DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/DS/ENG/94J-02

SPATIO-TEMPORAL PATTERN RECOGNITION USING HIDDEN MARKOV MODELS

DISSERTATION
Kenneth Henry Fielding
Captain, USAF

AFIT/DS/ENG/94J-02

AFIT/DS/ENG/94J-02

SPATIO-TEMPORAL PATTERN RECOGNITION USING HIDDEN MARKOV MODELS

DISSERTATION

Presented to the Faculty of the Graduate School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

Kenneth Henry Fielding, B.S.,B.S.E.E.,M.S.E.E.

Captain, USAF

June, 1994

AFIT/DS/ENG/94J-02

SPATIO-TEMPORAL PATTERN RECOGNITION USING HIDDEN MARKOV MODELS

Kenneth Henry Fielding, B.S.,B.S.E.E.,M.S.E.E.

Captain, USAF

Approved:

Dennis W. Ruck	5 MAY 94
	5-5-94
Byron M Welsh	5-5-94
Mark E. Oxley	5-5-94
Michael C. Roggemann	5-13-94

J S Przemieniecki 13 May 1994

J. S. PRZEMIENIECKI
Institute Senior Dean and
Scientific Advisor

Acknowledgements

The last three years at AFIT have been a great experience. I have learned much here and am deeply indebted to many important people who have made the success of this research possible. To those who have been my Professors, I am reminded of this

> A teacher affects eternity; he can never tell where his influence stops.
> *Henry Brook Adams*, 1907

To my advisor, Dr. Dennis Ruck, I thank you for your insight, patience, and sound advice that always kept me heading in the right direction. To my committee members, Dr. Steve Rogers, Dr. Byron Welsh, and Dr. Mark Oxley, I thank you for your advice and guidance. To Tom Burns, Doug Dyer, and Steve Troxel for your valued friendship. To Dan Zambon and Dave Doak, the two most excellent computer administrators, who kept my computer going in spite of me. To my sponsor, Greg Power, for your help and encouragement. I would also like to thank Dr. Michael Seibert and Dr. Alan Waxman from MIT Lincoln Laboratory for their advice and kindness in providing their computer code for creating the sphere plots.

Most importantly, I thank my wife, Susan, and my children, Katie and Kristy, with whom the Lord has blessed me. The success of this dissertation is due to your love, sacrifice, and unending support over the last three years.

Kenneth Henry Fielding

Table of Contents

List of Figures

List of Tables

List of Symbols

Abstract

A new spatio-temporal method for identifying 3D objects found in 2D image sequences is presented. The Hidden Markov Model technique is used as a spatio-temporal classification algorithm to identify 3D objects by the temporal changes in observed shape features. A new information theoretic argument is developed that proves identifying objects based on image sequences can lead to higher classification accuracies than single look methods. A new distance measure is proposed that analyzes the performance of Hidden Markov Models in a multi-class pattern recognition problem. The new distance measure is shown to be superior than those previously reported. A three class problem identifying moving light display objects provides experimental verification of the sequence processing argument. Individual frames of a MLD image sequence contain very little spatial information. The information content is highly temporal in that sense that image sequences are required for object identification. The single look classification rate for the moving light display imagery was observed to be near 50%. In contrast, the Hidden Markov Model classification rate was above 93%. The alternate nearest neighbor multiple frame technique classification rate was 20% below the Hidden Markov Models. A one sided t-test revealed a highly statistically significant difference between the Hidden Markov Model and multiple frame technique at a 0.01 level of significance. A five class problem consisting of tactical military ground vehicles is considered to provide verification using imagery with both spatial and temporal information. The classification accuracy of the Hidden Markov Model is compared to a single look and an alternate multiple frame technique. Results confirmed the new spatio-temporal pattern recognition method produces superior results by accessing the temporal information in the image sequences. A prototype automatic target recognition system is demonstrated. Objects in real video imagery are correctly identified by the spatio-temporal Hidden Markov Model classifiers trained on synthetic data.

SPATIO-TEMPORAL PATTERN RECOGNITION USING HIDDEN MARKOV MODELS

I. Introduction

1.1 Historical Background

The Air Force has been investigating automatic pattern recognition techniques for several decades. Exploratory and advanced development is occurring on methods that will lead to systems for speech recognizers for task load reduction, text readers, autonomous navigation systems, photographic interpretation systems, and automatic target recognition and tracking systems.

Automatic target recognition systems whose purpose is to identify three dimensional (3D) objects from two dimensional (2D) imagery derived from visual wavelength or infrared sensors, have access to a time indexed stream of such images. These systems, however, generally perform a given technique on a single frame of such imagery at a time. A determination of the desired information, such as object classification, is made and a new frame of imagery is then analyzed and the process repeated. The use of a single frame of imagery to determine the desired information will be referred to as a *single look* method in this document.

Recently, several researchers have been investigating techniques using several independent single look observations to improve the classification accuracy of 3D objects in 2D imagery. Wang et. al. (68) present a method for identifying 3D objects with a model based technique. 3D models of test objects are generated from a sequence of two to four 2D silhouettes taken randomly from different orientations around the object. The estimated model is then compared to a library for identification. Leung and Huang (41) describe a method for three-dimensional motion estimation and object identification using a pair of stereo images from two adjacent time indices. Motion is derived from the time pair using an optical flow

technique with recognition independently performed on the left image in each stereo pair. Liu and Tsai (46) report a method of 3D object recognition using 2D object silhouettes obtained from two distinct camera viewing angles. First, a top-view camera captures the object image. If the top-view shape is inadequate for discrimination, a lateral camera is activated and additional information is obtained. These investigations support the concept that multiple single look observations tend to improve the classification rate, however, they do not access and utilize the vast amount of information in the temporal sequence of imagery produced by the sensor.

It has become well known that the temporal changes an object undergoes when moving relative to an observer contains information that may aid in the interpretation of the motion and description of the object (2, 49). Animals as well as Man use this additional temporal visual information to aid in maneuvering in the environment, locating food, and detecting predators. Recent neurophysiological studies tend to support the position that temporal information is used by animals in their visual recognition systems. Barlow and Levick [illegible]) and Sakai and Naka (58) have identified cells in the retina of rabbits and catfish that sense and process information related to motion. Watson and Ahumada (69) refer to many neurophysiological papers on the properties of human motion perception that serve as a basis for their view of how humans sense the velocity of moving objects. Of particular interest to this study is the work of Perrett et al. (51) who discovered cells in the superior temporal sulcus of macaque monkeys that maximally respond to selective characteristic views of two and three dimensional faces. Perrett et al. (50) additionally found cells that were not responsive maximally to particular characteristic views, but responsive to a transition between two characteristic views. Perrett's work has shown the macaque monkey, and perhaps all mammals, gather and use information related to changes in object orientation over time as well as certain static views for recognition.

Several investigators have recognized the potential of using information in data sequences for object identification. One of the early pioneers of this concept is Le Chevalier et al. (40), who in the late 1970's recognized that single look identification of aircraft using radar produces unsatisfactory results because of feature ambiguity among different aircraft types.

They treat the moving target as a syntactic process or grammar that generates characteristic sequences of observation measurements. Sequence transitions are bound by evolutionary constraints that are particular to a given target. Libby (43) also investigated methods for dynamic object recognition using radar. Recognition is based on the joint likelihood of kinematic and feature observable events over time. His approaches involved parameter estimation using multiple-model Kalman filters and dynamic programming-based sequence comparison methods. Dewitt (19) used range profiles obtained from high resolution radar and the Hidden Markov Model technique to identify aircraft. Hidden Markov Models were trained to recognize the objects for specific look angles using range profiles consisting of 10 scattering centers.

The work of Seibert and Waxman (59) can be grouped with the Le Chevalier concept and is the only investigation known to the author where sequences of 2D views of 3D objects were used in the classification process. Seibert and Waxman (59) employ a differential equation based method as a temporal hypothesis test that reacts to previously learned transitions in object features.

1.2 Problem Statement and Scope

This research describes a new approach for recognizing moving 3D objects using a sequence of 2D images. The identification of 3D objects using the information contained in 2D image sequences is a largely unexplored and fertile area of research. This research studies the use of a spatio-temporal learning and classification algorithm, known as Hidden Markov Models, for solving this problem. The aspects of the solution to this problem are 1) develop an information theoretic argument for sequence processing, 2) develop a new distance measure to analyze the performance of the Hidden Markov Model classifiers, 3) analyze the performance of the Hidden Markov Model technique with moving light display imagery that has a low spatial/high temporal information content, 4) experimentally demonstrate the effectiveness of the spatio-temporal sequence processing on a five class military vehicle classification problem that has high spatial/high temporal information and, 5) demonstrate of recognition of real

world image sequences of two military vehicles using models trained on synthetic data. The research contributions made in these areas are reviewed below.

- **An Information Theoretic Argument For Sequence Processing.** A new argument advocating the use of sequence, rather than single look, processing will be developed. The argument is based on Shannon's definition of information and its relationship with entropy (61).
- **Hidden Markov Model Distance Measure.** A new method for analyzing the distance between a pair of Hidden Markov Models is proposed. The distance measure between pairs of Hidden Markov Models gives insight into the sensitivity of the model to changes in parameters. Additionally, the distance measure is an important tool for analyzing the performance of Hidden Markov Models in a multi-class pattern recognition problem. The proposed method uses higher order statistics, the mean and variance of the Hidden Markov Model output distributions, and the Bhattacharyya distance measure to find the distance between each Hidden Markov Model pair. A *worst case* example demonstrates that the new method is a superior approach yielding a more informative distance measurement between pairs of Hidden Markov Models.
- **Identification of Moving Light Displays.** This dissertation reports the first known pattern recognition algorithm applied to the identification of objects from a class of imagery known as moving light displays. All previously known automated techniques attempt to uncover the type of motion the moving light display object is undergoing. Individual frames of a MLD image sequence contain very little spatial information. The information content is highly temporal in that sense that image sequences are required by humans for object identification. A three class moving light display classification problem demonstrates the power and robustness of the spatio-temporal technique proposed here.
- **Use of Hidden Markov Models as a Spatio-temporal Classifier.** A novel algorithm employing the Hidden Markov Model technique is described to experimentally verify

the information theoretic argument for sequence processing. The Hidden Markov Model is used as a spatio-temporal pattern recognition algorithm that identifies 3D objects contained in 2D image sequences. Experimentation using a five class problem demonstrates the theoretical advantages of recognizing objects using image sequences. The Hidden Markov Model performance will be shown to be substantially superior to a single look and alternate multiple frame classification technique.

- **Identification of Real Imagery.** Identifying objects in real sensor imagery using classifiers trained on synthetic data is one of the most highly desired characteristics of a pattern recognition system. This characteristic, however, is seldom seen. This dissertation demonstrates such a system where real video image sequences of the M60 tank and M35 truck are successfully classified.

1.3 Dissertation Organization

This dissertation is organized into seven chapters. The following chapter reviews the concepts this research is based on. An argument based in information theory is described that substantiates the hypothesis that sequence processing can provide enhanced classification over single-look methods. The concept of the Hidden Markov Model and its application to this problem are reviewed along with training and testing methodologies. Chapter III introduces a new application of the Bhattacharyya distance to measure the distance between a pair of Hidden Markov Models. This distance measure is an important tool in analyzing the performance of Hidden Markov Models in a multi-class pattern recognition application. Chapter VI reports an investigation into this technique's ability to classify moving light display objects which have low spatial and high temporal information content. Chapter V is a demonstration and analysis of the Hidden Markov Model technique on a five class problem consisting of tactical military ground vehicles. Chapter VI describes a real world application of the technique where Hidden Markov Models trained on synthetic data sequences correctly classify targets contained in real video image sequences. The final chapter of this dissertation will discuss recommendations and conclusions derived from this work.

II. Background Material

2.1 Introduction

This research focuses on employing a spatio-temporal technique to classify 3D objects contained in 2D image sequences. This chapter will review the concepts and background material necessary to understand the approaches and results of this dissertation. First, information theory based arguments are presented that discusses why processing data sequences will result in equal or better classification rates than single look methods. Second, the spatio-temporal sequence processing technique used in this dissertation, the Hidden Markov Model, will be described. The next section will discuss the single look methods used as a baseline for Hidden Markov Model performance comparison. An alternate sequence processing method is also investigated. The last section will discuss the classifier error testing method used throughout this research.

2.2 An Advantage for Sequence Classifiers

Many single look pattern recognition techniques used to identify 3D objects in 2D imagery fit into the general category of statistical classifiers. This section will describe the concept of the statistical classifier and develop information theory based argument describing why sequence classification should outperform single look classification.

2.2.1 The Statistical Classifier. Statistical classifiers are based on mathematical classification rules formulated in a statistical framework (65). Classification is generally made by following Bayes decision rule (26)

$$\text{Decide } \omega_i \text{ if } p(\omega_i|\mathbf{x}) > p(\omega_j|\mathbf{x}) \text{ for all } j \neq i \tag{1}$$

where $\mathbf{x}$ is a set of object measurements, ω_i is the i-th object class, and $p(\omega_i|\mathbf{x})$ is the *a posteriori* probability. The Bayes, or statistical, classifier is optimum in the sense that it minimizes the probability of classification error if the true *a posteriori* probability density

functions are known (21). This statistically optimal classification rule is the accepted standard against which the performance of other classification algorithms are often compared (65). Generally, the measurements used to make a classification decision using Equation 1 result from the examination a single frame of imagery at a time. The set of object measurements can be arranged in a vector form that defines a *feature space*. A feature vector obtained from an object in a single image is associated with a point in the feature space. If the set of measurements are chosen well for a particular problem, feature vectors from objects of different classes will lie in disjoint partitions of the feature space. Usually, however, there is a degree of *ambiguity* in the measurement process causing an overlap in the partitions. This overlap induces error in the classification process.

