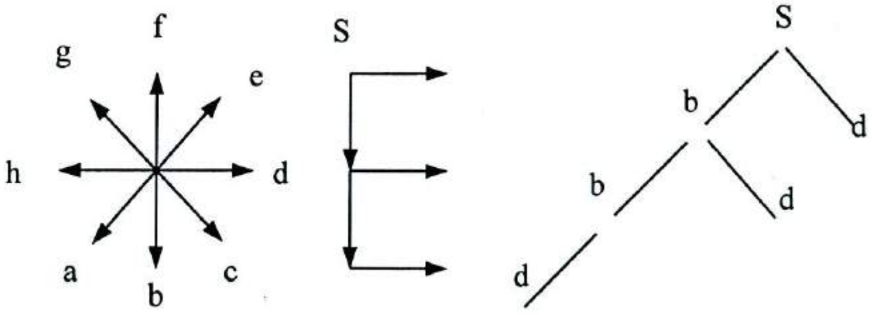


Figure 4.2 *Primitives, and the letter "E" with corresponding tree representation*

Tuissaint and Donaldson, stated that 1.) handwritten characters of roman alphabet are fairly well represented by their contours, and 2.) the confusions between letter category are well structured [Tuissaint and Donaldson, 1970]. In the approach the external line of the letter contour is taken into consideration, and the confusion issues are discussed for the case of digit recognition in Chapter 6. of this thesis.

Structured features are extracted by viewing the digit image by regions. However, the common way of extracting the shape based structural features include a image skeleton tracing [Amin et al., 1996]. The image skeleton tracing allows basically the introduction of a point-nature based features, like "end", "branch" or "cross" points. We have chosen a shape-nature based structural features, and the region viewing technique to be implemented allows additional structural information about the feature nature, i.e. containing the syntactical tree structure information of the character. The combination of the line-based features and the selected point-based features bears the same structural information as the image sub-region - shape based features. This information can be considered as an implicit one s in the former way, but in the later one there is no presence of such information.

The set of structural features is based on the information about shape primitives by which a letter or digit to be recognized is formed. After defining the set of shape primitives that are expected in the structure of the character (Fig.4.3.), for each primitive shape a region of the character image where the shape can be expected is defined. Thus, for various regions of the character image, we will have the information about what kind of shape primitive is present. This presence will be noted by numbers from the [0,1] interval, expressing the scale from sure non-existence to evident existence of the given shape primitive in the searched region. A shape-similarity procedure will be developed in order to estimate the similarity of the shape primitives presented in the image of the characters and the shapes from the pre-defined set primitives.

4.1.1 Shape Primitives and Regions

The set of shape primitives consists of straight-line strikes and arcs. The procedure which establish a shape similarity between those pre-defined shapes, and the ones to appear in the character image is scale-independent, so the length of line strikes and the rayon of the arc are not important.

The division of the Cyrillic letters in groups for the purposes of line-primitive definition can be made in these three sets:

1.) Letters of vertical and horizontal lines only: **Г, Е, Н, П, Т, Ц, Ч, Џ, Ш.**

2.) Letters containing slope lines: **А, Д, И, К, М, У, Х.**

3.) Letters containing arcs: **Б, В, Ж, З, Ј, Л, Љ, Њ, О, Р, С, Ф.**

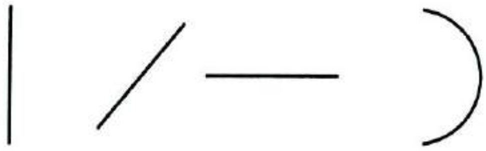


Figure 4.3 *Shape primitives*

An analysis of the possible regions of appearance of those primitives in the image of the character to be recognized is made. An example for Cyrillic capital letters is given in Fig.4.4.

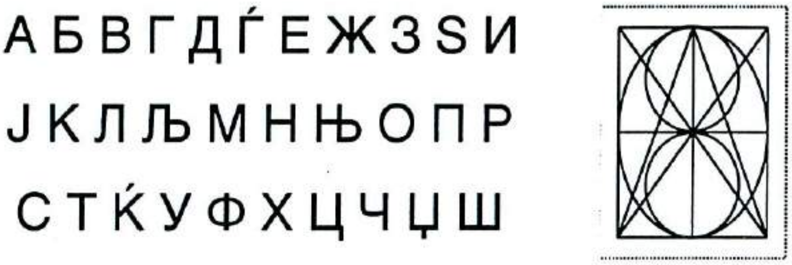


Figure 4.4 *Cyrillic printed capital characters (Macedonian Alphabet) and appearances of the primitives*

On the basis of these considerations, we define the regions of the image, where the primitives are expected and should be searched (Fig. 4.5.) The region segmentation is a common used idea in syntactic approaches [Basu and Fu, 1987].

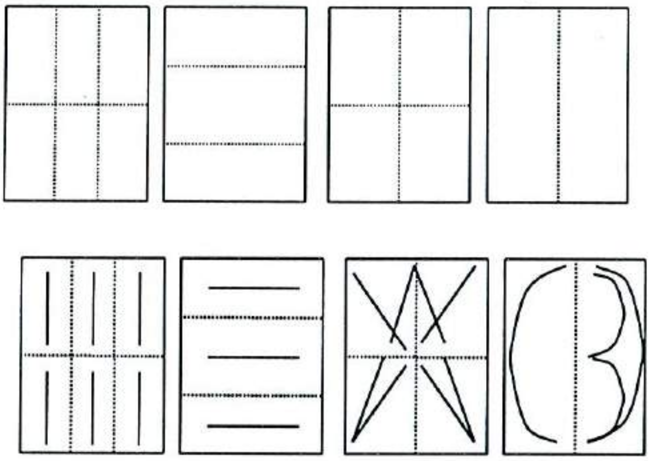


Figure 4.5 *The partition of the character image in primitive search regions and ther corresponding primitives*

We are going to investigate each of these regions of the character image in order to find out the appearances of the shape primitives. In fact, the procedure that will be explained further on, for each shape primitive will give a measure in terms of the probability of the presence of the corresponding primitive shape in the concerned region.

4.1.2 Feature Detection and Extraction

The aim of this phase is to find out the shapes appearing in these regions and to give an estimation of their nature, i.e. to detect which kind of shape primitive is presented, and where in the character image. The structural nature of this information lies in the very fundaments of the syntactical (structural) approach of the pattern recognition. Namely, the information about the presence of the shape primitives in the regions of expectation, in terms of the calculated similarity with the predefined shape primitives set of primitives allows a structural reconstruction of the character image [Radevski, 1995].

Structural descriptions assume that a complex pattern can be decomposed into simpler subpatterns and then characterized in terms of simple parts - primitives and of their relations. In a handwritten recognition applications, deformations on samples due to noise or distortion may cause an input sample to be quite different from the other samples of the same class. We address the problem of defining a distance between two given primitives, the predefined one and the primitive found in the digit image. Recently, Foggia and colleagues proposed a distance measure with two functions defined on the arc primitives and their relations [Foggia et al., 1999]. Having the radius of the arc as a description parameter, this approach embeds the lines from straight lines to closed circle. The approach of incorporating the structural features in the description and recognition phases that based on no properly structural classes of descriptors or recognizers, has showed a renewed actuality in the recent research works [Lou et al., 1999].

Our approach is based on no definition of classes of line primitives regarding the similarity measure to be established between two line primitives; from the two line primitives to be examined we extract some information, we make a description, that will permit establishing the relation of similarity between the two lines. The first part of this procedure is the *detecting the line* presented in each of these regions (Fig. 4.6.). Once the existing shape found in the image region is detected, a similarity criterion will be established in order to obtain the measure of the proximity of the found shape with the predefined ones. This information, presented as a number between 0 and 1, for the similarity of the line-shapes found and the predefined ones, will build up our set of structural nature features, or simply, *structural features*. As a consequence, no single existence - nonexistence information will be obtained, but a established likelihood measure will give the level of similarity of the found shape primitive with the predefined one, expected in the corresponding region.

The line presented in the character image in a specific region will be represented by a set of representative points. Those points are found as intersections with the a ray of parallel lines (Fig. 4.6.). Liou and Yang [Liou and Yang, 1996] tackle the problem of handwriting recognition where the strokes comprising the letters might be thick. Many algorithms for handwriting recognition will reduce the handwriting to a skeleton using thinning algorithms. This has the drawback that parts of the letters are often distorted this way, particularly in intersections, joints and ends of characters. In addition spurious pixels may be created. We have considered the external edge of the drawn line as a representative line for the general shape to be examined.

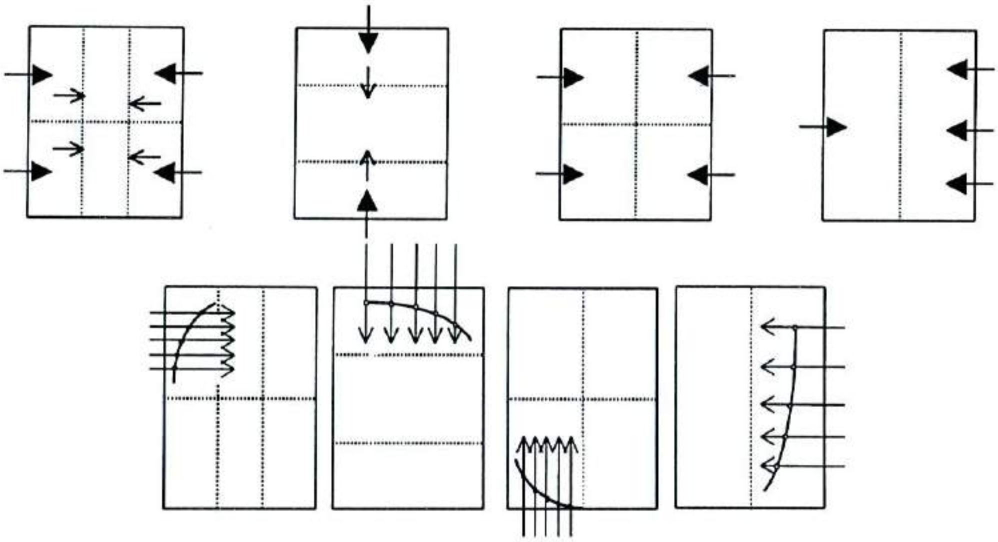


Figure 4.6 *Searching directions and determining shape control points*

The regions that have no external side are scanned from both sides and the highest similarity obtained is considered as final. The number of parallel lines is predetermined and fixed for each sub-region and is a parameter in the system. The lines are equidistant.

On this step we approximate the shape primitive detected in the image by the broken line passing by the extracted shape control points A_i (Fig. 4.7.).

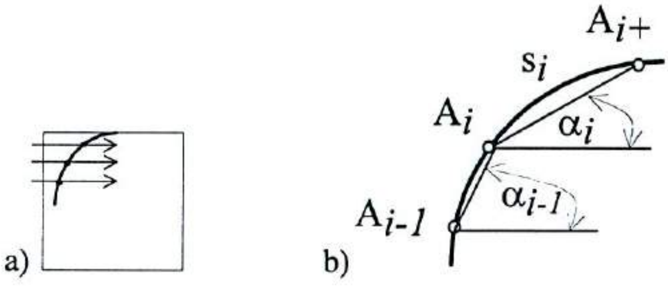


Figure 4.7 ***a)*** *Detected shape control points* ***b)*** *Line representative items.*

As an example, we show the middle horizontal line detecting in the case of one sample of the letter "A". The sub-region of searching does not have external sides perpendicular to the predefined parallel lines scanning. We scan the region from the both "possible" sides, and we take into consideration the "better" similarity with the line primitive found (Fig. 4.8).

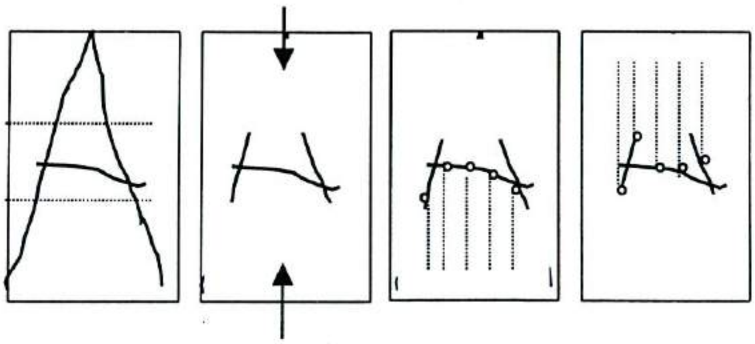


Figure 4.8 *An example of middle horizontal line primitive searching*

The obtained line, represented by the points A_i, is present as a 2xn matrix, where n is the number of detected control points minus one.

$$p = \begin{bmatrix} s_1 & s_2 & \dots & s_n \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \end{bmatrix}$$

The members of the first row are the lengths of the segments between the control points. Namely, the free-hand lines between the points A_i are approximated by the straight-line segments $s_i=[A_i, A_{i+1}]$ (Fig. 4.7.).

The second row consists of the angles α_i, between these approximated line segments and the horizontal *x*-axis (Fig. 4.7.). In order to define the line-similarity procedure, the corresponding items of the predefined line primitives are presented in the matrix *q*.

$$q = \begin{bmatrix} t_1 & t_2 & \dots & t_n \\ \beta_1 & \beta_2 & \dots & \beta_n \end{bmatrix}$$

It is clear that the angles $\beta_i=\pi/2$, i=1,n for the vertical line primitive (the first one in Fig. 4.1.), $\beta_i=\pi/4$, i=1,n for the second one, and $\beta_i=0$, i=1,n for the third one. The angles of the half-circle, the last line primitive in Fig. 4.3. have not constant values, i.e. they correspond to the angles describing the curve.

Having defined the matrices *p* and *q*, the one that describes the line found in the character image to be treated further on, and the one that represents the pre-defined primitive line, we will proceed to the definition of the line-similarity estimation function.

To ensure the scale free characteristic of the similarity-estimation procedure, the total length of the lines to be examined is scaled between 0 and 1.

$$s_i' = \frac{s_i}{\sum_{i=1}^{n} s_i} \qquad t_i' = \frac{t_i}{\sum_{i=1}^{n} t_i}$$

At this step, the representation matrices has the following form:

$$p=\begin{bmatrix} s'_1 & s'_2 & \dots & s'_n \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \end{bmatrix} \qquad q=\begin{bmatrix} t'_1 & t'_2 & \dots & t'_n \\ \beta_1 & \beta_2 & \dots & \beta_n \end{bmatrix}$$

The idea of the similarity estimation function between these two lines is to assume having an angle-information on same distances from the starting points of the lines to be examined. Further on we will proceed on reorganizing the first row data (distance data), first by replacing the lengths with cumulative lengths from the line endpoints, and afterwards to place the angle values of each line bellow the corresponding distance date from the first row.

Replacing the length data s_i and t_i, with the cumulative lengths from the line endpoint, gives this form of the representative matrices:

$$p=\begin{bmatrix} \sigma_1 & \sigma_2 & \dots & \sigma_n \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \end{bmatrix} \qquad q=\begin{bmatrix} \tau_1 & \tau_2 & \dots & \tau_n \\ \beta_1 & \beta_2 & \dots & \beta_n \end{bmatrix}$$

where $\sigma_i = \sum_{k=1}^{i} s_k$ and $\tau_i = \sum_{k=1}^{i} t_k$.

The distance measures σ_i and τ_i, i=1,n, will be ordered in increasing order, and will replace the distance date from the first row of the representative matrices:

$$\xi_i \in \{\sigma_1, \dots, \sigma_n, \tau_1, \dots, \tau_n\}, \quad \xi_i \le \xi_{i+1}$$

Now we are ready to place the corresponding angle measures of each of the matrices bellow the distance ξ_i that corresponds to the place where these angles were found. After placing the angles following the procedure

$$\alpha_i' = \begin{cases} \alpha_1, & \xi_i \le \sigma_1 \\ \alpha_k, & \sigma_{k-1} < \xi_i \le \sigma_k \end{cases} \qquad \beta_i' = \begin{cases} \beta_1, & \xi_i \le \tau_1 \\ \beta_k, & \tau_{k-1} < \xi_i \le \tau_k \end{cases}$$

we obtain the following form of the matrix representation of the two lines:

$$P = \begin{bmatrix} \xi_1 & \xi_2 & \cdots & \xi_{2n-1} \\ \alpha_1' & \alpha_2' & \ldots & \alpha'_{2n-1} \end{bmatrix} \qquad Q = \begin{bmatrix} \xi_1 & \xi_2 & \cdots & \xi_{2n-1} \\ \beta_1' & \beta_2' & \ldots & \beta'_{2n-1} \end{bmatrix}$$

The similarity measure between these two lines will be

$$d(P,Q) = \frac{\sum_{i=1}^{2n-1} |\alpha_i' - \beta_i'|}{(2n-1)}$$

Example:

We will show the evaluation of the similarity measure between the lines F_1 and F_2 shown hereby (Fig.4.9).

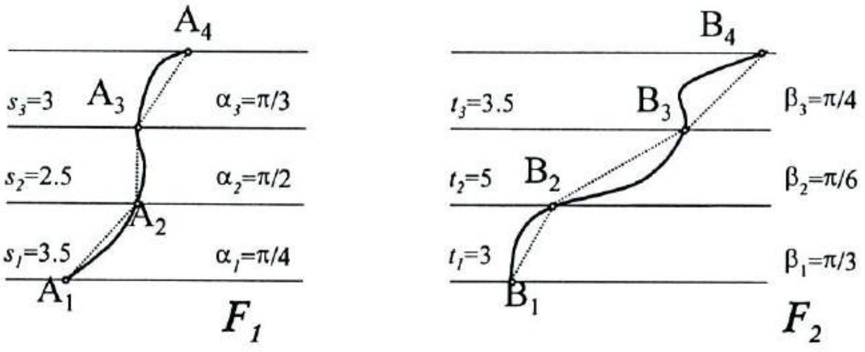


Figure 4.9 *The treatment of two lines*

We determine the corresponding line representative points A_i and B_i, i=1,4 of the lines F_1 and F_2 by intersection of the shapes with a set of parallel lines. The obtained broken lines A_i, i=1,4 and B_i, i=1,4 are represented by the matrices p and q, respectively, with the segments lengths in the first row, and the corresponding angles with x-axis, in the second row:

$$p = \begin{bmatrix} 3.5 & 2.5 & 3 \\ \pi/4 & \pi/2 & \pi/3 \end{bmatrix} \qquad q = \begin{bmatrix} 3 & 5 & 3.5 \\ \pi/3 & \pi/6 & \pi/4 \end{bmatrix}$$

Firstly, we normalize the total length of the broken line to 1:

$$p'=\begin{bmatrix}3.5/9 & 2.5/9 & 3/9\\ \pi/4 & \pi/2 & \pi/3\end{bmatrix} \qquad q'=\begin{bmatrix}3/11.5 & 5/11.5 & 3.5/11.5\\ \pi/3 & \pi/6 & \pi/4\end{bmatrix}$$

We replace the lengths with cumulative lengths, i.e. with the distances from the line end-point to the respective control point:

$$p''=\begin{bmatrix}3.5/9 & 6/9 & 9/9\\ \pi/4 & \pi/2 & \pi/3\end{bmatrix} \qquad q''=\begin{bmatrix}3/11.5 & 8/11.5 & 11.5/11.5\\ \pi/3 & \pi/6 & \pi/4\end{bmatrix}$$

From the two lists of cumulative lengths we construct an unique list where the distances from the shape end-point are in increasing order, regardless from the shape they came, F_1 or F_2. The final form of the two matrices p and q, consists of assumption of the angle information on the same places for the two lines. Namely, although we've found an angle π/4 at the distance 3.5/9 from the F_1 shape end-point, we will consider that we have been found the same angle at the 3/11.5 distance too, because it is the point from the F_2 shape appearing before the 3.5/9-point in the increasing ordering of the distance points.

$$P=\begin{bmatrix}3/11.5 & 3.5/9 & 6/9 & 8/11.5 & 1\\ \pi/4 & \pi/4 & \pi/2 & \pi/3 & \pi/3\end{bmatrix}$$

$$Q=\begin{bmatrix}3/11.5 & 3.5/9 & 6/9 & 8/11.5 & 1\\ \pi/3 & \pi/6 & \pi/6 & \pi/6 & \pi/4\end{bmatrix}$$

In that manner, we've ordered the angle-values of each of the shapes F_1 and F_2, at one unique distance vector obtained by ordering of the union of the corresponding distance values of the shapes F_1 and F_2 (Fig. 4.10.).

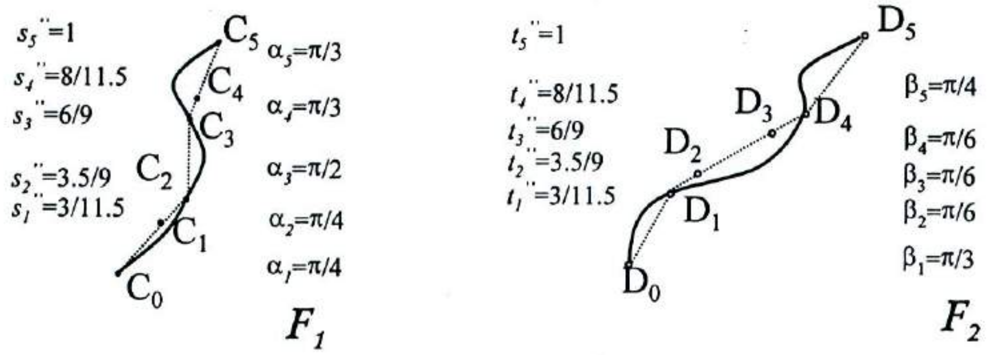


Figure 4.10 *Final approximation of the two free-hand lines and respective distance - angle values*

In fact, if in the beginning, the shapes F_1 and F_2, were approximated by the straight lines between A_i, i=1,4 and B_i, i=1,4 respectively, now they are approximated by the straight lines through C_i, i=0,5 and D_i, i=0,5. The similarity measure for these two shapes is

$$d(P,Q)=\frac{\sum_{i=1}^{2n-1}|\alpha_i'-\beta_i'|}{(2n-1)}=\frac{\sum_{i=1}^{5}|\alpha_i'-\beta_i'|}{5}=0.75$$

On Fig. 4.11. we show an example of a set of shape pairs for which this similarity measure will give the same results.

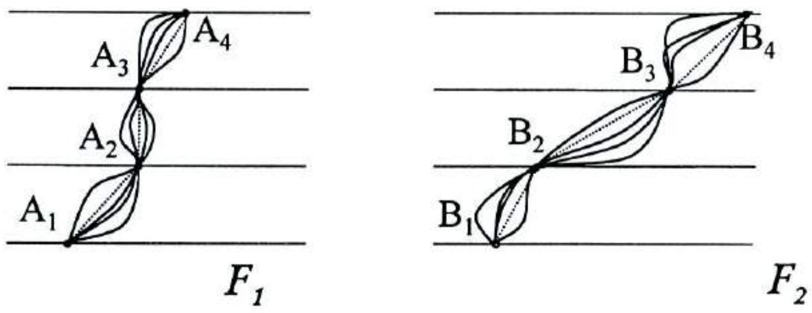


Figure 4.11 *An example of a set of equivalent lines before approximation*

In fact, the process of length normalization and angle placing for these two shapes gives the equivalence between, for example, these two set of shapes (Fig. 4.12).

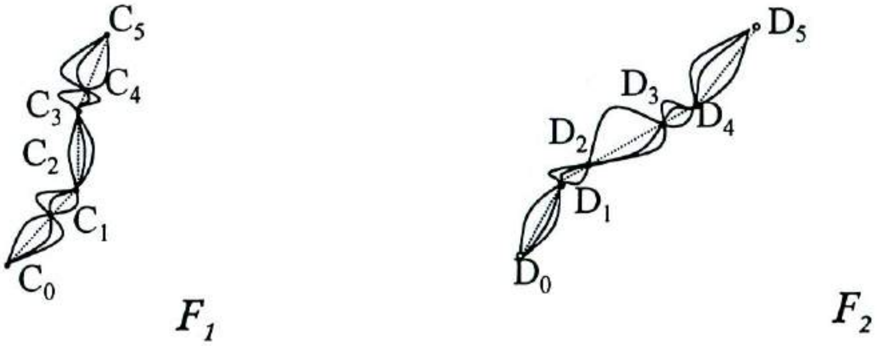


Figure 4.12 *The example of a set of equivalent lines after the approximation*

The procedures showed in 4.1.1. and 4.1.2. are easily transformable on other type of handwritten (or other drawing) image recognition. Namely, a slight modification of the proposed approaches will be implemented for the structural features part of the Handwritten digit recognition studied in chapter 5.

4.2 Clustering

In order to establish all preconditions for a multilevel recognition system, and estimating the Bayesian error on each level, we will examine the cluster structure of the obtained data structure after the feature extraction phase. Thus, we will be able to define the tree structure of the further decision procedure, as well as to implement the Bayesian error estimator on each level of the presumed decision tree.

