INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.
FEATURE STUDY FOR HIGH-RANGE-RESOLUTION BASED AUTOMATIC TARGET RECOGNITION: ANALYSIS AND EXTRACTION

DISSERTATION

Presented in Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy in the Graduate School of The Ohio State University

By

Junshui Ma, M.S.

* * * * *

The Ohio State University

2001

Dissertation Committee:
Professor Stanley C. Ahalt, Adviser
Professor Randolph L. Moses
Professor Fred Garber

Approved by

Stanley C. Ahalt
Adviser
Department of Electrical Engineering
ABSTRACT

Techniques for automated moving target/object recognition are required in many military and civilian applications. Recently, Synthetic Aperture Radar (SAR) has been investigated as an attractive sensor for Automatic Target Recognition (ATR) due to its ability to produce moderately high-resolution images under a range of weather conditions. However, the performance of Synthetic Aperture Radar (SAR) degrades considerably when applied to moving target scenarios. A relatively new radar technique named High Range Resolution (HRR) radar appears to be an attractive alternative. This investigation focuses on moving target ATR using HRR radar, with a special emphasis on classification features. The motivation is that a set of robust, high-performance HRR features may have a significant effect on the success of an HRR-based ATR system. This dissertation specifically studies feature definition and extraction, motivated by two basic questions: (1) Do complex HRR signatures contain more meaningful target information than magnitude-only HRR signatures with respect to improving classification performance? (2) What HRR features are potentially robust and of high-quality — and if there are such features, how can they be extracted?

After briefly discussing the basics of HRR radar sensors, the objectives of this investigation are formally defined in Chapter 1. In Chapter 2, an extensive review of previous research into HRR ATR is provided. In Chapter 3 we investigate the utility of
complex HRR signatures based on both theoretical analysis and experimental examinations. Chapters 4 and 5 are devoted to finding and extracting robust HRR signatures. In Chapter 4 we derive a physics-based HRR moving target model, and define the parameters of the model as a set of potential features. Additionally, two parameter estimation algorithms based on this model are developed. Subsequently these algorithms can serve as the feature-extraction algorithms for the HRR data described by our proposed model. However, the features defined and extracted in Chapter 4 are representational features, instead of classification-oriented features, and we thus cannot guarantee that they represent distinguishing information between targets. Thus, Chapter 5 approaches the feature extraction problem differently, and a new nonlinear feature extraction algorithm, named Kernel-based nonlinear Feature Extraction (KFE) algorithm, is proposed. This algorithm extends conventional linear scatter matrix-based feature extraction algorithms to the nonlinear domain via a technique referred to as the "kernel trick", which was first proposed in Support Vector Machines (SVMs), an emerging technique in pattern recognition. Both theoretical proofs and experimental tests demonstrate that high performance features can be defined and extracted using this KFE algorithm. Because SVMs are extensively involved in Chapter 5, a brief introduction to SVMs is provided in Appendix B to ensure that this dissertation self-contained.

The main contributions of this research are (1) a deeper understanding of the properties of complex HRR signatures, especially the utility of their phase information for classification, (2) a set of physics-based HRR moving target models, with different degrees of complexity, (3) a set of physical feature extraction algorithms based on our proposed
HRR models, and, (4) a new category of nonlinear algorithms which can be used to extract high performance classification-oriented features.
To my mother

獻給我的母親
ACKNOWLEDGMENTS

I will always be indebted to my adviser, Dr. Stanley C. Ahalt. He helped me rejoin my wife by accepting me as his Ph. D. student; he financially supported me during my four years of doctoral study; and he helped me to determine my research topics. He provided me with constant and inspiring guidance in all aspects of my research. He even assisted me in improving my writing skills by refining the language of almost all of my papers - sentence by sentence. In a word, this dissertation is because of Professor Stanley Ahalt.

I would also like to express my gratefulness to Professor Randolph Moses, Professor Fred Garber, and Professor Ashok Krishnamurthy. Their insightful comments and questions regarding my research directed me to make some very important adjustments that dramatically strengthened this dissertation.

I also wish to thank all the faculty members and students in the EPS lab. They created an academic atmosphere that I would like to keep in my mind for all my life.

Most part of my doctoral research was supported by a contract from Mission Research Cooperation. Here, I am grateful to Dr. Byron Welsh, Dr. Robert Hawley, and Dr. Micheal Clark. Their suggestions, data, and funding made some of the research results in this dissertation possible.

At last, I would like to thank my family, especially my wife.
VITA

1994 ................................................................. B.S. on Computer and Communications, Xidian University, Xi'an, China

1997 ................................................................. M.S. on Communication and Electrical Systems, Jiaotong University, Xi'an, China

PUBLICATIONS

Submitted Journal Papers:


Conference Papers:


FIELDS OF STUDY

Major Field: Electrical Engineering

Studies in:

(1) Communications and Signal Processing

(2) Computer Engineering

(3) Statistics
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>ii</td>
</tr>
<tr>
<td>Dedication</td>
<td>v</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>vi</td>
</tr>
<tr>
<td>Vita</td>
<td>vii</td>
</tr>
<tr>
<td>List of Tables</td>
<td>xiii</td>
</tr>
<tr>
<td>List of Figures</td>
<td>xiv</td>
</tr>
<tr>
<td>Chapters:</td>
<td></td>
</tr>
<tr>
<td>1. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Ground Moving Target Automatic Recognition Using High Range</td>
<td>2</td>
</tr>
<tr>
<td>Resolution Radar</td>
<td></td>
</tr>
<tr>
<td>1.1.1 HRR Radar: A Promising Technique for Automatic Ground Moving</td>
<td>3</td>
</tr>
<tr>
<td>Target Identification</td>
<td></td>
</tr>
<tr>
<td>1.1.2 Achieving High Range Resolution by Increased Bandwidth</td>
<td>6</td>
</tr>
<tr>
<td>1.1.3 Operational Scenarios Where HRR Radars Are Involved</td>
<td>8</td>
</tr>
<tr>
<td>1.1.4 HRR Radar Data Representation and Imaging Formation</td>
<td>10</td>
</tr>
<tr>
<td>1.2 Research Objectives</td>
<td>13</td>
</tr>
<tr>
<td>1.3 Contributions</td>
<td>16</td>
</tr>
<tr>
<td>2. A Review of Research Into HRR-Based ATR</td>
<td>18</td>
</tr>
<tr>
<td>2.1 General Studies In The Field of HRR-Based ATR</td>
<td>19</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>2.1.1 Fundamental Studies</td>
<td>19</td>
</tr>
<tr>
<td>2.1.1.1 Fundamental Studies Based on Measured HRR Data</td>
<td>20</td>
</tr>
<tr>
<td>2.1.1.2 Fundamental Studies Based on Physical Knowledge of HRR Data</td>
<td>24</td>
</tr>
<tr>
<td>2.1.2 Algorithm Formulation</td>
<td>27</td>
</tr>
<tr>
<td>2.1.2.1 Clutter (Jamming) Suppression Algorithms and Feature Extraction Algorithms</td>
<td>27</td>
</tr>
<tr>
<td>2.1.2.2 Classification Algorithms</td>
<td>29</td>
</tr>
<tr>
<td>2.1.2.3 Evaluation Algorithms</td>
<td>30</td>
</tr>
<tr>
<td>2.1.3 System Integration</td>
<td>32</td>
</tr>
<tr>
<td>2.2 HRR Data Modeling And Model-Based Feature Extraction Algorithms</td>
<td>33</td>
</tr>
<tr>
<td>2.2.1 Scatterer-Based Modeling And Feature Extraction For Stationary Targets</td>
<td>34</td>
</tr>
<tr>
<td>2.2.2 Scatterer-Based Modeling And Feature Extraction For Moving Targets</td>
<td>39</td>
</tr>
<tr>
<td>2.2.3 Random Model Of HRR Data</td>
<td>42</td>
</tr>
<tr>
<td>3. HRR Signature Studies: Complex Vs. Magnitude-Only</td>
<td>44</td>
</tr>
<tr>
<td>3.1 Analysis Of Complex HRR Signatures</td>
<td>44</td>
</tr>
<tr>
<td>3.1.1 Modeling</td>
<td>45</td>
</tr>
<tr>
<td>3.1.2 Simulated HRR Signature Analysis</td>
<td>47</td>
</tr>
<tr>
<td>3.1.2.1 HRR Signatures</td>
<td>47</td>
</tr>
<tr>
<td>3.1.2.2 HRR Signatures Containing Only A Single Scatterer</td>
<td>47</td>
</tr>
<tr>
<td>3.1.2.3 HRR Signature Containing Multiple Scatterers</td>
<td>49</td>
</tr>
<tr>
<td>3.1.2.4 Effects Of Aspect Angle Variance</td>
<td>52</td>
</tr>
<tr>
<td>3.1.3 Analysis Conclusions</td>
<td>54</td>
</tr>
<tr>
<td>3.2 Phase-Utility Experimental Results</td>
<td>56</td>
</tr>
<tr>
<td>3.2.1 Experiment 1</td>
<td>57</td>
</tr>
<tr>
<td>3.2.1.1 Experimental Description</td>
<td>57</td>
</tr>
<tr>
<td>3.2.1.2 Experimental Results</td>
<td>58</td>
</tr>
<tr>
<td>3.2.2 Experiment 2</td>
<td>61</td>
</tr>
<tr>
<td>3.2.2.1 Experimental Description</td>
<td>61</td>
</tr>
<tr>
<td>3.2.2.2 Experiment Results</td>
<td>61</td>
</tr>
<tr>
<td>3.2.3 Experiment 3</td>
<td>64</td>
</tr>
<tr>
<td>3.2.3.1 Experimental Description</td>
<td>64</td>
</tr>
<tr>
<td>3.2.3.2 Experiment Results</td>
<td>66</td>
</tr>
<tr>
<td>3.2.4 Experimental Summary</td>
<td>66</td>
</tr>
<tr>
<td>4. HRR Data Modeling And Feature Extraction Algorithms</td>
<td>68</td>
</tr>
<tr>
<td>4.1 Derivation Of A Physics-Based HRR Moving Target Model</td>
<td>71</td>
</tr>
<tr>
<td>4.2 Parameters Of The Radar System And The Target</td>
<td>77</td>
</tr>
<tr>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>4.3 Model Simplification</td>
<td>79</td>
</tr>
<tr>
<td>4.4 Matrix Form Of The HRR Data Model And Its 2D Auto-Correlation</td>
<td>83</td>
</tr>
<tr>
<td>Matrix</td>
<td>85</td>
</tr>
<tr>
<td>4.5 Parameter Estimation</td>
<td>86</td>
</tr>
<tr>
<td>4.5.1 Scatterer Extraction Using 1D Parameter Estimation (1DPE)</td>
<td>86</td>
</tr>
<tr>
<td>4.5.1.1 1D Scatterer Detection And Parameter Estimation</td>
<td>89</td>
</tr>
<tr>
<td>4.5.1.2 Combination Of The Two 1D Estimation Results</td>
<td>90</td>
</tr>
<tr>
<td>4.5.2 Scatterer Extraction Using 2D Discrete Fourier Transform (2DFT)</td>
<td>94</td>
</tr>
<tr>
<td>4.6 Simulation Experiment And Performance Comparison</td>
<td>95</td>
</tr>
<tr>
<td>4.7 Some Comments</td>
<td>100</td>
</tr>
<tr>
<td>5. Kernel-Based Feature Extraction</td>
<td>102</td>
</tr>
<tr>
<td>5.1 On Classification-Oriented Feature Extraction</td>
<td>104</td>
</tr>
<tr>
<td>5.2 Scatter Matrix-Based Linear Feature Extraction</td>
<td>109</td>
</tr>
<tr>
<td>5.3 Kernel-Based Feature Extraction</td>
<td>115</td>
</tr>
<tr>
<td>5.4 Modified Kernel-Based Feature Extraction</td>
<td>125</td>
</tr>
<tr>
<td>5.5 Experiments</td>
<td>128</td>
</tr>
<tr>
<td>5.5.1 Experiments Based On Synthetic Data</td>
<td>129</td>
</tr>
<tr>
<td>5.5.2 Experiments Based On Measured HRR Signatures</td>
<td>140</td>
</tr>
<tr>
<td>5.6 Conclusions</td>
<td>146</td>
</tr>
<tr>
<td>6. Conclusions</td>
<td>148</td>
</tr>
<tr>
<td>Appendices</td>
<td></td>
</tr>
<tr>
<td>A. Feature Extraction Processing In Phase-Utility Experiments</td>
<td>152</td>
</tr>
<tr>
<td>B. Introduction To Support Vector Machines</td>
<td>154</td>
</tr>
<tr>
<td>B.1 Historic Background</td>
<td>154</td>
</tr>
<tr>
<td>B.2 Learning From Data: SVMs Basics</td>
<td>155</td>
</tr>
<tr>
<td>B.3 Different Formulations Of SVMs</td>
<td>157</td>
</tr>
<tr>
<td>B.3.1 Linear SVMs For Linearly Separable Data</td>
<td>158</td>
</tr>
<tr>
<td>B.3.2 Linear SVMs For Linearly Inseparable Data</td>
<td>161</td>
</tr>
<tr>
<td>B.3.3 Nonlinear SVMs</td>
<td>163</td>
</tr>
<tr>
<td>B.3.4 Modified Formulations Of SVMs</td>
<td>166</td>
</tr>
<tr>
<td>B.4 Attractive Properties Of SVMs</td>
<td>167</td>
</tr>
</tbody>
</table>
Bibliography
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Correspondence between the value of $\alpha$ and the geometry of canonical scatterers</td>
<td>36</td>
</tr>
<tr>
<td>3.1</td>
<td>Summary of phase-utility experiments</td>
<td>66</td>
</tr>
<tr>
<td>4.1</td>
<td>Simplify $\overline{\omega}_k$</td>
<td>79</td>
</tr>
<tr>
<td>4.2</td>
<td>Simplify $\overline{\omega}_{2k}$</td>
<td>80</td>
</tr>
<tr>
<td>4.3</td>
<td>Simplify $\overline{\omega}_y$</td>
<td>80</td>
</tr>
<tr>
<td>4.4</td>
<td>Simplify $\overline{\omega}_{2y}$</td>
<td>80</td>
</tr>
<tr>
<td>4.5</td>
<td>Simplify $\overline{\omega}_{20k}$</td>
<td>81</td>
</tr>
<tr>
<td>4.6</td>
<td>Configuration of 6 experimental scatterers</td>
<td>96</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Time-frequency property of the chirp signal</td>
<td>7</td>
</tr>
<tr>
<td>1.2</td>
<td>HRR signatures of T72 Tank within a 10° aspect range, which centers on 75°</td>
<td>15</td>
</tr>
<tr>
<td>1.3</td>
<td>Roadmap of my doctoral research</td>
<td>17</td>
</tr>
<tr>
<td>3.1</td>
<td>Experimental design of Experiment 1</td>
<td>59</td>
</tr>
<tr>
<td>3.2</td>
<td>Experiment 1 results: obtained when k=3</td>
<td>60</td>
</tr>
<tr>
<td>3.3</td>
<td>Experimental Design of Experiment 2</td>
<td>62</td>
</tr>
<tr>
<td>3.4</td>
<td>Experiment 2 results</td>
<td>63</td>
</tr>
<tr>
<td>3.5</td>
<td>Experimental Design of Experiment 3</td>
<td>64</td>
</tr>
<tr>
<td>3.6</td>
<td>Experiment 3 results: obtained when k=3</td>
<td>65</td>
</tr>
<tr>
<td>4.1</td>
<td>Coordinate System</td>
<td>74</td>
</tr>
<tr>
<td>4.2</td>
<td>Implementation of IDPE algorithm</td>
<td>94</td>
</tr>
<tr>
<td>4.3</td>
<td>Distribution of the wrong estimations of K</td>
<td>97</td>
</tr>
<tr>
<td>4.4</td>
<td>Probability of estimation error</td>
<td>98</td>
</tr>
</tbody>
</table>
B.4 Conceptual steps implementing a nonlinear SVM............................................. 164
CHAPTER 1

INTRODUCTION

Automatic Target Recognition (ATR) has been an intensively studied topic in the past decade [6, 9, 36, 39, 44, 65, 72, 75, 79]. Many advances, including new sensor techniques, novel information processing algorithms, and original pattern recognition methods, have occurred due to these intense research efforts. Among the numerous remote sensors employed in ATR, radar technology continues to attract significant attention due to its attractive features, and because advanced hardware has been developed over the past 60 years. Indeed, the past 3 decades have witnessed a number of breakthroughs in radar technology, among which are Synthetic Aperture Radar (SAR) and High Range Resolution (HRR) radar [48, 49, 50].

Although SAR has demonstrated outstanding performance in stationary target identification [36,39], its resolution suffers in moving target scenarios. In contrast, HRR radar appears to be an attractive alternative for moving target identification. This is because HRR radar requires less dwell time, and because measured HRR target signatures can provide target information with relatively high range resolution.

This doctoral research is devoted to the study of moving target ATR using HRR radar, with a special emphasis on the study of the features used in HRR-based ATR. In order to
make our research goals more concrete, we pose several fundamental questions regarding feature studies, and then consider the answers to these questions as our research objectives.

In this chapter we define these questions, i.e., our research objectives. However, before delving into the details, we first provide some background knowledge on HRR-based radar. In Section 1.1 we provide the basics of HRR Radar with a special emphasis on its application to Automatic Ground Moving Target Identification (AGMTI). Hopefully this section will motivate readers' interests in this topic, and provide readers with the terminology used in the remainder of the dissertation. In Section 1.2 we formulate our research objectives. Finally, the contributions of this doctoral research are listed in Section 1.3.

1.1 Ground Moving Target Automatic Recognition Using High Range Resolution Radar

As noted above, although SAR has demonstrated exceptional performance in stationary target identification [36,39], due to the physical characteristics of the sensor, SAR images can be smeared in moving target scenarios, which dramatically degrades ATR performance. Fortunately, another radar technique, HRR radar, appears to be a promising alternative in moving target identification problems because HRR radar can provide target information with higher range resolution than conventional radar, and requires less dwell time to form HRR data than SAR.

In Subsection 1.1.1 we detail the advantages of HRR radar over other existing radar techniques when applied to moving target ATR, especially for ground moving target identification. In Subsection 1.1.2 we explain how high range resolution can be achieved
by HRR radar, and in Subsection 1.1.3 we describe HRR operational modes to provide readers with an operational context regarding the application of HRR radar. Finally, in Subsection 1.1.4, we present the signal representation of HRR radar data, as well as the processes involved in the generation of HRR signatures.

1.1.1 HRR Radar: A Promising Technique for Automatic Ground Moving Target Identification

Modern battlefields are highly dynamic and many of the ground-based vehicles involved in a battlefield campaign can be expected to be very mobile. Traditional target identification techniques become less reliable in this scenario. However, by identifying motion patterns and associating these patterns with any available identification information, we may be able to exploit object movement. However, given the complexity of these tasks and the sheer number of possible targets or objects which might need to be evaluated, it seems reasonable to assume that these types of tasks are most likely to be successful if they are performed automatically by machine. Furthermore, modern reconnaissance and surveillance platforms typically collect so much information in any one mission that it has become almost impossible to process the volumes of collected data without the help of automated prescreening techniques. Finally, in other situations the capacity to automatically track and identify ground moving targets becomes indispensable, for example targets such as the rocket launchers are primarily vulnerable when they travel from one place to another. Another example includes targets that constitute an increased threat while moving, such as tanks – or other moving vehicles, such as ambulances, that must be identified for humanitarian reasons.
All of these factors combine to make Automatic Ground Moving Target Identification (AGMTI) techniques increasingly critical in contemporary military applications. Additionally, AGMTI may find immediate application in civilian applications such as airport ground traffic control, event security operations, etc.

Although ATR has become an intensively studied in the past decade, techniques for Automatic Moving Target Identification (AMTI) are still far from established, and the AGMTI is proving to be especially challenging. This is because (1) ground target identification using active remote sensors, such as radars, is plagued by ground clutter which is introduced by the undesired background reflectors around the desired ground targets, and (2) ground targets are easily hidden in many military scenarios.

Conventionally, a radar technique named Moving Target Indicator (MTI) is employed for moving target detection, location, and tracking [1, 2, 27, 85]. This class of radars has continuously adopted new techniques, such as Space-Time Adaptive Processing (STAP) [29], and is installed in some contemporary military platforms such as the Global Hawk. However, it is generally understood that these systems are mainly designed for surveillance and are not well suited for Ground Moving Target Identification (GMTI) applications, because of limited range resolution.

One breakthrough in radar techniques has been the invention of SAR [48, 49, 50]. SAR radar can image targets with both very high range and cross-range resolution. Therefore, many ATR studies and systems are currently based on SAR [36, 39], and experiments have demonstrated that SAR is a very promising technique in stationary ground target
identification. However, SAR-based ATRs have a few inherent disadvantages when applied to AGMTI. First of all, SAR requires long dwell times for image-formation. Consequently, the motions of targets and the physics of the sensors result in blurred images, which dramatically degrade the subsequent identification performance. Moreover, intensive computation is required to form and compensate SAR images, and this may preclude the use of SAR in many real-time field applications.

Inverse synthetic aperture radar (ISAR), another new 2D radar imaging technique, is a special version of SAR. Compared with SAR imagery, which is formed with data collected as the radar platform moves around the targets, ISAR creates its imagery by making use of the target motions in the radar beam [48]. Therefore, it can be utilized operationally for moving ship, aircraft, and space object identification. However, it is also becomes problematic when applied to AGMTI. First, the clutter introduced by the interaction between moving targets and their surroundings significantly degrades the imagery. Additionally, it also suffers from the disadvantages commonly faced by SAR, such as the long dwell time for imagery formation, complex imaging algorithms, etc.

HRR radar, or radar operated in the HRR mode, has been proposed in this context as a promising solution for AGMTI because (1) HRR radar can provide more target information with much higher range resolution than conventional MTI, (2) HRR radar tolerates target motion due to the reduced dwell time required to form HRR radar data, and (3) HRR radar imagery formation is computational tractable.
1.1.2 Achieving High Range Resolution by Increased Bandwidth

Conventional radar theory nominally argues that range resolution is decided by the radar pulse width. The narrower the pulses are, the higher the range resolution is. However, a shorter pulse implies less energy transmitted per pulse given a finite peak-power, and less energy, in turn, results in a reduction in operational range. Therefore, this traditional range resolution notion forced radar engineers to make a bitter tradeoff between range resolution and operational range.

Fortunately, this problem was overcome with an increased understanding of radar range resolution. Radar researchers eventually determined that it is the radar signal bandwidth, instead of the pulse duration, that ultimately determines the range resolution, that is:

\[
\Delta R = \frac{c}{2B}
\]

where \( \Delta R \) is the range resolution, \( c \) is propagation speed of the radar signal, and \( B \) is bandwidth of the radar signal.

HRR radar is precisely based on this concept. HRR radar transmits specially designed radar waveforms with sufficient bandwidth to achieve high range resolution, and at the same time, with long enough pulse width to ensure adequate transmitted energy for the designed operational range. The property of the radar waveforms therefore plays an important role in HRR radar.

Until now, approximately a half dozen radar waveforms have been studied and employed in HRR radar to meet different application needs [48]. The most widely used waveform in ATR is generally referred to as the chirp signal, or FM signal. Our study focuses solely on the chirp-pulse HRR radar if not otherwise specified.
If we denote the chirp pulse as $\tilde{s}(t)$, it can be expressed as:

$$\tilde{s}(t) = \begin{cases} 
\exp\{j(\omega_0 t + \gamma^2 t^2)\}, & |t| \leq T_0 / 2 \\
0, & |t| > T_0 / 2
\end{cases}$$  \hspace{1cm} (1.2)

where $T_0$ is the pulse width, $\omega_0$ is the center carrier frequency, and $2\gamma$ is the chirp rate of the chirp pulse.

The reason why high range resolution can be achieved by this chirp signal is illustrated in Figure 1.1 by its time-frequency property.

From Figure 1.1, it is clear that suitable bandwidth, $B$, can be achieved by adjusting the chirp rate, $2\gamma$ given a constant pulse width, $T$. However, an extra de-chirp process in the radar receiver is required to remove the linear frequency modulation of the transmitted radar waveform. Since adding a de-chirp process to a chirp-pulse HRR radar is equivalent to sending a very narrow "effective" pulse, this de-chirp process is also referred to as pulse compression [48].
1.1.3. Operational Scenarios Where HRR Radars Are Involved

In order to assist readers in gaining a general idea of the application context of HRR radars, in this sub-section we briefly describe how a general reconnaissance and surveillance mission is orchestrated.

In real world applications it is seldom the case that only a single sensor platform is used for a whole reconnaissance and surveillance mission. In most cases, multiple radar platforms, such as aircrafts and satellites, work in concert to ensure birth-to-death tracking of targets, to estimate the target size and pose, and to achieve moving target identification. In addition, the radars in these platforms can operate in more than one operational mode, such as GMTI mode, SAR mode, and HRR mode. Different operational modes are used to achieve different application goals by employing different ground scanning patterns, different radar waveforms, and different transmission-receiving methods. Further, all these platforms are coordinated by platform managers in a ground station to capitalize on their cooperation. Meanwhile, the data obtained from all of these radar platforms is transmitted directly, or relayed via satellite, to the ground station for further processing and/or for display as feedback to the platform managers.

Specifically, when a radar platform is sent on a mission to fulfill part of a reconnaissance-surveillance mission, an operator in the ground station directs the platform to collect specific information regarding interesting targets and/or regions. Accordingly, a sensor resource manager in the platform determines the proper radar modes to accomplish the given task. Different radar modes work together to provide all necessary information about the region and targets, including (1) terrain imagery provided by strip-map SAR, (2)
stationary target imagery obtained by using spotlight SAR, (3) moving target indications for tracking-information-updating obtained by GMTI, and (4) moving target identification realized by HRR [1].

For reconnaissance and surveillance in certain regions, region-wide search is generally conducted by imaging the area with the strip-map SAR mode, or by tracking targets with the GMTI mode. When the GMTI mode is employed, moving targets are presented as colored dots to the operators, which provide the tracking systems necessary information to form logical track files of each interesting target, or multiple targets, such as convoys. A typical search pattern for a given region is to split the region into several bars with different ranges. The radar beam sweeps through a range bar in one direction, and turns around to sweep along the next adjacent range bar. It continues scanning one bar after another, and returns to the first one in a raster scan fashion when all of the bars are completed. A single beam position, or radar footprint, represents a fixed area on the ground, and spans a constant number of cross-range degrees, for example 2 degrees for Global Hawk [1]. Therefore, the total number of beam positions is jointly decided by the size of the beam and the size of the search region.

The GMTI radar mode is suitable for wide-area searching and tracking because it has a large beam footprint and can cover a large area at the same time. However, to accomplish the identification, HRR radar is required because this mode employs a chirp waveform to achieve high range resolution, through which detailed target information can be obtained. The cost paid to use the HRR mode is that the visible range size of the footprint under the HRR mode is much smaller than that used under the GMTI mode. This is because much
more data is sampled per unit range distance in HRR mode, and thus, the total visible range is shorter, given that the radar system has a fixed data buffer.

1.1.4 HRR Radar Data Representation and Imaging Formation

Our study is focused on classification features obtained from HRR data; we are therefore only interested in target information stored in electronic form. Usually, this information is referred to as signals by researchers in the signal-processing arena. Here we discuss how the HRR radar returns are processed and how HRR radar data is generated. However, before describing these processes, we first introduce the notation we will use to represent HRR radar signals.

Although HRR requires a large absolute bandwidth to achieve high range resolution, it is possible that the fractional bandwidth of the HRR signals, which is defined as the ratio of signal bandwidth to its center frequency, is quite narrow. Therefore, it is still reasonable for us to represent the HRR signals via a narrowband approximation. It is known that any narrowband signal, \( s(t) \), can be represented as:

\[
\begin{align*}
    s_r(t) &= a(t) \cos[\omega_0 t + \theta(t)] = \text{Re}\{a(t) \exp[\omega_0 t + \theta(t)]\} \\
    \text{Re}\{} & \text{represents the operation of obtaining the real part of the complex signal.}
\end{align*}
\]

where \( a(t) \) is the amplitude modulation; \( \theta(t) \) is the phase modulation around the carrier frequency \( \omega_0 \). \( \text{Re}\{} \) represents the operation of obtaining the real part of the complex signal. Therefore, the complex representation of the real narrowband signal is naturally expressed as:

\[
    s(t) = a(t) \exp[\omega_0 t + \theta(t)].
\]

Because this complex exponential form is more simplified, we will adopt it in our subsequent analysis and discussion.
HRR radar transmits a waveform, or signal, \( s_i(t) \), which is a burst of sequential radar pulses, and decodes the target information from the received signal, \( s_{bk}(t) \).

\[
s_i(t) = \sum_{l=1}^{L} \tilde{s}(t - lT),
\]
\[
s_{bk}(t) = \sum_{l=1}^{L} \tilde{b}(t - lT)
\]

where \( \tilde{s}(t) \) is a transmitted chirp pulse, \( \tilde{b}(t) \) is a received pulse corresponding to \( \tilde{s}(t) \), \( T \) is the time interval between two adjacent pulses, and \( L \) is the number of pulses sent each time.

