FEATURE SET EVALUATION FOR A GENERIC MISSILE DETECTION SYSTEM

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

K. SELÇUK AVAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

FEBRUARY 2007

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Canan ÖZGEN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. İsmet ERKMEN
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. T. Engin TUNCER
Supervisor

Examining Committee Members

Assoc. Prof. Dr. Gözde BOZDAĞI AKAR	(METU, EE)	
Assoc. Prof. Dr. T. Engin TUNCER	(METU, EE)	
Assoc. Prof. Dr. Aydın ALATAN	(METU, EE)	
Assist. Prof. Dr. Afşar SARANLI	(METU, EE)	
Çağrı ACAR	(ASELSAN)	

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : K. Selçuk AVAN

Signature :

ABSTRACT

FEATURE SET EVALUATION FOR
A GENERIC MISSILE DETECTION SYSTEM

AVAN, Kazım Selçuk

M.S., Department of Electrical and Electronics Engineering

Supervisor: Assoc. Prof. Dr. T. Engin TUNCER

February 2007, 128 pages

Missile Detection System (MDS) is one of the main components of a self-protection system developed against the threat of guided missiles for airborne platforms. The requirements such as time critical operation and high accuracy in classification performance make the 'Pattern Recognition' problem of an MDS a hard task. Problem can be defined in two main parts such as 'Feature Set Evaluation' (FSE) and 'Classifier' designs. The main goal of feature set evaluation is to employ a dimensionality reduction process for the input data set, while not disturbing the classification performance in the result. In this thesis study, FSE approaches are investigated for the pattern recognition problem of a generic MDS.

First, synthetic data generation is carried out in software environment by employing generic models and assumptions in order to reflect the nature of a realistic problem environment. Then, data sets are evaluated in order to draw a baseline for further feature set evaluation approaches.

Further, a theoretical background including the concepts of Class Separability, Feature Selection and Feature Extraction is given. Several widely used methods are assessed in terms of convenience for the problem by giving necessary justifications depending on the data set characteristics.

Upon this background, software implementations are performed regarding several feature set evaluation techniques. Simulations are carried out in order to process dimensionality reduction. For the evaluation of the resulting data sets in terms of classification performance, software implementation of a classifier is realized. Resulting classification performances of the applied approaches are compared and evaluated.

Keywords: Pattern Recognition, Feature Extraction, Feature Selection, Class Separability, Classification.

ÖZ

JENERİK BİR FÜZE TESPİT SİSTEMİNE İLİŞKİN
ÖZNİTELİK KÜME DEĞERLENDİRMESİ

AVAN, Kazım Selçuk

Yüksek Lisans Tezi, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Doç. Dr. T. Engin TUNCER

Şubat 2007, 128 sayfa

Füze Tespit Sistemi, hava platformları için güdümlü füze tehdidine karşı geliştirilen kendini koruma sistemlerinin ana bileşenlerinden biridir. Zaman kritik işlev ve sınıflandırma başarımında yüksek doğruluk oranı gibi gereksinimler, bir füze tespit sisteminin 'Örüntü Tanıma' problemini güçleştirmektedir. Problem, 'Öznitelik Küme Değerlendirmesi' (ÖKD) ve 'Sınıflandırıcı' tasarımı olarak iki ana kısımda tanımlanabilir. ÖKD'nin başlıca amacı, sonuçtaki sınıflandırıcı başarımını düşürmeksizin girdi olan veri kümesi için bir boyut indirgeme süreci yürütmektir. Bu tez çalışmasında, jenerik bir füze tespit sistemine ilişkin örüntü tanıma problemi için ÖKD yaklaşımları araştırılmıştır.

İlk olarak, jenerik modeller ve problem ortamının gerçekçi doğasının yansıtılabilmesi için gerekli varsayımlar ile yazılım ortamında yapay veri üretimi gerçekleştirilmiştir. Ardından, ileriki öznitelik küme değerlendirmesi yaklaşımları için bir temel oluşturması amacıyla veri kümeleri değerlendirilmiştir.

Ayrıca, 'Sınıf Ayrılabilirliği', 'Öznitelik Seçimi' ve 'Öznitelik Çıkarımı' kavramlarını temel alan bir kuramsal zemin sunulmuştur. Yaygın olarak kullanılan birkaç yöntem, problem için uygunlukları açısından veri kümesi özelliklerine dayalı gerekçeler gösterilerek değerlendirilmiştir.

Bu zemin üzerinde, birkaç öznitelik küme değerlendirmesi yönteminin yazılım gerçeklemeleri yapılmıştır. Veri setlerinin boyut indirgeme sürecini gerçekleştirmek amacıyla, yazılım benzetimleri gerçekleştirilmiştir. Sonuç olarak elde edilen veri kümelerinin sınıflandırma başarımları açısından değerlendirilmesi amacıyla da, bir sınıflandırıcı yazılım gerçeklemesi yapılmıştır. Uygulanan yaklaşımların sınıflandırma başarımları karşılaştırılarak değerlendirilmiştir.

Anahtar Kelimeler: Örüntü Tanıma, Öznitelik Çıkarımı, Öznitelik Seçimi, Sınıf Ayrılabilirliği, Sınıflandırma

To My Family,

Zeliş

and

My Grandfather

ACKNOWLEDGMENTS

I would like to thank Assoc. Prof. Dr. T. Engin Tuncer for his support and supervision throughout the development of this thesis study.

I should thank to my colleagues for their tolerances and for their smiling faces especially for the times under stress. I am grateful to ASELSAN Electronics Industries Inc. for the resources and facilities that I utilize throughout the study.

Thanks a lot to all of my close friends for their support for my motivation, whether intentionally or not, and their valuable help to complete this work.

And, I am grateful to my Zeliş, for her generous heart full of tolerance and love. There is no way I can forget the sacrifice of postponing future plans due to the completion of this study.

Finally, I would like to thank to my family for their great encouragement. Trust and always-caring hearts of my parents helped me keep the necessary determination alive and I could not survive without my little-mummy sister Zeynep's accompaniment late at nights during the lifetime of this study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AAC : Atmospheric Attenuation Coefficient

BAFS : Bhattacharyya Feature Selection

DBFE : Decision Boundary Feature Extraction

FA : False Alarm

FE : Feature Extraction

FG : Feature Generation

FOV : Field Of View

FS : Feature Selection

FSE : Feature Set Evaluation

IR : Infra Red

MAW : Missile Approach Warner

MLW : Missile Launch Warner

MDS : Missile Detection System

MWS : Missile Warning System

OS : Overlap Sum

PCA : Principal Component Analysis

PCAFE : Principal Component Analysis Feature Extraction

POCD : Probability of Correct Declaration

PR : Pattern Recognition

SMFS : Scatter Matrices Feature Selection

UV : Ultra Violet

CHAPTER I

INTRODUCTION

One main component of a self-protection system developed against the threat of guided missiles for airborne platforms is the Missile Detection System (MDS). The aim of an MDS is to trigger a warning to an engaging missile by discriminating the threat from the background clutter. Due to the criticality of the job of such systems, high probability of correct threat detection is desired. In addition, the warning is required to be generated as early as possible in order to have adequate time to activate a counter measure. As well as high accuracy in detection probability and high performance in its time critical nature, the rate of incorrect warnings is desired to be low, considering the deployment limitations of countermeasure systems. These hard tasks make the classifier design of such a system a difficult Pattern Recognition problem.

1.1 Problem Definition

The time critical characteristic of the classification problem of an MDS requires a certain computational power for fast response. However, there exist some certain constraints that might put a limit to the availability of computational power. These constraints mainly stem from the fact that an MDS is a system to be installed on an airborne platform. Due to the volume and weight restrictions, cooling might be a problem for high performance processing. In addition, strict environmental requirements might employ some certain limits to the hardware design of the system, which is inevitably dependent to the performance.

Since always a limit exists regarding the computational power especially for such real-time airborne systems, the other alternative for meeting the time requirements is the ‘data reduction’. Data reduction can be expressed in terms of reduction in the dimensionality with no (or in an acceptable amount of) loss of classification performance. The data reduction stage in the classification system design is realized by applying Feature Extraction approaches that are convenient with the nature of the problem. Hence, a systematic approach should be applied in the Feature Extraction stage taking the data characteristics of the problem into consideration.

In this thesis study, Feature Set Evaluation approaches are studied for a generic MDS classification problem, by utilizing synthetically generated data accordingly.

The problem is introduced in two different points of view: Electronic Warfare (EW) domain and in Pattern Recognition (PR) domain, before proceeding deeper into the details of the study. Handling the problem in the EW domain will briefly address the problem in the EW literature. Later, looking into the problem from the PR domain in short will give an insight to the abstraction of problem, and the approach for the solution.

1.1.1 View of the Problem in Electronic Warfare Domain

The struggle against varying forms of threats under the scope of Electronic Warfare (EW) requires continuous development of approaches in protection systems. The threat of guided missiles has brought the development of self-protection systems for airborne platforms. Missile Launch Warner (MLW), Missile Approach Warner (MAW), Missile Warning System (MWS) or Missile Detection System (MDS) is the unit that is responsible for generating warnings to threatening missiles. As the naming convention, MDS is used throughout the thesis, however since it is also a widely used name in EW literature, MWS is also used occasionally. It is possible to reach various definitions regarding the MDS in EW literature.

Missile Launch Warner (MLW) is a system capable of detecting the launch of a missile if it occurs within their Field of View (FOV) and detection range. These systems are designed to be installed on board aircraft that may be subject to sudden attack by IR-Guided missiles. Since the latter can be launched without any radio frequency emission, the only means of detecting their launch with any reliability is to exploit devices capable of detecting their IR emissions or, sometimes, UV emissions [1].

The missile defense system usually employs a set of sensors whose function is to provide warning of the terminal phase of a missile attack. The sensors range from passive Radar Warning Receivers (RWRs) and IR detectors to active missile warning radars [3].

As seen above, MDSs can be categorized into two: passive (non-radiating) and active systems (radars). Some older references mentions about the passive systems, as they have been on the development stage in late 1980's. In [1] it is stated: "IR systems capable of providing information about approaching missiles are under study, but at present, radar techniques remain the best tools for performing this function."

The generic MDS model used in the thesis study is under the passive systems category. Passive systems are relatively new to active systems. They are electro-optical non-radiating systems that detect and process the target's plume signature in IR or UV band.

The threat of man portable (man-pad) passive infrared homing surface-to-air (SAM) missiles has led to the production of passive missile warning sensors, which are used to trigger an alarm when an airborne platform is engaged by a missile [2].

Since passive IR guidance is known to be the oldest guidance technique, one can basically discuss that anti A/C missiles worldwide mainly would have passive IR guidance. Supporting this argument, in [5] it is stated that A/C losses tabulated in 1990 for the 1977-1985 time span indicated that 90 percent of these losses were due to IR-guided missiles. RF-guided missiles, on the other hand, accounted for 4.4 percent of the losses. Hence, taking this majority of IR threats into consideration, they cannot be detected by a RWR or a Laser Warning Receiver (LWR). This fact emphasizes the importance of passive MWS. In [4], the importance of the development of passive MWS against highly mobile, easily concealed Shoulder Launched Surface to Air Missiles (SAMs) as a recent threat is discussed.

Another advantage of passive MWS is that it does not betray its presence with emission [5]. In the generic data generation process part of the thesis study, SAM engagement scenarios were simulated, which will be given in detail later in the data generation section.

So far, it is justified that why passive MWS (or MDS) is preferable than active type. To have a further detailed insight about the passive MWS, a comparison of the types of passive MWS (IR and UV) can be found in [6]. Below in Table 1-1, the comparison is summarized for passive MWS.

In short, most apparent difference between IR and UV MWS is the background clutter. Since UV has minimal background clutter due to the existence of less amount of man made clutter sources than in the IR band, relatively simpler algorithms are applied in processing. Due to this simplicity reasons, data generation is performed under the assumption of generic UV passive MWS for implementation and experiments in this study.

Table 1-1 A comparison of IR and UV passive MWS systems

IR	IR detection typically in 3-5um band; either mechanical scanning or Focal Plane Arrays (FPAs).	
	Strengths	Detects both plume emission and hot engine parts, including Post Burn Out (PBO) detection; lower atmospheric attenuation; high angular separation of targets.
	Weaknesses	Strongly clutter limited performance, risk for saturation at short ranges due to need for high sensitivity to provide long-range detection, complexity due to need for cooling.
UV	UV detection of missile plume in the solar-blind region at 0.2-0.3um; built around an image intensifier.	
	Strengths	Minimal background clutter, hence lower demand on signal processing and reduced complexity; no cooling required; mature technology; lower cost.
	Weaknesses	PBO detection, restricted detection range due to ozone attenuation, UV clutter from man-made sources, requirement for sharp cut-off filter at approx. 0.29um, possibility that UV threat decoys will be developed.

1.1.2 View of the Problem in Pattern Recognition Domain

The task of the classifier of an MDS (or MWS) is simply to distinguish the signal derived from the signature of a missile from the signal of clutter. As obvious, this decision has to be made in real-time, therefore processing is time critic. In this two-class (missile and non-missile) classification problem,

feature vectors that are to be extracted via detection and tracking modules can have high dimensionality.
As the dimensionality increases, the increase in the computational overhead is often exponential. This drastic increase is called as the well-known term, "curse of dimensionality" [17]. Restrictions on the availability of computational power were discussed formerly, so another way of handling high dimensionality is to be employed. Feature Extraction (FE) techniques, in general can be applied to lower the dimensionality, even some features were primarily generated by some other means of feature extraction methods.

Thus, FE techniques can be considered as a form of data compression whose objective is to minimize the consequences reducing the dimensionality on class separability [7].

There exists a wide range spectrum of studies relating the subject of FE in Pattern Recognition literature. Generally, methods are specified to the characteristics of problems and successfully applied in accordance with the addressing of the problem.

Indeed there is one major point behind FE methods, which is well summarized in [7]: The resulting features from a FE technique do not often provide the 'optimal set' regarding the classification performance. In addition, the difficulty in extracting discriminative features from finite data sets increases as the dimensionality of the data grows. Study to be performed to find out this optimal (or sub-optimal in some cases) set for classification is named as 'Feature Set Evaluation' (FSE), which will be used also in this thesis.

FSE includes ways of studying feature set ranking and evaluation of feature sets according to class separability criterion. Methods are evaluated and applied by necessary customizations depending on the characteristics of the data and the nature of the problem. In this study, FE methods with convenient classifier adaptation are applied by the evaluation of the data. Beyond, implementations, outcome evaluations and comparisons are performed accordingly as a systematic FSE approach for the classification problem of a generic MDS.

1.2 Motivation

It is becoming necessary for all manned airborne combat platforms to be equipped with a reliable MWS [1]. However, the winner of the outgoing MWS vs. IR guided missile struggle is far from being resolved [5]. This nature of the subject as a real-life problem is one side of the motivation for the interest in the topic of MWS (or MDS) as a self-protection system.

The design of the FE and classification stages plays a critical role in the performance of MDS. As the threats and platforms change in time, characteristic of the data changes. In addition, improvements in detector and processor technologies tend to result in more complexity of the data, which can be considered as high dimensionality looking from the algorithm window. In order to mitigate the complexity, the need for smart FE increases. As a consequence of the diversity of FE techniques to be applied, a set of candidate features is generated. The evaluation and ranking of these feature sets for optimizing the classification performance brings the need for studying Feature Set Evaluation (FSE) techniques.

The diversity of the approaches in FE and FSE techniques shows that there is indeed no unified theory, which alone fits well for all pattern recognition problems [7].

In conclusion, FE and classification will always remain as the topics to be studied for development and adaptation to data characteristics. The place of the problem in Pattern Recognition constitutes the other side of the motivation for this thesis study.

1.3 Aim of the Study

Main objective of the study is basically to investigate an answer to the question regarding our generic MDS problem: "How will the 'right' features be selected/extracted among the candidates, so that one can be sure of optimizing his classifier to operate for the best possible class separability at certain dimensionality?"

More specifically, FSE is to be applied over the resulting feature sets of various FE techniques under class seperability criteria and finally find out the most convenient approaches for our problem.

1.4 Outline of the Thesis

This thesis is organized as follows. Chapter I is intended to be given as a background covering the subject from both EW and PR points of view by presenting the motivation for the study.

In Chapter II, Data Generation process performed in the implementation stage of the study will be described. The assumptions and necessary adaptations made during the data generation are given.

In Chapter III, data set characteristic is evaluated. Synthetic data generation is assessed for its correspondence to the real data. Understanding of the data characteristics will constitute the necessary background for further FSE approaches.

In Chapter IV, fundamentals of FE are introduced. Class separability measures and their relations with the error probability is presented. In following, details of the approaches applied are given together with related justifications for the selection of them. This is the main part of the study.

In Chapter V, appropriate classifier design approaches are discussed by taking the data of the problem into consideration.

Chapter VI is the review and evaluation of obtained results by making observations and comparisons over them.

Finally in Chapter VII, a brief summary of the study is given. Conclusions are derived relating the observations and evaluations of the results.

CHAPTER II

DATA GENERATION

Main parameter that defines the classification design of a system is the nature of the data set in the problem. It is obvious that working with real data for a real-life problem is essential. However, it is often not so trivial to obtain real data in satisfactory variety for several reasons such that the difficulties in constructing necessary environment to collect real data or high amount of cost. In order to cope with the lack of realistic data availability especially for EW problems, running simulations and generating synthetic data is the preferable alternative. In this case, the issue that how the synthetic data represents the characteristic of the real data is the main topic to be discussed. In this section, the synthetic data generation process in this study is described.

2.1 General Properties

The event detected and tracked that corresponds to a single detector output of the signature observed by our generic MDS sensor will be called as a '**signal**' in the following of the study. A signal is defined by its '**primitive features**', which are '**position**', the signal '**size**' on the detector screen and the digital '**intensity**' level of the detector. Position is defined as a couple of angles in polar coordinate system in synthetic data generation process, as '**azimuth**' and **'elevation'.** Size is expressed in terms of the number of pixels illuminated on the detector screen.

Continuous temporal characteristics of a signal are restored in a data structure, so called '**track**' during the tracking process. MDS detector gives output in a video stream format, such as every single frame holds signal data. Tracking is performed for a single signal among frames such that each signal in the current frame is compared with the previous frame data. Corresponding matched primitive data is restored as the track of that signal.

In an MDS classification problem, obviously two classes exist,

- The class of missiles, which will be called as '**m-class**',
- The class of non-threatening sources or false alarms (FA), which will be called as '**f-class'**.

Two types of scenarios were generated in order to extract data corresponding to each class. The details of the scenario generation will be given in the methodology section.

This two-class characteristic of the problem leads the designer to **supervised methods** in classification design, which means that the designer has to define a trainable system. Hence, a **training set** is required to be established as a former study in the classifier design, which is a set of feature vectors chosen for the training of the classifier.

In the following sections, the assumptions made and the methodology followed for the synthetic data generation are presented.

2.2 Assumptions

It is no doubt that the overall system performance of an MDS has to be considered as a result of the performances of subsystems as a whole. In other words, none of the subsystems such as the optics unit, detector, tracker or classifier module can individually and dominantly determine the overall system performance in the realistic case. However in the scope of this study, since the main focus is on the feature extraction and classification, detection and tracking stages are put out of consideration.

One assumption made during the data generation is 'the single source assumption'. This means that only one source object (either a threatening missile or a non-threatening source of radiation) is employed in each scenario. No multi-signal tracking is studied intentionally.

In accordance with the one mentioned above, a second assumption is made such that the detection and the tracking of the source defined in the scenario are performed free of errors. In other words, feature generation is to be performed on a track data, which is perfectly tracked even in some type of scenarios where the airborne platform is in high-speed maneuver.

Another assumption is about the environment defined in the scenarios. The environment here for our electro-optical system in question can be considered as the weather conditions. Factors that are effective in the transmittance of the source irradiance are expressed in terms of one single parameter, the **atmospheric attenuation coefficient (AAC)**. Attenuation coefficient in the realistic case is the consequence of some certain atmospheric parameters which have physical meaning such as visibility, humidity, transmittance blocking gas concentrations, wind etc. This AAC parameter is used as an input in the scenario definition and generation environment.

In addition, no optical obstacle such as cloud, smoke, bird or any kind of ground terrain is assumed to be present between the source object and the MDS detector in the scenarios.

2.3 Data Generation Process

2.3.1 Methodology

Constructing a data set consisting of training and test subsets for an MDS system includes difficulties technically, financially and strategically in reality. As for the training set construction for instance, it is extremely important to include real readings (data gathered from real missile firings and false alarm data collection studies). In order to cover a wide range of operational aspects of threat scenarios, those real readings are better to be in variety, considering various missile types, environmental conditions and carrier platform attributes. Hence, as obvious, collecting such a real reading set is an expensive process. Such a process will require a sophisticated site, specialized people and equipment, which are to be well organized. In addition, one has to take into account the secrecy issue of such a data from strategical point of view.

Whereas collecting real readings for the false alarm class might be of equal criticality as for the missile class, the variety and availability of the data belonging to f-class is much greater than that of the m-class.
Due to the difficulties presented above, the data set to be established is finite realistically. In order to enlarge the training set, synthetic data can be generated as an addition to the real measurements. These synthetic data can be generated by means of some simulation environment, in which the scenery of operational events (including missiles, other irradiance sources, airborne platforms, etc.) and the sensor itself are simulated.

In order those simulations can provide acceptable data for a realistic MDS system, the models utilized in the simulation environment have to be 'verified' and 'validated' in terms of being realistic. These models include irradiance profile models and kinematical models of missiles and other sources, such as airborne platforms, missiles and expendable counter measure decoys as flares. Clearly seen, verification and validation processes of those models are to be considered as another topic, whose cost is high either. A further discussion over the closely related terms verification, validation, simulation and modeling can be found in [6] from an EW point of view.

In this study, purely synthetic data sets generated from generic models will be used. The nature of these data has convenience with the characteristics of operational and realistic cases, although the content cannot be evaluated to be realistic. The strategy in constructing the working environment is to cover circumstances in the scenarios that are close to operationally realistic cases, by utilizing generic models. Since scenarios of both classes are defined by means of same models, the characteristic of the classification problem will have convenience to that of the realistic case.

Data generation process performed in the study can be summarized briefly as follows:

I. Scenario definition and generation (Simulation run)
II. Scenario evaluation, selection and calculation, primitive feature generation (position, size, intensity of signals)
III. Feature generation by utilization of primitive features.

Steps I and II were performed by utilizing the simulation environment in ASELSAN Inc. Microwave and Systems Technologies Division. As formerly stated, generic models were used for simulation of scenarios. Steps I and II as the parts of data generation process are considered as the former stages that provide input to the main scope of this thesis study. For Step III, software implementation was made in MATLAB® environment. Detailed explanations regarding this step are presented in the following paragraphs.

2.3.2 Scenario Definition and Generation

The main concern in defining scenarios is the coverage of a wide range of operational circumstances that can happen during real life for both m-scenarios and f-scenarios. Regarding this concern, scenario definitions are parameterized as follows. Defining each type of scenario with those parameters, feasible combinations are generated up to certain extent. A missile engagement scenario is called as an '**m-scenario**', whereas a false alarm scenario is called as an '**f-scenario**'. First **150 m-scenario'**s and **120 f-scenario**'s were defined and generated for data generating purposes. A further elimination was performed according to scenario evaluation criteria, which is presented in the corresponding paragraph.

2.3.2.1 M-scenarios

An m-scenario is defined by 6 parameters, which are as follows:

- platform altitude above ground (alt),
- Shooting direction angle (SDA),
- range (R),
- missile type (m_type),
- platform velocity (v),
- atmospheric attenuation coefficient (AAC)

In Table 2-1 defined values of m-scenario parameters are given.

Table 2-1 M-scenario parameters

No	Parameter	Possibilities	Values	Unit
1	alt	7	100, 200, 500, 1000, 1500, 2000, 3500	(m)
2	AAC	4	1.0, 1.5, 1.75, 2.0	-
3	R	7	200, 500, 1000, 1500, 2000, 3000, 5000	(m)
4	m_type	2	m1, m2	-
5	SDA	9	-40, -30, -20, -10, 0, 10, 20, 30, 40	(deg)
6	V	6	0, 50, 75, 100, 120, 150	(m/sec)

One can see from Table 2-1 that it is theoretically possible to obtain more than 20000 scenario combinations, however not all of the combinations are meaningful as an operational scenario. A symbolic m-scenario is illustrated in Figure 2-1 below.