By examining a single image frame at a time, the single look statistical classifier considers the position and orientation of the object in each frame of imagery to be independent of past or future positions and orientations. Experience tells us that objects moving in the world around us do not change perspective independently, moment to moment, but follow a characteristic behavior. The biological studies described in Chapter I emphasize the fact that the characteristic behavior of moving objects is important. It is reasonable, therefore, to consider the temporal changes a feature vector undergoes due to object motion in the classification process. Bayes decision rule for such a classifier may be written as

$$\text{Decide } \omega_i \text{ if } p(\omega_i|\mathbf{X}) > p(\omega_j|\mathbf{X}) \text{ for all } j \neq i \tag{2}$$

where $\mathbf{X}$ is a sequence of n time indexed feature vectors $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ and ω_i is the i-th object class. Equation 2 indicates that the *a posteriori* probabilities now depend on a time indexed history of measurements. Although an estimation of the true *a posteriori* densities is not undertaken in this dissertation, a method that does exploit the joint nature of the temporal changes in object features is investigated.

2.2.2 Le Chevalier's Reasoning for Sequence Processing. Le Chevalier et. al. (40) were among the first to recognize that examining the relationship of the changing

features from a moving object can improve classification. They found that single look identification of aircraft using radar cross section measurements produces unsatisfactory results because of feature ambiguity among different aircraft types. To improve the recognition capability, they treat the moving object as a syntactic process, or grammar, that generates characteristic sequences of observation measurements. The observed sequence measurements are bound by evolutionary constraints that are particular to a given object facilitating discrimination. This connection was made by Le Chevalier's knowledge of the finite state automaton and its one-to-one relationship with a grammar, a particular implementation of syntactic pattern recognition. The important point of Le Chevalier's work is that the search for, and recognition of, specific orderings of object measurements reduces the classification error rate found in the single look approach. The classification method used in this dissertation is itself a syntactic pattern recognition approach. The Hidden Markov Model is a special case of a regular stochastic grammar (14).

2.2.2.1 Syntactic Pattern Recognition. The statistical classifier is based on a mathematical approach to pattern recognition (65). The approach known as *syntactic pattern recognition* is rooted in the concept of formal language theory. The basic difference between syntactic and statistical pattern recognition is that syntactic pattern recognition explicitly uses the structure, or order, of patterns in the recognition process (65). Classification decision rules are essentially implementations of Equation 2. Grammars are the basic construct in syntactic pattern recognition and are the foundation of the majority of research in this area. The essential concepts associated with syntactic pattern recognition described in (65:317) are:

1. An *alphabet* is a finite set of symbols.
2. a *sentence* over an alphabet is any string of finite length composed of symbols in the alphabet.
3. A *language* is a set (not necessarily finite) of sentences of an alphabet
4. Each language has a unique *grammar*, which describes the structure of the language and is defined by the four-tuple $G = (V_N, V_T, P, S)$, where

(a) V_N is a set of *nonterminals* (variables).

(b) V_T is a set of *terminals* (constants).

(c) P is a set of *production* rules.

(d) S is the *root* symbol (corresponding to the sentence).

The language is then a set of strings which satisfy: (1) each string is composed only of terminals, and (2) each string can be derived from S by suitable applications of productions from the set P. There are four types of grammars differing only in the type of productions allowed. The four types are:

1. *unrestricted* - a symbol may be followed by either a nonterminal or terminal.
2. *context-sensitive* - production rules are of the form $\alpha_1 A \alpha_2 \rightarrow \alpha_1 \beta \alpha_2$. α_1, α_2, and β are terminals or nonterminals and A is a nonterminal.
3. *context-free* - productions are of the form $A \rightarrow \beta$.
4. *regular* - productions are of the form $A \rightarrow aB$ or $A \rightarrow a$ where a is a terminal and A and B are nonterminals.

It is interesting to note that all regular grammars are context free, all context-free grammars are context sensitive, and all context-sensitive grammars are unrestricted (65).

The framework for a syntactic pattern recognition system is to develop a grammar for each object class under consideration. The classification process matches a test string, or sentence, with each known language through a process known as *parsing*. Parsing can be accomplished in a top-down fashion which begins with the root and through repeated applications of the grammar productions, arrives at the sentence. A bottom-up method is also used that begins with the sentence to which the production rules are applied in reverse to recover the root. Both methods reveal the underlying grammatical structure of the test sequence which is associated with the language (class) that yields a parse consistent with its production rules. Classification difficulties can arise if a correct parse of a test sequence

cannot be made by any of the languages under consideration. This can occur when noise or other ambiguities affect the test sequence.

A statistical description of the language can be introduced by allowing the grammar production rules to be nondeterministic and assigning a certain probability measure to each of these productions. This type of grammar is called a *stochastic grammar*. The stochastic grammar approach allows for the construction of a grammar and language in problems where an explicit expression of the production rules cannot be formulated. An additional benefit is that the parsing process of a stochastic grammar yields a probability of association with each language, reducing the aforementioned problems with classification.

Key to the use of stochastic grammars in pattern recognition are efficient algorithms for learning production rule probabilities and for the recognition of test sentences. One such algorithm for the learning of regular stochastic grammars is the discrete Hidden Markov Model (14). The Hidden Markov Model will be the basic tool used in this research to learn the grammar of features associated with 3D objects moving in 2D image sequences. With the *languages* for each object in the specific problem defined, the classification of new data sequences can be made.

2.2.3 Libby's Reasoning for Sequence Processing. Libby also classified aircraft using sequences of radar based features (43). Libby used Dynamic Time Warping to recognize time indexed sequences of kinematic and non-kinematic features derived from an aircraft. Libby recognized that his approach is essentially an implementation of syntactic pattern recognition which not only examines the data sequence for the presence of certain features, but takes into account the time ordering. Libby attributes the enhanced performance of sequence processing over single look methods to the restriction of the *matching domain* of the class *a posteriori* probability densities. By restricting the matching domain, the classifier essentially ignores sequences of features that are not consistent with the learned model. The restriction decreases the *a posteriori* probabilities for incorrect feature sequences while not affecting correct sequences.

2.2.4 An Information Theoretic Argument. Statistical communication theory was founded by Shannon in 1948 (61, 62). The theory was formulated to express in quantitative terms the transmittal of information through a communication channel. To investigate this process and its relationship to sequence classification, the basic unit of information needs to be defined. The definition given by Abramson (1) is

> *Definition.* Let E be some event which occurs with probability $P(E)$. If we are told that event E has occurred, then the information received, $I(E)$, is
>
> $$I(E) = \log \frac{1}{P(E)}$$
>
> units of information.

Assume that an information source can produce symbols $a_1, a_2, \ldots, a_n$ from an alphabet S and that each symbol is generated with probability $p(a_1), p(a_2), \ldots, p(a_n)$. The average information over the entire alphabet, known as *entropy*, is denoted by $H(S)$ and is defined as (32)

$$H(S) = \sum_{i=1}^{n} p(a_i) \log \frac{1}{p(a_i)}$$

Entropy can also be thought of as a measure of uncertainty in the information. If each symbol in a alphabet is equiprobable, entropy is maximized. If only one symbol from an alphabet has a probability of occurrence (of one in this case), the entropy is zero. There seems to be a paradox that implies the more random the event, the more entropy, or information, is contained in the event. Cole (15) points out that there is really no paradox at all but merely a misinterpretation of the concept of entropy. When Shannon discusses entropy from the point of view of the sender of a message

information=uncertainty=entropy;

when he is discussing information from the perspective of the receiver of the message

information=reduction of uncertainty=reduction of entropy.

For the purpose of object classification, the recognition system is considered to be on the receiving end of the communication channel linking the object and the input sensor. The set of

possible measurements taken from the sensor defines the alphabet of the information source, or object. Since the recognition system is on the receiving end of the channel, an increase in information in the object features is directly related to a decrease in the corresponding entropy, thus, reducing the uncertainty in the class of the measurement. The method investigated here to reduce the entropy of the information from an object is to consider the entropy, or average information, of a sequence of observations.

Assume a discrete random process $\{\ldots, X_{-2}, X_{-1}, X_0, X_1, X_2, \ldots\}$ where X_i are identically distributed random variables taking on values in the source alphabet S of m symbols $a_1, a_2, \ldots, a_m$ with each symbol being generated with probability $p(a_1)$, $p(a_2)$, ... $p(a_m)$. The probability of every symbol is strictly positive or it is deleted from the alphabet.

The entropy of a block of random variables, denoted $H(X_1, X_2, \ldots, X_n)$, is defined by Blahut (11:59) as

$$H(X_1, X_2, \ldots, X_n) = -\sum_{S^n} p(X_1, X_2, \ldots, X_n) \log p(X_1, X_2, \ldots, X_n)$$

where $p(X_1, X_2, \ldots, X_n)$ is the joint probability density function of the n random variables. The summation over S accounts for all possible orderings of the sequence. Blahut (11:59) shows the relationship between the individual symbol and joint entropy can be expressed as

$$H(X_1, X_2, \ldots, X_n) \leq \sum_{i=1}^{n} H(X_i) \tag{3}$$

with equality holding if the random variables are independent.

Hamming (32:135) and Abramson (1:26) show that a similar relationship exists if an $(n-1)$-th order Markov process is assumed. The condition is expressed as

$$H_{n-1}(X_1, X_2, \ldots, X_n) \leq \sum_{i=1}^{n} H(X_i) \tag{4}$$

with equality holding for independence. Sometimes $H_{n-1}(X_1, X_2, \ldots, X_n)$ is written as $H(X_n|X_1, X_2, \ldots, X_{n-1})$ where

$$H_{n-1}(X_1, X_2, \ldots, X_n) = -\sum_{S^n} p(X_1, X_2, \ldots, X_n) \log p(X_n|X_1, X_2, \ldots, X_{n-1})$$

Equations 3 and 4 demonstrate that constraints, or dependency, on alphabet symbols reduce the entropy. In this particular case, the constraint is to examine a sequence of n symbols as described in Equation 3 or a sequence of n symbols with an $(n-1)$-th order Markov dependency as in Equation 4. Following this thought, it should be the case that tighter constraints on a sequence of n symbols will reduce the entropy. To this end, the following new theorem is proved.

Theorem 1. Assume a discrete random process $\{\ldots, X_{-2}, X_{-1}, X_0, X_1, X_2, \ldots\}$ where X_i are identically distributed random variables taking on values in the source alphabet S of m symbols $a_1, a_2, \ldots, a_m$ with each symbol being generated with probability $p(a_1), p(a_2), \ldots, p(a_m) \ni p(a_i) > 0 \; \forall \; i : 1 \leq i \leq m$. Then $H(X_1, X_2, \ldots, X_n) \geq H_{n-1}(X_1, X_2, \ldots, X_n)$.

Proof. Examine $H_{n-1}(\mathbf{X}) - H(\mathbf{X})$ where $\mathbf{X} = (X_1, X_2, \ldots, X_n)$.

$$H_{n-1}(\mathbf{X}) - H(\mathbf{X}) = \sum_{S^n} p(X_1, X_2, \ldots, X_n) \left[\log \frac{1}{p(X_n|X_1, X_2, \ldots, X_{n-1})} - \log \frac{1}{p(X_1, X_2, \ldots, X_n)} \right]$$

which becomes

$$H_{n-1}(\mathbf{X}) - H(\mathbf{X}) = \sum_{S^n} p(X_1, X_2, \ldots, X_n) \left[\log \frac{p(X_1, X_2, \ldots, X_n)}{p(X_n|X_1, X_2, \ldots, X_{n-1})} \right]$$

Using the relation $p(X_1, X_2, \ldots, X_n) = p(X_n|X_1, X_2, \ldots, X_{n-1}) p(X_1, X_2, \ldots, X_{n-1})$ and knowing that each factor is a probability mass function with value $0 < p(\cdot) \leq 1$, it

is seen that

$$p(X_n|X_1, X_2, \ldots, X_{n-1}) \geq p(X_1, X_2, \ldots, X_n)$$

This implies

$$\log \frac{p(X_1, X_2, \ldots, X_n)}{p(X_n|X_1, X_2, \ldots, X_{n-1})} \leq 0$$

yielding the result

$$H(X_1, X_2, \ldots, X_n) \geq H_{n-1}(X_1, X_2, \ldots, X_n)$$

Q.E.D.

If the $(n-1)$-th order Markov constraint is relaxed, the corresponding entropy should increase. This is demonstrated in the next theorem.

Theorem 2. Using the same presumptions given in Theorem 1, then
$H(X_1, X_2, \ldots, X_n) \geq H_1(X_1, X_2, \ldots, X_n) \geq H_{n-1}(X_1, X_2, \ldots, X_n)$
where $H_1(X_1, X_2, \ldots, X_n)$ represents the entropy of a 1st order Markov process.