For each transmitted radar pulse, \( \tilde{s}(t) \), one return pulse, \( \tilde{b}(t) \), is obtained, and this pulse contains the scattering information of the illuminated region. This single return has a very low Signal-to-Clutter Ratio (SCR), and cannot be directly used for any ATR application. A clutter suppression process thus becomes compulsory for HRR-based ATR. The Doppler effect introduced by target motion plays a critical role in this process.

In the HRR mode, the radar transmits a sequence of radar pulses in a very short time period at each beam position. More precisely, because the radar platform moves constantly, the beam footprint of each pulse also moves continuously on the ground. However, compared with the width of the beam footprint on the ground, this cross-range movement is very small. Therefore, this sequence of continuous beam footprints can be approximated by one beam position. Moreover, because the cross-range movement is very small during the time interval between two transmitted pulses, the Doppler effect introduced by the platform cross-range movement is generally negligible. As a result, only the target motion in the radial direction can produce a significant Doppler effect in the radar returns. This explains why the HRR data can convey the target motion information.
As previously mentioned, it is this Doppler effect that makes the data collected in the HRR mode usable. In an AGMTI application the backscattered pulse from each transmitted pulse accumulates all clutter returns with target returns at the same range bin. Therefore, for stationary targets, it is very challenging to obtain useful target information due to the extremely low SCR caused by heavy ground clutter. It is, however, possible to extract moving target information by making use of those Doppler frequencies of the moving targets. Conventional clutter suppression processing involves forming 2D range-Doppler images by coherent processing over a sequence of HRR radar returns. In the images obtained in this fashion, the stationary clutter is concentrated within the low Doppler frequency range, while the targets, with relative high radial speeds, are in the (relatively) high Doppler frequency range. Consequently, removal of the clutter becomes quite straightforward, and is conceptually equivalent to applying a high-pass filter to the range-Doppler images. [1]

To make this clearer, we employ some previously defined signal notations to state this processing more formally. If the time interval between two pulses, $T$, is small enough\(^1\) to track the Doppler frequencies introduced by target radial motions [51], we can line up all the radar pulse returns, $b(t - lT), l = 1,...,L$, to form an array of 1D HRR returns, and its discrete version is a 2D HRR Phase History Response (PHR) matrix, $D(n,l)$. The range-Doppler image of the illuminated ground area can be conventionally obtained from this 2D PHR data, $D(n,l)$ by applying a 2D FFT. In this proposal, we refer to this $D(n,l)$ as the 2D

\(^1\) $T$ should satisfy the Nyquist sampling theorem for a given maximum target speed [51].
HRR data, and some HRR-based ATR research have been based on this 2D HRR data [2, 5, 37, 46, 51].

This 2D $D(n,l)$ can also be further processed to obtain 1D HRR target signatures, or HRR target profiles. The conventional process to obtain HRR signatures can be briefly described as follows:

- Clutter can be removed by utilizing the Doppler effect, as noted earlier.
- If we assume that the targets appear as clusters of energy concentrated in small regions in the range-Doppler image, we can gate a small region out to form a range-Doppler chip for a particular target.
- A 1D HRR signature for any particular target is obtained by coherently integrating the range-Doppler chip along the Doppler direction.

This HRR signature can be treated as a 1D signal, and is assumed to contain all the important scattering information of the interesting targets. There are also a number of studies in the literature that are based on 1D signatures [7, 28, 35, 44, 45, 64, 65, 66, 72].

1.2 Research Objectives

A practical ATR system usually includes a set of sub-systems, such as pre-processors, classifiers, and sometimes, evaluators. The pre-processors are generally used for such purposes as clutter suppression, target pre-screen, noise reduction, and feature extraction. The classifiers are employed to identify targets given the processed data or given features produced by the pre-processors. Sometimes, a set of ATR performance evaluation processes are used to provide quantitative assessment of the classification results [89].
Regarding the general requirements of an ATR system, in addition to high classification performance, most of the application scenarios demand real-time operation and simple implementation. The common strategy currently employed is to utilize relatively simple classifiers to meet these implementation and speed requirements, and the classification performance is mainly achieved by using carefully designed pre-processing, especially pre-processing that yields highly separable features. Therefore, feature studies, including feature definition and extraction, become very critical for a successful HRR-based ATR system.

Although HRR data appears to be very promising for moving target ATR applications, especially for AGMTI, its practical application has proved to be a very challenging task. For example, when HRR signatures are employed as ATR data, the extremely irregular behavior of HRR signatures makes the underlying ATR system less robust than desirable. This irregularity is manifest in the way that any small change in aspect angle, in the target configuration, or in the background clutter, can cause dramatic changes in the HRR signatures, which is illustrated in Figure 1.2. As can be notionally see in this figure, even a small change in aspect can result in a very large change in the amplitude of the response in any given range bin.
Figure 1.2 HRR signatures of T72 Tank within a 10° aspect range, which centers on 75°

This is mainly due to an effect referred to as the scintillation effect, and scintillation occurs in such a way that a peak in a given HRR signature can suddenly disappear in an (aspect) adjacent signatures. This is mainly a result of the coherent integration process involved in generating the signatures. This scintillation effect dramatically corrupts the consistency of the HRR signatures and makes it a challenging topic to define a set of robust target features from HRR signatures. This is part of the reason why so many researchers in the field of HRR ATR avoid using extracted features derived from HRR signatures, and instead, design classifiers based directly on the entire HRR signatures [7, 28, 32, 44, 57, 58, 65, 66, 88]. However, this treatment, in addition to introducing more computation due to the high dimensionality of HRR signatures, also cannot guarantee robust classification performance.
Therefore, researchers in the field have repeatedly posed questions regarding the existence of robust feature sets for HRR-based ATR, as well as how they can be extracted. This research is focused on these questions, namely the study of HRR features with application to HRR-based ATR. To put it more specifically, our research objectives are defined as answering the following two groups of questions.

- Do complex HRR signatures contain more meaningful target information than magnitude-only HRR signatures with respect to classification performance?
- What HRR features are potentially robust and of high-quality, and, if such features exist, how can these features be extracted?

The first question is answered in Chapter 3 by employing both theoretical analysis and experimental tests. The second group of questions are approached from different perspectives, and two different sets of solutions are presented in Chapters 4 and 5 respectively. In Chapter 2 we review the related research to help readers understand the relationship between our feature studies and other ongoing work in HRR-based ATR.

1.3. Contributions

This research represents one attempt at studying HRR-based ATR with a clear focus on the classification features required for HRR data. This specialization allows us to study the special characteristics of HRR data, and enables us to refine exiting techniques and develop new techniques in order to accommodate the to meet these characteristics.

The contributions of this research can be grouped into three areas. Figure 1.3 illustrates these areas, as well as their relationships. The first area is feature analysis, which provides a
deeper understanding of the properties of complex HRR signatures such as scintillation, and the utility of phase. However, the simplicity of the HRR model used in this study suggests that our conclusions might represent only a snapshot of the basic properties of HRR signatures. The second area focuses on developing a set of model-based feature extraction algorithms. In this research a physics-based HRR moving target models is devised, and a set of model-based feature extraction algorithms are developed. However, our model is based on 2D HRR raw data, in contrast to conventional HRR signatures. Thus, because we do not have access to measured HRR signatures to verify our model and algorithms, we refrain from drawing too broad a set of conclusions from the results. Finally, in the third area of research, we introduce a new category of nonlinear algorithms capable of extracting high performance discriminant features. This significant contribution is a technique that is both theoretically sound and demonstrably effective in realistic experimental data.
CHAPTER 2

A REVIEW HRR-BASED ATR RESEARCH

HRR-based moving target ATR has been investigated by a number of researchers from different areas, and using different approaches, during the past decade. However this area of research is still very dynamic. Moreover, some techniques that have been developed in other scientific domains appear poised to be transitioned to HRR-based ATR. This chapter reviews these state-of-the-art techniques.

This Chapter consists of two sections. In Section 2.1 we classify the research related to HRR-based ATR into three categories, and we review and summarize the representative results in each category. Hopefully, this section offers readers an overview, as well as a framework, of the studies pertaining to HRR-based ATR. In addition, this general review is meant to provide readers with a clear understanding of the relationship between our proposed feature studies and other ongoing work in HRR-based ATR.

In Section 2.2 we summarize the existing techniques used for radar data modeling and model-based feature extraction. Several physics-based models, as well as their corresponding parameter estimation algorithms, are presented. Our emphasis on this body of research arises because these efforts are closely related to our second set of research objectives and results, which are presented in Chapter 4.
2.1 General Studies in the Field Of HRR-Based ATR

The need for moving target identification has motivated research into HRR-based ATR. Naturally, studies of HRR-based ATR stem from conventional ATR, and are inherently multidiscipline. Knowledge of radar systems, electromagnetic (EM) wave propagation, EM field scattering, signal processing algorithms, and methods of estimation and classification all play important roles in the development of a successful HRR-based ATR system.

In order to review HRR-related research in a more systematic way, we start by classifying the current literature into three categories, which we define as: (1) fundamental studies, (2) algorithm formulation, and (3) system integration. Fundamental studies are summarized in Subsection 2.1.1, algorithm formulation is presented in Subsection 2.1.2, and finally Subsection 2.1.3 contains our review of HRR system integration research. Admittedly, this classification is subjective. Indeed, some research covers more than one category, and, naturally, a strong correlation exists between research themes classified in these three categories.

2.1.1 Fundamental Studies

The research in this category is primarily focused on achieving the following goals:

- to provide a more complete understanding of the properties of HRR data, and
- to study the limitations of the HRR data when applied to ATR, and
- to establish a theoretical basis for developing HRR-based ATR algorithms.

Generally speaking, the behavior of HRR data can be studied from two perspectives. One is to explore the inherent structure of the measured HRR data via different signal-
processing techniques — this type of research is reviewed in Sub-subsection 2.1.1.1. The second perspective assumes a physical model. That is, physics-based models of HRR data are first devised by utilizing knowledge of the radar system, as well as the properties of EM propagation and scattering. In the latter case, subsequent studies of the HRR data properties are mainly based on the proposed models. Sub-subsection 2.1.1.2 reviews the research that falls into this category.

2.1.1.1 Fundamental Studies Based on Measured HRR Data

From a signal processing perspective, HRR signatures can be considered as 1D signals, while the raw HRR radar returns can be considered as 2D signals. Therefore, in order to extract information about the structure of HRR data, researchers have attempted to apply a number of signal processing techniques to HRR data. The research conducted in this area can be further divided into three groups.

(1) HRR data property studies using non-parametric signal processing methods.

The general methodology characterizing this research ensemble is to employ various non-parametric signal-processing algorithms to both analyze HRR data and to determine the inherent properties extant in raw HRR data but manifest in the signal processing.

Measured HRR data was studied in the time domain by Garber et. al. [10], Moffatt et. al.[18], and Dominek [20]. Each of these studies investigated the transient responses of 1D radar profiles, and demonstrated the correspondence between the characteristics of the transient responses and the physical target properties.

Alternatively, Garber et. al [10], Li et. al. [34], and Jouy et. al. [71] investigated HRR radar data in the spectral domain. The spectrum of the radar data was estimated using
various spectral estimation algorithms, and the general observation claimed is that the spectrum of the radar signals manifests target characteristics that are either more representative, more robust, or easier to extract.

Similarly, processing the HRR signatures using wavelet transforms was reported by Zhang et. al. [72, 75] and resulted in a representation of HRR signatures in the time-scale domain. Both of these investigations reported that time-scale domain representation of HRR signatures demonstrate robustness to noise, and a relative insensitivity to aspect angle variance.

Finally, both Kirk [77] and Butterfield [78] have claimed that HRR signatures of both clutter and targets are fractal signals. Therefore, they claim that the fractal dimension may convey distinguishing and representative information about either the clutter or the targets due to the difference between natural clutter and the man-made targets.

Note that no signal models are assumed in this body of research, and therefore only non-parametric signal processing techniques can be applied to the radar data. In contrast, many model-based (or parametric) signal-processing techniques have also been employed to study radar data, as discussed in the following section.

(2) HRR data property studies by using parametric signal processing methods

This body of research assumes that HRR radar returns conform to certain data models such as the Autoregressive (AR) model, the Autoregressive Moving Average (ARMA) model, or Prony models. Therefore, much of this research naturally concludes that radar signals possess properties similar to those models. We refer to this class of models as data-
driven models to emphasize that they are different from those physics-based models that we will discuss in details in Section 2.2.

This type of model representation for HRR signatures has several obvious advantages. First, only a limit number of parameters, or variables, exist in each model, and these parameters completely determine the properties of the model. Therefore, once the parameters of the model are estimated - perhaps by using only a relatively small number of radar signals - the complete behavior of the entire class of radar data can be derived from the properties of the estimated model. Additionally, the parameters of the models naturally serve as a set of representative features for the targets. Consequently, a number of researchers have attempted to use data-driven models, which we summarize here.

In [86] Nuthalapati assumed that HRR radar signals match the properties of an AR model. In contrast, the ARMA model is proposed by Carrière et. al. [87] to represent the radar signatures. An extension of these two studies was presented by Eom et. al. [45], where a multiscale ARMA was proposed. This study shows that "the radar signal at a different scale obeys an ARMA process if it is an ARMA process at the observed scale." They also provide an algorithm to obtain the ARMA parameters at different scales directly from the ARMA parameters at the observed scale.

Eom [40] introduced another development in this thread of research in which the non-stationary property of radar signatures was taken into consideration. A time-varying AR (TVAR) model was applied to characterize HRR radar signatures. This study concluded that the TVAR model is capable of representing a non-stationary radar signature with a smaller number of parameters, which are also suitable for use as a set of target features.
Studies of radar-data based on Prony models have also been conducted in the past decade. The 1D Prony model was employed by Carrière et. al. [8], Hurst et. al. [11], and Zhang et. al.[72] to represent the radar signatures, while Sacchini et. al. [24] developed a 2D Prony model and applied it to 2D radar data. All of these studies claim that the Prony model is a good approximation of radar signals.

(3) HRR data statistical property studies

Because HRR signatures demonstrate considerable variability in many application scenarios, and because they are hard to precisely describe by available signal models, some researchers have approached the problem from a different perspective. They have considered the radar data as random processes [7, 57, 64, 65, 66, 88]. Thus these studies of HRR data have become studies of the statistical properties of the measured HRR data.

Williams et. al. [57] experimentally studied the statistical properties of the Mean Square Error (MSE) of HRR signatures generated from MSTAR (Moving and Stationary target Acquisition and Recognition) program. Two types of distribution were examined, an in-class distribution and an out-class distribution. Distribution separability was quantified using the Fisher criterion and the Bhattacharyya distance. Results obtained from both measures demonstrated the feasibility of using HRR signatures for ATR applications. Moreover, the results obtained from the study employing Bhattacharyya distance showed that the separability between different targets is due mainly to the mean difference, while the variance difference contributes little to the classification performance.

In [88] the statistical properties (mean and standard deviation) of the correlation coefficient, or matching scores, between two HRR signatures were studied. Some
interesting observations were presented. For example, the HRR signatures obtained at a particular carrier frequency can serve as the feature vectors for radar signatures obtained at shifted frequencies if the range resolution is high enough. This study also noted that radar systems with higher range resolution could tolerate more aspect variance. All of these statements are useful for designing an HRR template database for ATR systems based on signature correlation classifiers.

2.1.1.2 Fundamental Studies Based on the Physics of HRR Data Generation

In this section we discuss research that incorporates an understanding of 1) the HRR generation mechanism, 2) the radar system processing, and, 3) the physical properties of EM propagation in order to construct mathematical models that approximate the actual physical mechanisms of HRR data generation. Based on these models, computer simulations can be constructed and used to generate synthetic radar data, and/or to predict the radar returns under given conditions.

(1) Mathematical models of HRR data

The models presented in this subsection are different from the data-driven models discussed in 2.1.1.1. Because the mathematical models discussed here are derived from knowledge of the radar data generation mechanisms, instead of being based on the behavior of the measured radar data, they are usually referred to as physics-based models of HRR data. These models are usually presented in simplified, and thus tractable mathematic forms. Because of the simplified forms employed in the models, extensive theoretical analysis of the properties of HRR data becomes possible. Moreover, all these models are
characterized by a set of physically meaningful parameters, which are therefore promising candidates as a set of representative physical features -- if they can be efficiently and accurately extracted.

These obvious advantages of physics-based models have motivated researchers to contribute considerable efforts in this direction, and currently a number of meaningful physics-based HRR models have been developed. For example, Li, et. al., devised a collection of moving target models [2, 5], Jacobs, et. al., employed a set of random models in their studies [7], Hurst, et. al., used Prony models to approximate the a physics-based radar model [11], and Potter, et. al., proposed a group of GTD-based radar models[62].

Feature extraction, especially model-based feature extraction, is related to one of our research focuses and we therefore present more detail concerning a selection of representative models as well as the related parameter estimation algorithms, or feature extraction algorithms, in Section 2.2.

Although closed-form expressions can be an efficient tool to represent radar data, computer codes can be used to demonstrate complex scattering interactions. Accordingly, computer simulations have become an increasingly important alternative for the study of radar data and are discussed in the following section.

(2) Computer simulations of radar data

In most cases, the physical generation processes of radar returns are far too complex to be accurately modeled by any simple, and thus tractable, mathematical expression. Moreover, many ATR research groups need radar data which represents certain radar,
target, and/or clutter conditions, such as the specified scatterer configuration of the targets, controlled target-clutter interaction, returns obtained at a particular aspect angle, etc. However, obtaining measured radar data under such a range of conditions is generally either prohibitively expensive or infeasible. As a result, sophisticated computer simulations have been developed to predict radar returns under controlled conditions.

These simulations describe the radar data generation process using programming codes. With consistently increasing computation power, more and more details of the radar generation mechanisms can be taken into consideration. Consequently, synthetic radar data has come to more closely resemble measured radar data. Currently, some of these systems are so successful that researchers tend to conduct ATR studies based on the synthetic data generated by these systems. [65,66]

One of the most established of these systems is named Xpatch, which was developed by DEMACO, Inc. It is a code suite for simulating and analyzing high-frequency radar signatures. It uses the *shooting and bouncing ray method* to create realistic far-field and near-field radar signature simulations for 3-D target models. It has proved quite accurate at predicting signatures for ground targets, air targets, missiles, and in-space objects. [90]

We classify all of the research we have reviewed here as fundamental studies because they aim at increasing the fundamental knowledge of HRR radar. These studies establish the theoretical foundation of subsequent HRR-based ATR research. That is, all of the following investigation, such as ATR algorithm formulation and ATR performance evaluation, is based on the knowledge and results obtained from the previously mentioned studies.

Next we turn our attention to the algorithms that can be used to process HRR data.
2.1.2 Algorithm Formulation

A successful HRR-based ATR system requires a combination of algorithms which achieve different goals. First, clutter and jamming suppression algorithms are used to improve the signal-to-clutter ratio (SCR), which is indispensable for any subsequent robust feature extraction processes. Feature extraction algorithms are then used to extract a set of robust and high performance target features. Based on this set of extracted features, classification algorithms are employed to define high performance classifiers. Finally, evaluation algorithms are used to predict the potential performance of extracted features and classifiers for previously un-sampled targets, configurations, etc.

We group our review of the currently available ATR algorithms in three different categories. In Sub-subsection 2.1.2.1 algorithms involved in pre-processing, such as clutter suppression and feature extraction, are presented. Different classifiers are summarized in Sub-subsection 2.1.2.2. Sub-subsection 2.1.2.3 presents some performance evaluation algorithms.

2.1.2.1 Clutter (Jamming) Suppression Algorithms and Feature Extraction Algorithms

Clutter generally refers to the radar returns arising from the many undesired scatterers naturally located around the targets, while jamming refers to the strongly interfering EM field emitted intentionally or unintentionally by a few isolated sources. We focus here mainly on clutter suppression to simplify our review.

In HRR radar mode, a single radar return includes all of the clutter located along the azimuth direction for each range bin. Each return typically has very low SCR [1]. Thus,
different radar imagery techniques have been developed to improve the SCR. In application scenarios involving stationary targets, SAR enhances the SCR by improving the cross-range resolution. This is achieved in SAR by the coherent processing of a sequence of radar returns from the same scatterer. In moving target scenarios, the Doppler effect is generally utilized to separate targets from clutter. Because the targets are moving and the clutter is assumed to be stationary, the Doppler frequencies of clutter will generally focus around a small spectral region jointly determined by the radar squint angle and the radar platform speed. In contrast, the spectral region occupied by the targets can spread over the spectrum. Therefore, it is possible to isolate the target returns from the clutter returns on the range-Doppler image. [1]

However, if the target returns happen to embed in the clutter's Doppler region in the range-Doppler image, extracting the target information is difficult and may require extra clutter suppression algorithms. Space-Time Adaptive Processing (STAP) is one of the radar signal processing techniques proposed to resolve this problem [29]. Implementing STAP requires sampling the radar returns at each element of an antenna array, and over a sequence of pulse returns. The output of STAP is a linear combination, i.e., a weighted sum, of the input signal samples, while the weights are adaptively computed to reflect the actual noise, clutter and jamming environment in which the radar operates.

Different clutter suppression schemes were also proposed in [2] and [37]. Because these clutter-suppression algorithms can also be considered as part of the feature extraction process, we will present them in Section 2.2 in greater details.

---

1 The "space" in the name of STAP refers to the spatial diversity among the radar elements.
2 The "time" in STAP refers to the time diversity obtained by processing sequences of returns.
Because developing feature extraction algorithms is one of our research objectives, and because many feature extraction algorithms are closely related to the specific radar data models employed, we are going to present the feature extraction modules as well as the related models, in detail in Section 2.2.

2.1.2.2 Classification Algorithms

Classification is the core part of an ATR system. All of the pre-processing steps, including feature extraction, are directed toward improving the final classification rate, or target identification rate.

Theoretically speaking, any classification technique and detection algorithm can be applied to HRR-based ATR [84]. However, besides high accuracy, a practical classifier involved in HRR-based ATR puts extra emphasis on implementation simplicity and real-time operation. In order to meet both of these two requirements, the general strategy currently employed is to use relatively simple classifiers to meet the implementation and the speed requirements, while the classification performance is mainly guaranteed by the high separability of the carefully defined and extracted HRR features.

Various classifiers have been proposed in the HRR-based ATR literature. For example, a correlation-based linear classifier was proposed in [28], a minimum distance classifier was utilized in [45], and a constrained quadratic classifier was used in [64]. Similarly, k-nearest neighbor classifiers were suggested in [18], a maximum likelihood classifier was employed in [7, 66], and neural network (NN) classifiers were examined in [32, 33]. All of these classifiers demonstrate advantages and disadvantages in different application scenarios, over different data formats, and for different feature separability characteristics.
Because the overall performance of an ATR system is mainly decided by the inherent separability of the classification features, and because our research is focused mainly on the feature studies, we will not expand on the topic of ATR classifiers here.

2.1.2.3 Evaluation Algorithms

Some HRR-based ATR systems, especially in their development phase, require a reliable end-to-end evaluation, or prediction, of their overall performance [89]. To achieve this goal, it is necessary to devise evaluation algorithms for different elements of an ATR system. Here, we only list some of the representative algorithms.

(1) Evaluation of the HRR data model

Radar model evaluation generally involves two issues. The first is examining the correspondence between the proposed model and the real-world radar data. HRR data models include both data-driven models and physics-based models. Because many radar data processes are based on these models, their accuracy becomes quite important. It is usually necessary to first evaluate the correspondence between the proposed models and real-world radar data before adopting those models. Unfortunately, data-driven models are generally utilized without extensive examination in the existing ATR literature. In contrast, many of the proposed physics-based radar models have been evaluated by using EM field analysis and experimental examination [62].

The second issue related to radar model evaluation is justification of the common practice of using a simple radar model to approximate the much more complex actual system. Some physics-based models, while relatively accurately describing the radar data, are too complicated to analyze, or in some cases too complicated to allow reliable
parameter estimation algorithms to be determined. Therefore, simpler models are typically devised to approximate the more complex models. Both quantitative justification and performance-degradation estimation have been studied [124].

(2) Extracted feature set evaluation algorithms

Feature set evaluation (FSE) is a general problem in the pattern recognition field. Basically, FSE is used to rank the capacity of different feature sets for distinguishing their underlying classes. This topic was first studied as a critical part of feature selection and extraction algorithms. Significant effort was expended in this area, and a number of important results were obtained in 1960s and 1970s [127, 128, 129, 130, 136]. A chapter in the "Handbook of Statistics", edited by Krishnaiah et al. [127] provides a comprehensive review of those results.

Because the ultimate objective of classification is to minimize the misclassification rate, the most natural criterion for evaluating the class separability under different feature sets is the probability of classification error, $P_e$. The feature set that produces smallest $P_e$ is certainly the best feature set. However, because of the difficulty in estimating $P_e$ [84], and because of the intrinsic problems embedded in employing $P_e$ as a criterion [127], it is not very practical to directly use $P_e$ except in a limited number of special cases. In order to circumvent the difficulty in directly using $P_e$, many $P_e$-related alternative criteria have been subsequently proposed. These primarily fall into three categories: (1) information (uncertainty)-based criteria [127, 131, 137], (2) distance-based criteria [127], and (3) dependence-based criteria [122, 127, 137]. Note that all these criteria are based on extensive knowledge of the probability distributions of the underlying classes. However,
except for some special cases, estimating the probability distribution from limited number
of patterns is an ill-posed problem, and reliable estimation results cannot be guaranteed.
Thus, most of the FSE algorithms are not easily put into practical use.

To avoid the difficulty of finding the explicit form of the probability distribution, other
criteria which are calculated directly from a set of given patterns, were proposed in 1980s
and 1990s. Among these criteria are Boundary method-based FSE method [120, 126],
neuro-fuzzy method based on feature evaluation index [117], set-cover-based algorithm
[118], decision-boundary-based algorithm [141], and scatter matrix-based criteria [84].

2.1.3 System Integration

The focus of this research category is to determine a practical HRR-based ATR system
constrained by the availability of algorithms and techniques, by a limited budget, and by
other practical factors. Obviously, the success of research in this category is heavily
dependent on the success of research in the first two categories (fundamental studies and
algorithm formulation). However, most research into HRR-based ATR is thus far in a
premature stage, and quite a few challenging obstacles still need to be overcome.
Consequently, designing a practical HRR-based ATR system remains a challenging goal.

However, a fieldable system is obviously the final goal of all the studies we have
previously mentioned. Moreover, system level considerations have already been proposed
as part of the studies on HRR-based ATR in several governmental- or military-supported
contracts, such as TRUMPETS [89]. Therefore, although only a few papers have been
devoted to this subject, some researchers have begun to take this system-integration issue
into consideration, and have tried to design the individual algorithms from a system perspective.

2.2 HRR Data Modeling and Model-based Feature Extraction Algorithms

Because 1D HRR data, or HRR signatures, is generally plagued by scintillation effects, extracting robust target features is a difficult task. Consequently, 2D HRR data has recently attracted more attention. However, compared with SAR and optical images, one of the major disadvantages of 2D HRR data is that 2D HRR does not directly manifest geometric information of targets, such as the shape and size of targets, and the optical relationship between targets and clutter, such as shadows. Thus, many image-based features and image-processing-based extraction algorithms are not applicable to 2D HRR data, or its corresponding range-Doppler images. This disadvantage limits the availability of both the candidate features and the candidate extraction algorithms for 2D HRR data. Model-based features and their extraction algorithms, therefore, play a more important role in HRR-based ATR than in SAR-based application scenarios.

There are currently a number of physics-based HRR data models proposed in the literature. The parameters of these models have an associated physical meaning, and are therefore a set of promising representative features. That is, in this context, feature extraction is equivalent to estimating the model parameters.
2.2.1 Scatterer-based Modeling and Feature Extraction for Stationary Targets

Radar returns can be approximated by a sum of the responses from a group of isolated scatterers if the radar operates at sufficiently high frequency [123]. This mechanism provides the foundation of a large class of physics-based radar models --- scatterer-based models. The popularity of scatterer-based models is due mainly to their combination of accuracy in representing the radar data and their inherent tractability when used in subsequent analysis and feature extraction.

There are a number of proposed stationary target models with different complexities, accuracies, and application scenarios [3, 4, 11, 26, 62] The most extensively studied and the most representative group of models are the Geometric Theory of Diffraction (GTD) - based models, which were proposed by the research group led by L. Potter and R. Moses. A summary of their investigations and results in this field was provided in [26], and we restate part of this summary here to make this proposal self-contained.

The general form of the far-field GTD-based models can be expressed as follows:

$$E(f, \theta) = \sum_{m=1}^{M} S_m(f, \theta) \exp\left\{-j \frac{4\pi f}{c} [x_m \cos \theta + y_m \sin \theta]\right\}$$

(2.1)

where $f$ is the frequency in Hz of the radar signal, $\theta$ is the target aspect angle, $S_m(f, \theta)$ is the $m$th scatterer's amplitude determined by the frequency and aspect angle, $M$ is the number of scatterers, $c$ is the propagation speed of the EM wave, and $(x_m, y_m)$ is location of the $m$th scatterer on the slant-plane .