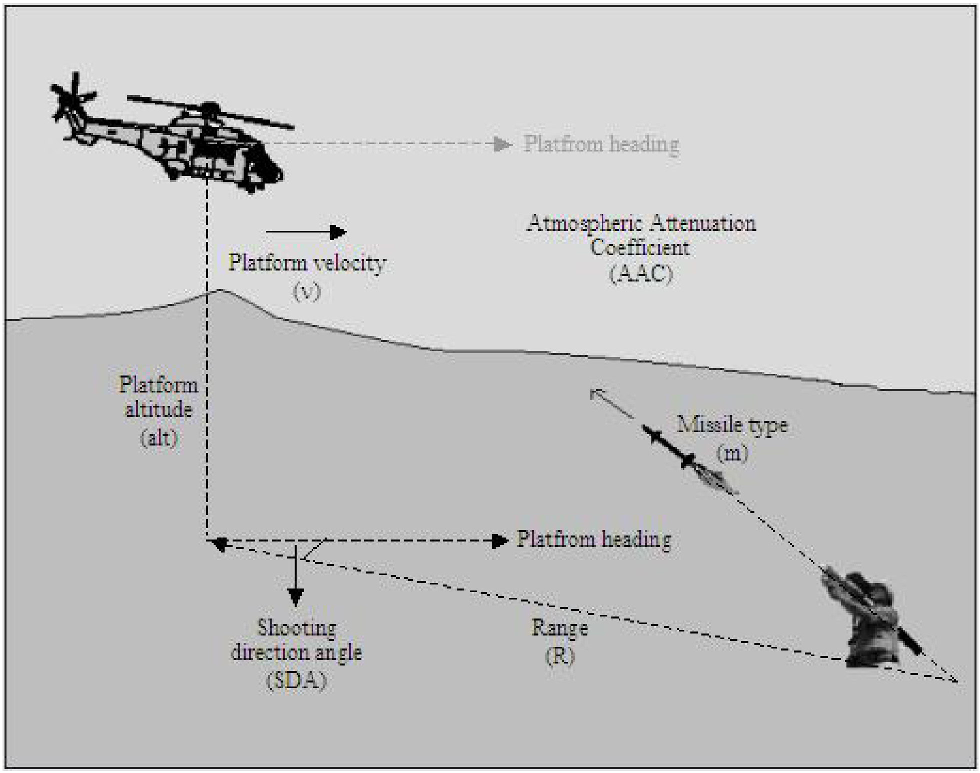

Figure 2-1 An illustration of an m-scenario

2.3.2.2 F-scenarios

In Table 2-2 defined values of f-scenario parameters are given. An f-scenario is defined similarly by 6 parameters as m-scenarios. However in this case, R is defined as the distance from platform to the FA source. As the FA source types, constant and fluctuating irradiance sources, own-ship and wingman flares, non-approaching missiles, approaching and leaving afterburners were defined to cover the variety of FA sources. The other difference is the Approaching Direction Angle (ADA) to the FA sources instead of SDA for m-scenarios. Again similarly, more than 100000 theoretical combinations exist, not all is operationally meaningful. Possible FA sources are illustrated on a symbolic f-scenario in Figure 2-2.

Table 2-2 F-scenario parameters

No	Parameter	Possibilities	Values	Unit
1	alt	5	0, 100, 200, 500, 1000	(m)
2	AAC	4	1.0, 1.5, 1.75, 2.0	-
3	R	9	0,100, 200, 500, 1000, 1500, 2000, 3000, 5000	(m)
4	f_type	10	1, 2, 3, f4, t4, rl4, lr4, 5, rl5,lr5	-
5	ADA	9	-40, -30, -20, -10, 0, 10, 20, 30, 40	(deg)
6	V	7	0, 50, 75, 100, 150, 200, 250	(m/sec)

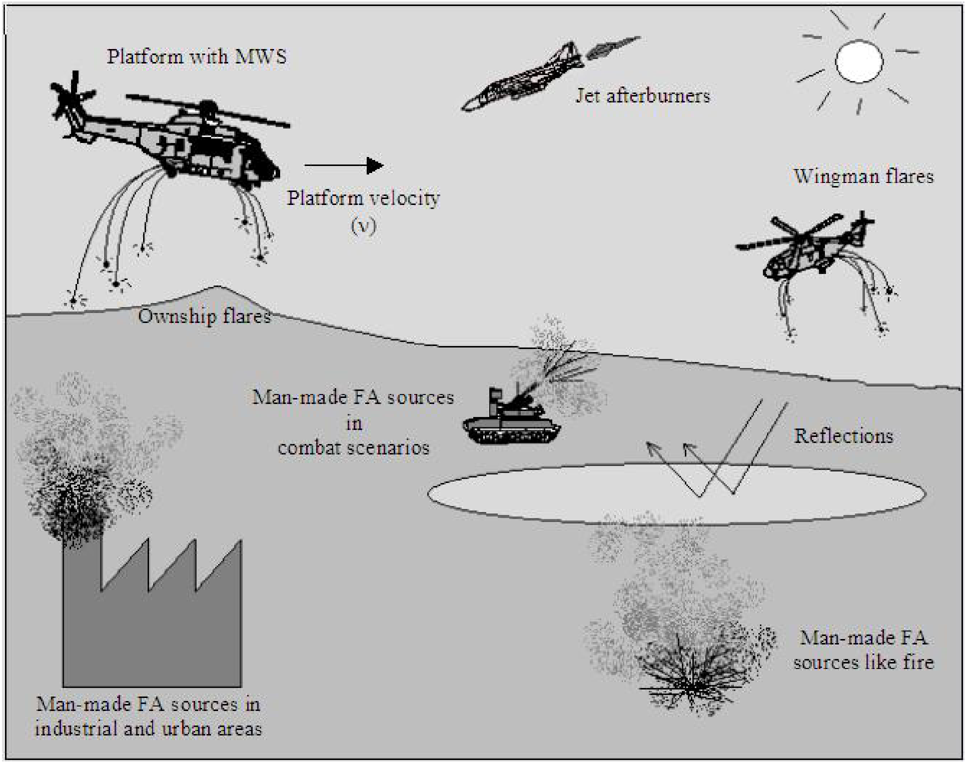

Figure 2-2 An illustration of an f-scenario

2.3.3 Scenario Evaluation, Selection and Calculation

Scenarios are simulated by the detector simulator SW in order to generate the primitive features. The detector simulator takes the scenario calculation files as input and calculates the primitive features as if they were observed from the system carrier airborne platform, which is pre-defined in the scenario to be processed. The calculation of the detector simulator mainly depends on the distance and the angular position of the system carrier platform to the FA sources, by taking the atmospheric attenuation into consideration. The simulator SW utilizes a generic camera model to convert the outer environment in the scenario into the digital objects having certain intensity and sizes in the FOV (Field of View). Size is expressed in terms of number of pixels and the intensity is unitless. As the object comes closer to the platform or the platform approaches to the source, intensity and size of the corresponding detector object increases according to the distance and generic camera model characteristic.

After 150 m-scenarios and 120 f-scenarios were simulated, generated primitive features were restored in track files for each scenario. Although these scenarios had operationally meaningful definition parameters, corresponding outputs (track files) were checked for eliminating inconvenience results of scenario calculation. Due to the fact that the effect of models to the simulation results cannot totally be seen just from the definition, the results should be evaluated. For this purpose, each scenario was assessed by means of visualization of the generated primitive features. Plots of primitive feature variations vs. time were evaluated to select scenarios to be used further. See Figure 2-3 and Figure 2-4 below as the visualizations of a sample m-scenario and an f-scenario respectively. In some of the scenarios, no track was established in the simulations due to reasons that the detector was too far to detect the source or the source went out of the field of view (FOV) of the detector.

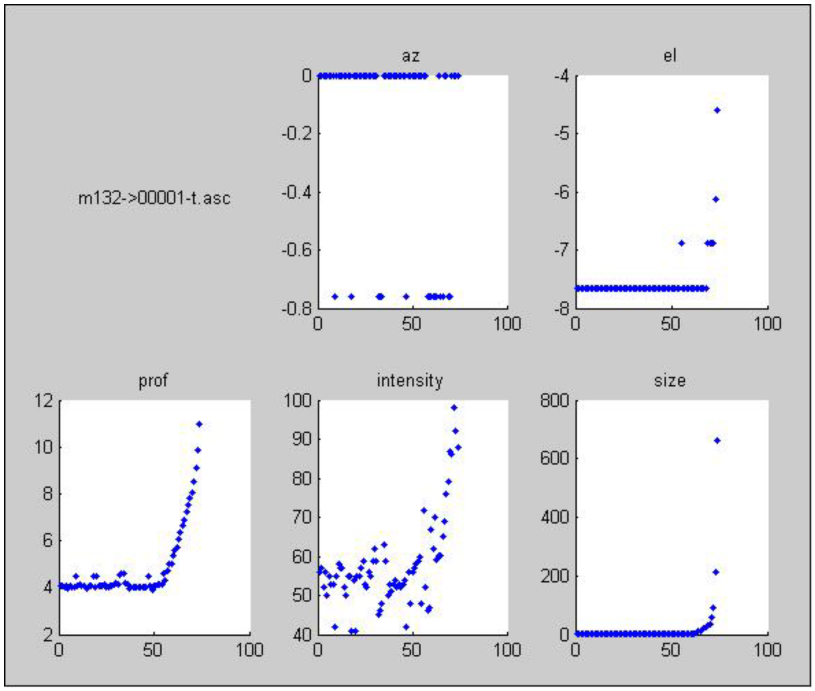

Figure 2-3 Visualization of primitive features for an m-scenario

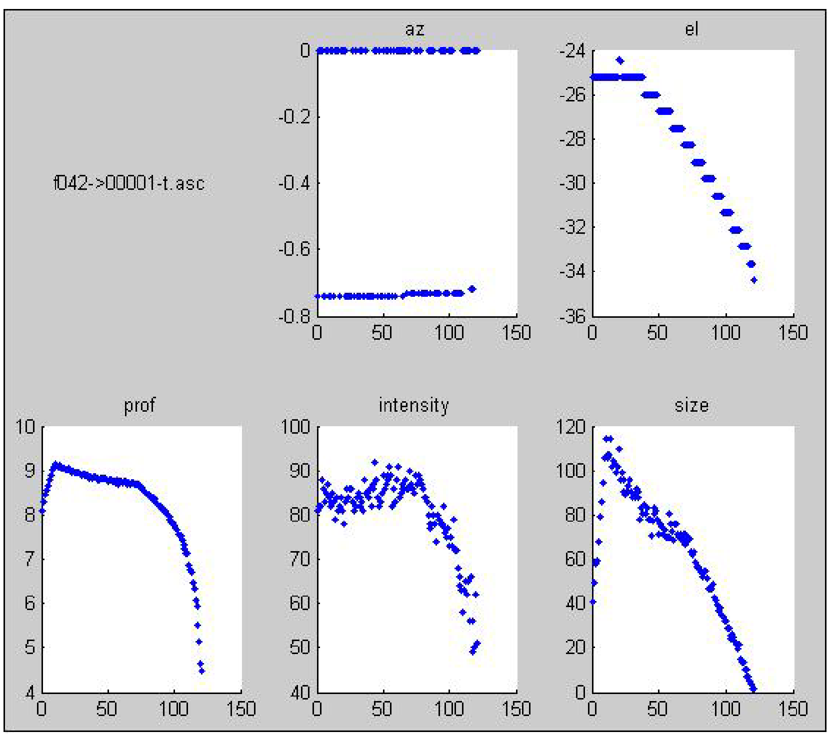

Figure 2-4 Visualization of primitive features for an f-scenario

Two assessment criteria were applied for eliminating an m-scenario: Scenarios with 'no tracks' and scenarios with 'Bad signatures'. First is obvious. The justification behind the second criterion is that in some cases the gathered profile from the source did not reflect the characteristic defined in the irradiation model, often due to the attenuation of signal in long distances. Since the size and the intensity of a signal exponentially increases when the platform approaches to a source (or vice versa), the expected characteristic for this primitive features is such an apparent exponential increase for an approaching missile. A sample temporal intensity characteristic of 'good signal' track is given in Figure 2-5 and the corresponding temporal size characteristic is given in Figure 2-6 below.

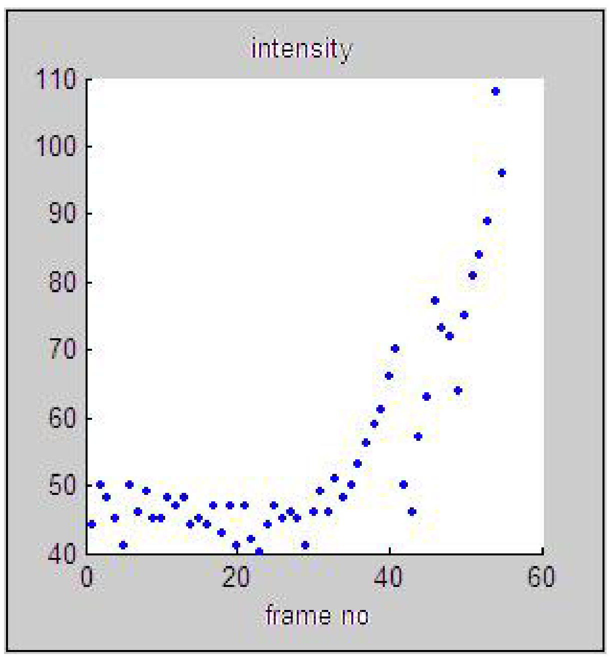

Figure 2-5 Temporal intensity characteristic of a 'good signal' track

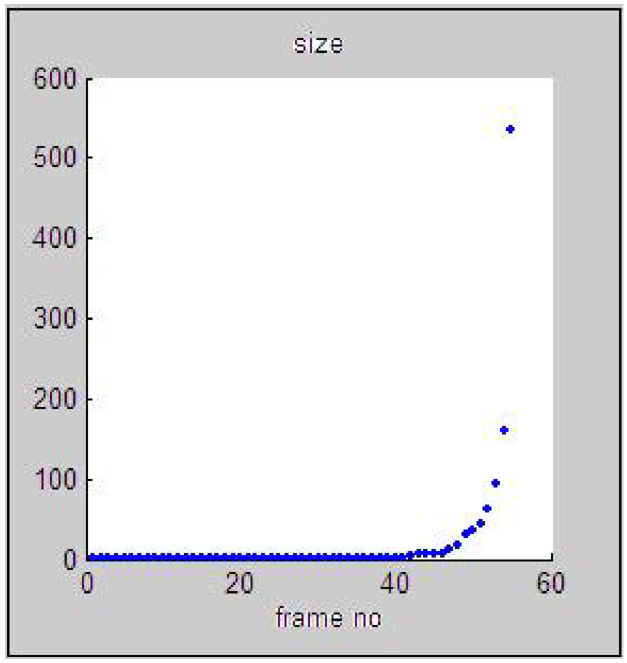

Figure 2-6 Temporal size characteristic of a 'good signal' track

As seen, both intensity and size variations of a good signal track are exponential as expected. At this point, one can make an observation that will be useful later, such that if logarithm of the multiplication of these two primitive features is taken making use of this exponential trend, a smooth additional primitive feature can be obtained. This additional parameter, which is called as '**profile**', is produced as given in (2-1) below. Also the corresponding profile variation obtained from the size and intensity graphs above is illustrated in Figure 2-7.

$$profile = \ln (size * intensity) \qquad \textbf{(2-1)}$$

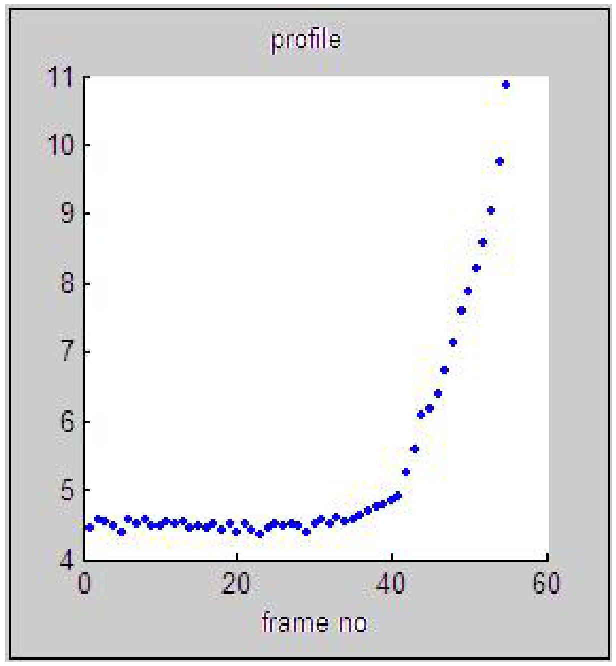

Figure 2-7 Temporal profile characteristic of a 'good signal' track

The intensity, size and profile variations of an eliminated (bad signature) m-scenario are given together within Figure 2-8.

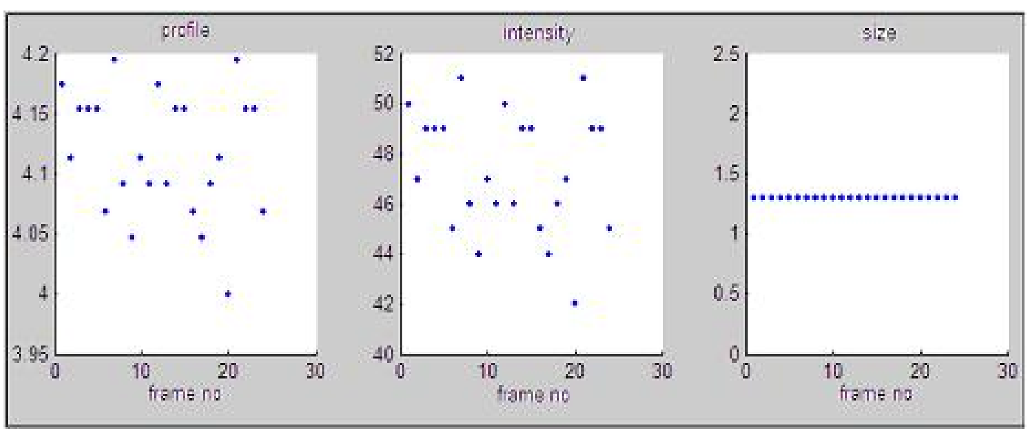

Figure 2-8 Characteristics of a 'bad signature' track

The single elimination criterion for f-scenarios is discarding the scenarios with no tracks. No signal characteristics comparison was made since FA sources vary and do not always have to obey a specific irradiation model. During the assessment of f-scenarios, the ones with close characteristics to m-scenarios were marked for further test and checking purposes.

A summary of the scenario definitions list with some markings related with evaluation and elimination are given in the tables below: Table 2-3 for m-scenarios and Table 2-4 for f-scenarios. In order to save space just a random selection of the scenarios are displayed in tables.

Table 2-3 A selection of m-scenario definitions after evaluation

No	alt	AAC	R	m_type	SDA	v	Notes
3	100	1.50	200	2	10.00	0	
4	200	1.00	500	1	0.00	0	
5	3500	1.75	5000	1	40.00	0	Bad signature
12	500	1.50	1500	2	40.00	0	
13	100	1.50	1000	2	10.00	0	
14	3500	1.50	5000	2	20.00	0	No track!
15	1000	1.75	5000	2	-40.00	0	Bad signature
16	500	2.00	1500	1	0.00	0	
107	500	1.50	1000	1	-40.00	50	
110	200	1.50	2000	1	-40.00	50	
111	1000	2.00	5000	2	20.00	50	Bad signature
116	200	1.75	1000	2	-30.00	75	
118	100	1.50	2000	1	20.00	75	
138	500	1.00	2000	1	20.00	100	
139	100	2.00	1000	2	30.00	100	
144	500	2.00	2000	1	0.00	150	
145	1000	1.75	5000	1	-20.00	150	
146	1000	1.50	5000	2	10.00	120	
148	500	1.00	5000	2	30.00	120	

As the result of scenario evaluation; **132** m-scenarios and **101** f-scenarios were assessed to be convenient for further processing.

Table 2-4 A selection of f-scenario definitions after evaluation

No	alt	AAC	R	f_type	v	ADA	Notes
4	500	1.00	2000	T4	200.00	-30.00	
5	100	1.75	2000	1	50.00	0.00	
10	100	1.50	1000	1	50.00	30.00	Potential FA!
11	100	1.00	3000	T4	200.00	-10.00	
12	100	1.00	1000	3	0.00	-30.00	Potential FA!
43	500	1.75	1000	2	50.00	10.00	
44	200	1.00	2000	1	75.00	30.00	Potential FA!
45	100	2.00	1000	2	0.00	10.00	
46	1000	1.50	5000	T4	150.00	0.00	Potential FA!
47	0	1.75	100	3	100.00	-30.00	No track!
48	500	2.00	2000	5	75.00	-20.00	
50	200	1.00	1500	F4	200.00	20.00	
51	200	1.75	3000	2	150.00	-10.00	No track!
98	200	1.50	1000	5	75.00	0.00	Potential FA!
99	100	2.00	500	5	75.00	0.00	Potential FA!
100	100	1.50	2000	2	50.00	-10.00	
103	500	1.50	0	3	0	-10.00	
104	500	1.00	0	3	50	20.00	No track!
118	100	1.50	1000	5	50.00	-20.00	Potential FA!
119	1000	2.00	1000	5	0.00	0.00	
120	100	1.50	500	5	50.00	-10.00	Potential FA!

2.3.4 Data Set Generation

Raw data generation process is described up to this point. Primitive features are restored in track files in some specified format. Since the primitive features alone in a single detection frame do not convey adequate amount of descriptive and discriminative information for both two classes (m-class and f-class), temporal characteristics are to be utilized to extract information. The methodology is explained in following.

In order to obtain information from the temporal characteristics, variation of each primitive feature vs. time is modeled. This modeling is performed first by fitting a curve on to the temporal characteristic of the feature, then by extracting some parameters from this curve-fitting process. In other words, information is extracted from the parameterization of the temporal characteristics of primitive features of scenarios.

Curve-fitting process can be performed over the complete track, i.e. between the first and the last frames. However, taking the classification process for a MDS into consideration, this feature production process employed over the whole track will not be meaningful, since one has to wait until the last frame in order to obtain a result by using the fitting parameters of the track. What is desired from such a real-time system is that it has to generate necessary outputs for performing classification in each frame while it tracks a source object. This requires the implementation of a dynamic decision making classifier. For this purpose, a 'moving window' approach is applied for the curve-fitting over a single track feature. The procedure is explained as follows.

Suppose the tracker starts to generate the primitive features of a detector signal as the output from the time frame it started tracking, 'n_0'. In order to apply curve-fitting on the temporal variation of one of the primitive features, one has to wait for some time period during which the history of variations is restored. Call this period as 'window (w)', which can be described in terms of number of frames. Then at the time frame ($n_0 + w - 1$), there exist w samples at hand, over which the curve-fitting can be applied. As the tracker generates further outputs in the proceeding frames, window is moved at a step size (*s*) and curve-fitting process continues in this fashion until the last frame. This process is visually represented in Figure 2-9 below.

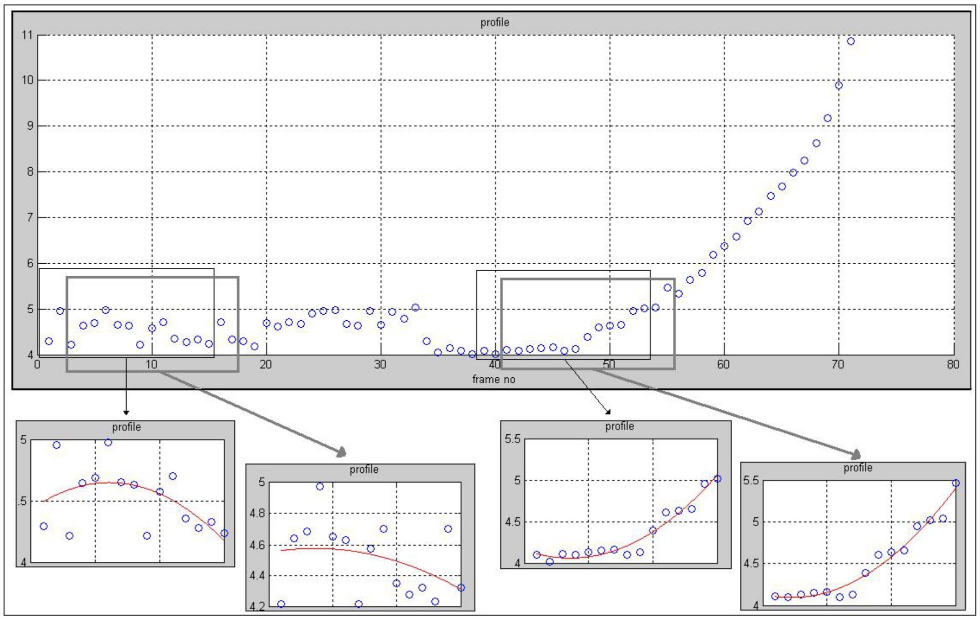

Figure 2-9 A representation of curve-fitting by moving window for 'profile'

In the figure, w is chosen as 15 frames, s is equal to 2 and the curve to be fitted is 2^{nd} degree polynomial for the primitive feature 'profile'. Corresponding residuals of fitting are not displayed, however such by-products are also used in parameterization of primitive features. Details of modeling approaches are introduced in the subsequent paragraph.