The term clustering is used for the techniques of data organization by abstracting the data structure; by grouping or by generating the hierarchy of groups [Jain and Dubes, 1988]. In the sense of the training phase used in the pattern recognition terminology, the clustering methods can be considered as a unsupervised training. The clustering will be implemented for all training examples of the handprinted Cyrillic characters base, where each character will be represented by the corresponding 21-component feature vector. The very base of each clustering technique are the distance matrices. As a distance measure we've used the Euclidean, Quadratic Euclidean, Manhattan, Chebishev distance, and 1-Pearson r measure [Milosavljevic and Radevski, 1996].

What we search in this phase is the hierarchical clustering of the referent training set of feature vectors. Hierarchical clustering is the procedure of transforming the distance matrix in a sequence of nested partitions. If **H** is the set of objects x_i to be clustered, a *partition* **C** of the set **H**, will be a division of the set **H** in a sequence of subsets such as:

$\mathbf{Ci} \cap \mathbf{Cj} = \varnothing$ and

$\mathbf{C_1 \cup C_2 \cup \ldots \cup C_m = H}$

A partition **B** is nested in partition **C** if and only if each element from the set B is e real subset of the components of **C**. We can say also, that the set **C** is obtained by concatenation of the elements of **B**. For example, for the set of objects

$\mathbf{H}=\{x_1, x_2, \ldots, x_{10}\}$ and the two clusters **C** and **B**:

$\mathbf{C}=\{ (x_1, x_3, x_5, x_7), (x_2, x_4, x_6, x_8), (x_9, x_{10}) \}$

$\mathbf{B}=\{ (x_1, x_3), (x_5, x_7), (x_2), (x_4, x_6, x_8), (x_9, x_{10}) \}$

we say that **B** *is nested in* **C**, and neither is nested in **D**:

$\mathbf{D}=\{(x_1, x_2, x_3, x_4), (x_5, x_6, x_7, x_8), (x_9, x_{10})\}$.

We consider the Euclidean distance between the vectors $\mathbf{x}_i$ and $\mathbf{x}_k$ by

$$d(i,k) = \left[\sum_{j=1}^{n} (x_j^i - x_j^k)^2 \right]^{1/2}$$

and we perform the complete linkage clustering on the training set of feature vectors. Here, complete linkage refers to the nature of the intermediate procedure from the distance matrice to the hierarchical cluster. Namely, on the basis of the distance matrice a threshold graphs are generated, and are the base of further hierarchical clustering. Now, if we search for a maximal complete graphs we have a complete linkage clustering, if we search for a maximal connected graphs, we obtain the single linkage clustering. Our experiments [Radevski, 1995] showed that the best clustering, in terms of least Bayesian error, is the complete linkage clustering, which for the Cyrillic handprinted characters is shown on Fig. 4.10.

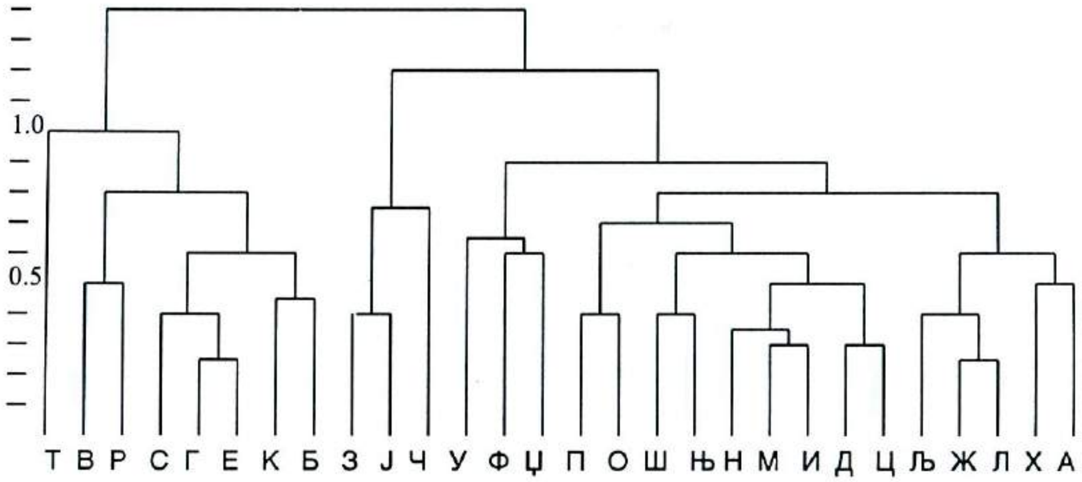


Figure 4.13 *Tree diagram for 28 cases; Complete linkage; Eucledian distances*

The obtained clustering scheme corresponds to the intuitive likelihood clustering that a human can made, and is a confirmation of the feature sets defined for the description. The clusters of Г Е and С, В Р and Т, П and О, М, И and Н, seems to be very natural clusterification of the Cyrillic letters.

4.3 Bayesian Error Estimation

The aim of this phase is measuring the discriminative power of extracted attributes by non-parametric Bayesian error estimation. All conditions for a multilevel recognition will be meet only after determining compact subsets of characters with minimal Bayesian error.

The estimation of the probability of misclassification, i.e. Bayesian error probability, in the case where we use the estimators of parameters, can be done in different ways. We've used the PARIS software (Pattern Analysis and Recognition Interactive System) developed in the Institute of applied mathematics and electronics, Belgrade, Yugoslavia [Buturovic, 1991]. Non-parametric estimation of Bayesian error is an open research problem, in the theory and practice of statistical pattern recognition. If the posterior class probabilities $p(W_i|\mathbf{x})$ and prior probabilities $p(W_i)$ where known, Bayesian error is fully determined. It is clear that in practice we do not dispose of analytical expressions of $p(W_i|\mathbf{x})$; neither of $p(W_i)$.

The estimation of $p(W_i|\mathbf{x})$ is direct: $P(W_i)=N_i/N$, where N_i is the number of training items from the class W_i, and N is the total number of items in the training ensemble. So, we can resume the process of Bayesian error estimation in these two steps:

1. Conditional class-probabilities functions estimations $P(\mathbf{x}|W_i)$.
2. Classification according to Bayesian decision rule using the the conditional class-probabilities functions estimations and prior class probabilities.

Classification error obtained in the second step is considered as a final Bayesian probability of error [Fukunaga, 1985, Fukunaga and Hummels, 1987].

For the hierarchical clustering described in 4.2. we have calculated the Bayesian error estimation on different levels of the clustering dendogram. From the highest to the deepest level, the results are shown in Figures 4.12. to 4.15.

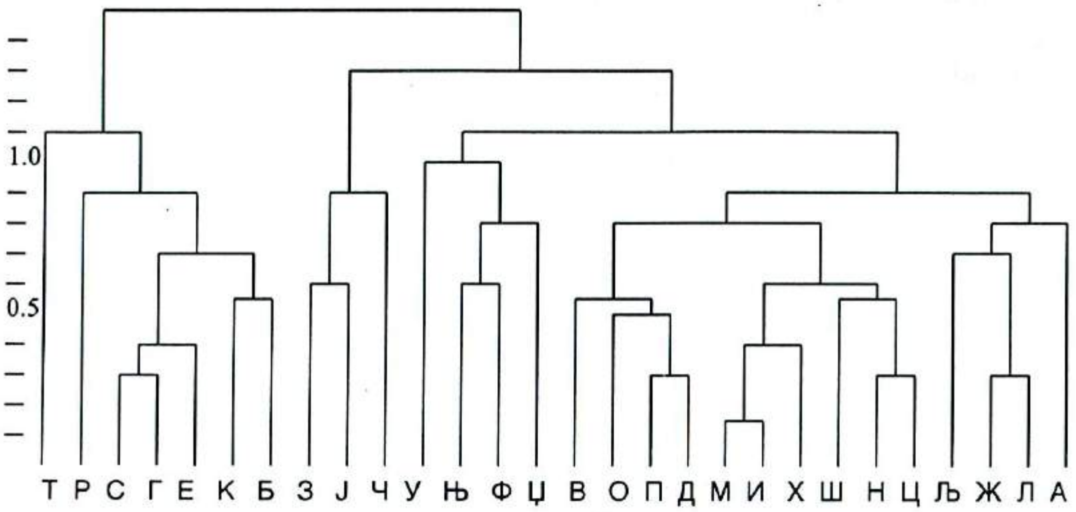


Figure 4.14 *Bayesian error estimation - Starting cluster position*

On the highest level we have the dendogram clustering scheme shown in Fig. 4.14.

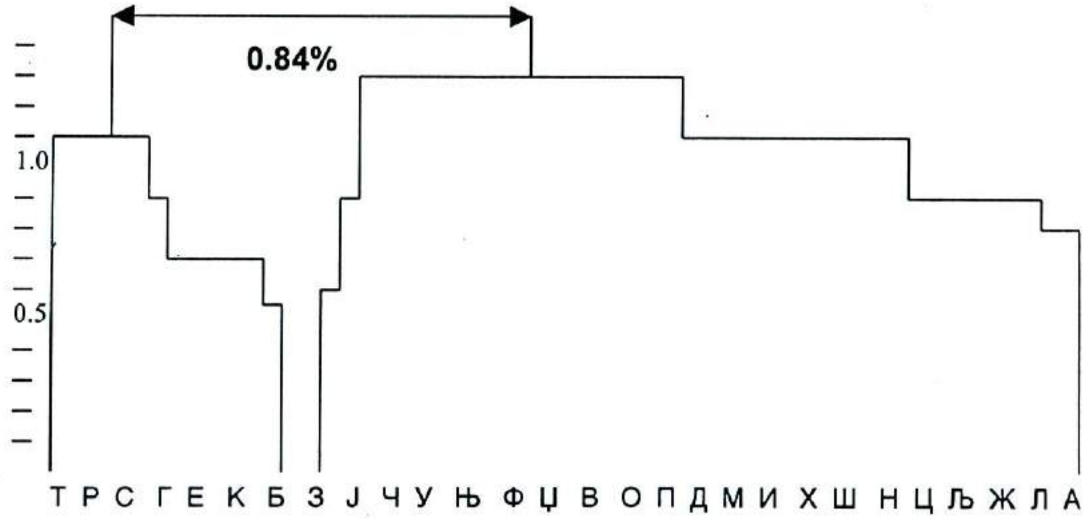


Figure 4.15 *Bayesian error estimation - Two classes partition*

Firstly we measure the Baysian error while deciding in which of the starting two clusters the unknown input letter belongs. The error at this stage is 0.84%.

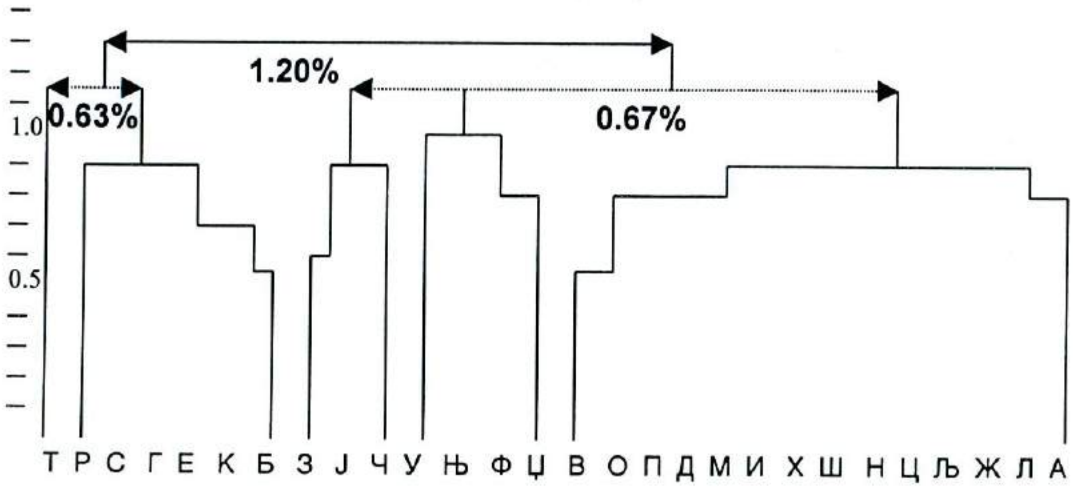


Figure 4.16 *Bayesian error estimation - Third stage partition*

On the next step (Fig.4.16.) we estimate the Bayes error while choosing among the five possible clusters for the input letter. The estimated error is 1.20%. It is obvious that the steps can be defined differently, in terms of how deep in the cluster dendogram we move for measuring the Bayseian error, at each step. Here we choose to show the results only of some characteristic "cuts" of the cluster dendogram.

At the same time we give the error values inside the main four primary clusters for this step, considering them as a clusters to be devised further. At this level the obtained values are 0.63% and 0.67% for the separability of the sub-clusters inside the two main clusters.

For the step of 6 cluster sets further on in the cluster dendogram, we show the results on Fig. 4.17. For the six cluster separation the error is achieves 6.45%.

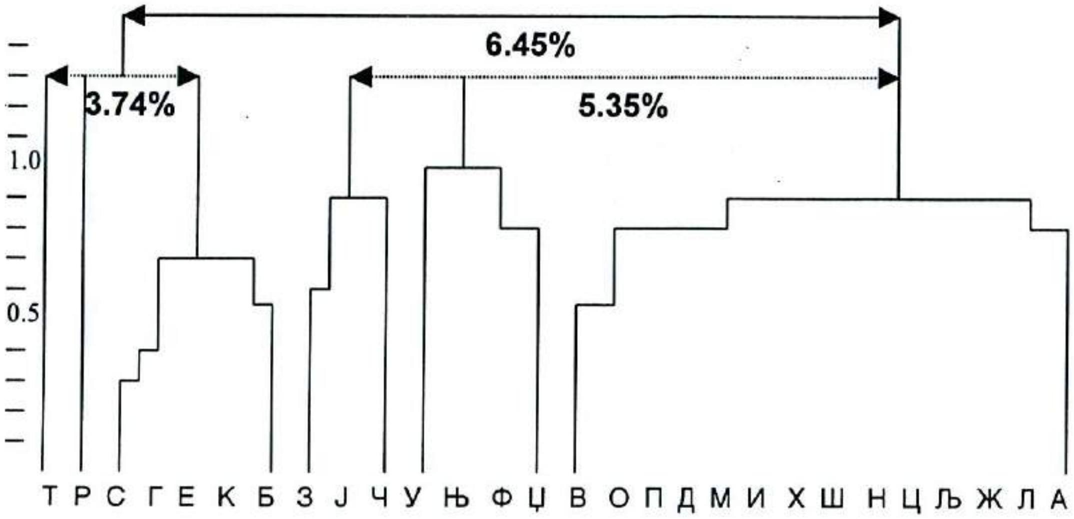


Figure 4.17 *Bayesian error estimation - Fourth stage partition*

On Fig.4.18. the value of Bayes error is shown considering directly each of the class letters alone. The value of 15.98% is the error to which we will have to face if we tackle the problem without using the clustering dendogram structure shown above, and proceed stepwise with the hierarchical decision steps that it provides.

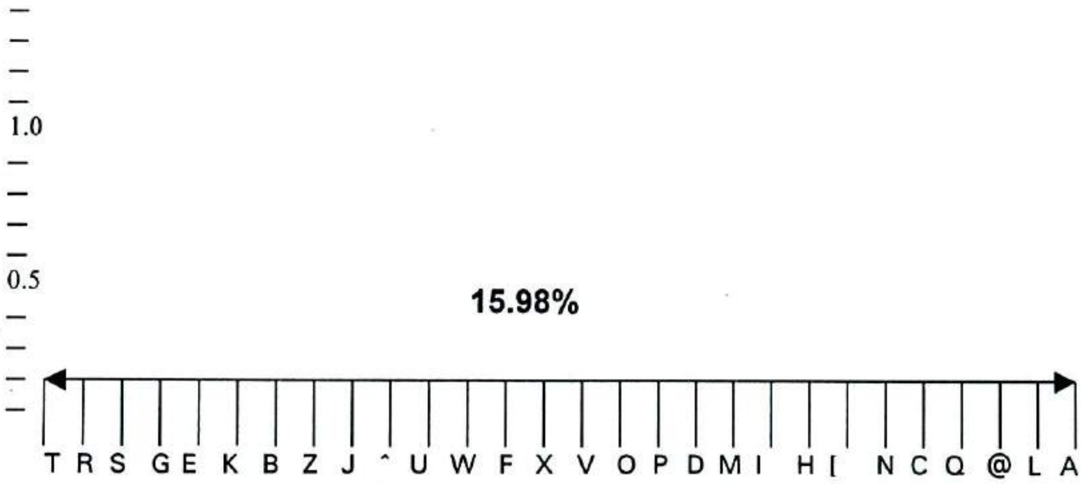


Figure 4.18 *Bayesian error estimation - Letter-class partition*

The measured Bayesian errors goes from 0.84% on the highest hierarchical level up to 15.98% on the lowest level. Thus, the task of recognition can be now based on simple decision rules on the top of the hierarchically decomposed character tree and more sophisticated rules on the bottom of division. Moreover, the measured discriminative powers and character subsets, open the possibility of acting inside the feature sets on each hierarchical decision level.

Chapter 5

Data Fusion: Incorporation of Structural and Statistical Features in Connectionist Digit Recognition System

D*ata fusion* is a formal framework in which are expressed means and tools for the alliance of data originating from different sources. It aims to obtain information of greater quality; the exact definition of "greater quality" will depend upon the application. In our case, we show a fusion of two kinds of measurement information, i.e. two set of features extracted from the character image.

The data base for this part of the experiments is the International NIST data base of handwritten digits [NIST, 1992]. The quantified information for fusion is spontaneously extracted from the digit image and is represented by two sets of multidimensional description *state vectors* (further on only *feature vectors*): statistical and structural features descriptors. Thus, the same digit image is seen twice: in the terms of detecting extracting statistical features and structural ones. So, in this phase we have a case of data fusion at *attribute level*. On the basis of the *measurements*, which in our case are the outputs of the digit image in 2-D on the pixel level, the properties of the digit image are expressed by their attributes seen from the two feature extractors.

The fusion of those two data types is performed by a neural network of multi-layer perceptron type. The neural network is the framework of the fusion of those data of different provenience and after the training and validation phase performs the classification task on the unseen digit sample.

5.1 System Architecture

The recognition system is constructed around a modular architecture of *features extraction* and *digit classification* unit. Preprocessed digit image is the input for the features extraction module, which outputs a set of extracted features (in the form of state vectors) used as inputs to neural network classifier (Fig.5.1). During the learning phase of the NN classifier a feature selection is performed to emphasize the importance of the reliable features [Radevski and Bennani, 1997].

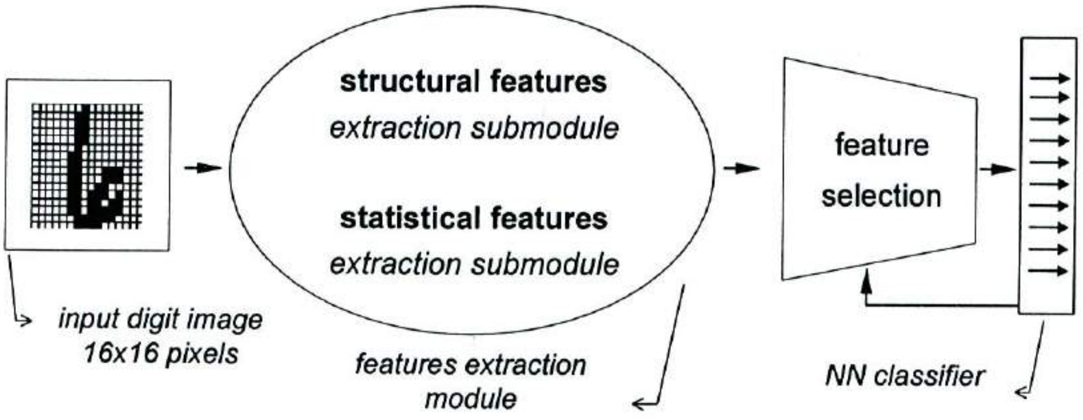


Figure 5.1 *System architecture*

It is clear that the known physiological mechanisms of the human visual perception and the actual level of the computing systems can not lead to a artificial recognition system which will literary imitate the human's one. Nevertheless the implementation of some concepts of the human vision lead to a construction of a robust recognition systems with remarkable performances. Fukushima's Neocognitron is a system which is inspired by the hierarchical structure of the visual system, where simple features are first extracted from a stimulus pattern, and then integrated into more complicated ones [Fukushima, 1988]. We proposed two set of features for describing the pattern (character: letter digit) in two different manners. The first one is the set of structural features, where the description of the "seen" image will be based on the existence / non-existence of the predefined primitives such as strokes, and arcs in the presented image, and the second one will be made up of statistical features describing the degree of filled regions in the pattern image.

Generally, this situates our approach in the group of features based approaches, approaches that are based on the construction of features sets in the wide sense of the word, contrary of the measure approaches where different measure based procedures for the recognition can be proposed to be performed on the pixel based image information, with an example for the later proposed by Simard and colleagues [Simard et al., 1992]. A comparison of the performance of several classifier algorithms, in terms of accuracy, training time, recognition time and memory requirements is given by Bottou and colleagues in [Bottou et al., 1994].

5.2 The Handwritten Digit Data Base

As data base for our experiments we used an extraction of NIST segmented handwritten digit data base. The digits images are in 128x128 pixels gray level format with real number pixel information ranging from -1 to 1. The total number of 23,898 digit images is divided in two groups: 17,952 samples for the training phase, and 5,946 samples for the test phase.

The digits from the original data base are rearranged in order to achieve that digits in the test set belong to different writers from those from the learning set. From the 128x128 gray-level format images, a 16x16 black and white are obtained, on which the smoothing and centralizing preprocessing techniques have been applied. Considering the nature of the recognition domain, we take the external outline of the digit primitive as the representative one for the digit shape itself. So, no skeletization or other technique for extracting the essential line shapes are performed.

We have the nearly same quantity of learning / test set appearances from the samples of the all 10 classes (Tab 5.1).

Class	Learning Set 17952	Test Set 5946	Total 23898
0	1860	606	2466
1	2026	670	2696
2	2750	594	2344
3	1895	622	2517
4	1714	556	2270
5	1801	516	2050
6	1726	591	2317
7	1878	613	2491
8	1783	589	2372
9	1785	590	2375

Table 5.1 *Composition of the digit data base set extracted from the NIST*

An example of the NIST handwritten digit data base is showed in Fig. 5.2.

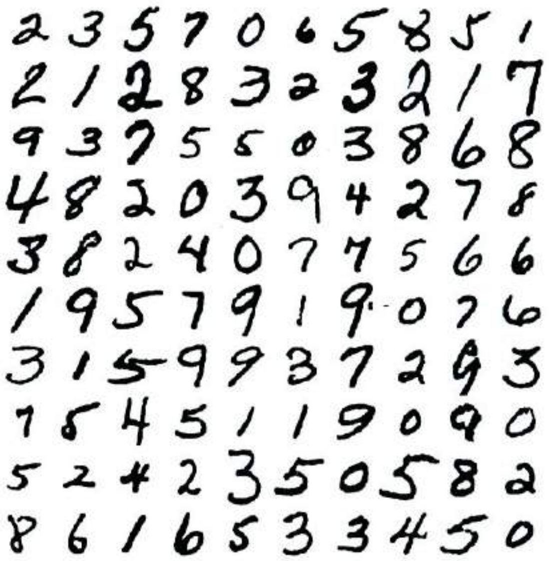


Figure 5.2 *Sample examples from the NIST data base*

5.3 Structural Features Definition and Acquisition

The structural features set is the first of the two feature set proposed for the digit image description in our work. The techniques for the feature detection and extraction follow the procedure proposed for the set of structural features in handwritten character recognition experiments showed in chapter 4. of this thesis. [Fu, 1982]. The structural features set is a domain dependent set. Its nature as well as the techniques implemented for the detection and extraction are strongly dependent of the nature of the objects to be recognized (handwritten digits). The possibility of the reconstruction of the character images, by a structural set features for example, ascertains that the proposed feature set captures the essential structural information from the image and approves their "structural" nature.

The proposed set of structural features is made up of 54 features. This is only the starting content of the structural feature set. We will se in the chapter of feature selection how the feature relevance is evaluated and how do we introduce the feature importance (or relevance) in the global classification system.

The first stage of creating the structural features set is defining the elementary sub-patterns to compose every possible digit appearance. It is clear that we propose a level of elementary sub-patterns in order to find a reasonable subset of the patterns from which one digit can be constructed. On the basis of our experience in the handwritten character analysis we have proposed the elementary sub-patterns showed in Fig. 5.3. We will search the digit image for these primitives twice: firstly on the original digit image orientation, and secondly on a digit image rotation for 90° , so the total number of line primitives - region conjunction is 54 - is the number of the elements in the feature vector.