The frequency and aspect-angle dependent amplitude, $S_m(f, \theta)$, can be separately represented as the frequency dependent term and the aspect-angle dependent amplitude
term. With regard to the frequency dependence, the relation between amplitude and the frequency is expressed in the form of \((f)^{\alpha_m}\), where \(\alpha_m\) is a discrete parameter, and is an integer multiple of \(1/2\). Geometric patterns of the scatterers can be predicted from the value of \(\alpha_m\), and thus, \(\alpha_m\) is referred to as the type parameter. The correspondence between the value of \(\alpha_m\) and the geometric pattern of canonical scatterers is listed in Table 2.1. With regard to the angular dependence, there is no widely accepted simplified form. In [62], L. Potter adopted a simple but somewhat general form \(\exp\{\beta_m \theta\}\) to represent the angle dependence over a small angular interval.

As a result, a more specific form of the radar-backscattered electric field can be presented as follows.

\[
E(f, \theta) = \sum_{m=1}^{M} A_m \left( \frac{j f}{f_c} \right)^{\alpha_m} \exp\{\beta_m \theta\} \exp\{-j \frac{4\pi f}{c} [x_m \cos \theta + y_m \sin \theta]\}
\]

where \(f_c\) is the carrier frequency in Hz of the radar system, \(A_m\) is a complex value used to characterize the basic scattering properties of the \(m\)th scatterer, \(\alpha_m\) is the type parameter to characterize the \(m\)th scatterer's frequency dependence that is a dimensionless, discrete type parameter characterizing the geometry [62] of the \(m\)th scatterer. Finally, \(\beta_m\) is the angular dependence parameter of the \(m\)th scatterer.
Value of $\alpha$ & Possible Geometries of Scatterer
---
-1 & corner diffraction
- $\frac{1}{2}$ & edge diffraction
0 & point scatterer, doubly curved surface reflection, straight edge specular
$\frac{1}{2}$ & singly curved surface reflection
1 & flat plate at broadside, dihedral

Table 2.1 Correspondence between the value of $\alpha$ and the geometry of canonical scatterers

A simplified 1D version of Eq. (2.2) is more widely adopted in related literature [38, 43, 61, 62]. In the 1D model, the aspect angle dependent term is ignored, and it can therefore be presented as follows:

$$E(f) = \sum_{m=1}^{M} A_m (j \frac{f}{f_c})^{\alpha_m} \exp\{-j \frac{4\pi f}{c} d_m\}$$

(2.3)

where $A_m$ and $\alpha_m$ carry the same physical meaning as those in (2.2), and $d_m$ is the range of the $m$th scatterer with respect to a zero phase reference point.

If the aspect angle interval is very small, and if the radar is narrowband, a discrete form of the model (2.2) can be obtained under the condition of sampling occurring on a polar grid:

$$E(k,l) = \sum_{m=1}^{M} \tilde{A}_m (1 + j \frac{\Delta f}{f_c})^{\alpha_m} \exp\{\beta_m \Delta \theta\} \exp\{-j \frac{4\pi f_c}{c} \frac{\Delta f}{f_c} (k x_m + \Delta \theta y_m)\}$$

(2.4)

where $\Delta f$ is the frequent sampling interval, and $\Delta \theta$ is the aspect angle sampling interval.

The accuracy of this group of GTD-based models degrades with the decrease of the target's electric size, $L/\lambda$, where $L$ is the target length, and $\lambda$ is the wavelength.
From equation (2.2) and (2.4) we see that the physical information of targets is encoded in a collection of model parameters:

\[
\{\widetilde{A}_m, x_m, y_m, \alpha_m, \beta_m\}_{m=1}^{M} \quad (2.5)
\]

As we mentioned previously, extracting a set of physical features from measured radar data is equivalent to estimating this set of parameters, (2.5), from the measured data.

A straightforward method is to estimate those parameters by a Least Squares Fit (LSF) to the measured data. Under the assumption of additive complex Gaussian white noise, the nonlinear LSF problem can be reduced to Maximum Likelihood (ML) estimation, and the estimation variance asymptotically achieves the Crame-Rao lower bound if the a proper model is selected. D. Chiang [124] derived a ML estimator and provided a detailed description of the parameter estimation algorithm of the proposed GTD-based model.

However, accurately solving the ML estimator is computationally intensive, and the solution is very sensitive to the initial estimation of the parameters. Moreover, obtaining the globally optimal solution cannot be guaranteed. Therefore, considerable effort has been expended to find robust and computational affordable parameter estimation algorithms.

In a number of studies, it has been argued that the GTD-based model presented by Eq. (2.4) can be well approximated by damped exponential signal models if the radar is narrowband [8, 11, 12, 15, 16, 17]. D. Chiang [124] quantified the upper bound of the error between the Hankel matrix of GTD-based models and that of the damped exponential models, which further justifies the model approximation, and showed that the estimation of scatterer locations obtained from the damped models closely approximates those estimated from the GTD-based models.
Once the GTD-based models are replaced with the damped exponential models, a large number of both computationally efficient and statistically accurate parameter estimation algorithms are readily available. Here, we provide only a partial list of the 2D algorithms since this is not a primary objective of our research.

Most of the proposed 2D estimation algorithms, such as the 2D TLS-Prony [24], MEMP [138], and 2D Method Of Direction Estimation (MODE)[140], are simply the combination of two separate 1D estimations. One of the 1D estimations is conducted downrange, the other along the cross-range direction. The estimated downrange modes and cross-range modes are finally combined to form the 2D modes by a process named *pairing*. Different algorithms employ different 1D estimation algorithms and different pairing schemes.

However, the 2D IQML technique [139] is fundamentally different from the other techniques, and is therefore called by some researchers the "true 2D" algorithm. This is because it exploits the 2D exponential structure directly from the data, and no pairing process is needed in this algorithm.

Obviously, all of these 2D estimation algorithms have strong points and weak points. A comparison between these 2D algorithms was conducted by Ying et. al. [58] with respect to their relative memory requirements, computation requirements, and estimation accuracy.

In some cases, the GTD-based models can be further simplified with undamped exponential models. In [124], D. Chiang also quantified the upper bound of the error between GTD-based models and the undamped exponential models.

The estimation algorithms used for undamped exponential models are basically the same as those for damped models, except that the undamped models can exclusively...
employ 2D FFTs for parameter estimation. A 2D FFT-based estimation algorithm, 2D CLEAN, was also compared with other algorithms in [58]. It was labeled as simple, computationally efficient, but with low-resolution, which is an inherent drawback of the FFT.

All of the models presented previously intentionally ignore the effects of antenna and wave polarization to simplify the models. However, many studies have demonstrated that considerable amounts of additional target information could be extracted from the different combinations of the transmission and receiving polarizations [12, 13, 14, 39, 60, 69, 70, 73]. Therefore, many physics-based models were subsequently presented in polarimetric form. Some of those models are simple extensions of the single polarization models produced by replacing the scalar amplitude with a vector of amplitudes of different polarimetric pairs. We do not list those models here because the essential portions of those models are consistent with the non-polarimetric models we presented previously. For those readers who have interest in the application of polarization properties, please refer to the listed references.

2.2.2 Scatterer-based Modeling and Feature Extraction for Moving Targets

Although the models presented in the previous subsection can describe the radar data generation mechanism of stationary targets, they cannot be directly applied to represent the radar returns of moving targets. This problem occurs because the Doppler frequencies, one of the most important properties of moving target radar returns, are not included in those
models. That is, valid moving target radar models must incorporate the target Doppler effect, arising from the target's radial motion.

Currently, only a few moving target models have been proposed in the literature. In [2], the research group led by Li proposed a discrete moving target HRR model for a single rigid-body target. The model has the form:

$$y(n,l) = \sum_{k=1}^{K} \alpha_k \exp\{j2\pi f_k n\} \exp\{j2\pi f_d l\} + c(n) + e(n,l)$$

(2.6)

where $K$ is the number of scatters, $\alpha_k$ is complex amplitude of $k$th scatterer, $f_k$ is the frequency of $k$th scatterer, $f_d$ is the normalized Doppler shift of the moving target, $c(n)$ represents the undesired background clutter, $e(n,l)$ denotes the additive noise, $N$ is the number of available data samples for each pulse, and $L$ is the number of pulses transmitted by the HRR radar during the dwell time.

The rigid-body requirement implies that all of the scatterers on the target have the same Doppler frequencies, which significantly simplifies the form of the proposed model, and therefore facilitates the derivation of parameter estimation algorithms. From Eq. (2.6), we know the feature set, or the parameters of the model, can be listed as follows:

$$\{f_d, C, \{\alpha_k, f_k\}_{k=1}^{K}\}$$

(2.7)

where $C = [c(0), c(1), ..., c(N-1)]^T$

There are three types of parameters to be estimated: the clutter vector, $C$; the Doppler frequency, $f_d$; and the range feature modes, $\{\alpha_k, f_k\}$.

A set of estimation algorithms named MOVE-1 and MOVE-2 were derived and proposed to estimate these parameters in [2]. The estimation method employed is based on
a nonlinear LSF. However, estimating all three types of parameters at the same time was
shown to be a complicated problem, and obtaining the globally optimal result was
problematic. This motivated the authors of [2] to present an alternating optimization
approach, i.e. MOVE-1, which iterates between two NLS sub-problems until certain
terminating criteria are met. One sub-problem is estimating the Doppler frequency, \( f_d \), and
the range feature modes, \( \{ \alpha_k, f_k \} \) given the most recently obtained clutter vector, \( C \). Another
sub-problem is estimating the clutter vector, \( C \), given the most recently estimated Doppler
frequency and range modes. However, when two scatterers are closely spaced, namely,
when two range frequencies, \( f_k \)'s, are close to each other, the MOVE-1 algorithm exhibits
slow convergence, and thus requires intensive computation. To solve this problem, MOVE-
2, a revised version of MOVE-1, was provided. MOVE-2 employs a modified MODE
algorithm to estimate the range frequency, \( f_k \). As for determining the model order, or the
number of scatterers, the Generalized Akaike Information Criterion (GAIC) is utilized in
[2].

However, limiting the moving target to only rigid-bodies is not very practical, and
makes application of the model difficult because there is no clear dividing line between the
rigid body and non-rigid body targets. Instead, this model would no longer be valid in very
common application scenarios, such as cases in which both rigid body and non-rigid body
targets appear in the same measurements, as well as cases when tracked and/or wheeled
vehicles are being illuminated.

Therefore, Li extended this rigid-body moving target model to scenarios where non-
rigid-body targets would probably appear in [5]. The new model can be expressed as
follows:
\[ y(n,l) = \sum_{k=1}^{K} \alpha_k \exp\{j2\pi f_k n\} \exp\{j2\pi f_{dk} l\} \exp\{j2\pi f_{nk} nl\} + c(n) + e(n,l) \quad (2.8) \]

\[ n = 0,1,\ldots,N-1, \quad l = 0,1,\ldots,L-1. \]

where \( N \) is the number of available data samples for each pulse, \( L \) is the number of pulses transmitted by the HRR radar during the dwell time, \( K \) is the number of scatters, \( \alpha_k \) is the complex amplitude of the \( k \)th scatterer, \( f_k \) is the frequency of the \( k \)th scatterer, \( f_{dk} \) is the normalized Doppler shift frequency of the \( k \)th scatterer, \( f_{nk} \) represent the coupling between the range frequency and the Doppler frequency of the \( k \)th scatterer, \( c(n) \) denotes the undesired background clutter, and \( e(n,l) \) represents the additive noise. The relation between \( f_{dk} \) and \( f_{nk} \) is \( f_{dk} = \rho f_{nk} \), where \( \rho \) is a system-dependent constant. No parameter estimation algorithm was provided for this moving target model.

### 2.2.3 Random Model of HRR Data

All of the models described above assume that the underlying radar is operated at very high frequency. Therefore, targets can be approximated by a group of isolated scatterers, and radar signals can be described by a sum of the returns from all the individual scatterers. However, if the high-frequency condition is not satisfied, the integration over the radar returns from a continuous target surface is needed to replace the summation of those individual returns.

One model that takes these changes into account was presented by Jacobs, et al., [7]. We refer to this model as a random model to maintain consistency with the name used in [7], although this name is not very meaningful in our context.

The model can be presented as follows:
\[ r(t) = s(t; \theta, \alpha) + w(t) \]  

where:

\( w(t) \): additive noise,

\( \theta \): orientation of the target,

\( \alpha \): target type,

\[ s(t; \theta, \alpha) = \sqrt{E} \int \{ \exp \{ j \frac{4\pi f}{c} u^T x \} c(x) d\mathbf{x} \mathcal{S}_r(f) \} \exp \{ j 2\pi f t \} df \]

where

\( u = [\sin \theta \cos \phi, \sin \theta \sin \phi, \cos \theta]^T \): the body-centered line-of-sight unit vector,

\( \mathcal{S}_r(f) \): the Fourier transform of the transmitted signal,

\( x = [x, y, z]^T \): the location of different parts of the target.

\( c(\rho) = c_g(\rho) \exp \{ j \psi(\rho) \} + c_s(\rho) \): the reflectivity density of the target at location \( \rho \),

where

\( c_g \): is a deterministic glint component, and,

\( c_s \): is a diffuse or speckle component, and is a complex Gaussian random process that is spatially white,

\( \psi \): is a random variable with a known distribution depending on the angle between the line-of-sight and the surface normal at position \( \rho \).

Because of the complexity of this model, it is hard to apply it to practical applications, and no parameter estimation algorithm is currently available for this model.

The next chapter addresses the properties of HRR signatures.
HRR SIGNATURE STUDIES: COMPLEX VS. MAGNITUDE-ONLY

Although HRR signatures are obtained and stored as complex values [53], most of the current HRR-signature-based ATR algorithms only make use of the amplitude part [7, 28, 32, 44, 57, 58, 65, 66, 88]. In most cases the phase information has simply been omitted without careful examination and explanation. However, in the past 4 to 5 years, some researchers have questioned this treatment, and argued that the phase of the complex signatures might contain meaningful target information [50, 54, 55], and therefore, we may throw away valuable information if we only use the magnitude of the signatures.

Does the phase of HRR signatures indeed convey meaningful classification information? Finding the answer to this question is one of the main purposes of this chapter. Both theoretical analysis and experimental examination are employed to hopefully provide some persuasive observations on this issue.

3.1 Analysis of Complex HRR Signatures

The theoretical analysis in this section is based on a simple complex HRR signature model. We argue that any insight regarding phase-utility should come from a careful study
of the properties of the complex signatures. Therefore, this section is devoted to the analysis of the general properties of the complex HRR signatures, with a special emphasis on the utility of phase in target classification.

We first present a simple HRR signature model on which our analysis is based in Sub-section 3.1.1. The analysis of HRR signatures containing either a single-scatterer or multiple scatterers is presented in Sub-section 3.1.2. In addition, we discuss the relationship between the aspect angle and the HRR signature variance in Sub-section 3.1.2. In Sub-section 3.1.3 our analysis results are summarized.

### 3.1.1 Modeling

The model employed in this paper represents the Phase History Response (PHR) of HRR radar, and can be expressed as:

\[
Y(k) = C \sum_{l=1}^{N} a_l \exp\left\{ j 4\pi \frac{d_l}{\lambda} \right\} \exp\left\{ j 2\pi \frac{\frac{d_l}{W} k}{\lambda} \right\}, \quad k = 0...N - 1. \tag{3.1}
\]

where

- \( Y(k) \): is the complex PHR of the HRR radar at discrete temporal frequency \( k \);
- \( N \): is the number of samples in each HRR signature;
- \( C \): is a complex constant that models both the unknown radar system gains and unpredictable atmospheric effects;
- \( a_l \): is the complex scattering coefficient of the \( l \)th scatterer;
- \( d_l \): is the relative location of the \( l \)th scatterer along the radar line-of-sight (LOS) relative to a reference point in the radar patch.
\( \lambda \): is the nominal radar wavelength, which satisfies \( \lambda = \frac{c}{f} \), where \( c \) is the speed of light, and \( f \) is the center frequency of the radar signal.

\( W \): is the length of the radar ground patch.

Generally speaking, the Pulse Repetition Interval (PRI) of an HRR system is around 150us, and HRR radar sends around 128 pulses during each transmission period, so the dwell time for a HRR radar is thus only \( 150\text{us} \times 128 = 0.0192s \). Parameter \( C \) models the totality of effects due to many unpredictable and subtle factors, such as the changes of atmosphere and EW propagating channel, which we assume won't dramatically change during a period of 0.0192s. Therefore, \( C \) can be approximated as a constant during reasonably short time intervals. More precisely, \( C \) can be considered a constant across a sequential collection of HRR signatures. Also, any change of \( C \) will have a uniform effect on all the scatterers, and canceling this parameter will not affect the relative values between scatterers. Subsequently, we ignore \( C \) in our following analysis.

We need to clarify one point. Although this model is simple, it is typical and representative [47-50]. For example, with the exception of \( C \), it is easy to show that this model is indeed a simplified, discrete version of the model presented by Welch and Hawley, 1998 [47]. Therefore, our analysis of (3.1) may also provide insights into the other HRR signature models.
3.1.2 Simulated HRR Signature Analysis

3.1.2.1 HRR Signature

The HRR signature can be obtained from the inverse Fourier Transform of the PHR [48, 50]. Thus, the discrete HRR signature can be modeled as:

\[ P(n) = \text{IFFT}(Y(k)) \]

\[ = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{i=1}^{s} a_i \exp\{j4\pi \frac{d_i}{\lambda} \exp\{j2\pi \frac{n}{N} k\} \exp\{j2\pi \frac{n}{N} k\} \]

\[ = \frac{1}{N} \sum_{i=1}^{s} a_i \frac{\sin\{\pi\left(\frac{d_i}{W} + \frac{n}{N}\right)N\}}{\sin\{\pi\left(\frac{d_i}{W} + \frac{n}{N}\right)\}} \exp\{j4\pi \frac{d_i}{\lambda} \} \exp\{j\pi\left(\frac{d_i}{W} + \frac{n}{N}\right)(N-1)\} \]

(3.2)

where \( n \) is the range-bin index. That is, \( P(n) \) can be expressed as:

\[ P(n) = \sum_{i=1}^{s} A(l, n) \exp\{j\Phi(l, n)\} \]

(3.3a)

where

\[ A(l, n) = \frac{1}{N} |a_i| \frac{\sin\{\pi\left(\frac{d_i}{W} + \frac{n}{N}\right)N\}}{\sin\{\pi\left(\frac{d_i}{W} + \frac{n}{N}\right)\}} \]

(3.3b)

\[ \Phi(l, n) = \frac{\pi(N-1)}{N} n + \left(\frac{4\pi}{\lambda} + \frac{\pi(N-1)}{W}\right)d_i + \phi_i \]

(3.3c)

\[ a_i = |a_i| \exp\{j\phi_i\} \]

(3.3d)

3.1.2.2 HRR Signatures Containing Only A Single Scatterer

We first study the case where \( P(n) \) contains only a single scatterer, i.e., \( s=1 \). From (3.3a), we obtain:

\[ P(n) = A(l, n) \exp\{j\Phi(l, n)\} \]
Basically, Equation (3.4) encodes two pieces of physical knowledge regarding the underlying scatterer: (1) its location, which is denoted by $d_i$, and (2) its scattering coefficient, denoted by $a_i = |a_i| \exp{j\phi_i}$.

From (3.3b), we know that $A(l,n)$ is the discrete version of the sinc function. Thus $|a_i|$ is encoded by its peak value, and $d_i$ is encoded in the peak location. Therefore, $A(l,n)$ captures the characteristics of the scatterer. From (3.3c), we note that $\Phi(l,n)$ is a linear function of $n$, and thus the essential knowledge of the scatterer is only encoded in the initial phase value, $\Phi(1,0)$. These observations enable us to draw two conclusions about the phase of a single scatterer in the HRR signature:

The target knowledge contained in the phase of a single-scatterer complex signature is redundant. That is, the knowledge embedded in the phase can also be derived from the magnitude in the observed/measured signatures.

The target knowledge encoded in the phase of different range bins is equivalent. That is, knowledge identified via the absolute phase value at any single range bin is equivalent to that obtained via the phase at the initial range bin.

Of course, obtaining the absolute phase value can be a very difficult problem. One difficulty, for example, is that $\exp{j\phi}$ is a periodic function of $2\pi$, and therefore, all the detected phases are folded into a interval of $2\pi$ length [84]. Another interesting observation is that the initial phase $\Phi(1,0)$ is the sum of $d_i$ and $\phi_i$. That is, from the viewpoint of the complex signature phase, $d_i$ and $\phi_i$ are interchangeable, and different combinations of $d_i$ and $\phi_i$ can manifest the same phase values, which further exacerbates the estimation of the absolute phase.
3.1.2.3 HRR Signature Containing Multiple Scatterers

Next we study the case where $P(n)$ contains more than one scatterers, that is $s > 1$.

Let

$$P(n) = M(n) \exp \{j \Theta(n) \}. \tag{3.5a}$$

From (3.3) we can obtain that

$$\Theta(n) = \tan^{-1} \left\{ \frac{\sum_{l=1}^{s} A_l(n) \sin(\Phi_l(n))}{\sum_{l=1}^{s} A_l(n) \cos(\Phi_l(n))} \right\}, \text{ and} \tag{3.5b}$$

$$M^2(n) = \left\{ \sum_{l=1}^{s} A_l(n) \cos(\Phi_l(n)) \right\}^2 + \left\{ \sum_{l=1}^{s} A_l(n) \sin(\Phi_l(n)) \right\}^2$$

$$= \sum_{l=1}^{s} A_l^2(n) + \sum_{m=1, k \neq m}^{s} A_m(n) A_k(n) \cos(\Phi_k(n) - \Phi_m(n))$$

$$= \sum_{l=1}^{s} A_l^2(n) + \sum_{m=1, k \neq m}^{s} A_m(n) A_k(n) \cos(\rho d_{mk} + \phi_{mk}) \tag{3.5c}$$

where

$$\rho = \frac{4 \pi}{\lambda} + \frac{\pi(N-1)}{W}, \tag{3.5d}$$

$$\Delta d_{mk} = d_k - d_m, \text{ and} \tag{3.5e}$$

$$\Delta \phi_{mk} = \phi_k - \phi_m. \tag{3.5f}$$

Note that $A_l(n)$ and $\Phi_l(n)$ are short-hand notation for $A(l,n)$ and $\Phi(l,n)$ respectively.

(1) Phase analysis

Because it is difficult to acquire any insight into the phase behavior by directly evaluating Equation (3.5b), we first simplify the equation by assuming that there are only two scatterers in the signature. That is, $s = 2$. Equation (3.5b), thus, becomes:
\[ \Theta(n) = \tan^{-1}\left\{ \frac{A_1(n)\sin(\Phi_1(n)) + A_2(n)\sin(\Phi_2(n))}{A_1(n)\cos(\Phi_1(n)) + A_2(n)\cos(\Phi_2(n))} \right\} \] (3.6)

By taking the properties of \( A_1(n) \) into consideration, we reach the following conclusions:

- For all \( n \) where \( A_1(n) \) is much bigger than \( A_2(n) \), \( \Theta(n) \equiv \Phi_1(n) \);
- For all \( n \) where \( A_2(n) \) is much bigger than \( A_1(n) \), \( \Theta(n) \equiv \Phi_2(n) \);
- For all \( n \) where \( A_1(n) \equiv A_2(n) \), \( \Theta(n) \equiv (\Phi_1(n) + \Phi_2(n))/2 \);
- For all but the above three cases, \( \Theta(n) \) transitions between the cases listed above.

The above statements indicate that the phase behavior of a multi-scatterer signature is influenced by the same factors as those noted in the single scatterer case if all the scatterers are widely separated across the range. On the contrary, when the scatterers along the range bins are close to each other, the phase behaviors of the scatterers interact. That is, the phase behaviors of the multi-scatterer signatures are basically the same as those of the single-scatterer signatures, except that there are transition regions in the multi-scatterer case.

Moreover, the magnitude alone appears to provide more accurate target information for both the locations and the magnitudes of the underlying scatterers, which is manifested in the subsequent analysis. This analysis demonstrates that (1) all the target knowledge encoded in the phase of the HRR signatures is also stored in the magnitude of the signatures, and (2) the magnitudes of the HRR signatures are easier to extract than their phase counterparts. However, we must note that our discussions here regarding the HRR signature phase-utility are far from conclusive, because (1) our discussion is based on a
model describing a stationary target, and we cannot yet determine whether phase is useful in moving target scenarios. (In fact, according our study in Chapter 4, the phase plays an important role in estimating the target.) and, (2) our discussion is based only on a single HRR signature, and we cannot conclude whether phase is useful if we make use of more than one signature at the same time.

(2) Magnitude analysis

To reinforce our conclusions regarding the magnitudes of multi-scatterer signatures, we analyze the behavior of magnitudes in this section. Our discussion of the magnitude of multi-scatterer signature is based on Equation (3.5c), (3.5d), (3.5e), and (3.5f).

When all the scatterers are widely separated along the radar line-of-sight (LOS), the square magnitude

$$M^2(n) = \sum_{i=1}^{s} A_i^2(n)$$

(3.7)

can be obtained by referring to the properties of $A_i(n)$. That is, every scatterer's existence is revealed by a peak in the magnitude of the HRR signature.

Assuming there are two scatterers mapped to the same range bin, $n$, in the signature, and $\Delta d_{12} \equiv 0$, (3.5c) can be simplified as:

$$M^2(n) = [|a_1|^2 + |a_2|^2 + 2|a_1||a_2|\cos(\Delta \phi_{12})]S_1^2(n)$$

(3.8a)

where

$$S_1(n) = \frac{\sin\{\pi N(\frac{d_1}{W} + \frac{n}{N})\}}{N\sin\{\pi(\frac{d_1}{W} + \frac{n}{N})\}}$$

(3.8b)
Therefore, if $\Delta \phi_{12} \equiv 0$, $\hat{M}(n) = |a_1| + |a_2| S(n)$, and if $\Delta \phi_{12} \equiv \pi$, $\hat{M}(n) = (|a_1|-|a_2|) S(n)$.

Thus, in agreement with our intuition, scatterers in the same range bin can interact either coherently or destructively. This suggests that the changes of the scattering coefficients are, at least, one of the sources of the scintillation effects notable in HRR signatures.

Assume that $\Delta d_{12}$ is small enough so that two scatters appear in the same range bin, $n$, in the signature. That is, $\Delta d_{12}$ is smaller than the radar range resolution. This $\Delta d_{12}$ can be introduced by the local vibration of scatterers. From Equation (3.5c), we know that $\Delta d_{12}$ could have the same effect on $M(n)$ as $\Delta \phi_{12}$. That is, changing the scattering coefficients of the two scatterers could have an effect equivalent to a small change of the relative location of these two scatterers.

### 3.1.2.4 Effects of Aspect Angle Variance

From the above analysis we know that there are two parameters for each scatterer encoded in the HRR signature. One is the complex scattering coefficient, $a$, and the other is the relative range location, $d$. Here, we ignore the effect of aspect angle on $a$, and only consider its effect on $d$. As we mentioned previously, changing the aspect angle affects $d$, which, in turn, has an affect on the signatures. Of course, the relation between $d$ and the aspect angle $A$ is [50]:

$$d = x \cos(A) + y \sin(A) = r \cos(A - S),$$  \hspace{1cm} (3.9)

where $(x, y)$ is the ground location of the scatterer, $S = \tan^{-1}(y/x)$, and $r = \sqrt{x^2 + y^2}$ is the ground distance between the scatterer and the ground radar reference point.
From Equation (3.3c) and (3.9), we can obtain the relation between the variance of the aspect angle and that of the phase behaviors of single-scatter signatures, which is:

\[ \Delta \Phi(l, n) = \rho \sin(A - S) \Delta A. \tag{3.10} \]

Therefore, the correlation between phase variance, \( \Delta \Phi(l, n) \), and aspect angle variance, \( \Delta A \), depends heavily on the ground location of the scatterer. If \( A \equiv S \), then \( \Delta \Phi(l, n) \) is nearly independent of \( \Delta A \). In contrast, if \(|A-S| \approx \pi/2\) any small \( \Delta A \) will be magnified by \( \rho \), and \( \rho \) is generally a large number. (\( \rho \) is approximately 561 if we employ typical operational parameters and set \( r = 1.5 \) meters.) This observation suggests that the different dependent patterns of the phases on the aspect angles have the potential to provide information regarding the relative locations of scatterers. However, in order to exploit this, the aspect angle interval between HRR signatures needs to be small enough to correctly trace the dependent patterns.

We further study the effect of aspect angle variance on the HRR signature magnitude by assuming a two-scatterer case. From Equation (3.9) we can obtain

\[ d_{12} = (x_2 - x_1) \cos(A) + (y_2 - y_1) \sin(A) = r_{12} \sin(A - S_{12} - \frac{\pi}{2}). \tag{3.11} \]

where \( S_{12} = \tan^{-1}((y_2 - y_1)/(x_2 - x_1)) \), and \( r_{12} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \) is the ground distance between these two scatterers.