Proof. First, show that $H(X_1, X_2, \ldots, X_n) \geq H_1(X_1, X_2, \ldots, X_n)$

Begin with the relationship

$$\begin{aligned} p(X_1, X_2, \ldots, X_n) &= p(X_n|X_1, X_2, \ldots, X_{n-1})p(X_{n-1}|X_1, X_2, \ldots, X_{n-2}) \\ &\ldots \quad p(X_2|X_1)p(X_1) \end{aligned}$$

Since $0 < p(X_1) \leq 1$ this implies that

$$p(X_1, X_2, \ldots, X_n) \leq p(X_n|X_1, X_2, \ldots, X_{n-1})p(X_{n-1}|X_1, X_2, \ldots, X_{n-2}) \ldots p(X_2|X_1)$$

Assuming a first order Markov process, the terms on the right hand side are modified to become

$$p(X_1, X_2, \ldots, X_n) \leq p(X_n|X_{n-1})p(X_{n-1}|X_{n-2}) \ldots p(X_2|X_1)$$

Now examine the difference $H_1(X_1, X_2, \ldots, X_n) - H(X_1, X_2, \ldots, X_n)$. This is given as

$$\begin{aligned} H_1(\mathbf{X}) - H(\mathbf{X}) &= \sum_{S^n} p(\mathbf{X}) \left[\log \frac{1}{p(X_n|X_{n-1})p(X_{n-1}|X_{n-2}) \ldots p(X_2|X_1)} \right. \\ &- \left. \log \frac{1}{p(X_1, X_2, \ldots, X_n)} \right] \end{aligned}$$

which becomes

$$H_{n-1}(\mathbf{X}) - H(\mathbf{X}) = \sum_{S^n} p(\mathbf{X}) \left[\log \frac{p(X_1, X_2, \ldots, X_n)}{p(X_n|X_{n-1})p(X_{n-1}|X_{n-2}) \ldots p(X_2|X_1)} \right]$$

The log term is always less than or equal to zero which gives the result

$$H(X_1, X_2, \ldots, X_n) \geq H_1(X_1, X_2, \ldots, X_n) \tag{5}$$

Now show that $H_1(X_1, X_2, \ldots, X_n) \geq H_{n-1}(X_1, X_2, \ldots, X_n)$. Here, begin with the relationship

$$\begin{aligned} &p(X_n|X_1, X_2, \ldots, X_{n-1})p(X_1, X_2, \ldots, X_{n-1}) \\ &= p(X_n|X_1, X_2, \ldots, X_{n-1})p(X_{n-1}|X_1, X_2, \ldots, X_{n-2}) \ldots p(X_2|X_1)p(X_1) \end{aligned}$$

which can be rearranged as

$$\begin{aligned} p(X_n|X_1, X_2, \ldots, X_{n-1}) &= p(X_n|X_1, X_2, \ldots, X_{n-1})p(X_{n-1}|X_1, X_2, \ldots, X_{n-2}) \\ &\ldots \quad p(X_2|X_1)\frac{p(X_1)}{p(X_1, X_2, \ldots, X_{n-1})} \end{aligned}$$

The fraction on the right hand side is greater than or equal to one yielding

$$\begin{aligned} p(X_n|X_1, X_2, \ldots, X_{n-1}) &\geq p(X_n|X_1, X_2, \ldots, X_{n-1})p(X_{n-1}|X_1, X_2, \ldots, X_{n-2}) \\ &\ldots \quad p(X_2|X_1) \end{aligned}$$

Applying the 1st order Markov property to the right hand side gives

$$p(X_n|X_1, X_2, \ldots, X_{n-1}) \geq p(X_n|X_{n-1})p(X_{n-1}|X_{n-2}) \ldots p(X_2|X_1)$$

Now examining the difference $H_1(X_1, X_2, \ldots, X_n) - H_{n-1}(X_1, X_2, \ldots, X_n)$ yields

$$\begin{aligned} H_1(\mathbf{X}) - H_{n-1}(\mathbf{X}) &= \sum_{S^n} p(\mathbf{X}) \left[\log \frac{1}{p(X_n|X_{n-1})p(X_{n-1}|X_{n-2}) \ldots p(X_2|X_1)} \right. \\ &- \left. \log \frac{1}{p(X_n|X_1, X_2, \ldots, X_{n-1})} \right] \end{aligned}$$

which becomes

$$H_1(\mathbf{X}) - H_{n-1}(\mathbf{X}) = \sum_{S^n} p(\mathbf{X}) \left[\log \frac{p(X_n|X_1, X_2, \ldots, X_{n-1})}{p(X_n|X_{n-1})p(X_{n-1}|X_{n-2}) \ldots p(X_2|X_1)} \right]$$

Here, the log term is greater than or equal to zero yielding

$$H_1(X_1, X_2, \ldots, X_n) \geq H_{n-1}(X_1, X_2, \ldots, X_n) \tag{6}$$

Equations 5 and 6 are combined to give the desired result

$$H(X_1, X_2, \ldots, X_n) \geq H_1(X_1, X_2, \ldots, X_n) \geq H_{n-1}(X_1, X_2, \ldots, X_n)$$

Q.E.D.

The results of Theorem 1 and Theorem 2 can be combined with Equations 3 and 4 to show in general terms the affects of putting constraints the production of alphabet symbols. The final result is

$$\sum_{i=1}^{n} H(X_i) \geq H(X_1, X_2, \ldots, X_n) \geq H_1(X_1, X_2, \ldots, X_n) \geq H_{n-1}(X_1, X_2, \ldots, X_n)$$

It has been proven that if there is dependency in the sequence of occurrences from the alphabet, the entropy of the joint occurrence is less than the entropy of the individual events. A pattern recognition algorithm accessing the greater wealth of information in the image sequence will have equal or *greater* classification accuracy than single look methods. Furthermore, by putting additional constraints on the type of dependency of the sequence, in this case a 1st and $(n-1)$-th order Markov process, the entropy is reduced. Classification using an algorithm designed to account for this dependency will increase. This information theoretic argument is consistent with those attributed to Le Chevalier (40) and Libby (43) and form the basis for the consideration of the spatio-temporal sequence classification method used in this research.

2.3 The Hidden Markov Model - A Sequence Recognizer

Now that it has been established that sequence classification can have advantages over single look methods, an algorithm to implement sequence classification must be found. The sequence classification method used in this research is the Hidden Markov Model.

When attempting to characterize the property of real-world signals, modeling methods can be categorized into two main types: deterministic and statistical. Deterministic models generally exploit a known property of the signal and concentrate on estimating signal parameters. As an example, a deterministic model would estimate amplitude, frequency, and phase for a sine wave signal. Statistical models assume the signal can be characterized as a parametric random process and that the parameters can be estimated accurately. Gaussian, Markov, and the Hidden Markov Model, which is used in this research, are examples of statistical models.

Hidden Markov Model techniques have been used extensively in the area of speech recognition over the past 15 years and have become the technique of choice among many researchers because of their ability to successfully learn the time varying characteristics of the spoken word. Here, concentration is given to modeling signals, the time varying features of 3D objects in 2D image sequences, for the purpose of object identification. Excellent reviews of the Markov process and its extension to the Hidden Markov Model are given by

Rabiner (55), Rabiner and Juang (53), Levinson (42), and Poritz (52). A brief description of the Markov process and its extension to the Hidden Markov Model based on Rabiner (55) follows.

Consider a system which can be in one of N distinct states, $S_1, \ S_2, \ \ldots \ , S_N$ where a state corresponds to a physical event or observation. Figure 1 illustrates the case for $N = 5$.

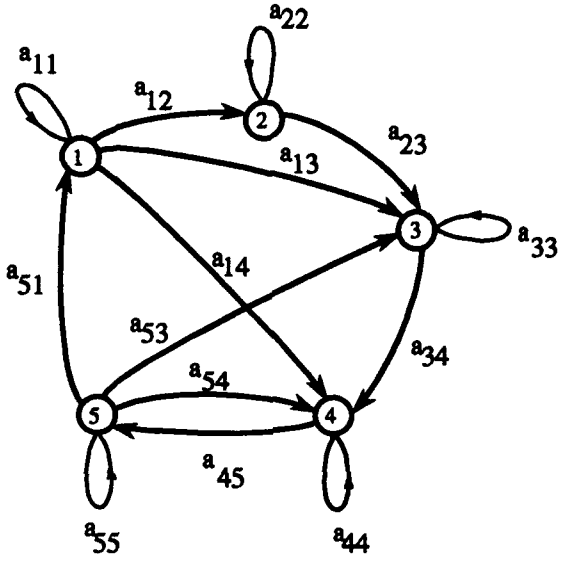


Figure 1. Architecture of a five state Markov process. Several possible links were omitted for clarity.

Assume at regularly spaced times, the system undergoes a change in state (possibly to the same state) and associate with the state changes the time variable $t \ = \ 1, \ 2, \ \ldots, \ T$. Associate the actual state at time t with the variable q_t. A full probabilistic description requires the specification of the current state and all past states. For the special case where the system behaves as a first order Markov process the probability can be written as

$$a_{ij}(t) = P\left(q_t = S_j | q_{t-1} = S_i, q_{t-2} = S_k, \ldots\right) = P\left(q_t = S_j | q_{t-1} = S_i\right) \qquad (7)$$

If the process is stationary, the transition probabilities are defined as

$$a_{ij} = P\left(q_t = S_j | q_{t-1} = S_i\right), \quad 1 \leq i, \ j \leq N \qquad (8)$$

The discrete Markov process is called *observable* if the output process is the set of states at each time instant that corresponds to an observation from a sequence $\mathbf{O} = \{o_1, o_2, \ldots, o_T\}$. If the observations are a probabilistic function of state, the process is called a Hidden Markov Model. Here, a sequence of observations is known but the actual state the process was in when the observation occurred is unknown. Thus, a direct connection between a state and a physical event cannot be made as with the observable Markov process. This characteristic enables the Hidden Markov Model to be applied to wide range of problems where the direct connection between states and physical events is not possible. A discrete first order Hidden Markov Model is typically defined by the number of states, N, the number of distinct observation symbols per state, M, and the 3-tuple $\lambda = (A, B, \Pi)$ where

1. $A = \{a_{ij}\}$ represents state transition probabilities $a_{ij} = P\left(q_t = S_j | q_{t-1} = S_i\right)$.
2. $B = \{b_{jk}\}$ is the output observation probability distribution in state j, given as $b_j(k) = P(v_k \text{ at t } | q_t = S_j)$ with $k \in M$ and v_k representing an individual observation symbol.
3. Π is the initial state distribution $\pi_i = P(q_1 = S_i)$

These parameters induce the probability measure $P(\mathbf{O}|\lambda)$ which indicates how closely an observation sequence is associated with a particular model λ.

Rabiner (55:261) describes three problems that must be overcome to use Hidden Markov Models. These are

- **Problem 1.** Given an observation sequence $\mathbf{O} = \{o_1, o_2, \ldots, o_T\}$ and a model $\lambda = (A, B, \Pi)$, how is $P(\mathbf{O}|\lambda)$ computed in an efficient manner?
- **Problem 2.** Given an observation sequence $\mathbf{O} = \{o_1, o_2, \ldots, o_T\}$, how is the corresponding state sequence $\mathbf{Q} = \{q_1, q_2, \ldots, q_T\}$ determined?
- **Problem 3.** How are the model parameters, $\lambda = (A, B, \Pi)$, adjusted to maximize $P(\mathbf{O}|\lambda)$ for a specific set of training data?

This research will not be concerned with the answer to Problem 2 since the estimated state sequence is not part of the classification method used here. The answers to Problems 1 and 3 are related and described below.

In answering Problem 1, Rabiner considers an observation sequence of the form, $\mathbf{O} = \{o_1, o_2, \ldots, o_T\}$, and a single fixed state sequence $\mathbf{Q} = \{q_1, q_2, \ldots, q_T\}$ (55:262). Assuming statistical independence of a particular observation conditioned on a particular state, the probability of the observation sequence for this state sequence is

$$P(\mathbf{O}|\mathbf{Q}, \lambda) = \prod_{t=1}^{T} P(o_t|q_t, \lambda)$$

This can also be written as a product of B matrix entries as

$$P(\mathbf{O}|\mathbf{Q}, \lambda) = b_{q_1}(o_1)b_{q_2}(o_2)\cdots b_{q_T}(o_T)$$

The probability of the state sequence is a product of the initial state probability and entries from the A matrix

$$P(\mathbf{Q}|\lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \cdots a_{q_{T-1} q_T}$$

Since $P(\mathbf{O}, \mathbf{Q}|\lambda) = P(\mathbf{O}|\mathbf{Q}, \lambda)P(\mathbf{Q}|\lambda)$ we can sum this product over all possible state sequence to arrive at

$$P(\mathbf{O}|\lambda) = \sum_{\mathbf{Q}} P(\mathbf{O}|\mathbf{Q}, \lambda)P(\mathbf{Q}|\lambda) \tag{9}$$

Given that the model parameters are known, Equation 9 is of little use in calculating $P(\mathbf{O}|\lambda)$ for a test observation sequence. The number of mathematical operations required to evaluate Equation 9 is on the order of $2TN^T$. For a five state Hidden Markov Model and a sequence of 20 observations, the required number of operations is on the order of 10^{21}. Fortunately, dynamic programming algorithms have been developed to reduce the complexity of the calculation.

Baum and Eagon (8), Baum and Sell (10), and Baum (7) developed what is known as the *Forward-Backward Procedure*. Here, two intermediate quantities are defined, namely the forward variable $\alpha_t(i)$ and backward variable $\beta_t(i)$ where i represents an allowable state and t is the time index. Define

$$\alpha_t(i) = P(o_1, o_2, \ldots, o_t, q_t = S_i|\lambda)$$

which is the probability of observing the partial sequence and being in state S_i at time t given the model. It is shown in Appendix A that the forward variable follows

$$\alpha_1(i) = \pi_i b_i(o_1), \quad 1 \leq i \leq N$$

where N is the number of states in the Hidden Markov Model. By induction

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i)a_{ij}\right] b_j(o_{t+1}), \quad 1 \leq t \leq T-1 \text{ and } 1 \leq j \leq N$$

Summing the forward variable over all states at time T yields

$$P(\mathbf{O}|\lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{10}$$

The calculation of $P(\mathbf{O}|\lambda)$ using this method requires on the order of $2N^2T$ operations for evaluation. For the five state model and a sequence of 20 observations this is 1000 operations compared with 10^{21} as seen before. Equation 10 will be the method used to calculate $P(\mathbf{O}|\lambda)$ throughout this dissertation.

The backward variable is similarly defined as

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \ldots, o_T|q_t = S_i, \lambda)$$

that is, the probability of being in state S_i at time t and observing the remainder of the sequence. $\beta_T(i)$ is initialized to 1 for all states. For all other times the following recursive formula holds

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j), \quad t = T-1, T-2, \ldots, 1 \text{ and } 1 \leq j \leq N$$

With the solution to Problem 1 in hand and the forward and backward variables defined, attention can be turned to the solution of Problem 3. Determining a method of adjusting the model parameters to maximize the probability of a training observation sequence is the most difficult Hidden Markov Model problem to overcome. There is no known way to analytically solve for the model which maximizes the probability of an observation sequence. In fact, given any finite observation sequence as training data, there is no optimal way of estimating the model parameters (55:264). Baum and his colleagues, however, have determined an efficient iterative method of locally maximizing the model parameters in a maximum likelihood sense (6, 7, 8, 9, 10). For two models λ (estimate) and $\bar{\lambda}$ (re-estimated) they defined the *auxiliary* function

$$\hat{Q}(\lambda, \bar{\lambda}) = \sum_{\mathbf{Q}} P(\mathbf{O}, \mathbf{Q}|\lambda) \log P(\mathbf{O}, \mathbf{Q}|\lambda) \tag{11}$$

Baum and his colleagues have shown that maximization of Equation 11 results in three iterative equations used to update the model parameters given a training observation sequence. Embedded in these equations are the forward and backward variables previously defined. The following equations used to learn Hidden Markov Model parameters from a training observation sequence are collectively known as the *Baum-Welch re-estimation formula.*

$$\bar{\pi}_i = \frac{\alpha_1(i)\beta_1(i)}{P(\mathbf{O}|\lambda)}$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{t=1}^{T-1} \alpha_t(i)\beta_t(i)}$$

$$\bar{b}_j(k) = \frac{\sum_{t=1, O_t=k}^{T} \alpha_t(j)\beta_t(j)}{\sum_{t=1}^{T} \alpha_t(j)\beta_t(j)}$$

It was shown by proof that $P(\mathbf{O}|\bar{\lambda}) \geq P(\mathbf{O}|\lambda)$ with each iteration of the above rules (7). Equality holds when a local maximum in the parameter space is encountered. Derivations of the reestimation formula are found in Appendix A. The Baum-Welch reestimation formula will be the Hidden Markov Model learning method used in this dissertation.