2.3.4.1 Modeling of Primitive Features

Information extracting methodology depends on the modeling of primitive features during the lifetime of a track, since the classification is subjected to provide output without waiting till the end of the track. For this reason, it was explained why a moving window is to be applied for modeling. Modeling is applied first by curve-fitting, then extracting parameters accordingly. However, there is no unique way to perform the fitting. Therefore, one has to employ various types of curves for different window size (w) and step size (s) and generate some several raw data sets for further processing. At the end, final system performances of different raw data sets have to be compared, in order to make a design decision for the modeling.

In the study, two values are chosen for w: 15 and 20. Lower values are not selected due to the reason that it would be hard to gather descriptive information for such small values of w at the staring phase, so further feature extraction and classification depending on the corresponding data might not be so meaningful.

For the parameter s, two values are chosen: 2 and 4. These sub-sampling sizes are employed for the reason that the number of feature vectors at the end might get too large when the s is selected to be 1.

Three types of curves are employed for the fitting process:

- **Type-1**: 1^{st} Degree Polynomial (Linear Fitting)
- **Type-2**: 2^{nd} Degree Polynomial (Quadratic Fitting)
- **Type-3**: Combination of 2^{nd} Degree Polynomial and Exponential Fitting

At first, Type-1 and Type-2 were applied. Then, depending on the observation in the exponential increase trend of size and intensity features, Type-3 was applied as an additional approach. So, while exponential fitting was being applied for intensity and size, quadratic fitting was employed for the other primitive features (Refer to Figure 2-5 and Figure 2-6). Figure 2-9 also might give some insight about the possible alternatives of curve types for fitting. As seen, line fitting can be evaluated to be adequate for low complexity concerns, however quadratic curve (or exponential for some cases) can be evaluated to be adequate for quality concerns.

During the study, the naming convention applied for configuration control of generated data sets is explained in Table 2-5.

Table 2-5 Naming convention for raw data sets

Parameter	Value	Explanation
t	1	1st degree polynomial
	2	2nd degree polynomial
	3	Combination of both 2nd degree polynomial and exponential
w	15	-
	20	-
s	2	-
	4	-

An example naming of a raw data set can be given as '**t1w15s4**'. This means, for this data set, curve type is 1st degree polynomial, w is 15 and $s$is 4. In this convention, 12 raw data sets were generated. A summary of the generated data sets are displayed in Table 2-6 below.

Table 2-6 Summary of generated raw data sets

Type No	No	Raw Data Set
1	1	t1w15s2
	2	t1w15s4
	3	t1w20s2
	4	t1w20s4
2	5	t2w15s2
	6	t2w15s4
	7	t2w20s2
	8	t2w20s4
3	9	t3w15s2
	10	t3w15s4
	11	t3w20s2
	12	t3w20s4

2.3.4.2 Feature Vector Set Generation

Feature vector set generation mainly depends on the modeling of data sets. The aim of the modeling is to extract as much information as possible relating the temporal characteristic of a track, regardless of the quality of the extracted information according to any classification criteria at this point. This evaluation will be held under the scope of FSE.

Parameterization is applied by making use of basic data fitting approaches, which are quite well known [28] [29]. Raw data sets are classified under three main groups according to the curve type used. For each group, feature vectors are generated by extracting parameters of modeling, i.e. each entity of a corresponding feature vector is an extracted parameter. Therefore, now it is clear that the dimensionality of the feature vector sets are determined by the number of extracted parameters.

Over the complete lifetime of a track in a given scenario, curve fitting is applied in Least Square Sense by moving window approach for various w as stated before. For each primitive feature; azimuth (az), elevation (el), size, intensity and profile, the parameter set to be extracted is composed of the following:

- Curve coefficients of polynomial and exponential fitting.
    - '**p1**', '**p2**', '**p3**'
- Standard measurement errors regarding the curve coefficients. (calculated over the confidence intervals of coefficients)
    - '**stdp1**', '**stdp2**', '**stdp3**'
- Actual value of the fitted curve at a given point
    - '**fit(n)**'
- Root mean square error of fitting
    - '**rmse**'
- Parameters regarding the goodness of fitting [30]
    - '**rsquare**: (R square) Square of the multiple correlation coefficient
    - '**adjrsquare**': Adjusted rsquare

MATLAB® Curve Fitting Toolbox was utilized for the calculation of these parameters. Parameterization of each group is explained in the following paragraphs.

2.3.4.2.1 Type-1

As in (2-2) basically two parameters exist: '**p1**' and '**p2**' for the 1st degree polynomial fitting.

$$fit(x) = p1 * x + p2 \tag{2-2}$$

In Table 2-7, feature vector definition for Type-1 is given. As can be seen from the table, dimension of a Type-1 feature vector is **40**. Also a plot of Type-1 for intensity (t1w15s4) is given in Figure 2-10 as a sample visual representation.

Table 2-7 Feature vector definition table for Type-1

No	Feature Name	No	Feature Name	No	Feature Name
1	p1_az	17	p1_size	33	p1_prof
2	p2_az	18	p2_size	34	p2_prof
3	stdp1_az	19	stdp1_size	35	stdp1_prof
4	stdp2_az	20	stdp2_size	36	stdp2_prof
5	fit(w)_az	21	fit(w)_size	37	fit(w)_prof
6	rmse_az	22	rmse_size	38	rmse_prof
7	rsquare_az	23	rsquare_size	39	rsquare_prof
8	adjrsquare_az	24	adjrsquare_size	40	adjrsquare_prof
9	p1_el	25	p1_int		
10	p2_el	26	p2_int		
11	stdp1_el	27	stdp1_int		
12	stdp2_el	28	stdp2_int		
13	fit(w)_el	29	fit(w)_int		
14	rmse_el	30	rmse_int		
15	rsquare_el	31	rsquare_int		
16	adjrsquare_el	32	adjrsquare_int		

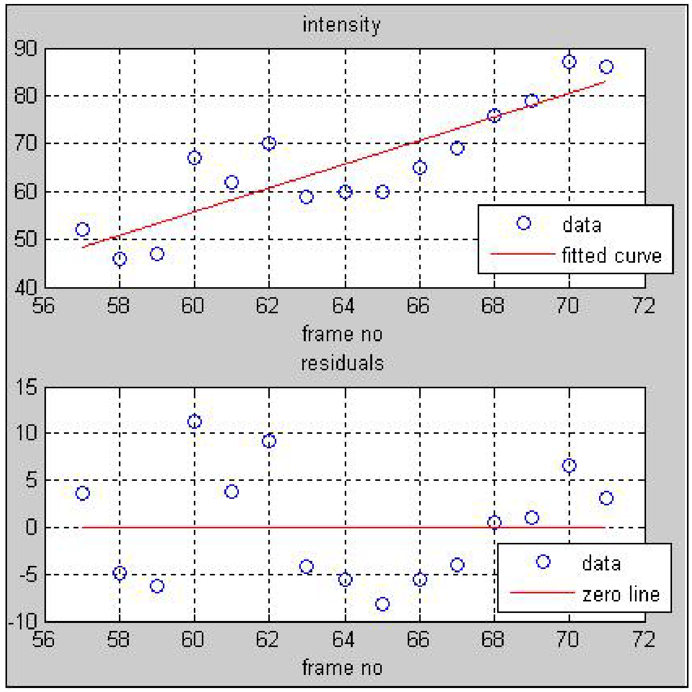

Figure 2-10 A sample Type-1 fitting plot (t1w15s4, intensity)

2.3.4.2.2 Type-2

Similar procedure applies for the Type-2 case, however the equation of the curve is as in (2-3) below.

$$fit(x) = p1 * x^2 + p2*x + p3 \tag{2-3}$$

In Table 2-8, feature vector definition for Type-2 is given. The dimension of a Type-2 feature vector is **50**. Also a plot of Type-2 for intensity (t2w15s4) is given in Figure 2-11 as a sample visual representation.

Table 2-8 Feature vector definition table for Type-2

No	Feature Name	No	Feature Name	No	Feature Name
1	p1_az	21	p1_size	41	p1_prof
2	p2_az	22	p2_size	42	p2_prof
3	p3_az	23	p3_size	43	p3_prof
4	stdp1_az	24	stdp1_size	44	stdp1_prof
5	stdp2_az	25	stdp2_size	45	stdp2_prof
6	stdp3_az	26	stdp3_size	46	stdp3_prof
7	fit(w)_az	27	fit(w)_size	47	fit(w)_prof
8	rmse_az	28	rmse_size	48	rmse_prof
9	rsquare_az	29	rsquare_size	49	rsquare_prof
10	adjrsquare_az	30	adjrsquare_size	50	adjrsquare_prof
11	p1_el	31	p1_int		
12	p2_el	32	p2_int		
13	p3_el	33	p3_int		
14	stdp1_el	34	stdp1_int		
15	stdp2_el	35	stdp2_int		
16	stdp3_el	36	stdp3_int		
17	fit(w)_el	37	fit(w)_int		
18	rmse_el	38	rmse_int		
19	rsquare_el	39	rsquare_int		
20	adjrsquare_el	40	adjrsquare_int		

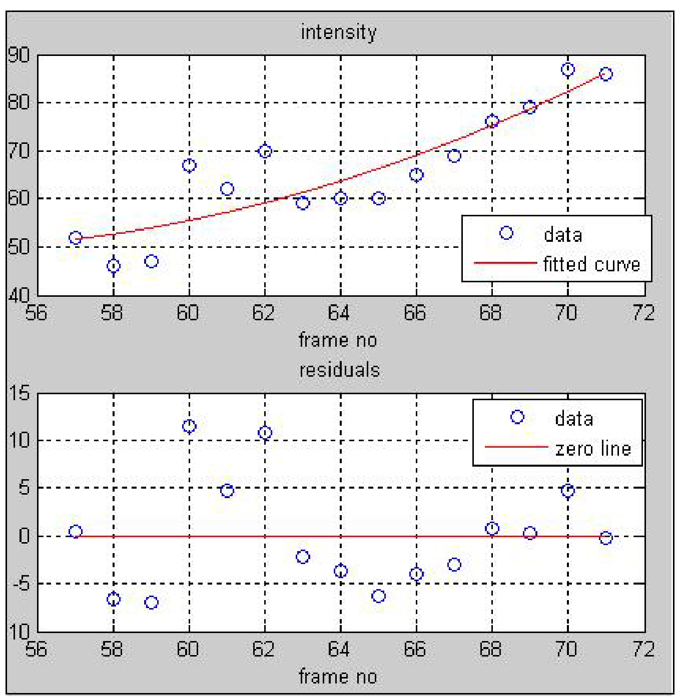

Figure 2-11 A sample Type-2 fitting plot (t2w15s4, intensity)

2.3.4.2.3 Type-3

The procedure is composed of two parts for this type. For the primitive features az, el and profile, quadratic curve is fitted according to the equation (2-3) as explained above. On the other side, exponential fitting is applied for size and intensity according to the equation (2-4) below.

$$fit(x) = a * \ exp\ (b * x) \tag{2-4}$$

In Table 2-9, feature vector definition for Type-3 is given. The dimension of a Type-3 feature vector is **46** as seen from the table. In addition, a visual representation of Type-3 for intensity (t2w15s4) is given in Figure 2-12.

Table 2-9 Feature vector definition table for Type-3

No	Feature Name	No	Feature Name	No	Feature Name
1	p1_az	21	a_size	37	p1_prof
2	p2_az	22	b_size	38	p2_prof
3	p3_az	23	stda_size	39	p3_prof
4	stdp1_az	24	stdb_size	40	stdp1_prof
5	stdp2_az	25	fit(w)_size	41	stdp2_prof
6	stdp3_az	26	rmse_size	42	stdp3_prof
7	fit(w)_az	27	rsquare_size	43	fit(w)_prof
8	rmse_az	28	adjrsquare_size	44	rmse_prof
9	rsquare_az	29	a_int	45	rsquare_prof
10	adjrsquare_az	30	b_int	46	adjrsquare_prof
11	p1_el	31	stda_int		
12	p2_el	32	stdb_int		
13	p3_el	33	fit(w)_int		
14	stdp1_el	34	rmse_int		
15	stdp2_el	35	rsquare_int		
16	stdp3_el	36	adjrsquare_int		
17	fit(w)_el				
18	rmse_el				
19	rsquare_el				
20	adjrsquare_el				

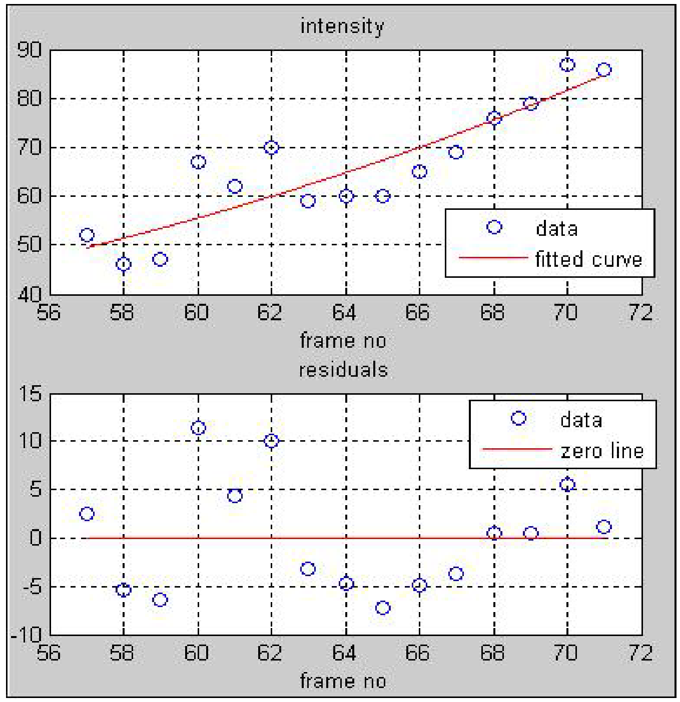

Figure 2-12 A sample Type-3 fitting plot (t3w15s4, intensity)

CHAPTER III

DATA SET CHARACTERISTICS

In most cases pattern recognition systems are fed with input data that are obtained as a result of sub-systems such as transducers, detectors or sensors. These prior units can be taken as lossy converters of observations from the real world environment to some electrical signals. Therefore, the complexity and difficulty of the classification problem may well depend on the characteristics and limitations of them [9].

In order to well define the problem in our generic MDS system, we have to know the nature of data set, which was synthetically generated as described in the previous chapter. Hence, before going deeper into the subject of FSE, it will be convenient to make an evaluation of data set to be used. In this chapter, qualitative and statistical characteristics of the data set will be evaluated.

3.1 Qualitative Characteristics

Feature vector sets (data set) were generated in 12 groups according to 3 different modeling types as described earlier in Table 2-6. A more detailed summary of the generated data set can be found in Table 3-1 below.

Table 3-1 A detailed summary of the data set

Type No	No	Name	Dimension	# of Vectors (m-class)	# of Vectors (f-class)
1	1	t1w15s2	40	4274	6534
	2	t1w15s4	40	2239	3347
	3	t1w20s2	40	3948	6282
	4	t1w20s4	40	2076	3224
2	5	t2w15s2	50	4274	6534
	6	t2w15s4	50	2239	3347
	7	t2w20s2	50	3948	6282
	8	t2w20s4	50	2076	3224
3	9	t3w15s2	46	4274	6534
	10	t3w15s4	46	2239	3347
	11	t3w20s2	46	3948	6282
	12	t3w20s4	46	2076	3224

One can observe that the number of feature vectors for the m-class is less than the f-class. As stated in Chapter-II, this is one of the most apparent situations for the data characteristic of two classes. In the real world case, the situation for relative amount of data is even worse than this synthetic case, since there exists more frequent and much variety in the false alarm sources.

Another reason for this difference in the resulting feature vector amount is that the lifetime for a track belonging to the m-class is generally shorter than that of a track of f-class in the average. The dimensionality of the data sets varies from 40 to 50, which can be considered as quite high for a real-time system, depending on the complexity of the classifier if no reduction of dimensionality is applied.

In addition to the dimensionality, number of feature vectors to be handled for the training of a real world system might probably be much higher than obtained for our synthetic case.

These properties all point to the necessity of a smart data reduction as a pre-processing step for the classifier design and operation.

3.2 Statistical Characteristics

As well as quality characteristics, statistical characteristics of the feature vectors sets were investigated. In this part, statistical characteristics of the data is covered, however in the following sections, where the adaptation of FSE and class separability approaches are presented, necessary reference will be given to the statistical characteristics of the data.

3.2.1 Correlation

Correlation between the individual features was investigated for 12 data sets. For each data set, a matrix of correlation coefficients was produced in the MATLAB® environment. In addition to the correlation coefficient matrix, a matrix of probability values (P-values) were also computed as a measure for the significance of the correlation calculation. Small values (such as <0.05) of the entities in the P-values matrix represent that the correlation is significant at their locations in the correlation coefficient matrix.

When one supposes that the dimension of the data set is '**D**', then the resulting correlation coefficient matrix is a '**DxD**' matrix. Since minimum D is 40, the smallest size of matrices is 40x40, which can be hard to follow. For an easier understanding of the mutual correlation among individual features by means of these matrices, a gray-scale coding is applied and the matrices were drawn as images. In Section-1 in Appendix, correlation matrices relating to each data set are given.

Correlation coefficient matrices mostly include significant correlations as widely seen on the P-value matrices. Correlation tends to be more significant for s2 types than s4 types, which is mainly because the number samples of s2 is higher. By keeping this observation in mind, one can conclude that the mutual correlation of individual features varies from type to type, however the major part is not so small. The reasons for the high correlation can be summarized as follows.

The variety of these synthetic data sets might not cover a wide range of possibilities for any case. The situation is more valid especially for m-scenarios.

Since the features are being generated from primitive features of tracks, they cannot be considered to be totally independent from each other. Besides, this consideration is one more time supported by the generation method of the profile, which is a product of the two other primitives, intensity and size.

The physical behavior and the resulting temporal characteristics of intensity and area can have similar trend especially for the m-scenarios, where the intensity and the size of the signal corresponding to the threat tend to increase exponentially.

The reason for high correlation of some parameters relating the 'goodness of model fit' or measurement errors for estimates of curve coefficients can be discussed to be mainly due to the fact that the error characteristics of the fitting process is similar for some features, especially for tracks with disturbed signal characteristics.

3.2.2 Histograms

It is essential to have reliable estimates of the probabilistic distributions for data sets of a pattern recognition problem. The complexity of the design of the classification module decreases, as it gets more feasible to make reliable estimations on the data distributions or one can parameterize the distributions. However depending on the data set, when parameterization is not feasible, non-parametric approaches should be considered for further classification design.

In order to see distribution characteristics of individual features of m-class and f-class, histogram plots were generated for each feature of every data set type. Totally 544 histogram plots were generated and analyzed. Since the number is quite high, some several plots of t1w15s4 (derived from odd numbered scenarios) were placed in the Appendix B (See Figures B1-13). As can be seen from the plots, reliable parameter estimation is not always applicable. Characteristic is mainly similar for the other data set types.

One can make some pre-assumptions on the statistical distributions of the data set, such that for instance Feature-2 (p2_az) and Feature-5 (fit(w)_az) can be assumed to be uniform in a certain range, since a threat or non-threat source can occur in any azimuth position depending on the horizontal FOV of the sensor. However for the major part of the features, it is hard to make such assumptions. In addition to that, for the m-class, since the variety of the data is low especially for the real life case, statistical data of samples may not give significant information. However, in order to further see the effect of the parameterization performed depending on the sample histograms, normal distribution fitting was applied depending on the sample mean and standard deviations, as can also be seen on the figures. In the following, the fitted distribution parameters are used in feature set evaluation and corresponding classification results are compared with other non-parametric techniques.

3.3 Normalization

Normalization or standardization can be required for the values of the features vary in large ranges due to the fact that they may be in different units. During the further analysis of the feature vectors, bigger values of some features may give a misleading orientation to the whole feature vector or the smaller values may disappear among large values in a comparison or a distance calculation. In order to cope with this situation, normalization should be applied, which is a scaling that maps all the entities of a feature vector to a predefined region so that all the entities of a feature vector can be controlled in the same range.

In our problem, normalization is needed due to the characteristics of the data sets. For example, while the values of some features corresponding to angular measurements may vary in [0, 90] degrees, the values of some other features corresponding to the fitting quality measures may vary in [0,1].

The main normalization strategy followed in this thesis study is as follows. Since the primary objective of a missile detection system is to correctly discriminate a threat, normalization parameters are derived from the m_class. In other words the base for normalization is the m_class members. A linear mapping is applied for the m_class members first (in [0,1]). Then using the same normalization parameters, f_class members are mapped. The basic normalization (mapping) formula is given in (3-1) below for a vector x, which lies in the interval [x_{min}, x_{max}].

$$\bar{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{3-1}$$

3.4 Summary

As a summary of data set characteristics discussion, the following arguments can be stated. Data set is evaluated both qualitatively and statistically. From qualitative point of view;

- Dimensionality of data set types is high. Reduction is required for classification design.

- Number of feature vectors (data set size) depends on the availability of data and feature generation methodology. In order not to miss a discriminatively valuable feature, one should generate as much features as possible, which will provide satisfactory information about the problem environment.

And from the statistical point of view;

- Correlation between the individual features is calculated and evaluated. Since the modeling strategy depends on deriving parameters from the temporal characteristics of a few primitives, there exist features, which are highly correlated. However, in general correlation is not so high. Necessary reasoning for high correlation is given in detail.

- Histogram plots are assessed together with possible pre-assumptions relating the distributions and it is concluded that, it is hard to extract significant statistical information.

In conclusion, the data characteristics show that mainly non-parametric approaches appear to be more suitable for the nature of our problem in classification design.

CHAPTER IV

FEATURE SET EVALUATION

This chapter consists of two main divisions, the theory and the practice. First the theoretical background relating the widely studied concept, feature extraction in the literature will be given and secondly, the approaches regarding the dimensionality reduction in feature extraction and class separability, that are considered to fit well for our problem will be presented, by giving necessary justifications for the adaptation of them.

4.1 Theoretical Background for FSE Methods

4.1.1 Introduction to FSE

A typical pattern recognition system is defined as in Figure 4-1 below in [9]. Complete system can be categorized into some main components as shown. Feature extraction and classification stages are the two fundamental units of a pattern recognition system, which have close relationship. This relationship can be constructed both conceptually and operationally in the design depending on problems. However, feature extraction is the much more problem domain dependent part as a former block to the classification. Thus, in order feed the classifier with valuable features, domain dependent knowledge is essential. The main goal of the feature extraction is then, to identify discriminative features among candidates in order to obtain optimal (or sub-optimal) class separability for the classifier. In this manner, one can discuss that feature extraction is a form of data reduction by taking care of the class separability [7].

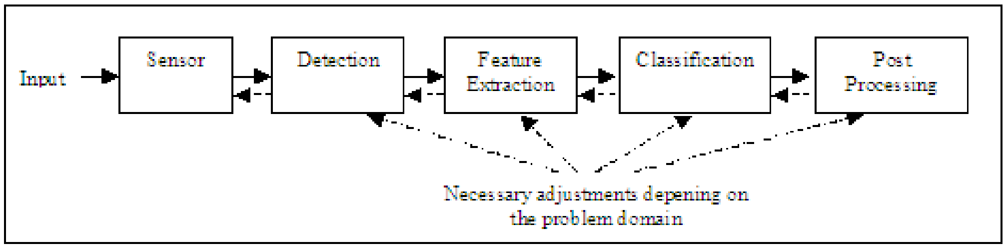

Figure 4-1 A typical Pattern Recognition System

It is a well-known fact that, as the data dimensionality increases, complexity of a classification system rises drastically, so the design becomes more complicated and in practice, classification performance may not increase but decrease. This undesired burden to the system is termed as the well-known argument, 'curse of dimensionality' [17], [7].

In [19], the factors stimulating the curse of dimensionality are discussed in detail, which can be summarized as 'sparsely consisting feature vector sets', 'noise', 'irrelevance' and 'redundancy' which are mainly problems of finite data sets in practice. These factors indicate that a classification system needs to make use of some automated feature extraction methods in order to reduce dimensionality.

4.1.2 Need for FSE

Resulting features from a feature extraction method do not often provide an optimum set for classification. Further, extracting discrimination features from finite data sets increases in difficulty as the data dimensionality increases. The need for features to reduce complexity, combined with the difficulties of extracting features, justifies the need for studying the ways of ranking feature sets for classification [7], [19], [11]. This points to the need for FSE techniques.

The concept of FSE is both related to '*methods of producing features*' and '*class separability measures*'. As introduced formerly, the nature of the problem mainly determines the way, how the features can be produced to extract information. Further application of a convenient FSE method gives an answer to the question: 'Which are the optimal features?'