Assuming that these two scatterers fall in the same range bin, \( n \), or \( \Delta d_{12} \equiv 0 \), from Equation (3.11) we, therefore, know that \( A - S_{12} - \frac{\pi}{2} \) is a very small value. Thus, we can simplify (3.11) as:
\[ \Delta d_{12} = r_{12} \left( A - S_{12} - \frac{\pi}{2} \right) \]  

(3.12)

Substituting (3.12) and \( s = 2 \) into (3.5c), we obtain:

\[ M^2(n) = A_1^2(n) + A_2^2(n) + 2A_1(n)A_2(n) \cos(\rho r_{12}(A - S_{12} - \frac{\pi}{2}) + \Delta \phi_{12}) \]  

(3.13)

Note that (3.13) holds only when \( \Delta d_{12} \) does not exceed the range resolution, which implies that:

\[ A - S_{12} - \frac{\pi}{2} < \frac{W}{N r_{12}} \]  

(3.14)

Generally speaking, \( \rho r_{12} \) is a big number (approximately 1122 if we employ the same HRR radar parameters as previously used and we assume \( r_{12} \) is 3 meters). Therefore, in some cases even a very small change of \( A \), for example from 0 to \( \pi(\rho r_{12}) \), could cause \( M(n) \) to change from \( |A_1(n)| + A_2(n)| \) to \( |A_1(n)| - A_2(n)| \). This result is in agreement with the analysis presented in [50]. However, an interesting phenomenon can be observed if we further increase \( A \), and \( M(n) \) stays between \( |A_1(n)| + A_2(n)| \) and \( |A_1(n)| - A_2(n)| \) due to the periodic property of the cosine function. That is, a huge change of \( A \) can have the exactly the same effect on the signature magnitudes as a tiny change of \( A \).

3.1.3 Analysis Conclusions

The observations obtained in the previous section suggest the following conclusions regarding the properties of complex HRR signatures, as well as their phase utility.

- The phase of complex HRR signatures has limited utility in identifying the properties of scatterers if they are utilized on the basis of a single signature. This
is because, compared with the magnitudes, the phases do not encode extra target knowledge, and is also harder to use for identifying target properties.

- Phase may provide extra knowledge regarding scatterer relative locations if they are utilized on the bases of a sequence of signatures over aspect angles. However, successfully extracting that knowledge requires very small aspect-angle intervals between these sequential signatures.

- The scintillation effect of the magnitude of HRR signatures may be due to more than one factor, such as the local vibration of scatterers, the change of the scattering coefficient of the scatterers, and the variance of the aspect angles.

Although, at the first glance, our first statement regarding phase utility is quite different from the results in [100], these two studies are not in conflict, for the following reasons. (1) The study in [100] is based on a wide-band radar system. For a wide-band radar system, the signal model includes a type-parameter in the scattering coefficient part of each scatterer. (Please refer to Equation 2.2 and Table 2.1 for detailed explanation on the type-parameter.) It is this type-parameter that the experiments in [100] demonstrated that the complex radar data is more efficient in estimating. However, the HRR radar system is assumed to be narrow-band, and therefore the type-parameter is usually negligible. A verification of this statement can also be obtained from Equation (4.13) and (4.18) in Chapter 4. (2) The experiments in [100] were based on estimation of model parameters based 2D radar data, while our phase-utility conclusions are drawn from observations regarding a single HRR signature, which is 1D data. Indeed we note that the second phase-utility observation calls attention to the fact that phase can provide extra physical information regarding the relative locations of scatterers when a sequence of HRR signatures are employed.
3.2 Phase-utility Experimental Results

In the previous section we analyzed the properties of complex HRR signatures based on a simple theoretical model, Equation (3.1). However, the credibility of those statements may be questioned because of the simplicity of the model. Therefore, in this section we examine the phase-utility of complex signatures from a totally different perspective – an experimental test of measured data.

We start from a set of measured HRR signatures [53], and design experiments to quantitatively evaluate the phase-utility of the complex signatures when they are utilized on the basis of a single signature.

Because it is difficult to design experiments directly reflecting the utility of phase for classification, we adopt an alternative technique, which reduces that difficult problem to a relatively easier problem of Feature Set Evaluation (FSE). That is, we form two Feature Sets (FS) in each experiment, one is the magnitude-only set, which is obtained by using only the magnitudes of the complex signatures, while the other is the complex set, which is obtained by using both the magnitude and the phase information of those signatures. After applying the same FSE methods on these two sets, we compare the experimental results to determine which set is more separable. If the complex set (magnitude-only) has better classification performance, we may reasonably conclude that the phase information does (not) contribute to the classification performance.

In order to avoid drawing classifier-dependent conclusions, two systematic different FSE methods are employed in our experiments. One is k-Nearest Neighbors (kNN)[84], and the other is Boundary Methods (BM) [126]. The output of the kNN is an estimation of
the Probability of classification error (Pe), while that of the BMs is a positive value called
the Overlap Sum (OS). The experimental data set generating smaller Pe or OS value is
supposed to have better separability.

In total, we designed and implemented three experiments, and they are presented in the
Sub-sections 3.2.1, 3.2.2, and 3.2.3 respectively. In Sub-section 3.2.4, a brief summary of
all experimental results is provided.

3.2.1 Experiment 1

3.2.1.1 Experimental Description

The design of this experiment is illustrated in Figure 3.1, which is valid for both the
magnitude-only and the complex HRR signatures. First, a feature extraction operation
(refer to Appendix A for details) is employed to reduce the data dimensionality of HRR
signatures. Because the HRR signatures are very sensitive to aspect angle, most of the
target characteristics manifested in the signatures remain stable only over a small aspect
angle range. Therefore, knowledge of target aspect angle is usually a prerequisite for target
identification using HRR signatures. Significant research has been and is being devoted to
providing reliable estimates of the target aspect angles [7]. In our experiment, we assume
that the target aspect angle has been estimated, and the maximal estimation error is ±5°.
Therefore, we divide all the signatures, including both training set and test set, into small
groups centering on a particular aspect angle and covering a 10° range. Then one group of
signatures from the training set and one from the test set are combined to form the input
data for kNN method. The output of the kNN method is the $P_e$ for the center aspect angle of the input signatures.

3.2.1.2 Experimental Results

The experimental results are illustrated in Figures 3.2 and 3.3. Each figure has different value of $k$, which is the parameter used in the kNN method to define the number of neighbors involved in declaring a class. In this experiment, we selected four different targets, which are T72, BMP2, 2S1, and BRDM2. Each plot in the figure displays the resultant $P_e$ of one target-pair for two of the four selected targets over aspect angles.
Figure 3.1 Experimental design of Experiment 1
Figure 3.2 Experiment 1 results: obtained when k=3
3.2.2 Experiment 2

3.2.2.1 Experimental Description

The design of Experiment 2 is illustrated in Figure 3.3. In this experiment BMs are employed as the analysis tool. Because BMs do not require the discrimination of training set and test set, we only utilized the test set here. The outputs of the BMs are OS's.

3.2.2.2 Experiment Results

The six plots in Figure 3.4 show the results of Experiment 2. Each plot illustrates the OS of one target pair for any two of the four selected targets over aspect angles. They four selected targets used in this experiment are T72, BMP2, 2S1, and BRDM2. Note that smaller OS values imply better separability of the two corresponding feature sets.
Figure 3.3 Experimental Design of Experiment 2
Figure 3.4 Experiment 2 results
3.2.3 Experiment 3

3.2.3.1 Experimental Description

To avoid the possibility that the feature extraction process may destroy useful phase information contained in the complex signatures, we remove that process and directly apply the whole signatures to kNN method in Experiment 3. The design of Experiment 3 is illustrated in Figure 3.5. This experiment is almost the same as experiment 1, except that the feature extraction process is removed.

![Figure 3.5 Experimental Design of Experiment 3](image-url)
Figure 3.6 Experiment 3 results: obtained when k=3
3.2.3.2 Experiment Results

The experimental results are presented here in the same form as those in Experiment 1. Note that the results show that the complex signatures demonstrate much worse classification performance than the magnitude-only ones. This extraordinary performance difference is probably because of the joint effects of the randomness of the phases of complex signatures and the limitation of Euclidean distance metric utilized in the kNN method.

3.2.4 Experimental Summary

A brief summary of the three phase-utility experiments are listed in Table 3.1.

<table>
<thead>
<tr>
<th>Experiment Index</th>
<th>Experiment Data</th>
<th>FSE Method</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>Feature Vector</td>
<td>KNN</td>
<td>Magnitude-only feature set slightly better than the complex feature set</td>
</tr>
<tr>
<td>#2</td>
<td>Feature Vector</td>
<td>BMs</td>
<td>Magnitude-only feature set roughly comparable to the complex feature set</td>
</tr>
<tr>
<td>#3</td>
<td>Feature Vector</td>
<td>KNN</td>
<td>Magnitude-only profiles better than the complex profiles</td>
</tr>
</tbody>
</table>

Table 3.1 Summary of phase-utility experiments

In sum, these experimental results are consistent regarding the phase-utility of complex HRR signatures, and consistent with our theoretical analysis in Section 3.1. That is, the phase information contained in complex HRR signatures, when utilized on the basis of a
single signature, is either very hard to extract or has only limited utility in target
classification in the single-look, stationary-target scenario.
CHAPTER 4

HRR DATA MODELING AND FEATURE EXTRACTION ALGORITHMS

As we mentioned in Chapter 1, 1D moving target HRR signatures are generated from 2D raw HRR data, and are believed to be good representations of the characteristics of the underlying targets. Therefore, most of the current HRR-based ATR research has been conducted directly using these 1D HRR signatures [7, 28, 32, 44, 57, 58, 65, 66, 88]. Nevertheless, the resultant algorithms are degraded by the variable behavior of HRR signatures. One promising solution is to extract the target information directly from the 2D raw HRR data, and avoid using the (sensitive) 1D HRR signature as target features [2]. The method presented in this Chapter is one step in this direction.

Currently, a common methodology is widely employed in HRR data process, including feature extraction and clutter suppression. The HRR data is assumed to abide by a physics-based HRR data model, and subsequent data processing algorithms are developed based on this model [2, 5, 37, 48, 49, 50], rather than on the properties of measured data. A review of the relevant literature can be found in Section 2.2 of Chapter 2. This methodology enables researchers to convert the complex non-parametric data into a simple parametric form, and to easily incorporate prior knowledge of the underlying targets. Obviously, a physics-based
A radar model is the prerequisite for this category of research. Devising a reliable 2D HRR model for moving targets has thus become an increasingly important topic.

A set of physics-based models for rigid body and non-rigid body moving targets has been previously presented in [2] and [5] respectively. A modified version of a moving target model was presented in [37], with an assumption that the chirp rate $\gamma$ is reasonably small, which is, however, not the general case. Moreover, the non-rigid body moving target model presented in [5] is simplified by assuming that the radar and target parameters satisfy certain conditions. However, those assumptions seem to be violated in practical system, because unrealistic values may be obtained when applying a set of realistic radar parameters to the model. Furthermore, classifying targets as rigid body and non-rigid body makes application of the model difficult because there is no a clear dividing line between the rigid body and non-rigid body targets. Also, distinctions can become blurred in very common application scenarios, such as in the case where both rigid body and non-rigid body targets appear at the same time, or in the case where a rigid body target includes a rotating part such as a wheel or track.

In this chapter we derive and simplify a new scattering-based HRR moving target model. The contributions of this new model are as follows:

- We provide two simplified HRR moving-target models with different degrees of simplification, and furnish some rudimentary checks on the validity of our simplifications.
- The proposed models are valid for both rigid and non-rigid body targets.
- These models make no assumptions regarding the distribution of the clutter.
Performance bounds on the subsequent feature extraction algorithms are naturally obtained during the simplification procedure.

Although the proposed model is derived with the assumption that the HRR radar operates in X frequent band, the same methodology employed here can be easily applied to other radar system and a different operational frequency band.

After a physics-based model is readily available, the subsequent feature extraction and clutter suppression are reduced to a model parameter estimation problem. Therefore, we also derive two parameter estimation algorithms to achieve this goal in this chapter.

In Section 4.1 we first derive a moving target model, and then simplify this model by instantiating a set of real-world radar and target parameters to the model in Sections 4.2 and 4.3. Two feasible model parameter estimation algorithms are derived and proposed in Sections 4.4 and 4.5 respectively, and experiments and performance comparison are presented in Section 4.6.

Before jumping into the details, we would like to first explain the difference between HRR radar and SAR radar to highlight the significance of our results, given the similarities in the formulation of our HRR models and some available SAR models.

The main differences between HRR and SAR radar lie in both hardware and software involved in their realization. As for the aspect of hardware implementation, SAR radar generally has much wider bandwidth than its HRR counterpart, and therefore can capture more target information, such as the type-parameters of scatterers. In contrast, the PRI of HRR is much shorter than that of SAR, and thus HRR can capture the target motions. With regard to processing, SAR and HRR raw data are processed differently. Because a SAR image is usually generated from raw SAR data, there are a number of refinement steps
involved in forming a high quality SAR image. Also, all SAR-based algorithms use SAR data as 2D data. However, current widely used HRR data is HRR signatures, which are 1D data, instead of the 2D HRR data. (Please refer to Subsection 1.1.4 in Chapter 1 for a brief expression of the generation of HRR signatures.)

Therefore, one of the contributions of our research is to approach HRR data differently -- extracting features directly from the 2D raw data, instead of from the 1D HRR signatures. Another motivation is that conventional processes involved in generating HRR signatures use FFTs, which is not a high-resolution technique. Therefore, our research suggests a model-based estimation algorithm to improve the resolution of scatterer-extraction.

4.1 Derivation of A Physics-based HRR Moving Target Model

To facilitate readers’ understanding we restate portions of the derivations contained in [5, 37]. Our model derivation is based on four basic assumptions:

- The radar returns can be approximated by a sum of the scattered signals from a group of discrete scatterers if the radar operates at sufficiently high frequencies [123].
- The scatterers on targets and the scatterers arising from clutter are difficult to separate by their scattering property [39].
- Targets are moving and the clutter is stationary [2,5,37].
• The radar angle is fixed at 90 degree to the platform-moving trajectory. This assumption ensures that the Doppler frequencies of the stationary clutter cluster around 0 Hz [1].

Radars obtain target information by transmitting an electromagnetic (EM) field toward interesting targets and by acquiring the backscattered EM field, which encodes information about the targets. The transmission antennas of radars are employed to transform the electronic radar signals, or radar pulses, into EM waves. It is the EM waves that propagate through the free space, and are scattered backwards when they impinge on materials with different propagation properties. The receiving antennas of radars are utilized to convert the received EM waves back into electronic signals. Because our research only focuses on the information stored in the electronic signal form, we choose to ignore the processes of converting the electronic signals to and from EM waves, and we directly adopt the radar signal representation to describe the whole transmission-scattering-receiving procedure. This liberates our formulation from utilizing complex EM field description, and greatly simplifies our subsequent analysis and processing. Of course, this simplification is achieved at the cost of losing some potentially useful physical properties manifested only in the EM waves, such as the polarization diversity. For those who have interests in these physical properties, please refer to [12, 13, 14, 39, 60, 69, 70, 73]

The radar signal is a chirp pulse, which is denoted by \( \bar{s}(t) \)

\[
\bar{s}(t) = \begin{cases} 
\exp \{ j(\omega_0 t + \pi^2) \}, & |t| \leq T_0 / 2 \\
0, & |t| > T_0 / 2 
\end{cases}
\]

(4.1)
where $T_a$ is the pulse width, $\omega_0$ is the center carrier frequency, and $2\gamma$ is the chirp rate of the linear FM pulse. During a dwell time, a sequence of radar pulses are transmitted by an HRR radar. The sequence of $L$ pulses can be expressed by $s_i(t)$:

$$s_i(t) = \sum_{l=0}^{L-1} s(t - lT),$$

(4.2)

where $T$ is the time interval between pulses, and $L$ is the number of pulses sent each time.

The scattered signal from $k$th scatterer is denoted by $s_{bk}(t)$,

$$s_{bk}(t) = \sum_{l=0}^{L-1} a_k(t) \tilde{s}_{bk}(t - lT),$$

(4.3)

where $a_k(t)$ is the $k$th scatterer's scattering coefficient determined by the frequency.

Because the radar signal is a linear FM waveform, the relationship between time and frequency can be represented as:

$$\omega = \omega_0 + 2\gamma t, \text{ where } |t| < T_0 / 2.$$  

(4.4)

According to the GTD [26, 38, 62, 123], the frequency dependence of the scatterer's amplitude has the form of:

$$\bar{a}_k(\omega) = A_k \left(\frac{\omega}{\omega_0}\right)^{\alpha_k}$$

(4.5)

Therefore $a_k(t)$ can be expressed in the form of:

$$a_k(t) = A_k \left[1 + \frac{2\gamma}{\omega_0} (t - lT)\right]^{\alpha_k}$$

(4.6)

where the complex scalar $A_k = \rho_k \exp(j \psi_k)$. The parameter $\alpha_k$ is a dimensionless, discrete type parameter characterizing the geometry [62] of the $k$th scatterer. Please refer to Table 2.1 for the correspondence between the value of $\alpha$ and the geometry of the scatterer.
Because the dwell time of HRR radar is usually very short, it is impossible for the aspect angle of the target to change significantly during one dwell period of time. Thus, it is reasonable to ignore the effects of aspect angle on the scatterer amplitude $a_k(t)$.

One of the scattered pulses from $k$th scatterer is denoted:

$$\tilde{s}_{k}(t - lT) = \exp \left( j \omega_0 (t - lT - \frac{2R_k(t)}{c}) + j \gamma(t - lT - \frac{2R_k(t)}{c})^2 \right).$$  \hspace{1cm} (4.7)

We denote the direction of radar line-of-sight (LOS) as $r_0 = [\cos \varphi \cos \theta, \cos \varphi \sin \theta, \sin \varphi]^T$, where $\theta$ is the aspect angle, and $\varphi$ is the depression angle of the radar to the targets, which is illustrated by Figure 4.1.

![Figure 4.1 Coordinate System](image)

If we assume the $k$th scatterer is at ground location of $L_k$ at $t = 0$, and it is moving at a speed of $V_k$, its relative location to the radar at time $t$, or $R_k(t)$, is
\[ R_{nk}(t) = R_0 - L_k^T r_0 - V_k^T r_0 t. \] (4.8)

Substituting (4.8) into (4.7), we obtain

\[ \tilde{s}_{bk}(t-lT) = \exp \left\{ j \omega_0 (t-lT) - t_0 + t_k + \frac{2V_k^T r_0}{c} + j\gamma(t-lT) - t_0 + t_k + \frac{2V_k^T r_0}{c} \right\} \]

\[ = \exp \left\{ j \omega_0 (\hat{t} + t_k) + j\gamma(\hat{t} + t_k)^2 \right\} \exp \left\{ j \omega_{dk} \hat{t} \right\} \exp \left\{ j 2\gamma(\hat{t} + t_k) C_{vk} \hat{t} \right\} \exp \left\{ j\gamma(C_{vk} t)^2 \right\} \] (4.9)

where \( t_0 = \frac{2R_0}{c}, t_k = \frac{2L_k^T r_0}{c}, \omega_{dk} = \omega_0 \frac{2V_k^T r_0}{c}, C_{vk} = \frac{2V_k^T r_0}{c} \), and \( \hat{t} = t - t_0 - lT \).

We mix the scattered pulse \( \tilde{s}_{bk}(t-lT) \) with a delayed and conjugated transmitted pulse \( \tilde{s}^*(t-t_0-lT) \) to obtain the de-chirped scattered pulse \( \tilde{y}_k(t-t_0-lT) \) of the \( k \)th scatterer:

\[ \tilde{y}_k(t-t_0-lT) = \tilde{s}_{bk}(t-lT)\tilde{s}^*(t-t_0-lT) \]

\[ = \exp \left\{ j \omega_0 t_k + j\gamma t_k^2 \right\} \exp \left\{ j 2\gamma \hat{t} \right\} \exp \left\{ j \omega_{dk} \hat{t} \right\} \exp \left\{ j 2\gamma(\hat{t} + t_k) C_{vk} \hat{t} \right\} \exp \left\{ j\gamma(C_{vk} t)^2 \right\}. \] (4.10)

If we replace \( t = \hat{t} + t_0 + lT \) into (4.10), we obtain

\[ \tilde{y}_k(t-t_0-lT) = \exp \left\{ j[\omega_0 t_k + \gamma t_k^2 + \omega_{dk} t_0 + 2\gamma C_{vk} t_k t_0 + \gamma(C_{vk} t_0)^2] \right\} \]

\[ \cdot \exp \{j[2\gamma t_k + \omega_{dk} + 2\gamma C_{vk} (t_0 + t_k) + 2\gamma C_{vk}^2 t_0]^2 t \} \]

\[ \cdot \exp \{j[\omega_{dk} + 2\gamma C_{vk} t_k + 2\gamma C_{vk}^2 t_0] T \} \]

\[ \cdot \exp \{j[2\gamma C_{vk} + 2\gamma C_{vk}^2] T \hat{t} \} \]

\[ \cdot \exp \{j[2\gamma C_{vk} + \gamma C_{vk}^2] \hat{t}^2 \} \]

\[ \cdot \exp \{j\gamma C_{vk}^2 T^2 \hat{t}^2 \}. \] (4.11)

We can discretize \( \tilde{y}_k(t-t_0-lT) \) by defining \( \hat{t} = T_n \), and rewrite \( \tilde{y}_k(t-t_0-lT) \) as a 2-D discrete function \( \tilde{D}_k(n,l) : \)
\[ \bar{D}_k(n,l) = \exp \{j\phi_k\} \exp \{j\bar{\omega}_k n\} \exp \{j\bar{\omega}_{nk} l\} \exp \{j\bar{\omega}_{mk} n^2\} \exp \{j\bar{\omega}_{2nk} l^2\} \]  

(4.12a)

where

\[ \phi_k = \omega_0 t_k + \gamma t_k^2 + \omega_{\Delta k} t_0 + 2\gamma C_{\Delta k} t_0 + \gamma (C_{\Delta k} t_0)^2, \]  

(4.12b)

\[ \bar{\omega}_k = [2\gamma t_k + \omega_{\Delta k} + 2\gamma C_{\Delta k} (t_0 + t_k) + 2\gamma C_{\Delta k}^2 t_0] T_s, \]  

(4.12c)

\[ \bar{\omega}_{nk} = [\omega_{\Delta k} + 2\gamma C_{\Delta k} t_k + 2\gamma C_{\Delta k}^2 t_0] T_s, \]  

(4.12d)

\[ \bar{\omega}_{nk} = [2\gamma C_{\Delta k} + 2\gamma C_{\Delta k}^2] T_s, \]  

(4.12e)

\[ \bar{\omega}_{2nk} = [2\gamma C_{\Delta k} + \gamma C_{\Delta k}^2] T_s, \]  

(4.12f)

\[ \bar{\omega}_{2nk} = \gamma C_{\Delta k} T_s^2. \]  

(4.12g)

Accordingly, \( a_k(t) \) can be discretized as:

\[ a_k(n) = A_k (1 + \frac{2\gamma}{\omega_0} T_s n)^{a_k}, \]  

(4.13)

When the radar operates at a sufficiently high frequency, the radar returns can be approximated by a sum of scattered signals from a group of discrete scatterers [123]. Also, additive noise, \( e(n,l) \), can be introduced to represent the totality of effects of many unpredictable nuisance factors, such as measurement errors and interference in the radar system, variance of the propagation media, etc. Therefore, the moving target model in 2D discrete form is obtained.

\[ D(n,l) = \sum_{k=1}^{K} a_k(n) \bar{D}_k(n,l) + e(n,l), \quad n = \frac{N}{2} \ldots 0 \ldots \frac{N}{2}; l = 0 \ldots L - 1, \]  

(4.14)
where $K$ is the number of “visible” scatterers\(^1\) within the illuminated area.

### 4.2 Parameters of the Radar System and the Target

We will utilize the radar system parameter set based on the Global Hawk and SEP systems [1] in our following simplification of the proposed moving target model (4.14). The parameters are listed as follows:

- The radar is operating in the HRR mode.
- The radar signal consists of linear FM chirp pulses at X-band, which implies that the carrier center frequency is 9.4 GHz, or $f_0 = 9.4 \times 10^9 \text{ Hz}$;
- Each chirp pulse has a bandwidth of 648 MHz, or $B = 648 \times 10^6 \text{ Hz}$;
- The width of each pulse is 50μs, or $T_\text{p} = 50 \times 10^{-6} \text{ s}$;
- There are 128 pulses in each burst transmission of the radar signal, or $L = 128$.
- The propagation speed of the radar signal is the speed of light, or $c = 3 \times 10^8 \text{ m/s}$;
- The radar platform is about 90km away from the targets, or $R_0 = 90 \times 10^3 \text{ m}$;
- The depression angle of the radar system is 15° or 17°;
- The radar platform is moving with a horizontal velocity of 180m/s, or $V_{ph} = 180 \text{ m/s}$;

\(^1\) Generally speaking, “visible” scatterers are those scatterers with relatively larger magnitudes. A more precise definition depends heavily on the particular feature extraction algorithm employed.
• The sampling frequency of the radar A/D converter is 50MHz, or \( f_s = 50 \times 10^6 \, \text{Hz} \). That is the sampling time interval is 0.02\( \mu \text{s} \), or \( T_s = 0.02 \times 10^{-6} \, \text{s} \);

• The minimum detectable target radial speed is 2.3 m/s, or \( V_{r_{\text{min}}} = 2.3 \, \text{m/s} \), which is determined by the main-lobe-width of the radar antenna.

• Given that the maximum detectable target ground speed is 198 km/hour, or about 124 mile/hour, the maximum target radial speed is about \( V_{r_{\text{max}}} = 53 \, \text{m/s} \).

This value also determines the selection of the interval time \( T \) of two pulses;

• The total length of the range swatch is 500m, or \( L_{\text{max}} = 500 \, \text{m} \);

From the above basic parameter set, we can further obtain the following parameters.

• The chirp rate \( \gamma \) of the chirp signal is \( 4.0715 \times 10^{13} \, \text{rad/s} \), from \( \gamma = \frac{2\pi B}{2T_0} \);

• The minimum Doppler frequency: \( \omega_{D_{\text{min}}} \) is 905.61 rad, from \( \omega_{D_{\text{min}}} = \frac{4\pi V_{r_{\text{min}}} f_0}{c} \), while the maximum Doppler frequency: \( \omega_{D_{\text{max}}} \) is 20869 rad, from \( \omega_{D_{\text{max}}} = \frac{4\pi V_{r_{\text{max}}} f_0}{c} \);

• The theoretical range resolution: \( r_c \) is about 0.23 m, from \( r_c = \frac{c}{2B} \). However, the actual resolution is usually about 0.30m, or 1 foot.

• The minimum time difference between two resolvable scatterers is 2\( \mu \text{s} \), from \( t_{\text{m}_{\text{min}}} = \frac{2r_c}{c} \), while the maximum time difference is 1.67\( \mu \text{s} \), from \( t_{\text{m}_{\text{max}}} = \frac{L_{\text{max}}}{c} \);
• The time interval of radar pulses is 150μs, from \( T = \frac{\pi}{\omega_{p_{\text{max}}}} \).

4.3 Model Simplification

Note that model (4.14) consists of 5 frequency components, and each frequency component is further compiled of several terms. Our basic idea is to instantiate the values of previously defined parameters to the model, and observe that the value of each frequency component is dominated by only a small number of terms. Therefore, the original model can be simplified by retaining only those terms with dominating values.

According to (4.12c), frequency \( \bar{\omega}_k \) is compiled of terms in the following table:

<table>
<thead>
<tr>
<th>Term</th>
<th>( 2\gamma_k T_s )</th>
<th>( \omega_D t_s )</th>
<th>( 2\gamma C \omega_k (t_0 + t_k) T_s )</th>
<th>( 2\gamma C \omega^2 t_0 t_s )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. Value (rad)</td>
<td>2.7143</td>
<td>4.1737 \times 10^{-2}</td>
<td>3.4622 \times 10^{-2}</td>
<td>1.2199 \times 10^{-9}</td>
</tr>
<tr>
<td>Min. Value (rad)</td>
<td>3.2572 \times 10^{-3}</td>
<td>1.8112 \times 10^{-3}</td>
<td>1.4983 \times 10^{-3}</td>
<td>2.2974 \times 10^{-12}</td>
</tr>
<tr>
<td>Keep this term?</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

Table 4.1 Simplify \( \bar{\omega}_k \)
Similarly, according to (4.12d), $\omega_{D_k}$ is compiled of terms in the following table:

<table>
<thead>
<tr>
<th>Term</th>
<th>$\omega_{D_k} T$</th>
<th>$2\gamma C_{vk} T_k T$</th>
<th>$2\gamma C_{vk}^2 t_0 T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. Value (rad)</td>
<td>3.1303</td>
<td>$7.1930 \times 10^{-3}$</td>
<td>$9.1495 \times 10^{-3}$</td>
</tr>
<tr>
<td>Min. Value (rad)</td>
<td>0.1358</td>
<td>$3.7458 \times 10^{-3}$</td>
<td>$1.7231 \times 10^{-3}$</td>
</tr>
<tr>
<td>Keep this term?</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

Table 4.2 Simplify $\omega_{D_k}$

According to (4.12e), $\omega_{r_k}$ is compiled of terms in the following table:

<table>
<thead>
<tr>
<th>Term</th>
<th>$2\gamma C_{vk} T T_k$</th>
<th>$2\gamma C_{vk}^2 T T_k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. Value (rad)</td>
<td>$8.6316 \times 10^{-5}$</td>
<td>$3.0498 \times 10^{-11}$</td>
</tr>
<tr>
<td>Min. Value (rad)</td>
<td>$3.7458 \times 10^{-5}$</td>
<td>$5.7435 \times 10^{-12}$</td>
</tr>
<tr>
<td>Keep this term?</td>
<td>Depends...</td>
<td>No</td>
</tr>
</tbody>
</table>

Table 4.3 Simplify $\omega_{r_k}$

According to (4.12f), $\omega_{\theta_k}$ is compiled of terms in the following table:

<table>
<thead>
<tr>
<th>Term</th>
<th>$2\gamma C_{vk} T^2$</th>
<th>$\gamma C_{vk}^2 T^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. Value (rad)</td>
<td>$1.1509 \times 10^{-8}$</td>
<td>$2.0332 \times 10^{-15}$</td>
</tr>
<tr>
<td>Min. Value (rad)</td>
<td>$4.9944 \times 10^{-10}$</td>
<td>$3.8290 \times 10^{-18}$</td>
</tr>
<tr>
<td>Keep this term?</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

Table 4.4 Simplify $\omega_{\theta_k}$

80
Finally, according to (4.12g), $\bar{\omega}_{2Dk}$ is compiled of a term in the following table:

<table>
<thead>
<tr>
<th>Term</th>
<th>$\gamma C_{\nu k} T^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. Value (rad)</td>
<td>$1.1437 \times 10^{-7}$</td>
</tr>
<tr>
<td>Min. Value (rad)</td>
<td>$2.1538 \times 10^{-10}$</td>
</tr>
<tr>
<td>Keep this term?</td>
<td>No</td>
</tr>
</tbody>
</table>

Table 4.5 Simplify $\bar{\omega}_{2Dk}$

Eq. (4.12a) can be simplified by deleting all the negligible terms in the above 5 tables.