2.3.1 Hidden Markov Model Varieties. This section will discuss the types of Hidden Markov Models that will be used in the experimentation for this dissertation. These include the discrete ergodic, discrete left-right, and continuous left-right Hidden Markov Model.

The type of Hidden Markov Model that has been discussed so far has the property that any state will be revisited with probability one and that such revisits are not required to take place at periodic time intervals (53:12). This type is known as an *ergodic* Hidden Markov Model. The ergodic model can be thought of as having a full transition matrix. The state diagram is the same as the Markov process shown in Figure 1. The observation probabilities for this model are probability mass functions determined by the Baum-Welch reestimation formula. The ergodic model is trained with a single long observation sequence.

For applications such as speech recognition, another type of model architecture has been found to account for observed properties of the signal better that the standard ergodic model (55:266). This model is called a *left-right* or Bakis Hidden Markov Model (55:266)(3) and has the property that as time increases, the state index increases or stays the same. The left-right Hidden Markov Model has the desirable property that it can readily model signals whose properties change over time (55:266). The transition matrix for the left-right model is upper triangular. Also, the initial state probabilities have the property of $\pi_i = 1$ for $i = 1$ and 0 otherwise. That is, the left-right model always starts is state one. The state diagram for a left-right Hidden Markov Model is shown in Figure 2.

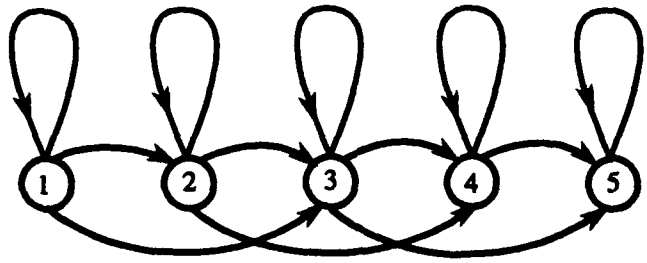


Figure 2. Architecture of a five state left-right Hidden Markov Model. Some links from lower to higher states have been omitted for clarity.

The left-right Hidden Markov Model is typically trained with many observation sequence samples from the signal to be modeled. The Baum-Welch reestimation formula require a minor modification. Denote the set of K observation sequences as

$$\mathbf{O} = \left[\mathbf{O}^1, \mathbf{O}^2, \ldots, \mathbf{O}^K\right]$$

where $\mathbf{O}^k = \left[\mathbf{O}_1^k, \mathbf{O}_2^k, \ldots, \mathbf{O}_T^k\right]$. The reestimation formula now follow

$$\tilde{a}_{ij} = \frac{\sum_{k=1}^{K} \frac{1}{P(\mathbf{O}^k|\lambda)} \sum_{t=1}^{T-1} \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{k=1}^{K} \frac{1}{P(\mathbf{O}^k|\lambda)} \sum_{t=1}^{T-1} \alpha_t(i) \beta_t(i)}$$

$$\tilde{b}_j(k) = \frac{\sum_{k=1}^{K} \frac{1}{P(\mathbf{O}^k|\lambda)} \sum_{t=1, O_t=k}^{T} \alpha_t(j) \beta_t(j)}{\sum_{k=1}^{K} \frac{1}{P(\mathbf{O}^k|\lambda)} \sum_{t=1}^{T} \alpha_t(j) \beta_t(j)}$$

The closeness (classification) measure for this multiple observation case is calculated by

$$P_k(\mathbf{O}|\lambda) = \prod_{k=1}^{K} P(\mathbf{O}^k|\lambda)$$

Both the discrete ergodic and discrete left-right Hidden Markov Model reestimation formula suffer from an implementation problem. Even in problems with a moderate number of states and relatively small observation sequence lengths, the forward-backward variables exponentially decrease toward zero and cause underflow errors in most computers. Ra-

biner (55:272) suggests a scaling technique to eliminate this problem. This dissertation circumvents this problem by performing all calculations in logarithmic arithmetic. The summations in the reestimation formula can be calculated with the following rule where it is assumed $x > y$.

$$\log(x + y) = log(y) + \log\left(1 + 10^{\log(x)-\log(y)}\right)$$

The logarithmic arithmetic versions of the Baum-Welch reestimation formula are found in Appendix A.

The final type of Hidden Markov Model used in this dissertation is the continuous left-right model (54). Here, the state transitions follow the same property as its discrete version, but the observation probabilities are modeled with Gaussian mixtures. The Gaussian mixtures allow for the creation of probability density functions in place of the probability mass functions of the discrete case. Until now it has been assumed that the data in the observation sequence was one-dimensional, or multi-dimensional data that had been vector quantized. The continuous Hidden Markov Model can process multi-dimensional data and may have advantages in problems where vector quantization causes serious loss in the information contained in the observation sequence.

The discrete ergodic and discrete left-right Hidden Markov Model training and classification algorithms are implemented in the *C* programming language on a variety of computer platforms that include NeXT, Sun Sparcstation 10, and Silicon Graphics workstations. The continuous Hidden Markov Model was implemented on a Sun Sparcstation 10 using Entropic's HTK Hidden Markov Model Toolkit (23).

2.3.2 Classification Methodology. Once the model for each class has been defined with training observation sequences, the probability of a new observation sequence, $P(\mathbf{O}_{new}|\lambda)$, can be calculated. The correspondence between an observation sequence and the probability of generation allows identification of new sequences relative to those used to train models. The recognition of an unknown observation sequence results from choosing the

maximum $P(\mathbf{O}_{new}|\lambda)$ over all models which can be written as

$$C = \underset{1 \le i \le N}{\text{argmax}}\, P(\mathbf{O}_{new}|\lambda^i) \tag{12}$$

where i is an index over the N object models and C is the classification. A block diagram of the process is shown in Figure 3. This method of classification will be used in this dissertation.

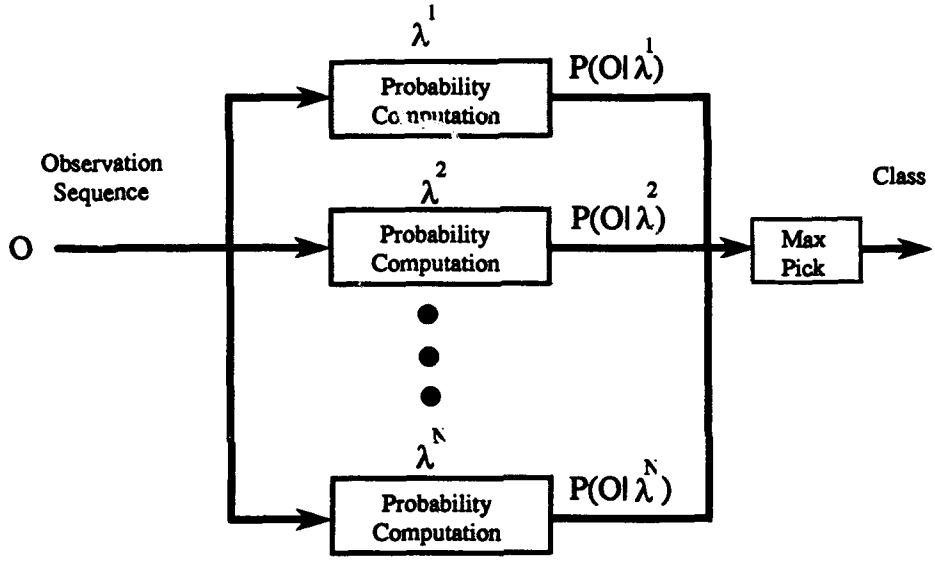


Figure 3. A block diagram of a Hidden Markov Model classification system. The new observation sequence is compared to every model. Classification corresponds to the model with the largest response.

Note that this form of classification is not Bayes optimal. The method would be Bayes optimal if an infinite amount of data were available, the data were generated by a 1st order Markov source, and the Hidden Markov Model training procedure found a global maximum in the parameter space (24:372).

2.4 Single Look Classifiers

To measure the classification advantage of the Hidden Markov Model technique, two single look classifiers will be used as a performance baseline. The two single look techniques are distinguished by the dimensionality of the data used. The single look *vector quantizer*

(VQ) classifier uses one dimensional data and is used as the benchmark for comparison with the discrete ergodic and discrete left-right Hidden Markov Model classifiers. A one nearest neighbor technique will be used as a benchmark for comparison with the continuous Hidden Markov Model classifier (21:98).

Hidden Markov Models will be trained and tested using different numbers of sequences of various lengths. Hidden Markov Model classification rates are determined by the percent of the sequences correctly classified. To find the equivalent single look classification rate, a single look classification will be made on the object in each frame of the image sequences considered. The percentage of frames correctly classified from the total number examined will define the single look classification rate of image sequences.

2.4.1 Vector Quantizer Classifier. Discrete Hidden Markov Models accept sequences of one dimensional data for processing. Before using multi-dimensional data, it must be vector quantized to convert the sequences of high dimensional data to sequences of codewords or integers. For implementing a Hidden Markov Model classifier, data from all classes under consideration are used in creating the vector quantizer codebook. The particular method of creating a vector quantizer for this dissertation is the LBG algorithm (44). The LBG algorithm is a k-means like procedure.

The number of clusters used in creation of the vector quantizer is directly related to the desired fidelity of the quantized representation of the original multi-dimensional data. When the vector quantizer is created, it is usually the case that some clusters will contain more that one class of data. Each cluster in the vector quantizer will be given a label according to which class has the most representation. With each cluster labeled by class, multi-dimensional data gathered from an object in a test image is vector quantized. If the codeword produced by the vector quantization process is associated with the same class as the test object, the classification is successful. If the codeword produced is associated with another class, a classification error occurred. This process will be known as the single look vector quantizer classifier.

2.4.2 One Nearest Neighbor Classifier. The one nearest neighbor (1-NN) classifier is chosen to be the benchmark for the classification performance of the continuous left-right Hidden Markov Model. The set of prototype feature vectors of dimension m from each of the N classes is denoted by Y^i, $i = 1, \ldots, N$. Each prototype set contains k_i feature vectors. A feature vector from a test object is collected and compared to every prototype feature vector from each class using a Euclidean distance measure. The distance of the test vector to each class, Y^i, is defined by

$$d(\mathbf{x}, Y^i) = \min_{j=1,\ldots,k_i} \|\vec{\mathbf{x}} - \vec{y}_j\|^2 \quad \text{where} \quad \vec{y}_j \in Y^i$$

The class, C, of the test object is determined by choosing the class with minimum distance to the test vector using the following relation

$$C = \underset{1 \le i \le N}{\operatorname{argmin}}\, d(\mathbf{x}, Y^i)$$

A correct association is a classification success. A match with an out of class object is a classification error.

The 1-NN single look classifier is straightforward to implement and is not computationally expensive for moderate feature vector and class prototype sizes. The 1-NN algorithm is a suboptimal procedure whose use will usually lead to an error rate greater than the Bayes rate (21:98). With unlimited prototype sets, the error rate is never worse that twice the Bayes rate.

2.5 Multiple Frame Single Look Classification

An alternate sequence identification technique is also investigated in this dissertation with its classification performance compared to that of the Hidden Markov Model. The multiple frame single look technique is an extension of the single look classifiers described in the previous section. Here, each frame in an image sequence is classified according to the appropriate single look technique, 1-NN or VQ. A classification decision made for the

entire sequence is based on which class is identified in a plurality of the frames considered. If the class with the plurality of decisions is the same as the test class, a proper classification is made. It is important to emphasize that this is a *multiple frame* technique and not a true *sequential* classification technique as is the Hidden Markov Model. The multiple frame single look classification technique will provide the same answer regardless of the order in which the frames are processed. The Hidden Markov Model inherently is concerned with the processing order since it is a form of syntactic pattern recognition.