4.1.3 Methods of Producing Features

There exist plenty of ways of producing feature sets. One can categorize those methods taking different approaches into consideration. In [8], feature extraction is introduced under two main branches; *induction* and *deduction*. Induction is from the historical point of view, the older method. It was used before the advent of modern science. The point is that certain hypothesis about the object under study leads to the extraction of information that (possibly) describes the object or process in a correct manner. Experience is essential in this case. Deduction is a method, which is typical for modern science and engineering. It includes physical modeling, application of fundamental laws of nature, making mathematical modeling, solving equations of the mathematical model and finally finding the variables describing the interesting properties of the system. Whereas the deduction is more desirable than induction due to its property of giving 'insight' about the data, today's many problems need description by making use of both induction and deduction.

Further systematical taxonomy of feature production methods is given in [7]. There, approaches in the literature are categorized into three. *Feature Extraction* (FE), *Feature Selection* (FS) and *Feature Generation* (FG). (Even if these names are used to define the differences among each other, still one can encounter them to be used interchangeably in the literature.)

FE techniques make use of class separability optimality criteria to generate a mapping from data space into the feature space [10], [18]. Majority of FE techniques include linear mappings. FS is a special case of linear FE where the resulting feature vector's elements are a subset of the data vector's elements. FG techniques descriptively represent the data. These techniques include modeling techniques (just as described for the deduction method above), where the parameter estimates of the model are the resulting features. As another point, FG differs from FE/FS techniques in the methodology that it does not depend on any class separability optimization criteria.

We can discuss that the data generation process relating our problem mostly falls under FG techniques in the sense that no optimality criteria of class separability are applied during feature production. The aim was to extract parameters from the models adapted to primitive data in order to represent the temporal characteristics of tracks.

However during the adaptation of FSE approaches we do not have to stick to this definition in the sense that further feature extraction can still be applied over the feature vectors at hand, which were derived from primitive features. Main intention behind the adaptation of such definitions is to localize our problem among the approaches in the feature extraction literature.

4.1.4 Class Separability

FSE methods aim to diminish the dimensionality by keeping the classification performance at a desired rate. The performance of classification can be considered in terms of the misclassification rate or the classification error, which is related with the separability of classes. Using discriminatively powerful features enforces the class separability in classification design. These optimal features can be obtained by either directly generating them according to some optimality criterion or by discarding discriminatively irrelevant features from the data set. By this 'optimality' argument here, the property of the features that provide minimum classification error is meant. And it is clear that, in order to define the optimality, employment of a 'class separability measure' is required. Some several class separability measure concepts are employed as the optimality criteria in FSE methods.

The main and indeed unique class separability measure is the '*Bayes Error Rate* (ε)' in the sense of optimality. By theoretical definition, the minimum error rate or misclassification rate obtainable is ε [17]. However, calculation or even estimation of ε is often hard. A basic investigation of the equations regarding ε will indicate these difficulties. In (4-1) the famous Bayes misclassification rate is given below, where w_i represents the ith class and x is the data vector in n-dimension space.

$$\varepsilon = \int \left[1 - \max_i \mathrm{P}(w_i \mid x) \right] p(x)dx \tag{4-1}$$

Here, $\mathrm{P}(w_i|x)$ is the a posteriori probability function of the ith class and $p(x)$ is the probability of the data vector x. $p(x)$ can also be expressed in terms of a posteriori probability and class probability ($\mathrm{P}(w_i)$) as in the equation (4-2).

$$p(x) = \sum_{i=1}^{L} p(x \mid w_i)\mathrm{P}(w_i) \tag{4-2}$$

And in equation (4-3), the relation of the a posteriori probability function and the class conditional density can be seen.

$$\mathrm{P}(w_i \mid x) = \frac{p(x \mid w_i)\mathrm{P}(w_i)}{p(x)} \tag{4-3}$$

As these equations indicate, direct estimation of ε depends on the estimation of probability density function ($p(x|w_i)$) and the knowledge of class probabilities ($\mathrm{P}(w_i)$). In addition, computational burden for the numerical integration increases with the dimensionality.

In our case, neither the density estimation of data nor the class probabilities are known. Further, in realistic case, the availability of real measurements corresponding to m-class is quite restricted. Thus, direct estimation of ε will suffer from these factors.

The obstacles in calculating ε have led to the development of various approaches of class separability measures and made 'class separability' a widely studied subject in the literature [17], [7], [30]. The concept is merged into several feature extraction methods and applied successfully in literature [10], [18], [19], [23]-[27]. While some of the approaches search for sub-optimal error bounds by converging to ε, some others appear to be heuristic in the sense that computational burden is the main concern rather than direct relation to ε. Further FSE study will benefit this background of class separability. In the following, several fundamental class separability measures are presented, which have popularity and wide usage within several feature set evaluation techniques.

4.1.4.1 Probabilistic and Mathematical Measures

Theoretically thinking, as in the Bayes error case, separability of the classes in a pattern recognition problem can be best described by using the complete knowledge about the probabilistic structure of classes [11]. Class conditional probability density functions (pdf's) and a priori class probabilities define the probabilistic structure. As can be deduced from (4-1), the overlap of the pdf's constitutes the classification error, thus the measure of class separability. Then, employing a criterion function '***J(.)***' where,

- $J(.) > 0$,
- $J(.)$ is max, when w_i and w_j are disjoint in x (data) space,

- *J(.)*=0, when pdf's are identical.

In [11], it is also indicated that the probabilistic dependence can be used as a basis for the measure of class separability.

4.1.4.1.1 Chernoff Bound

Considering the two-class problem, the criterion function of Chernoff bound is given as in (4-4).

$$J_C = -\ln \int p^s(x \mid w_1) p^{1-s}(x \mid w_2) dx \tag{4-4}$$

The parameter '*s*' lies within the interval [0,1]. Chernoff bound is an upper bound for ε (Bayes Error), where the tightness of the bound is determined by the value of parameter '*s*'.

4.1.4.1.2 Bhattacharyya Bound

Bhattacharyya distance measure is the special case of Chernoff Bound, where *s*=1/2. When the class separability is mainly defined due to the class means, experimental results indicate that the optimal '*s*' is close to ½ [17]. See (4-5) for the equation of Bhattacharyya bound (or distance).

$$J_B = -\ln \int \sqrt{p(x \mid w_1) p(x \mid w_2)} dx \tag{4-5}$$

4.1.4.1.3 Patrick-Fisher Bound

The criterion function for Patrick-Fisher bound is given in (4-6) below.

$$J_P = \left\{ \int \left[p(x \mid w_1)P(w_1) - p(x \mid w_2)P(w_2) \right]^2 dx \right\}^{(1/2)} \quad \textbf{(4-6)}$$

4.1.4.1.4 Entropy Measure

Making use of statistical entropy measure as a class separability criterion is an informational theoretic approach [7], [11]. This approach depends on the statistical dependence of the two variables *x (*feature vector*)* and *w* (class). How much information one can obtain from observing the outcome of *w,* given data vector *x* is the measure of separability power of feature vector. In other words, the uncertainty about the *w* determines the discriminative value of the feature vector *x* in that, the smaller the uncertainty the better the feature vector, *x*. For instance, when *w* and *x* are statistically independent, p($w|x$) will be equal to p(w), meaning that *x* do not convey any discriminative information about the *w* which it belongs to. Uncertainty is described as entropy. Then the procedure becomes as this; first observe *x,* then compute the entropy criterion function to determine the amount of discriminatory class information. Below in (4-7), a generalized form of entropy criterion function of degree α is given, where the number of classes is L.

$$J_L^{\alpha} = (2^{1-\alpha} - 1)^{-1}\left[\sum_{i=1}^{L} p^{\alpha}(w_i \mid x) - 1\right] \quad \textbf{(4-7)}$$

For α=1, applying L'Hospital Rule, the resulting criterion function obtained is known as 'Shannon Entropy' (4-8). The optimal feature vector set is obtained by minimizing (4-8).

$$J_S = -\int \sum_{i=1}^{L} p(w_i \mid x)\log_2\left[p(w_i \mid x)\right]p(x)dx \quad \textbf{(4-8)}$$

For α=2, the entropy criterion function is known as 'Quadratic entropy' which is given as (4-9).

$$J_L^2 = 2\left[1 - \sum_{i=1}^{L} p^2(w_i \mid x)\right] \quad \textbf{(4-9)}$$

The criterion function in this case is always less than 2. In order to minimize it, the summation expression is to be maximized. Corresponding criterion function is (4-10), which is known as 'Bayesian distance'.

$$J_Q = \int \sum_{i=1}^{L} p^2(w_i \mid x)p(x)dx \quad \textbf{(4-10)}$$

Further importance of the quadratic entropy is for the Nearest Neighbor decision rule, which will be presented in proceeding paragraphs.

4.1.4.1.5 Discussion

As seen, both first three well-known class separability measures (Chernoff, Bhattacharyya and Patrick-Fisher) are defined for the two-class problem. Each of them is related with ε in that, each provides a bound for ε and the reliability of the measures is evaluated according to the tightness of bounds. However, corresponding error bounds cannot directly be expressed in terms of the criterion functions and one should choose to use according to the feature extraction method in the design [11].

Also as seen for the Gaussian assumption for the Bhattacharyya bound, when the class conditional pdf's can be expressed in terms of any parametric distributions, further analytical simplification can be made on these class separability criterion functions, which also simplifies the estimation of the error bound.

In entropy approach, still a priori class probabilities and pdf's are required. There is no two-class restriction, however this makes no sense for the problem in this study. Besides, there is no direct relation with the optimal class separability, ε.

In our case, there exist several characteristics of the problem and restrictions on the data set, which makes these measures hard to apply. First of all, no significant assumption can be made easily about the class probabilities of m-class and f-class. Further, since there is not so big amount of data available, it is difficult to obtain reliable estimates of pdf's. Some several histogram plots given as samples in Appendix B can be evaluated as evidence. In addition to this, remembering our data generation is synthetic, the situation may be worse in terms of data amount for obtaining parametric pdf estimation in real world case.

In conclusion, parametric and probabilistic class separability concepts that require complete probabilistic information do not appear to be suitable for our problem. Thus, some other alternative approaches worth evaluating for adaptation to our case.

4.1.4.2 Non-Parametric Measures

Parametric approaches require the knowledge of the probabilistic structure of the data, which means the knowledge of the form of the distributions and the parameters of this form. However, in many realistic cases there is no way to know the form of the probabilistic distributions of data, further there may be multi-modal densities [9]. In order to circumvent this situation, two fundamental approaches of non-parametric techniques exist: k-nearest-neighbors (kNN) and Parzen window.

Both of these techniques make use of the local region concept around each sample feature vector, x and investigating the ratio of number of enclosed samples (k_{obs}) to the total number of samples (n). When this ratio is further normalized with the volume of local region V, (4-11) is obtained for p(x).

$$p(x) \approx \frac{k_{obs} / n}{V} \tag{4-11}$$

This expression converges under the assumptions:

- k (or n) goes to infinity (k/n goes to zero) or,

- V goes to zero.

The difference between the two approaches is that, Parzen technique fixes the volume of the region (V) whereas the kNN fixes the number of enclosed samples. Both methods converge; as a result, unknown probabilities can be obtained by using large number of samples. Though, it is very difficult to state significant arguments about their finite sample behavior [9]. Both of the two methods can be advantageous depending on data set characteristic and convergence performance. However kNN technique is popular due to its relation to Bayes error. For k=1 case (Nearest Neighbor), the error rate for NN is less than twice the error rate of Bayes (ε) for unlimited number of samples [9]. A further comparison of these methods can be found in [7].

4.1.4.2.1 Parzen

Parzen window approach depends on the selection of initial local region volume, V. For a hypercube of d dimensions, whose edge length is h_n, the number of samples falling in it can be obtained by using the window function in (4-12) [9].

$$k_n = \sum_{i=1}^{n} \Phi\left(\frac{x - x_i}{h_n}\right) \tag{4-12}$$

Corresponding estimate for the density is given in (4-13).

$$p_n(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{V_n}\Phi\left(\frac{x - x_i}{h_n}\right) \tag{4-13}$$

For finite data sets, in order to obtain an acceptable estimate, one has to seek for a value of h_n experimentally.

4.1.4.2.2 kNN

Instead of choosing the local region volume, V as an arbitrary function of number of samples as in the Parzen method, it is chosen as a function of the data for kNN case. In order to estimate p(x) from n samples, one shall locate a volume around each data set element x, and let it grow until k_n nearest neighbors are reached. In this case, k_n is a function of n. Resulting p(x) is given in (4-14) [9].

$$
\begin{aligned}
&p(x) = \frac{k_n / n}{V_n}, where \\
&\lim_{n\to\infty} k_n = \infty, \\
&\lim_{n\to\infty} \frac{k_n}{n} = 0
\end{aligned}
\tag{4-14}
$$

4.1.4.2.3 Discussion

Non-parametric approaches are most of the realistic cases required. kNN provides a bound to Bayes error, which is a desired relation with ε as the optimal class separability criterion. However, reliable estimates can be obtained only when the amount of the data is arbitrarily large. In our problem, synthetic data generation process could generate large amount of data. However in order to reflect the nature of the problem, availability of the data set is intentionally kept under a certain limit.

As another argument for non-parametric techniques is that, the computation cost of non-parametric methods can be very large for high dimensional data [7], [26], [23].

4.1.4.3 Interclass Distance Measures

Above summarized two groups of class separability measures: probabilistic and mathematical measures, and non-parametric measures are the alternatives to the optimal Bayes error rate. However, they require either the information on the probabilistic structure of the problem data domain or large amount of data sets for obtaining reliable estimates of density functions, difficulty in the usage and adaptation of them is apparent. Thus, a group of methods that can be introduced as alternatives mostly for circumventing the computational burden of the above methods are known as the interclass distance measures. These approaches are interpreted as geometric intuitive methods [7] and depend on the data set distance calculations in order to evaluate the class separability property of patterns. The distance between two feature vectors x_k and x_l can be defined as $d(x_k, x_l)$. Subsequently, the procedure basically is first to define the distance between two feature vectors of different classes, then to seek for the feature vectors that deliver the maximum average interclass distance between the elements of the classes by utilizing some feature set evaluation methods (feature subset selection or feature extraction).

Mostly, interclass distance methods do not give direct relation to Bayes error, however usage of these approaches present practical and systematic feature extraction stages [17], [11], [26]. In the following, first some several fundamental interclass distance metrics are presented and further usage of this metrics in class separability criterion functions will be presented.

4.1.4.3.1 Distance Metrics

Several common distance metrics are listed below. Dimension of the feature space is supposed to be *d*.

- *Minkowski Metric* (4-15)

$$d_M(x_k, x_l) = \left[\sum_{i=1}^{d} \left| x_{ik} - x_{il} \right|^s \right]^{1/s} \quad \textbf{(4-15)}$$

where *s* is the order parameter.

- *City Block Metric* (4-16) (Special case of (4-15), for *s*=1)

$$d_C(x_k, x_l) = \sum_{i=1}^{d} \left| x_{ik} - x_{il} \right| \quad \textbf{(4-16)}$$

- *Euclidian Metric* (4-17) (Special case of (4-15), for *s*=2)

$$d_M(x_k, x_l) = \left[\sum_{i=1}^{d} \left| x_{ik} - x_{il} \right|^2 \right]^{1/2} = \left[(x_k - x_l)^T (x_k - x_l) \right]^{1/2} \quad \textbf{(4-17)}$$

Euclidian metric allows for analytical, computational simplifications and physical interpretation. Therefore it is widely employed in criterion functions. Below in (4-18), a generalized form of criterion function in which interclass distance metrics are utilized is given.

$$J(x)=\frac{1}{2}\sum_{i=1}^{L}P(w_i)\sum_{j=1}^{L}P(w_j)\frac{1}{n_i n_j}\sum_{k=1}^{n_i}\sum_{l=1}^{n_j}d_M(x_k,x_l) \quad \textbf{(4-18)}$$

where L is the number of classes, n_i and n_j are the number of feature vectors and $P(w_i)$ is the a priori class probabilities.

4.1.4.3.2 Scatter Matrices

The main idea behind the Scatter Matrices is that the class separability is dependent on the separateness of the class means and tightness of class covariances. These concepts are expressed in terms of scatter matrices.

Using squared Euclidian distance as the metric (4-17) in the criterion function (4-18), one can obtain (4-19) as a raw criterion function, which is further processed in detail to obtain Scatter Matrices in [11].

Within-class and between-class scatter concepts depend on the first and the second terms respectively in (4-19), where m_i is the class mean vector and m is the total (or mixture) mean vector of all classes.

$$\begin{aligned}&\tilde{\tilde{J}}_1(x)=\sum_{i=1}^{L}P(w_i)\left[\frac{1}{n_i}\sum_{k=1}^{n_i}(x_{ik}-m_i)^T(x_{ik}-m_i)+(m_i-m)^T(m_i-m)\right]\\&where,\\&m=\sum_{i=1}^{L}P(w_i)m_i\end{aligned} \quad \textbf{(4-19)}$$

Within-class scatter matrix (S_W), between-class scatter matrix (S_B) and the mixture scatter matrix (S_M) are expressed in (4-20).

$$
\begin{aligned}
S_W &= \sum_{i=1}^{L} P(w_i) E\{(x-m_i)(x-m_i)^T \mid w_i\} = \sum_{i=1}^{L} P(w_i)\Sigma_i \\
S_B &= \sum_{i=1}^{L} P(w_i)(m_i - m)(m_i - m)^T \\
S_M &= S_W + S_B
\end{aligned}
\tag{4-20}
$$

Typical criteria can be found with detailed evaluations in [17], [30], [7], which are given in (4-21).

$$
\begin{aligned}
J_1 &= tr(S_2^{-1} S_1) \\
J_2 &= \ln\left|S_2^{-1} S_1\right| \\
J_3 &= \frac{tr(S_1)}{tr(S_2)}
\end{aligned}
\tag{4-21}
$$

Typical examples for $\{S_1, S_2\}$ are given as $\{S_B, S_W\}$, $\{S_B, S_M\}$ and $\{S_W, S_M\}$, regardless of comparing one of them to be advantageous among others [17]. A non-parametric expansion of scatter matrices can also be found in [17].

4.1.4.3.3 Non-parametric Class Overlap

Numerical assessment of class separability can be done by investigating the class overlap as presented in [12]. By taking the motivation from non-parametric scatter matrices and edited nearest neighbor approach, a class overlap concept can be introduced as a non-parametric measure of class separability. More specifically, individual class overlaps are calculated by employing k nearest neighbors and their corresponding class affiliations. However, in addition to the numerical assessment of overlap, a measure for the complexity of the class boundary is essential.

In [7], overlap sum is introduced to be a class separability measure and it is utilized in application of boundary methods. It is also demonstrated that the overlap sum is directly related to Bayes error rate with a much cheaper computational complexity.

4.1.4.3.4 Discussion

Interclass distance metric based criteria do not provide a true indication of class separability, since detailed information about probabilistic distribution of classes is not used [11]. In scatter matrices approach, no direct relation to Bayes error rate exists and it is limited to 2^{nd} order statistics, which is not always descriptive, or in some cases which may be misleading. Despite these factors, interclass distance methods are attractive for adaptation to many situations due to their computational effectiveness.

4.1.5 Feature Set Evaluation Methods

FSE methods make use of class separability measures to reach the optimal (or sub-optimal) feature sets in design. In other words, class separability discussions up to this point draw a picture of the tools that can be utilized for feature set evaluation purposes. Reduction of dimensionality to be obtained by FSE under an optimality criterion in terms of class separability can be investigated in two main groups of techniques, which were introduced before as 'Feature Selection' and 'Feature Extraction'.

More specifically, suppose that D measurements are taken over a data set, which means D features are gathered from the problem environment. Then, the dimension of the problem is D. In feature selection, feature space is evaluated by employing class separability measures in order to determine the optimal combination of d features out of D features, where $d \leq D$. In feature extraction context, class separability measures are utilized in order to optimize a (Dxd) linear transformation, W that will define the feature vector (x), where y is the unprocessed features having the dimensionality, D. (See (4-22))

$$x = W^t y \tag{4-22}$$

The concept followed in feature selection/extraction for classification purposes is somehow different from the concept for representation purposes. The criteria to assess the discriminative potential of features for classification must be a measure of class separability or overlap among data set, apart from being a measure of fit as the mean square error [17]. Therefore, although the Karhunen-Loeve transformation (also known as the Principal Component Analysis) is optimum for signal representation in the sense that it provides the smallest mean square error for a given number of features, the features defined by the Karhunen-Loeve transformation may not be optimum with regard to class separability.

4.1.5.1 FSE Approaches Based On Feature Selection

Feature selection or feature subset selection is a widely studied subject both in pattern recognition and information theory literature [17], [19], [22], [24], [27], [11], [30]. In general, feature selection methods have two components, the criterion function, formerly defined as *J*(.) and a search engine for finding those optimal features. According to the characteristics of the criterion function, feature selection methods can be investigated under two subdivisions: wrappers and filters, as more recent approaches. While filter approach does not include any classifier concern during feature selection process, wrapper approaches incorporates classifier modules as a complete classification optimization step.

Basically, *J(.)* should be employed each feature set for ranking them and the search engine should utilize an algorithm, preferably one other than the 'exhaustive search' in order to find out d optimal features out of D. It is apparent that even for moderate values of d and D, direct exhaustive search becomes impossible to handle, see (4-23).

$$\binom{D}{d} = \frac{D!}{(D-d)!d!} \tag{4-23}$$

The only optimal search method that makes it possible to inspect the feature space without the exhaustive search is the well known 'Branch and Bound' algorithm [17], [11]. In this algorithm, inspection of all the possible combinations of features is allowed in a computationally effective manner achieved by an effective organization of the search process. However, in order to be able use Branch and Bound method, 'monotonicity' assumption should be held by the criterion function. According to this assumption, if S_i is a feature set, where i=1,2,…,k, for (4-24),

$$S_1 \supset S_2 \supset \ldots \supset S_k \tag{4-24}$$

(4-25) should be held. In other words, criterion function should increase with the dimensionality of the data set.

$$J(S_1) \geq J(S_2) \geq \ldots \geq J(S_k) \tag{4-25}$$

For many data sets of pattern recognition problems, the Branch and Bound search may not be effective in terms of computational performance. In addition to this, the monotonicity assumption may not be valid in order it to be applied. Even if the monotonicity assumption is applicable, for data sets with more than thirty features, exponential nature of the search space may lead to an infeasible search [19]. In these circumstances, some several suboptimal search methods can be employed. Best feature selection (selecting the features yielding the best rankings individually), sequential forward selection and backward selection (SFS and SBS) are some of the suboptimal search methods, details of which are given in [11], [17].

To reiterate for convenience, the criterion function does not have any relation with the searching algorithms to be applied in feature selection approaches. Therefore any suitable criterion function can be chosen depending on the data set characteristics. It may be either a parametric one such as Mahalanobis distance under normality assumption, Bhattacharyya distance or a non-parametric interclass distance metric such as the scatter matrices or overlap sum.

4.1.5.2 FSE Approaches Based On Feature Extraction

Feature extraction methods, being different from feature selection techniques, basically deduce the features by processing the input features through a linear transformation as in (4-25). In this scope, ideally, class separability criterion functions are optimized for all the possible (Dxd) linear transformations, W, which define the resulting feature vector x (dx1), from the unprocessed input features $y(D$x$1)$, where $D \geq d$. Thus, one should take (4-26) into account for the feature extraction context [11].

$$J(x) = \max_{W} J(W^T y) \qquad \textbf{(4-26)}$$

Further $J(W)$ to be derived from the criterion functions presented under class separability should be optimized in terms of the employment of the feature extractor W. This optimization often requires numerical computation of the gradient of $J(W)$. However, since the criterion function is not often differentiable, further gradient estimation operations are to be employed for optimization [11]. Discriminant analysis is well known to be a computationally simple gradient estimation method and is studied and applied widely in the literature [14], [17], [11], [23].

As well as linear transformations, non-linear transformations can also be employed up to a certain information loss level as feature extractors. However, non-linear transformations are more computational power demanding.

Using probabilistic distance measures, non-parametric approaches and intuitive interclass distance measures, so many feature extraction methods are studied and applied in literature. A detailed discussion and theory of the feature extraction fundamentals can be found in [11], [17], [13], [21]. Some of the additional and relatively more recent work about feature extraction can be found in [10], [18], [20], [23], [25], [26].

4.2 Implemented FSE Approaches

In the scope of the study, the FSE approaches adapted and applied to our problem domain is explained in this section. These approaches can be categorized under feature selection and feature extraction according to the way the final feature set is derived. Under these categories, implementation of each method is explained below.

4.2.1 Feature Selection Approaches

Under this category, the final feature vector set to be utilized in the classification module is generated as a subset of the existing features in accordance with the background given previously. As discussed in the Section 4.1.5.1, Branch and Bound algorithm, which yields the optimal feature combination, is not practical, the suboptimal searching methods are employed, which are the **Sequential Forward Selection** (**SFS**) algorithm and **Best Features Selection (BFS)**. Both are implemented in software environment and incorporated with the FSE techniques.