Thus, a simplified $\tilde{D}_{k}(n,l)$ can be represented as follows:

$$\tilde{D}_{k}(n,l) = \exp\{j\phi_{k}\} \exp\{j\bar{\omega}_{k} n\} \exp\{j\bar{\omega}_{2Dk} l\} \exp\{j\bar{\omega}_{rk} nl\}$$

(4.15)

where $\bar{\omega}_{k} = 2\gamma_{k} T_{s}$, $\bar{\omega}_{2Dk} = \omega_{2Dk} T$, and $\bar{\omega}_{rk} = 2\gamma_{C\nu k} TT_{s}$.

If we retain only those terms with dominating values, we can further simplify (4.15) as:

$$\tilde{D}_{k}(n,l) = \exp\{j\phi_{k}\} \exp\{j\bar{\omega}_{k} n\} \exp\{j\bar{\omega}_{2Dk} l\}$$

(4.16)

where $\bar{\omega}_{k} = 2\gamma_{k} T_{s}$, and $\bar{\omega}_{2Dk} = \omega_{2Dk} T$

Substituting (4.13) and (4.16) into (4.14), A simplified version of model (4.14) can be obtained as follows:

$$D(n,l) = \sum_{k=1}^{K} a_{k} (n - N / 2) \exp\{j\phi_{k}\} \exp\{j(\bar{\omega}_{k} n - N / 2)\} \exp\{j\bar{\omega}_{2Dk} l\} + e(n,l)$$

$$= \sum_{k=1}^{K} \rho_{k} \exp\{j\bar{\phi}_{k}\} (1 - \frac{\gamma_{T_{k}} N}{\omega_{0}} + \frac{2\gamma_{T_{k}} T_{s} n}{\omega_{0}}) a_{k} \exp\{j\bar{\omega}_{k} n\} \exp\{j\bar{\omega}_{2Dk} l\} + e(n,l)$$

(4.17)

where $n = 0 \ldots N; l = 0 \ldots L - 1$, and $\bar{\phi}_{k} = \phi_{k} + \phi - \bar{\omega}_{k} N / 2$. 

81
Furthermore, we observe that \( \frac{2\pi}{\omega_0} \) is as small as \( 5.518 \times 10^{-5} \) by instantiating the values of the radar and target parameters to it. Therefore, if the total number of range bins, \( N \), is not too large, (4.13) can be further simplified as:

\[
a_k(n) = A_k
\]  

That is, the effect of the type parameter, \( \alpha_k \), is negligible.

Substituting (4.18) in (4.17), we obtain a very simplified model:

\[
D(n,l) = \sum_{k=1}^{K} \rho_k \exp \{j\phi_k\} \exp \{j\overline{\omega}_k n\} \exp \{j\overline{\omega}_m l\} + e(n,l),  
\]  

where \( n = 0 \ldots N; l = 0 \ldots L - 1 \), and \( \phi_k = \psi_k + \phi_k - \overline{\omega}_k N / 2 \).

From (4.12b) and (4.17), we know that the initial phase term \( \phi_k \) is affected by many subtle factors, such as the surface material, structure, range position, radial speed, and propagation channel of the \( k \)th scatterer. Any small change in any of these factors will probably lead to a big difference in \( \phi_k \). Therefore, it is reasonable for us to consider \( \phi_k \) as random variable. Here, we assume that \( \phi_k \) is a uniform-distributed random variable over \([0, 2\pi]\).

In addition, Table 4.1 shows that that the minimum value of \( \overline{\omega}_k \) is \( 0.003257 \) rad., which is determined by the resolution of the HRR radar system at X-band. This value can be considered as a criterion for the performance of subsequent feature extraction algorithms applied to this model. That is, the estimation errors of subsequent feature extraction algorithms based on this model should be less than this minimum value if we wish to extract all available target information encoded in the radar data.
4.4 Matrix Form of the HRR Data Model and Its 2D Auto-correlation Matrix

With regard to (4.19), if we define

\[ \Omega_k = [1 \ \exp\{j\bar{\omega}_k\} \ \cdots \ \exp\{j\bar{\omega}_kN\}]^T \]

\[ E = [\Omega_1 \ \Omega_2 \ \cdots \ \Omega_K] \],

\[ \Omega_{\delta k} = [1 \ \exp\{j\bar{\omega}_{\delta k}\} \ \cdots \ \exp\{j\bar{\omega}_{\delta k}(L-1)\}]^T \],

\[ E_D = [\Omega_{\delta 1} \ \Omega_{\delta 2} \ \cdots \ \Omega_{\delta K}] \],

\[ \Lambda = \begin{bmatrix} \rho_1 \exp\{j\bar{\phi}_1\} \\ \rho_2 \exp\{j\bar{\phi}_2\} \\ \vdots \\ \rho_K \exp\{j\bar{\phi}_K\} \end{bmatrix} \], \text{and}

\[ W = \begin{bmatrix} e(0,0) & \cdots & e(0,L-1) \\ \vdots & \ddots & \vdots \\ e(N,0) & \cdots & e(N,L-1) \end{bmatrix} \].

Model (4.19) can then be expressed simply in matrix form as:

\[ D = E \Lambda E_D^T + W . \] (4.20)

Let us define the 2D auto-correlation matrix \( R \) as

\[ R = \begin{bmatrix} R(0,0) & \cdots & R(0,L-1) \\ \vdots & \ddots & \vdots \\ R(N-1,0) & \cdots & R(N-1,L-1) \end{bmatrix} , \] (4.21a)

where

\[ R(m_1,m_2) = E\{D(n+m_1,l+m_2)D^*(n,l)\} , \]

\[ m_1 = 0,1,\cdots,(N-1) ; m_2 = 0,1,\cdots,(L-1) , \text{ and} \] (4.21b)
$E\{\cdot\}$ denotes the expectation of a random variable.

By substituting (4.19) into (4.21b), we obtain

$$R(m_1, m_2) = E\left\{ \sum_{k=1}^{\infty} \sum_{c=1}^{\infty} a_k a_c \exp \{ j[\bar{\omega}_b(n + m_1) - \bar{w}c n + \bar{\omega}_{D_0}(l + m_2) - \bar{\omega}_{D_0} l + (\phi_b - \phi_c)] \} \right\}$$

$$+ E\{e(n + m_1, l + m_2)e^\ast(n, l)\}$$

$$= \sum_{k=1}^{\infty} a_k^2 \exp \{ j[\bar{\omega}_b m_1 + \bar{\omega}_{D_0} m_2] \} + \sigma^2 \delta_{m_1, m_2},$$

$$m_1 = 0, 1, \cdots, (N-1); m_2 = 0, 1, \cdots, (L-1), \quad (4.22)$$

where $\sigma^2$ is the variance of additive white noise $e(n, l)$.

Therefore, the 2D auto-correlation matrix $R$ is:

$$R = E\Sigma E^\ast_\Theta + \Theta \quad (4.23)$$

where

$$\Sigma = \begin{bmatrix} a_1^2 & a_2^2 & \cdots & a_K^2 \\ a_2^2 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ a_K^2 & \cdots & \cdots & a_1^2 \end{bmatrix}, \quad \text{and} \quad \Theta = \begin{bmatrix} \sigma^2 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}.$$ 

In practice, the estimation of the 2D auto-correlation matrix, $\hat{R}$, is defined as:

$$\hat{R} = \begin{bmatrix} \hat{R}(0,0) & \cdots & \hat{R}(0,L-1) \\ \vdots & \ddots & \vdots \\ \hat{R}(N-1,0) & \cdots & \hat{R}(N-1,L-1) \end{bmatrix}, \quad (4.24a)$$

where

$$\hat{R}(m_1, m_2) = \frac{1}{NL} \sum_{n=0}^{N-m_1-1} \sum_{l=0}^{L-m_2-1} D(n + m_1, l + m_2)D^\ast(n, l). \quad (4.24b)$$
4.5 Parameter Estimation

From the model (4.19) we know there are three parameters, \( \{ \rho_k, \bar{\omega}_k, \bar{\omega}_{Dk} \} \), that need estimation for each scatterer. The whole set of estimable parameters of the model is thus denoted as:

\[
P = \{ \rho_k, \bar{\omega}_k, \bar{\omega}_{Dk} \}_{k=1}^K.
\]  

(4.25)

We name frequency \( \bar{\omega}_{Dk} \) the speed frequency of the \( k \)th scatterer. This is because \( \bar{\omega}_{Dk} \) is proportional to the Doppler frequency of the \( k \)th scatterer and is therefore proportional to the radial speed of the \( k \)th scatterer. Here we assume that the targets are moving while the clutter is stationary [51]. Also, the radar angle is supposed to be perpendicular to the platform-flying trajectory. All these assumptions ensure that the Doppler shift frequency of the clutter will be close to zero. As a result, we can discriminate whether a scatterer is a target-scatterer or a clutter-scatterer via the value of \( \bar{\omega}_{Dk} \). That is, scatterers with sufficient small \( \bar{\omega}_{Dk} \) are considered as the clutter-scatterers, while the others can be considered as target-scatterers. Meanwhile, the parameters defined in Eq. (4.25) capture the major information of the underlying targets, and can be a set of promising representative features. Consequently, the feature extraction and the clutter suppression process of HRR signals are reduced to a single fundamental problem — devising reliable parameter estimation algorithms based on the model.

In the subsequent parts of this section we derive and present two feasible parameter estimation algorithms for this model. The first algorithm reduces the 2D-estimation problem to two 1D-estimation problems, and solves the problems by employing some
mature 1D-estimation techniques. The second algorithm utilizes the 2D Discrete Fourier Transform (DFT) to estimate the model parameters.

Indeed, parameter estimation for this model is a well-studied topic. More than 4 different algorithms have been proposed to deal with this problem. A brief review of these algorithms was provided in Section 2 of Chapter 2 of this proposal.

4.5.1 Scatterer Extraction Using 1D Parameter Estimation (1DPE)

Note that the model Eq. (4.19) is a 2D function in a very special form, which can be easily rewritten in a 1D function form. This property naturally suggests the use of a straightforward parameter estimation algorithm, 1DPE. The basic idea of this 1DPE algorithm is (1) to reduce the original scatterer-detection and parameter-estimation problem to two 1D sub-problems, and then (2) to combine the two 1D estimation results to obtain the parameters of the scatterers encoded in the 2D data.

4.5.1.1 1D Scatterer Detection and Parameter Estimation

The model can be easily re-written as a 1D function if we fix one dimension, for example, the dimension of variable $l$.

$$D_l(n) = D(n, l) = \sum_{k=1}^{K} \beta_k(l) \exp\{j\omega_k n\} + e_i(n), \quad (4.26)$$

where $\beta_k(l) = \rho_k \exp\{j\omega_k l + j\phi_k\}$, and $e_i(n) = e(l, n)$.

Although it is possible that more than one scatterer will have the same position frequency $\omega_k$, we will delay consideration of this case until later. To simplify our current
derivation, we first assume that all of the *position frequency* components, \( \{ \omega_k \}_{k=1}^K \), are unique.

In fact Eq. (4.26) is a very typical and intensively studied 1D signal model if all the frequency components \( \{ \omega_k \}_{k=1}^K \) are unique [80]. Many mature parameter estimation algorithms, such as Nonlinear Least Square (NLS), the High-order Yule-walker algorithm (HOYW) [81], MUSIC [82], and ESPRIT [83], address this model and are readily available. Here we adopt the ESPRIT algorithm in our 1DPE for its robustness and accuracy with moderate SNR.

However, before we employ the ESPRIT algorithm, we must first estimate the model order \( K \) in (4.26), because the ESPRIT algorithm requires the model order \( K \) as an input parameter. Fortunately, there are also quite a few model order estimation algorithms applicable to this model (4.26). We select the Minimum Description Length (MDL) algorithm to implement our initial estimation of model order \( K \). Unfortunately, experimental results show that the output of MDL, \( \hat{K}_1 \), is not always reliable, and is bigger than the actual model order \( K \) in quite a few cases. In order to make our current derivation simple we choose to initially tolerate this estimation error and solve it in a latter stage of the algorithm.

Thus we can directly apply the result of the MDL algorithm, \( \hat{K}_1 \), to the ESPRIT algorithm. The output of the ESPRIT algorithm is a set of estimated parameters of (4.8), which are the \( \hat{K}_1, \text{ position frequency} \) components.

\[
S = \{ \hat{\omega}_n, \hat{\beta}_n (l) \}_{l=1}^{\hat{K}_1},
\]  

(4.27)
where \( \{ l_1, l_2, \ldots, l_{\hat{K}} \} \) is a permutation of the number set of \( \{ 1, 2, \ldots, \hat{K} \} \).

Because \( \hat{K} \) is not necessarily equal to \( K \), some spurious frequency components will be introduced in the estimation results. We remove these spurious components according to their magnitudes. The magnitude of each frequency component is defined as \( \hat{a}_k = |\hat{\beta}_k(l)| \).

It is reasonable to consider the components with very small magnitudes as spurious frequency components. Therefore, we can use an adaptive threshold, \( Th \), to remove the spurious frequency components. That is, we consider all the frequency components whose magnitudes are smaller than the threshold, \( Th \), as spurious components, and remove them from the estimation set \( S \) to get a new estimation set, \( S' \),

\[
S' = \{ \hat{\omega}_i, \hat{\beta}_i(l) \}_{i=1}^{\hat{K}},
\]

where \( \{ l_1, l_2, \ldots, l_{\hat{K}} \} \) is a permutation of the number set of \( \{ 1, 2, \ldots, \hat{K} \} \).

The threshold currently employed in our 1DPE algorithm is simply:

\[
Th = \max \{ 0.1, \sigma / \sqrt{2} \}.
\]

In fact, this 1D parameter estimation can be conducted on \( D_l(n) \) with different fixed \( l \). In order to get more accurate estimation, we recommend implementing this 1D algorithm along more than one \( l \) and then selecting the best estimation results as the final 1D estimation results. The best result can be determined by the error between the input data and the data generated from the estimation results. The estimation result that produces minimal error is naturally regarded as the best one. Of course, this improvement of the estimation accuracy is at the cost of computation efficiency.
Similarly, model (4.19) can also be re-written as a function of variable \( l \) if we fix the variable \( n \) in (4.19).

\[
D_n(l) = D(n, l) = \sum_{k=1}^{K} \beta_{dk}(n) \exp\{j\omega_{dk}l\} + e_n(l),
\]

where \( \beta_{dk}(n) = a_k \exp\{j\omega_k n + j\phi_k\} \) and \( e_n(l) = e(n, l) \).

By following exactly the same procedure mentioned above, we can also obtain a set of estimated parameters of Eq. (4.29), which are \( \hat{K}_D \) speed frequency components.

\[
S_D' = \{\hat{\omega}_{dk}, \hat{\beta}_{dk}(n)\}_{k=1}^{\hat{K}_D},
\]

where \( \{l_1, l_2, \ldots, l_{\hat{K}_D}\} \) is a permutation of the number set of \( \{1, 2, \ldots, \hat{K}_D\} \).

4.5.1.2 Combination of the Two 1D Estimation Results

Now the remaining problem is how we can obtain the estimation of the parameter set \( P \) from the two 1D estimation results, \( S' \) and \( S_D' \). The most straightforward method is to match the position frequency \( \hat{\omega}_{lp} \) from \( S' \) with the speed frequency \( \hat{\omega}_{dk} \) from \( S_D' \) by employing the fact of \( a_k = |\beta_k| = |\beta_{dk}| \), where \( k = lp = lq \). Unfortunately, this is problematic in practice because:

- The ESPRIT algorithm can only estimate all the unique frequency components contained in the 1D data. If a group of scatterers have the same position frequency, \( \omega_k \), or speed frequency, \( \omega_{dk} \), the ESPRIT algorithm can only output one frequency component for this group of scatterers in \( S' \) or \( S_D' \). This means that our combining algorithm must have the capacity of matching one position
frequency component \( \hat{\omega}_p \) in \( S' \) to multiple speed frequency components in \( S'_D \),
or vice versa, which is quite difficult.

- Although \( |\beta_k| \) is theoretically equal to \( |\beta_{Dk}| \), its estimate, \( |\hat{\beta}_p| \), does not necessarily match \( |\hat{\beta}_{Dk}| \) due to estimation error.

Therefore, alternate combination methods are required to combine the two 1D estimation results. In order to facilitate the development of a new combination method, we first define two matrix operations.

- **Column-exchanging operation** \( X_l \): \( X_l \) is the matrix such that, if \( AX_l = B \), \( B \) is the matrix obtained by exchanging the first and the \( l \)th column of \( A \).

- **Column-duplicating operation** \( C_l \): \( C_l \) is the matrix such that, if \( AC_l = B \), \( B \) is the matrix obtained by duplicating the \( l \)th column of \( A \), and inserting the new column after the \( l \)th column in \( A \).

From the estimated 1D frequency components in \( S' \) and \( S'_D \), we can form the following two exponential matrices \( \hat{E}' \) and \( \hat{E}'_D \):

\[
\hat{E}' = \begin{bmatrix} \hat{\Omega}_k & \hat{\Omega}_{k_1} & \ldots & \hat{\Omega}_{k_{L-1}} \end{bmatrix},
\]

where \( \hat{\Omega}_k = \begin{bmatrix} 1 & \exp{j\hat{\omega}_k} & \ldots & \exp{j\hat{\omega}_k(N-1)} \end{bmatrix}^T \), and

\[
\hat{E}'_D = \begin{bmatrix} \hat{\Omega}_{D_k} & \hat{\Omega}_{D_{k_1}} & \ldots & \hat{\Omega}_{D_{k_{L-1}}} \end{bmatrix},
\]

where \( \hat{\Omega}_{D_k} = \begin{bmatrix} 1 & \exp{j\hat{\omega}_{D_k}} & \ldots & \exp{j\hat{\omega}_{D_k}(L-1)} \end{bmatrix}^T \).

Also, we define the estimates of \( E \) and \( E'_D \) as \( \hat{E} \) and \( \hat{E}'_D \).
\[
\hat{E} = [\hat{\Omega}_1 \quad \hat{\Omega}_2 \quad \ldots \quad \hat{\Omega}_K], \text{ and } \tag{4.33}
\]
\[
\hat{E}_D = [\hat{\Omega}_{D1} \quad \hat{\Omega}_{D2} \quad \ldots \quad \hat{\Omega}_{DK}]. \tag{4.34}
\]

From (4.31) through (4.34), as well as the definition of \( X_l \) and \( C_l \), we obtain:
\[
\hat{E} = \hat{E}'P, \tag{4.35}
\]

where
\[
P = C_{n1}C_{n2}\ldots C_{nb}X_{n1}X_{n2}\ldots X_{nt},
\]
\[
\hat{E}_D = \hat{E}'D_P, \tag{4.36}
\]

where
\[
P_D = C_{nD1}C_{nD2}\ldots C_{nDb}X_{ID1}X_{ID2}\ldots X_{IDt}.
\]

By substituting (4.35) and (4.36) into (4.23), we obtain:
\[
\hat{R} = \hat{E}'A\hat{E}_D^T + \hat{O}, \tag{4.37a}
\]

where
\[
A = P\Sigma P_D^T. \tag{4.37b}
\]

From (4.37b) we know that \( A \) can be obtained by duplicating some columns and rows of \( \Sigma \), and then exchanging some columns and rows of \( \Sigma \). Therefore, the theoretical form of \( A \) is that some non-zero elements are embedded in many zero elements, and the non-zero elements have the same values as the non-zero elements in \( \Sigma \). In practice, because of the estimation error and numerical error, the estimate of \( A \), \( \hat{A} \), is a matrix composed of some large-value elements, or "nontrivial elements", embedded in many small-valued elements, or "trivial elements".

From (4.23) we know that each non-zero element in \( \Sigma \) corresponds to a frequency pair combined by a position frequency component from \( S' \) and a speed frequency component from \( S'_D \). Therefore, from the relationship between \( A \) and \( \Sigma \), i.e. (4.37b), a new method to match position frequency and speed frequency can be obtained.
• We can simply use a heuristic method to select those "nontrivial" elements in
\( \tilde{A} \). That is we first select a maximal element from each row and each column of
\( \tilde{A} \). Then we choose the smallest value among those selected elements as the
threshold to separate the "nontrivial" elements from those "trivial" ones. If \( \tilde{K} \)
"nontrivial" elements are finally selected, \( \tilde{K} \) will be considered as the final
estimation of the number of scatterers, or \( K \), encoded in the 2D radar data.

• Each selected element corresponds to a matched frequency pair in that its row
index indicates the index of the position frequency in \( S' \), and its column index
shows the position of the speed frequency in \( S'_D \). Specifically, if the element
\( A(p,q) \) is a chosen element, the position frequency component \( \{\tilde{\Omega}_{lp}, \tilde{\beta}_{lp}\} \) from \( S' \),
and the speed frequency component \( \{\tilde{\omega}_{Dlp}, \tilde{\beta}_{Dlp}\} \) from \( S'_D \) will form a matched
frequency pair, \( \{\tilde{\beta}_{lp}, \tilde{\omega}_{lp}; \tilde{\beta}_{Dlp}, \tilde{\omega}_{Dlp}\} \)

Note that we have two complex scattering coefficients, \( \tilde{\beta}_{lp} \) and \( \tilde{\beta}_{lp} \), in the above
matched pair. Which complex scattering coefficient should we keep as the final estimation
of the scattering coefficient of the corresponding scatterer? The strategy we employed is as
follows:

• If the position frequency, \( \tilde{\Omega}_{lp} \), is unique in the matrix (4.33), we can simply
adopt the magnitude of \( \tilde{\beta}_{lp} \) as the estimated scattering coefficient of the
corresponding scatterer, i.e. \( \tilde{a}_k = |\tilde{\beta}_{lp}| \);
• If the position frequency, $\hat{\omega}_{lp}$, is not unique in (4.33), we will adopt the magnitude of $\hat{\beta}_{Dlp}$ as the estimated scattering coefficient of the corresponding scatterer, i.e. $\hat{a}_k = |\hat{\beta}_{Dlp}|$.

The rationale behind this treatment is that it can reduce the estimation error caused by scintillation effects, which occurs when multiple scatterers exist at the same range position [4]. As a matter of fact, this is a fairly critical point. We know that we almost have no way to resolve the scatterers at the same range position if we only utilize the 1D HRR range signatures [4]. However, we can distinguish between these scatterers as long as all these scatterers have different radial speeds, or Doppler frequencies. This improvement in the scatterer resolution is simply due to our introduction of a new radar sensor dimension, the Doppler frequency, compared with the 1D HRR range signatures. Hopefully, this improvement will inspire us to find more new radar sensor dimensions to continuously refine the radar resolution.

The remaining problem is how we can obtain the matrix $A$. According to (4.37) we can calculate $A$ as:

$$A = (\hat{E}^H \hat{E}^\prime)^{-1} \hat{E}^H (\hat{R} - \hat{\Theta}) \hat{E}^\prime_C (\hat{E}^D_\prime \hat{E}^C_D)^{-1},$$

where $(\bullet)^H$ stands for the Hermitian transposition operation, and $(\bullet)^C$ stands for the complex conjugate operation.

The overall implementation of 1DPE is summarized in Figure. 4.2.
4.5.2 Scatterer Extraction Using 2D Discrete Fourier Transform (2DFT)

The second parameter-estimation algorithm is presented in this section. If we directly apply 2D DFT to Eq. (4.22), we obtain:

$$
\hat{R}(\omega_1, \omega_2) = \sum_{m_2=0}^{L-1} \sum_{m_1=0}^{N-1} R(m_1, m_2) \exp{-j \omega_1 m_1} \exp{-j \omega_2 m_2}
$$

$$
= \sum_{k=1}^{K} \eta_k(\omega_1, \omega_2) \frac{\sin[N_1(\omega_1 - \omega_k)/2]}{\sin[(\omega_1 - \omega_k)/2]} \frac{\sin[L_1(\omega_2 - \omega_{dk})/2]}{\sin[(\omega_2 - \omega_{dk})/2]} + \sigma^2, \quad (4.39)
$$

where

$$
\eta_k(\omega_1, \omega_2) = a_k^2 \exp{j(\omega_1 - \omega_k)(N_1 - 1)} \exp{j(\omega_2 - \omega_{dk})(L_1 - 1)},
$$

$$
N_i \geq N, \text{ and } L_i \geq L
$$

94
Eq. (4.39) suggests a very straightforward way to estimate the scatterer parameters:

- Apply 2D FFT to the correlation matrix, $R$, of the 2D radar data, $D$;
- Pick $K$ highest amplitude peaks from the 2D FFT results.
- If we denote the locations of the $K$ peaks as $\{\sigma_{1k}, \sigma_{2k}\}_{k=1}^{K}$, and denote the corresponding complex values at those peaks as $\{v_k\}_{k=1}^{K}$, the parameter set of the $K$ scatterers, $P$, can be obtained as:

$$\bar{\omega}_k = \sigma_{1k}, \quad (4.40a)$$
$$\bar{\omega}_{dk} = \sigma_{2k}, \quad \text{and} \quad (4.40b)$$
$$a_k = \sqrt{|v_k|}. \quad (4.40c)$$

Because we have not determined a reliable method to estimate the number of scatterers, $K$, in this algorithm, we have to specify $K$ as a known parameter. Another primary problem with this algorithm is that it has limited frequency resolution due to the low-resolution property of the 2D DFT. If we employ Eq. (4.24) to compute the correlation matrix $R$ of the 2D radar data, it is equivalent to adding a 2D triangular window on the 2D radar data. Therefore, the main lobe width for each scatterer will be $2\pi N$ by $2\pi L$, where $N$ and $L$ are the size of the 2D radar data. That is, any two scatterers whose relative distances are within an area of $[2\pi N, 2\pi L]$ are difficult to extract (differentiate) using this algorithm.

### 4.6 Simulations and Performance Comparison

In order to evaluate the performance of the above two parameter-estimation algorithms, we tested the algorithms on a set of synthetic 2D HRR radar data. We assume that there are 6 scatterers encoded in the radar returns. The parameters of these scatterers are listed in the following table.
Note that the speed frequencies of both scatterer 3 and 4 are 0. That is, these two scatterers are supposed to be the clutter-scatterers. Also, note that the position frequencies of both scatterer 2 and 6 are the same, which implies that these two scatterers are at the same range position, and this will cause scintillation effect in 1D HRR signatures. Moreover, the distances between scatterer 2 and 6 are [0, 0.5-0.36], which are smaller than $[\frac{2\pi}{N}, \frac{2\pi}{L}]$. Therefore, 2DFT algorithm is expected to have difficulty in properly extracting these two scatterers.

We employ Eq. (4.20) to generate a series of 2D radar data with $N=L=32$, and with different SNRs. The SNR is controlled by changing the standard deviation (STD) of the CGAW noise from 0.15 to 0.51 with an incremental step of 0.02. The relationship between SNR and STD in our experiments is shown as follows:

$$SNR = 10 \log_{10}(1.3226/STD^2) \text{ dB},$$

where 1.3226 is the average power of the 6 scatterers employed in our experiments. Therefore, the range of the SNR is approximately from 7.1 dB to 17.7 dB.
To achieve consistent evaluation results, we generate 200 realizations for each STD value, and run each algorithm on each of these 200 data sets, and average all the 200 experimental results to serve as the final result of the algorithm at that STD, or SNR.