The multiple frame classification method has roots in a discipline known as sequential analysis (72). The multiple frame single look classification method corresponds to a sampling inspection technique called the *single sampling plan.* The purpose of the single sampling plan is to inspect a lot of some item for quality control. N items are selected from the lot and examined for defects. The lot is discarded if more than c, $c < N$, defects are found. The interesting point is that all N items are inspected before a decision is made. This scenario is equivalent to a two class recognition problem. In a sequence of frames of size N, each frame is classified with the single look 1-NN or VQ technique. The class that wins more than $N/2$ of the frames becomes the class label for the sequence. Considering Class 1 of the two class problem and assuming independence, it is interesting to note that the probability that a certain number of frames will be classified correctly obeys the Binomial probability density function given as (72:7)

$$P(X_1 = k) = \frac{N!}{k!(N-k)!} p^k (1-p)^{N-k} \tag{13}$$

where p is the single look probability of correct classification of Class 1 while $(1-p)$ is the single look probability of error. X_1 denotes Class 1 and k the the number of correctly identified frames from the sequence of size N. The probability of correctly classifying the sequence is the sum of the probabilities found by Equation 13 where $k > N/2$. For the multi-class problem, the probability of correctly classifying a certain number of frames follows the Multinomial probability density function whose form is (60:30)

$$P(X_1 = k_1, X_2 = k_2, \ldots, X_l = k_l) = \frac{N!}{k_1! k_2! \cdots k_l!} p_1^{k_1} p_2^{k_2} \cdots p_l^{k_l} \tag{14}$$

where X_i are class indicators, k_i are number of the N frames determined to be Class i, and p_i are the single look correct classification rates. The probability of classifying a sequence as a certain class is found by summing Equation 14 for all combinations of k_i where $k_1 > k_i$ for $i = 2, 3, \ldots, l$. The effects of sequence length and single look error rate are demonstrated in the following example. Consider a three class problem where these effects are shown on class 1. To Examine the effects changing the number of frames in the sequence, N, has on the sequence classification rate, consider Figure 4. Figure 4 graphically shows this effect where

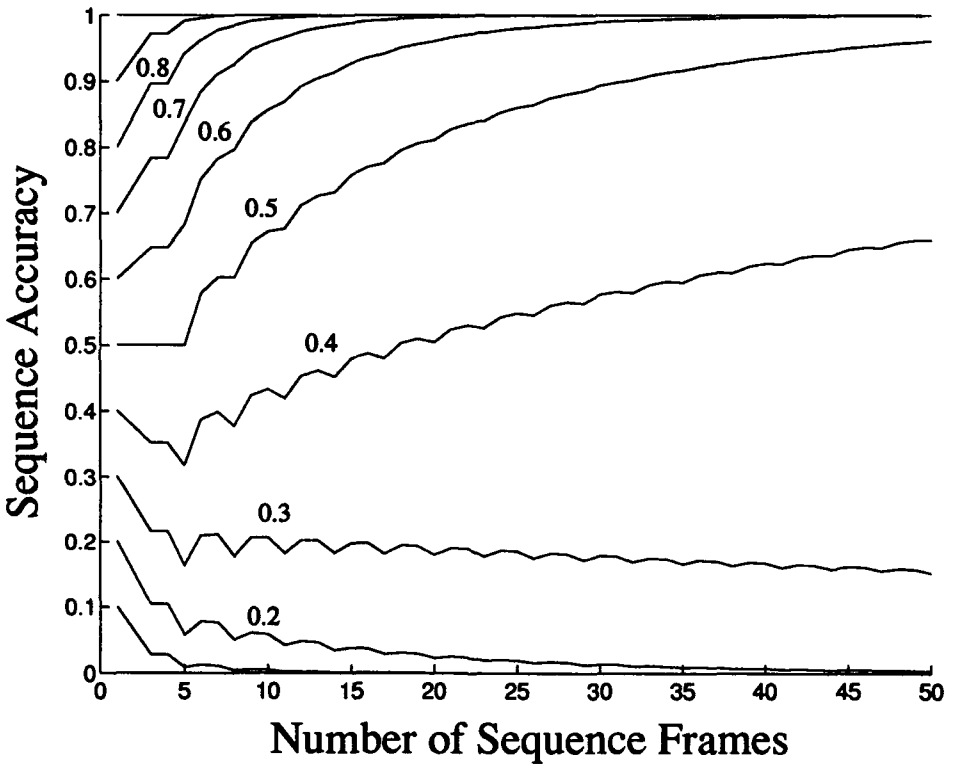


Figure 4. The effect of varying the total sequence frames on the multiple frame classifier for a three class problem. 10 values of the single look classification rates are examined with the remaining error divided evenly between Class 2 and Class 3.

10 different Class 1 single look correct classification rates are considered. The corresponding error rate of Class 1 is divided evenly between Class 2 and Class 3. For a fixed single look correct classification rate above 0.2, the probability of correctly classifying a sequence increases with the number of frames in the sequence. For single look rates below 0.33, the

sequence classification rate decreases with increasing sequence length since the probability of the correct class winning a plurality of the frames decreases.

Now consider holding the number of frames in the sequence fixed and varying the single look classification error rate for Class 1. The single look error rate is varied between 0.0 and 1.0 in increments of 0.1. Figure 5 depicts this effect for sequence frame lengths from 5 to 50 in increments of 5.

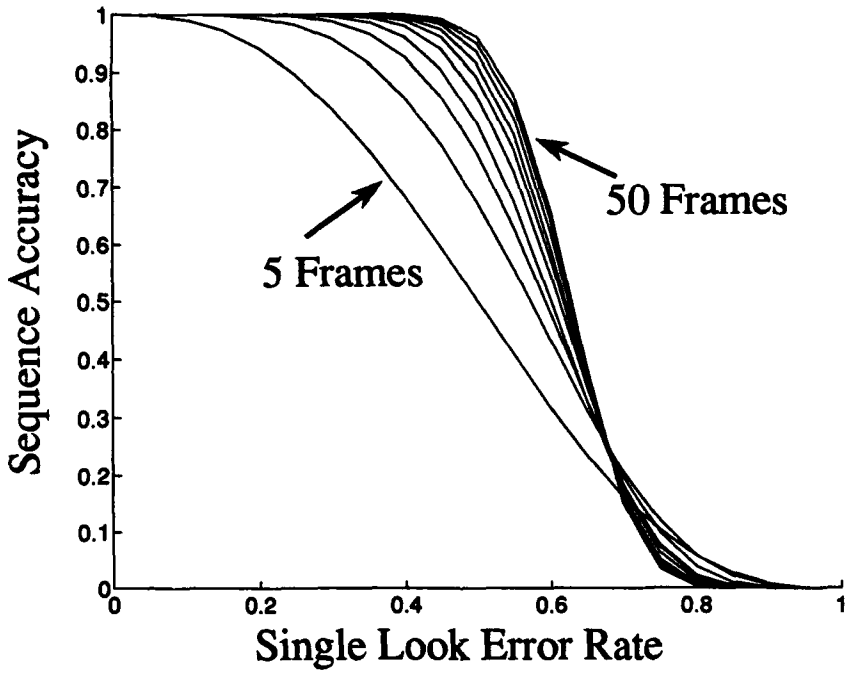


Figure 5. The effects of varying single look error rate of Class 1 on the Multiple Frame Sequence Classifier for a three class problem. The single look error rate is varied between 0.0 and 1.0 in increments of 0.1. Effects on sequence frame lengths from 5 to 50 in increments of 5.0 are shown.

Given a fixed single look error rate, the probability of correctly classifying a sequence again increases with the sequence length.

2.6 *Leave-One-Out Error Estimation*

Error counting methods of error estimation allow the calculation of a performance index for a given pattern classification algorithm. The performance index of interest here is the average probability of misclassification of the algorithm yielding an estimate of its future performance. There are many varieties of approaches to error estimation which include: resubstitution, hold-out, hold-one-out, rotation (18:343–357), and bootstrap (26:239–252) (22). The error counting method used in this dissertation is the leave-one-out method. A brief review of the well known resubstitution and holdout methods will help with the understanding of the leave-one-out technique.

The resubstitution method consists of using all known data to design and test the classification algorithm. This method often underestimates the actual error rate of the algorithm, producing an optimistically biased estimate, if the sample size is not large enough (66:473). Unfortunately, a number cannot be put on the minimum sample size for this method to yield accurate results. However, the minimum sample requirement rules of Cover (16) and Foley (25) provide a reasonable lower bound.

The hold-out method is a procedure where the sample data set is divided between a portion used to design the classifier with the remainder used as an independent test set for performance evaluation. A common splitting method for hold-out is to divide the data evenly between the design and test set, however, other researchers have used smaller fractions of the data for the design set (33) (71). When the sample size is not large enough, this method makes poor use of the data presumably because a better classifier could have been designed using all of the data. Thus, the hold-out method yields a pessimistically biased estimate of the true error rate. Devijver and Kittler report for small to moderate sample sizes the resubstitution and holdout methods may have very significant discrepancies in the observed error rate with the hold-out method yielding error rates an order of magnitude larger than the resubstitution method (18:355). The hold-out method can be made more reliable by averaging the error over many possible partitions of the data of fixed size (66:473) (71:286).

The leave-one-out error estimation method is a variant of the hold-out method where the hold-out portion is a single data sample. An estimate of the error is obtained by holding out 1 sample of data, training the classifier on the remaining data, and then testing the held out sample. This sequence is repeated for each piece of data. The leave-one-out method has been described in the literature by (18, 26, 27, 28, 38, 39, 47, 71) and is attributed to Lachenbruch (38). This method also appears to have been independently arrived at in the same year by the Russian researchers Lunts and Brailovskiy (47). The leave-one-out method has the advantage of reducing the bias of the hold-out error estimate since virtually all samples are used in each design and all samples are used in a test. The distribution of the design and test samples $\hat{D}$ and $\hat{T}$ are virtually identical and if the number of samples is large enough, approximate the true distribution. The reduction in bias of this estimator is achieved at the price of an increase in the variance (18:356). This increase is due to the correlation of the decision function between test trials (39:5). Generally, this correlation is small and its effects are limited. Because the leave-one-out method uses the available data in an efficient manner and has been shown to yield a tighter upper bound on the estimation of the Bayes error rate, it will be the error estimation method of choice used to estimate the error rates of all algorithms in this dissertation.

III. A Hidden Markov Model Distance Measure for Classification Analysis

3.1 Introduction

A measure of the distance between pairs of Hidden Markov Models gives insight into the sensitivity of the model to changes in parameters (35). Additionally, the distance measure is an important tool for analyzing the performance of Hidden Markov Models in a multi-class pattern recognition problem. If the distance measure between two Hidden Markov Models is small, the recognition system can be expected to have a worse classification accuracy than when the distance is large. This dissertation will use the distance measure as a classification analysis tool. A new, more robust method for measuring the distance between pairs of Hidden Markov Models is presented. Previous methods are briefly reviewed to put the new method in context. The new method described here does not depend on a specific Hidden Markov Model architecture and is valid for Hidden Markov Models that use discrete or continuous observation densities.

3.2 Previous Methods

The first distance measure used for comparing pairs of Hidden Markov Models was proposed by Levinson et al. (42). This method is a Euclidean distance measure of the state observation probability matrices between two models given by

$$d(\lambda_1, \lambda_2) \equiv \|B^1 - B^2\| \equiv \left\{ \frac{1}{MN} \sum_{j=1}^{N} \sum_{k=1}^{M} \left[b^1_{jk} - b^2_{p(j)k} \right]^2 \right\}^{\frac{1}{2}} \tag{15}$$

Equation 15 is called the "measure of estimation error". A minimum bipartite matching scheme is used to determine the optimum state permutation for aligning the states of two models. This measure does not depend on the initial state probabilities, Π, or the state transition matrix, A. The reason given is that the B matrix is the more sensitive parameter related to Hidden Markov Model closeness.

Juang and Rabiner (35) point out three problems with the distance measure in Equation 15. These are: 1) it does not take into account all of the Hidden Markov Model parameters; 2) its evaluation requires a great deal of computation in the discrete state observation density case and becomes intractable for continuous densities; and 3) it is unreliable when comparing Hidden Markov Models with highly skewed densities. Juang and Rabiner (35) propose a new distance measure, denoted as $d(\lambda_1, \lambda_2)$, to alleviate these problems which has the form

$$d(\lambda_1, \lambda_2) \equiv \frac{1}{T} \log P(\mathbf{O}_T^1|\lambda_1) - \frac{1}{T} \log P(\mathbf{O}_T^1|\lambda_2) \tag{16}$$

The superscript on $\mathbf{O}_T^1$ indicates the training sequence of T frames is from class 1. This distance measure is nonsymmetric in the sense that a similar evaluation with the training sequence from class 2 may not yield the same result. The extension of this measure to a symmetrized version, $D_s(\lambda_1, \lambda_2)$, is

$$D_s(\lambda_1, \lambda_2) = \frac{1}{2}\left[d(\lambda_1, \lambda_2) + d(\lambda_2, \lambda_1)\right]$$

where $d(\lambda_2, \lambda_1)$ has the form

$$d(\lambda_2, \lambda_1) \equiv \frac{1}{T} \log P(\mathbf{O}_T^2|\lambda_2) - \frac{1}{T} \log P(\mathbf{O}_T^2|\lambda_1)$$

Juang and Rabiner describe how their distance measure is derived from statistical analysis of probabilistic functions of Markov Chains and give an interpretation from the Kullback-Liebler statistic point of view (35:393). Their method is easily extendible to continuous state observation cases. Equation 16 is originally formulated for ergodic Hidden Markov Models which are trained with one sufficiently long sequence. Left-right Hidden Markov Model must be trained with multiple sequences. Juang and Rabiner's extension of Equation 16 for the left-right Hidden Markov Model takes the form

$$d(\lambda_1, \lambda_2) \equiv \frac{1}{\hat{T}} \sum_{n=1}^{N} \log P(\mathbf{O}_n^1|\lambda_1) - \frac{1}{\hat{T}} \sum_{n=1}^{N} \log P(\mathbf{O}_n^1|\lambda_2)$$

where $\hat{T}$ is the total length of the N sequences used for training. A more compact form is

$$d(\lambda_1, \lambda_2) \equiv \frac{1}{\hat{T}} \sum_{n=1}^{N} \log \frac{P(\mathbf{O}_n^1|\lambda_1)}{P(\mathbf{O}_n^1|\lambda_2)} \tag{17}$$

The distance measure of Juang and Rabiner (35:403) that is extended to the case of the left-right Hidden Markov Model has the flavor of a statistical distance measure of the Hidden Markov Model output distributions, but it is not. Notice that Equation 17 is not the difference between the means of the distribution of the responses of each Hidden Markov Model. The difference between the means would be an example a first order statistical analysis of the output distributions and would be of the form.

$$d(\lambda_1, \lambda_2) \equiv \frac{1}{N} \sum_{n=1}^{N} \frac{1}{T_n} \log P(\mathbf{O}_n^1|\lambda_1) - \frac{1}{N} \sum_{n=1}^{N} \frac{1}{T_n} \log P(\mathbf{O}_n^1|\lambda_2) \tag{18}$$

Equation 18 can be written as

$$d(\lambda_1, \lambda_2) \equiv \frac{1}{N} \sum_{n=1}^{N} \frac{1}{T_n} \log \frac{P(\mathbf{O}_n^1|\lambda_1)}{P(\mathbf{O}_n^1|\lambda_2)} \tag{19}$$

What in essence is a variation of the distance between the means was proposed by D'Orta et. al. (20) and has the form.

$$d(\lambda_1, \lambda_2) \equiv \frac{1}{N} \sum_{n=1}^{N} \log \frac{P(\mathbf{O}_n^1|\lambda_1)}{P(\mathbf{O}_n^1|\lambda_2)} \tag{20}$$

D'Orta's (20) measure, Equation 20, does not normalize out the effects of different sequences lengths as does the measure of Equation 19. Since the Hidden Markov Model output probabilities are directly affected by the sequence length, not normalizing spreads the output distributions and can detrimentally affect the analysis.