4.2.1.1 Searching Methods

4.2.1.1.1 Sequential Forward Selection (SFS) Algorithm

SFS is a basic bottom-up search procedure. At each iteration step, a new feature is added to the current feature set [11]. The implemented SFS algorithm can be summarized as given in [17]. The first step of the SFS algorithm is the computation of the criterion results of each individual features. Taking the 'best' individual feature as the starting feature, every other remaining features are individually combined to the starting feature and the corresponding criterion function is computed. The best resulting combination is then taken as the basis for the third step of the iteration. After the desired dimensionality is reached, the iteration stops. Consequently, a sub-optimal feature combination is obtained.

Despite this algorithm does not provide the optimal subset, it is computationally simple. Also no monotonicity assumption is required, which is not realistic in high dimensional problems even if the appropriate criterion function is selected [19]. Since for the methods whose criterion function requires long computation time, SFS is not applied. The main purpose for the implementation of SFS is to make a comparison of SFS and BFS where the computation is fast, as a by-product of the study.

4.2.1.1.2 Best Features Selection (BFS)

In addition to the SFS algorithm implementation, **Best Features Selection (BFS)** approach is also utilized as the alternative sub-optimal searching method. According to the BFS, the resulting sub-optimal feature subset is the combination of the features, which yields best rankings individually.

4.2.1.2 Bhattacharyya Approach

In order to see the effect of the parameterization performed depending on the sample histograms, normal distribution fitting was applied depending on the sample mean and standard deviations (See Appendix B). In the following, the fitted distribution parameters are used in this feature set evaluation method and corresponding classification results are compared with other non-parametric techniques. Even if the classes have mostly non-parametric distributions, in [11] it is stated that it still may be advantageous to employ parametric approaches.

First, Bhattacharyya approach should be adapted. When a pdf's are assumed to be Gaussian, the error bound is obtained to be the equation (4-27) [7].

$$\varepsilon_u = \sqrt{P(w_1)P(w_2)}e^{-J_B} \tag{4-27}$$

where;

$$J_B = \frac{1}{8}(M_2 - M_1)^T \left[\frac{\Sigma_1 + \Sigma_2}{2}\right]^{-1}(M_2 - M_1) + \frac{1}{2}\ln\frac{\left|\frac{\Sigma_1 + \Sigma_2}{2}\right|}{\sqrt{|\Sigma_1||\Sigma_2|}} \tag{4-28}$$

where M_1, M_2 are the class means and Σ_1, Σ_2 are the class covariances.

Normal distribution parameters were estimated for every individual feature. Utilizing the criterion function above, both two search methods (SFS and BFS) were employed to find out the sub-optimal feature combination. The resulting feature combinations are summarized in Table 4-1, Table 4-2 and Table 4-3.

Further, data sets were classified by the kNN classifier. First the original data sets were classified. Then the dimensionality reduction applied data sets were classified for evaluating the performance of this FSE method. Classification results of the original and reduced data sets are compared in the Chapter 6.

Since the data sets belonging to the odd numbered scenarios were employed for constructing the training set of the classifier, these sets were processed by the FSE approaches. Even numbered data sets were used as the test set. This is valid for the whole study.

Table 4-1 Bhattacharyya Feature Selection (BAFS) for type-1 dataset

t1w15s4_odd		t1w20s4_odd	
BFS	**dim=10**	**BFS**	**dim=10**
ndx	**J**	**ndx**	**J**
18	0.67029	27	1.9747
33	0.6538	30	1.9747
22	0.56359	28	1.9747
19	0.56359	18	0.70818
20	0.56359	33	0.62458
26	0.39452	22	0.44705
38	0.27792	19	0.44705
35	0.27792	20	0.44705
36	0.27792	26	0.39777
21	0.26999	36	0.39624
SFS		**SFS**	
dim=10	**dim=8**	**dim=10**	**dim=8**
f_comb	**f_comb**	**f_comb**	**f_comb**
18	18	27	27
20	20	18	18
19	19	26	26
40	40	30	30
39	39	13	13
1	1	19	19
2	2	20	20
36	36	37	37
3		28	
4		2	

On the tables, 'ndx' represents the best feature index and 'f_comb' represents the 'feature combination' as the result of the SFS algorithm.

Table 4-2 Bhattacharyya Feature Selection (BAFS) for type-2 dataset

t2w15s4_odd		t2w20s4_odd	
BFS	dim=10	BFS	dim=10
ndx	J	ndx	J
28	0.77366	26	0.67192
26	0.77366	28	0.67192
25	0.77366	25	0.67192
24	0.77366	24	0.67192
23	0.52209	41	0.59637
41	0.45186	23	0.55068
21	0.4053	10	0.30939
33	0.25943	9	0.30939
10	0.21729	29	0.27949
9	0.21729	30	0.27949
SFS		SFS	
dim=10	dim=8	dim=10	dim=8
f_comb	f_comb	f_comb	f_comb
28	28	26	26
24	24	28	28
50	50	50	50
49	49	6	6
48	48	25	25
5	5	49	49
25	25	48	48
23	23	9	9
26		10	
47		27	

Table 4-3 Bhattacharyya Feature Selection (BAFS) for type-3 dataset

t3w15s4_odd		t3w20s4_odd	
BFS	dim=10	BFS	dim=10
ndx	J	ndx	J
21	1.5345	21	1.6811
22	0.85081	22	0.91879
32	0.57409	24	0.61026
37	0.45186	37	0.59637
24	0.40979	23	0.50348
29	0.40098	29	0.41708
23	0.27825	32	0.34898
30	0.22599	10	0.30939
10	0.21729	9	0.30939
9	0.21729	30	0.24433
SFS		SFS	
-		-	

As can be seen, dimensionality of both types of datasets was reduced to 8 and 10.

4.2.1.3 Overlap Sum Approach

Numerical assessment of class separability was computed by investigating the class overlap as presented in [12]. According to the algorithm, which is described below, an overlap sum estimator was implemented. Class overlap of each individual feature is calculated and ranked. Then, the best features were chosen according to the BFS. Procedure is given in [12] as follows:

The k nearest neighbors of every feature vector is investigated and their class affiliations are examined. Overlap degree is determined for an individual vector P as q_0 below in (4-29).

$$q_0 = \frac{\sum_{i=1}^{n} q_{x_{NN_i}} + \sum_{i=1}^{n} k_i}{2\sum_{i=1}^{n} k_i};$$

$$k_i = 1 - \frac{d_{NN_i}}{d_{NN_n}}; \qquad \text{(4-29)}$$

$$q_{x_{NN_i}} = \begin{cases} k_i : w_x = w_i \\ -k_i : w_x \neq w_i \end{cases}$$

Where k_i stands for the weighting factor for the position of the i-th nearest neighbor NN_i, d_{NNi} denotes the distance between P and NNi, d_{NNn} designates the distance between P and the most distant nearest neighbor NNn, qx_{NNi} stands for the overlap for P with regard to NNi, w_x is the class of P and w_i is the class of NNi. The influence of every NNi decaying with the position in the nearest neighbor list is defined. Hence the weighting factors k_i are defined such that the influence decreases to zero for NNn. The quality q_x is increased by one k_i when the class belongings are the same ($w_x = w_i$), however it is decreased by same amount when the class belongings are different.

The resulting feature rankings are given in Table 4-4, Table 4-5, Table 4-6. The dimension of both three types of datasets were reduced to 10 in this case.

Table 4-4 Overlap Sum Feature Selection (OSFS) for type-1 dataset

t1w15s4_odd		**t1w20s4_odd**	
BFS	dim=10	**BFS**	dim=10
ndx	J	ndx	J
18	0.82513	18	0.82544
13	0.80887	5	0.80026
5	0.80299	13	0.79980
2	0.79370	2	0.79449
10	0.78353	10	0.77732
21	0.73782	9	0.77074
17	0.72250	11	0.74320
22	0.72011	14	0.74258
19	0.71943	21	0.74012
20	0.71656	19	0.73740

Table 4-5 Overlap Sum Feature Selection (OSFS) for type-2 dataset

t2w15s4_odd		t2w20s4_odd	
BFS	dim=10	**BFS**	dim=10
ndx	J	ndx	J
17	0.80510	17	0.80858
7	0.79510	7	0.79369
13	0.79416	13	0.79195
3	0.78353	3	0.78922
27	0.74958	25	0.76964
28	0.74253	28	0.76673
25	0.74236	24	0.76665
24	0.74195	27	0.74717
15	0.72741	21	0.72645
18	0.72722	23	0.72611

Table 4-6 Overlap Sum Feature Selection (OSFS) for type-3 dataset

t3w15s4_odd		t3w20s4_odd	
BFS	dim=10	**BFS**	dim=10
ndx	J	ndx	J
21	0.80872	21	0.82780
17	0.80510	24	0.81132
7	0.79510	17	0.80858
13	0.79416	7	0.79369
24	0.78531	13	0.79195
3	0.78353	3	0.78922
22	0.75613	22	0.76818
25	0.74585	25	0.73630
23	0.73484	14	0.72040
15	0.72741	15	0.72037

4.2.1.4 Scatter Matrices Approach

A scatter matrices feature selection methodology is implemented, which utilizes between-class (S_B) and within-class (S_W) scatter matrices for the criterion function, SFS and BFS approaches for the searching purposes in accordance with the background given in paragraph 4.1.4.3.2. Three types of criterion functions are given in (4-21) with the general matrix names S1 and S2.

Both three functions were implemented, however only $J_3(.)$ will be used in the scope of this approach. The justification for this choice depends on an observation made during the simulation runs. Since the dimensionality of the data sets is high, singularity problem can arise regarding the scatter matrices, when $J_1(.)$ and/or $J_2(.)$ are to be employed, which include matrix inversion operations. In order to avoid the singularity problem, $J_3(.)$ was used in simulations. The criterion function is re-stated here for convenience in (4-30).

$$J_3 = \frac{tr(S_1)}{tr(S_2)} \tag{4-30}$$

For $\{S_1, S_2\}$ pair two choices $\{S_B, S_W\}$ and $\{S_B, S_M\}$ were implemented. As an observation it was seen that $\{S_B, S_W\}$ and $\{S_B, S_M\}$ yielded very similar resulting features. Thus, the corresponding results of $\{S_B, S_W\}$ are presented in the following part of the thesis.

As another point, since the calculation of the scatter matrices includes a priori class probabilities, assumptions were required to be made on these probabilities. Regarding the class probabilities $P(w\_m)$ and $P(w\_f)$, a pair of assumptions was made as (0.5, 0.5) and (0.2, 0.8) accordingly.

The resulting feature sets are summarized in Table 4-7. Dimensions of both data sets were reduced to 8.

Table 4-7 Scatter Matrices Feature Selection (SMFS) for type-1 dataset

t1w15s4_odd				t1w20s4_odd			
P(m)=P(f)=0.5		P(m)=0.2, P(f)=0.8		P(m)=P(f)=0.5		P(m)=0.2, P(f)=0.8	
BFS	dim=8	BFS	dim=8	BFS	dim=8	BFS	dim=8
ndx	J	ndx	J	ndx	J	ndx	J
33	0.15272	33	0.27315	7	0.16	33	0.26623
13	0.13339	38	0.14912	8	0.16	36	0.23878
31	0.11459	35	0.14912	38	0.15811	38	0.23878
32	0.11459	36	0.14912	36	0.15811	35	0.23878
10	0.11159	32	0.13423	35	0.15811	8	0.22249
38	0.10598	31	0.13423	33	0.15462	7	0.22249
35	0.10598	40	0.094727	31	0.13772	31	0.15029
36	0.10598	39	0.094727	32	0.13772	32	0.15029

SFS	SFS	SFS	SFS
f_comb, dim=8	f_comb, dim=8	f_comb, dim=8	f_comb, dim=8
33	33	7	33
22	22	8	36
13	20	38	38
38	19	36	35
36	38	35	22
35	35	33	8
20	36	22	7
19	25	20	20

Table 4-8 Scatter Matrices Feature Selection (SMFS) for type-2 dataset

t2w15s4_odd				t2w20s4_odd			
P(m)=P(f)=0.5		P(m)=0.2, P(f)=0.8		P(m)=P(f)=0.5		P(m)=0.2, P(f)=0.8	
BFS	**dim=8**	**BFS**	**dim=8**	**BFS**	**dim=8**	**BFS**	**dim=8**
ndx	**J**	**ndx**	**J**	**ndx**	**J**	**ndx**	**J**
41	0.1467	41	0.23933	41	0.19803	41	0.33459
17	0.1334	39	0.12778	40	0.16114	40	0.17113
39	0.1203	40	0.12778	39	0.16114	39	0.17113
40	0.1203	49	0.11243	17	0.13765	9	0.13343
50	0.11707	50	0.11243	50	0.11334	10	0.13343
49	0.11707	10	0.080302	49	0.11334	49	0.1048
13	0.11193	9	0.080302	13	0.10772	50	0.1048
46	0.060897	17	0.079591	10	0.092829	20	0.091834

SFS	SFS	SFS	SFS
f_comb, dim=8	**f_comb, dim=8**	**f_comb, dim=8**	**f_comb, dim=8**
41	41	41	41
17	21	40	28
21	28	39	26
22	26	31	24
28	39	21	25
39	40	28	40
40	25	26	39
26	24	25	21

Table 4-9 Scatter Matrices Feature Selection (SMFS) for type-3 dataset

t3w15s4_odd				t3w20s4_odd			
P(m)=P(f)=0.5		P(m)=0.2, P(f)=0.8		P(m)=P(f)=0.5		P(m)=0.2, P(f)=0.8	
BFS	dim=8	BFS	dim=8	BFS	dim=8	BFS	dim=8
ndx	J	ndx	J	ndx	J	ndx	J
37	0.1467	37	0.23933	37	0.19803	37	0.33459
17	0.1334	22	0.22408	30	0.141	22	0.23648
22	0.12007	30	0.15105	17	0.13765	30	0.17452
30	0.11954	45	0.11243	22	0.12948	9	0.13343
46	0.11707	46	0.11243	46	0.11334	10	0.13343
45	0.11707	24	0.10651	45	0.11334	24	0.12928
13	0.11193	10	0.080302	13	0.10772	45	0.1048
24	0.065182	9	0.080302	10	0.092829	46	0.1048

SFS	SFS	SFS	SFS
f_comb, dim=8	f_comb, dim=8	f_comb, dim=8	f_comb, dim=8
37	37	37	37
36	36	36	36
35	35	35	35
17	22	30	22
30	30	22	30
22	26	17	24
26	24	1	26
1	40	26	1

4.2.2 Feature Extraction Approaches

In feature extraction as explained earlier in the theoretical background, a transformation matrix is facilitated in order to yield new features, unlike in the feature selection, where a subset of the current features is selected. Since there is no systematical way to find out an optimum non-linear transformation, linear transforms are investigated more generally. In accordance with this, two feature extraction methodologies were realized in software environment in this thesis study, the outcomes of which were used in classification simulations.

4.2.2.1 Principal Component Analysis Approach

Since the Principal Component Analysis is well-known procedure mainly utilized for signal representation, no theoretical background is given this thesis, rather the implemented procedure is explained below.

In Principal Component Analysis (PCA) approach, the procedure for the feature extraction is as follows. First the feature vector set for the training set of the classifier is specified (odd numbered data sets in our case). Before the PCA to be performed, a former normalization stage is in general necessary for whitening purposes, so that the input data set would have zero mean and unit variance. Then, PCA is applied for the training set. The tolerable elimination level of variance is determined for the selection of the eigenvalues accordingly. After the selection of the non-zero eigenvalues, the transformation matrix is derived. This transformation matrix is restored together with the normalization parameters so that further new input vectors can be scaled similarly and transformed by the same matrix for dimensionality reduction purposes.

For the whole data sets used as the training set in the classifier, the percentage of the elimination of the total variation was adapted between 3.8% and 4.5%, so that dimensions of all the data set types were reduced to 8. Corresponding classification performances is compared in Chapter 6.

4.2.2.2 Non-parametric Decision Boundary Approach

Non-parametric Decision Boundary feature extraction is a relatively new algorithm based on decision boundaries for non-parametric classifiers [26]. The method that is proposed by Lee and Landgrebe in [26] was implemented and employed in this study. Necessary additional adaptations are noted following in this section. An explanation of the implementation is presented below. For further details, one can refer to [26].

Feature extraction is addressed as keeping the “discriminantly informative features" and the definition of a discriminantly informative feature is dependent on the decision boundary of the classifier to be used. Therefore here, the feature set evaluation for dimensionality reduction is performed as being incorporated with the classification stage.

Lee and Landgrebe make the definition of a ‘**discriminantly informative feature**’ and **‘discriminatively redundant feature**’ in [26] (See Figure 4-2). In order to extract discriminantly informative features and discriminantly redundant features from the decision boundary, **‘decision boundary feature matrix (DBFEM)’** is defined also as following.

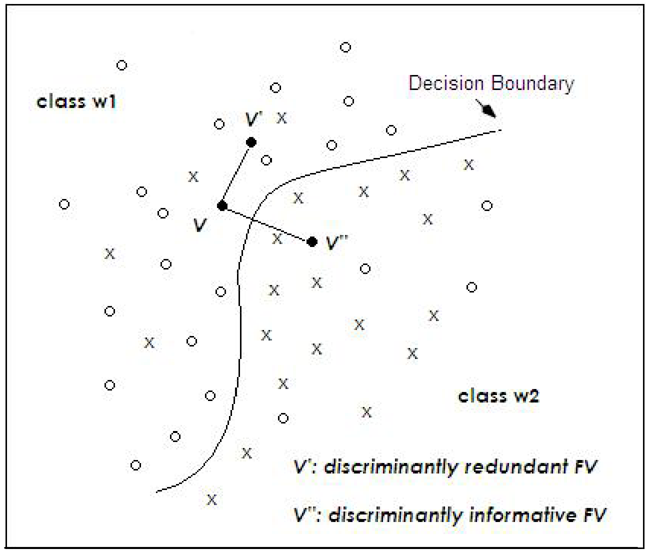

Figure 4-2 Discriminantly informative and redundant features in Decision Boundary Feature Extraction

Let '*Nx*' be the unit vector normal to the decision boundary at a point ***X*** on the decision boundary for a given data set. Then the decision boundary feature matrix M_{DBFM} is defined in (4-31).

$$M_{DBFM} = \frac{1}{K}\int_S N_X N_X^t \, p(x)dx$$
$$where,$$
$$K = \int_S p(x)dx \tag{4-31}$$

The unit normal vector '*Nx*' is calculated as in (4-32).

$$N_x = \frac{\nabla h(x)}{|\nabla h(x)|}$$
$$where \qquad (4\text{-}32)$$
$$\nabla h(x) \approx \frac{\Delta h}{\Delta x_1} x_1 + \frac{\Delta h}{\Delta x_2} x_2 + \ldots + \frac{\Delta h}{\Delta x_n} x_n$$

Where, *h(x)* is defined as in (4-33) below, by assuming Bayes' decision rule for minimum error is used.

$$h(x) = -\ln \frac{P(x \mid w_1)}{P(x \mid w_2)} \qquad (4\text{-}33)$$

In [26] and [18] it is theoretically justified that the rank of the decision boundary feature matrix yields the smallest dimension by which the same classification performance could be obtained as in the original data set.

Decision boundary feature extraction procedure for non-parametric classification is presented as follows. In [26], a Parzen classifier is used to estimate the density function of the data set, however a kNN classifier is utilized in this study. So necessary assumptions were to be made in accordance with [26] especially for determining the unit normal vectors on the decision boundary.

- **Step1.** Classify the training set data and sort out correctly classified samples, x_c.

- **Step2.** For each x_c of class_m, find the nearest x_c of class_f.

- **Step3.** Connect the pairs found in Step2. Find the point P_i on the estimated decision boundary that lies on the connecting line between the pair of samples at the same time.

- **Step4.** At the point P_i, estimate the unit normal vector N_i to the decision boundary. Since no class probability density is defined by kNN classifier, *h(x)* cannot be calculated as in (4-33) to be used in (4-32). Therefore, unit normal vector is estimated in accordance with [26] as the vector along which the classification result varies most rapidly. In other words, the smallest Δx_i is investigated, such that the classification result of ($X+\Delta x_i$ $\mathbf{x}_i$**)** is different from that of X. According to this assumption, (4-34) is used below.

$$N = \frac{V}{|V|}$$
$$where \qquad \textbf{(4-34)}$$
$$V \approx \frac{1}{\Delta x_1} x_1 + \frac{1}{\Delta x_2} x_2 + \ldots + \frac{1}{\Delta x_n} x_n$$

- **Step5.** Estimate the decision boundary feature matrix (M_{DBFM}_1) using the unit normal vectors found in **Step4** by (4-35) below.

$$M_{DBFM} = \sum_i N_i N_i^t \qquad \textbf{(4-35)}$$

- **Step6.** Follow the steps up to here for the f_class and derive M_{DBFM}_2. Then, find the resulting decision boundary feature matrix M_{DBFM} as the sum of M_{DBFM}_1 and M_{DBFM}_2.

- **Step7.** Select the eigenvectors of the M_{DBFM} to construct the transformation matrix for feature extraction.

CHAPTER V

CLASSIFIER DESIGN

The resulting data sets of implemented feature set evaluation (FSE) techniques are evaluated according to the outputs of class separability criterion functions regarding each technique. However, final evaluation criterion of comparing various FSE approaches is the classification performance. In order to assess the applied approaches in terms of classification performance a kNN classifier is implemented in software environment.

The evaluation strategy is as follows. First, the classification performance of the raw data set, the dimensionality of which is not reduced by any FSE method, is computed via simulation. Then, the classification performances of the processed data sets, which are generated by certain FSE approaches from the same input data set, are computed and compared.

In the design of the classifier and during the evaluation stage, the data set is divided into two non-overlapping subsets as the 'Training Set' and the 'Test Set'. In the implementation, data sets corresponding to 'odd' numbered scenarios were used for the training set and the data sets derived from 'even' numbered scenarios are used as the test set.

5.1 kNN Classifier

Voting kNN rule is applied in the classification design. The rule is simply as follows for the two-class case:

- Take the data vector x to be classified.
- Take 'k' elements of the training data set in the neighborhood of x.
- Examine the class labels of these k-nearest-neighbors in order to make a vote.
- Make the class decision for x according to the winner label corresponding to neighbors.

In the asymptotic case (when the data set size is infinitely large), kNN rule converges to true estimation of the probability density function. Also, as stated earlier in the theoretical background section, there exists a bound for the Bayes error rate in kNN classification. The error estimation for the finite data set case will not be given since the kNN is a well-known approach, which is widely analyzed and discussed. Further analysis on the error estimate for the finite data set size case can be found in [15], [16], [17], [13].

5.2 Implementation

In our case, the system has to generate a decision by performing classification in each frame starting after the accumulation of at least one window size width of samples for the model fitting purpose. After this period, in a pipelined fashion, feature vectors are calculated in each consecutive frame by moving the model-fitting window over the samples.

During the feature vector set (data set) generation processes up to this point, sub-sampling parameter '*s*' was chosen as 2 and 4. The purpose of this sub-sampling was to keep the data set size in a convenient size, by eliminating the close vectors. By sub-sampling in other words, the redundancy in the training set was aimed to be reduced. However for the testing of scenarios during simulations, such a sub-sampling is not applied, since the classification is performed in every consecutive frame.

The neighborhood parameter '*k*' is chosen as 5 and 7 during the simulations. The classifier operation procedure is simply as follows.

- Starting from the first feature vector of a scenario, feed the classifier by each consecutive feature vector,

- Perform classification and generate the resulting outcome. '0' stands for the classification result of a non-threatening object in the scenario, while '1' stands for the classification result of a threat.

- If the outcome is '0', continue classification by processing the consecutive frame, else stop classification and declare an alarm.

When classification results obtained according to the procedure above, one can easily realize that the number false alarms is high. In order to mitigate this high false alarm rate problem, the classifier operation procedure is slightly modified by introducing a 'sequence' parameter '*seq*'. The parameter '*seq*' defines the number of consecutive frames that the classifier has to wait before making a declaration. This means that, the classifier does not declare an alarm for the first frame that the classification outcome is '1'. Rather, operating in a pessimistic mode, it counts the consecutive frames that the output is '1'. Hence, one can state that the parameter '*seq*' defines the level of pessimism of the classifier.

By the utilization of '*seq*', number of false alarms was reduced. Corresponding simulation results are given in Table 5-1 below for the data set 't1w20s4', with no dimensionality reduction is applied. Results are similar for the other datasets.