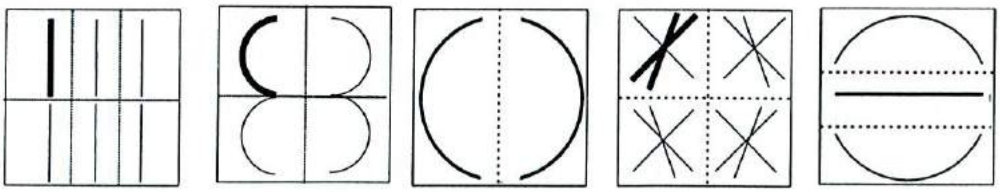


Figure 5.3 *Set of elementary sub-patterns and image sub-regions*

According to the proposed feature detection and extraction procedure, the definition of the set of the elementary sub-patterns, primitives (regarding the structural features set), is closely related to the division of the digit image in sub-regions. Namely, each elementary sub-pattern is seen as a sub-pattern which can appear in a pre-defined part (sub-region) of the digit image. The procedure of creating the structural features vector is based on searching the digit image for the pre-defined elementary sub-patterns. The components of the structural features vector are values of the "similarity" between the patterns present in the digit image, and those from the predefined elementary sub-patterns set.

The detection and the extraction of structural features is performed by dividing the image binary matrix into two, three four and six sub-regions depending on the primitive type whose existence is to be examined. The existing shape primitive in each of those sub-regions is compared with the referent, idealized primitives whose existence is expected in corresponding positions. As a consequence, no single existence - non-existence information is provided, but a likelihood criteria is developed giving the level of similarity of the found primitive with the expected idealized one. The evaluation of the similarity between the two shapes follows this procedure:

Each sub region is treated with a set of parallel lines from up, down, left or right side, predefined to search each sub-region. The direction rule for treating the digit image is "when possible - from outside, and orthogonal to the larger side of the sub-region". Sub-regions that have no external side are scanned from both sides and the highest similarity obtained is considered as final. The eventual existing line in the sub-region is characterized by its intersections with those parallel lines (Fig. 5.4).

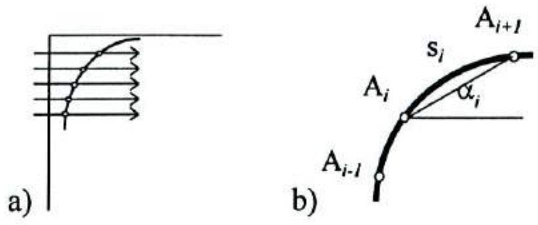


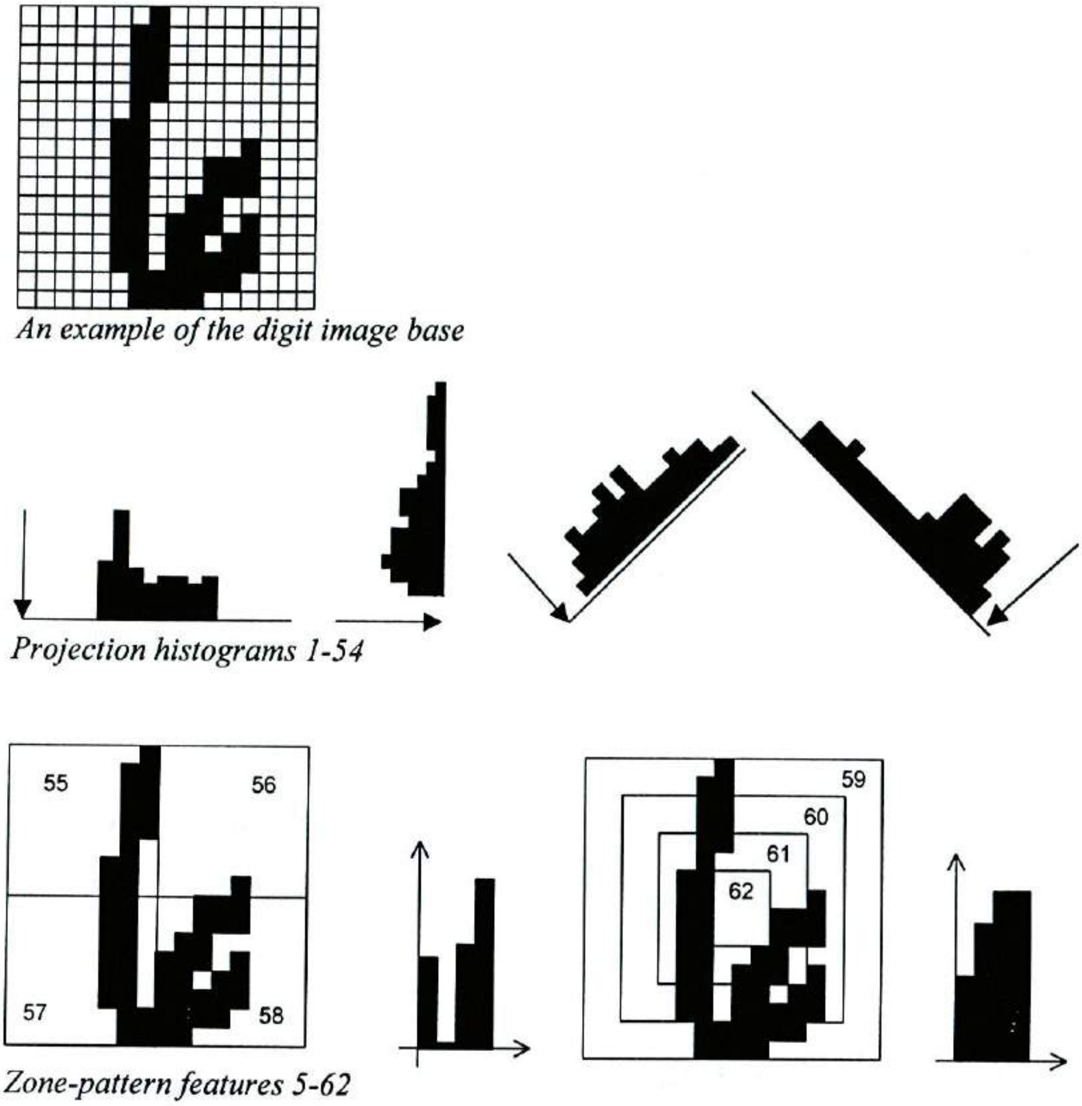
Figure 5.4 *a) Line representation by control points b) Line representative points*

For the proposes of our research we have made the search with five parallel lines, so on the basis of the maximum five detected points we should describe the found line. The line is describe by a 2x5 matrix, where in the first row we have the lengths of line segments between the control points s_i, while in the second row we have the corresponding angles between approximated line segments and the x-axe.

The structural features vector will be composed of 54 values of the calculated similarities between the pre-defined and found elementary line-primitives. The definition calculation of the similarity follows the procedure described in Chapter 4.1.2. of this thesis.

5.4 Statistical Features Definition and Acquisition

The second set of features considered in this work is the set of statistical features. The set of 62 features gives the statistical, pixel-based information of the digit image, in the terms of density of the lit pixels in various regions of the image. The numerical values of these features are the result of the percentage values of the projection histograms and the zone-pattern features (Fig.5.5.). The first 54 statistical features are obtained from the projection histograms issued from the vertical (16), horizontal (16) and two diagonal projections (22). The last 8 features from this group are obtained from the zone-pattern features showed in Fig. 5.5.



An example of the digit image base

Projection histograms 1-54

Zone-pattern features 5-62

Figure 5.5 *The projection histograms and zone-pattern features*

Each of the numerical values of the 62 statistical features represent the filled up percentage of the corresponding projection histogram and zone-pattern feature. So, as a result of this phase, we have a 62-component vector of the scaled values between 0 and 1 of the statistical features which describe the digit image. This kind of features has been exploited in many proposed recognition systems in a different forms and definitions. Burel and colleagues called them *oriented profiles* (the *projection histograms* in our case)in the group of *metric features*, and simply *statistical features,* the ones that are refereed as a zone-pattern features in our case [Burel et al., 1992].

5.5 Data Fusion Through a NN Classifier

The data fusion part of this phase of the work concerns the fusion of the attribute level of the extracted digit image information. Namely, we have obtained two set of attributes - the scaled values of the correspondent structural and statistical features. The attributes are numerical descriptions of the phenomena of the existing features. The first set values are the similarity evaluation for the existence of the structurally defined elementary line primitives, and the second one, the percentage values of measured filled parts and sectors of the digit image. Each of those sets represents a digit image descriptor in the terms of extracted features. A recognition system can be constructed that will "look" at the digit image from each of these two proposed "points of view". Further more, the very principles of these two "points of view" are based on different ways of seeing the image. The question is can we make these two approaches and the obtained observed information to work together.

Later in this work (Chapter 6.) we will develop a decision fusion system and study the problem of the data fusion with much more details. At this stage our task is to enable the fusion of these two attribute level information in order to study their complementary capacity. We claim that, having on mind the different nature of the information provided by each of these two "observers" it should be possible to make them "work" together in order to achieve higher recognition performances.

5.5.1 NN Classifier Architectures

A very natural way to fusion the data from the two sets of features is to present them as an input in a neural network classifier and to train the classifier on the entries provided simultaneously by the two sets, and this without any distinction for the differences or importance of the data provided on the input. It is clear that this stage of investigations has many possible directions of research, but lets consider a simple multilayer perceptron (MLP) NN architecture as a starting point for the study.

We have in mind that it has been shown in various papers [Hornik, 1989] that MLP's with a single hidden layer are universal classifiers, in the sense that they can approximate decision surfaces of arbitrary complexity, given that the number of neurons in the hidden layer is large enough (there is no simple rule which indicates how many hidden units are required for learning a given task). Idan and colleagues give a comparative study of neural networks and non parametric statistical methods for off line character recognition [Idan et al., 1992]. The methods are compared on two data bases consisting of 1000 and 15456 patterns respectively, which are the statistical representation of the zip-code digits distribution in a particular French post-office. Auger and colleagues published are study comparing a back-propagation-like network integrating feature selection notions introduced in a class of neocognitron class networks with a supervised learning algorithm based on Kohonen's self-organizing feature maps [Auger et al., 1992].

The way the classification phase is considered falls within the *supervised learning* paradigm. This task orientation assumes that we have been given a set of training examples (also called training *instances*), which are customarily represented by feature vectors; in our case regrouped in two feature sets. Each training example is labeled with a class target, a member of a set of 10 digit class labels. The goal of supervised learning is to predict tha class labels of examples that have not been seen, and to do so accurately and efficiently. We use the term *recognize* simultaneously with *classify*.

Firstly, we investigate the discrimination power of each of the feature set separately, through two separate MLP NN classifiers which architecture is shown in Fig. 5.6.

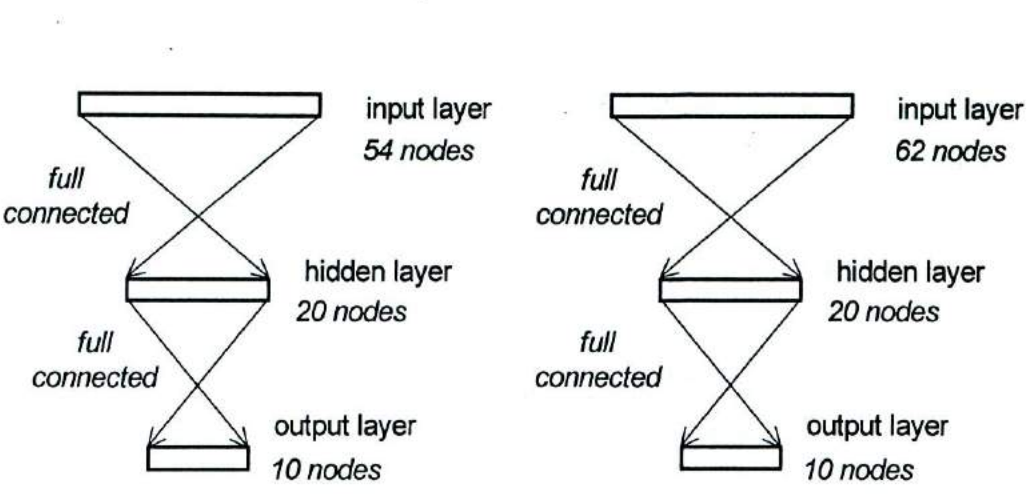


Figure 5.6 *The two classifiers on separate feature sets*

On the basis of these first investigations, and having them as a refereeing results, we propose a fusion on the input data through NN classifiers as follows.

We show three NN based classifiers architecture in this stage of our study. The first attempt of fusion at this level consists of considering the simple MLP NN classifier of 116 = 54 + 62 input nodes, one hidden-layer of 35 nodes and 10 - nodes output layer, one for each of the ten digit classes to be recognized. We call this architecture a Structural - Statistical Features based recognition system (SSF1) and its architecture is showed in Fig. 5.7.

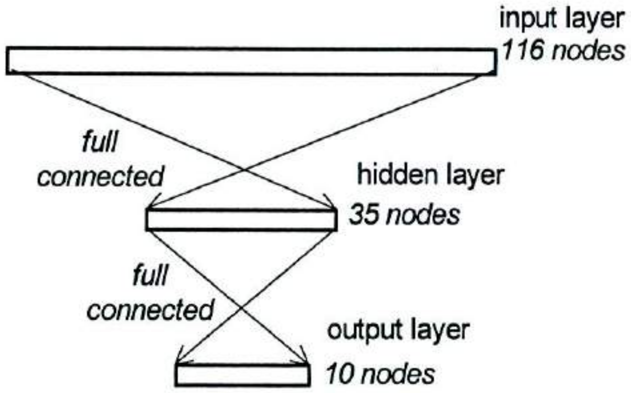


Figure 5.7 *The SSF1 MLP NN classifier architecture*

The second architecture is a modified version of a previous one. Namely, we will see in next section a way of implementing a features selection technique based on the individual importance of the features. On the basis of the implemented feature selection technique, we can introduce the obtained feature importance together with the feature value on the input level of the classifier. In that manner we obtain a modified version of the SSF-1, the architecture SSF-2 which is shown in Fig. 5.8.

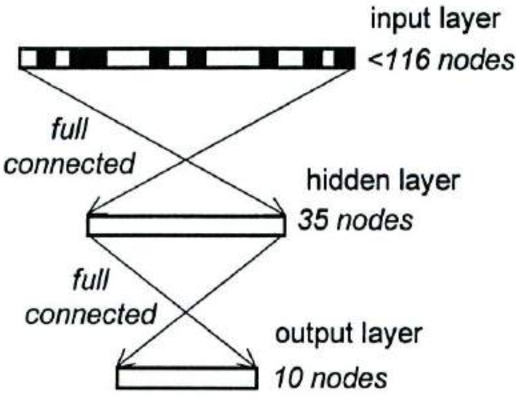


Figure 5.8 *The SSF2 MLP NN classifier architecture*

The last architecture studied in this section will be an attempt to fusion the input data only after the hidden layer. Precisely, the data from each of the feature sets will be input on separate hidden layer in full connected manner. Afterwards, both hidden layers are fully connected to the unique 10-nodes output level. This architecture is called SSF3 and is showed in Fig. 5.9.

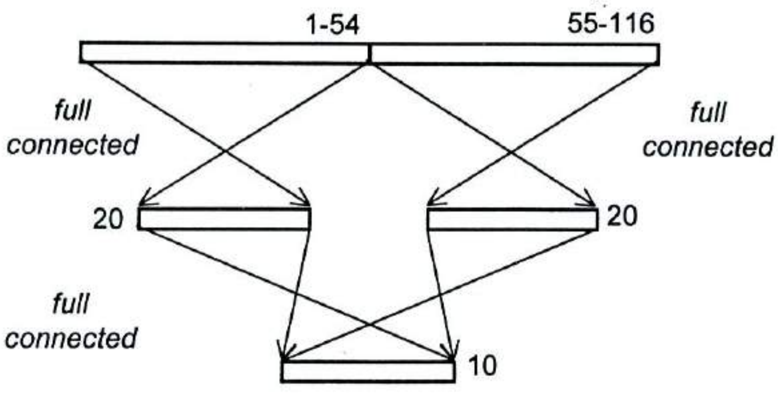


Figure 5.9 *The SSF3 MLP NN partly connected classifier architecture*

This classifier has less nodes and half as much connections as it has the SSF1. A side an important complexity diminution, the attempt of this architecture is to offer a fusion on a different level, i.e. to allow the training phase to accommodate the connection weights in a different manner then the previous ones. The results of those implementations will be showed in section 5.5.3.

5.5.2 Feature Selection

The way from the input space measurements to the set of optimal features, leads through the feature selection phase. We have implemented a supervised feature selection technique based on Optimal Cell Damage (OCD) method [Cibas et al., 1994]. The features are selected one by one, according to the obtained error rate of the classifier being used, in our case, the full connected MLP NN classifier which achieved a convergence. Basically, the classifier dependent feature selection methods, for the NN classifier case, are based on nodes elimination, or on an appropriate weights modification and this during and / or after the training phase. The OBD Optimal Brain Damage method, which is the basis on which the OCD is proposed, is based on the pruning the least important connections based on the cost variation resulting from the weight suppression. This cost variation, regarding the network weights is called a weight saliency. With an analogy to this, a cell saliency can be defined by simple summarizing the weight salience for all weights going from the particular network node. Formally, the saliency S_{ij} of each connection and the sum of salience of connections going from the unit S_i are as follows:

$$S_{ij} = \frac{1}{2}\frac{\partial^2 C}{\partial^2 W_{ij}} W_{ij}^2 \quad S_i = \sum_{j \in out(i)} \frac{1}{2}\frac{\partial^2 C}{\partial^2 W_{ij}} W_{ij}^2 .$$

We have implemented this feature selection approach on a classifier architecture SSF-1, and a SSF-2 architecture is a result of the implementation of this feature selection approach. In next section we show the obtained experimental results for the proposed classification architectures and the implemented feature selection method, as well as the influence of the feature selection to the classification phase.

5.6 Results and Discussion

5.6.1 Preliminary Data Investigation

At the first phase of the investigations on the input data issued from the digit images in the form of two sets of numerical features of different nature, we input each of the feature sets separately in recognition systems based on full connected neural networks (Fig. 5.6). Thus, we have the evaluation of discrimination power of each of the feature sets and the recognition performances on the correspondent classifiers. The detailed results by digit class are showed for the structural and statistical features set in the Tables 5.2 and 5.3 respectively.

labeled / recognized	0	1	2	3	4	5	6	7	8	9
0	572	3	2	1	2	4	0	1	9	7
1	7	618	6	6	13	5	10	0	25	1
2	1	5	538	23	3	8	5	9	11	0
3	2	6	16	553	2	2	2	1	0	4
4	2	4	11	3	480	3	4	6	8	9
5	3	0	2	2	1	464	4	2	8	10
6	7	4	10	0	9	10	563	0	5	0
7	1	10	3	17	6	4	0	564	7	22
8	10	19	5	15	13	16	3	15	504	17
9	1	1	1	2	27	0	0	14	12	520
Total 5946	606	670	594	622	556	516	591	612	589	590
recognized (%) average 90,35%	94,39	92,24	90,57	88,91	86,33	89,92	95,26	92,16	85,57	88,14

Table 5.2 *Misclassification matrix for the classifier based on structural features set*

labeled / recognized	0	1	2	3	4	5	6	7	8	9
0	571	4	5	12	4	10	1	1	3	0
1	7	646	1	1	1	1	5	6	12	6
2	3	0	540	8	1	10	8	0	1	0
3	2	6	9	546	1	20	2	1	6	1
4	1	2	9	1	520	0	2	10	6	12
5	7	3	4	18	1	471	4	1	10	5
6	3	0	11	1	8	1	562	0	0	0
7	0	0	4	5	0	0	0	570	0	9
8	9	7	8	24	2	3	7	6	542	6
9	3	2	3	6	18	0	0	17	9	551
Total 5946	606	670	594	622	556	516	591	612	589	590
recognized (%) **average 92,77%**	94,22	96,42	90,91	87,78	93,53	91,28	95,09	93,14	92,02	93,39

Table 5.3 *Misclassification matrix for the classifier based on statistical features set*

This, gives us a first idea how each of the feature sets represents the digit images. The average value is better for the statistical feature representation 92,77% face on 90,35% for the structural features set. Nevertheless, in the case of three digit classes (0, 3, 6) the structural feature representation gives better values then a statistical one. Even more, if we analyze the structure of the misclassification, we can se that those two set of features make different "errors" i.e. in many cases we can claim that there could be an important complementary information provided by the two features sets.

As a next step toward an effective fusion of the two feature sets we investigate the k-NN performance (for k=1) on the subsets of the features from the two sets. We start with the subset of the structural features formed by the first 23 features. Further on, we add the features by partitions of the structural and statistical feature sets, up to the total number of 116 features. The results are showed in Table 5.4.

number of features in the feature subset	k-NN performance k=1 (%)
23	81.82
54	88.08
62	93.49
85	94.62
89	94.24
116	94.12

Table 5.4 *k-NN performances evaluated on various subsets of the feature set*

The remarkable result of this investigation is the relatively high discriminatory power of the first 23 features of the structural features set. Describing the digit image only by the means of the first 23 features from the structural feature set, we achieve the performance of 81,88% of the represented digit images from the test set being the first nearest neighbors of the images with the same label from the training set. An other important conclusion from this results is the fact that the higher performance is obtained not by the description in which contribute all 116 features from the both sets. Namely, for the case of 85-feature elements representation we have higher percentage of well labeled nearest neighbors than for the complete 116-elements feature set.

5.6.2 Feature Selection Results

These observations will be completed by the results of the feature selection implemented on the structural and statistical feature sets separately and together. As we have seen in section 5.5.2. we have implemented an OCD feature selection method on the bases of the responses from the SSF-1 classifier. The architecture of the classifier SSF-1 (Fig. 5.5) doesn't make any difference on the importance of the features on his input. They are all considered in the same manner, and the training phase of the classifier adopt the network weights regarding the output responses. A way to introduce a sort of "weighted" inputs in the input layer of the classifier, i.e. to introduce the importance of each feature, is to perform the OCD method on the classifiers input nodes. In that way, the input features whose importance was evaluated during the training phase as a high one, will have more decisive influence on the classification and this can be an important amelioration of the classification phase in two ways: Firstly, we expect better performances, and secondly, the complexity of the classifier can have an important improvement. Here are the results that we have obtained.

In Fig. 5.10. and Fig. 5.11. we show the level of importance for each of the structural and statistical features considered separately.

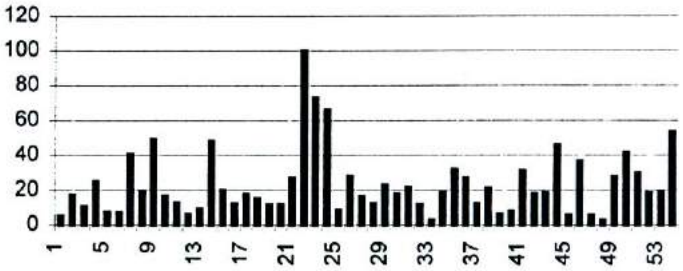


Figure 5.10 *The importance level for each of the structural features*

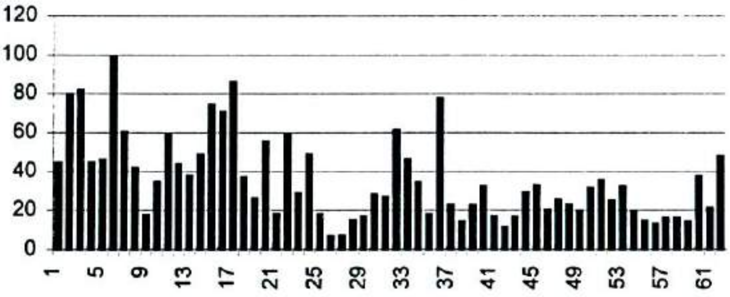


Figure 5.11 *The importance level for each of the statistical features*

Those results give an important information on the very basis of the approach as whole. It is clear that this information can be used in various manners to improve the overall performances of the classification system. Firstly, as a feedback information to the definition on the feature sets. Normally, that will be a valid source of information for the improvement of the feature sets only for the case of the implementation of the classifier that was used to evaluate their discriminative power, or importance, in our case a MLP NN based classifier SSF-1. Nevertheless, the result can be easily generalized for the ensemble of the classifiers of the same category.

This information is very important for another possible research direction, i.e. the architecture of the hierarchical multi-level classifier showed in Fig. 5.12.