3.3 A New Method

This paper proposes a new method for evaluating the distance between a pair of Hidden Markov Models that is based on a statistical evaluation of the distribution of the output responses. For a Hidden Markov Model to perform well, there must be an adequate length or number of training sequences for the problem at hand from which information is extracted allowing a reasonable estimate of the model parameters. Once the Hidden Markov Model for each class is trained, test sequences of each class under consideration can be evaluated against each Hidden Markov Model. The measurement obtained from each Hidden Markov Model for a particular sequence is $\log P(\mathrm{O}|\lambda)$ which is normalized by its sequence length. When enough test sequences from a given class are evaluated against each Hidden Markov Model, a distribution of Hidden Markov Model output values begins to form. A statistical measure of the distance between the output distributions is a sound method of determining the distance between a pair of Hidden Markov Models where classification is the goal.

Basing a distance measure solely on the means of the two sequence normalized distributions in question can be misleading. Two distributions whose means are far apart and whose distributions have significant overlap would appear *better* than two distributions whose means were close and whose distributions did not overlap. This condition is seen in Figure 6.

The procedure proposed here does not suffer from this problem. Higher order statistics are used in conjunction with the *Bhattacharyya* distance for measuring the separability between the output distributions of a pair of Hidden Markov Models (26:99). The Bhattacharyya distance is derived from an analysis of determining an upper bound on the Bayes error rate of a two class problem and evolves from a special case of the Chernoff bound (26:98). This special case assumes Gaussian distributions and is considered to be an important measure of class separability. This distance measure is also reasonable to use with non-Gaussian distributions (26:103). The general form of the Bhattacharyya distance, denoted $d_B(\lambda_1, \lambda_2)$,

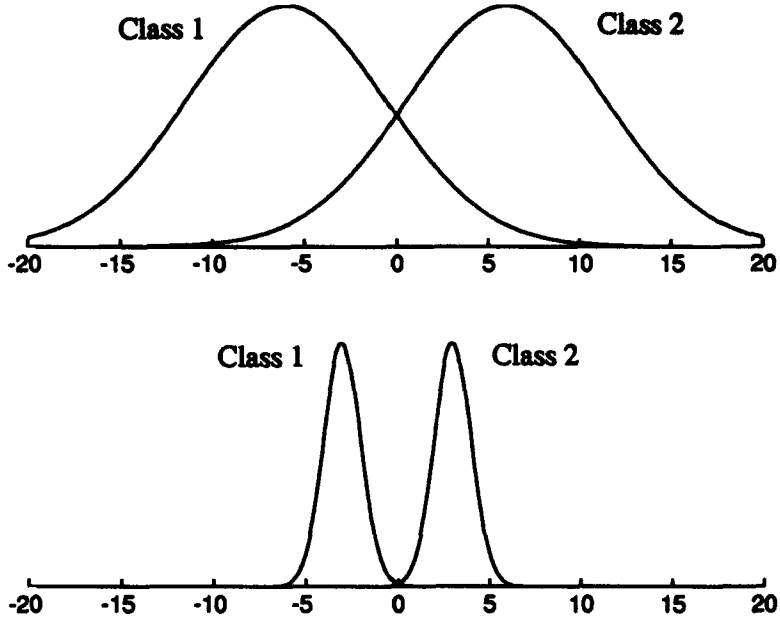


Figure 6. (Above) Example where the means of two distributions are far apart but have significant overlap. (Below) Example where the means of the two distributions are closer but have limited overlap.

is

$$d_B(\lambda_1, \lambda_2) \equiv \frac{1}{8}(M_2 - M_1)^T \left(\frac{\sum_1 + \sum_2}{2}\right)^{-1} (M_2 - M_1) + \frac{1}{2} \ln \left(\frac{\left|\frac{\sum_1 + \sum_2}{2}\right|}{\left(|\sum_1||\sum_2|\right)^{\frac{1}{2}}} \right) \quad (21)$$

M represents a class mean vector and $\sum$ represents a class covariance matrix. The first term of Equation 21 is a measure of the class separability due to the difference in the means while the second term is a measure of separability due to the covariance difference. The Bhattacharyya distance for the 1-dimensional case used in this dissertation is

$$d_B(\lambda_1, \lambda_2) \equiv \frac{1}{8}(M_1 - M_2)^2 \left(\frac{\sigma_1^2 + \sigma_2^2}{2}\right)^{-1} + \frac{1}{2} \ln \left(\frac{\frac{\sigma_1^2 + \sigma_2^2}{2}}{\left(\sigma_1^2 \sigma_2^2\right)^{\frac{1}{2}}} \right) \quad (22)$$

The value of this new distance measure is seen in the following example where the nonsymmetric distance measures are evaluated. Consider a three class problem where the output responses of classifier 1 are Gaussian with the means and variances shown in Table 1. Two thousand samples (simulating responses to observation sequences of length 20) from these distributions are generated for evaluation. The sample statistics are also reflected in Table 1. Notice in this example that the distance between the means of Class 1 and the other

Table 1. Means and variances of the output of classifier 1 to three classes of data. True and Sample statistics are shown.

Class	True Mean	True Variance	Sample Mean	Sample Variance
1	-3.000	1.000	-2.925	1.052
2	-4.500	1.000	-4.531	1.019
3	-4.500	0.063	-4.501	0.062

two classes are chosen to be ideally equal. This is the regime where the new measure, based on the Bhattacharyya distance, will have the greatest effect in producing a more reliable distance comparison. Figure 7 is provided to graphically show the output distributions of classifier 1 for the three class example. The two thousand data samples are processed by each of the previously described distance measures. The computed distances from Class 1 to Class 2 and Class 1 to Class 3 are shown in Table 2.

Table 2. Nonsymmetric distance results for the example problem.

Distance Measure	Class1-Class2	Class1-Class3
Means	0.080	0.078
Juang	0.080	0.078
D'Orta	1.610	1.576
Proposed	0.484	1.117

Examination of Figure 7 clearly shows a substantial difference in the overlap of the three class distributions. The Class 1 distribution overlaps Class 2 to a greater degree than it does Class 3. In the Bayes classifier sense, more classification errors would occur between Class 1 and Class 2 rather than Class 1 and Class 3. However, the only distance measure that

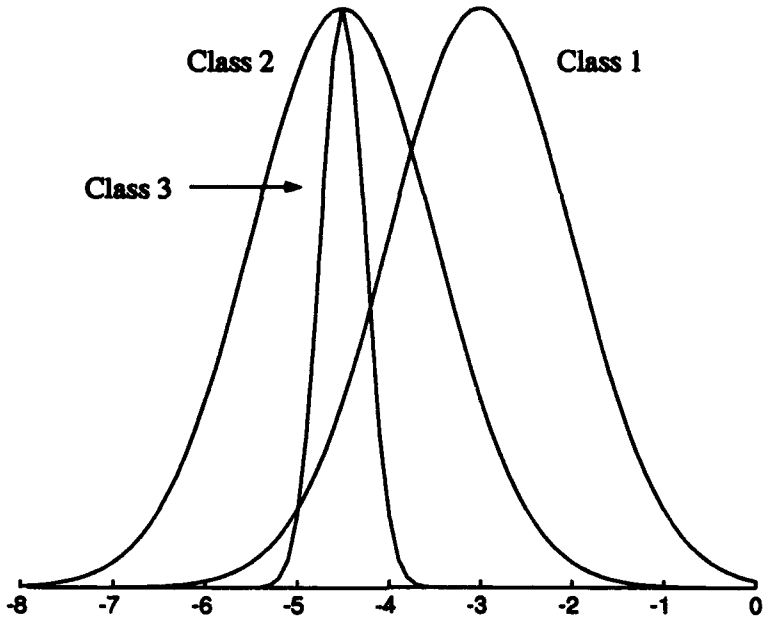


Figure 7. Plot of the output distributions of classifier 1 for the three class example.

clearly reveals this situation is the new measure based on the Bhattacharyya distance. This is because the other measures are essentially determining the distance between the distribution means while ignoring the contributions of the class variances.

3.4 Conclusion

A new method for analyzing the distance (distance) between a pair of Hidden Markov Models has been proposed. This method uses higher order statistics and the Bhattacharyya distance measure to find the distance between the output distributions of each Hidden Markov Model using the training sequences as inputs. A worst case example has demonstrated that this method is a sound approach yielding a realistic distance measurement between Hidden Markov Models when used for classification. The symmetric distance measures will be evaluated on experimental data in Chapter IV and in Chapter V and will aid in analyzing classification performance.

IV. Temporal Image Classification Using Moving Light Displays

4.1 Introduction

This chapter describes the experimentation and evaluation of the proposed spatio-temporal pattern recognition technique using a class of imagery known as moving light displays (MLDs). Individual frames of a MLD image sequence contain very little spatial information. The information content is highly temporal in that sense that image sequences are required for object identification. For this reason, the MLD class of imagery presents a challenging performance test for the Hidden Markov Model classifier. The continuous left-right Hidden Markov Model is used to evaluate the MLD imagery. The single look and multiple frame methods are also investigated with their performance compared with the classification accuracy of the Hidden Markov Models. It will be shown that the Hidden Markov Model classifier significantly outperforms the alternate techniques.

4.2 What is a Moving Light Display?

Moving Light Displays are image sequences which contain only selected points of a 3D object in each frame. MLD's have long been associated with psychophysical research into how humans recognize moving humans (4, 17, 29, 34, 56). The conclusions drawn are applied to the recognition of moving objects in general. A typical way to make a MLD is to attach reflective tape to a person's major joints, focus a strong light on the subject, and record a video sequence with the contrast adjusted so that only the reflective tape is seen (4:215). A single frame of such imagery is unrecognizable. However, sequences of this imagery gives (typically in 0.4 second) not only a perception of motion of a 3D object but allows recognition of the sequence as a person and a description of the type of motion (4:215). A single frame from a MLD sequence contains very little spatial information about the object. The image sequence, however, does contain a high level of temporal information sufficient for human recognition of the object and the type of motion it is displaying.

Most automated techniques applied to MLD imagery only determine the type of motion an MLD is undergoing in an image sequence. The motion of human MLD imagery is typically determined by analyzing the pendulum motion between pairs of points (29, 34, 37, 56). Heuristic rules are then applied to identify the specific type of motion. Bulpitt and Allinson have a method that uses a neural network to interpret the motion in MLDs (12). A measure of the relative position of each point to the center of object movement is the information provided to the neural network for motion identification.

4.3 Moving Light Display Classification

This dissertation, in contrast, investigates the automated recognition of MLDs using object shape features and the Hidden Markov Model classifier. MLD sequences of three geometric shapes are used in this experimentation. These shapes are the cube, sphere, and pyramid shown in Figure 8.

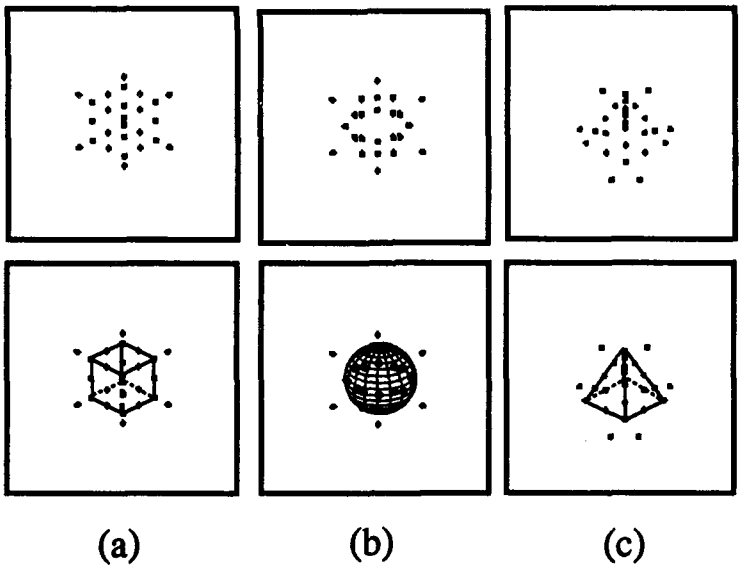


Figure 8. Moving Light Display imagery of the (a) cube, (b) sphere, and (c) pyramid (above). Connected versions of the shapes are shown below.

Each geometric object is surrounded by six stationary points to add additional confusion to single frame recognition by humans. The stationary points were also included in this

experimentation to be consistent with previous psychophysicological research. The cube in Figure 8 is constructed with 21 points, the sphere with 19 points, and the pyramid with 20 points. A model description of each geometric shape along with its stationary background points were generated in BRL-CAD (67). All MLD image sequences are produced through a C language program that accessed the BRL-CAD library functions. To demonstrate the lack of spatial information in the individual MLD frames, several views of the cube moving in a clockwise direction around the z-axis is shown in Figure 9. The z-axis (vertical) of the cube is canted toward the observer by 30 degrees.