In Experiment 1, we illustrate IDPE’s capacity of estimating $K$. We ran the IDPE algorithm a total of 3600 (200×19) times, and the distribution of the wrong estimations of $K$ is illustrated in Figure 4.3. From the results we know that the probability of estimation error is only about 0.47% (18/3600) when SNR is greater than 10 dB. Therefore, the IDPE algorithm has reasonably reliable performance in estimating $K$ when the SNR is not inordinately challenging.

![Graph showing distribution of wrong estimations of $K$ vs. SNR](image)

Figure 4.3 Distribution of the wrong estimations of $K$

As we mentioned previously, IDPE can estimate the scatterer number, $K$, while 2DFT cannot. Therefore, to keep the experimental results comparable, we only compare IDPE’s
results in the case where \( K \) is correctly estimated with 2DFT’s results in the following two experiments.

We define the estimation of the position and speed frequency of a scatterer as frequency estimation. If the position frequency estimation error of a scatterer is bigger than 0.05, or the speed frequency estimation error of a scatter is bigger than 0.065, this scatterer is considered as a mis-estimated scatterer. In order to illustrate and compare these two algorithms scatterer estimation capacities, we will define the probability of scatterer estimation error, \( P_e \), as:

\[
P_e = M / T ,
\]

(4.41)

where \( M \) is the number of mis-estimated scatterers, and \( T \) is the total number of estimated scatterers.

The results of Experiment 2 are illustrated in Figure 4.4.

![Figure 4.4 Probability of estimation error](image)

Figure 4.4 Probability of estimation error

98
Figure 4.4 clearly suggests that 1DPE exhibits significantly better scatterer estimation accuracy.

Because $P_r$ cannot illustrate the estimation accuracy when the scatterers are correctly estimated, we need another measurement to quantify the algorithms' estimation accuracy for the properly estimated scatterers. Therefore, we define a new expression to meet this need:

$$V = MC + H,$$

(4.42)

where $V$ is the evaluation value, and $C$ is a constant, which quantifies the penalty for each mis-estimated scatterer. We let $C$ equal to $0.7^2$ in our experiments.

$H$ is the Mean Square Error (MSE) of those properly estimated scatterers. The results of Experiment 3 are illustrated in Figure 4.5.

---

2 Because (a) there are 3 parameters, position frequency, speed frequency, and scattering coefficient, for each scatterer, (b) if the position frequency estimation error of a scatterer is bigger than 0.05, and the speed frequency estimation error is bigger than 0.065, it is considered as a mis-estimated scatterer, and (c) the penalty to the scattering coefficient of a mis-estimated scatterer is defined as 0.585, the Penalty/scatterer $=0.05+0.065+0.585=0.7$. 

99
From the above experimental results, we observe that 1DPE algorithm significantly outperforms 2DFT algorithm under reasonable SNR. Moreover, 1DPE exhibits reliable performance in automatically detecting the number of scatterers, $K$, when the SNR is not too low. However, 2DFT also has some attractive properties [1], which include:

- 2DFT has weaker requirements on the distribution of the additive noise, i.e. it is not necessary to assume the additive noise to be circularly Gaussian distributed.
- 2DFT can tolerate greater model mismatching.

4.7 Some Comments

We aware that our HRR models, as well as the associated parameter estimation algorithms, might not be appropriate for direct application to a fieldable HRR-based ATR
system. For example, our 1DPE algorithm will be computationally demanding if the size of the 2D HRR raw data increases, such as data sized 1024-by-128, instead of the 32-by-32 data used in the simulations presented in Section 4.6. This is one of reasons we have also proposed a 2DFT algorithm, which is simply an interpretation of a general 2D FFT process based on the proposed model. However, the 1DPE is worth considering for the following reasons. First, the algorithm itself is a new parameter estimation algorithm for a general 2D exponential signal model, and especially the pairing scheme used to combine the two sets of estimation results obtained from the 1D parameter estimation. Compared with some other available algorithms in the signal processing area, our pairing method has some advantages such as computational simplicity and it is capable of distinguishing scatterers with the same Doppler frequency and different range positions. However, a systematic comparison of our algorithm and other available algorithms is still an open topic worth further study. Second, this is one of only a limited number of attempts to apply high-resolution estimation techniques to HRR data processing. Consequently, more effort is needed to devise scalable parameter estimation algorithms. Further, such studies will contribute to modified formulations of models for different application scenarios.
In Chapter 4 we proposed a physics-based HRR moving target model, and defined the parameters of the model as a set of potential features. Subsequently, two feature extraction algorithms based on this model were devised and presented. However, when this set of features is employed for classification-based applications, such as ATR, they exhibit a problem, which arises because these features are extracted under the criterion of maximally retaining the energy in the original radar signals, and are therefore categorized as representational features. Consequently, they are not necessarily features that are optimal for discriminating targets, or for classification. In other words, although the features defined and extracted in this way can preserve target information, there is no guarantee that those features maintain the major discriminating information between targets. However, it is the discriminating information that plays the most important role in classification. From this perspective, we have not satisfactorily achieved the goal of finding a high-performance feature extraction algorithm for HRR radar signals. Moreover, the algorithms presented in Chapter 4 extract target features from raw 2D HRR data, while, in some cases, only HRR data available is HRR signatures. We thus focus here on the challenge of HRR signatures, and we develop a feature extraction algorithm applicable to HRR signatures.
In this chapter we present a Kernel-based nonlinear Feature Extraction (KFE) algorithm that is capable of handling high dimensional HRR signatures and demonstrates attractive performance characteristics. As we show subsequently, the derivation and implementation of this KFE algorithm requires no assumptions on the properties of the underlying patterns, and its application is thus not limited to HRR signatures for HRR-based ATR. In fact, this is a general feature extraction algorithm, and can find potential applications in many scenarios. Therefore, in this chapter, we attempt to derive and present this KFE algorithm from a more general pattern recognition perspective instead of focusing only on feature extraction algorithms for HRR-based applications. As a consequence, we switch our terminology to utilize the vocabulary of the pattern recognition area, replacing the corresponding terms we previously employed from the radar literature. For example, we will use patterns to replace 1D radar signals or signatures, we use features of a pattern to represent the elements in 1D radar signals or signatures, and use classes to replace targets in radar ATR area.

In Section 5.1 we first summarize some of the basic ideas involved in feature extraction. The purpose of this section is to unify our terminology, to define our research goal and methodology, and to build a conceptual foundation for our subsequent discussion. Also, in order to make this chapter self-contained, in Section 5.2 we review a set of linear feature extraction algorithms based on scatter matrices which can be directly computed from a given set of patterns, or a training set. In Section 5.3 we derive a high performance nonlinear feature extraction algorithm, or KFE algorithm, by extending the linear feature extraction algorithms using a technique named the "kernel trick". A modification of the primitive form of KFE is introduced in Section 5.4 to improve the robustness and flexibility.
of the original KFE algorithm. We then conduct example experiments by applying the KFE and Modified KFE (MKFE) algorithms to synthetic pattern sets and measured HRR signatures. Our experimental results are presented in Section 5.5, and our observations are presented in Section 5.6.

5.1 On Classification-oriented Feature Extraction

In pattern recognition, in order to simplify the design of classifiers, pre-processing techniques are usually applied to the original patterns to achieve goals such as: (1) reducing the pattern dimensionality with a minimal impact on the class separability, or (2) improving the class separability with a minimal number of extracted features.

These pre-processing techniques are further classified by some researchers into two subcategories: (1) feature selection, and (2) feature extraction. However, other authors have tended to ignore the subtle difference between these two phrases, and choose to consider feature selection as either a synonym or a special case of feature extraction [84]. For the sake of simplicity, we also do not differentiate these two terms. Consequently, we will use feature extraction to represent all of the relevant pre-processing techniques that attempt to achieve the goals mentioned above.

When designing feature extraction algorithms the criteria defining the “best features”, as well as the potential applications of the extracted features, play a determining role in defining the features to extract. According to different feature extraction criteria and applications, current feature extraction algorithms are primarily classified into two categories [84]: feature extraction algorithms for either representation or for classification.
The first class generally uses energy-retention as the feature extraction criteria. That is, the resultant features maximally maintain the energy in the original patterns. Thus, these kinds of features are named *representative features*. Many algorithms that directly stem from signal processing techniques, such as *spectral estimation* [80] and *Principle Component Analysis (PCA)* [98], belong to this group. Although it is claimed that in many scenarios these types of features may also have good performance when applied to classification problems, it is difficult, if not impossible, to guarantee that the best classification performance can be reached using these representative features. For this reason researchers in the pattern recognition community have proposed different algorithms aimed at maximally maintaining the class separability information embedded in the original patterns, i.e., *feature extraction for classification*. The extracted features derived by employing this class of algorithms are generally called *discriminating features*, or *classification-oriented features*. Here we focus on devising another example of this type of feature extraction algorithms. Unless otherwise noted, we will generally utilize the term “feature extraction” to refer to this specific category of feature-oriented pre-processing.

There are several challenges in designing a high performance discriminating feature extraction algorithm. These are discussed below.

(1) **Definition of Class Separability Criteria**

As mentioned above, discriminating features are designed to maximally maintain class separability among patterns from different classes. However, exactly how best to quantitatively measure the class separability is a difficult issue that is a longstanding research topic, usually referred to as *Feature Set Evaluation (FSE)* [120] since the 1960s.
Basically, FSE is any technique used to rank the capacity of different feature sets for distinguishing their underlying classes. A brief review of this topic is presented here. Although classification error, $P_e$, is the most natural criterion for evaluating the class separability under different feature sets, the difficulty in estimating $P_e$ [84], and the intrinsic problems embedded in employing $P_e$ as the ranking criterion [127] make it impractical to directly use $P_e$, except for some special cases. Alternatively, many $P_e$-related criteria have been proposed. These primarily fall into three categories: (1) *information (uncertainty)-based criteria* [127, 131, 135], (2) *distance-based criteria* [127], and (3) *dependence-based criteria* [122, 127, 137]. However, all of these criteria are based on extensive knowledge of the probability distributions of the underlying classes. Estimating the probability distribution from limited number of patterns is an ill-posed problem, and reliable estimation results cannot be guaranteed. Thus, most of these FSE algorithms are not easily incorporated into practical applications. To avoid this difficulty other criteria that are calculated directly from a set of given patterns were proposed in the 1980s and 1990s. Among these criteria are *Boundary method-based FSE methods* [120, 126], *neuro-fuzzy methods* based on a feature evaluation index [117], *set-cover-based algorithms* [118], *decision-boundary-based algorithms* [141], and *scatter matrix-based criteria* [84].

One class of criteria, *scatter matrix-based criteria*, are attractive due to their simplicity, intuitive, appeal performance, and robustness. Consequently, we will focus on this category of criteria as discussed below in Section 5.2.
(2) Linear Feature Extraction Algorithms vs. Nonlinear Algorithms

Before designing a feature extraction algorithm we have to decide whether we should develop a linear algorithm or a nonlinear one.

A linear algorithm is actually a linear transformation, which can be represented as:

\[ R = E^T X \]  

(5.1)

where \( X \) is a \( n \)-feature, or \( n \)-dimensional, original pattern, and \( R \) is a \( m \)-feature, or \( m \)-feature, pattern formed from \( m \) extracted features, generally with \( m < n \). \( R \) is sometimes referred to as the extracted pattern. \( E \) is the feature extraction matrix. Designing a linear feature extraction algorithm is equivalent to finding the matrix \( E \), which is the most effective transform for preserving the class separability encoded in the original patterns.

Similarly, a nonlinear feature extraction can be defined as:

\[ R = F(X) = \begin{bmatrix} F_1(X) \\ F_2(X) \\ \vdots \\ F_m(X) \end{bmatrix} \]  

(5.2)

where \( F \) is a nonlinear function vector, which maps a \( n \)-feature original pattern, \( X \), to a \( m \)-feature extracted pattern, \( R \).

The primary advantages of linear algorithms are their simplicity and robustness. The simplicity is manifest mainly in two ways: (a) the intrinsic simplicity of the linear operation on a linear space results in most of the linear feature extraction algorithms being mathematically manageable, and therefore, most of them are theoretically optimal; (b) linear algorithms are generally easier and faster to implement. However, it can be shown that linear operators are limited to affine transformations, which is naturally a limitation to
exploring all of the class separability options among distributions in complex classification problems. Thus, linear algorithms are generally not sufficient for improving the classification performance of complex classifiers. In contrast, nonlinear algorithms are difficult to analyze theoretically - due to their non-linearities - and are commonly more computationally demanding than linear algorithms. However, they are more capable of extracting discriminating information embedded in the original patterns, and many current studies have demonstrated that a proper nonlinear feature extract algorithm can either dramatically reduce the complexity of classifiers, or boost classification performance [94].

In this chapter, we devise a practical nonlinear feature extraction algorithm.

(3) Determination of the Number of Features Retained in the Extracted Patterns

The third issue involved in designing feature extraction algorithms is to determine the number of features we should retain in the extracted patterns.

Let us first consider an ideal scenario. As we know, the Bayes classifier associates a pattern with the class possessing the largest posteriori probability for the pattern. Therefore, if a Bayes classifier is employed, the best feature extraction algorithm is naturally the procedure that computes the posteriori probability of each input pattern for each class. Under this condition, only \( L-1 \) features are needed for each pattern, given a \( L \)-class classification problem in order to achieve the best performance. That is, for a 2-class problem, one feature is good enough to accomplish the best classification result. This observation implies that if the feature extraction algorithm is efficient, a small number of features are sufficient to achieve the best performance. Namely, the quality of features plays a more critical role in determining the classification performance than the quantity of
features. Meanwhile, this observation also suggests that nonlinear feature extraction algorithms are more promising in extracting features with optimal performance than linear ones, because nonlinear algorithms are more likely to approximate the posteriori probability function of the original patterns, which generally is a nonlinear function of the original pattern.

However, in practice it is typically difficult to estimate the posteriori probability of a given set of patterns. Thus, it is typical to employ alternative algorithms that are more readily realized. When the extracted features are not optimal, the larger the number of features typically needed, and the more likely the extracted features will potentially convey additional discriminating properties among classes. Generally speaking, the greater the number of extracted features, the better the potential classification results. However, when the number of extracted features is larger, classification requires more computation, and more importantly, the classifiers become more difficult to design. If the classifiers are not properly chosen they will not be able to exploit the class separability embedded in the extracted patterns, which leads to a deterioration in classification performance. This trade-off between the numbers of features and the quality of the features plays an important role in determining the optimal number of features to be extracted.

5.2 Scatter Matrix-based Linear Feature Extraction

We are going to design a set of nonlinear feature extraction algorithms in this chapter. However, because our proposed nonlinear algorithms are developed from a set of linear algorithms, referred to as scatter-matrix based linear feature extraction, we first review the
concepts of this set of linear algorithms in order to make this chapter self-contained, as well as to unify the notation.

As we mentioned previously, many of the available class separability criteria require knowledge of the probability distribution of the evaluated features. However, except for some special cases, estimating the probability distribution from limited numbers of patterns is an ill-posed problem, and reliable estimation results cannot be guaranteed. To avoid the difficulty of finding the explicit form of the probability distribution, a set of criteria that can be directly computed from given patterns have been proposed, and these criteria are referred to as scatter matrix-based criteria [84] in statistical discriminant analysis.

To facilitate our subsequent derivation and representation, we first define the following terms:

- $X_k$: the $k$th pattern in the given pattern set, or training set;
- $X_i^{(i)}$: the $k$th pattern among all patterns from the $i$th class in the given pattern set.
- $I_N$: a $N$-by-$N$ identity matrix
- $\Lambda_m$: a $m$-by-$m$ diagonal matrix
- $1_{N_1 \times N_2}$: a $N_1$-by-$N_2$ matrix, whose all elements are all one. When $N_1 = N_2$, this can be simplified as $I_{N_1}$
- $\Theta_{X}^{(i)} = [X_1^{(i)} X_2^{(i)} \cdots X_{N_j}^{(i)}]$: a matrix formed by combining all the patterns from class $i$, where $X_k^{(i)}$ is the $k$th pattern from class $i$,
- $\Theta_X = [\Theta_{X}^{(1)} \Theta_{X}^{(2)} \cdots \Theta_{X}^{(L)}]$: a matrix formed by combining all $\Theta_{X}^{(i)}$, $i = 1 \ldots L$, where $L$ is the total number of classes.
The foundation of statistical discriminant analysis is the formation of three scatter matrices. Note that all of these scatter matrices are designed to be coordinate-shift invariant [84].

(1) **Within-class matrix, $S_w$**

Define the within-class scatter matrix, $S_w$, as:

$$S_w = \sum_{i=1}^{L} P_i E\{(X - M_i)(X - M_i)^T | \omega_i\} = \sum_{i=1}^{L} P_i \Sigma_i$$

where $L$ is the number of classes, $P_i$ is the prior probability of class $i$, or $\omega_i$. $M_i$ is the mean of patterns from $\omega_i$. That is, $M_i = E\{X|\omega_i\}$. This within-class matrix indicates the spread of patterns around their individual class means.

If we replace $P_i$ with the pattern frequency of class $i$, and $E\{X\}$ with the pattern average, we can obtain the estimate of $S_w$ directly from the pattern set:

$$S_w = \frac{1}{N} \sum_{i=1}^{L} \frac{N_i}{N} \sum_{k=1}^{N_i} (X_i^{(i)} - M_i)(X_i^{(i)} - M_i)^T$$

$$= \frac{1}{N} \sum_{i=1}^{L} \frac{N_i}{N} \Theta_X^{(i)} \left( I_{N_i} - \frac{1}{N_i} 1_{N_i} \right) \left( I_{N_i} - \frac{1}{N_i} 1_{N_i} \right)^T \Theta_X^{(i)^T}$$

$$= \frac{1}{N} \Theta_X \begin{bmatrix} N_i 1_{N_i} - 1_{N_i} & \cdots & \cdots \ \cdots & \cdots & \cdots \ N_L 1_{N_L} - 1_{N_L} \end{bmatrix} \Theta_X^T$$

where $N_i$ is the number of patterns in class $i$, and $N$ is the total number of patterns from all classes. Note that $(I_{N_i} - \frac{1}{N_i} 1_{N_i}) = (I_{N_i} - \frac{1}{N_i} 1_{N_i})(I_{N_i} - \frac{1}{N_i} 1_{N_i})$ when deriving (5.4).

(2) **Between-class matrix, $S_b$**

Define the between-class scatter matrix, $S_b$, as:
\[ S_b = \sum_{i=1}^{L} P_i (M_i - M_0)(M_i - M_0)^T, \quad (5.5) \]

where \( M_0 = E\{X\} = \sum_{i=1}^{L} P_i M_i \) denotes the mean of the mixture distribution of all the classes. This between-class matrix captures the spread of the mean of each class around the mean of all the classes.

The estimate of \( S_w \) can be obtained similarly to that of \( S_w \), and can be expressed as:

\[
S_b = \sum_{i=1}^{L} \frac{N_i}{N} (M_i - M_0)(M_i - M_0)^T
\]

\[
= \frac{1}{N} \Theta X \left[ \begin{array}{cccc}
N_1 & & & \\
& \ddots & & \\
& & \ddots & \\
& & & N_L
\end{array} \right] W^T \Theta X^T
\]

(5.6)

where \( W = \left[ \begin{array}{cccc}
\frac{1}{N_1} 1_{N_1 \times L} & & & \\
& \ddots & & \\
& & \ddots & \\
& & & \frac{1}{N_L} 1_{N_L \times L}
\end{array} \right] \)

(3) Mixture scatter matrix, \( S_m \)

Define the mixture scatter matrix, \( S_m \), as:

\[ S_m = E\{(X - M_0)(X - M_0)^T\} = S_w + S_b \quad (5.7) \]

This mixture scatter matrix is the covariance matrix of all patterns from all classes.

The estimate of \( S_m \) can be represented as:

\[ S_m = \frac{1}{N} \sum_{i=1}^{N} (X_i - M_0)(X_i - M_0)^T \quad (5.8) \]

Intuitively, it is easy to reason that bigger between-class scatter and smaller within-class scatter imply higher class separability, which in turn suggests that the involved feature
set is better. In order to quantify this intuition, this idea can be mathematically expressed in many forms. We list four of the most frequently adopted criteria [84] here:

\[ J_1 = \text{tr}(S_2^{-1}S_1) \]  
\[ J_2 = \ln |S_2^{-1}S_1| = \ln |S_1| - \ln |S_2| \]  
\[ J_3 = \text{tr}S_1 - \mu(\text{tr}S_2 - c) \]  
\[ J_4 = \frac{\text{tr}S_1}{\text{tr}S_2} \]

where \( \text{tr}\{A\} \) denote the trace operation of the matrix \( A \), and \( S_1 \) and \( S_2 \) are a pair chosen from \( S_w, S_b, \) and \( S_m \). However, \( S_b \) cannot be \( S_2 \), because \( S_b \) is generally not full rank, which can be observed from Equation (5.5). In order to simplify our following derivation, we fix \( S_1 \) as \( S_b \) and \( S_2 \) as \( S_w \). It can be shown that different combinations generate similar separability evaluation results [84]. Thus, employing different forms of criteria does not significantly change the extracted features. In our subsequent discussion we derive our feature extraction algorithm based on \( J_1 \) due to its simplicity and another attractive property that is presented in Theorem 1.

**Theorem 1:** Criterion \( J_1 \) is invariant under nonsingular linear transformations. That is, it is a coordinate-independent criterion.

**Proof:**

Assume a nonsingular linear transform, \( B \), which transforms \( R \) in Equation (5.1) to \( Y \).

\[ Y = B^T R \]

Thus, from Equation (5.9) we obtain,

\[ \text{tr}\{S_{wR}^{-1}S_{bR}\} = \text{tr}\{(B^T S_{wB})^{-1}(B^T S_{bB})\} = \text{tr}\{B^{-1}S_{wB}^{-1}S_{bB} B\} = \text{tr}\{S_{wR}^{-1}S_{bR}\} \]
It is worth mentioning that the collection of criteria presented in Equation (5.9)-(5.12) have two drawbacks. First, they can only predict the feature performance under linear classifiers. Second, if there is no mean difference between the classes, the class separability is dominated by the difference in variance. Under these conditions the evaluation results obtained by these criteria are no longer valid [84].

Once we have a class separability criterion, \( J_I \), devising a linear feature extraction algorithm in the form of Equation (5.1) becomes a linear optimization problem, which can be represented as follows:

\[
\max_{E} J_I(E; m) = tr\{S_{wE}^{-1}S_{bE}\} = tr\{(E^T S_{wE}^{-1}E)^{-1}(E^T S_{bE} E)\}
\]

where \( S_{wE} \) and \( S_{wE} \) are the within-class scattering matrices of the extracted patterns and the original patterns respectively, and \( S_{bE} \) and \( S_{bE} \) are the between-class scattering matrix of the extracted patterns and the original patterns respectively.

It is not difficult to prove that (1) the eigenvalues of matrix \( S_{wE}^{-1}S_{bE} \) are real and positive, and (2) the transformation matrix, \( E \), in Equation (5.1) can be formed by \( m \) eigenvectors corresponding to the \( m \) biggest eigenvalues of matrix \( S_{wE}^{-1}S_{bE} \) [84]. That is,

\[
E = [e_1 \ e_2 \ \cdots \ e_m],
\]

where \( (S_{wE}^{-1}S_{bE})e_i = \lambda_i e_i, \ i = 1 \ldots n, \) and \( \lambda_1 > \lambda_2 > \ldots > \lambda_n. \)
5.3 Kernel-based Feature Extraction

As we previously mentioned, the linear feature extraction algorithm presented in Section 5.2 has two critical drawbacks that dramatically hamper its performance and application. These drawbacks are:

- From Equation (5.5) and (5.6) we can see that, if the mean pattern, $M_i$, of any class $i$, is the same as the mean pattern of all the patterns, $M_0$, $S_b$ will lose the ability to distinguish class $i$ from the remaining classes.

- The features extracted by this linear algorithm are mainly used for linear classifiers, because these features only exploit the linear separability among classes.

However, in many cases, although the distributions from different classes are linearly inseparable, they may be nonlinearly separable. Also we need a general algorithm that can explore every aspect of class separability, including both mean-difference induced separability and non-mean-difference induced separability, i.e. variance-difference induced separability. All of these requirements motivate us to develop a practical nonlinear feature extraction algorithm that can overcome the difficulties faced by its linear counterpart.

The basis of our nonlinear algorithm is to employ a technique named the "kernel trick", which can transform a linear operation in a very high dimensionality space into a nonlinear operation in a space with much lower dimensionality. For the background of this kernel trick, please refer to B3.3. in Appendix B.
If we represent the nonlinear feature extraction function, $F_i(X), i=1 \ldots m$, in Equation (5.2) as the linear combination of a group of kernel functions, $K(X_j, X), j=1 \ldots N$, the general form of the nonlinear feature extraction in Equation (5.2) can be expressed as:

$$
R = F(X) = \begin{bmatrix}
F_1(X) \\
F_2(X) \\
\vdots \\
F_m(X)
\end{bmatrix} = \sum_{i=1}^{N} \alpha_i K(X_i, X) = A^T K(\ast, X),
$$

where

$$
A = \begin{bmatrix}
\alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,m} \\
\alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{2,m} \\
\vdots & \vdots & \ddots & \vdots \\
\alpha_{N,1} & \alpha_{N,2} & \cdots & \alpha_{N,m}
\end{bmatrix} = \begin{bmatrix}
\alpha_1 \\
\alpha_2 \\
\vdots \\
\alpha_m
\end{bmatrix},
$$

where $\alpha_{ij}$ are a set of coefficients, $X_i (i=1 \ldots N)$ are a set of given representative patterns, or training set, and

$$
K(\ast, X) = \begin{bmatrix}
K(X_1, X) \\
K(X_2, X) \\
\vdots \\
K(X_N, X)
\end{bmatrix},
$$

where the kernel function, $K(X_j, X), j=1 \ldots N$, is a map from a subspace of $(\mathbb{R}^n \times \mathbb{R}^n)$ to $\mathbb{R}^l$.

Therefore, developing a nonlinear feature extraction algorithm defined in Equation (5.13) is equivalent to finding a matrix $A$, which maximizes the class separability of the extracted patterns, $R$.  

116
As described in Appendix B, we know that, if our chosen kernel function, \( K(X,Y) \), satisfies the Mercer Theorem, the \( K(X,Y) \) can be reformulated as the inner product of two functional vectors. That is:

\[
K(X,Y) = \langle \Phi(X), \Phi(Y) \rangle = \Phi(X)^T \Phi(Y),
\]

where \( \Phi(X) \) is a functional vector mapping from a subspace of \( \mathbb{R}^n \) to that of \( \mathbb{R}^f \), where \( f \) is generally much larger than \( n \). For example, if we choose the \( K(X,Y) \) as

\[
K(X,Y) = \exp \{-\gamma \|X - Y\|^2\},
\]

\( f \) can even be infinite. Therefore, \( \Phi(X) \) can map the original pattern \( X \) from a relatively low dimensional space, or input space, \( I \), into a very high dimensional space, or feature space, \( F \).

Substituting Equation (5.14) into (5.13) we obtain:

\[
R = \begin{bmatrix}
\sum_{i=1}^{N} \alpha_{i,1} \Phi(X_i)^T \\
\sum_{i=1}^{N} \alpha_{i,2} \Phi(X_i)^T \\
\vdots \\
\sum_{i=1}^{N} \alpha_{i,m} \Phi(X_i)^T
\end{bmatrix} \Phi(X) = E_F^T \Phi(X)
\]

(5.16)

By comparing Equation (5.16) with (5.1), we note that the nonlinear feature extraction defined in Equation (5.13) can be reformulated as a linear feature extraction process in the high-dimensional feature space. In fact, this is a critically important observation, because adapting the linear feature extraction algorithm presented in Section 5.2 to the feature space, \( F \), thus becomes straightforward.

**Theorem 2**: Any set of \( N \) patterns in feature space, \( F \), is linearly separable, if at least \( N-1 \) patterns in the set are linearly independent, and the dimensionality of \( F \) is at least \( N-1 \).
Proof: This can be directly obtained from Theorem 1 in [95], which suggests, for a 2-class problem in a space of \( \mathbb{R}^D \), any set of \( N \) patterns with \( N \leq D + 1 \) is always linearly separable as long as at least \( N-1 \) patterns are linearly independent. However, we know that any \( L \)-class problem can be reduced to \( L \) 2-class problems, and each of the 2-class problems is a 1-to-rest classification problem.

Corollary 2.1: Any number of independent patterns is linearly separable in the feature space, \( F \), induced by the kernel function in Equation (5.15).

Proof: As shown in [94], we know that the Feature space, \( F \), induced by the kernel function in Equation (5.15), is an infinite-dimensional space. Thus, the number of linearly separable independent patterns in \( F \) can be an arbitrarily large finite number according to Theorem 2.

Theorem 2 and its Corollary 2.1 imply that distributions of different classes are generally linearly separable in feature space, \( F \), even though they might not be linearly separable in the input space, \( I \). Thus, linear feature extraction algorithms tend to demonstrate better performance in the feature space.