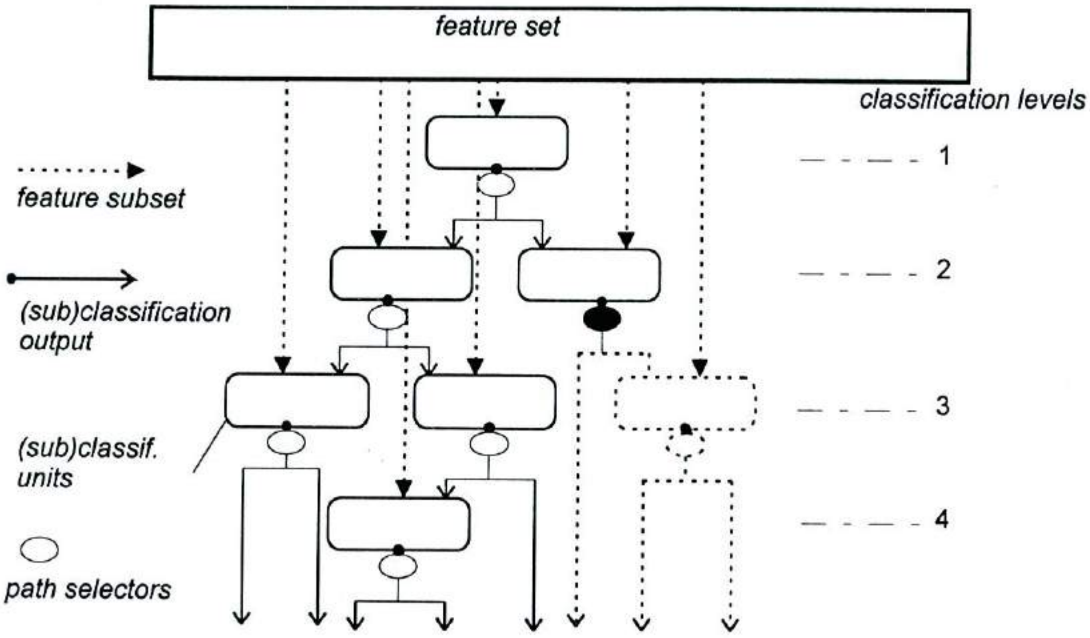


Figure 5.12 *The hierarchical multi-level classifier - the possible implementation of the feature importance estimated by the NN classifier*

Namely, the information obtained in form of individual feature importance can be of crucial importance for constructing a multilevel hierarchical recognition (or classification) system which will use the reliable feature importance information on each level of a decision process. The similar idea but based on boolean neaural networks was implemented by Gazula and Kabuka [Gazula and Kabuka, 1995].

For the purposes of the subject of our investigation of the effective data fusion possibility in the case of different features representation of a digit image we lead our investigations toward a system which will consider the entire set of features and involve their own, previously estimated, importance. At the next step of our experiments we evaluate the feature importance for the ensemble of features presented in the input layer of the SSF1 classifier. The feature importance evaluated by the OCD method is showed in Fig. 5.13.

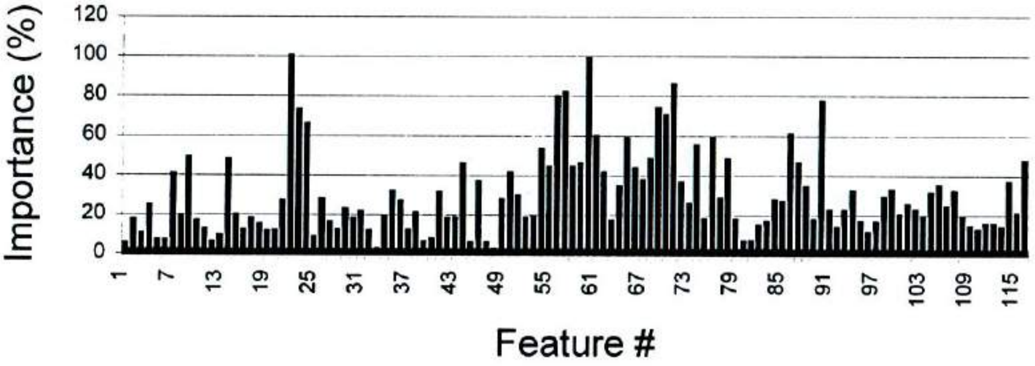


Figure 5.13 *Calculated feature importance on the feature set as a whole*

This result leads to the possibility of implementing a less complex architecture then a one proposed as SSF-1. Namely, the feature importance information is obtained on the basis of the SSF61 architecture. Now, we can include the feature importance in the classification architecture which will take into consideration the important features only. For this purpose, after same experiments we have fixed a threshold level which will reduce the input feature set from 116 to 62 only and this we show in Fig. 5.14.

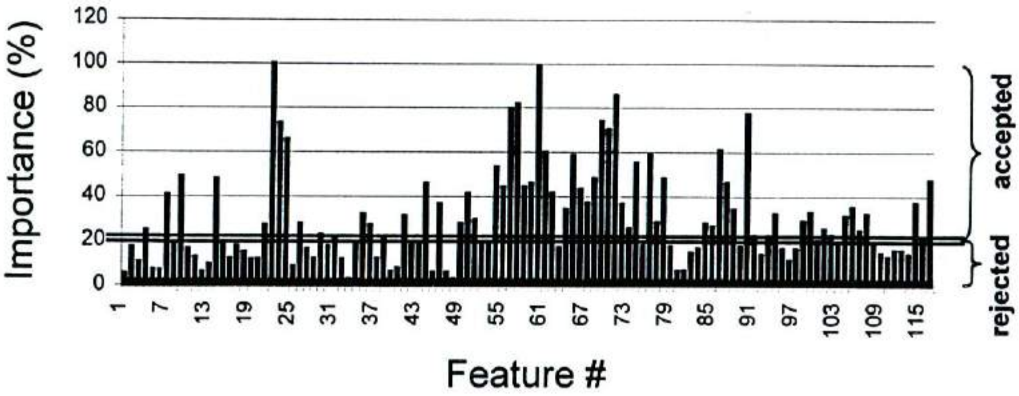


Figure 5.14 *Feature importance threshold*

In this improved architecture participate the best 19 structural and the best 43 statistical features. The new, reduced feature set will be the input of the simplified classifier SSF-2 whose architecture is showed in Fig. 5.8.

5.6.3 Complexity, Performances and Comparison

In Table 5.5. we show a review of the complexity parameters of the proposed classification systems.

Classification system	Multiplications in the Feature extraction phase	NN architect. (input)>(hidden)>(output)	Connections	Nodes
SSF1 - full connected without feature selection	<656	116>35>10	4410	161
SSF2 - full connected with feature selection		**62>35>10**	**2520**	**107**
SSF3 - partially connected without feature selection		116>(20+20)>10	2760	166

Table 5.5 *Complexity parameters for the classification systems proposed*

In the presented approach the computing complexity of the module of feature detection and extraction is linear by the number of points in the input digit image. Precisely, for the structural features set it is linear by the chosen number of control points for the line detection and representation. For the case of 16x16 digit images, as is the case for our experimental base of digits, this number is less than 656 multiplications.

Complexity presentation in Table 5.6. shows an important complexity reduction in the terms of connections (weights) in the system SSF2, compared to the number of connections in the system SSF1 where there was no feature selection implementation. Starting from the basic architecture of SSF1, which contains no feature selection methods, we attempt a complexity reduction in two possible ways: in the system SSF2 by the means feature selection implementation (which allows a connections and nodes reduction), and in the system SSF3 which makes an important connections reduction by modifying the network architecture, i.e. reduced connection by the network architecture concept, although without feature selection methods implemented. By those two modifications we expect an improved, or at least the same classification performance with reduced complexity of the system.

In Table 5.6. we show the complexity and the performances of the proposed systems together with some of the best up to day known systems for the recognition of the same data base.

System	FE phase complexity (multiplications)	NN Architecture	Nodes	Classification phase complexity (mult.)	Performance (%) Test Set
SSF1	**< 656**	116>35>10	161	4,410	96.3
SSF2		**62>35>10**	**107**	**2,520**	**96.5**
SSF3		116>(20+20)>10	166	2,760	94.7
LeNet	96,512	multi-modular NN	4364	3,780	98.6
MMA1	1,798		561	13,656	97.5
MMA2	833		561	12,010	98.0

Table 5.6 *Complexity and performances in comparison*

The conclusion of this stage of our experiments is that we can keep or even augment the recognition rate by an effective feature selection method by which a less important features will be eliminated. Comparing the recognition systems SSF1 and SSF2 the important complexity reduction is evident by reducing the connections (weights) in the proposed NN architecture of the classifier. This is crucial element of the classifiers complexity in terms of training and recognition complexity. From the experiments performed the best result 96.5% recognition on the Test set is obtained for the system SSF2 which has only a 57% of the weights of the SSF1 system.

The architecture Le Net [Le Cun et al., 1990] is a Time Delay Neural Network (TDNN) with an extremely good feature extraction, but requiring 96512 multiplications. The system LeNotre is a TDNN which ahs in total 394 units, 2348 connections and 1196 independent parameters.

The MMA architectures [Fogelman-Soulie et al., 1993] are Multi Modular Architectures for feature extraction and classification which are two stage combination of LeNotre and a Radial Basis Function (RBF) network (MMA1) and a MLP and RBF network (MMA2).

In Table 5.7. a detailed analysis of the considered handwritten digit recognition systems is showed.

System/ parameters	Le Net	Le Notre	MLP +LVQ	MLP +RBFc	LeNotre +knn	LeNotre +LVQ	LeNotre +RBF	VR+NN1 (SSF1)	VR+NN2 (SSF2)
neurons	4 635	394	305 +256	305 +	646	646 +	646 +	**161**	**156**
connections	96 512 +1930	1 798 +550	833 +12 800	833 +	1 798 +	1 798 +	1798 +	**4 410**	**2 520**
weights	648 +1930	646 +550	65 +9 800	65 +12 010	646 +	646 +10 800	646 +13 010	**4 410**	**2 520**
free parameters	96 512 mult. FE	1196 ind.param.	9 800 +10 800	+12 010	10 800 +1500 *(54dd)	1196 +10 800	1196 +13 010	**4 410**	**2 520**
Performance	**98.6**	**95.9**	**96.9**	**98.0**	**97.5**	**97.2**	**97.0**	**96.3**	**96.5**

Table 5.7 *System comparison*

The systems results show that the architectures in the two last columns SSF1 (VR - feature definition and extraction, and NN1 - the NN classifier) and SSF2 (VR - feature definition and extraction, and NN2 - the NN classifier) have very close recognition rate to the rates reported by among the best known systems with significantly less complexity. This confirms essentially the feature definition and extraction phase, and opens perspectives for further search of their accuracy utilization to be examined further on in Chapter 6. of this thesis.

Chapter 6

Decision Fusion

The idea of combining various experts with the aim of compensating for the weakness of each single expert while preserving its own strength, has been investigated widely in the recent works of the pattern recognition area. It has been shown that by suitably combining the results of a set of experts according to a rule, the performance obtained can be better than that of any individual expert. The preliminary experimental results encouraged the approach, and a results are published which aim to determine the complementary nature of experts to be used, their optimal number, and the optimal combination topology [Lincoln and Skrzypek, 1990, Xu et al., 1992, Lee and Srihari, 1995, Huang and Suen, 1995, Kittler, 1996, 1998]. Most of the research has been devoted to finding different combining rules able to determine the most likely class on the basis of the responses of the experts in the case of discordance. Another direction, specially present in the field of neural computation, is to combine different experts each of which is defined over a local region of the input space.

In every case, the mixture of experts approach does not make it possible to exploit the different information coming from different descriptions of the same sample, as all the experts work on the same input space, i.e. on the same description. Further on we represent and develop some of main ideas known up to date and evaluate an original approach for the case of the fusion of the decisions of the classifiers based on different feature spaces extracted from the input digit image.

6.1 Fusion Through Combining Classifiers

Combining the predictions of a set of classifiers has been shown to be an effective way to create composite classifiers that are more accurate than any of the component classifiers. Data from more than one source which may have been separately processed, can often profitably be re-combined to produce more concise, more complete and/or more accurate situation descriptions. A way to define the idea of combining classifiers can be that classifiers with different methodologies or different features are probably complementary to each other; hence the combination of different classifiers may reduce errors considerably and achieve a higher performance accuracy [Huang et al., 1995]. The authors see the difference between a conventional and multi-expert recognition system as is showed in Fig. 6.1. Namely, a conventional recognition system consists of only a single classifier; the output of the system is the output of the only classifier present in the system. From the other way, in a multi-expert recognition system several classifiers are involved, and the outputs of the member classifiers become the input of a combination module, whose output forms the final output of the system. The analogy with the human world is that the decision of a panel of human experts is usually superior to that of any individual.

Kittler considers the problem of classifier combination in the context of the two main fusion scenarios: fusion of opinions based on identical and on distinct representations [Kittler, 1998]. The theoretical framework is developed for classifier combination in these two scenarios, and the classifier combination is seen as a multistage classification process where the a posteriori class probabilities generated by the individual classifiers are considered as features for a second stage classification scheme. It is shown that when the linear or nonlinear combination functions are obtained by training, the distinctions between the two scenarios fade away, and the classifier fusion can be seen in a unified way.

Jouseau and Dorizzi propose a fusion of the data delivered by each agent through fuzzy logic rules [Jouseau and Dorizzi, 1998]. The idea is developed on a vehicles passing detecting system. In this case, once more, the modularity of the system assures its evolution. Kojima and colleagues propose a handwritten digit recognition using neural networks based on approximate reasoning architecture [Kojima, 1993]. The improvement of the recognition rate over conventional NN systems is based on dividing the feature space into plural sub-spaces, where the first stage recognition is to be held. Final recognition is given by integrating the obtained results through a fuzzy inference rule. Cordella and colleagues propose a method to evaluate the reliability of each classification act performed by a given expert, and use this value to weight the vote over the combining instance [Cordella et al, 1999].

The idea of our approach will be to make the neural network classifiers based on statistical and structural features each cooperate through some form of committee classifier. Further on (Chapter 6.2) we introduce more sophisticated way of cooperation, including rule-based reasoning on the outputs of the two classifier. In Fig. 6.1. we show the general scheme for the conventional and multi-classifier recognition system. .

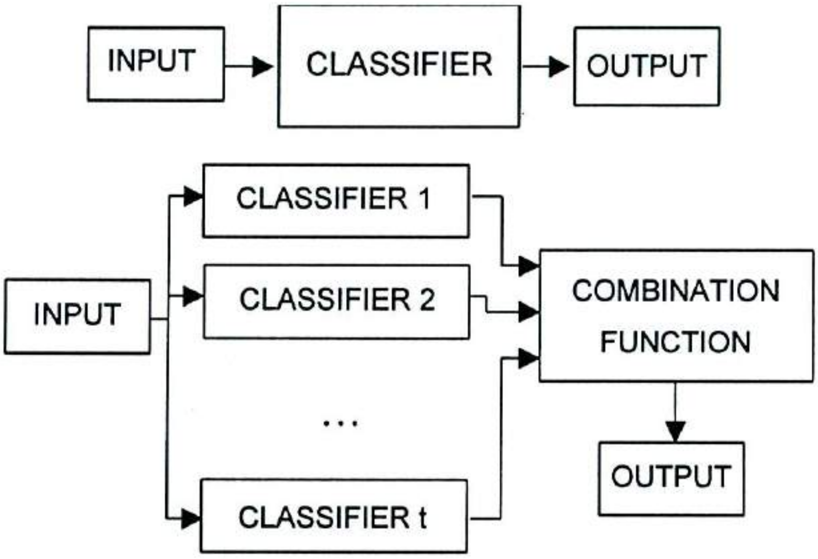


Figure 6.1 *Conventional and multi-classifier recognition system*

There are three important issues on this stage of the discussion of the multi-classifier pattern recognition system. 1.) What is the nature and the properties of the member classifiers and what should be the criteria for their engagement to work together; 2) What kind of output information those classifiers can support; and 3) How should we construct the combination function.

On the level of output information to be used by the combination function, the ideas varies from purely measurement level up to the unique classifier decision-label outputs. Some authors used candidate subset combining and reranking approaches [Ho et al., 1994]. Various techniques can be implemented on the measurement level, as a distance measurements based Dempster-Shafer theory [Xu et al., 1992]. Huang and colleagues propose *a salient approach* which enables to combine classifiers with different meanings and scales. The main contributions of their approach is in the profound use of the measurement level information and this without an assumption of independent behavior of the classifiers [Huang et al., 1995]. Sridhar and colleagues implement the stacked generalization for neural network models by integrating multiple neural networks into an architecture known as stacked neural networks [Sridhar et al., 1999]. The proposed algorithm identifies and combines useful models regardless of the nature of their relationship to the actual output.

Wilson and colleagues propose a set of 45 input networks which discriminate between all two-class pairs for the problem of ten-digits classification problem. The results are shown for the final stage of a single trained network and a simple majority vote rule implementation [Wilson et al., 1995] The majority voting implementation on the outputs of the simple binary decision MLP networks was reported as the one with the highest recognition rate.

. A mixture of experts through a *gating* network has been proposed by Jordan and Jacobs [Jordan and Jacobs, 1994]. The outputs of the experts are the conditional means of the input vectors. Another idea used to combine multiple backpropagation networks is the one of clustering of the multiple backpropagation networks proposed by Linkoln and Skrzypek [Lincoln and Skrzypek, 1990]. The authors show that the clustering of multiple backpropagation networks increase the performance and fault tolerance over a single network.

A two stage neural network architecture for classification purposes is proposed by Hansen and Salamon [Hansen and Salomon, 1990]. After implementing the crossvalidation as a tool for optimizing network parameters and architecture, the authors proposed the reduction of the remaining residual generalization error by invoking ensembles of similar networks using the consensus scheme to decide the collective classification by voting.

Kittler and colleagues describe a common theoretical framework for combining classifiers which use distinct pattern representations and show that many existing schemes can be considered as special cases of compound classification where all the pattern representations are used jointly to make a decision [Kittler et al., 1996]. A methodology for determining the best mix of individual classifiers (examining if they are redundant or detrimental) proposed by Woods and colleagues [Woods et al., 1997]. The authors propose a method for combining classifiers that uses estimates of each individual classifier's local accuracy in small regions of feature space surrounding an unknown test sample.

Huang and Suen propose a neural network at the final classification stage to classify the output values of the member classifiers, which by their own, acts on the transformed forms of likeness measurement [Huang and Suen, 1994].

Merz and Pazzani propose the use of the principal components of a set of learned models for their combination [Merz and Pazzani, 1999]. Chen and Chi present a method of combining multiple probabilistic classifiers on different feature sets under the framework of linear opinion pools [Chen and Chi, 1998]. Based on the soft competition mechanism for automatic feature rank, a generalized finite mixture model is proposed as a linear combination scheme and an Expectation-Maximization (EM) learning algorithm is developed for parameter estimation in the linear combination scheme. The method is applied to the speaker identification task.

In summary, the main classifier decision combining algorithms include the majority vote [Lam and Suen, 1994, Xu et al., 1992], Borda count [Ho et al., 1994], unanimous consensus [Xu et al., 1992, Ho et al., 1994], thresholded voting [Xu et al., 1992], polling methods which utilize heuristic decision rules [Kimura and Shridar, 1991, Nadal et al., 1990], the "averaged Bayes classifier" [Xu et al., 1992], logistic regression to assign weights to the ranks produced by each classifier [Ho et al., 1994], Dempster-Shafer theory to derive weights for each classifier's vote [Xu et al., 1992, Mandler and Schurmann, 1988] and methods of multistage classification [Huang and Suen, 1995]. Regarding the "target" of the combination two main groups can be noted: the combiners who acts on the differentiation of the input data space [Jordan and Jacobs, 1994] and the ones who acts on the differentiation of the learning data set [Chan and Stolfo, 1995, Alpaydin, 1997]. At the later group belongs the system proposed by Fan colleagues where the combiner and stacked generalization are compared from the perspective of training efficiency versus accuracy .

A mathematical framework that explains the reasons for expecting the improvements of the performances of the combination of the outputs of several classifiers is given by Tumer and Ghosh [Tumer and Ghosh, 1995, 1996, 1999]. The authors show that combining networks in output space reduces the variance in boundary locations about the optimum (Bayes) boundary decision. The same authors propose a family of order statistics combiners as an alternative to linear combiners [Tumer and Gosh, 1996].

6.1.1 Classifier Combining Criteria

The problem of decision fusion is the problem of using the specific capabilities of a set of member, or individual classifiers for an improved generalization.

In time perspective the data fusion stage can be performed *sequentially*, where a sequence of decisions of the member classifiers, or generally experts leads to the final decision. This is the case generally when the member classifiers perform a decisions that make up a hierarchical or simply sequential suite of sub-decisions by which a definitive decision can be made.

More frequent case of implementation of the decision fusion is the case of *parallel decision fusion*, where the goal of a fusion is to make more accurate decision than the decision provided by the member classifiers simultaneously.

So, the starting point of our research is the fact that we dispose of two neural network classifiers (Section 5.5.1) which are working on two different feature sets of the same presented digit from the digit data base set. As a consequence of the fact that the two feature sets "see" the same digit image from two different (essentially different) points of view, we will examine the possibility of fusion of the decisions of the two classifiers in a way to provide a more accurate decision at the output. For the instance where the final decision will be made we use the terms Committee Classifier, Ensemble Classifier, Combined Classifier or even Composite Classifier.

The first three criteria to be imposed to create a decision on the basis of existence of multiple member classifiers decisions are: the **accuracy** of the Committee classifier, the **diversity** of the member classifiers, and the **efficiency** of the entire composite classifier system [Skalak, 1997].

The **accuracy** criterion arises from a desire to make the Committee classifier independently accurate. In the great majority of the work from the area, this criterion if not considered as a most important, at least was in the majority of cases a leading goal of the research. Despite an intuitively clear statement that the inaccurate classifiers cannot be used to improve accuracy, and that the accuracy of the member classifiers is of intrinsic importance for the combined classifier along with their diversity [Ali, 1996], same research was done in a direction of developing the idea that the member accuracy may not be of paramount importance in classifier combination. Namely, each of the member classifiers could appear to "see" the input object from different and moreover, complementary perspective, and can provide to the Committee classifier a portion of a needed information. On the experiments with our two member classifiers we will base the aim of constructing of the Committee classifier on the fact that the two member classifiers "see" different aspects of the input digit, and these differences, if they are not useful formally on the decision level, they will be for sure on other levels of the decision process that the Committee classifier should be able to capture. An interesting example of a system that was a 69% accurate on a word pronunciation task and was boosted to 88% accuracy by combinig its decisions with two classifiers that were 23% and 25% accurate when applied individually [Wolpert, 1992].

The **diversity** criterion arises from the essential observation that combining the decisions of a set of classifiers that all make the same errors, or all give their decisions on the same basis cannot lead to any improvement in the accuracy of the Committee classifier. It is obvious that it is difficult to satisfy the accuracy and diversity criterion simultaneously. The only work in direction of explicit attempt to deal with the relationship between accuracy and diversity is the genetic algorithm application to the task of architecture selection for a Committee classifier that combines a set of neural networks, proposed by Opitz and Shavlik [Opitz and Shavlik, 1995]. The fitness function that they apply maximizes a linear combination of classifier accuracy and diversity, but the relative weights are permitted to vary, according to whether the ensemble error and the population diversity are increasing or decreasing. From examining the relative trend of error and diversity, heuristics determine whether to change the relative weighting. The initial weighting heavily favors accuracy.

Inside the two general groups of approaches to determine the diversity of the member classifiers, the methods are rather classical ones. The classification being supervised or not is the major determining factor to divide the approaches in regard to diversity. We have followed the line proposed by Ali and Pazzani [Ali and Pazzani, 1996] which is basically the comparison of the outputs of each member classifier with the desired (target) label during the training phase. We went further on by defining and assuming the measure of confidence in each of the classifiers decision, and we will show in details those aspects further on in this work.

A manner to avoid the strong constraint of this criterion Huang and Suen, propose a Behaviour-Knowledge Space method (BKS), which is basically introducing the information derived from a knowledge space which can concurrently record the decisions of all classifiers on each learned sample; the example where the importance of the data collection phase is emphasized [Huang and Suen, 1995]. The experiments are performed on the data base close by dimension of the one used in this thesis, and besides the fact that the proposed BKS method performs better than Bayseian, Voting and Dempster-Shafers combining rule, it doesn't pass the 95.3% of recognition with 95.6% reliability, whilst for the reliability systems of 98.7% and 99% the recognition rate achieve 89.0% and 82.0% respectively for the combination of three classifiers of individual recognition rate of 90.4%, 90.9% and 92.1% respectively.

The **efficiency** criterion is an obvious constraint about the time and memory consumption in a reasonable amount. The number of member classifiers is a factor of efficiency in a sort of top - down approach of constructing the Committee classifier, which is not the case in our study. In regard of the number of classifiers our starting point is the existence of the two member classifiers with their specific attitudes, so the task is to make a fusion of the existing proposed decisions of each of them.

6.1.2 Committee Classifier Architectures

Generally, we can classify the combining architecture approaches in two groups depending on the nature of the output information provided by the member classifiers. In the case of real-value outputs the common combiners are the function approximation methods, linear or not. In the case of class label outputs essentially we found classification voting and non-voting combiners. There are many examples of the implementation of each of those groups [Wang et al., 1996].