Table 5-1 Classification results of t1w20s4, no dimensionality reduction applied

Training Set: ODD
Test Set: EVEN

kNN Paramers:		k=5 seq=1	k=3 seq=1	k=7 seq=1	k=5 seq=3	k=5 seq=4	k=7 seq=4
t1w20s4	POCD(%)	100	100	100	100	96,87	95,312
	FA (#)	28	31	25	22	18	17

In the table above 'POCD(%)' stands for the Probability of Correct Declaration in percentages, and FA(#) means the number of False Alarms resulted in the corresponding simulation. In the test set above, 64 m-scenarios and 52 f-scenarios exist. False Alarm (FA) result is given in number. General convention for an MDS regarding the FA characteristic is expressing the FA rate (FAR), which is the ratio of number of FA's to the flight time (in hours) in the scenario. For realistic case, the flight time in the definition stands for the duration of operational scenarios. However, since the f-scenarios generated in simulation environment represent special maneuvers to the FA sources in short scenario durations, they do not indicate operational flight behavior. Therefore, since expressing the FAR might be misleading, only the total number of FA's is presented.

Throughout the study in further simulations, the parameters are chosen as k=5 and seq=4, which yields convenient results comparatively.

CHAPTER VI

SIMULATION RESULTS AND PERFORMANCE EVALUATION

The evaluation strategy mainly depends on the classification performance. Since the primary goal of this study is to investigate the methods of feature set evaluation, the evaluation criterion depends on the comparisons of the classification performances of the original data sets (no FSE/dimensionality reduction applied) and the processed data sets, on which dimensionality reduction is employed. In addition to this main assessment, additional further investigations are held and corresponding discussions are noted.

6.1 Simulation Results

Simulation results regarding the data set type: 't1w15s4' is presented in the Table 6-1. In the table, BAFS means Bhattacharyya Feature Selection, OS means Overlap Sum Feature Selection, SMFS means Scatter Matrices Feature Selection, PCAFE means Principal Component Analysis Feature Extraction and DBFE means Decision Boundary Feature Extraction. The rows other than the first row corresponds to further investigations related with the approaches BAFS and SMFS.

Table 6-1 Simulation Results of the dataset: t1w15s4

t1w15s4								
Training Set:		ODD						
Test Set:		EVEN						
Original Dimension:		40						
Reduced Dimension after FSE:		8						
Tested m-scenarios:		64						
Tested f-scenarios:		52						
Training Set Size:		2774						
Test Set Size:		11248						
kNN Paramers: **k=5, seq=4**		No FSE	OS	BAFS	SMFS	PCAFE	DBFE
BAFS:BF	POCD(%)	100.00	76.56	95.31	96.88	98.44	98.44
SMFS: BF, Pm=0.2, Pf=0.8	FA (#)	20	29	26	38	19	27
BAFS:FS	POCD(%)	-	-	96.88	98.44	-	-
SMFS: FS, Pm=0.2, Pf=0.8	FA (#)	-	-	29	38	-	-
	POCD(%)	-	-	-	92.19	-	-
SMFS: BF, Pm=0.5, Pf=0.5	FA (#)	-	-	-	35	-	-
	POCD(%)	-	-	-	92.19	-	-
SMFS: FS, Pm=0.5, Pf=0.5	FA (#)	-	-	-	41	-	-

Simulation results regarding the data set type: 't1w20s4' is presented in the Table 6-2.

Table 6-2 Simulation Results of the dataset: t1w20s4

t1w20s4							
Training Set:		ODD					
Test Set:		EVEN					
Original Dimension:		40					
Reduced Dimension after FSE:		8					
Tested m-scenarios:		64					
Tested f-scenarios:		52					
Training Set Size:		2627					
Test Set Size:		10692					
kNN Paramers: **k=5, seq=4**		No FSE	OS	BAFS	SMFS	PCAFE	DBFE
BAFS:BF	POCD(%)	96.88	92.19	90.63	100.00	98.44	100.00
SMFS: BF, Pm=0.2, Pf=0.8	FA (#)	18	29	26	19	15	21
BAFS:FS	POCD(%)	-	-	96.88	98.44	-	-
SMFS: FS, Pm=0.2, Pf=0.8	FA (#)	-	-	28	16	-	-
	POCD(%)	-	-	-	100.00	-	-
SMFS: BF, Pm=0.5, Pf=0.5	FA (#)	-	-	-	19	-	-
	POCD(%)	-	-	-	98.44	-	-
SMFS: FS, Pm=0.5, Pf=0.5	FA (#)	-	-	-	16	-	-

Simulation results regarding the data set type: 't2w15s4' is presented in the Table 6-3.

Table 6-3 Simulation Results of the dataset: t2w15s4

t2w15s4							
Training Set:		ODD					
Test Set:		EVEN					
Original Dimension:		50					
Reduced Dimension after FSE:		8					
Tested m-scenarios:		64					
Tested f-scenarios:		52					
Training Set Size:		2774					
Test Set Size:		11248					
kNN Paramers: **k=5, seq=4**		No FSE	OS	BAFS	SMFS	PCAFE	DBFE
BAFS:BF	POCD(%)	98.44	71.87	98.44	98.44	96.88	95.31
SMFS: BF, Pm=0.2, Pf=0.8	FA (#)	21	30	16	25	20	19
BAFS:FS	POCD(%)	-	-	98.44	98.44	-	-
SMFS: FS, Pm=0.2, Pf=0.8	FA (#)	-	-	24	18	-	-

Simulation results regarding the data set type: 't2w20s4' is presented in the Table 6-4.

Table 6-4 Simulation Results of the dataset: t2w20s4

t2w20s4							
Training Set:		ODD					
Test Set:		EVEN					
Original Dimension:		50					
Reduced Dimension after FSE:		8					
Tested m-scenarios:		64					
Tested f-scenarios:		52					
Training Set Size:		2627					
Test Set Size:		10692					
kNN Paramers: **k=5, seq=4**		No FSE	OS	BAFS	SMFS	PCAFE	DBFE
BAFS:BF	POCD(%)	98.44	75.00	96.88	100.00	96.88	95.31
SMFS: BF, Pm=0.2, Pf=0.8	FA (#)	15	32	14	18	16	19
BAFS:FS	POCD(%)	-	-	98.44	98.44	-	-
SMFS: FS, Pm=0.2, Pf=0.8	FA (#)	-	-	15	15	-	-

Simulation results regarding the data set type: 't3w15s4' is presented in the Table 6-5.

Table 6-5 Simulation Results of the dataset: t3w15s4

t3w15s4								
Training Set:		ODD						
Test Set:		EVEN						
Original Dimension:		46						
Reduced Dimension after FSE:		8						
Tested m-scenarios:		64						
Tested f-scenarios:		52						
Training Set Size:		2774						
Test Set Size:		11248						
kNN Paramers: **k=5, seq=4**		No FSE	OS	BAFS	SMFS	PCAFE	DBFE	
BAFS:BF	POCD(%)	60.94	87.50	87.50	42.19	78.13	79.69	
SMFS: BF, Pm=0.2, Pf=0.8	FA (#)	14	28	36	14	22	29	
BAFS:FS	POCD(%)	-	-	-	-	-	-	
SMFS: FS, Pm=0.2, Pf=0.8	FA (#)	-	-	-	-	-	-	

Simulation results regarding the data set type: 't3w20s4' is presented in the Table 6-6.

Table 6-6 Simulation Results of the dataset: t3w20s4

t3w20s4								
Training Set:		ODD						
Test Set:		EVEN						
Original Dimension:		46						
Reduced Dimension after FSE:		8						
Tested m-scenarios:		64						
Tested f-scenarios:		52						
Training Set Size:		2627						
Test Set Size:		10692						
kNN Paramers: **k=5, seq=4**		No FSE	OS	BAFS	SMFS	PCAFE	DBFE	
BAFS:BF	POCD(%)	70.31	87.50	96.88	21.87	85.94	82.81	
SMFS: BF, Pm=0.2, Pf=0.8	FA (#)	11	29	47	2	19	21	
BAFS:FS	POCD(%)	-	-	-	-	-	-	
SMFS: FS, Pm=0.2, Pf=0.8	FA (#)	-	-	-	-	-	-	

6.2 Discussions

The most critical performance parameter of an MDS is obviously the POCD (Probability of Correct Declaration). However, FAR (False Alarm Rate) is also important for the reliability of the system and for the fact that dispensable countermeasures are finite. Hence, the number FA's should also be taken into account when the performance evaluation is being made. FAR is desired to be low. Further techniques can be employed for FAR reduction, which is not directly under the scope of this study.

Investigating the first rows of the table, one can observe that in general the methods SMFS, PCAFE and DBFE yield better results.

For t1w15s4 and t1w20s4, the best POCD was given by the SMFS approach. DBFE yielded also a 100 percent of correct declaration rate. However the number of FA's is more than that of SMFS approach. The minimum number of FA's was provided by PCAFE. In the overall, SMFS and PCAFE are the successful methods.

Both methods yielded satisfactory results but considering the classification performance of the original (unreduced) data sets, PCAFE is the best for t1w15s4 and the DBFE can be considered to the best for t1w20s4.

For t2w15s4 and t2w20s4, the best POCD was given by the SMFS and the BAFS approach. The minimum number of FA's was provided by BAFS and SMFS approaches for both t2w15s4 and t2w20s4. In the overall, both BAFS, SMFS, PCAFE and DBFE techniques yielded satisfactory results, whereas the first two are the better ones.

Considering the classification performance of the original (unreduced) data sets, BAFS is the best approach for t2w15s4 and the SMFS technique together with the BAFS can be considered to the best for t2w20s4.

Having a general look over the whole data sets, one can observe that the classification performance of the data sets t3w15s4 and t3w20s4 are lower than the other data set. This result indicates that the modeling type of those data sets is not suitable for our case.

Referring again to the simulation results of the first two types of data sets (t1w… and t2w…), one can observe that the SFS (Sequential Forward Selection) algorithm yielded better results in general than the BFS (Best Features Selection) method. However, in the overall the classification performances are close.

Further investigation was performed for the SMFS method as another evaluation point. In this approach, two different sets of a priori class probabilities, which one cannot make a reliable estimate in the realistic case for our problem, were employed in the class separability criterion function as (Pm=0.2, Pf=0.8) and (Pm=Pf=0.5). In the problem domain, however it is strongly scenario dependent, the argument that the a priori class probability of the m_class is lower than the f_class is meaningful, since the samples of f_class, which is the clutter class, are much more probable to be observed considering the natural and man-made sources of radiation other than a missile. Therefore, the classification performances regarding these two a priori probability cases are to be compared. As can be seen from the simulation results, relatively better POCD were gathered for t1w15s4, for

(*Pm*=0.2, *Pf*=0.8). However, no change was observed for t1w20s4. In the overall not so big classification performance difference was obtained for this two a priori class probability pairs for the SMFS approach. As a note, no additional simulation is run for the remaining sets depending on the results.

As an additional study, variation of the classification performance of some certain methods according to the dimensionality variation was investigated. For this purpose, only the approaches that yield the best classification performance were chosen for each type of data set. Fluctuating the resulting dimension between 3 and 12 (and 13), the trend of the classification performance variation was observed. Results of this investigation are presented below both in table and graph forms for better visualization.

For t1w20s4, PCAFE was selected for this investigation. Below in Table 6-7, the classification results according to each dimension (from 3 to 12) are given. Also in Figure 6-1, the variation of POCD (%) and FA (#) vs. dimension can be seen.

As can be observed from the Table 6-7 and Figure 6-1, dimension = 8 can be evaluated as a good compromise regarding the POCD and FA.

Table 6-7 Variation of the classification performance according to the dimensionality for the data set t1w20s4 by employing PCAFE approach

t1w20s4
PCAFE

dim	3	4	5	6	7	8	9	10	12
POCD(%)	96,88	96,88	95,31	96,88	96,88	98,44	95,31	96,88	98,44
FA(#)	18	19	21	23	17	15	19	16	17

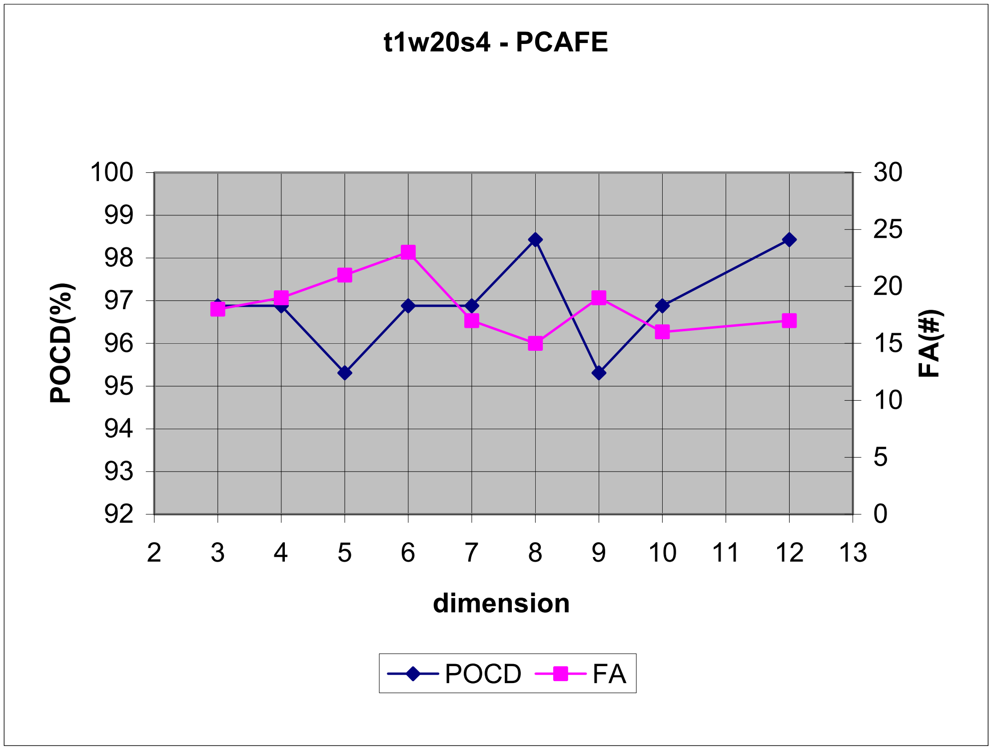

Figure 6-1 Variation of the classification performance according to the dimensionality for the data set t1w20s4 by employing PCAFE approach

For t2w20s4, BAFS and SMFS approaches were selected for this investigation, where Sequential Forward Selection (SFS) was utilized. Below in Table 6-8, the classification results according to each dimension (from 3 to 12) are given for BAFS-SFS. Also in Figure 6-2, the variation of POCD (%) and FA (#) vs. dimension can be seen for this approach.

As one can realize from the Table 6-8 and Figure 6-2, dimension = 8 and dimension = 10 can be evaluated to be acceptable dimensionalities regarding the POCD and FA.

Table 6-8 Variation of the classification performance according to the dimensionality for the data set t2w20s4 by employing BAFS –SFS approach

t2w20s4
BAFS

dim	3	4	5	6	7	8	9	10	12
POCD(%)	95,31	93,75	93,75	92,19	98,44	98,44	98,44	98,44	95,31
FA(#)	21	27	27	26	21	15	15	14	20

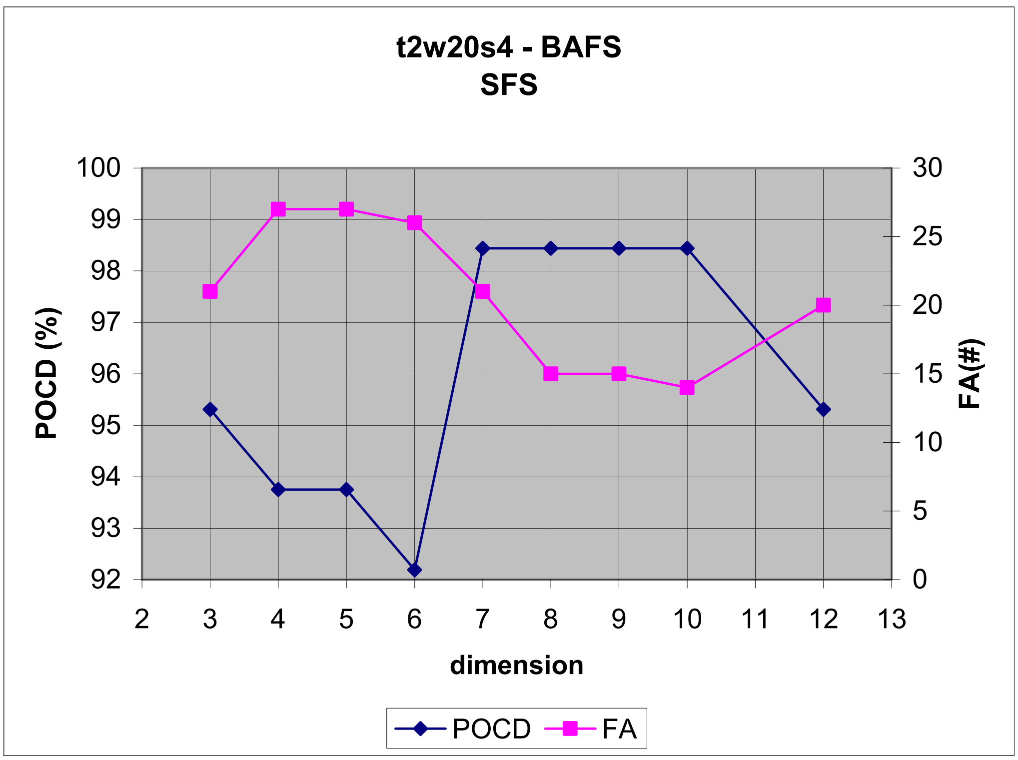

Figure 6-2 Variation of the classification performance according to the dimensionality for the data set t2w20s4 by employing BAFS – SFS approach

Below in Table 6-9 the classification results according to each dimension (from 3 to 13) are given for SMFS-SFS. As the a-priori class probabilities, the values Pm=0.2 for the m-class and Pf=0.8 for the f-class were assumed. In

Figure 6-3, the variation of POCD (%) and FA (#) vs. dimension can be seen for this approach.

As can be seen from the Table 6-9 and Figure 6-3, both dimension = 7, dimension = 8 and dimension = 12 can be assessed to be convenient dimensionalities taking the POCD and FA into account.

Table 6-9 Variation of the classification performance according to the dimensionality for the data set t2w20s4 by employing SMFS-SFS approach (Pm=0.2, Pf=0.8)

t2w20s4
SMFS

dim	3	4	5	6	7	8	9	10	12	13
POCD(%)	95,31	95,31	95,31	98,44	98,44	98,44	98,44	98,44	98,44	100
FA(#)	18	18	18	17	15	15	22	22	14	17

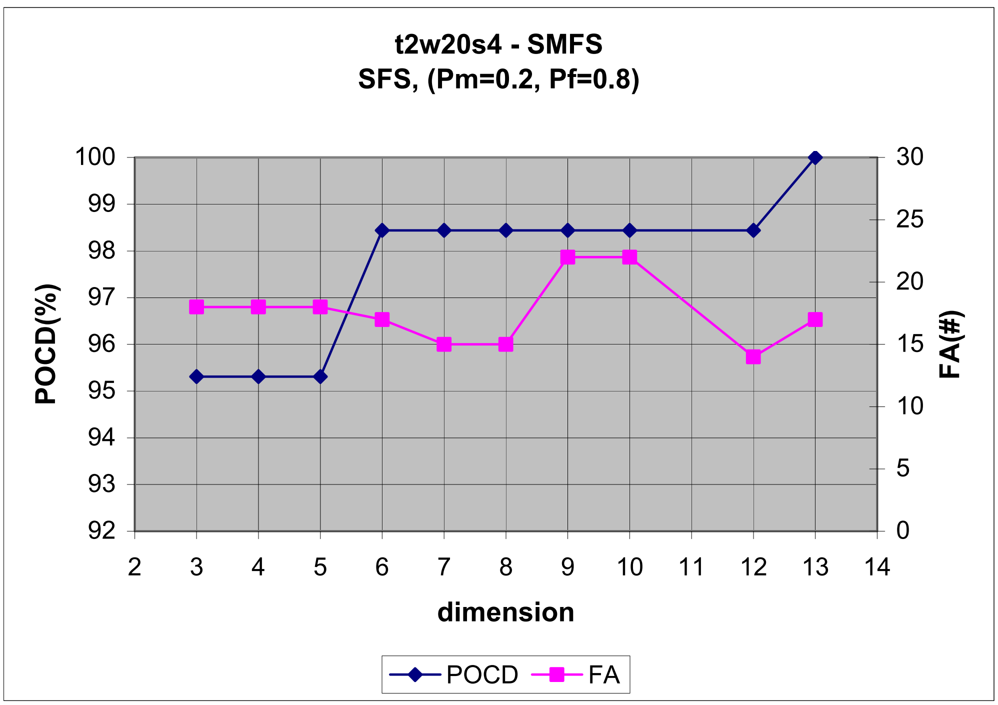

Figure 6-3 Variation of the classification performance according to the dimensionality for the data set t2w20s4 by employing SMFS-SFS approach, where Pm=0.2, Pf=0.8

Another additional investigation is the Receiver Operating Characteristic (ROC), which is carried out for the approaches yielding close classification performances for the data sets t1w20s4 and t2w20s4. For this investigation a threshold value (or confidence value) is to be introduced to the kNN classifier. Since the classifier is implemented according to the 'voting kNN' principle, the operation depends on the majority vote; therefore an adjustable threshold value is not directly applicable. Due to the fact that the ROC supplies valuable information for comparison, the slight modification regarding the employment of the threshold value is implemented according to (6-1) below.

$$\begin{aligned} score &= raw\_score * weight, \\ raw\_score &= 1 - \frac{k_{FA}}{k} \\ weight &= 1 - \frac{d_{mean-m} / d_{mean-f}}{6} \end{aligned} \tag{6-1}$$

where k_{FA} is the number of f-class vectors in the k neighborhood, d_{mean-m} and d_{mean-f} is the mean of the distances of the vectors belonging to the m-class and f-class respectively to the incident vector. The '*raw_score*' comes from the voting principle and the '*weight*' is introduced as the effect of the distances of nearest vectors. As a result, when the mean distances of vectors of both classes are similar, the score will be calculated around 0.5. Therefore the operation will be approximate to the original voting kNN, when the threshold is adjusted to 0.5 while the parameters *seq*=4 and *k*=5.

The ROC curve obtained for the data set t1w20s4 is given in Figure 6-4 below.

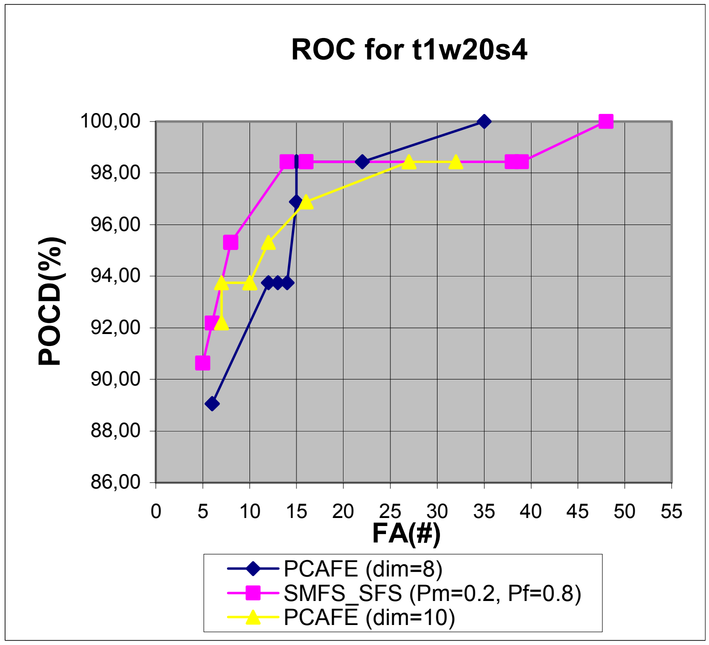

Figure 6-4 The ROC curve for t1w20s4, where PCAFE (dim=8 and dim=10) and SMFS_SFS (dim=8) are employed

The threshold values (T) vary from 0.1 and 1, where the upper-right corner of the graph corresponds to T=0.1 and the lower-left corner corresponds to T=1. Corresponding values of the ROC curve is given below in Table 6-10.

Table 6-10 ROC investigation values for t1w20s4

t1w20s4						
	PCAFE (dim=8)		PCAFE (dim=10)		SMFS_SFS (dim=8)	
threshold	POCD(%)	FA(#)	POCD(%)	FA(#)	POCD(%)	FA(#)
0,1	100,00	35	98,44	32	100,00	48
0,2	98,44	22	98,44	27	98,44	39
0,3	98,44	22	98,44	27	98,44	38
0,35	98,44	15	96,88	16	98,44	16
0,4	98,44	15	96,88	16	98,44	16
0,45	98,44	15	96,88	16	98,44	16
0,5	96,88	15	95,31	12	98,44	14
0,55	93,75	14	93,75	10	95,31	8
0,6	93,75	13	93,75	10	95,31	8
0,65	93,75	12	93,75	10	95,31	8
0,7	89,06	6	93,75	7	92,19	6
0,8	89,06	6	92,19	7	90,63	5
1	89,06	6	92,19	7	90,63	5

As can be seen from the graph, the ROC for SMFS_SFS reaches to higher POCD and lower FA faster than PCAFE up to a point and then they indicate similar characteristic.