**Definition 1:** Kernel Matrix, \( K \), and Kernel Matrix of Class \( i \), \( K^{(i)} \), are defined respectively as:

\[
K = \Theta_{\Phi(i)}^T \Theta_{\Phi(i)} = [K(X_i, X_j)]_{N \times N} \tag{5.17}
\]

and

\[
K^{(i)} = \Theta_{\Phi(i)}^T \Theta_{\Phi(i)} = [K(X_i, X^{(i)}_j)]_{N \times N} \tag{5.18}
\]

where \( \Theta_{\Phi(i)} = [\Phi(X^{(i)}_1) \ \Phi(X^{(i)}_2) \ \cdots \ \Phi(X^{(i)}_{N_i})] \), and

\[
\Theta_{\Phi(i)} = [\Theta_{\Phi(i)}^{(1)} \ \Theta_{\Phi(i)}^{(2)} \ \cdots \ \Theta_{\Phi(i)}^{(L)}]
\]

**Definition 2:** Kernel Within-matrix, \( G_w \), for a set of given patterns \( X_i, i = 1 \ldots N \), is defined as:

118
By substituting (5.4) and (5.17) into (5.19), we can obtain:

\[
G_w = \Theta^T \Theta_{\Phi(X)} S_{\Phi(X)} \Theta_{\Phi(X)} \tag{5.19}
\]

By substituting (5.4) and (5.17) into (5.19), we can obtain:

\[
G_w = \frac{1}{N} \begin{bmatrix} N_1 I_{N_1} - I_{N_1} & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ N_L I_{N_L} - I_{N_L} \end{bmatrix} K^T \tag{5.20}
\]

**Definition 3:** *Kernel Between-matrix, G\(_b\), for a set of given patterns \(X_i, i = 1 \ldots N\), is defined as:

\[
G_b = \Theta^T_{\Phi(X)} S_{\Phi(X)} \Theta_{\Phi(X)} \tag{5.21}
\]

By substituting Equation (5.6) and Equation (5.17) into Equation (5.21), we can obtain:

\[
G_b = \frac{1}{N} KW = \begin{bmatrix} N_1 & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ N_L & \cdots & \cdots & \cdots \end{bmatrix} W^T K^T \tag{5.22}
\]

where \(W\) is defined in Equation (5.6).

**Theorem 3:** Both *Kernel Within-matrix, G\(_w\), and Kernel Between-matrix, G\(_b\), are singular matrices, and their eigenvalues are non-negative and real. Also, the \(\text{rank}(G_w) \leq N-L\), and the \(\text{rank}(G_b) \leq L-1\).

**Proof:**

According to Equation (5.20) and (5.22), both \(G_w\) and \(G_b\) can be reformulated in the form of \(AA^T\), where \(A\) is a real value matrix. Therefore, all the eigenvalues of \(G_w\) and \(G_b\) are non-negative and real-valued.

As for *Kernel Within-matrix, G\(_w\), we know that \(I_{N_1}\) only has one non-zero eigenvalue, which is \(N_1\). Matrix, \(N_1 I_{N_1} I_{N_1}\), therefore only has \(N_1-1\) non-zero eigenvalues, and matrix,
\[
\begin{bmatrix}
N_1 \mathbf{I}_{N_1} - 1_{N_1} \\
& \ddots \\
& & N_L \mathbf{I}_{N_L} - 1_{N_L}
\end{bmatrix},
\]
only has \(N-L\) non-zero eigenvalues. Therefore, according to (5.20), \(\text{rank}(\mathbf{G}^v) \leq N-L < N\).

As for Kernel Between-matrix, \(\mathbf{G}_b\), from (5.6), we know that the \(L\) columns of \(\mathbf{W}\) are linearly dependent, which can be derived by noting that \(\sum_{i=1}^{L} \mathbf{W}(\bullet, i)N_i = 0\), where \(\mathbf{W}(\bullet, i)\) is the \(i^{th}\) column of matrix \(\mathbf{W}\). Therefore, \(\text{rank}(\mathbf{G}_b) \leq L-1\). Thus, \(\text{rank}(\mathbf{G}_b) \leq L-1 < N\) according to (5.22). \(\blacksquare\)

**Definition 4:** Conditioned Kernel Within-matrix, \(\overline{\mathbf{G}}_w\), is defined as:

\[
\overline{\mathbf{G}}_w = \mathbf{G}_w + \tau \mathbf{I}
\]

where \(\tau > 0\) and is called the conditioning coefficient, and \(\mathbf{I}\) is the identity matrix.

**Theorem 4:** Kernel-based nonlinear Feature Extraction (KFE)

If we utilize criterion \(J_i\) in Equation (5.9) to quantify the class separability, the matrix \(\mathbf{A}\) in (5.13) can be formed by \(m\) eigenvectors corresponding to \(m\) biggest eigenvalues of matrix \(\overline{\mathbf{G}}_w^{-1}\mathbf{G}_b\). That is:

\[
\mathbf{A} = [a_1 \ a_2 \ \cdots \ a_m]
\]

where \((\overline{\mathbf{G}}_w^{-1}\mathbf{G}_b)a_i = \lambda_i a_i, \ i = 1\ldots m,\) and \(\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m\).

**Proof:**

In order to facilitate readers’ understanding of our derivation, we first ignore the difference between the Kernel Within-matrix, \(\mathbf{G}_w\), and the Conditioned Kernel Within-matrix, \(\overline{\mathbf{G}}_w\), and use \(\mathbf{G}_w\) in our subsequent derivation. As a matter of fact, neglecting this difference has no effect on the spirit of our proof, and introducing of \(\overline{\mathbf{G}}_w\) is mainly for implementation considerations.
If we employ the within-matrix $S_w$ as $S_2$, and the between-matrix $G_b$ as $S_1$, the separability criterion can be obtained according to Equation (5.9) as:

$$J_i(m) = tr\{S_{wR}^{-1}S_{br}\} \quad (5.25)$$

According to Theorem 1, we know that criterion $J_i$ is invariant under any nonsingular transformation. Therefore, assume the existence of a nonsingular transformation, $T$, which transforms $R$ to $Y, Y = TR$. If we choose $T = \Omega A_{m,wR}^{-1/2} \Psi$, where $A_{m,wR}$ and $\Omega$ are respectively the eigenvalue and eigenvector matrices of $S_w$, and $\Psi$ is the orthonormal eigenvector matrix of a matrix $K = A_{m,wR}^{-1/2} \Omega S_{br} \Omega A_{m,wR}^{-1/2}$, then we obtain:

$$T^TS_{br} = A_m, \text{ and at the same time, } T^TS_{wR} = I_m. \quad (5.26)$$

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_m \end{bmatrix}, \text{ and } \lambda_i \text{ are the eigenvalues of matrix, } K.$$ 

Under this transformation, $J_i$ remains the same, that is

$$J_i(m) = tr\{S_{wR}^{-1}S_{br}\} = tr\{TA_m T^{-1}\} = \sum_{i=1}^{m} \lambda_i \quad (5.27)$$

By substituting Equation (5.4), (5.6), (5.16), (5.17), and (5.19) into Equation (5.25), we obtain:

$$J_i(A; m) = tr\{(E^TS_{wR}(x) E) E^{-1}(E^TS_{br}(x) E) E\} = tr\{((A^T G_w A)^{-1} (A^T G_b A))\} \quad (5.28)$$

Now, we are trying to find a matrix, $A$, which can maximize $J_i$.

Let $\frac{\partial J_i(A; m)}{\partial A} = 0$, that is,

$$-2G_w A (A^T G_w A)^{-1} (A^T G_b A) (A^T G_w A)^{-1} + 2G_b A (A^T G_w A)^{-1} = 0, \text{ or}$$

$$(G_w^{-1} G_b) A = A ((A^T G_w A)^{-1} (A^T G_b A)) \quad (5.29)$$

By substituting Equation (5.4), (5.6), (5.16), (5.17), and (5.19) into Equation (5.29), we obtain:

$$(G_w^{-1} G_b) A = A (S_{wR}^{-1} S_{br}) \quad (5.30)$$

By substituting Equation (5.26) into Equation (5.30), we obtain:
From (5.31), we know that the diagonal elements of $\Lambda_m$ are $m$ eigenvalues of $(G_w^{-1}G_b)$, while the columns of $(AT)$ are $m$ corresponding eigenvectors of $(G_w^{-1}G_b)$. According to Equation (5.26), we know that all $\lambda_i$ are real and non-negative because they are the eigenvalues of a symmetric matrix, $K$. In order to maximize $J_i$, $\Lambda_m$ should be formed from the $m$ biggest eigenvalues of $(G_w^{-1}G_b)$ according to Equation (5.27). Therefore, $(AT)$ is formed by $m$ eigenvectors of $(G_w^{-1}G_b)$ corresponding to the $m$ biggest eigenvalues in $\Lambda_m$. Because $J_i$ is invariant under any nonsingular transformation, removing $T$ from $(AT)$ will not affect the final value of $J_i$. Thus, $A$ can be obtained by selecting $m$ eigenvectors of $(G_w^{-1}G_b)$ corresponding to the $m$ largest eigenvectors of $(G_w^{-1}G_b)$. However, according to Theorem 3, $G_w$ is a singular matrix, and is therefore not invertible. In order to make $G_w$ invertible, we have to replace it with the Conditioned Kernel Within-matrix, $\overline{G_w}$. Therefore, if we form $A$ by selecting $m$ eigenvectors of $(\overline{G_w^{-1}}G_b)$ corresponding the $m$ largest eigenvectors of $(\overline{G_w^{-1}}G_b)$, the maximum class separability can be achieved with respect to the criterion of $J_i$.

The next topic is to discuss the role the conditioning coefficient $\tau$ in Equation (5.23) plays in this Kernel-based nonlinear Feature Extraction (KFE) algorithm. From Equation (5.23) we know that the larger $\tau$ is, the more significantly $\overline{G_w}$ deviates from the original $G_w$, while $\overline{G_w}$ becomes better conditioned, which makes computing the inverse matrix of $\overline{G_w}$ more numerically stable. Therefore, $\tau$ should be properly chosen to achieve an optimal trade-off between theoretical accuracy and numerical stability.
During the implementation of the KFE algorithm, we adopt an adaptive method to choose the conditioning coefficient \( \tau \) which is presented in Theorem 5.

**Theorem 5:** The conditioning coefficient \( \tau \) in Equation (5.23) can be chosen as:

\[
\tau = \max \left\{ \frac{\lambda_{\max,G_w}}{C_{\text{allow}}}, \tau_{\text{min}} \right\} 
\]

(5.32)

where \( \lambda_{\max,G_w} \) is the maximal eigenvalue of \( G_w \), \( \tau_{\text{min}} \) is a predetermined minimal conditioning coefficient, \( \tau \) which ensures that \( \overline{G}_w \) does not significantly deviate from \( G_w \), and \( C_{\text{allow}} \) is a predetermined value chosen so that any matrix whose condition is less than this value can be numerically stably inverted under the available computation precision.

**Proof:**

From (5.23) we know that the condition of \( G_w \) and \( \overline{G}_w \) are \( C_{G_w} \) and \( C_{\overline{G}_w} \) respectively, which can be calculated as:

\[
C_{G_w} = \frac{\lambda_{\max,G_w}}{\lambda_{\min,G_w}} \quad (5.33)
\]

\[
C_{\overline{G}_w} = \frac{\lambda_{\max,\overline{G}_w}}{\lambda_{\min,\overline{G}_w}} = \frac{\lambda_{\max,G_w} + \tau}{\lambda_{\min,G_w} + \tau} \quad (5.34)
\]

Let \( C_{\overline{G}_w} = C_{\text{allow}} \), and substitute it into Equation (5.33) and (5.34), we obtain:

\[
\tau = \frac{\frac{\lambda_{\max,G_w}}{C_{G_w}} - C_{\text{allow}}}{C_{G_w}(C_{\text{allow}} - 1)} \quad (5.35)
\]

By taking into consideration of the facts that \( C_{\text{allow}} \gg 1 \), and \( C_{G_w} \gg C_{\text{allow}} \), Equation (5.35) becomes:

\[
\tau \equiv \frac{\lambda_{\max,G_w}}{C_{\text{allow}}}
\]

Thus, \( \tau = \max \left\{ \frac{\lambda_{\max,G_w}}{C_{\text{allow}}}, \tau_{\text{min}} \right\} \).
Basically, Theorem 5 only provides an approximate estimate of the conditioning coefficient, \( \tau \), and it is not necessary to calculate coefficient \( \tau \) exactly. In fact, we can further reduce the computation by replacing \( \lambda_{\max,G_w} \) in Equation (5.32) with its estimate. For example, we can replace \( \lambda_{\max,G_w} \) with the 1-norm of \( G_w \) in practice. The rationale behind this replacement is that the 1-norm of \( G_w \) should somewhat reflect the scale of the 2-norm of \( G_w \), which is exactly \( \lambda_{\max,G_w} \).

As for the choice of the Kernel function, \( K(X,Y) \), as long as it can be reformulated in the form of Equation (5.14), it is valid to serve as the Kernel function in this KFE algorithm, and is not limited to the form of Radial Based Function (RBF) in Equation (5.15). Another typical form of Kernel function is the polynomial kernel function:

\[
K(X,Y) = (\langle X, Y \rangle + 1)^d = (X^TY + 1)^d
\]  

(5.36)

In fact, if we define the Kernel function as a linear inner product between two vectors,

\[
K(X,Y) = \langle X, Y \rangle = X^TY
\]  

(5.37)

the KFE algorithm presented in Theorem 4 can be reduced to the linear feature extraction algorithm presented in Section 5.2. Therefore, from this perspective, the conventional linear feature extraction algorithm is only a special case of our KFE algorithm.

Finally, it is worth to mentioning that, according to Theorem 3, the \( \text{Rank}(G_{ib}) \leq L-I \), where \( L \) is the number of classes involved in the problem. Thus, this KFE algorithm can only extract a maximum of \( L-I \) meaningful features from the original patterns, \( X \), even if \( m \) is set larger than \( L-I \). That is, the dimensionality of the extracted patterns, \( R \), in Equation
(5.13) can maximally be $L-1$. For example, for a 2-class problem, only one feature can be extracted, and $R$ is thus a scalar.

### 5.4 Modified Kernel-based Feature Extraction

As we mentioned in Section 5.3, the KFE algorithm can only extract $L-1$ features, where $L$ is the number of classes involved in the problem. This limitation dramatically degrades the flexibility of this KFE algorithm. Therefore, here we attempt to modify the algorithm to allow it to extract up to $N$ features, where $N$ is number of patterns in the training set. This is achieved by modifying the formulation of the kernel between-matrix $G_b$ to increase its rank.

In order to simplify the following derivation, we assume the underlying problem is a 2-class problem. In fact, the fundamental idea can be readily expended to multi-class problems.

Fortunately, another result from statistical discriminant analysis, named the nonparametric Between-matrix, $S_{nb}$ provides us with an alternative to the between-matrix $S_b$, and the rank of $S_{nb}$ is only decided by the internal structure of the patterns in the training set, and is not limited by the number of classes involved.

The Nonparametric Between-matrix $S_{nb}$ is defined as:

\[
S_{nb,x} = P_1E\{w_{x,k}[X - M^{(2)}_k(X)][X - M^{(2)}_k(X)]^T | \omega_1\} + P_2E\{w_{x,k}[X - M^{(1)}_k(X)][X - M^{(1)}_k(X)]^T | \omega_2\}
\]  

(5.38a)

where
\[
\mathbf{M}_i^{(j)}(\mathbf{X}) = \frac{1}{k} \sum_{i=1}^{k} \mathbf{X}_{i_{\text{NN}}}^{(j)}, \text{ where } \mathbf{X}_{i_{\text{NN}}}^{(j)} \text{ is the } i^{\text{th}} \text{ Nearest Neighbor (NN) in class } i \text{ to the pattern } \mathbf{X}, \text{ which is in the class different from class } i.
\]

\[
w_{X,k} = \frac{1}{\beta_{X,K} + 1}, \text{ where } \beta_{X,K} = \frac{\max(||\mathbf{X} - \mathbf{X}_{k_{\text{NN}}}^{(1)}||^2, ||\mathbf{X} - \mathbf{X}_{k_{\text{NN}}}^{(2)}||^2)}{\min(||\mathbf{X} - \mathbf{X}_{k_{\text{NN}}}^{(1)}||^2, ||\mathbf{X} - \mathbf{X}_{k_{\text{NN}}}^{(2)}||^2)}.
\]

(5.38b)

This \(w_{X,k}\) is a coefficient that more heavily weights the patterns falling in the boundary area between two classes. When pattern \(X\) is in the boundary area, \(w_{X,k}\) is close to \(\frac{1}{2}\), and when \(X\) is far away from the boundary area, \(w_{X,k}\) approaches zero.

The estimate of \(S_{\text{nb}}\) can therefore be expressed as:

\[
S_{n_{b,x}} = \frac{1}{N} \sum_{i=1}^{N} w_{X,k}^{(1)} \left[ (\mathbf{X}_i^{(1)} - \mathbf{M}_i^{(2)}(\mathbf{X}_i^{(1)})) \left[ (\mathbf{X}_i^{(1)} - \mathbf{M}_i^{(2)}(\mathbf{X}_i^{(1)}))^T \plus \frac{1}{N} \sum_{i=1}^{N} w_{X,k}^{(2)} \left[ (\mathbf{X}_i^{(2)} - \mathbf{M}_i^{(1)}(\mathbf{X}_i^{(2)})) \left[ (\mathbf{X}_i^{(2)} - \mathbf{M}_i^{(1)}(\mathbf{X}_i^{(2)})) \right]^T \right]\right]
\]

(5.39)

**Definition 5:** The *Kernel Nonparametric Between-matrix* \(G_{\text{nb}}\), for a set of given patterns \(\mathbf{X}_i\), where \(i = 1 \ldots N\), is defined as:

\[
G_{\text{nb}} = \Theta_{\phi(\mathbf{X})}^T S_{n_{b,\phi(\mathbf{X})}} \Theta_{\phi(\mathbf{X})}
\]

(5.40)

where \(\Theta_{\phi(\mathbf{X})}\) is defined in (5.18).

By substituting Equation (5.14) and Equation (5.39) into Equation (5.40), we obtain:
where

\[ \mathbf{D} = \begin{bmatrix} d_1^{(1)} & \cdots & d_i^{(1)} & \cdots & d_{N_2}^{(1)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ d_1^{(2)} & \cdots & d_i^{(2)} & \cdots & d_{N_2}^{(2)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ d_1^{(3)} & \cdots & d_i^{(3)} & \cdots & d_{N_2}^{(3)} \end{bmatrix}, \]

and

\[ \rho_{l,k}^{(i)} = \frac{1}{\beta_{X_{l}^{(i)},k} + 1}, \]

where \( \beta_{X_{l}^{(i)},k} = \frac{\max(r_K(X_{l}^{(i)} - X_{k}^{(i)}), r_K(X_{l}^{(i)} - X_{k}^{(2)})]}{\min(r_K(X_{l}^{(i)} - X_{k}^{(i)})]} \), and

\[ r_K(X_{l}^{(i)} - X_{k}^{(i)}) = K(X_{l}^{(i)}, X_{l}^{(i)}) + K(X_{k}^{(i)}, X_{k}^{(i)}) - 2K(X_{l}^{(i)}, X_{k}^{(i)}) \]

Therefore, the Modified Kernel-based nonlinear Feature Extraction (MKFE) can be obtained simply by replacing the \( \mathbf{G} \) in the KFE algorithm with \( \mathbf{G}_{nb} \), as defined in (5.41).

**Theorem 6:** Modified Kernel-based nonlinear Feature Extraction (MKFE)

If we employ criterion \( J_1 \) in Equation (5.9) to quantify the class separability, the matrix \( \mathbf{A} \) in Equation (5.13) can be formed by the \( m \) eigenvectors corresponding to the \( m \) biggest eigenvalues of matrix \( \mathbf{G}_{w}^{-1} \mathbf{G}_{nb} \). That is:

\[ \mathbf{A} = [a_1 \ a_2 \ \cdots \ a_m] \]
where \((\overline{G}^{-1}G_{nb})a, i = 1...m, \) and \(\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_m\).

It is interesting to note that, when we obtain the Matrix A according to Theorem 6, we also automatically acquire a list of corresponding eigenvalues \(\lambda_i, i = 1...m\), which are supposed to be real and non-negative. In fact, the values of these eigenvalues can serve as quantitative indications of the relative importance of the features extracted, and can be used as the weights in subsequent algorithms that require assigning a weighting value for each feature, such as the weighted Euclidean distance.

At this point, we have finished our derivation of the \textit{Kernel-based nonlinear feature extraction algorithm}. Note that this algorithm has a group of parameters used to adjust its behavior, which include (1) the type of kernel functions, e.g., an \textit{inner product}, a \textit{polynomial kernel}, or an \textit{RBF kernel}; (2) the parameters of the selected kernel function, e.g., the degree of the polynomial kernel, and the \(\gamma\) of the RBF kernel; (3) \(k\), which is involved in forming the \textit{Kernel Nonparametric Between-matrix} \(G_{nb}\) in Equation (5.38), and (4) the number of features to extract, \(m\). How these parameters affect the extracted features, and how to find a set of optimal parameters for different application scenarios are still an open topic. However, experiments suggest a generally applicable parameter-selection scheme, which is to use the RBF kernel with \(\gamma = 1\), and \(k = 3\). In most cases, this set of parameters produces high performance features.

5.5 Experiments

In order to illustrate the efficacy of our FKE and MKFE algorithms, experiments based on both synthetic and measured patterns have been designed and implemented. In all
of these experiments, the algorithm parameters are set according to the parameter-selection scheme mentioned in Section 5.4, if not otherwise specified.

In Sub-section 5.5.1, a series of experiments based on synthetic patterns are implemented, and the results are plotted. Because our original motivation in devising new feature extraction algorithms stems from the application of HRR-based ATR, in Subsection 5.5.2, our MKFE algorithm is applied to measured HRR signatures, and comparison between classification rates obtained from both the original HRR signatures and the extracted patterns is conducted.

5.5.1 Experiments Based on Synthetic Data

As we have mentioned previously, the proposed KFE and MKFE algorithms are capable of extracting both linear and nonlinear features by adopting different form of kernel functions. When the kernel function is defined as Equation (5.37), which is inner product of two vectors, the KFE algorithm will extract linear features. In contrast, if the kernel function is defined as either the RBF (shown in Equation (5.15)), or the polynomial kernel (shown in Equation (5.36)), the KFE algorithm will be able to extract nonlinear features. In order to illustrate this point, and also to help readers to better understand the difference between linear and nonlinear feature extraction, our first synthetic problem is illustrated in Figure 5.1.
This is a 2-class problem, and each pattern only has 2 features. That is, they are 2-element vectors. After applying the MKFE algorithm with a kernel function of inner-product form, we find that there are only two non-zero eigenvalues in Equation (5.42), which suggests that only two meaningful features can be extracted from this synthetic problem. This result is in accordance with our expectations. The Feature Extraction (FE) functions are shown in Figure 5.2, while the distribution of the extracted patterns, which are formed by the 2 extracted features, is presented in Figure 5.3. Based on these plots, the following two observations are obtained: (1) linear feature extraction functions are in the form of hyperplanes; and (2) linearly extracted features are not very efficient in improving class separability for this problem.
Figure 5.2 (a) 3-D plot of the linear FE function for the first extracted feature; (b) Gray-scale representation of FE function in (a), overlapping with the data; (c) 3-D plot of the linear FE function for the second extracted feature; (d) Gray-scale representation of FE function in (c), overlapped with the data.
Figure 5.3 Distribution of extracted patterns, which are formed by 2 extracted features

Figure 5.4 (a) 3-D plot of the RBF-base FE function for the first extracted feature; (b) Grayscale representation of FE function in (a), overlapped with the data
Next, we apply the MKFE algorithm with a RBF kernel to the same problem, and obtain many non-zero eigenvalues in the Equation (5.42). That is, we can extract many meaningful features from the original pattern set. Compared with the results from the linear case, this is indeed a very interesting observation, which suggests that the number of features extractable by our MKFE algorithm (with a nonlinear kernel) can be larger than the original number of features. In fact, this breaks a fundamental limitation embedded in all the feature extraction algorithms based on linear operations. Because the rank of a pattern set is always equal or less than the dimensionality of the patterns, all the linear FE algorithms can only maximally extract $N$ new features, providing that the original pattern dimensionality is $N$. Therefore, the common notion that FE reduces the pattern dimensionality is no longer valid for our MKFE algorithm. Similar observations and comments can also be found in other nonlinear FE algorithms, such as KPCA in [94]. Meanwhile, although many of the eigenvalues are nonzero, there is only one dominant eigenvalue, which suggests that it is possible to preserve the major discriminant information by only extracting the most critical feature using our MKFE algorithm. The resultant nonlinear FE function is presented in Figure 5.4, and the distribution of the

Figure 5.5 Distribution of extracted patterns, which are formed by extracting only one feature
extracted 1-feature patterns is shown in Figure 5.5. Note that (1) the FE function in this case no longer complies with the form of hyperplanes; and (2) the one extracted feature is sufficient to roughly distinguish the two underlying classes. From Figure 5.4, we know that it is the unlimited variation of nonlinear FE functions that provides nonlinear FE algorithms much greater capabilities in extracting high performance discriminant features.

The second synthetic problem is also a 2-class problem. The distribution of the training patterns is illustrated in Figure 5.6. As we know, most of the available linear FE algorithms fail to extract meaningful features given this kind of distribution. However, our MKFE algorithm (using a RBF kernel) can cleverly find a set of FE functions (see Figure 5.7), which can extract highly separable features. Figure 5.8 (a) illustrates the distribution of the extracted patterns formed by the first extracted feature, or extracted feature 1, while Figure 5.8 (b) presents the distribution of extracted patterns formed by 2 extracted features. From these plots we conclude that one dominant extracted feature can preserve most of the discriminating information of a given 2-class problem, although sometimes more features are likely to facilitate higher class separability.
Figure 5.6 A 2-class problem composed by 2-feature patterns
Figure 5.7 (a) 3-D plot of the RBF-based FE function for the first extracted feature; (b) Gray-scale representation of FE function in (a), overlapped with the data; (c) 3-D plot of the RBF-based FE function for the second extracted feature; (d) Gray-scale representation of FE function in (c), overlapped with the data.
Figure 5.8 (a) Distribution of new patterns, which are formed only by the dominant feature; (b) Distribution of new patterns, which are formed by 2 extracted features

The last synthetic problem is a 3-class problem, which is demonstrated in Figure 5.9. We utilize the KFE algorithm with a RBF kernel in this problem. As we mentioned previously, for a 3-class problem, the KFE algorithm can extract 2 features that can maintain most of the discriminating information of the original pattern set. The FE functions for these two features are presented in Figure 5.10. The distribution of the new patterns, which are formed by these two extracted features, is shown in Figure 5.11. From Figure 5.11, we can observe that the two extracted features are sufficient to clearly separate the 3 classes in this problem.
Figure 5.9 A 3-class problem composed by 2-feature patterns
Figure 5.10 (a) 3-D plot of the RBF-based FE function for the first extracted feature; (b) Gray-scale representation of FE function in (a), overlapped with the data; (c) 3-D plot of the RBF-based FE function for the second extracted feature; (d) Gray-scale representation of FE function in (c), overlapped with the data.
5.5.2 Experiments Based on Measured HRR signatures

In this subsection, our MKFE algorithm is applied to measured HRR signatures to test its efficacy experimentally. Generally speaking, HRR signatures have many features (around 35), the returns are corrupted by ground clutter, they are very sensitive to many radar and target parameters, and therefore, demonstrate extremely irregular behavior. As a result, the distribution of HRR signatures from interesting targets is quite variable, and it is hard to verify that it complies with normal, or any other known distributions. Moreover, the border region between the HRR signatures from different targets, or classes, is nonlinear, and sometimes very erratic. These characteristics of HRR signatures unfortunately rule out
most of the existing classification-oriented feature extraction methods [99]. Indeed, this is one of the reasons why we developed this MKFE algorithm.

According to Theorem 6, the features extracted by our MKFE algorithm are the best available features under the criterion \( J_i \) in Equation (5.9). However, an intrinsic problem with criterion \( J_i \) is that its relationship with the Bayes classification error, \( P_e \), is not straightforward. Therefore, in this subsection, we adopt an alternative methodology to testify to the quality of the extracted features by relating them to the classification error, \( P_e \). This method uses specific classifiers to classify both the original HRR signatures and the new patterns formed by extracted features, and employs the resultant classification rates as an indirect performance measurement of the features obtained by our MKFE algorithm. Therefore, it is worth mentioning that the conclusions drawn in this context are sometimes classifier-dependent.

The experiment is designed as follows:

1. We choose 7182 HRR signatures from each of the 3 targets, \( T72, 2S1, \) and \( BMP2 \). These are equally split into 18 groups according to their aspect angle, with each group covering an aspect angle range of 10 degrees. Therefore, each group has 399 signatures.

2. We select two groups covering the same aspect angle range from two different targets, and combine them to form a 2-class problem with 798 signatures. There are a total of 48 of this kind of 2-class problems, because there are 3 different target pairs, and for each pair, there are 18 different aspect angle ranges.

3. For each 2-class problem obtained in (2), we randomly choose up to \( P \) percentage of total signatures to form the training set for this problem, and utilize half of the
remaining signatures to form the testing set. This is called one realization of the training set and testing set. This random process is repeated 100 times to form 100 different realizations of the training set and testing sets for this problem.