Further more, the combining architectures can be divided in two groups according to the representation nature of the input patterns. Namely, in the first scenario, all the classifiers use the same representation of the input pattern. A typical example of this category is a set of *k*-nearest neighbor classifiers, each using the same measurement vector, but different classifiers parameters (number of nearest neighbors, or distance metrics used). In the second scenario, each classifier uses its own representation of the input pattern. As an important issue of this scenario is the possibility to integrate physically different types of measurements and / or features [Kittler et al., 1996]. As an important outcome from the performed experiments the authors state that the combination rule developed under the most restrictive assumptions - the sum rule - and its derivates consistently outperfomed other classifier combination schemes.

From the point of view of the member classifiers activity for reaching the final decision Ho and colleagues distinguish: classifier fusion and dynamic classifier selection [Ho et al., 1994]. Woods and colleagues propose a method of dynamic classifier selection that uses estimates of a classifier's accuracy in local regions of future space [Woods et al., 1997].

A high performance system for handwritten character recognition by a mixture of multi-stage and multiple expert approaches is proposed by Suzuki and colleagues [Suzuki et al., 1996]. The system is constructed around a two stage strategy: the first stage being simple, fast and reliable recognition with few substitution errors, while the second one is composed of a sophisticated structural classifier and pattern matching method, and these two algorithms run in parallel. The combination is realized through a simple two-dimensional matrix obtained by treating each pair of candidates individually and on the basis of training set.

Lu and Ito propose a task decomposition based on class-relations for the combination of the individual classification modules. The combined classification is based on *minimization* and *maximization* principles of integration of the outputs [Lu and Ito, 1997].

6.1.2.1 Combining - The Starting Points

Let us recall the starting points for the architecture through which we are going to make the two classifiers work together. We have two full connected neural network of type MLP classifiers. Although the inputs are the 54 and 62 respective feature sets, a feature selection technique is implemented after the classifier Training phase, and for the Test phase the member classifiers act on the reduced feature sets, i.e. with the reduced network architecture in the input and hidden layer [Radevski and Bennani, 1997a]. The general combining architecture is presented in Fig. 6.2.

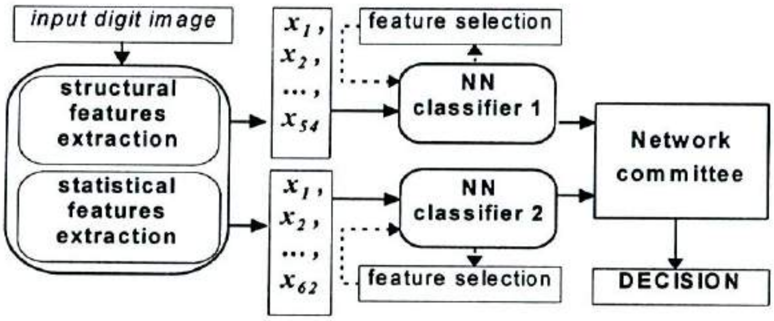

input digit image
structural features extraction
statistical features extraction
x_1, x_2, ..., x_{54}
x_1, x_2, ..., x_{62}
feature selection
NN classifier 1
NN classifier 2
feature selection
Network committee
DECISION

Figure 6.2 *General combining classifiers architecture*

A brief review of the main results and starting performances for the member classifiers to be combined is given in the first two rows of the Table 6.1.

System	NN Architecture	Recognition (%)	Substitution (%) (misclassification)
structural features based NN classifier	54*>20>10	90,4	9,6
statistical features based NN classifier	62*>20>10	92,8	7,2
SSF1 structural and statistical features	116>35>10	96,3	3,7
SSF2 structural and statistical features + feature selection	62>35>10	96,5	3,5

Table 6.1 *Individual and fusion classification review*

As a reference results we keep the last two lines obtained by data fusion of the input feature set.

6.1.2.2 Linear combination

The linear combination may be simple averaging or weighted combination of predictions [Breiman, 1992], [Tumer and Ghosh, 1996]. Those approaches are appropriate for a formal analysis, and often seem to work well. The problem is that the single linear combination cannot reflect the local expertise of the member classifiers. The idea for weighted combination of the prediction of the member classifiers was implemented on various data sets (but not on the handwritten digit recognition) by Hu et colleagues [Hu, et al., 1997]. We have followed the idea of a generalized committee prediction by weighted combination of the prediction of the members for the combination of the predictions of our two member classifiers. Namely, if we denote the true regression function to be approximated with $h(\mathbf{x})$, then we can write the mapping function of each member classifier $y_i(\mathbf{x})$, i=1,2 as the desired function $h(\mathbf{x})$ plus an error function $\varepsilon_i(\mathbf{x})$, i=1,2:

$$y_i(\mathbf{x}) = h(\mathbf{x}) + \varepsilon_i(\mathbf{x}).$$

According to Bishop [Bishop, 1995] the average sum-of-squares error for the model $y_i(\mathbf{x})$ can be written as

$$\mathrm{E}_i = E[\{y_i(\mathbf{x})\text{-}\ h(\mathbf{x})\}^2] = E[\varepsilon_i^2]$$

where $E[.]$ denotes the expectation, and corresponds to an integration over $\mathbf{x}$ weighted by unconditional density $\mathbf{x}$ so that

$$E[\varepsilon_i^2] = \int \varepsilon_i^2(x)\, p(x)\, d\,x$$

The average error is given by

$$E_{AV} = \frac{1}{L}\sum_{i=1}^{L} E_i = \frac{1}{L}\sum_{i=1}^{L} E[\varepsilon_i^2]$$

for the general case of L member classifiers. For the simplest for of the committee being average of the outputs of the L networks, the committee prediction can be written

$$y_{com}(x) = \frac{1}{L}\sum_{i=1}^{L} y_i(x)$$

with the error of the committee of the form

$$E_{com} = E\left[\left(\frac{1}{L}\sum_{i=1}^{L} y_i(x) - h(x)\right)^2\right] = \left[\left(\frac{1}{L}\sum_{i=1}^{L} \varepsilon_i\right)^2\right]$$

Under assumption that the errors $\varepsilon_i(x)$ have zero mean and are uncorrelated,

$$E[\varepsilon_i] = 0, \text{ and } E[\varepsilon_i\varepsilon_j] = 0, \quad \text{for } i \neq j$$

we can relate the committee error to average error of the networks acting separately as follows

$$E_{com} = \frac{1}{L^2}\sum_{i=1}^{L} E[\varepsilon_i] = \frac{1}{L} E_{AV}$$

This represents apparently rather dramatic result that the sum-of-squares error can be reduced by a factor of L simply by averaging the predictions of L networks. In practice, the reduction in error is generally much smaller than this, because the errors $\varepsilon_i(x)$ of different models are typically highly correlated. However, it can easily been shown that the committee averaging process cannot produce an increase in teh expected error by making use of Couchy's inequality in the form

$$\left(\sum_{i=1}^{L} \varepsilon_i\right)^2 \leq L\sum_{i=1}^{L} \varepsilon_i^2$$

which gives $E_{com} \leq E_{AV.}$

The idea of **Simple average combination** is a straightforward one. After scaling the outputs of the NN classifiers in the 0 to 1 interval, the posterior probability of belonging of the input vector $\boldsymbol{x}$ to the class C_i for the case of combining the two member classifiers is given by

$$y_{comb}(\mathbf{x}) = \frac{1}{2}\sum_{k=1}^{2} y_k(\mathbf{x})$$

It is a straightforward way of combining the two classifiers, but it includes nothing about the measure of certainty in each of the member classifiers decisions. Normally, we make an average on the output nodes values of each of member classifiers. The results of this, the simplest implementation of a classifier combination are shown in the Table 6.2.

System	Recognition (%)	Substitution (%) (misclassification)
structural features based NN classifier	90,4	9,6
statistical features based NN classifier	92,8	7,2
A Simple average committee	94.97	5.03

Table 6.2 *A Simple average committee recognition rate*

A **generalized committee prediction** can be given by a weighted combination of the predictions of the members:

$$y_{GEN}(\mathbf{x}) = \sum_{i=1}^{2}\alpha_i\, y_i(\mathbf{x}) = h(\mathbf{x}) + \sum_{i=1}^{2}\alpha_i \varepsilon_i(\mathbf{x})$$

The idea of this schema is based on the fact that we might expect that some members of the committee will typically make better predictions than other members. We would therefore expect to be able to reduce the error if we give greater weight to some committee members than to others.

According to Bishop [Bishop, 1995] and Hu [Hu, et al., 1997], for the finite-sample approximation of the correlation matrix **C**, $C_{ij} = E[\varepsilon_i(x)\ \varepsilon_j(x)]$, the error of the generalized committee can be written as:

$$E_{GEN} = E[\{y_{GEN}(\boldsymbol{x}) - h(\boldsymbol{x})\}^2] =$$

$$E\left[\left(\sum_{i=1}^{L}\alpha_i\varepsilon_i\right)\left(\sum_{j=1}^{L}\alpha_j\varepsilon_j\right)\right] = \sum_{i=1}^{L}\sum_{j=1}^{L}\alpha_i\alpha_j C_{ij}$$

and the optimal values for α_i can determined by minimization of E_{GEN}. In order to find non-trivial minimum of this we need to constrain the α_i by, for example

$$\sum_{i=1}^{L}\alpha_i = 1$$

Using a Lagrange mutliplier λ [Bishop, 1995] to enforce this constraint, we see that the minim searched occurs when

$$2\sum_{i=1}^{L}\alpha_j C_{ij} + \lambda = 0$$

with solution $\alpha_i = -\frac{\lambda}{2}\sum_{j=1}^{L}\left(C^{-1}\right)_{ij}$, which after finding λ, in summary gives:

$$C_{ij} \approx \frac{1}{N}\sum_{n=1}^{N}(y_i(\mathbf{x}^n) - t^n)(y_j(\mathbf{x}^n) - t^n) \qquad \alpha_i = \frac{\sum_{j=1}^{2}(C^{-1})_{ij}}{\sum_{k=1}^{2}\sum_{j=1}^{2}(C^{-1})_{kj}}$$

where t^n is the target value corresponding to input $\mathbf{x}^n$, and N is the number of examples in the training set, we obtain the solution for the parameters α_i.

It is obvious that the generalized committee being the special case of simple average committee we have the inequality $E_{GEN} \le E_{com}$.

Following this procedure we have obtained α_1=0.37 and α_2=0.63, so the Committee acts according to

$$y_{\mathrm{GEN}}(\mathbf{x}) = 0.37y_1(\mathbf{x}) + 0.63y_2(\mathbf{x})$$

In this implication it is clear that the $y_i(\mathbf{x})$ are the values of the nodes of the output layer of the two member classifiers. The final decision will be given by the maximal value of the $y_{\mathrm{GEN}}(\mathbf{x})$. The generalized committee acts like the member classifiers in terms of absence of rejection criterion. The recognition rate on the test set is shown in Table 6.3.

System	Recognition (%)	Substitution (%) (misclassification)
structural features based NN classifier	90,4	9,6
statistical features based NN classifier	92,8	7,2
Generalized Committee	95.09	4.91

Table 6.3 *A Generalized committee recognition rate*

6.1.2.3 Non-Linear combination

Non-linear combining classifiers [Zhang et al., 1992] are the first step ameliorating the behavior of the linear combination. It is expected that the components of the committee classification would have a different relative contributions in different parts of the space. A linear function of the classifiers predictions alone cannot track the selective contributions of the components. Usually the implementation is through neural networks, radial basis or multi-layer perceptrons. We show further on a multi-layer perceptron as a non-linear combiner of the member classifiers. However, it is expected that the components will have different relative contributions in different parts of the feature space, or feature spaces.

In summary, the effective use of the neural networks as a combining tools for a neural network member classifiers systems is based on a the following four valuable characteristics: 1.) they behave as a collective systems; 2.) they can infer subtle, unknown relationships from the data; 3) they can generalize, meaning that they can respond correctly to patterns that are only similar to the original training data; and 4) they are non-linear, that is, they can solve some complex problems more accurately then linear techniques do. Those are the same characteristics desired for the combining function in the general combining architecture [Huang et al., 1995].

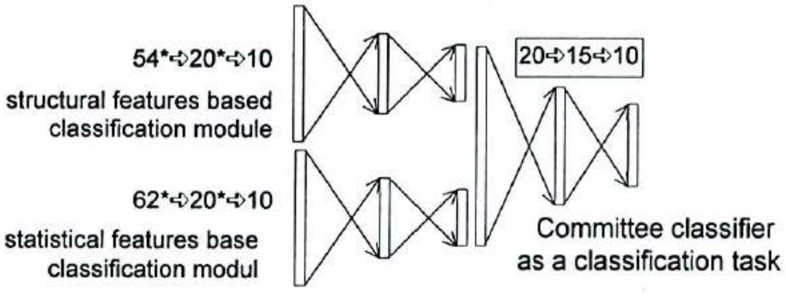


Figure 6.3 *Non-linear classifiers combination*

The "*" sign in the Fig.6.3 is to denote that the given values for the respective layer number of nodes is only indicative and is the starting value given before the feature selection process. After the feature selection phase, all these values are less than the indicated ones.

The input layer of the Committee classifier to perform the combination task is made up of 20 nodes (ten for each of the member classifiers output layers nodes), and the hidden layer has 15 nodes. The output layer has 10 nodes, one for each of the output classes. The results obtained by this combination are given in Table 6.4.

System	Recognition (%)	Substitution (%) (misclassification)
structural features based NN classifier	90,4	9,6
statistical features based NN classifier	92,8	7,2
A non-linear combination	95.22	3.78

Table 6.4 *A Non-linear Combination Recognition Rate*

It worth of mentioning that some authors even argue that concerning speed and recognition accuracy, the multi-layer perceptron outperform other Combination of Multiple Experts techniques [Huang et al., 1995]. We prefer to cite the of multi-layer perceptron as a combination technique only in the light of a very appropriate technique with an excellent performance and speed-recognition accuracy. If we consider the classifier combination techniques in a slightly larger perspective, and introduce the decision fusion aspects and techniques, we will see (Chapter 6.2) a multileveled decision fusion system with advantages in regard of a simple multi-layer perceptron as a combination function.

6.1.2.4 Voting and Non-Voting Techniques

A straightforward way of implementing the voting principle as the framework for the combining of member classifiers is the majority voting. For our case of study of two member classifiers a straight implementation of a voting principle is not possible. The "more than two members" is not a trivial requirement for the voting principle. The combination set of two member classifiers only has been shown as a one providing the greatest incremental gain [Battiti and Colla, 1994]. The cited article gives an extensive analysis of the voting techniques. Skalak has shown that for some real applications the optimal tradeoff between accuracy and computational expense may be made with only two member classifiers [Skalak, 1996]. In section 6.2 we show a way to combine same voting issues, as ranking, or associating a confidentiality during voting in the case of two member classifiers.

The general form of **Bayesian combination** uses the Bayes rule to assign to each instance the class that maximizes the degree of belief that an instance belongs to that class. The errors of each classifier are usually described by the corresponding confusion matrices $[n_{ij}^{1}]$ and $[n_{ij}^{2}]$. The element $[n_{ij}^{k}]$ means that that number of elements of the class C_i have been assigned to the class C_j by the classifier *k*. If we consider the confusion matrices as a sources of a prior knowledge of the classifiers, we can use them for the estimation of the certainty for each of the member classifiers.

With this knowledge, the conditional probability that propositions $\mathbf{x}\in C_i$, i=1,10 are true under the decision made by each of the classifiers $M_k(\mathbf{x})=Cj$ can be estimated by :

$$P(\mathbf{x} \in C_i / M_k(\mathbf{x}) = C_j) = \frac{n_{ij}^{k}}{\sum_{i=1}^{10} n_{ij}^{k}}$$

In this case we use the following estimation of $P_{comb}(\mathbf{x}\in C_i \mid \mathbf{x})$:

$$P_{comb}(\mathbf{x} \in C_i / x) = \frac{1}{\eta} \prod_{k=1}^{2} P(\mathbf{x} \in C_i / M_k(\mathbf{x}) = C_j)$$

with a normalization constant η given by:

$$\eta = \sum_{i=1}^{10} \prod_{k=1}^{2} P(\mathbf{x} \in C_i / M_k(\mathbf{x}) = C_j).$$

The results of the implementation a Bayesian formalism for the combination of the two member classifiers are given in Table 6.5.

System	Recognition (%)	Substitution (%) (misclassification)
structural features based NN classifier	90,4	9,6
statistical features based NN classifier	92,8	7,2
Bayseian formalism	95.04	4.96

Table 6.5 *A Bayesian formalism for classifiers combination*

Non-voting methods include Ranking algorithms [Ho et al., 1994], nearest neighbor and the algorithm based on the Dempster-Schafer theory of evidence [Xu et al., 1992].

The reasoning based on the **Dempster-Schafer** theory follows this reasoning: The probability in favor of the class C_i is expressed by the product of the individual outputs $P_k(\mathbf{x}\in C_i \mid \mathbf{x})$ of each of two classifiers:

$$y_{comb}(\mathbf{x} \in C_i) = \frac{1}{\eta}\prod_{k=1}^{2} y_k(\mathbf{x} \in C_i),$$

$$\text{with } \eta = \sum_{i=1}^{10}\prod_{k=1}^{2} y_k(\mathbf{x} \in C_i)$$

where η is a i-*th* normalization constant, which is given for ten classes and two classifiers to be combined. The importance of this method is that it takes into account the fuzziness of classifiers votes, giving less confidence to less certain votes. The result of the implementation of the Dempster-Schafer theory for the combination of the member classifiers are given in Table 6.6.

System	Recognition (%)	Substitution (%) (misclassification)
structural features based NN classifier	90,4	9,6
statistical features based NN classifier	92,8	7,2
Dempster-Schafer	94.87	5.13

Table 6.6 *A Dempster-Schafer formalism for classifiers combination*

6.1.3 Summary of the Combining Classification Results

In Table 6.7 a summary of the classifier combination is shown. The common characteristic for all combining schema is that we have not included a rejection criterion, so we have the results of the form recognized - not-recognized (missclassified) digits. All classifiers combining schema succeed to reach better results than each of the member classifiers. However, the best results are obtained by non-linear combination of the member classifiers [Cakmakov et al., 2000]. Moreover, the data fusion SSF2 classifier overperforms the non-linear combination as a combining scheme.

Thus, for the purposes of the combination of the two member classifiers decisions the non-linear combination is better when the input layer includes all features from the two feature sets and inputs them in the hidden layer, rather than "mixing" the decisions level of each member classifier.

System		Recognition (%)	Substitution (%) (misclassification)
member classifiers individually	structural features based NN classifier	90,4	9,6
	statistical features based NN classifier	92,8	7,2
data fusion	SSF2	96,5	3,5
combining classifiers	A Simple average	94.97	5.03
	Generalized Committee	95.09	4.91
	A non-linear combination	95.22	3.78
	Bayseian formalism	95.04	4.96
	Dempster-Schafer	94.87	5.13

Table 6.7 *Summary of classifier combination results*

We develop this idea further on in the chapter 6.2 of this thesis, together with the implementation of the rejection criteria for separating the non-recognized inputs in two classes: the class of rejected and the class of miss-classified, and introducing the reliability of the system.

6.2 Improved Fusion

Data from more than one source which may have been separately processed, can often profitably be re-combined to produce more concise, more complete and/or more accurate situation descriptions. Such combination is called data fusion and the aim is to involve the efficient application of suitable inferencing and decision making techniques whilst taking into account the nature of the raw data and the end use.

After our experiments for an appropriate fusion of our two data sets of structural and statistical features, we continue the research in direction of considering two separate NN classifiers, each of them trained on one of two feature sets, and their combination. We search for a possibility to create an effective combination of the classifiers individual decisions by means of a decision fusion algorithm. Thus, the general architecture to apply the given idea is showed in Fig. 6.4.

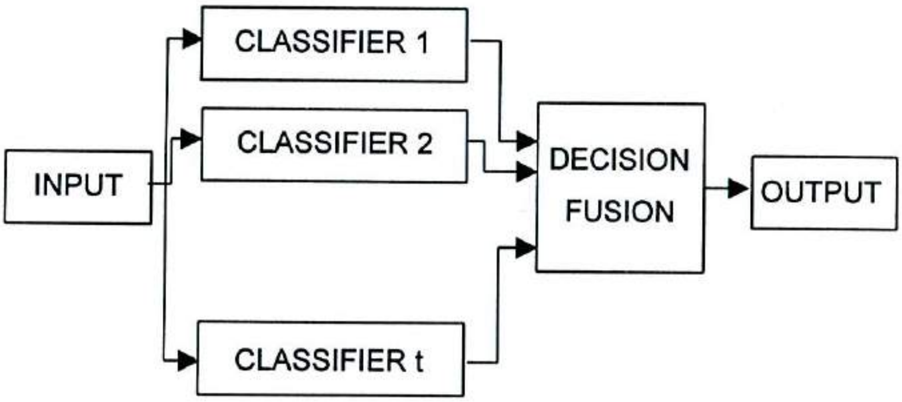


Figure 6.4 *Decision fusion classification system*

As a first stage of investigations in this direction we introduce the rejection criteria in order to divide the set of misclassified inputs in two sets: the set of "unknown inputs" and the set of misclassified inputs. This involves the notion of the reliability of the system, and provides the essential information for the basis on which the classifier outputs a decision. Those information can be used for more effective fusion of the member classifiers decisions on a higher level of decision.

Next phase toward a definitive system for digit recognition based on the combination of the two member classifiers will be a hybrid rule-based (or case-based) recognition system founded on multilevel information provided by the member classifiers to be explained in sections 6.2.3 and 6.2.4. The general architecture to perform the decision fusion in our case is presented in Fig. 6.5.

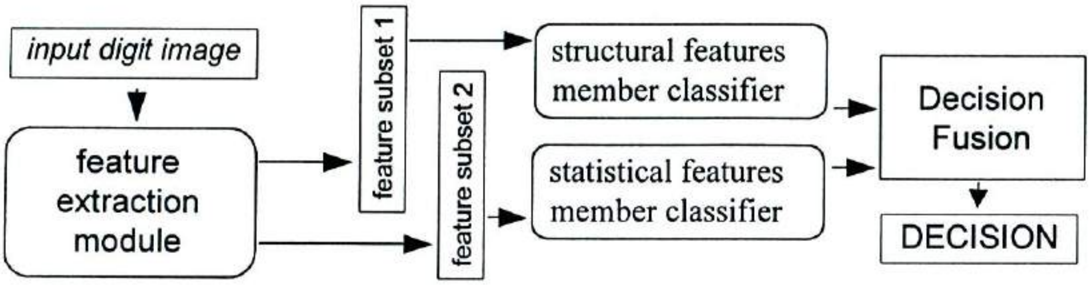

input digit image
feature extraction module
feature subset 1
feature subset 2
structural features member classifier
statistical features member classifier
Decision Fusion
DECISION

Figure 6.5 *General decision fusion system's architecture*

The member classifiers are trained on the set of structural and statistical features respectively. After the training phase the OCD feature selection method is implemented to optimize the classifier's neural network architecture, and a features with an importance below the chosen feature importance threshold are not taken into consideration in the network computing. After the member classifiers convergence achieved, on the entry of the decision fusion instance the two member classifiers output levels are available, in terms of output nodes activation values and the decision choice labels.

6.2.1 Decision Fusion Procedure

The theory of combined statistical solutions through decision fusion levels is a statistical decision theory for building a reliable systems from unreliable components. According to Barabas [Barabas, 1983], the theoretical basis of the problem definition have been given in the work of John Von Neumann " Probabilistic logic and synthesis of reliable organisms from unreliable components", 1956, and later in the work of Shannon C. and Mure E., " Reliable schemes from unreliable inputs", 1960. The work of Rastrigin and Erenstein [Rastrigin and Erenstein, 1981] gives a well developed theoretical basis for the so-called *collective decisions* in the recognition. The general optimality criteria for the pattern recognition cases are based on the work of Kovalevskii [Kovalevskii, 1976].

The main questions for the decision fusion theory can be resumed in following: What are the qualitative new characteristics of the system? What are the conditions to be fulfilled by the member deciders? How to construct the processing engines for the treatment of the information? Questions about error probability calculating; and the optimization of the number of members and the cooperation scheme.

6.2.1.1 Decision Fusion Strategies

A possible essential criterion, to classify the decision fusion schema can be the nature of the information provided by the member experts (classifiers in our case) which contribute in the fusion schema. A global framework for the decision fusion schema from the point of view of the information available on the level where the fusion should be done is given in [Barabas, 1983], and a review of the theoretical foundations of the recognition fusion in [Rastrigin and Erenstein, 1981].

We consider a system of k-members, classifiers, recognition automata, and a decision fusion instance - a committee with the following characteristics:

Each of the members of the system measure the *characteristics* of the recognition object and make a *description*.

Each of the members *take a decision*, classify the presented recognition object in one of the classes from a common set of classes.