Previously, the variation of the classification performance according to the dimensionality is investigated. In order to support the observations with the ROC curve an example case is employed. Referring to Figure 6-1 and Table 6-7, the classification performance for PCAFE dim=8 appears to be better than that of the case where dim=10 for t1w20s4. In parallel to these results, the ROC for dim=8 is preferable to that of the data set with dim=10, as it reaches to higher POCD while maintaining lower FA faster.

The ROC curve obtained for the data set t2w20s4 is given in Figure 6-5 below.

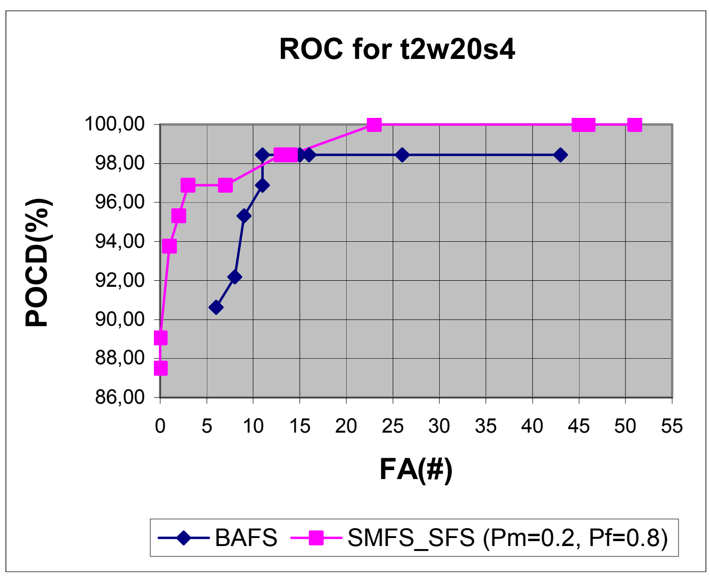

Figure 6-5 The ROC curve for t1w20s4, where BAFS_SFS (dim=8) and SMFS_SFS (dim=8) are employed

Similar to the t1w20s4 case, the threshold values (*T*) vary from 0.1 and 1, where the upper-right corner of the graph corresponds to *T*=0.1 and the lower-left corner corresponds to *T*=1. Corresponding values of the ROC curve is given below in Table 6-11.

Table 6-11 ROC investigation values for t2w20s4

t2w20s4				
	BAFS_SFS (dim=8)		SMFS_SFS (dim=8)	
threshold	POCD(%)	FA(#)	POCD(%)	FA(#)
0,1	98,44	43	100,00	51
0,2	98,44	26	100,00	46
0,3	98,44	26	100,00	45
0,35	98,44	16	100,00	23
0,4	98,44	15	98,44	14
0,45	98,44	15	98,44	13
0,5	98,44	14	96,88	7
0,55	98,44	11	96,88	3
0,6	96,88	11	95,31	2
0,65	95,31	9	93,75	1
0,7	92,19	8	89,06	0
0,8	90,63	6	87,50	0
1	90,63	6	87,50	0

Regarding the ROCs of BAFS_SFS and SMFS_SFS, it is seen that SMFS_SFS yields lower FA up to a point, however BAFS_SFS reaches a better classification result for an operating point where POCD(%)=98.44 and FA=11.

CHAPTER VII

CONCLUSION

In this thesis study, feature set evaluation approaches were investigated in the pattern recognition problem environment of a generic missile detection system. In this problem environment, which was synthetically created in software, feature vector data sets were generated and then processed via the implementations of some certain FSE (Feature Set Evaluation) approaches to make a systematic assessment both in terms of class separability criteria and classification performances. The main purpose of the FSE is to reduce the dimensionality. The resulting classification performances of the reduced data sets being the product of various FSE approaches are compared with the classification performances of the unreduced original (full dimension) data sets. Observing the simulation results, one can realize that the employment of FSE approaches results in various classification performances for each data set type, some of which are at satisfactory levels considering the realistic performance requirements of an MDS. Moreover, the classification performances of the reduced sets of some certain FSE techniques are even better compared to that of full dimension sets. More specifically, the POCD (Probability of Correct Declaration) increases and the number of FA's (False Alarms) decreases for some occasions, by the application of a convenient FSE technique.

The main aim of the study was to investigate answer to the question: "How will the discriminatively effective features be selected/extracted among the candidates, so that the dimensionality is reduced and accordingly the classifier can be trained and operate with discriminatively powerful feature sets at the desired dimensionality?" This thesis study addresses this question by a systematical approach applied from the input data generation through the implementation of FSE methods based on theoretical background in literature. The notion of Class Separability was incorporated with the main feature generation concepts as Feature (Subset) Selection and Feature Extraction in the implementations. The main understanding regarding the use of the employment of FSE approaches is gained and this outcome is supported by the simulation results.

The route of the study can be summarized as follows. The problem environment is established synthetically via software simulations under certain assumptions and adaptations by utilizing missile engagement and false alarm source scenarios, in which certain generic models are used. Feature vector data sets are generated by processing the outputs of the simulations for the further parts of the study. Feature vector data set generation is one of the major implementation parts of this study, which can be taken as the input generation part basically. After the synthetic data generation process, the characteristics of data sets are analyzed in order to draw a baseline for further feature set evaluation approaches. Further, a theoretical background including the concepts of Class Separability, Feature (Subset) Selection and Feature Extraction is presented. Several widely used methods are assessed and their convenience for the problem is discussed by giving necessary justifications depending on the data set characteristics. Depending on the theoretical background, software implementations are realized regarding several feature set evaluation techniques, with necessary adaptations. Reduced-dimension feature vector sets are generated by each FSE implementation. For the evaluation of the resulting data sets in terms of classification performance, a classifier software implementation is realized. As the assessment methodology, first simulations are carried out for the original full-dimension data sets and the classification performance is noted. Then, simulations are re-run for the resulting reduced data sets of each FSE technique. The classification performances of the original and reduced data sets are compared and evaluated.

More specifically, the following can be concluded from the study. The classification results of the reduced data sets, which are the product of the BAFS approach, indicated that the parametric approaches may also yield good results even if it is difficult to derive reliable probabilistic information from the data set. Simulation results verified the argument that the employing Gaussian assumption for the class distributions and applying parametric class separability criteria may be advantageous even in problems with classes having non-parametric distributions, since the resulting error due to the Gaussian assumption may be lower than the error coming from the non-parametric distribution estimations.

Simulation results regarding the non-parametric approach DBFE indicates that such non-parametric approaches may be useful for the problems as in our case with irregular decision boundaries, where non-parametric classifiers are implemented. In this synthetically created problem environment, the data sets can be generated arbitrarily large for both two classes, however the situation is obviously different for the realistic case. In order the non-parametric estimations of the class densities to be reliable and convergent, the data set size (sample size) should be large.

Another conclusion regarding the simulation results of the implementation of SMFS approach is that the FSE approaches, which utilize interclass distance measures as the class separability criterion yield satisfactory classification performance. This outcome indicates that these methods can be applicable for the problems where deriving probabilistic information is not trivial. These approaches are computationally simple, however they produce meaningful results at the end if the class means are not so close and the variances are tight enough.

As stated before in the theoretical background section, in the literature it is discussed that whereas the PCA is the optimal concept in signal representation, it may not generate discriminatively optimal features. However in our problem, evaluating the simulation results regarding the method of PCAFE, one can clearly conclude that there are cases where this method yielded the best classification performance. The reasoning for this result may be discussed to be related with the generation method of the features. The modeling by curve fitting in the least square sense approach is based on best representing the temporal characteristics of a track. This 'modeling for the best representation' approach can be discussed to be somehow in parallel with the notion of extracting principal components in PCA approach. Therefore PCA appeared to be advantageous for our case.

Another evaluation can be made among the data set types regarding the window size parameter 'w'. As can be observed from the simulation results, classification performances corresponding to the data sets, where w=20 is better in general than those with *w*=15, for both types t1 and t2. This result can be explained with the modeling quality of the temporal characteristic. The phase characteristic of the track data can better represented by *w*=20. Also the other conclusion to be derived is that when *w* is selected to be equal to 20, the minimum data accumulation time before starting to generate features is longer than the case when *w*=15 is selected, as obvious. However the classification performance is better for *w*=20. Operationally thinking, starting to generate features as early as possible is important for generating early decision. Depending on the experimental results, one can discuss that when we start with a smaller window size and then grow to the bigger window in consecutive frames, it would be a better approach for both early processing start and better classification performance.

As a future work, the systematic approach followed in this study can be adapted and applied to the realistic case. FSE techniques can be employed for the data sets gathered from real recordings and scenario simulations generated by the utilization of realistic models. The expectation for the realistic data sets is that mainly the FSE approaches based on the class separability criteria related to interclass distance measures would be more convenient rather than probabilistic and mathematical measures. For the utilization of non-parametric approaches, sample set size should be high enough to obtain reliable estimates of density functions. Since the sample set size cannot be arbitrarily large especially for the m-class, non-parametric approaches may not yield satisfactory results.

As stated in the previous chapter, the performance evaluation strategy is mainly depending on the classification performance. While the primary concern is the POCD, the FAR is the other essential factor an MDS regarding the capability of the counter measure system in cooperation and the reliability of the system. High FAR is a major recent problem of today's systems. As another future work, special care can be taken for FAR reduction in the employment of FSE approaches. Together with FSE, the classifier optimization can be carried out accordingly in order to reach an acceptable FAR while the POCD is maintained at the desired rate.

REFERENCES

[1] Neri, F. "*Introduction to Electronic Defense Systems*", 2nd edition, Artech House, Inc., 2001.

[2] DeCosimo, L. et al., "*Passive UV Missile-warning Systems"*, The Journal of Electronic Defense, December 1999, pp. 51-54

[3] Schleher, D.C., "*Introduction to Electronic Warfare*", Artech House, 1986

[4] Foster, J. P., Richardson, M. A., "*Options for the Defeat of Solar Blind Ultra Violet Wavelength Missile Warning Sensors*", Journal of Battlefield Technology, Vol.4, No.3, November 2001, pp. 30-35

[5] Herskovitz, D., "*You Bet Your Life: Today's Missile Warning Systems*", The Journal of Electronic Defense, April 2000, pp. 51-56

[6] Heikell, J., "*Electronic Warfare Self-Protection of Battlefield Helicopters: A Holistic View*", Helsinki University of Technology, Applied Electronics Laboratory Series E: Electronics Publications E18

[7] Pierson, W. E., "*Using Boundary Methods in Estimating Class Separability, Dissertation",* The Ohio State University, 1998

[8] Rothe, H., "*Approaches to Pattern Recognition, Advanced Pattern Recognition Techniques*", NATO RTO-EN-2 AC/323 (SET) TP/1, 1998

[9] Duda, O., Hart, E., Stork, G., "*Pattern Classification*", Wiley-Interscience Publication, 2001

[10] Biem, A., Katagiri, S., Juang, B., "*Pattern Recognition Using Discriminative Feature Extraction*", IEEE Transactions on Signal Processing, pages 388-400, April 1993.

[11] Devijer, P.A., Kettler, J., "*Pattern Recognition: A Statistical Approach*", Prentice Hall, 1982

[12] König, A., Bulmahn, O., Glesner, M., *"Systematic Methods for Multivariate Data Visualization and Numerical Assessment of Class Separability and Overlap in Automated Visual Industrial Quality Control"*, Proceedings of the 5th British Machine Vision Conference BMVC'94, September 1994, 195 – 204.

[13] Devroye, L., Gyorfi, L., Lugosi, G., "*A Probabilistic Theory of Pattern Recognition*", Applications of mathematics 31, Springer, New York, 1996.

[14] Fix, E., Hodges J. L., "*Discriminatory Analysis, Nonparametric Discrimination: Consistency Properties"*. Technical Report 21-49-004, USAF School of Aviation Medicine, Randolph Air Field, TX, 1949. Report No. 4.

[15] Fukunaga, K., Hummels, D. M., "*Bayes Error Estimation Using Parzen and knn Procedures"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, pages 634-643, September 1987.

[16] Fukunaga, K., Hummels, D. M., "*Bias of Nearest Neighbor Error Estimates*", IEEE Transactions on Pattern Analysis and Machine Intelligence, pages 103-112, January 1987.

[17] Fukunaga, K., "*Introduction to Statistical Pattern Recognition*", Academic Press, New York, 2nd edition, 1990.

[18] Lee, C., Landgrebe, D. A., "*Feature Extraction Based on Decision Boundaries*", IEEE Transactions on Pattern Analysis and Machine Intelligence, pages 388-400 April 1993.

[19] Scott, M. J. J., Niranjan, M., Prager, R., Parcel, W.," *Feature Subset Selection in Variable Cost Domains*", Technical Report, Cambridge University, Cambridge, May 1998.

[20] Xie, Q., Laszlo, C., A., Ward, R. K., "*Vector Quantization Technique for Nonparametric Classifier Design*", IEEE Transactions on Pattern Analysis and Machine Intelligence, pages 1326-1330, December 1993.

[21] Young, T. Y., Fu, K., "*Handbook of Pattern Recognition and Image Processing*", Academic Press, Orlando, Florida, 1986.

[22] Kohavi, R., John, G. H., "*Wrappers for Feature Subset Selection*", Standford University, May 1997

[23] Lee, C., Landgrebe D. A., "*Feature Extraction and Classification Algorithms for High Dimensional Data"*, School of Electrical Engineering Purdue University, January 1993

[24] Wang, L., Chan, K. L., "*Learning Kernel Parameters by using Class Separability Measure"*, School of Electrical and Electronic Engineering Nanyang Technological University, Singapore

[25] Kuo, B. C., Landgrebe, D. A., "*Nonparametric Weighted Feature Extraction for Classification"*, IEEE Transactions on Geoscience and Remote Sensing, Volume 42 No: 5, pp 1096-1105, May 2004

[26] Lee, C., Landgrebe, D. A., "*Decision Boundary Feature Extraction for Non-Parametric Classification"*, IEEE Transactions on System, Man and Cybernetics, Vol. 23 No.2, pp 433-444, March/April 1993.

[27] Cantu-Paz, E., "*Feature Subset Selection, Class Separability and Genetic Algorithms*", Genetic and Evolutionary Computation Conference, Seattle WA, June 26-30 2004.

[28] Giordano, A. A., Hsu, F. M., "*Least Square Estimation with Applications to Digital Signal Processing*", Wiley-Interscience, 1985

[29] MATLAB® Curve Fitting Toolbox Documentation

[30] Wang, L., Hartley, R., "*Kernel-based Class Separability: An Efficient Feature Selection Criterion For Image Classification*" Research School of Information, Science and Engineering (RSISE), Australian National University, Nov. 2005.

APPENDIX A

CORRELATION PLOTS OF DATA SETS

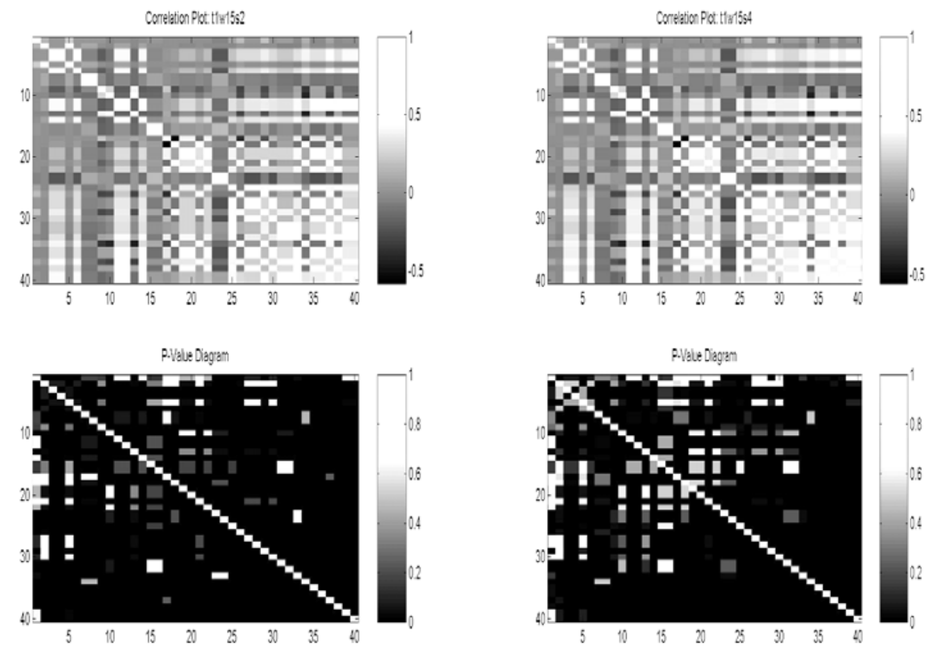

Figure A-1 Correlation plots for data type: t1w15s2 and t1w15s4

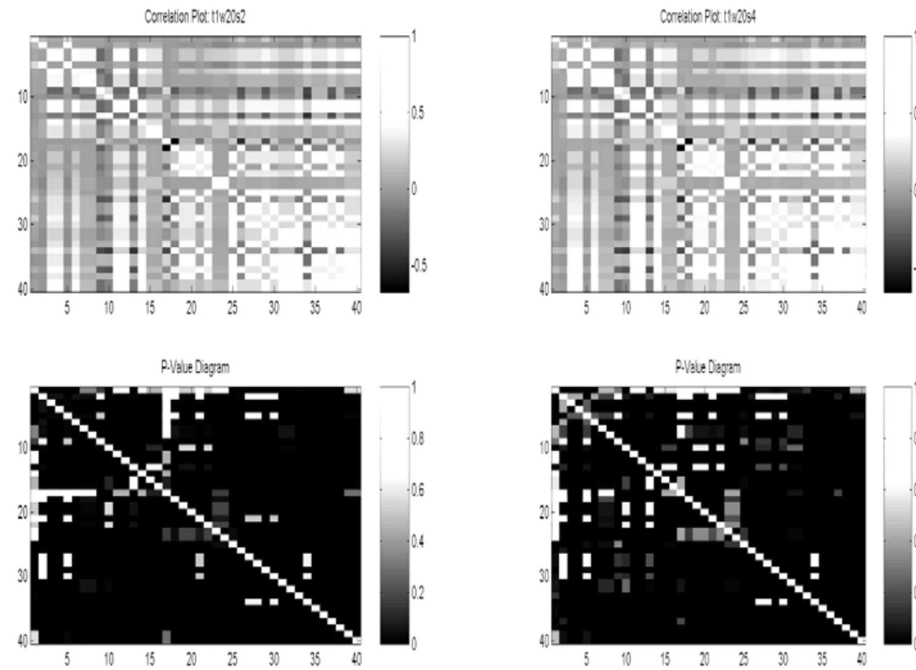

Figure A-2 Correlation plots for data type: t1w20s2 and t1w20s4

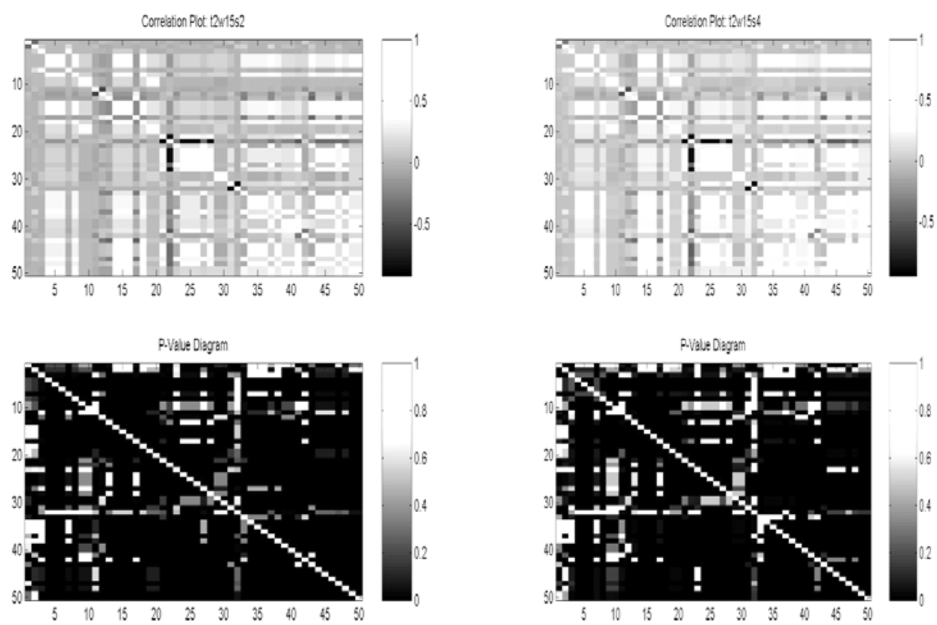

Figure A-3 Correlation plots for data type: t2w15s2 and t2w15s4

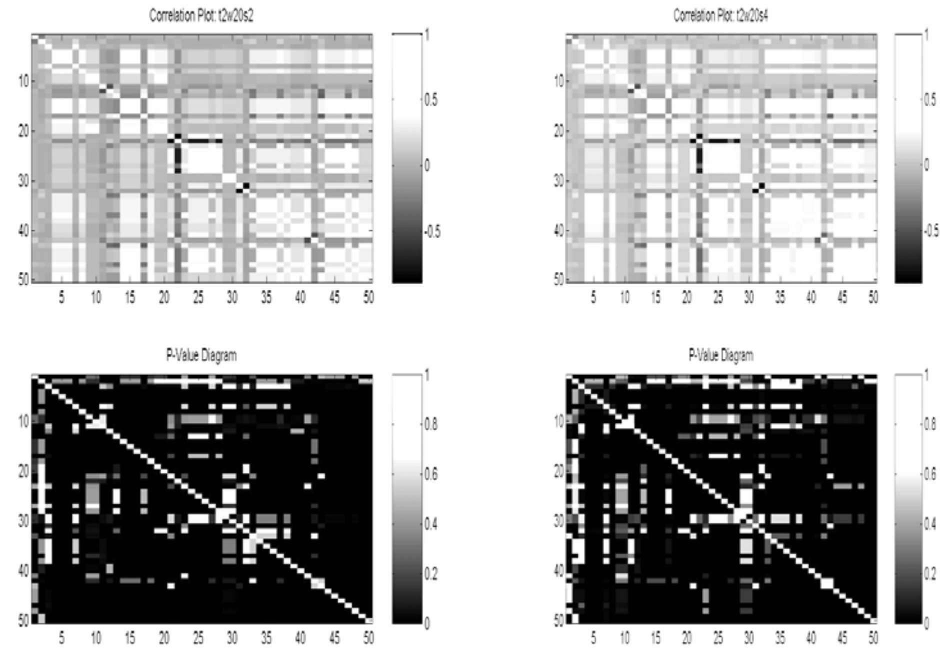

Figure A-4 Correlation plots for data type: t2w20s2 and t2w20s4

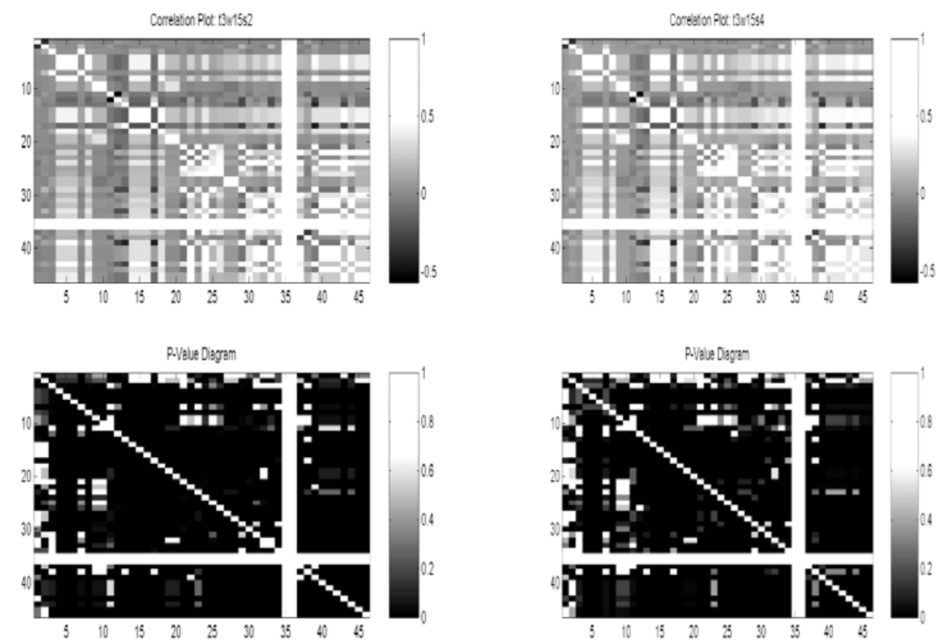

Figure A-5 Correlation plots for data type: t3w15s2 and t3w15s4

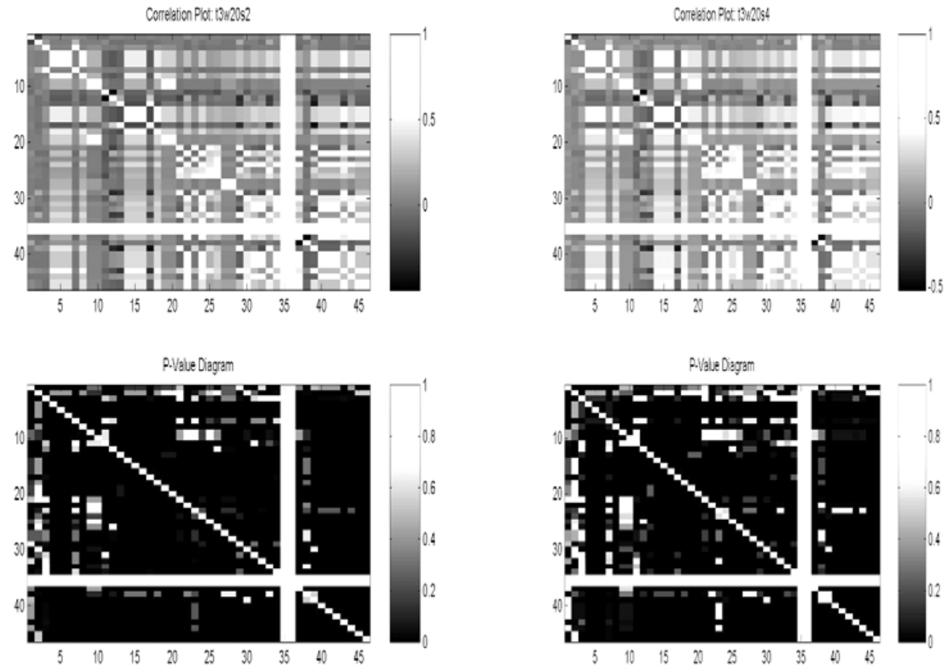

Figure A-6 Correlation plots for data type: t3w20s2 and t3w20s4

APPENDIX B

HISTOGRAM PLOTS

Histogram Plots for the data set ‘t1w20s4_odd’ is given below in the following figures. ‘_odd’ represents that this data set is constructed from only ‘odd numbered’ scenarios. As a note, not all the histogram plots corresponding to each of 40 features but a just a selection of them is given in order to save space.