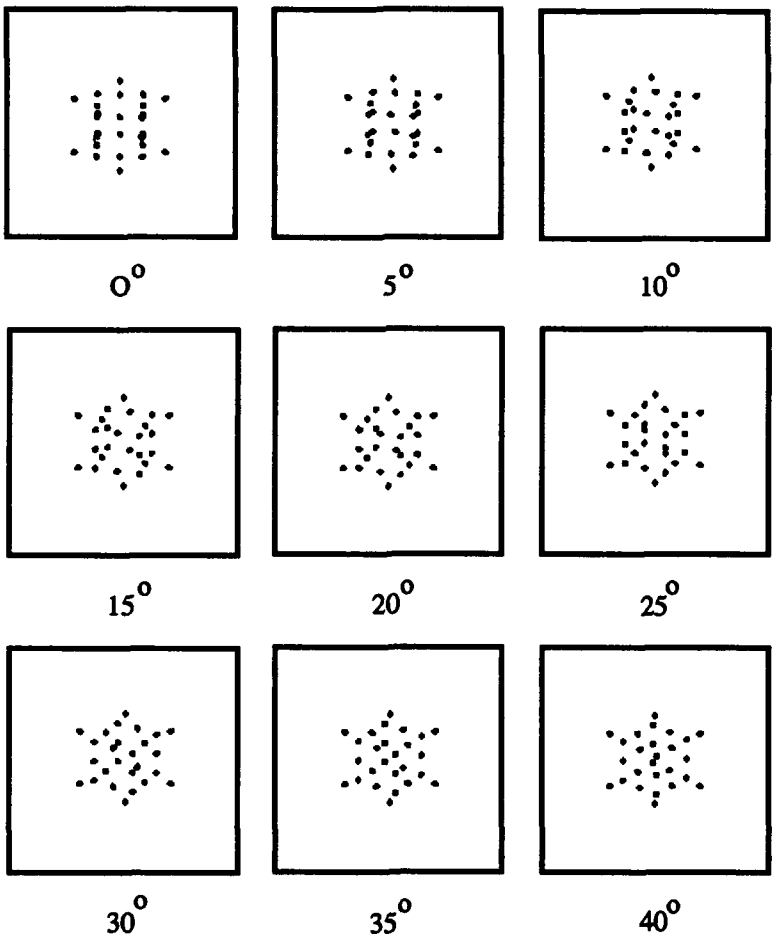
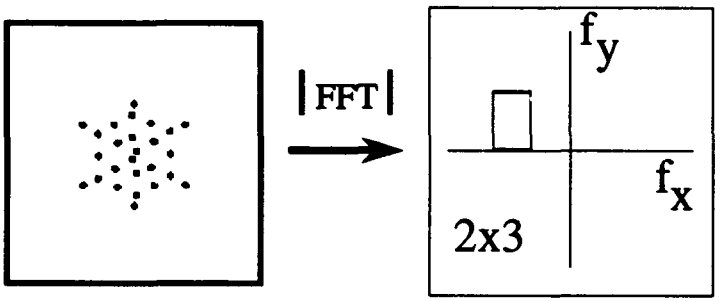


Figure 9. Nine views of the cube MLD as it rotates in a clockwise azimuth direction. The nine views are spaced in five degree increments.

4.3.1 Data Preparation. The MLD imagery used in this experimentation has pixel dimensions of 128 × 128. Six Fourier magnitude coefficients are chosen as feature vector components for the MLD imagery. The six features are gathered from a 2 × 3 rectangle whose lower right corner rests on the f_x axis at ($f_x = -1$, $f_y = 0$) as shown in Figure 10.



FFT Magnitude

Figure 10. Feature collection process for the MLD imagery. Six low frequency Fourier Magnitude coefficients are kept as shape features.

The upper left hand corner of the rectangle is at ($f_x = -2$, $f_y = 2$). These features were selected to avoid the DC component which simply measures the total energy in each image. Due to the nature of the MLD imagery, the DC component would not be significantly different for the three geometric shapes making it a poor choice as a feature. Each feature vector component is statistically normalized using the normalization formula

$$\hat{x}_i = \frac{x_i - \mu_i}{\sigma_i} \tag{23}$$

where x_i is the ith component, μ_i and σ_i are the mean and standard deviation of the ith component computed from all feature vectors under consideration.

The motion scenario for the MLD imagery is based on Callahan and Weiss's (13) modification of the aspect graph generation method of Koenderink and van Doorn (36). A viewer centered approach is adopted with the object at the center of a transparent sphere. As the observation point moves on the surface of the sphere, the observed object features change.

The region of interest for observing MLD object movement is restricted the portion of the viewing sphere for object azimuth angles 0° to 180° and elevation angles from 0° to 75°.

Left-right motion sequences are used to expresses *nominal* object movement for this investigation and are given the name because the azimuth angle associated with each frame in a sequence can only increase. The left-right sequence data set consists of 200 randomly generated sequences per class constrained to the area defined by azimuth angles 0° to 180° and elevation angles 0° to 75°. The initial observation position of a sequence is randomly chosen to begin in an azimuth angle range of 0° to 90° with any elevation in the previously described range allowed. The sequences are random in the sense that two uniform random numbers are used in determining the viewing orientation of each image in the following way. A uniform random number is used to pick an azimuth angle stepsize in the range of 4° to 6°. This range is chosen to allow a sequence to move over a significant portion of the viewing sphere in the azimuth direction. A second uniform random number is generated to choose if the elevation angle would rise, remain unchanged, or decrease. The probabilities for these three choices are 40%, 20%, and 40% respectively. The absolute elevation angle change is chosen to be 5° to obtain significant movement of the object through the sequence. For the 200 image sequences generated per class, 50 sequences each are generated with sequence lengths of 14, 16, 18, and 20 frames.

4.3.2 Experimentation and Results. The 200 left-right observation sequences for each of the Moving Light Display objects are generated. The six dimensional Fourier magnitude feature vector for each of the 3400 images generated per class are calculated. The multi-dimensional sequence data are prepared and formatted for the HTK - Hidden Markov Model Toolkit (23) used to implement the continuous left-right Hidden Markov Model. The continuous Hidden Markov Model architecture used in all MLD experiments is a five state left-right model. The state observation probability density functions are modeled with four Gaussian Mixtures. The covariance matrix for each Gaussian Mixture is constrained to be diagonal. The training of each continuous left-right Hidden Markov model consists of 20 iterations of the segmental K-means algorithm (23) and 50 iterations of the Baum-Welch

re-estimation formula. The segmental K-means algorithm is a clustering technique used to determine a good initial guess at the model parameters for training (54). The segmental K-means technique can also be used in a full training scheme and has been shown to yield good results, however, the technique has no proof of convergence (54). The Baum-Welch re-estimation algorithm is the traditional training method discussed in Chapter II, Section 3.

The single look 1-NN and multiple frame nearest neighbor algorithms are also investigated. These algorithms are implemented in the manner described in Chapter II. An initial investigation with the single look 1-NN classifier resulted in a classification accuracy rate of 85% over all classes. The error rate was estimated using the leave-one-out procedure. To further reduce the single look classification rate, white Gaussian noise is added to each feature vector. The addition of the Gaussian noise simulates reading the feature vectors through a noisy data channel. The reduction of the single look classification rate is intended to challenge the Hidden Markov Model based classifiers and demonstrate the advantages of classifying objects using image sequences.

4.3.2.1 First Noise Experiment. In the first of two experiments, zero mean Gaussian random noise with a variance of 2.25 is added to each component of every feature vector for each of the three classes to form a noisy test data set. Eleven unique noisy data sets are generated for each class using this process. The continuous left-right Hidden Markov Model, single look 1-NN, and multiple frame nearest neighbor technique classification accuracy rates are determined for each noisy data set using the leave-one-out error estimation method. All classifier are trained using noiseless data sequences. The mean and standard deviation of the classification accuracy of the 11 data sets produced by the three classifiers are shown in Table 3.

The classification accuracy of the Hidden Markov Model classifiers is 10% higher than the alternate multiple frame technique and substantially outperforms the single look 1-NN method. The distribution of the classification accuracy of the three techniques were compared using a one-sided t-test (63). The t-test is a statistical hypothesis analysis used to

Table 3. MLD classification results for the single look nearest neighbor (1-NN), multiple frame nearest neighbor (M-NN), and continuous Hidden Markov Model (C-HMM) classifiers. The mean and standard deviation of the 11 experimental trials is shown.

Classifier	Mean	σ
1-NN	56.3	0.7
M-NN	87.4	1.3
C-HMM	97.0	2.6

determine if the Hidden Markov Model's performance is significantly greater than the other two techniques. The t-test revealed there is a highly statistically significant difference in the classification accuracy the continuous Hidden Markov Model classifier and the 1-NN and M-NN techniques at a significance level of 0.01. This means that there is a 99% probability that the true error rate for the C-HMM classifier is greater than that for the M-NN classifier. The Hidden Markov Model classification accuracy is indeed superior to either alternative.

The relationship between the M-NN classifier and the multinomial distribution is investigated. The results reported in Table 4 are averaged over the 11 experimental trials.

Table 4. Experimental results (in percent) verifying of the relationship between the multiple frame classifier and multinomial distribution for the MLD Imagery.

1-NN Accuracy				Multinomial					M-NN
Object	Cube	Sphere	Pyramid	20	18	16	14	AVE	AVE
Cube	64.8	21.4	13.8	98.1	97.4	96.5	95.1	**96.8**	**98.9**
Sphere	21.1	48.4	30.5	76.7	75.2	72.9	69.4	**72.9**	**80.6**
Pyramid	14.1	30.8	55.1	85.6	84.0	82.0	78.5	**82.6**	**83.0**

These results, again, verify that the classification accuracy of the alternate multiple frame technique follows a multinomial distribution where the probabilities are derived from the single-look classifier.

The distance measures described in Chapter III are evaluated on the Hidden Markov Model outputs for each of the 11 trials. The measure of estimation error distance measure is not evaluated because it only applies to discrete Hidden Markov Models. The distance

measures will give insight into the classification decision process of the the Hidden Markov Models. Hidden Markov Model outputs obtained from the leave-one-out error estimation method are used in the analysis. The distance measure results are shown in Table 5.

Table 5. Distance Measure results for the three MLD objects averaged over the 11 data sets.

	Cube–Sphere	Cube–Pyramid	Sphere–Pyramid
Proposed	0.29	0.40	0.21
Mean	0.09	0.13	0.05
Juang	0.09	0.13	0.05
D'Orta	1.55	2.26	0.81

To help interpret the distance measure results, the classification confusion matrix is computed. This matrix provides information on the percentage of each class data set that is classified by each of the three object Hidden Markov Models. The results presented in Table 6 are averaged over the 11 data sets.

Table 6. The confusion matrix for the MLD data set for noise variance of 2.25. The results, in percent, are averaged over the 11 data sets.

Class Test\Class ID	Cube	Sphere	Pyramid
Cube	99.8	0.1	0.1
Sphere	0.4	95.5	4.1
Pyramid	0.2	3.0	96.8

The information from each distance measure in Table 5 leads to the same classification conclusions. This similarity of the conclusions is due to the structure of the output distributions of the three classifiers. In this case, the information in the variance is low. The proposed distance measure is essentially reduced to a difference between the means calculation of the output distributions. The four distance measures show that the cube Hidden Markov Model is relatively *far* from both the sphere and pyramid model. The distance between the sphere and pyramid is about half of that found between the cube and either shape. Therefore, more classification errors should occur between the sphere and pyramid than between these two objects and the cube. This conclusion is confirmed by examining the classification

identification matrix in Table 6. Very few of the cube sequences are classified as a sphere or pyramid. 4.1% of the sphere sequences are classified as a pyramid while 3.0% of the pyramid data sequences are classified as a sphere.

4.3.2.2 Second Noise Experiment. In the second of the two experiments, zero mean Gaussian random noise with a variance of 4.0 is added to each component of every feature vector for each of the three classes. Ten unique noisy data sets are generated for each class using this process. The continuous left-right Hidden Markov Model, single look 1-NN, and multiple frame nearest neighbor technique classification accuracy rate is determined for each data set using the leave-one-out error estimation method. The mean and standard deviation of the classification accuracy for the three classifiers are shown in Table 7.

Table 7. MLD classification results for the single look nearest neighbor (1-NN), multiple frame nearest neighbor (M-NN), and continuous Hidden Markov Model (C-HMM) classifiers. The mean and standard deviation of 10 experimental trials for a noise variance of 4.0 is shown.

Classifier	Mean	σ
1-NN	46.8	1.5
M-NN	72.2	4.6
C-HMM	93.4	5.8

This level of noise reduced the average single look classification accuracy below 50%. However, the average Hidden Markov Model classification accuracy rate is above 93%. This is an excellent and significant result. The variance of the M-NN and continuous Hidden Markov Model classifier appears to be high. This is due to the small sample size and the fact that two of the data sets produced unusually low classification rates. If the two data sets were left out of the analysis, the Hidden Markov Model results would have a mean of 96.1% with a variance of 0.47. The t-test was performed on the classification distributions to determine if the Hidden Markov Model outperforms the other two classifiers in a statistical sense. The t-test revealed there is a highly statistically significant difference in the classification accuracy of the continuous Hidden Markov Model classifier and the 1-NN and M-NN techniques at

a level of significance of 0.01. The Hidden Markov Model classification accuracy is, again, superior to either alternative.

The distance measures described in Chapter III are evaluated on the Hidden Markov Model outputs for each of the 10 trials. The measure of estimation error distance metric is not evaluated because it only applies to discrete Hidden Markov Models. The symmetrized distance measure results are shown in Table 8.

Table 8. Distance Measure results for the three MLD objects averaged over the 10 data sets. The Noise variance is 4.0.

	Cube-Sphere	Cube-Pyramid	Sphere-Pyramid
Proposed	0.21	0.25	0.18
Mean	0.06	0.08	0.03
Juang	0.06	0.08	0.03
D'Orta	1.03	1.47	0.48

The classification identity matrix averaged over the 10 data sets is shown in Table 9.

Table 9. The identification matrix for the MLD data set for noise variance of 4.0. The results, in percent, are averaged over the 10 data sets.

Class Test\Class ID	Cube	Sphere	Pyramid
Cube	98.4	1.2	0.4
Sphere	0.8	90.8	8.3
Pyramid	0.9	8.0	91.1

Again, the information from each distance measure in Table 8 leads to the same classification conclusions. The four distance measures show the cube Hidden Markov Model is relatively *far* from both the sphere and pyramid model. The difference here is not as great because the data is much more noisy. The distance between the sphere and pyramid is generally about half of that found between the cube and either shape. Therefore, more classification errors should occur between the sphere and pyramid than between these two objects and the cube. This is indeed the case which is confirmed by examining the classification identification matrix in Table 6. Only 1.6% of the cube sequences are classified as a sphere or pyramid.