Transmit to the committee *the description* of the object and their *decision.*

The system as whole should be capable to get the feed-back information from the committee and correct the a-priori information in the members environment and their propre decisions.

Taking account of all these proprieties, the decision to be taken in the fusion instance can be seen as a function H_i, i=1,...,M, of the signals corresponding to decision classes k_{it}, i=1,...,M, t=1,...,T (number of member classifiers) and the a-posterior probabilities of the hypothesis of member's possible solutions, or other analogue information according to member's decision scheme h_{it}, i=1,...M, t=1,...,T:

$$H_i = \varphi\,(\,k_{it},\ \max_i h_{it}).$$

Before regrouping the possible fusion strategies of the member classifiers let us consider the following notations: Let denote with S the pattern space of M mutually exclusive sets $S = C_1 \cup C_2 \cup \ldots \cup C_M$ where C_i are the sets representing the specified patterns called class, $i \in \Lambda = \{1,\ldots,M\}$. In the case of T-member classifiers, let denote with O^t the classifier t, where t=1,...,T. The action performed by each classifier, for the input pattern **x**, can be seen as an assignment

$$O^t(\mathbf{x}) = \{\ m_i^t(\mathbf{x}) | \forall i\,,\ 1 \le i \le M\}$$

Now we can define the research focus of the decision fusion: When each of the T experts gives its measurement values of M classes for an unknown input, how can these values be fusioned efficiently and effectively to produce a final decision?

$$\begin{array}{l} O^1(\mathbf{x}) = m_1^1, \ldots, m_M^1 \\ O^2(\mathbf{x}) = m_1^2, \ldots, m_M^2 \\ \\ O^T(\mathbf{x}) = m_1^T, \ldots, m_M^T \end{array} \quad \xRightarrow{?} \quad E(\mathbf{x}) = j, j \in \Lambda \cup \{M+1\}$$

where Λ will signify the correct solution or the substitution, the M+1 a rejection.

As we have seen, k_{it} and $\max_i h_{it}$ determine the class in which favor the t^th member took his decision. So, in general case the decision on the fusion instance for the presented pattern **x** will be given by the maximum of the weighted sum

$$E_i = \sum_{t=1}^{T} a_{it} k_{it}, \; i=1,\ldots,M.$$

and we can summarize the decision fusion strategies in the Table 6.8 according to Baras [Barabas, 1983].

Fusion strategy	Information provided	Decision method	a_{it}
IDEALIZED OBSERVER	x_j	-	-
SIMPLE VOTING	$\{k_{it}\}$	majority of member decisions	$a_{it} = 1$
WEIGHTED VOTING	$\{k_{it}\}$, λ_t-prob.	maximal of weighted member decisions	$a_{it} = \psi(\lambda_t)$
OPTIMAL VOTING	$\{k_{it}, \max_i h_{it}\}$		$a_{it} = \psi(\max_i h_{it})$
RULE OF MAXIMAL CONVICTION	$\{k_{it}, \max_i h_{it}\}$	maximal of member's probability hypothesis	$a_{it} = 1$ for t corresp. to $\max_{it} h_{it}$ $a_{it} = 0$ for others t

Table 6.8 *Decision fusion strategies*

In general, a minimal probability of error would be provided by the rule of Idealized observer, and it is the only recommended for the cases where the independence of the error probabilities can not be fulfilled. Ordered of increasing the probability of error the list goes: Rule of maximal conviction, Optimal Voting, Weighted voting and Simple voting. The most simple in terms of implementation is the Simple voting strategy, however, this is inconvenient for applications where the member classifiers have different recognition error probabilities. The Rule of maximal conviction for the cases of independent probabilities errors of the member classifiers is better than the Optimal and Weighted voting.

6.2.1.2 Information to be Provided to the Fusion Instance

We show information that can be provided from each of deciders, in our case the two classifiers, in the decision fusion scheme. The leading ideas are the implication of the analysis of the decision fusion strategies showed above. Namely, we claim that the efficient combination of the different levels information provided by the member classifiers can lead to an efficient decision fusion on the committee classifiers level. The information provided by the member classifiers goes from the measure level to the decision labels level, and this in the different stages of the information processing at the decision fusion instance.

The two MLP NN based classifiers based on the structural and statistical features sets respectively, for the input pattern vector **x** provide the information which can be divided in four levels:

Level-1: At the abstract level each of the classifiers O^i, i=1,2 outputs a label j, $O^i(\mathbf{x})=j$, being the decision label output for that classifier. In fact, regarding the general idea that we implement in the fusion of the decisions of the two classifiers, this will be referenced as a *first label decision choice* for the given classifier. This is issued from the fact that we will consider the outputs of the classifier not only in the terms of the decision that could have been taken if the classifier worked in the autonomous mode, i.e. as an object which outputs a one label. In terms of how the decision is created in the output level of a neural network based classifier, we can obtain an ordered list of decisions (output levels) for the given classifier, and normally, the top choice of this list is the decision label taken as a classifier's output. For the purposes of constructing an effective fusion of the output information from each of member classifiers that make part of the decision fusion system we will use other decisions from the implicit ordered classifier's output list.

Level-2: Each of the classifiers O^i, i=1,2 outputs a second decision choice label l, and ranks it as a second one;

$$\mathrm{O}^i(\mathbf{x})=l \quad \text{iff} \quad O_l^i(\mathbf{x}) = \max_{k \neq j} O_k^i(\mathbf{x})$$

where $\mathrm{O}^i(\mathbf{x})$ is the output label decided by the classifier i, when a pattern $\mathbf{x}$ is presented, and $O_k^i(\mathbf{x})$ is the value of the output node correspondent to the decision activation of the k-th class of the i-th member classifier. This is the label given by the second most active node from the output level of the given neural network based classifier. We will see that this decision level can be a useful source of information, particularly in the decision fusion environment. The second decision label, accompanied with a predefined measure of importance, or level of confidence to be defined further on, can distinguish the comportment of the classifier in the cases of ambiguous input digits particularly.

Level-3: At the measurement level each classifier O^i, i=1,2 to each of the choices in 1. and 2. (above) attributes a measurement value to address the degree that the input pattern (vector) $\mathbf{x}$ has that label. This is what we addressed as a measure of confidence in the given decision, or the measure of importance of the decision.

Let we denote with $O^i j(\mathbf{x})$ the value of the output node which corresponds to class j, for the i^{th} classifier, when the pattern vector $\mathbf{x}$ is presented to the classifier. Then, the information provided on this level can be expressed through the variable

$$d^i(\mathbf{x}) = \max_j O^i_j(\mathbf{x}) - \max_{k \neq j} O^i_k(\mathbf{x})$$

Thus, when the decision output for the classifier O_i is the class j, variable d^i express de measurement value for the certainty of the choice of the class j. Effectively, this is most commonly used rejection criterion in the cases of neural network based classifiers.

The use of the variable d^i as a rejection criteria is through the relation

$$\text{if } d^i(x) \leq \theta \text{ then reject } x \quad (0 \leq \theta \leq 1, \; \textit{threshold parameter})$$

Level-4: As a general information, and after the training phase, the mean μ^i_k i=1,2 k=1,3 and the standard deviation σ^i_k i=1,2 k=1,3 for the differences between the values of the output nodes that gave the first and the second decision levels for three cases of behavior of the classifiers can be a referencing source of information.

We will use this information for establishing the modified and improved rejection criteria, as well as a part of the cooperation scheme of the two classifiers through a committee of classifiers. Those information give an additional point of reference and control, at least for the ambiguous and outliers digits from the test base. An effective consideration of the values of these variables as a parameters of the input feature sets through the training phase can be of use specially for the task of decision fusion.

Having on mind the information from the levels 1. to 3. cited above, we can define the classification process of the classifier O^i by

$$O^i(\mathbf{x}) = \begin{cases} j, & if \quad d^i \geq \theta \\ M+1, & otherwise \end{cases}$$

where M is the number of classes, ten in our case, and M+1 stays for the additional, rejection class. The parameter θ, $0 \leq \theta \leq 1$, is a threshold parameter. It controls the rejection rate according to the information on the measurement level for the certainty of the choice of class *j* as the output class for the classifier *i*, i.e. the value of d^i.

It is clear that going further in direction on searching significant information that could be provided by the member classifiers as they are defined in out research environment we can develop the same ideas further on, taking into account not only the second decision choice of each of the classifiers but to continue further on in the ordered list of decision labels and put in the game the third fourth and other choices. The same approach could lead us to compare the decision labels which are taken into consideration together with the confidence measures. The fact that we consider no more decision choices than a second one is in concordance with the obvious limits of the tasks for this work, as well as in concordance of the idea of the decision fusion as an instance that will an ameliorated accuracy without consequences on the efficiency of the system as a whole.

6.2.2 Basic Reliability Improving

Up to now the evaluation of the recognition was considered as a percentage of recognized digits. The rest is a set of not recognized digits i.e. missclassified or substituted ones. Between them there are obviously ambiguous ones, oultiers or definitely unrecognizable digits which is proper situation with handwriting recognition. The introduction of a rejection criterion should separate the rejected ones from missclassified, or more precisely should establish a criterion for rejecting the unrecognizable ones. This will allow involving the reliability parameter of the system which can be defined as follows:

$$Reliability = \frac{Recognition}{100 - Rejection}$$

where Reliability, Recognition and Rejection are given in percentages.

6.2.2.1 Member Classifiers and Reliability

We've seen in Chapter 5.6. the results of the two individual classifiers, each one performing on the input of one of the feature sets. We recall and summarize the obtained results in Table 6.9.

System	NN Architecture	Recognition (%)	Substitution (%) (misclassification)	Rejection (%)
structural features based NN classifier	54>20>10	90,4	9,6	0
statistical features based NN classifier	62>20>10	92,8	7,2	0
SSF1 structural and statistical features	116>35>10	96,3	3,7	0
SSF2 structural and statistical features + feature selection	62>35>10	96,5	3,5	0

Table 6.9 *Individual and fusion classification review*

At this point of the research we are interested in reduction of the substitution (misclassification) rate. A way to do this is to define a rejection criteria. So, the aim is to treat the substituted digits in order to discard same of them as rejected, and so to reduce the misclassified ones [Radevski and Bennani, 1998].

The rejection criteria should reject both ambiguous digits i.e. the digits which lie close to the border between two classes (ambiguous digits) as well as outliers, the digits which sit far outside the range covered by the distribution of learning examples. Dealing with the outliers has been recognized as a major problem of MLP, and many rejection principles were proposed which can be summarized generally in three major rejection principles given by Fogelman and colleagues [Fogelman et al., 1993]. A variation of the cited three rejection criteria can be found to appear as additional AND or other connections in the output layer [Knerr et al., 1992] or similar. Cao and colleagues introduces a rejection criterion to catch the pattern possibly misdirected to the incorrect sub-classifier by the clustering network [Cao et al., 1994]. The rejection criteria is by definition the intrinsic part of the final classification or of the classification in general. Some authors emphasize the implementation of the rejection criteria further in the classification process. Namely, Baker and Nayar propose a *rejector,* an algorithm that eliminates very quickly a large fraction of the candidate classes (object in recognition) or inputs (i.e. local image brightness values in feature detection) [Baker and Nayar, 1996], as well as Bottou and colleagues [Bottou et al., 1994]. On the basis on multiple simple rejectors they propose a collection of general purpose algorithms for the implementation of simple rejectors. Lim and colleagues propose a cascaded connection of a sigle-layer perceptron network and a simple combinational circuit for the two-level rejection procedure [Lim et al., 1994].

We denote with x^k the presented input to the network, with O^k the computed output and with i_1 and i_2 the indices of the two most active output cells with their respective activities p_1 and p_2. We recall that the computed output O^k approximates the Bayseian a posteriori distribution of the classes, given the example x^k, and if the network is correctly trained, the components of O^k practically add up to 1.

For the classifiers presented in Chapter 5. of this work we have a classifier decision criterion with zero reject and the pattern x_k is classified into the most probable class, the class with index i_1:

$$i_1 = \arg\max_{j} O_j^k \qquad p_1 = O_{i_1}^k$$

$$i_2 = \arg\max_{j \neq i_1} O_j^k \qquad p_1 = O_{i_2}^k$$

We can now define the rejection criteria used in this work as follows:

Rejection-1: If $p_1 < \theta$ then reject x^k. This criterion should handle ambiguous patterns and outliers in about the same way. As soon as the activity is spread on two (ambiguous) or more (outliers) units, p_1 will be low.

Rejection-2: If $p_1 - p_2 \leq \theta$ then reject x^k. This criterion introduces information about the second most active cell: it captures the idea that, for ambiguous patterns, activities p_1 and p_2 should be similar. It is the criterion most commonly used in the literature. It could also pick outliers, in the case where they make all cell activities low.

The introduction of the rejection criterion should divide the set of substituted (misclassified) digits in two sets: the set of substituted digits and the set of rejected ones. In the case of our two classifiers, based on corresponding structural and statistical features sets the aim of this phase of experiments is to determine the possibility of proposing the recognition with modifiable reliability rate, i.e. to use a variable rejection threshold θ. We show the results obtained for the two individual classifiers in Table 6.10.

Structural features based NN classifier					Statistical features based NN classifier				
θ	Rec.%	Sub.%	Rej.%	Rel.%	θ	Rec.%	Sub.%	Rej.%	Rel.%
0	90.35	9.65	0	90.35	0	92.77	7.23	0	92.77
$2\text{x}10^{-5}$	88.43	7.00	4.57	92.66	$2\text{x}10^{-5}$	91.49	5.45	3.06	94.37
$4\text{x}10^{-5}$	86.24	5.25	8.51	94.26	$4\text{x}10^{-5}$	89.77	4.31	5.92	95.41
$6\text{x}10^{-5}$	83.88	3.85	11.08	94.33	$6\text{x}10^{-5}$	87.47	3.53	9.00	96.12
$8\text{x}10^{-5}$	80.86	2.99	16.15	96.54	$8\text{x}10^{-5}$	79.77	4.09	16.14	95.10

Table 6.10 *Reliability rate for the two individual classifiers*

However, as it can be seen, introducing the rejection criterion in this way is not a straightforward way to place a part of substituted patterns into the set of rejected ones. Imposing a threshold level for accepting / not accepting a given decision will not only discard the outliers and ambiguous patterns from the set of previously substituted ones, but will discard a part of well-recognized digits; those ones which were well-recognized despite the "uncertainty" of the classifier, i.e. the proximity of other class-candidates.

Thus, for the classifier based on the set of structural features, involving the rejection criterion with the first of proposed value for the threshold θ, $\theta = 2\text{x}10^{-5}$, we obtain 11.57% unrecognized features, due of 7% for substituted (missclassified) and 4.57% for the rejected ones (Table 6.10.). This is more than the 9.65% unrecognized (at the same time missclassified) in the same classification system without rejection criterion. The difference reduce the set of well-recognized digits from 90.35% to 88.43%. We can observe the same process in the case of statistical features based classifier where the 7.23% of unrecognized (missclassified) digits in absence of rejection criterion, become 8.51% of unrecognized after the introduction of the first rejection threshold. Reducing the percentage of missclassified digits from 7.23% to 5.45% provoke 3.06% of rejected digits, but at the same time reducing the recognized ones from 92.77% to 91.49%.

These observations and results open two possible directions of investigations and applications: First, it is obvious that with an effective modification an improved rejection criteria can be established which will take into account the activities of all output units of the given classifier. Moreover, and that will be the direction that will be developed further in this work, an integration between activities values and the output classes can improve the reliability of the system, and this, keeping the high recognition rate.

The reflections that we made on the relation rejection criterion - reliability of the system for the case of two classifiers based on structural and statistical set of features respectively, will be emphasized for the case where the decision will be given by an instance who is considering the results from the above described two classifiers at the same time. In next chapter we discus same possibilities of combining the decisions of the two classifiers to finally achieve a decision fusion system where the reliability improvement facilities for each of the classifiers will be integrated in a combined rule-based committee classifier recognition system.

6.2.2.2 The SSF-1 With Improved Reliability

In the very analogue way the rejection criterion can be implemented on the proposed architecture for data fusion, the SSF1 architecture (Chapter 5.). We observe a similar way of comportment of the improving the reliability rate while adding digits to the rejected digits set. Reliability rate attends higher values, and now in around 98% of reliability has been obtained for a recognition rate around 94% (Table 6.11.) which is essentially due to the higher starting point of the recognition rate in the system before introducing the rejection criterion.

θ	Rec.%	Sub.%	Rej.%	Rel.%
0	96.35	3.64	0	96.35
$2x10^{-5}$	95.34	2.67	1.98	97.26
$4x10^{-5}$	93.95	2.07	3.98	97.84
$6x10^{-5}$	92.41	1.51	6.07	98.38
$8x10^{-5}$	90.46	1.16	8.38	98.73
$1x10^{-4}$	87.62	0.84	11.53	99.03

Table 6.11 *SSF1 Results of trade-offs as θ increases*

The SSF1 classifier "sees" the digit image through the set of 116 features which are essentially divided in two subsets of features of different nature. For the classifier there is no difference between them. After the phase of feature selection a subset of more significant 62 features is chosen in which participate features from both structural as well as from statistical features set. A possible direction for improving the recognition and the way that we will handle the unrecognized digits is to take into consideration the eventual differences of how the input digit is seen by each feature set. That will be the aim of the experiments and research given in the next two chapters.

6.2.3 Decision Fusion Sources

In order to get the most relevant information out of the two member classifiers, and to confirm the minimum diversity and independence of their error probabilities we need to know same specifies of their performances when act in autonomous environment.

There is a wide variety of choice of approaches and the search for the reliable information for the decision instance. In what follows we show two kind of information sources that were used in the evaluation of our approach of multi-level decision fusion.

6.2.3.1 Quantitative sources

The set of quantitative preliminary experiments has for purpose to give an image about the ordering in the decision labels ordered list of the candidate decisions. Normally, in a standard mode of running, each of classifiers outputs an unique decision label, as the label associated to the most active node from the nodes in the output level of the neural network based classifier. Nevertheless, taking into consideration the Bayseian nature of the activity values of the nodes in the classifiers output level, we can order the top candidates in the decision labeling procedure of each classifier. If, instead of taking a single output label, designed by the most active node in the output level, we take into consideration the second classifier's choice, and then compare the target class label value with one of the top two "propositions" by the classifier, we can have an idea, "how far" for the possibly mistaken first decision choice, the right one is. At least, if we consider only the second decision choice, we can see, in the case of wrong first decision, if the right one is fairly "close".

Comparing the target class label values with the set of first two choices for each of the member classifiers on the Test set, we have obtained the percentages of correct decision labeling shown in Table 6.12.

Classification module	Recognition rate first decision label	Recognition rate for « Top two » decision labels (%)
structural features classifier	**90.35** [89.35 ; 91.25]	**95.56** [94.85 ; 96.18]
statistical features classifier	**92.77** [91.91 ; 93.58]	**97.00** [96.40 ; 97.50]

Table 6.12 *Recognition rate for the first and "top two" decision labels*

In average, about a 5% more of the presented digits are with a target decision label value in the "top two" set on the output of each of the member classifiers. From this result it is promising to refine the way of considering the output level node values of each of the member classifiers in order to capture the right answer, at least in the cases when it is "not so far" from the outputted one.

But, what is most important from the point of view of the fusion of the decisions issued from the two classifiers, is the differences in their right and wrong answers. The independence of errors of the member classifiers is of crucial importance for the effective combination, in terms of the fusion of their decisions. If we take a single decision label output from each of the member classifiers, and count the concordance of the right answers for the two classifiers we have the simultaneously right decision label in 86.26%, and about 10% more of right answers if we allow the appearance of the decision label target value to be one of the first decision labels of member classifiers. A big majority of target values, a 98.77% is within the "top two" set of decision labels outputs of the member classifiers. These observations are summarized in Table 6.13.

Notation	Event	Percentage
T - the target class a1, a2 and b1, b2 the first and the second label class decision of the structural, respectively statistical features based classifier	T=a1 **and** T=b1	86.26%
	T=a1 **or** T=b1	96.86%
	T=a1 **or** T=a2 **or** T=b1 **or** T=b2	98.77%

Table 6.13 *The basic relations between member classifier's decision-label outputs*

Those results make evident the important percentage of the existence of the target labels between top two choices of each classifier (line 2 and 3). These observations on the decision label outputs in the respective classifiers show that there could be a space for an implementation of more sophisticated procedures of fusion of the decisions of the member classifiers in order to establish an effective cooperation scheme through which a complementary implicit information can lead to recognition system with improved recognition and reliability rate.

6.2.3.2 Qualitative Sources

In the spirit of involving different information levels to the final decision fusion procedure, the next step of searching for the additional information about the ways each of the member classifiers has opted for a specific decision is the measurement level information. Namely, the decision level outputs are based on the values of the neural network nodes of the output layer. The idea is to use the values of the output level nodes that "won the competition" for the decision labels as a complementary source of information about the confidence that we should have in the label that was outputted by the classifier. This information can be used in different ways, principally as a numerical values of the output node from the output layer that gave that decision label or in relative sense - considering the relations between the numerical values of the "winning" node and other nodes from the same layer.

We've calculated the means and the standard deviations of the differences between the numerical values of the nodes that gave the first and the second decision choices, the Level-3 information from the section 6.2.1.2.

We consider the meaning and the standard deviation of the output nodes values of the measurement level output information for the two member classifiers individually. It is clear that, if needed the transformation and the normalization, not only of the similarity and confidence, but also of the distance measurements can be easily implemented. Several methods of data transformation proper to the handwritten recognition task and the various measurement level information that can be expected on the fusion level is given in the work of [Huang et al., 1995].

The values of the statistical values observed in our experiment obtained during the training phase of the neural network classifiers are given in Table 6.14. There was no need of any transformation of the measurement level data in this phase of the experiments. This is implicated by the fact of the same nature of the output information provided by the member classifiers, the same class of the member classifiers involved in the fusion process. However, this is due essentially to the properly chosen feature definition and extraction phases. Namely, the pixel - level measurements have been transformed in two kind of features in a manner of representing the measurements of confidence nature of the same interval. We recall that the structural features are in fact the probabilities of appearance of the structural primitive elements in the digit image, while the statistical ones can be seen as the probabilities of the black-fulfilled regions of the input digit image.

While the processes of feature selection, network training and acting there is no action which will transform the two feature set elements in a principally different way. As an implication of these, we have the same nature of measurement level information at the output nodes of the two member classifiers. They have the same meaning and scale.

Classification module	**Event**	**Mean** (μ^{l}_{k})	**St. deviation** (σ^{l}_{k})
on	1st decision is the right one	μ^{1}_{1} =1.4908	σ^{1}_{1} =0.5237
Structural	2nd decision is the right one	μ^{1}_{2} =0.0249	σ^{1}_{2} =0.1455
Features	none of the decisions is correct	μ^{1}_{3} =0.5854	σ^{1}_{3} =0.4733
on	1st decision is the right one	μ^{2}_{1} =1.6179	σ^{2}_{1} =0.4669
Statistical	2nd decision is the right one	μ^{2}_{2} =0.0258	σ^{2}_{2} =0.1551
Features	none of the decisions is correct	μ^{2}_{3} =0.7457	σ^{2}_{3} =0.6086

Table 6.14 *Output nodes descriptive statistic values*

The obtained elementary statistical values for the both classifiers will be a source of a complementary information during the decision fusion phase (Table 6.14.).

6.2.4 A Rule Based Decision Fusion

The information available on the entry of the decision fusion instance as cited above (Chapters 6.2.1.2 and 6.2.3.) satisfies the conditions settled by Huang and colleagues [Huang et al., 1995]. Namely it is clear that:

1.) The performance rank order is not changed through data transformation, i.e. the measurement values and their corresponding transformed values have the same preference rank orders.

2.) The range of the values of the output level of the member classifiers are between 0.0 and 1.0.

3.) The larger value in the output node of the member classifier, the more likely the corresponding class is the class of that pattern.

We will consider the responses of the member classifiers, at all information levels. Let define confidence intervals around the mean values for the events cited in Table 6.14.

$$\Omega^i_j = [\mu^i_j - s^* \sigma^i_j \, , \, \mu^i_j + s^* \sigma^i_j]$$

For i=1,2 we obtain the corresponding intervals for the two member classifiers. The parameter ***s*** is a real number from the interval]0, 2], giving a 68% of the output values into the intervals Ω^i_j for the value of s=1, and 95% of the output values for s=2, assuming a normal distribution.

We have showed the possibility of using the variable $d^i(x)$ for establishing a rejection criteria within each of the member classifiers.

The idea of the improved cooperation between the member classifiers will be based on the use of the output information of the two member classifiers at various levels defined in sections 6.2.1.2 and 6.2.3. Namely, at the first stage of classification our aim will be to classify the input pattern vector with as higher as possible recognition rate, keeping as low as possible the substitution rate. This stage of classification will use each of the member classifiers separately. As an output from this stage we will have a set of recognized and classified patterns with very low rate of substitution, and a set of rejected patterns. The rejected patterns can be further on processed by a classifier which has higher recognition rate (a generalized committee cooperation scheme, for example). Thus, the substitution (misclassification) rate of the classifier implemented in the second stage of classification will introduce less confusion in the final results, being implemented on the set of "hard" patterns only. Further on we will show the implementation of the two stage strategy.