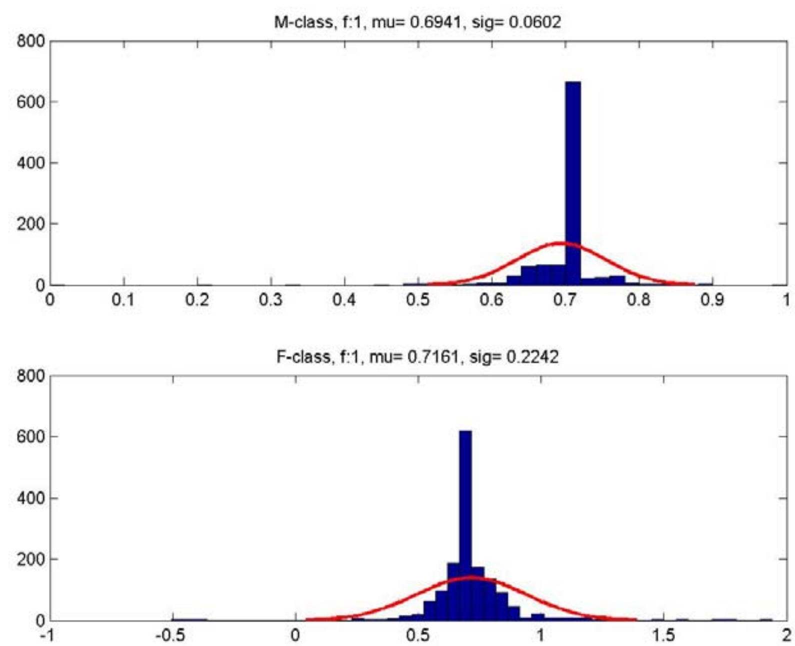

Figure B-1 Histogram plot for t1w20s4, Feature: 1

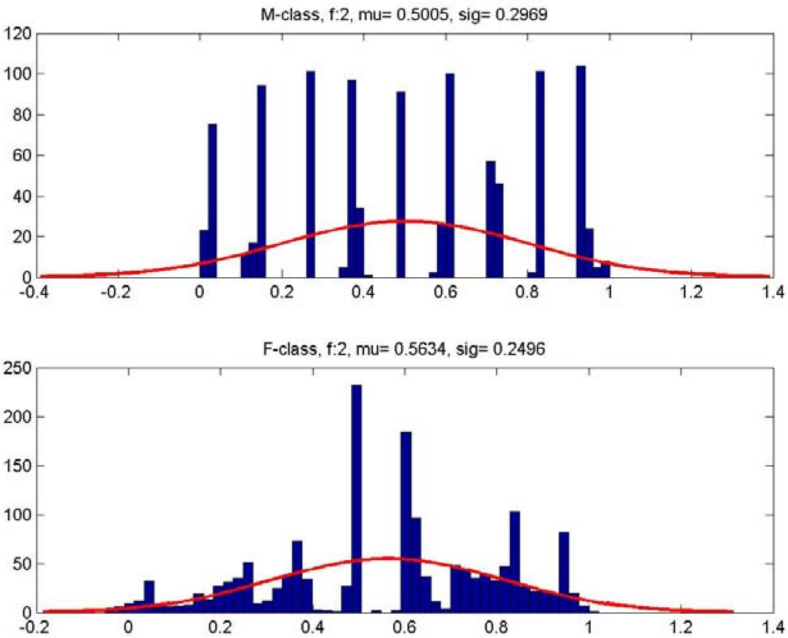

Figure B-2 Histogram plot for t1w20s4, Feature: 2

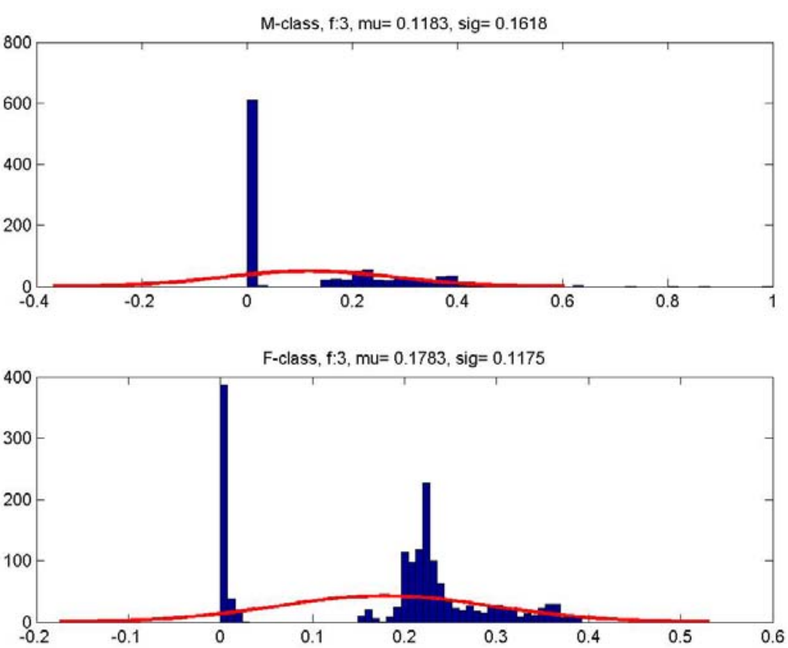

Figure B-3 Histogram plot for t1w20s4, Feature: 3

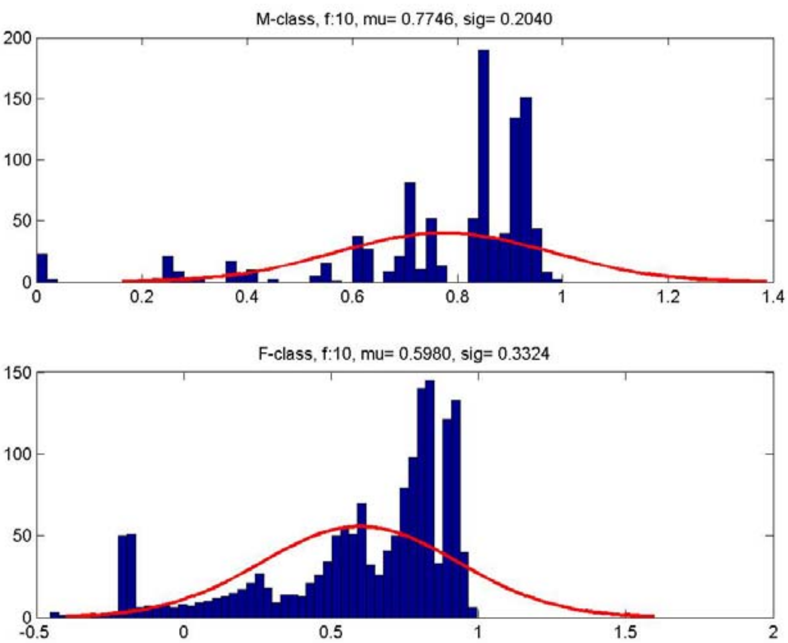

Figure B-4 Histogram plot for t1w20s4, Feature: 10

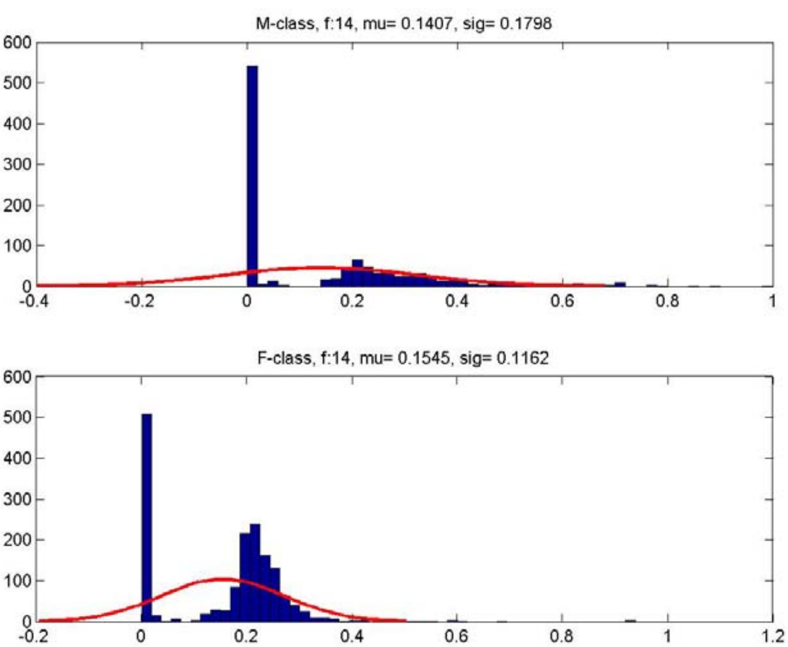

Figure B-5 Histogram plot for t1w20s4, Feature: 14

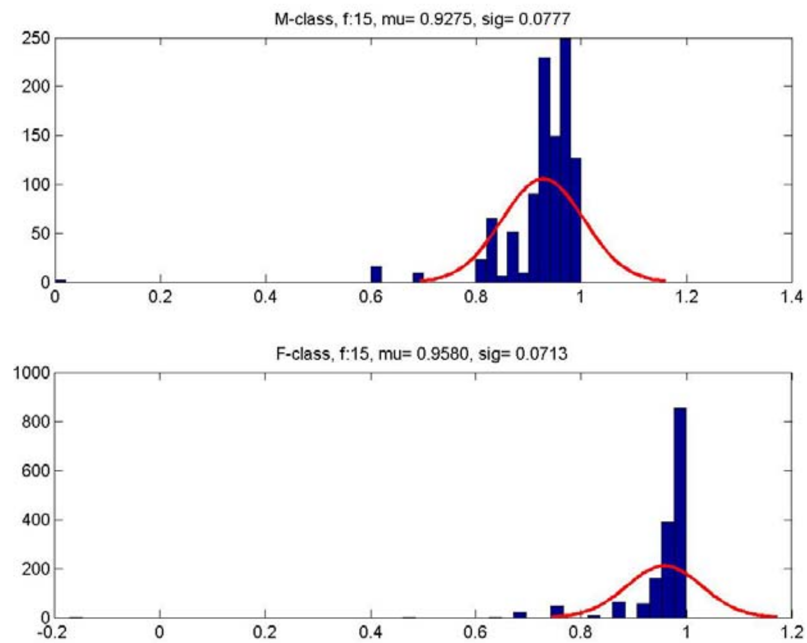

Figure B-6 Histogram plot for t1w20s4, Feature: 15

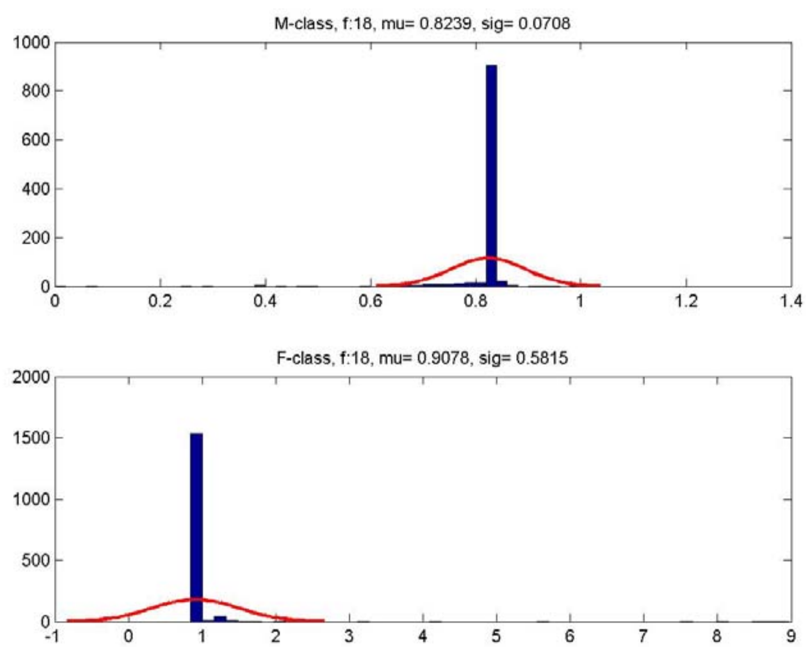

Figure B-7 Histogram plot for t1w20s4, Feature: 18

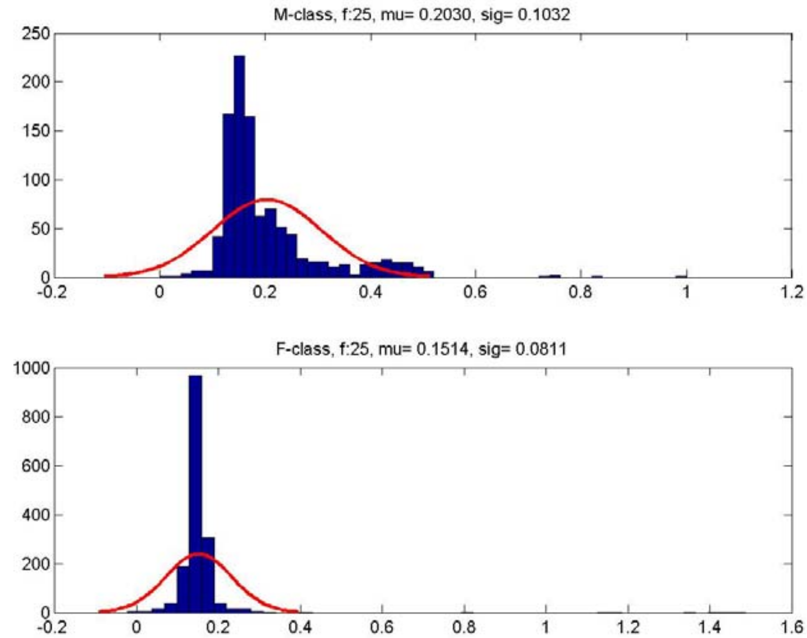

Figure B-8 Histogram plot for t1w20s4, Feature: 25

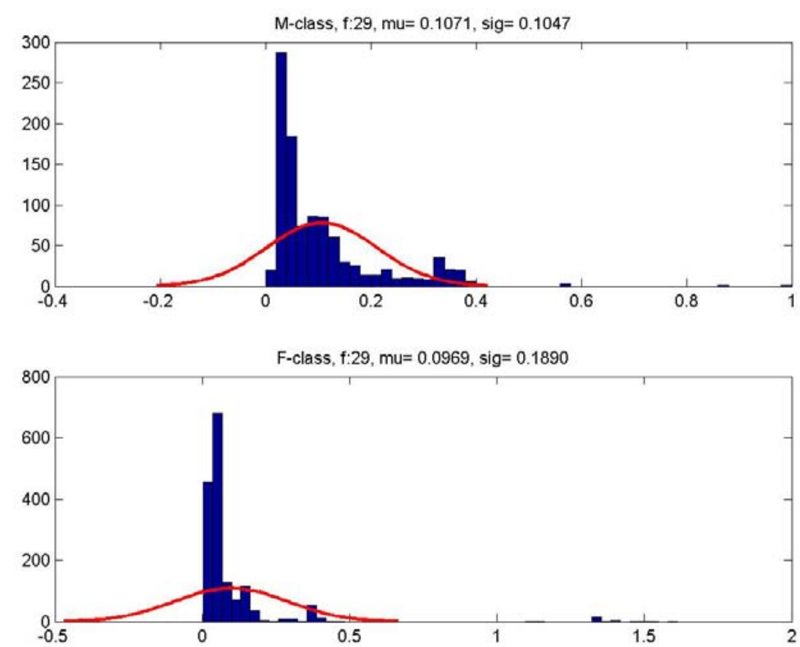

Figure B-9 Histogram plot for t1w20s4, Feature: 29

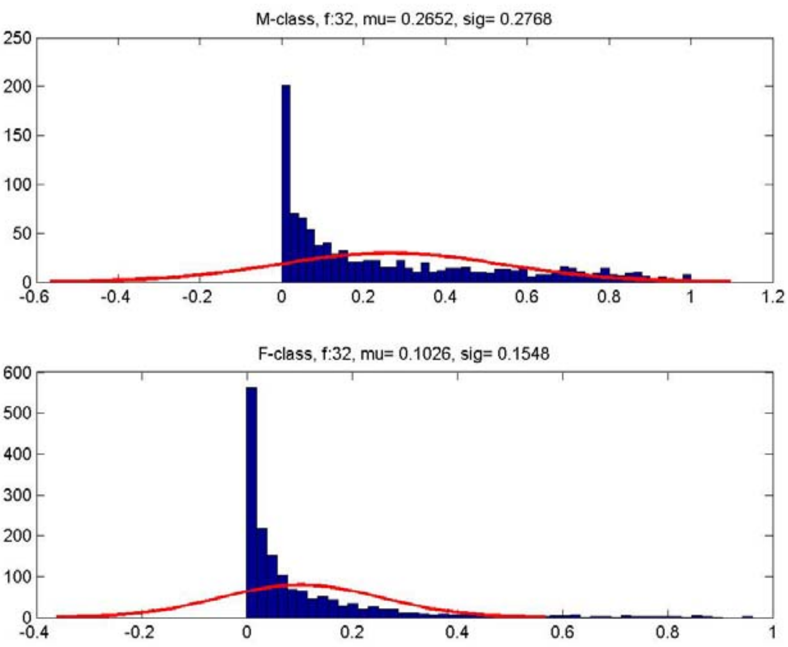

Figure B-10 Histogram plot for t1w20s4, Feature: 32

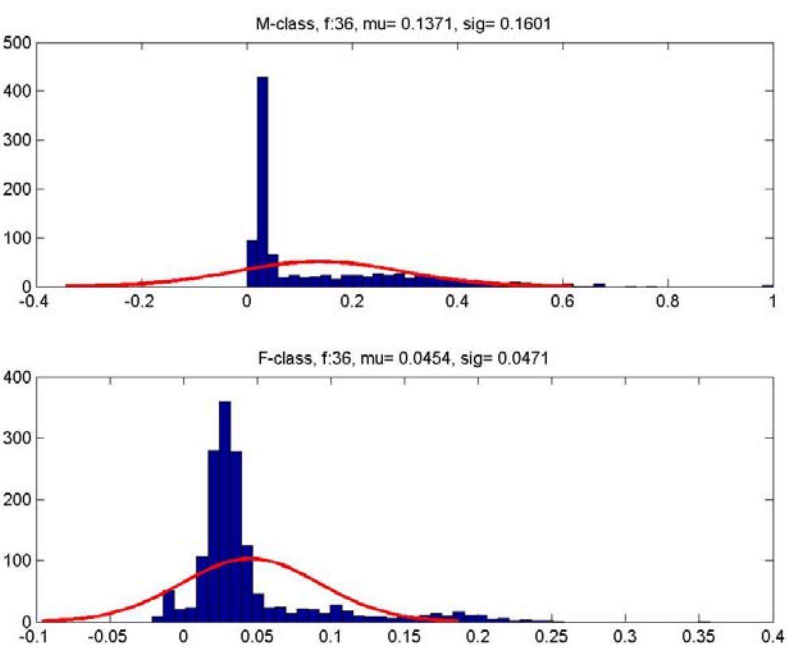

Figure B-11 Histogram plot for t1w20s4, Feature: 36

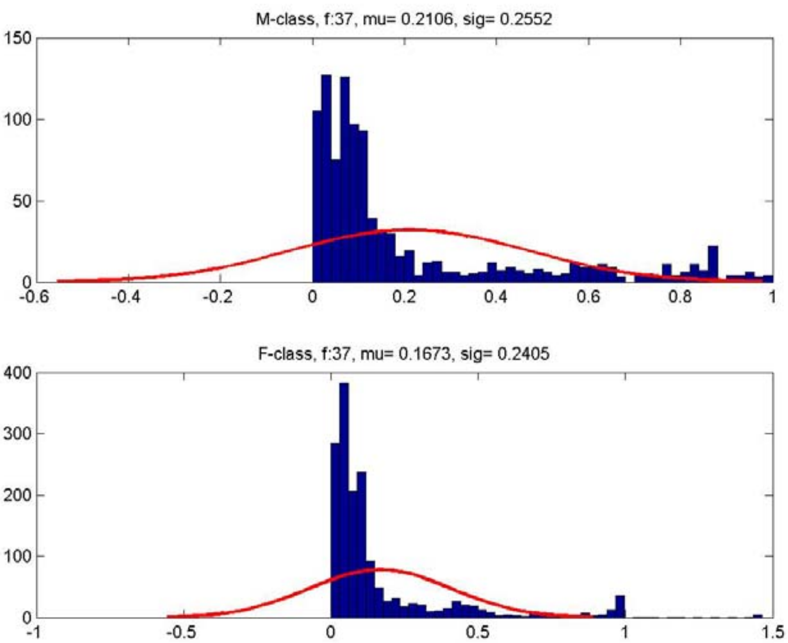

Figure B-12 Histogram plot for t1w20s4, Feature: 37

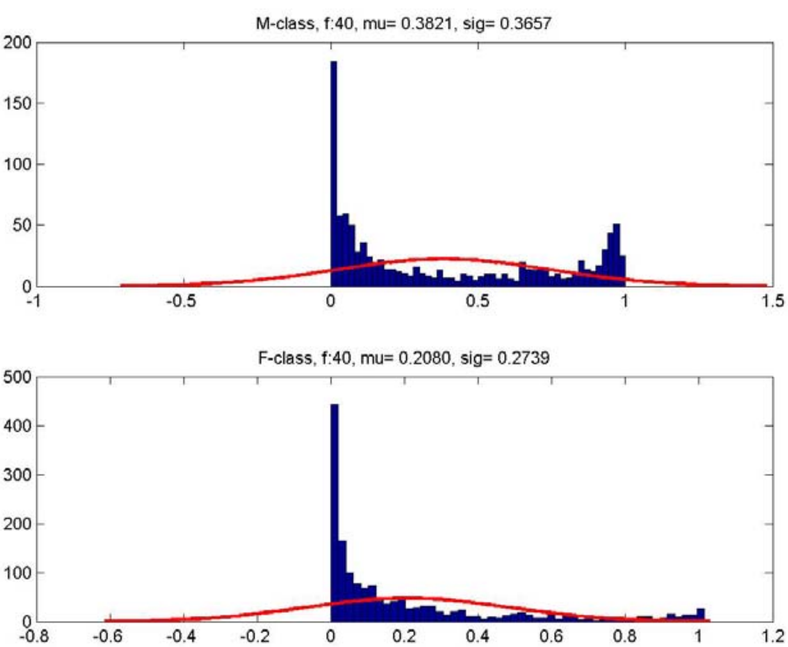

Figure B-13 Histogram plot for t1w20s4, Feature: 40