(4) For the $i^{th}$ ($i = 1...100$), realization obtained in (3), we use the training set to construct a RBF-based SVM nonlinear classifier, and apply the testing set to the constructed SVM classifier to obtain the classification rate for this realization, $O(i)$. (For details of the SVM classifiers, please refer to Appendix B.) Meanwhile, we use the MKFE algorithm on the training set to find an RBF-based FE function that is capable of extracting the dominant feature, and generate a new testing set by applying this FE function to each of the HRR signatures in the original testing set. Because all of the patterns in this new testing set are only 1-feature scalars, a proper threshold can serve as a simple linear classifier. The classification rate, $F(i)$, obtained in this way is used to represent the performance of MKFE algorithm.

(5) We average $O(i)$ and $F(i)$ over all 100 realization and 3 different target pairs to obtain the mean classification rates for the original HRR signatures and the MKFE-induced patterns respectively. The resultant classification rates are plotted against aspect angles in Figure 5.12 (a), (b), and (c) for specific training set percentage, $P=15\%$, $30\%$, and $45\%$ respectively. Finally, the overall mean classification rates obtained by averaging over all aspect angles are plotted against training set percentage, $P$ in Figure 5.12 (d).

From the plots in Figure 5.12, we observe that:

(1) The experimental results suggest that our MKFE algorithm is very efficient in extracting high performance features. Especially from Figure 5.12 (d), we can see
that a simple linear classifier based on the 1-feature MKFE-induced patterns can even outperform a complex SVM classifier based on the original high dimensional signatures. In fact, this seldom happens for linear FE algorithms, because linear algorithms generally reduce the number of features at the cost of losing discrimination information.

(2) The complexity of FE algorithms and classifiers are interchangeable. Similar classification rates can be achieved by adopting either a sophisticated FE algorithm with a simple classifier, or a simple FE algorithm with a complicated classifier.
Figure 5.12 (a) Mean classification rates vs. aspect angle when train set percentage is 15%;
(b) Mean classification rates vs. aspect angle when train set percentage is 30%;
(c) Mean classification rates vs. aspect angle when train set percentage is 45%;
(d) Overall mean classification rates vs. train set percentage.
Note that we only use the 1-feature patterns in the above experiment. However, our MKFE algorithm is capable of extracting many features for a given 2-class problem. Will the new patterns formed by these multiple-features have better classification performance? In order to study this, we repeat the above experiment 24 times by changing the number of extracted features from 1 to 24 with an incremental step of 1. The only adjustment is that we also use RBF-based SVM classifiers to classify the new patterns, instead of the threshold-based simple linear classifier for the 1-feature pattern case. The final results are illustrated in Figure 5.13. This plot suggests some interesting issues. When the training set percentage is low, such as shown for 15%, the classification rate gradually increases with the increase of number of extracted features. However, when the training set percentage is high, such as 45%, a larger number of extracted feature can even results in worse classification performance. Although this observation may be classifier-dependent, it still at least conveys a very important message. Patterns formed with more extracted features using our MKFE algorithm are not necessary "better" than patterns formed with less features for a given classifier.
5.6 Conclusions

A novel nonlinear kernel-based feature extraction algorithm, named the KFE algorithm, has been proposed in this chapter. This algorithm, with the help of the "kernel trick", is directly extended from its linear counterpart, and is thus theoretically founded. Also, the algorithm's efficacy is experimentally verified by both synthetic data and measured data. From this perspective, what is presented in this chapter is a relatively complete work. However, in order to make this algorithm more practical, there are still a couple of open topics. These include how to refine this algorithm to make it less computationally
demanding, and how to theoretically guide users to select a set of optimal algorithm parameters given a particular problem.
CHAPTER 6

CONCLUSIONS

In this dissertation we have focused on two aspects of a feature-oriented study of HRR-based ATR by 1) analyzing the properties of HRR signatures and by 2) developing feature extraction algorithms for HRR data.

Toward these ends, we have completed three major research thrusts in the area of HRR signature analysis. Our goal was to provide the reader with a deeper understanding of the fundamental properties of HRR signatures by examining the difference between complex HRR signatures and magnitude-only HRR signatures. First, we have conducted a theoretical study based on a simple, but representative, HRR signature model that reveals some fundamental insights into scintillation effects that play a critical role in assessing the complex behavior of HRR signatures. Second, the phase utility of complex HRR signatures was evaluated, and our results suggest that while the phase of complex HRR signatures may be meaningful if a sequence of complex HRR signatures are taken into consideration, the phase of a single isolated HRR signature has limited utility. Third, we have carried out a group of experiments using measured HRR signatures to experimentally test the phase utility of complex HRR signatures, and the results are in accordance with the conclusions.
obtained via our theoretical analysis. We believe that these results have contributed significantly to an understanding of HRR signatures.

Additionally we have developed an elegant methodology that supports the development of feature extraction algorithms. Our research has culminated in two new feature extraction algorithms. This work is generally considered as a model-based feature extraction algorithm. This is because these algorithms are based on a particular physics-based radar model; the features are the parameters of the model, and the feature extraction is equivalent to estimation of the parameters of the model. Our contributions include a new HRR moving-target model, and the feature extraction algorithms we have proposed are based on this derived model. Compared with currently available models, the model we propose has a number of advantages. For example, our model presumes no assumptions on the distribution of the clutter, and it is uniformly applicable to a wide range of moving target application scenarios.

As we note above, the new feature extraction algorithms are based on our parameter estimation algorithms arising from this model. These two algorithms have advantages and disadvantages, depending on their potential applications. For example, the first algorithm estimates the model parameters more accurately under the condition of moderate SNR, but requires the HRR data to match the HRR model with some fidelity to achieve good performance. In contrast, the second algorithm has lower resolution in estimating model parameters, but can tolerate considerably more model mismatch.

It is worth noting that these contributions are significant in another context. Conventionally, most algorithms developed in the area of HRR-based ATR were based on the use of 1D HRR signatures. However, due to the complex behavior of HRR signatures,
these algorithms are generally not robust. We propose a different scheme for processing HRR data - we avoid the use of HRR signatures, and we extract features directly from high dimensional, raw HRR. We have shown that this new technique can dramatically reduce scintillation effect, which plagues the conventional algorithms.

Of course, there are still practical issues that hinder the direct application of both the models and the algorithms we have developed in their application to a fieldable ATR system. For example, due to the limited availability of measured high dimensional, raw HRR data, the proposed model and algorithms have not been sufficiently experimentally verified. Also, some of our algorithms are computationally demanding, and it thus is necessary to significantly reduce their computational complexity before they become feasible for a fieldable ATR system.

Although we have shown that these proposed feature extraction algorithms are complete and work reasonably well, they are inherently designed to extract representative features, which might not be optimal for classification applications. Therefore, we have also developed an entirely new feature extraction algorithm that is capable of extracting classification-oriented features. The key technique employed in this algorithm is generally referred to as the “kernel trick”, and the algorithm is therefore named the Kernel-based Feature Extraction (KFE) algorithm. This algorithm introduces a general form of nonlinear feature extraction, and can deal with multiple-class problems. We have proven that the features extracted using this algorithm maximally retain the class separability. Additionally, our experimental results also illustrate the performance of this algorithm. Indeed, the KFE algorithm is a general feature extraction algorithm, and is not limited to application to HRR-based ATR. However, due to very promising performance, theoretical foundation,
and ease of use, we are eager to see its extensive applications in various pattern recognition areas in the future. We note that further study is necessary to reduce its computational complexity, which would make this technique more widely acceptable.

It is clear that feature studies for HRR-based ATR is a very challenging topic, and a topic that needs more investigation. With respect to the research results presented in this dissertation, we suggest that additional investigations are needed to produce short-term results and/or to strengthen the consequences of the contributions presented here. Some of the topics that might warrant immediate future work are:

1. Feature extraction algorithms play a critical role in the phase-utility experiments in Chapter 2. The conclusions drawn there can be made more persuasive if additional types of features are utilized in the experiments.

2. In Chapter 5 we employ a scatter-matrix discriminant criterion to derive our KFE algorithm. It would be very worthwhile to try different discriminant criteria and then compare their final performance.

3. Also in Chapter 5, we introduced a conditioned matrix to solve the singularity problem, thereby making the corresponding matrix invertible. However, there are alternative solutions to this problem, and it also worth determining which alternative solutions are amenable to implementation in this framework, and what corresponding effects these solutions have on the performance of our KFE algorithm.

4. Finally, again in Chapter 5, our MKFE algorithm solves only two-class problems. An obvious, and vital, research extension is to modify this algorithm to deal with multi-class problems.
APPENDIX A

FEATURE EXTRACTION PROCESSING IN PHASE-UTILITY EXPERIMENTS

Feature Extraction is the procedure used in Experiments #1 and #2 in Section 3.2 of Chapter 3 to generate magnitude-only (complex) Feature Vectors (FVs) from 1-D magnitude-only (complex) signatures. The details of the process are as follows:

For each signature (complex or magnitude-only) at aspect angle $A^\circ$:

**Input:** 1-D complex or magnitude-only signature at aspect angle $A^\circ$

**Parameters:** $SC_N$, the number of scattering centers

**Output:** FV at aspect angle $A^\circ$. For the magnitude-only case, the FV is a $2*(SC_N-1)$-by-1 vector, while for the complex case the FV is a $3*(SC_N-1)$-by-1 vector.

**Procedure Description:**

(1) Pick $SC_N$ local maximums, which are assumed to be the scattering centers, from the signature and save their values and locations. For the complex signatures, use the magnitude as the criterion to select the scattering centers, but save both the real and imaginary part of the complex scattering centers, along with their locations.

(2) Sort all the scattering centers in descending order according to their values (their magnitudes for complex signature).
(3) Compute the difference in values and locations between the first scattering center, or the "brightest" scattering center, and all the other scattering centers. That is, compute the relative values and locations between the "brightest" scattering center and the other scattering centers. Store the resultant relative values and locations as the $FV$ at aspect angle $A^\circ$. 
INTRODUCTION TO SUPPORT VECTOR MACHINES

In this appendix we briefly introduce the fundamental ideas and basic formulations of Support Vector Machines (SVMs). First, we present the historic development of SVMs in Section B.1, and their basic concepts are expressed in B.2. In B.3 we provide a simple derivation of the three basic formulations of SVMs, which are (1) linear SVMs for separable data, (2) linear SVMs for non-separable data, and (3) nonlinear SVMs used for non-separable data.

B.1 Historic Background

In the 60s a group of Russia researchers, including Vapnik, Chervonenkis, and Lerner, developed a linear pattern recognition algorithm named the Generalized Portrait algorithm[91]. This algorithm is firmly based on Statistical Learning Theory [92], or VC Theory, which characterizes the generalization ability of knowledge learned from a given set of data to unseen data. At that time, the algorithm was only capable of generating linear decision boundaries, and thus, attracted modest attention.

One of the breakthrough developments of SVMs occurred in 1992, when Boster, Guyon, and Vapnik, a group of researchers at AT&T Bell Labs, generalized the original
linear algorithm to the nonlinear case by employing a technique, which was later referred to as the "Kernel trick"[108]. This "trick" cleverly replaces the computationally prohibitive high dimensional vector dot product with a simple evaluation of a low dimension kernel function. We present this kernel trick in details in the Section B.3.

Since 1996, several fast implementation algorithms of SVMs have been proposed [109, 110, 111, 112], and successful application stories have been presented in different areas, such as Optical Character Recognition (OCR) [113], object recognition [114], facial image detection [115], text categorization [116], spam categorization [102], image classification [103], etc. These theoretical and practical advances have jointly resulted in an increased attention on SVMs. Currently, SVM research on both theoretical and application issues are being conducted all over the world.

One point should be clarified. Although the SVMs were first invented and developed for pattern recognition, they have also demonstrated outstanding performance in the application of regressive estimation and time series prediction [96]. However, because our research interests are focused on the utility of SVMs in pattern recognition, we will not address the regression estimation formulation of SVMs in this appendix.

B.2 Learning From Data: SVMs Basics

Conventional pattern recognition techniques are generally based on intensive knowledge of the pattern distribution, such as the probability density function. Unfortunately, estimating distributions from limited number of patterns is an ill posed problem, and is almost impossible if the input pattern dimensionality becomes high. One
widely adopted solution to this problem is the introduction of standard distribution models with a limited number of parameters. Thus, estimation of the distribution is reduced to a problem of estimating the model parameters from the input patterns. However, this treatment implies that the real distribution of the problem should follow, or at least approximate, the distribution described by the adopted model. One of the most frequently employed distribution models is the normal distribution. Therefore, many conventional classification techniques are heavily dependent on a normal distribution assumption. We use Figure B.1 to illustrate this idea.

Figure B.1 Methodology behind many conventional pattern recognition techniques

Note from Figure B.1 that this conventional methodology turns a relative straightforward problem, which is to find a classification boundary, into a much more challenging sub-problem, which is to estimate the whole distribution behind the given input patterns. Moreover, this treatment deviates considerably from our understanding of human learning, which focuses on the boundary regions of the input patterns.
SVMs are a set of new techniques, which learn directly from a given data set, or training examples\(^1\). The maxim behind SVMs is a statement given by Vladimir N. Vapnik, one of the founders of the statistical learning theory:

"If you possess a restricted amount of information for solving some problem, try to solve the problem directly and never solve a more general problem as an intermediate step. It is possible that the available information is sufficient for a direct solution but is insufficient for solving a more general intermediate problem." [92]

Therefore, the formal definition of a two-class problem that can be addressed by SVMs is presented as follows:

**Given:** A set training examples, \( T = \{(x_1, y_1), ..., (x_k, y_k)\} \), where \( x_i \) is the input pattern, and \( x_i \in \mathbb{R}^n \), \( y_i \) is the pattern class label, and \( y_i \in \{-1, 1\} \), where \( i = 1 ... k \).

**Find:** the decision function, \( f(x) \), which minimizes the misclassification rate of the underlying distribution.

### B.3 Different Formulations of SVMs

SVMs are capable of finding two categories of classification boundaries from a set of given training examples. One category is the linear boundary; the other is the nonlinear boundary. With respect to the linear boundaries, the underlying problem can be further classified into the linearly separable case and the linearly inseparable case. Therefore, there are a total of three different basic formulations of SVMs: (1) linear SVMs for linearly separable data, (2) linear SVMs for linearly inseparable data, and (3) nonlinear SVMs for use on either type of data.

---

\(^1\) Input data, input examples, training examples are exchangeable terms in our proposal.
We derive and present these three basic formulations in B.3.1, B.3.2, and B.3.3, respectively. In B.3.4, we list some recently developed formulations of SVMs. Each of these offers attractive properties under certain application scenarios.

**B.3.1 Linear SVMs for Linearly Separable Data**

A linear decision function, \( f(x) \), which is a hyperplane when \( x \) is a vector, can be expressed as:

\[
  f(x) = \text{sgn}\left( <w, x> + b \right)
\]  

(B.1)

where \( w \) is normal to the hyperplane, and \( <a, c> \) denotes the dot product of vectors \( a \) and \( c \), which can also be expressed as \((a*c)\). The classification boundary is, therefore,

\[
  <w, x> + b = 0
\]  

(B.2)

Therefore, finding the optimal linear boundary is equivalent to finding the optimal value of \( w \) and \( b \) to minimize the misclassification rate.

The formulation of the linear SVMs for separable data can be presented as follows:

\[
  \text{minimize } \frac{1}{2} \|w\|^2
\]  \hspace{1cm} (B.3a)

\[
  \text{subject to } y_i(<w, x_i> + b) \geq 1
\]  \hspace{1cm} (B.3b)

A simple geometric explanation of the rationale behind this SVMs formulation is provided in Figure B.2. This interpretation considers the proposed linear SVMs as a maximum margin classifier.
Figure B.2 Geometric interpretation of the formulation of linear SVMs for separable case

From Figure B.2 we can see that the two hyperplanes, \( \langle w, x \rangle + b = -1 \) and \( \langle w, x \rangle + b = 1 \), define a region, which is called the margin, and the physical meaning of the linear constraint in Eq. (B.3b) is to make sure that all the training examples are outside the margin region. This constraint can only be strictly satisfied for the linearly separable data. Also, it is easy to derive that the width of the margin region is \( 2 / \| w \| \), where \( \| w \| \) represents the Euclidean norm of vector \( w \). The wider the margin, the better the corresponding linear boundary, because the margin can be intuitively regarded as a confidence region of the classifier. Minimizing \( \frac{\| w \|^2}{2} \) is equivalent to maximizing the margin width. Thus, the formulation defined in Eq. (B.3) ensures that the optimal classification boundary is the linear boundary that achieves the widest margin, and therefore, the best potential training generalization given the training examples.
In order to solve Eq. (B.3) we first resort to its Lagrangian formulation by introducing a set of nonnegative Lagrange multipliers, \( \alpha = [\alpha_1, \cdots, \alpha_k]^T \geq 0 \).

\[
L(w, \alpha, b) = \frac{1}{2} ||w||^2 - \sum_{i=1}^{k} \alpha_i \{y_i(<w, x_i> + b) - 1\} \tag{B.4}
\]

By requiring the gradient of \( L \) with respect to \( w \) and \( b \) to vanish, we obtain the following relationships:

\[
\frac{\partial L}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^{k} \alpha_i y_i x_i \tag{B.5}
\]
\[
\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{k} \alpha_i y_i = 0
\]

By substituting (B.5) into (B.4), we can obtain the "dual" formulation of the primal optimization problem (B.3).

\[
\text{maximize } D(\alpha) = \sum_{i=1}^{k} \alpha_i - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{k} \alpha_i \alpha_j y_i y_j <x_i, x_j> \tag{B.6a}
\]

\[
\text{subject to } \alpha \geq 0, \text{ and } \sum_{i=1}^{k} \alpha_i y_i = 0 \tag{B.6b}
\]

From Eq. (B.4) and its constraint we know that solving the dual problem in Eq. (B.6) is equivalent to solving the primal problem in Eq. (B.3), but Eq. (B.6) is easier to solve than Eq. (B.3). In fact, this is a typical quadratic programming with linear constraints, and it can be solved by many readily available optimization packages.

Also, according to the Karush-Kuhn-Tucker (KKT) conditions, the following relationship between Lagrange multipliers \( \alpha \) and the constraint in Eq. (B.3) exists:

\[
\alpha_i [y_i(<w, x_i> + b) - 1] = 0 \tag{B.7}
\]

where \( i=1 \ldots k \).
After acquiring the optimal set of the $\alpha$, $w$ can be obtained from Eq. (B.5a), and $b$ can be obtained from Eq. (B.7). Therefore, the resultant decision function, $f(x)$, is:

$$f(x) = \text{sgn}(\sum_{i=1}^{k} \alpha_i y_i < x_i, x > + b)$$

(B.8)

Generally speaking, most of the $\alpha$'s in the optimal solution set are zero, and only a small fraction are non-zero. From Eq. (B.7) we know that the training examples corresponding to the non-zero $\alpha$'s are those which fall on the two margin hyperplanes, $< w, x > + b = -1$ and $< w, x > + b = 1$. Therefore, these training examples are called the **Support Vectors (SVs)**. In fact, only these SVs determine the resulting classification boundary. That is, removal the non-SVs will not change the optimal SVM classifier. This can also be derived from Eq. (B.5a) and (B.7), where only the non-zeros $\alpha$'s have an effect on the value of $w$ and $b$.

### B.3.2 Linear SVMs for linearly Inseparable Data

The linear SVM formulation is only suitable for linearly separable datasets, and will not yield a solution when applied to a linearly inseparable problem. Therefore, it is necessary to modify the formulation to handle the inseparable data.

In the modified formulation, a set of non-negative slack variables $\xi = [\xi_1, ..., \xi_k]^T$ are introduced to allow some training examples falling in the margin region or even on the wrong side of the margin region. One generally adopted linear SVMs formulation for inseparable data can be presented as follows:

$$\text{minimize} \quad \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{k} \xi_i$$

(B.9a)
subject to \( y_i (< w, x_i > + b) \geq 1 - \xi_i \) and \( \xi_i \geq 0. \) \hspace{1cm} (B.9b)

In fact, this formulation explicitly demonstrates the spirit of SVMs — the tradeoff between the "capacity" of the classifier, which is related to \(|w||\), and the accuracy obtained on the given training examples, which is presented in the term \( \sum_{i=1}^{k} \xi_i \). Here, \( C \) is a priori weighting parameter to control the relative importance between those two factors.

By following a similar derivation procedure, the dual of problem (B.9) is obtained as:

\[
\text{maximize} \quad D(\alpha) = \sum_{i=1}^{k} \alpha_i - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{k} \alpha_i \alpha_j y_i y_j < x_i, x_j > 
\]

subject to \( C \geq \alpha \geq 0, \) and \( \sum_{i=1}^{k} \alpha_i y_i = 0. \) \hspace{1cm} (B.10b)

Note that the only difference between (B.6) and (B.10) is that \( \alpha \) is upper-bounded by \( C \) in (B.10). Also, from the relationships obtained from the gradient of the Lagrangian formulation and the KKT conditions we know that the slack variable \( \xi_i \) is non-zero only when it corresponds to \( \alpha_i = C \), which, in turn, implies that the corresponding training examples, \( (x_i, y_i) \), either fall into the margin region or fall on the wrong side of the margin region. These kinds of training examples are called margin errors. Note that margin errors are not necessarily the same as the classification errors, because the margin errors can be those examples, which fall into the margin area but on the right side of the classification boundary.

The remaining portion of deriving the inseparable case follows the description for the separable case, and we will not duplicate the arguments here.
B.3.3 Nonlinear SVMs

The intuitive motivation behind nonlinear SVMs arise from the observation that a non-linearly separable training set may become linearly separable, or easier to separable, if it is mapped to a feature space of relative higher dimension. This idea is illustrated in Figure B.3.

Figure B.3 Illustrate the basic idea that motivates the introduction of nonlinear SVMs

Thus, the concept behind the implementation of nonlinear SVMs can be expressed as the following three steps.

1. Define a mapping function to map a input pattern \( x_i \) from the input space, \( I \), into a feature pattern \( \Phi(x_i) \), in the feature space, \( F \), which is of higher dimension than \( I \).

2. Use the linearly inseparable SVM algorithm on the feature space, \( F \), to obtain a decision hyperplane, \( L(x) \).
(3) Map the resultant decision hyperplane from the feature space, $F$, back to the input space, $I$, and thus obtain a non-linear decision function, $f(x)$, in the input space.

This procedure is illustrated in Figure B.4.

![Figure B.4 Conceptual steps implementing a nonlinear SVM](image)

Figure B.4 Conceptual steps implementing a nonlinear SVM

However, this ideal implementation is problematic. First, finding a suitable and invertible mapping function, $\Phi(x)$, is not a straightforward problem. Second, when the dimensionality of feature space, $F$, becomes extremely high, solving the quadratic optimization problem over this space becomes computationally prohibitive. The solution to these problems is a technique referred to as the "kernel trick".

If examining the linear formulation (both separable and non-separable cases) of SVMs, one can find that all of the input pattern vectors appear only in the dot product form in both formulations (B.6) and (B.10), as well as in the decision function, (B.8). This allows us to define a "kernel function", $K(x_i,x_j)$, such that:
\( K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle \) \hfill (B.11)

Consequently, formulation (B.10) becomes

\[
\text{maximize } D(\alpha) = \sum_{i=1}^{k} \alpha_i - \frac{1}{2} \sum_{i=1}^{k} \alpha_i \alpha_j y_i y_j K(x_i, x_j)
\]

subject to \( C \geq \alpha \geq 0 \), and \( \sum_{i=1}^{k} \alpha_i y_i = 0 \). \hfill (B.12a)

and the decision function, (B.8), becomes

\[ f(x) = \text{sgn} \left( \sum_{j=1}^{k} \alpha_j y_j K(x_j, x) + b \right) \]

where \( b \) can be determined from:

\[ \alpha_i [y_i (\sum_{j=1}^{k} \alpha_j y_j K(x_j, x_i) + b) - 1] = 0 \]

From (B.13) we know that the classification boundary will be naturally nonlinear. Also, note that it is not necessary to explicitly compute \( w \) for a nonlinear SVMs. Therefore, this "kernel trick" allows us to implement the three steps of nonlinear SVMs in one step, and without explicitly defining the map function, \( \Phi(x) \). Also, the computational burden is dramatically reduced by replacing the high dimensional dot product with the low dimensional function evaluation.

The remaining question is what kind of kernel function, \( K(x_i, x_j) \), satisfies Eq. (B.11). Fortunately, the Mercer Theorem provides the admissible conditions for a kernel function, \( K(x_i, x_j) \), which is [94]:

If \( K(x, y) \) is a continuous symmetric kernel of a positive integral operator \( T \), i.e.
\[ T f(y) = \int_{\mathcal{E}} K(x, y)f(x)dx \quad (B.15a) \]

with

\[ \int_{\mathcal{E}\times\mathcal{E}} K(x, y)f(x)f(y)dxdy \geq 0 \quad \text{for all} \quad \int f(x)^2 dx < \infty, \quad \text{then} \quad (B.15b) \]

\( K(x, y) \) can be a dot product in some feature space.

The two most widely used kernel functions are

(1) Polynomial kernel:
\[ K(x, y) = (x \cdot y + 1)^d \quad (B.16) \]

where \( d \) is an integer big than 1.

(2) Gaussian Radial Based Function (RBF) kernel:
\[ K(x, y) = \exp\left\{-\frac{\|x - y\|^2}{\sigma^2}\right\} \quad (B.17) \]

where \( \sigma \) is a positive real value.

\section*{B.3.4 Modified Formulations of SVMs}

Besides the three basic formulations of SVMs mentioned previously, there are several modified versions developed recently to emphasize different application scenarios. These new formulations include:

- **\( \nu \)-SVMs\ [97]:** This form of SVMs is capable of automatic adjusting the margin width according to a parameter, \( \nu \). This parameter is the lower bound on the fraction of SVs, and the upper bound on the fraction of margin errors, which refer to those training examples either in the margin area or on the wrong side of the margin area. Moreover, the fractions of both SVs and margin errors converge
to $v$ with probability 1 as the training set size goes to infinity. Therefore, by adjusting the value of $v$, we can approximately control the width of the margin region, as well as the fraction of SVs and margin errors.

- **Semi-parametric SVMs** [142]: The basic formulations of SVMs are non-parametric. However, some prior knowledge regarding the underlying problem is known in many cases. Therefore, semi-parametric SVMs were proposed to incorporate this prior knowledge. A common approach used in semi-parametric SVMs is "to fit the data with the parametric model and train the non-parametric add-on on the errors of the parametric part" [96].

- **Single-class SVMs** [106]: All the SVM formulations presented previously are based on two-class problems. The single-class SVMs, however, only deal with the examples from one class. It is capable of obtaining a boundary enclosing the support area of the distribution of the examples from that class.

## B.4 Attractive Properties of SVMs

According the formulation of SVMs presented previously, a number of attractive properties are embedded in this group techniques to ensure their easy utility and exceptional performance. Here, we mainly list 5 most important properties.

1. **Obtaining classification boundary directly from the training set**

As we mentioned previously, most of the conventional classification methods are based on extensive knowledge of the underlying distribution of data. In contrast, SVMs evade this
trap. They allow us to obtain the explicit form of a classifier's decision function directly from the training set.

(2) Extracting and storing classification information exclusively in the Support Vectors (SVs)

As we all know, not all training samples are equally important in determining the classification boundary. SVMs are capable of locating all the Support Vectors (SVs), a small fraction of the training set, which exclusively determine the form of the decision function. Therefore, the classification information of the whole training set is extracted, and is solely stored in the SVs. That is, even you discard all the non-SVs, and still get the same decision function. [92, 94, 95]

(3) Possessing clear geometric meaning

Although SVMs are based on the Structured Risk Minimization Theory, which is part of Statistical Learning Theory, they also can be geometrically interpreted as a set of margin-maximum classifiers. This geometric meaning provides us with many intuitions, which is hard to obtain when working with such conventional classification methods as Bayesian Classifiers and Neural Network Classifiers. [94, 95, 96]

(4) Obtaining globally optimal classification boundary of the training set

One of the headaches of training Neural Network Classifiers is that it cannot be guaranteed to result in a classification boundary that is globally optimal, because most of their underlying optimization algorithms cannot avoid being trapped in local optimums. In contrast, because the underlying optimization problem of SVMs is a convex quadratic
programming, the resultant classification boundary is guaranteed to be globally optimal, and in many cases, unique for a given training set. [95, 105]

(5) Avoiding the curse of dimensionality

Most conventional classifiers degrade dramatically in performance, or become computationally prohibitive with increase in sample dimension. This phenomenon is called the curse of dimensionality. In contrast, SVMs can easily circumvent the curse of dimensionality, because 1) no distribution estimation is needed in SVMs, and 2) only dot product (for linear SVMs), or function evaluation (for non-linear SVMs), is applied to the samples. As a matter of fact, the computation complexity of SVMs is only decided by the training set size. [94, 95]
BIBLIOGRAPHY


[89] TRUMPETS (Target Recognition using Mathematical and Physical Exploitation of Target Signatures) program proposal, MRC, 1997.


180