Combined systems based on neural networks and rule-based frameworks for pattern recognition have been proposed in various variants and implementing a wide variety of ideas. Greenspan and colleagues propose an approach which enables unsupervised and supervised learning, where neural network clustering scheme is used for the quantization of the input features at the first stage. A supervised stage follows where labeling of the quantized attributes is achieved using a rule-based system [Greenspan et al., 1992].

6.2.4.1 A Rule Based Decision Fusion - The First Passage

The aim of this first stage of the classification is to classify as much as possible input patterns, keeping the low substitution (misclassification) rate. So, the result of this phase will be a set of well classified patterns and a set of "hard" patterns, rejected from this phase of classification. To complete this task we will use the information provided by the two member classifiers in these forms:

1.) at the abstract level we will take into consideration the top two class labels of each of the classifiers: ***a1*** and ***a2*** for the structural features classifier, and ***b1*** and ***b2*** for the statistical features classifiers respectively;

2.) the belonging of the values of $d^{i}(x)$

$$d^i(\mathbf{x}) = \max_j O_j^i(\mathbf{x}) - \max_{k \neq j} O_k^i(\mathbf{x})$$

into intervals $\Omega^i_j = [\mu^i_j - s^*\sigma^i_j\ ,\ \mu^i_j + s^*\sigma^i_j\ ]$, for i=1 (structural features classifier) will be denoted by $f_j = d^i(\mathbf{x}) \in \Omega^i_j$ and for i=2 (statistical features classifier) by $g_j = d^i(\mathbf{x}) \in \Omega^i_j$. The meaning of each of the intervals is given in Table 6.15., in the column of corresponding events.

Classification module	**Event**	**Notation**
on Structural Features	1st decision is the right one	$f_1 = d^1 \in \Omega^1_2$
	2nd decision is the right one	$f_2 = d^1 \in \Omega^1_2$
	none of the decisions is correct	$f_3 = d^1 \in \Omega^1_3$
on Statistical Features	1st decision is the right one	$g_1 = d^2 \in \Omega^2_1$
	2nd decision is the right one	$g_2 = d^2 \in \Omega^2_2$
	none of the decisions is correct	$g_3 = d^2 \in \Omega^2_3$

Table 6.15 *Notation of the characteristic events*

Using the above defined information we can introduce the rule-based decision procedure for the first stage of the classification process. Namely, we will show the results of the classification at this stage for some possible rule-based decision schemes. We recall that the aim of this stage of classification is to obtain a low substitution rate with the recognition rate as high as possible. In Table 6.16. we show the results of the implementation of some rule-based strategies. The idea is to cooperate the abstract, decision level output information from the member classifiers with the measurement level information in order to obtain more reliable decisions.

#	sequence of rules IF	decision	s	rec.(%)	sub.(%)	rej.(%)	rel.(%)
1.	a1=b1	c=b1	s=0.2	86.33	1.5	12.18	98.30
	(g1and f2) and b2=a1	c=a1					
	(f1 and g2) and a2=b1	c=a2					
2.	g2 or a1=b1	c=b1	s=2	94.13	4.76	1.11	95.19
	f2	c=a1	s=0.2	88.26	2.05	9.69	97.73
	(g1 or f2) and b2=a1	c=b2					
	(f1 or g2) and a2=b1	c=a2					
3.	g2 and a1=b1	c=b1	s=2	91.19	2.24	11.79	97.46
	f2	c=a1	s=1	85.97	5.21	3.60	94.59
	g1 and b2=a1	c=b2					
	f1 and a2=b1	c=a2					
4.	g2 or a1=b1	c=b1	s=2	94.30	5.70	0.0	94.30
	f2	c=a1	s=1	93.90	5.01	1.09	94.93
	g1 or b2=a1	c=b2					
	f1 or a2=b1	c=a2					

Table 6.16 *Various rule-based strategies and corresponding recognition rates*

It is clear that starting from each of the systems presented in Table 6.16. one can easily create classification systems with lower substitution rate by decreasing the interval around the mean value for the corresponding events of decision. This is releasable by changing the value of the parameter *s* which controls the interval around the mean value obtained during the learning phase for the specific event. Here, we show only some characteristic results obtained for some extreme values for the parameter s.

The strongest criteria for the decision of the member classifier is given in case 1 of the Table 6.16. The sequence of rules that have been used for this classification process results in lowest substitution rate. In this case, firstly, if the two member classifiers gave the same class as the first choice class, we take that decision as the decision for the classification of this stage too. In the cases where the two member classifiers don't give the same class as the first choice class, we take the second choice label of the better individual classifier (the statistical features classifier) b2 only if it gives the same decision label as the first choice of the structural features classifier a1=b2 and the both classifiers are enough sure in their decision. In this case, to be sure in the given decision means that the statistical features classifier outputs the value of b2 as a decision label along with the event g2. We recall that the event g2 means $d^2 \in \Omega^2{}_2$, the value of the output node that gave the decision b2 drops into the interval around the mean of the values for this event obtained during the learning phase.

6.2.4.2 A Rule Based Decision Fusion - Final Classification

As a result from each of the cases described in 5.3.1 we have the set of well recognized and classified patterns with low substitution rate, which means low number of misclassified (substituted) patterns (1.5% for the first set of rules in Table 16.). Along with this set of classified patterns we have a set of rejected patterns. Those patterns do not fulfill the strong criteria posed in the decision rule-based sequences from the Table 6.16.

For the set of rejected patterns from the first stage of classification process, described in 5.3.1, we can implement more sophisticated decision procedures, involving the good recognition performances of the generalized committee classification cooperation. Thus, at the first pass of classification, using a strengthen rule-based cooperation schema we have avoided the high misclassification rate of the generalized committee decision scheme. However, we can implement the generalized committee decision schema, as the best recognition rate schema, on the smaller set of the "hard" patterns that have been rejected from the first stage (5.3.1), thus obtaining a high reliability recognition system.

If we take the rejected patterns from the cases 1. and 3. from the Table 6.16., and put them as an input of the generalized committee classifier we will obtain the results shown in Table 6.17. below.

System	Rec. (%)	Sub.(%)	Rej.(%)	Rel.(%)
Strategy 1 (Table 16) + generalized committee	**95.1**	**1.50**	**3.4**	**98.3**
Strategy 3 (Table 16) + generalized committee	94.8	2.24	2.96	97.68

Table 6.17 *Final classification recognition results*

The results of the final classification showed in Table 6.9. show that the primary goal of the research is reached. Namely, we have obtained the highest reliability rate for the systems that keep high recognition rate as well. The cooperation of the member classifiers is optimized from the point of view high recognition rate along with high reliability rate [Radevski and Bennani, 2000, Radevski et al., 2000].

We have shown a possibility for cooperation of classifiers based on different feature sets through a Committee classifier. The feature sets are based on both domain dependent and domain independent features. For each of the two set of features a simple MLP NN classifier has been constructed. The behavior of those classifiers has been studied and a multilevel information has been extracted taking into consideration various information about the top two decision labels from each of the classifiers. A rejection criteria is established and its usefulness for constructing the systems with changeable reliability rate is shown. The classification process is made up of two main phases, one for each of the two implemented cooperation schemes. Firstly, all patterns are processed by rule based reasoning cooperation of the outputs of the two classifiers. Here, the top two decision levels obtained from the two member classifiers are combined with the other reinforcement data from the learning phase of each of the classifiers. By adjusting the value of the rejection threshold parameter, systems with various reliability level are obtained. As an output from this phase, are high reliability classification systems are obtained. Those systems have a very low misclassification (substitution) rate, but still not a desirable recognition rate. The rejected patterns from this phase of classification are input in the second phase of classification. These "hard" patterns are processed by a modified generalized committee classifier. Thus, the disadvantage of the Generalized committee to have a high misclassification is avoided by its implementation on a reduced set of patterns, and its high recognition rate in addition to the patterns recognized in the first phase of classification results in a high reliability recognition system.

Chapter 7

Conclusion

7.1 Summary of the Dissertation and Contributions

The main results of this thesis are situated in the field of feature definition and fusion and the field of decision fusion.

Firstly, an exhaustive state of the art is given of the appearances of the phases of feature extraction and selection in pattern recognition application is given. The ambiguity of the terminology and the implementation in the available literature is detected, and a classification and definition of the crucial phases of feature extraction and selection is given.

A set of structural features is proposed for the description of handwritten characters. The feature definition is tested on the application of hand-printed Cyrillic letters and on the set of segmented handwritten digits from the NIST data base. An original line primitive similarity criterion is proposed for the construction of the set of structural features and its behavior is tested on the two data set bases. The proposed method has been shown as a low cost effective presentation of handwritten characters and its incorporation in the further stages of a pattern recognition systems has been shown as an advantageous one.

On the example of hand-printed letters, all conditions are established for an effective multistage recognition system based on the clustering of the input letters presented by two set of features of different nature. The Bayes error has been measured on each stage of the hierarchical dendogram of the Cyrillic letters and has the value of 0.84% of recognition error on the first stage up to 15.98% regarding the input data by classes, without any clustering. The conditions established fulfill all conditions for the construction of an effective classifier based on set of structural and statistical features, with the implementation of rule-based decisions of different complexity on each level of hierarchical structure of the learning data.

Next important contribution of this thesis is the fusion of the set of features of different nature: structural and statistical for the handwritten digit recognition through a neural network based classifier. The comparative tables of the results (Table 5.6 Complexity and performances in comparison and Table 5.7 System comparison) situate the advantages of the proposed system: very near result to the best known recognition systems on the same data base, with lower complexity and an open possibility of intervention in multiple stage recognition environment; the recognition rate achieved of 96.5% recognition, with a low complexity preprocessing procedures and low cost pruned neural network classifier, and on the basis of the two individual classifiers that have the recognition rates of 90.35% and 92.77% when acts in autonomous environment.

The classical classifier combing criteria have been implemented to the problem of combing the two neural network classifiers based on each of the feature sets defined previously. The comparative list of the results being showed in Table 6.7, shows that implementing the standard combining scheme do not improve the individual recognition rates significantly. We give an exhaustive study for the decision fusion for the case of the fusion of the decisions of two neural network based classifiers. A combined two stage rule-based decision fusion is proposed and the reliability improvement is shown to achieve the final recognition system with decreased substitution rate from 3.5% to 1.5% only for a system with 95.1% recognition rate.

7.2 Issues for Future research

We have discussed three main subjects regarding pattern recognition systems based on multiple features and simple or multiple classifiers: the feature extraction and selection phase, the data fusion phase and the decision fusion phase.

The perspectives for further research in each of them can be summarized as follows:

In the feature definition, acquisition and extraction phase the set of structural features can be enriched with more domain dependent features obtained after the classifier feedback, and other line primitive similarity functions can be defined.

The proposed feature subset selection method should be tested with a dynamical classifier dependent component.

On the basis of the hierarchical clustering structure, and the estimated limits of the Bayes error on each level, the effective set of rule-based and combined decisions should complete the multi-level simple classifier recognition environment.

At the decision fusion stage more sophisticated yet simple rules can be tested, and modified classical combining scheme can provide a reliable recognition on more than two existing levels of "easy" and "hard" patterns. The weighting of the member classifiers decision can be tested on a different-values-by-node principle and not only by different-values-by-classifier principle as is actually. The basis of the reliability of the systems - the rejection criterion can be tested while on only the best two decisions of each classifier are considered but deeper in the output node values of the member classifiers.

Bibliography

[Alpaydin, 1997] Alpaydin E. 1997. Voting Over Multiple Condensed Nearest Neighbors. *Artificial Intelligence Review, (Special Issue on Lazy Learning).* 11: 115-132.

[Ali, 1996] Ali K.M. 1996. Learning Probabilistic Relational Concept Descriptions. PhD Dissertation, *Dept. of Information and Computer Science,* University of California, Irvine, CA

[Ali and Pazzani, 1995] Ali K.M., Pazzani M.J. 1996. Error Reduction Through Learning Multiple Descriptions. *Dept. of Information and Computer Science Technical Report 95-39,* University of California, Irvine, CA

[Amin et al., 1996] Amin A., AL-Sadoun H., Fischer S. 1996. Hand-Printed Arabic Character Recognition System Using an Artificial Network. *Pattern Recognition* 29 (4): 663-675.

[Auger et al., 1992] Auger J.M., Idan Y., Chevallier R., Dorizzi B. 1992. Complementary Aspect of Topological Maps and Time Delay Neural Networks for Character Recognition. *International Joint Conference on Neural Networks* 4: 444-449.

[Baker and Nayar, 1996] Baker S., Nayar S. K. 1996. Algorithms for Pattern Rejection. *Proceedings of 13th International Conference on Pattern Recognition* 869-874.

[Barabas, 1983] Barabas I.L. 1983. *Kollektivnie Statisticeskie Resenia pri Raspoznavanii.* Radio i sviaz, Moskva (in russian)

[Basu and Fu, 1987] Basu S., Fu K.S. 1987. Image Segmentation by Syntactic Method. *Pattern Recognition* 20 (1): 33-44.

[Battiti, 1994] Battiti R. 1994. Using Mutual Information for Selecting Features in Supervised Neural Net Learning. *IEEE Transactions on Neural networks* 5 (4)

[Bauer and Kohavi, 1998] Bauer E., Kohavi R. 1999. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. *Machine Learning* 36 (1-2): 105-139.

[Beale and Jackson, 1990] Beale R., Jackson T. 1990. *Neural Computing: An Introduction.* Adam Hilger.

[Ben-Bassat, 1978] Ben-Bassat M. 1978. Irrelevant Features in Pattern Recognition. *IEEE Transactions on Computers* 27 (8): 746-749.

[Ben-Bassat, 1980] Ben-Bassat, M 1980. On the Sensitivity of the Probability of Error Rule for Feature Selection. *IEEE Transaction on PAMI* 2 (1): 56-60.

[Bengio et al., 1995] Bengio Y., Le Cun Y., Nohl C., Burges C. 1995. LeRec: A NN/HMM Hybrid for On-Line Handwriting Recognition. *Neural Computation* (7): 1289-1303.

[Bennani and Bossaert, 1995] Bennani Y. and Bossaert F. 1995. A Neural Network Based Variable Selector. *Intelligent Eng. Systems Through Artificial Neural Networks, (et. Dagli C.H. et al.), ASE Press, NY*.

[Bennani and Gallinari, 1995] Bennani Y., Gallinari P. 1995. Neural Networks for Discrimination and Modelization of Speakers. *Speech Communication* (17).

[Bennani, 1994] Bennani Y. 1994. Multi-expert and Hybrid Connectionist Approach for Pattern Recognition: Speaker Identification Task. *International Journal of Neural Systems* 5 (3).

[Bishop, 1995] Bishop C.M. 1995. *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford, 1995.

[Bloedorn and Michaelski, 1998] Bloedorn E., Michaelski R.S. 1998. Data-Driven Constructive Induction. *IEEE Intelligent Systems & Their Applications Special Issue Feature Transformation and Subset Selection* 30-37.

[Blum and Langley, 1997] Blum A.L., Langley P. 1997. Selection of Relevant Features and Example in Machine Learning. *Artificial Intelligence* 245-271.

[Bollacker and Ghosh, 1996] Bollacker K. D., Ghosh J. 1996. Linear Feature Extractors Based on Mutual Information. *Proceedings of 13th International Conference on Pattern Recognition* 720-724.

[Bonnlander and Weigend, 1994] Bonnlander B. V., Weigend A. S. 1994. Selecting Input Variables Using Mutual Information and Nonparametric Density Estimation. *Proceedings if International Symposium on Artificial Neural Networks ISANN* 42-50.

[Bottoua etal., 1994] Bottou L., Corte C., Denker J.S., Drucker H., Guyon I., Jackel L.D., LeCun Y., Muller U.A., Sackinger E., Simard P., Vapnik V. 1995. Comparison of Classifier Methods: A Case Study in Handwritten Digit Recognition. *Proceedings of the 12th IAPR International Conference on Pattern Recognition* 2: 77-82.

[Breuel and Simon, 1996] Breuel T. M., Simon J.-C. 1996. Report of the Handwriting Working Group. 458-461.

[Brill et al., 1992] Brill F.Z., Brown D. E., Martin W. M. 1992. Fast Genetic Selection of Features for Neural Network Classifiers. *IEEE Transactions on Neural Networks* 3 (2): 325-328.

[Burel et al., 1992] Burel G., Pottier I., Catros J. Y. 1992. Recognition of Handwritten Digits by Image Processing and Neural Network. *Proceedings of IJCNN* 3: 666-671.

[Buturovic, 1991] Buturovic Lj. 1991. Optimalne i suboptimalne transformacije obelezja u sistemima za prepoznavanje oblika (Optimal and Subotpimal Feature Transformation in Pattern Recognition Systems). Doktorska disertacija, Univerzitet u Beogradu, Jugoslavija (PhD thesis, University of Belgrade, Yugoslavia).

[Cakmakov and Radevski, 1998] Cakmakov D., Radevski V. 1998. Strategies for Feature Selection Using Individual Feature Importance. *Proceedings of 18th International Conference "Information Technology Interfaces, ITI'98, Pula, Croatia.*

[Cakmakov and Radevski, 1999] Cakmakov D., Radevski V. 1999. Experiments in Neural Networks Based OCR Using Committee Classifiers. *Proceedings of 21st International Conference "Information Technology Interfaces, ITI'99, Pula, Croatia, June 15-18.*

[Cakmakov et al., 2000] Cakmakov D., Radevski V., Bennani Y. 2000. Experiments in Handwritten Digit Recognition Using Committees of Neural Networks Classifiers. *Proceedings of NNSP 2000, Neural Network for Signal Processing, 11-13 december,2000 Sydney, Australia.*

[Cao et al., 1994] Cao J., Ahmadi M., Shridar M. 1994. Handwritten Numeral Recognition With Multiple Feature and Multistage Classifiers. *IEEE International Symposium Circuits Systems* (6): 323-326.

[Cao et al., 1995] Cao J., Ahmadi M., Shridar M. 1995. Recognition of Handwritten Numerals with Multiple Feature and Multistage Classifier. *Pattern Recognition* 28 (2): 153-159.

[Cao et al., 1997] Cao J., Ahmadi M., Shridhar M. 1997. A Hierarchical Neural Networks Architecture for Handwritten Numeral Recognition. *Pattern Recognition* 30 (2): 289-294.

[Chan and Stolfo, 1995] Chan P., Stolfo C. 1995. Scaling Learnig by Meta Learning Over Disjoint and Partially Replicated Data. *[http://www.cs.columbia.edu/~sal/hpapers/metalrep.ps]*

[Chen et al., 1995] Chen Y.Q., Nixon M. S., Thomas D. W. 1995. Statistical Geometrical Features for Texture Classification. *Pattern Recognition* 28 (4): 537-539.

[Chi and Yan, 1995] Chi Z., Yan H. 1995. Handwritten Numeral Recognition Using a Small Number of Fuzzy Rules With Optimized Defuzzification Parameters. *Neural Networks* 8 (5): 821-827.

[Cibas et al., 1994] Cibas T., Fogelamn F., Gallinari P., Raudys S. 1994. Variable Selection with Optimal Cell Damage. *Proceedings of ICANN'94* (1): 727-730.

[Cibas et al., 1996] Cibas T., Fogelman Soulié F., Gallinari P., Raudys S. 1996. Variable Selection With Neural Networks. *Neurocomputing* 12 (2-3): 223-248.

[Cordella et al., 1999] Cordella L.P., Foggia P., Sansone C., Tortorella F., Vento M. 1999. Reliability Parameters to Improve Combination Strategies in Multi-Expert systems. *Pattern Analysis and Applications* 2: 205-214.

[Davies, 1990] Davies E.R. 1990. *Machine Vision: Theory, Algorithms, Practicalities.* Academic Press, London.

[Deco and Blasig, 1993] Deco G., Blasig R. 1993. Handwritten Digit Recognition with Principal Component Analysis and Radial Basis Functions. *Proceedings of IJCNN* 2253-2256.

[Devijver and Kittler, 1981] Devijver P.A., Kittler J. 1981. *Pattern Recognition: A Statistical Approach.* Prentice Hall International.

[Devroye et al., 1996] Devroye L., Gyorfi L., Lugosi G. 1996. *A Probabilistic Theory of Pattern Recognition.* Springer-Verlag.

[Duda and Hart, 1973] Duda R., Hart P. 1973. *Pattern Classification and Scene Analysis.* Wiley.

[Duda and Hart, 1973] Duda R., Hart P. 1973. *Pattern Classification and Scene Analysis.* Wiley, New York.

[Duerr et al., 1980] Duerr B., Haettich W., Tropf H., Winkler G. 1980. A Combination of Statistical and Syntactical Pattern Recognition Applied to Classification of Unconstrained Handwritten Numerals. *Pattern Recognition* 12: 189-199.

[Fogelman-Soulie et al., 1993] Fogelman-Soulie F., Viennet E., Lamy B. 1993. Multi-modular Neural Networks Architectures: Applications in Optical Character Recognition and Human Face Recognition. *Advances in Pattern Recognition Systems Using Neural Network Technologies (I. Guyon and P.S.P. Wang ed.)* 77-111.

[Foggia et al., 1999] Foggia P., Sansone C., Tortorella F., Vento M. 1999. Definition and Validation of a Distance Measure Between Structural Primitives. *Pattern Analysis and Applications* 2: 215-227.

[Freeman, 1961] Freeman H. 1961. On the Encoding of Arbitrary Geometric Configurations. *IRE Trans. El. Comp.*

[Fu, 1982] Fu K.S. 1982. *Syntactic Pattern Recognition and Applications.* Prentice Hall.

[Fukunaga and Hummels, 1987] Fukunaga K., Hummels D. M. 1987. Bayes Error Estimation Using Parzen and k-NN Procedures. *IEEE Transactions on PAMI* 9 (5): 634-643.

[Fukunaga, 1985] Fukunaga K. 1985. TheEstimation of the Bayes Error by the k-NN Approach. *Progress in Pattern Recognition 2, L.N. KANAL and A. Rosenfled (ed.), Elsevier Science Publishers B.V.* 169-187.

[Fukunaga, 1990] Fukunaga K. 1990. *Introduction to Statistical Pattern Recognition.* Academic Press Inc.

[Fukushima, 1988] Fukushima K. 1988. Neocognitron: A Hierarchical Neural Network Capable of Visual Pattern Recognition. *Neural Networks* 1 (2): 119-130.

[Gazula and Kabuka, 1995] Gazula S., Kabuka M. R. 1995. Design of Supervised Classifiers Using Boolean Neural Networks. *IEEE Transactions on PAMI* 17 (12): 1239-1246.

[Govindan and Shivaprasad, 1990] Govindan V. K., Shivaprasad A. P. 1990. Character Recognition - A Review. *Pattern Recognition* 23 (7): 671-683.

[Greenspan and Godman, 1992] Greenspan H. K., Godman R. 1992. Combined Neural Networks and Rule-Based Framework for Probabilistic Pattern Recognition and Discovery. *Advances in Neural Information Processing Systems (NIPS) In J. E. Moody, S. J. Hanson, and R. P. Lippman (eds.), San Mateo, CA: Morgan Kaufmann Publishers* 4: 445-451.

[Grother, 1993] Grother P.J. 1993. Cross Validation Comparison of NIST OCR Databases. *D.P.D'Almato ed. SPIE, San Jose* 1906.

[Guo and Gelfand, 1992] Guo H., Gelfand S. B. 1992. Classification Trees with Neural Network Feature Extraction. *IEEE Transactions on Neural Networks* 3 (6): 923-933.

[Guorong et al., 1996] Guorong X., Peiqi C., Minhui W. 1996. Bhattacharyya Distance Feature Selection. *Proceedings of the 13th International Conference on Pattern Recognition. IEEE Comput. Soc. Press, Los Alamitos, CA, USA* 2: 195-199.

[Halici and Erol, 1995] Halici U., Erol A. 1995. A Hierarchical Neural Network for Optical Character Recognition. *Proceedings of ICANN `95 Neuronimes `95* (2): 251-256.

[Hansen and Salamon, 1990] Hansen L.K., Salamon P. 1990. Neural Network Ensembles. *Transactions on PAMI* 12 (10): 993-1001.

[Haykin, 1994] Haykin Simon 1994. Neural Networks. Prentice Hall.

[Heutte et al., 1996] Heutte L., Moreau J. V., Paquet T., Lecourtier Y., Olivier C. 1996. Combining Structural and Statistical Features for the Recognition of Handwritten Characters. *Proceedings of 13th International Conference on Pattern Recognition* 210-214.

[Ho et al., 1994] Ho T. K., Hull J. J., Srihari S. N. 1994. Decision Combination in Multiple Classifier Systems. *IEEE Transactions on PAMI* 16 (1): 66-77.

[Hornik, 1989] Hornik K. 1989. Multilayer Feedforward Networks are Universal Approximators. *Neural Networks* (2): 359-366.

[Hu et al., 1997] Hu Y. H., Park J. M., Knoblock T 1997. Committee Pattern Classifiers. *Proceedings of ICASSP '97* 3389-3392.

[Huang and Suen, 1994] Huang Y.S., Suen C.Y. 1994. A Method of Combining Multiple Classifiers - A Neural Network Approach. *Proceedings 12th International Conference Pattern Recognition and Computer Vision, Jerusalem* 473-475.

[Huang et al., 1995] Huang S. Y., Ke Liu, and Suen C.Y. 1995. The Combination of Multiple Classifiers by a Neural Network Approach. *International Journal of Pattern Recognition and Artificial Intelligence* 9 (3): 579-597.

[Hung and Suen, 1995] Hung Y.S., Suen C. Y. 1995. A Method of Combining Multiple Experts for the Recognition of Unconstrained Handwritten Numerals. *IEEE Transactions on PAMI* 17 (1): 90-94.

[Idan et al., 1992] Idan Y., Auger J.M., Darbel N, Sales M., Chevallier R., Dorizzi B., Cazuguel G. 1992. Comparative Study of Neural Networks and Non Parametric Statistical Methods for off-line Handwritten Character Recognition. *Proceedings of the 1992 International Conference Artificial Neural Networks* 2: 1607-1610.

[Jackson, 1996] Jackson Stuart 1996. *Connectionism and Meaning: From Truth Conditions to Weight Representations.* Ablex Publishing Corporation, New Jersey.

[Jain and Dubes, 1988] Jain A.K., Dubes R.C. 1988 *Algorithms for Clustering Data.* Prentice Hall.

[Jain and Zongker, 1997] Jain A., Zongker D. 1997. Feature Selection: Evaluation, Application, and Small Sample Performance. *IEEE Transactions on PAMI* 19 (2): 153-158.

[Jean et al., 1994] Jean J.S.N., Xue K., Goel S. 1994. Pattern Theory for Character Recognition. *IEEE International Conference on Neural Networks. IEEE World Congress on Computational Intelligence, NY, USA* 4198-4203.

[Jordan and Jacobs, 1994] Jordan M., Jacobs A.1994. Hierarchical Mixture of Experts and the EM Algorithm. *Neural Computation* 6:181-214.

[Jouseau and Dorizzi, 1998] Jouseau E., Dorizzi B. 1998. Neural Networks and Fuzzy Data Fusion for On-line and Real Time Vehicle Detection. *Proceedings of the International Conference on Multisource-Multisensor Information Fusion* 2: 695-701.

[Kimura and Shridar, 1991] Kimura F., Shridar M. 1991. Handwritten Numerical Recognition Based on Multiple Algorithms. *Pattern Recognition* 24 (10): 969-983.

[Kimura et al., 1996] Kimura F., Wakabayashi T., Miyake M. 1996. On Feature Extraction for Limited Class Problem. *Proceedings of 13th International Conference on Pattern Recognition* 191-194.

[Kittler et al., 1996] Kittler J., Hatef M., Duin R. P. W. 1996. Combining Classifiers. *Proceedings of 13th International Conference on Pattern Recognition* 897-901.

[Kittler J., 1998] Kittler J. 1998. Combining Classifiers: A Theoretical Framework. *Pattern Analysis and Applications* 1 (1): 18-27.

[Knerr et al., 1992] Knerr S., Personnaz L., Dreyfus G. 1992. Handwritten Digit Recognition by Neural Networks with Single-Layer Training. *IEEE Transactions on Neural networks* 3 (6): 962-968.

[Kohavi and John,] Kohavi R. John G.H. Wrappers for Feature Selection. *Artificial Intelligence Journal, Special Issue on Relevance* 273-324.

[Kojima et al., 1993] Kojima Y., Yamamoto H., Kohda T., Sakaue S., Maruno S., Shimeki Y., Kawakami K., Mizutani M. 1993. Recognition of Handwritten Numeric Characters Using Neural Networks Designed on Approximate Reasoning Architecture. *Proceedings of IJCNN* 2161-2164.

[Koller and Sahami, 1996] Koller D., Sahami M. 1996. Toward Optimal Feature Selection. *Proceedings of the 13th International Conference Machine Learning (ICML'96)* 284-292.

[Kovacs, 1995] Kovacs-V Z.M. 1995. A Novel Architecture for High Quality Hand-Printed Character Recognition. *Pattern Recognition* 28 (11): 1685-1692.

[Kovalevskii, 1976] Kovalevskii V.A. 1976. *Metodi optimalnih resenii v raspoznavanii izobrazenia.* Nauka, Moskva, 1976 (in russian)

[Krishnan et al., 1996] Krishnan S., Samudravijaya K., Rao P. V. S. 1996. Feature Selection for Pattern Classification with Gaussian Mixture models: A New Objective Criterion. *Pattern Recognition Letters* (17): 803-809.

[Lam and Suen, 1994] Lam L., Suen C.Y. 1994. A Theoretical Analysis of the of the Application of Majority Voting to Pattern Recognition. *Proceedings 12th International Conference Pattern Recognition and Computer Vision, Jerusalem* 418-420.

[Lampinen and Smolander, 1996] Lampinen J., Smolander S. 1996. Self-Organizing Feature Extraction in Recognition of Wood Surface Defects and Color Images. *International Journal of Pattern Recognition and Artificial Intelligence* 10 (2): 97-113.

[Lang et al., 1990] Lang K. J., Waibel A. H., Hinton G. E. 1990. A Time-Delay Neural Network Architecture for Isolated Word Recognition. *Neural Networks* 3: 23-43.

[Lavrac et al., 1998] Lavac N., Gamberger D., Turney P. 1998. A Relevancy Filter for Constructive Induction. *IEEE Intelligent Systems & Their Applications Special Issue Feature Transformation and Subset Selection* 50-56.

[Le Cun and Bengio, 1995] Le Cun Y., Bengio Y. 1995. *Pattern Recognition and Neural Networks.* In M. A. Arbib, editor, The Handbook of Brain Theory and Neural Networks. MIT Press

[Le Cun et al., 1989] Le Cun Y., Boser B., Denker J. S., Henderson D., Howard R. E., Hubbard W., Jackel L. D. 1989. Backpropagation Applied to Handwritten Zip Code. *Neural Computation* (1): 541-551.

[Le Cun et al., 1990] Le Cun Y., Boser B., Denker J.S., Henderson D., Howard R.E., Hubbard W., Jackel L.D. 1990. Handwritten Digit Recognition with a Back-Propagation Network, in Advances in Neural Information Processing Systems 2, ed. Touretzky D., Morgan Kaufman, 396-404.

[Le Cun et al., 1992] Le Cun Y., Boser B., Denker J. S., Henderson D., Howard R. E., Hubbard W., Jackel L. D. 1992. Handwritten Digit Recognition with a Back-Propagation Network. In Lisboa P.G.J., editor, *Neural Networks, current applications*, Chappman and Hall 396-404.

[Lee and Srihari, 1995] Lee D.S., Srihari S.N. 1995. A Theory of Classifier Combination: The Neural Network Approach. *Proceedings 3rd ICIDAR* 42-45.

[Lee, 1996] Lee S.W. 1996. Off-Line Recognition of Totally Unconstrained Handwritten Numerals Using Multilayer Cluster Neural Networks. *IEEE Transactions on PAMI* 18 (8): 648-652.

[Lim et al., 1994] Lim J.., Eelwan L., Sooh I.C. 1994. Character Recognition by Neural Networks with Single-Layer Training and Rejection Mechanism. *IEEE International Symposium on Circuits an Systems* 6: 327-330.

[Lincoln and Skrzypek, 1990] Lincoln W. P., Skrzypek J. 1990. Synergy of Clustering Multiple Back Propagation Networks. *Advances in Neural Information Proceedings System* (2): 650-657.

[Liu et al., 1998] Liu H., Motoda H., Dash M. 1998. A Monotonic Measure for Optimal Feature Selection. Claire Nédellec, Céline Rouveirol (Eds.) *Machine Learning: ECML-98*, 10th European Conference on Machine Learning, Chemnitz, Germany, April 21-23, 1998, Proceedings. Lecture Notes in Computer Science, Springer 1398 101-106.

[Lou et al., 1999] Lou Z., Liu K., Yang J.Y., Suen C.Y. 1999. Rejection Criteria and Pairwise Discrimination of Handwritten Numerals Based on Structural Features. *Pattern Analysis and Applications* 2: 205-214.

[Lu and Ito, 1997] Lu B. L., Ito M. 1997. Task Decomposition on Class Relations: A Modular Neural Networks Arch for Pattern Classification. *Biological and Artificial Computation: From Neuroscience to Technology Lecture Notes in Computer Science* 1240: 330-339.

[Mandler and Schurmann, 1988] Mandler E., Schurmann J. 1988. Combining the classification Results of Independent Classifiers Based on the Dempster-Shafer Theory of Evidence. *Pattern Recognition and Artificial Intelligence, North Holland. Elsevier Science Publishers B.V.* 381-393.

[Mantas, 1987] Mantas J. 1987. Methodologies in Pattern Recognition and Image Analysis - A Brief Survey. *Pattern Recognition* 20 (1): 1-6.

[Mao and Jain, 1995] Mao J., Jain A. K. 1995. Artificial Neural Networks for Feature Extraction and Multivariate Data Projection. *IEEE Transactions on Neural Networks* 6 (2): 296-317.

[Marjanovic et al., 1994] Marjanovic M., Tomovic R., Stankovic S. 1994. A Topological Approach to Recognition of Line Figures. *Bulletin T.CVII de l'Académie Serbe des Sciences et des Arts - 1994, Sciences mathématiques N°19* 43-64.

[Meisel, 1972] Meisel Wiliam 1972 *Computer-oriented Approaches to Pattern Recognition.* Academic Press

[Meng et al.,1994] Meng Y., Junbo F., Fan J. 1994. A Hybrid Method for Recognizing Handwritten Numbers. *? IEEE* 4269-4271.

[Merriam-Webster, 2000] 2000. Merriam-Webster Online Dictionary *[http://www.m-w.com]*

[Merz and Pazzani, 1999] Merz C. J., Pazzani M.J. A Principal Components Approach to Combining Regression Estimates 1999. *Machine Learning 36(1-2):9-32.*

[Milgram, 1993] Milram M. 1993. *Reconnaissance des formes: méthodes numériques et connexionnistes*, ed. Armand Colin.

[Milosavljevic and Radevski, 1996] Milosavljevic M., Radevski V. 1996. Clustering of Handwritten Cyrillic Characters. In *Grouping Analysis II*, Bogosavljevic S., Kovacevic M. ed., Federal Statistics Institute, Belgrade, Yugoslavia

[Nadal et al., 1990] Nadal C., Legault R., Suen C.Y. 1990. Complementary Algorithms for the Recognition of Totally Unconstrained Handwritten Numerals. *Proceedings 10th International Conference Pattern Recognition, Atlantic City, N.J.* 443-449.

[Nakanishi and Fukui, 1993] Nakanishi I., Fuku Y. 1993. Pattern recognition Using Hierarchical Feature Type and Location. *Proceedings of IJCNN* 2165-2168.

[Narendra and Fukunaga, 1977] Narendra P.M. and Fukunaga 1977. A Branch and Bound Algorithm for Feature Subset Selection. *IEEE Transactions on Computer.* 26: 917-922.

[Nirmalya et al.,] Nirmalya Chowdhury, Murthy C. A., Pal S. K Cluster Detection Using Neural Networks. *INSPEC* 2166-2169.

[Nishida, 1995] Nishida H. 1995. Structural Feature Extraction Using Multiple Bases. *Computer Vision and Image Understanding* 62 (1): 78-89.

[Nishida, 1996] Nishida H. 1996. Analysis and Synthesis of Deformed Patterns Based on Structural Models. *Proceedings of 13th International Conference on Pattern Recognition* 315-319.

[NIST, 1992] Nist Special Database 3 Replaced by Nist Special Handprinted Forms and Characters Database 19. http://www.nist.gov/srd/nistsd19.htm

[Novovicova et al., 1996] Novovicova J., Pudil P., Kittler J. 1996. Divergence Based Feature Selection for Multimodal Class Densities. *IEEE Transactions on PAMI* 18 (2): 218-222.

[Opitz and Shavlik, 1995] Opitz D.W., Shavlik J.W. 1995. Generating Accurate and Diverse Members of a Neural-Network Ensemble. In Touretzky D.S., Mozer M.C., Hasselmo M.E. ed. *Advances in Neural Information Processing Systems, 8.* MIT Press, MA. 535-541.

[Pao, 1989] Pao Y.-H. 1989. *Adaptive Pattern Recognition and Neural Networks.* Addison-Wesley

[Pao, 1993] Pao, Y.H. 1993. *Neural Network Computing for Pattern Recognition.* Handbook of Pattern Recognition and Computer Vision, Eds. C.H. Chen, L.F. Pau, P.S.P. Wang 125-162.

[Parberry, 1995] Parberry I. 1995. Structural Complexity and Discrete Neural Networks. *The Handbook of Brain Theory and Neural Networks*, (Michael Arbib, Ed.), MIT Press 945-948.

[Parizeau and Plamandon, 1995] Parizeau M., Plamandon R. 1995. A Fuzzy-Syntactic Approach to Allograph Modeling for Cursive Script Recognition. *IEEE Transactions on PAMI* 17 (7): 702-712.

[Pavlidis, 1980] Pavlidis T. 1980. *Structural Pattern Recognition.* Springer-Verlag (second edition)

[Pudil and Novovicova, 1998] Pudil P., Novovicova J. 1998. Novel Methods for Subset Selection with Respect to Problem Knowledge. *IEEE Intelligent Systems & Their Applications Special Issue Feature Transformation and Subset Selection* 66-74.

[Pudil et al., 1994] Pudil P., Novovicova J., Kittler J. 1994. Floating Search Methods in Feature Selection. *Pattern Recognition Letters* (15): 1119-1125.

[Pudil et al., 1995] Pudil P., Novovicova J., Choakjarernwanit N., Kittler J. 1995. Feature Selection Based on the Approximation of Class Densities by Finite Mixtures of Special Type. *Pattern Recognition* 28 (9): 1390-1397.

[Radevski and Bennani, 1997] Radevski V., Bennani Y. 1997. Combining Structural and Statistical Features for Handwritten Digit Recognition. *Proceedings of 2nd International Conference on Computational Intelligence and Neuroscience, Research Triangle Park* (2): 102-105.

[Radevski and Bennani, 1997a] Radevski V., Bennani Y. 1997. Committee Neural Classifiers for Structural and Statistical Features Combination. *Proceedings of ANNIE'97 Artificial Neural Networks In Engineering, Missouri, USA*

[Radevski and Bennani, 1998] Radevski V., Bennani Y. 1998. A Higher Recognition Reliability Through Combining Classifiers. *XLII Conference for Electronics, Telecommunications, Computers, Automation and Nuclear Engineering (IEEE section of Yugoslavia), June 2-5, Vrnjacka Banja, Yugoslavia*

[Radevski and Bennani, 2000] Radevski V., Bennani Y. 2000. Reliability Control in Committee Classifier Environment. *IJCNN International Joint Conference on Neural Networks, 2000*

[Radevski et al., 2000] Radevski V., Bennani Y., Cakmakov D. 2000. A Reliability Improvement of Neural Networks Based OCR Using Rules and Committee Classifiers. *International Conference on Information Technology Interfaces, Pula, Croatia, June 13-16*

[Radevski, 1995] Radevski V. 1995. *Prepoznavanje rukom pisanih znakova statisticko-sintaktickim metodama prepoznavanja oblika (Handwritten Character Recognition Applying Statistical and Syntactical Methods).* Magistarska teza, Univerzitet u Beogradu, Jugoslavija (Master of Science Thesis, University of Belgrade, Yugoslavia)

[Ramdas et al., 1994] Ramdas V., Sridhar V., Krishna G. 1994. An Effective Clustering Technique for Feature Extraction. *Pattern Recognition Letters* (15): 885-891.

[Rastrigin and Erenstein, 1981] Rastrigin L.A., Erenstein R.H. 1981. *Metod kollektivnogo raspoznavania.* Energoizdat, Moscow (in russian)

[Raudys and Pikelis, 1980] Raudys S., Pikelis V. 1980. On Dimensionality, Sample Size, Classification Error, and Complexity of Classification Algorithm in Pattern Recognition. *IEEE Transactions on PAMI* 2 (3): 242-252.

[Rocha and Pavlidis, 1995] Rocha J., Pavlidis T. 1995. Character Recognition Without Segmentation. *IEEE Transactions on PAMI* 17 (9): 903-909.

[Roli et al., 1995] Roli F., Serpico B., G.Vernazza 1995. Image Recognition by Integration of Connectionist and Symbolic Approaches. *International Journal of Pattern Recognition and Artificial Intelligence* 3 (3): 485-515.

[Rui-Ping et al., 1996] Li R.-P., Mukaidono M., Turksen I.B. 1996. Study on Feature Weight and Feature Selection in Pattern Classification Neural Networks. *IEEE International Conference on Systems, Man and Cybernetics. Information Intelligence and Systems* 3: 1972-6.

[Schurmann, 1996] Schrmann J. 1996. *Pattern Classification.* John Wiley & Sons, Inc.

[Shustorovich et al., 1996] Shustorovich A., Thrasher C. W. 1996. Neural Network Positioning and Classification of Handwritten Characters. *Neural Networks* 9 (4): 685-693.

[Siddiqui et al., 1994] Siddiqui K.J., Liu Y. –H ., Hay D. R., Suen C. Y. 1994. Feature Selection Using a Proximity-index Optimization Model. *Pattern Recognition Letters* (15): 1137-1141.

[Siedlecki and Sklansky, 1988] Siedlecki W., Sklansky J. 1988. On Automatic Feature Selection. *International Journal of Pattern Recognition and Artificial Intelligence* 2 (2): 197-220.

[Simard et al.,1993] Simard P., Le Cun Y., Denker J. 1993. Efficient Pattern Recognition Using a New Transformation Distance. *Advances in Neural Information Processing Systems, Morgan Kaufmann, San Mateo, CA* 50-58.

[Skrzypek and Hoffman, 1991] Skrzypek J., Hoffman J. 1991. Visual Recognition of Script Characters and Neural Network Architectures. *Neural Networks: Advances and Applications, E. Gelenbe (Editor)* 109-144.

[Solaiman and Autret, 1991] Solaiman B., Autret Y. Association d'une methode connexionniste et d'une methode structurelle pour la reconnaissance de chiffres manuscrits.*8ème congrès, Reconnaissance de Formes et Intelligence Artificielle, RFIA91, 25-29 Nov., Lyon* 719-728.

[Sridhar et al., 1999] Sridhar D.V., Barlett E.B., Seagrave R.C. 1999. An Information Theoretic Approach for Combining Neural Network Process Models. *Neural Networks* 12 (6): 915-926.

[Steppe et al., 1996] Steppe J.M., Bauer K. W., Rogers S. T. 1996. Integrated Feature and Architecture Selection. *IEEE Transactions on Neural Networks* 7 (4): 1007-1014.

[Stromberg et al., 1991] Stromberg J.E., Zrida J ., Isaksson A. 1991. Neural Trees - Using Neural Nets in a Tree Classifier Structure. *IEEE International Conference on Acoustics, Speech and Signal Processing (Toronto)* 137-140.

[Subramanian et al., 1997] Subramanian D., Greainer R., Pearl J. 1997. The Relevance of Relevance. *Artificial Intelligence* 1-5.

[Suen et al., 1993] Suen C., Legault R., Nadal C., Cheriet M., Lam L. 1993. Building a New Generation of Handwriting Recognition Systems. *Pattern Recognition Letters* 14 (4): 303-315.

[Suzuki et al., 1995] Suzuki T., Nishida H., Nakajima Y., Yamagata H., Tachikawa M., Sato G. 1995. A Handwritten Character Recognition System by Efficient Combination of Multiple Classifiers. *International Association for Pattern Recognition Workshop on Document Analysis Systems, World Scientific, Singapore* 169-187.

[Takahashi, 1991] Takahashi H. 1991. A Neural Net OCR Using Geometrical and Zonal-Pattern Features. *Proceedings of the First International Conference on Document Analysis and Recognition Saint-Malo, France* 821-828.

[Tanaka, 1995] Tanaka E., 1995. Theoretical Aspects of Syntactic Pattern Recognition. *Pattern Recognition* 28 (7): 1053-1061.

[Theodoridis and Koutroumbas, 1999] Theodoris S., Koutroumbas K. 1999. *Pattern Recognition.* Academic Press

[Thompson, 1978] Thompson M.L. 1978. Selection of Variables in Multiple Regression: A Review and Evaluation. *International Statistical Review* 46: 1-19.

[Toraichi et al., 1996] Toraichi K., Kumamoto T., Yamamoto K. Yamada H. 1996. Feature Analysis of Handprinted Chinese Characters. *Pattern Recognition Letters* (17): 795-800.

[Toussaint, Donaldson, 1970] Toussiant G., Donaldson R. 1970. Algorithms for Recognizing Contour-Traced Handprinted Characters. *IEEE Trans.Comp., Jun.*

[Trier et al., 1996] Trier D.O., Jains K.A., Taxt T. 1996. Feature Extraction Methods for Character Recognition - Survey. *Pattern Recognition* 29 (4)

[Tumer and Ghosh, 1995] Tumer K., Ghosh J. 1995. Theoretical Foundations of Linear and Order Statistics Combiners for Neural Pattern Classifiers. *TR-95-02-98, The Computer and Vision Research Center, The University of Texas at Austin* 35p.

[Tumer and Ghosh, 1996] Tumer K., Ghosh J. 1996. Error Correlation and Error Reduction in Ensemble Classifiers. *INTERNET/ps* 1-29.

[Tumer and Ghosh, 1999] Tumer K., Ghosh J. 1999. Linear and Order Statistics Combiners for Pattern Classification, A.Sharkey ed. *Combining Artificial Neural Nets*, Springer-Verlag, 127-162

[Vafaie and De Jong, 1998] Vafaie H., De Jong K. 1998. Feature Subset Selection Using a Genetic Algorithm. *IEEE Intelligent Systems & Their Applications Special Issue Feature Transformation and Subset Selection* 44-49.

[Wang et al., 1996] Wang S., Zhu X., Jin Y. 1996. Multiple Experts Recognition System Based on Neural Network. *Proceedings of 13th International Conference on Pattern Recognition* 452-456.

[Williams, 1994] Williams C.K.I. 1994. Combining Deformable Models and Neural Network for Handprinted Digit Recognition. *Technical Report CRG-TR-94-2, University of Toronto*

[Wilson and Blue, 1993] Wilson C.L., Blue J.L. 1992. Neural Network Methods Applied to Character Recognition. *Soc. Sci. Comput. Rev* 10: 173-195.

[Wilson et al., 1995] Wilson C.L., Grother P. J., Barnes C. S. 1995. Binary Decision Clustering for Neural Networks Based OCR. *Pattern Recognition* 29 (3): 425-437.

[Woods et al., 1997] Woods K., Kegelmeyer W. P., Bowyer K. 1997. Combination of Multiple Classifiers Using Local Accuracy Estimates. *IEEE Transactions on PAMI* 19 (4): 405-410.

[Xu et al., 1992] Xu L., Krzyzak A., Suen C. Y. 1992. Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition. *IEEE Trans on Systems, Man and Cyb.* 22 (3): 418-435.

[Yan, 1995] Yan H. 1995. Comparison of Multilayer Neural Networks and Nearest Neighbor Classifiers for Handwritten Digit Recognition. *International Journal of Neural Systems* 6 (4): 417-423.

[Yang and Liou, 1996] Yang H-C, Liou C-Y. 1996. Handprinted Character Recognition Based on Spatial Topology Distance Measurement. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18: 941-945.

[Zadeh, 1977] Zadeh L.A. 1977. Fuzzy Sets and Their Application to Classification and Clustering. *Classification and Clustering (J. Van Ryzin ed.),* Academic Press, New York 251-299.

[Zhang et al., 1996] Zhang M., Suen C. Y., Bui T. D. 1996. Feature Extraction in Character Recognition with Associative Memory Classifier. *International Journal of Pattern Recognition and Artificial Intelligence* 10 (4): 325-347.

[Zongker and Jain, 1996] Zongker D., Jain A. 1996. Algorithms for Feature Selection: An Evaluation. *Proceedings of 13th International Conference on Pattern Recognition* 18-22.

[Zupan et al., 1998] Zupan B., Demsar J., Bratko I. 1998. Feature Transformation by Function Decomposition. *IEEE Intelligent Systems & Their Applications Special Issue Feature Transformation and Subset Selection* 38-43.