8.3% of the sphere sequences are classified as a pyramid while 8.0% of the pyramid data sequences are classified as a sphere.

4.4 Conclusion

In this chapter, the performance of the spatio-temporal pattern recognition technique was explored with a class of imagery known as moving light displays. This class of imagery presents a great challenge to the Hidden Markov Model classifier because individual frames of MLD imagery contain very little spatial information and are not easily recognized. The sequence of MLD frames, however, contains a high level of temporal information leading to recognition. In the two experiments performed in this chapter, the single look classification rate of the MLD imagery was near or below 50%. In contrast, the Hidden Markov Model classification rate was above 93%. The alternate multiple frame technique classification rate was at least 10% below the Hidden Markov Models for noise variance of 2.25 and 20% below for a noise variance of 4.0. Analyzing the C-HMM and M-NN classification results using a *t*-test revealed that there is a highly statistically significant difference between the classification accuracies for the two approaches at a significance level of 0.01. The classification results produced with this difficult data set clearly demonstrate the power and robustness of the proposed sequence processing technique.

V. Spatio-temporal Image Classification

5.1 Introduction

This chapter describes the experimentation and evaluation of the Hidden Markov Model based spatio-temporal pattern recognition technique. The first section of this chapter discusses the methods used to prepare the data for a five class pattern recognition problem. The test objects are selected tactical military ground vehicles. The second section describes the experimentation and results using the discrete and continuous left-right and discrete ergodic Hidden Markov Models. The single look and multiple frame classifiers are also investigated with the results compared with the classification performance of the Hidden Markov Models. It will be shown that the Hidden Markov Model classifiers outperform the alternate techniques.

5.2 Data Preparation

This section describes the techniques used to prepare the synthetically generated training and test imagery for the classification experimentation. First, the method of calculating the Fourier magnitude coefficients used as shape features in this dissertation is described. Next, a method of regulating the range of motion of the test objects to conduct the investigation in a controlled environment is described. The regulation method is discussed in the context of creating the vector quantizer for use with the discrete classifier techniques. Finally, the procedure for adding noise to the imagery to test the robustness of the Hidden Markov Model classification system is reviewed.

5.2.1 Image Generation. To demonstrate the pattern identification capability of the Hidden Markov Model classification technique, a five class problem of identifying the tactical military ground vehicles shown in Figure 11 is investigated. The imagery is generated with a constructive solid geometry based computer aided design (CAD) package known as BRL-CAD (67). The objects are modeled in precise detail to their real counterparts and can be rendered at any scale and orientation. A computer program written in the *C* programming

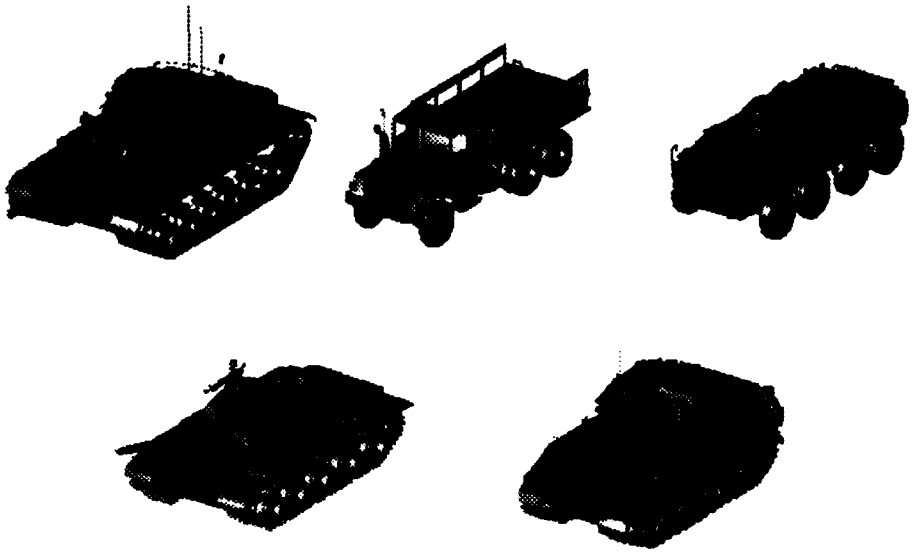


Figure 11. Objects used in this study: (Top) M60 Tank , M35 Truck, BTR60 Armored Personnel Carrier. (Bottom) T62 Tank, and M2 Infantry Fighting Vehicle (67).

language is used to generate test and training image sequences within the constraints of the regulation method described later. It should be noted that the military vehicles can be grouped into two categories, *wheeled* and *tracked* vehicles. This distinction is shown to be important as the results of the distance measures are analyzed.

5.2.2 Feature Collection. The object features selected to demonstrate the feasibility of the Hidden Markov Model technique are low frequency Fourier magnitude coefficients. These features are chosen because they have been shown to provide good shape discrimination (64) and are straightforward to generate. 256 × 256 pixel images, 8-bit greyscale, of the military vehicles are used in this investigation. Each image is thresholded at a greyscale level of 45 to produce a binary image, Fourier transformed, and the magnitude obtained. Twenty-eight low frequency coefficients, a 7 × 4 rectangle whose lower edge rests on the f_x axis and is centered on f_y, are retained to form the feature vector. The rectangle on and above

the f_x axis is chosen so the majority of features are not redundant as would be the case if a square region centered at ($f_x = 0$, $f_y = 0$) were used. This follows from the symmetry property of the Fourier magnitude spectrum for a real image $f(x, y)$ given as (30:77)

$$|F(f_x, f_y)| = |F(-f_x, -f_y)| \tag{24}$$

The feature vector is a concatenation of the four horizontal rows encompassed by the rectangle previously described. The top row of the rectangle constitutes the first seven components of the feature vector with the three remaining rows added accordingly. The feature collection process is illustrated in Figure 12. Each feature vector component is statistically normalized

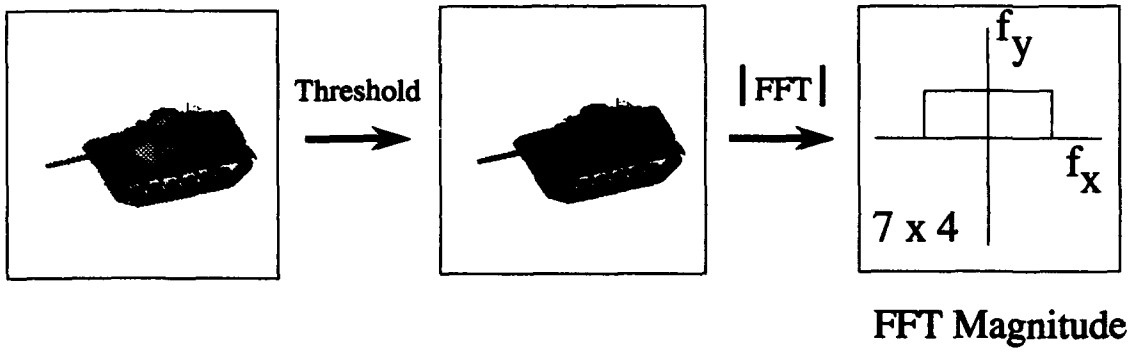


Figure 12. Feature collection process for the military vehicle imagery. Twenty-eight low frequency Fourier magnitude coefficients are kept as shape features.

using the normalization formula

$$\hat{x_i} = \frac{x_i - \mu_i}{\sigma_i} \tag{25}$$

where x_i is the ith component, μ_i and σ_i are the mean and standard deviation of the ith component computed from all feature vectors under consideration.

5.2.3 Motion Scenario and Vector Quantizer Design. To conduct this investigation in a controlled environment, the range of motion of the test objects is regulated. The regulation of motion is best described in the context of the requirements for the creation of

the vector quantizer needed to implement the discrete left-right and ergodic Hidden Markov Model classifiers.

The range of viewing perspectives chosen for this experimentation is based on Callahan and Weiss's (13) modification of the aspect graph generation method of Koenderink and van Doorn (36). A viewer centered approach is adopted with the object at the center of a transparent sphere as shown in Figure 13.

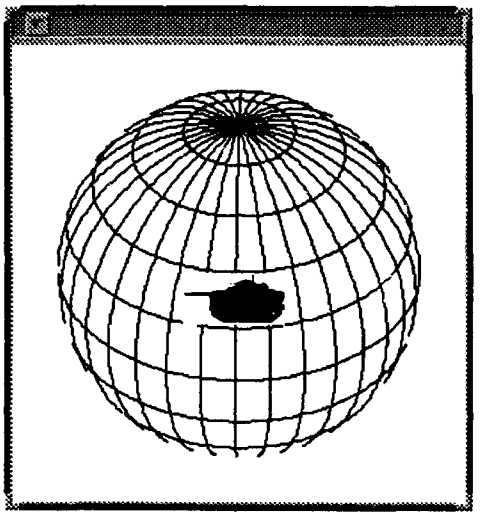


Figure 13. The motion scenario. The viewer centered approach places the object at the center of a transparent sphere. The observer moves on the surface of the sphere.

As the observation point moves on the surface of the sphere, the observed object features change. For the continuous classifiers investigated here, the observation sequences are simply sequences of 28 dimensional feature vectors. In the discrete case, a metric is used to group observations over a region of interest on the sphere's surface into areas of constant characteristic view or aspect that can be equated to Perrett's definition of a characteristic view described in Chapter I. Callahan and Weiss identified object singularities (surfaces, edges, vertices) and defined aspects as the grouping of vantage points that observed the same set of singularities (13). Gray (31) modified this approach by using cluster analysis on observed features to group views into aspects. Seibert and Waxman (59) used the ART2 clustering algorithm to identify the areas of constant aspect. This investigation uses the clustering algorithm proposed

by Linde, Buzo, and Gray (44), known as LBG, to create a vector quantizer whose clusters correspond to areas of similar aspect or characteristic view. The LBG algorithm was chosen because it is has been shown to provide good results for speech recognition (44).

The region of interest for observing object movement is restricted the portion of the viewing sphere for object azimuth angles 0° to 180° and elevation angles from 0° to 75°. The data used to create the vector quantizer begins with a 256 × 256 pixel image of each object generated for every 5° increment in azimuth and elevation, giving a total of 592 images per object. The 28 dimensional Fourier magnitude features are collected from each image. The 2960 feature vectors (592 from each class) are statistically normalized and then processed by the LBG algorithm to produce a 64 codeword vector quantizer. The 64 codeword vector quantizer was chosen because a 128 codeword vector quantizer showed little improvement in reducing the cluster distortion. Cluster distortion is calculated with the mean square error metric and is plotted versus codebook size in Figure 14.

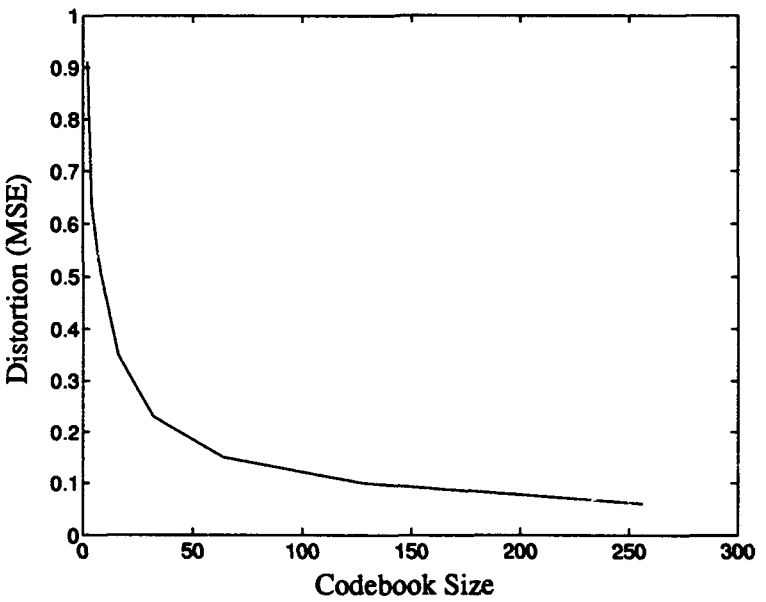


Figure 14. Distortion vs. Codebook Size for the vector quantizer designed with data from all 5 classes. A 64 codeword vector quantizer is selected for use.

Vector quantizing the same feature vectors used to create the codebook associates each viewing position with an aspect or characteristic view. An interesting way to examine the associated mapping of each viewing position with a particular codebook entry is through the viewing sphere plots shown in Figure 15. Each codeword is given a separate greyscale level.

Shades of gray common among the spheres in Figure 15 represent shared aspects. It is interesting to see that some viewing positions map to a common codeword for many of the classes while some codewords are associated with very different viewing orientations. For the 64 codeword vector quantizer designed here, 51 of the clusters (80%) were ambiguous and 13 clusters (20%) were unambiguous. Ambiguity arises when more than one object class occupies a given cluster. The 13 unambiguous clusters were distributed by class in the following manner: M60 - 6, M35 - 3, BTR60 - 1, T62 - 1, and M2 - 2.

5.2.4 Correlated Gaussian Noise. To test the robustness of the Hidden Markov Models, a classification evaluation using noisy test sequences is accomplished. All classifiers will be trained with noiseless data sequences and tested with the same sequences to which correlated Gaussian noise is added prior to the thresholding and feature collection steps. Correlated Gaussian noise is chosen over white Gaussian noise since pixels in the background of real imagery are correlated. The correlated noise was produced following the procedure described by Weeks et. al (70). A 256×256 pixel zero mean white noise image is generated using a Gaussian probability density function as a source. A Fourier transform of this image is taken and a circularly symmetric Gaussian filter is used to low pass filter the frequency spectrum. The result is inverse Fourier transformed and added to an object image. The noisy greyscale image is thresholded at a greyscale level of 45 and the low frequency Fourier magnitude features are obtained. The noisy image generation process is shown in Figure 16.

Twenty cases that vary in signal-to-noise ratio (SNR) and correlation level are examined. The correlation level refers to the full width of the autocorrelation response of the filtered noise image at the half maximum amplitude point. Image sequences will be generated with SNRs of 20dB, 15dB, 10dB, 5dB, and 0 dB. For each SNR, the bandwidth of the low pass filter is

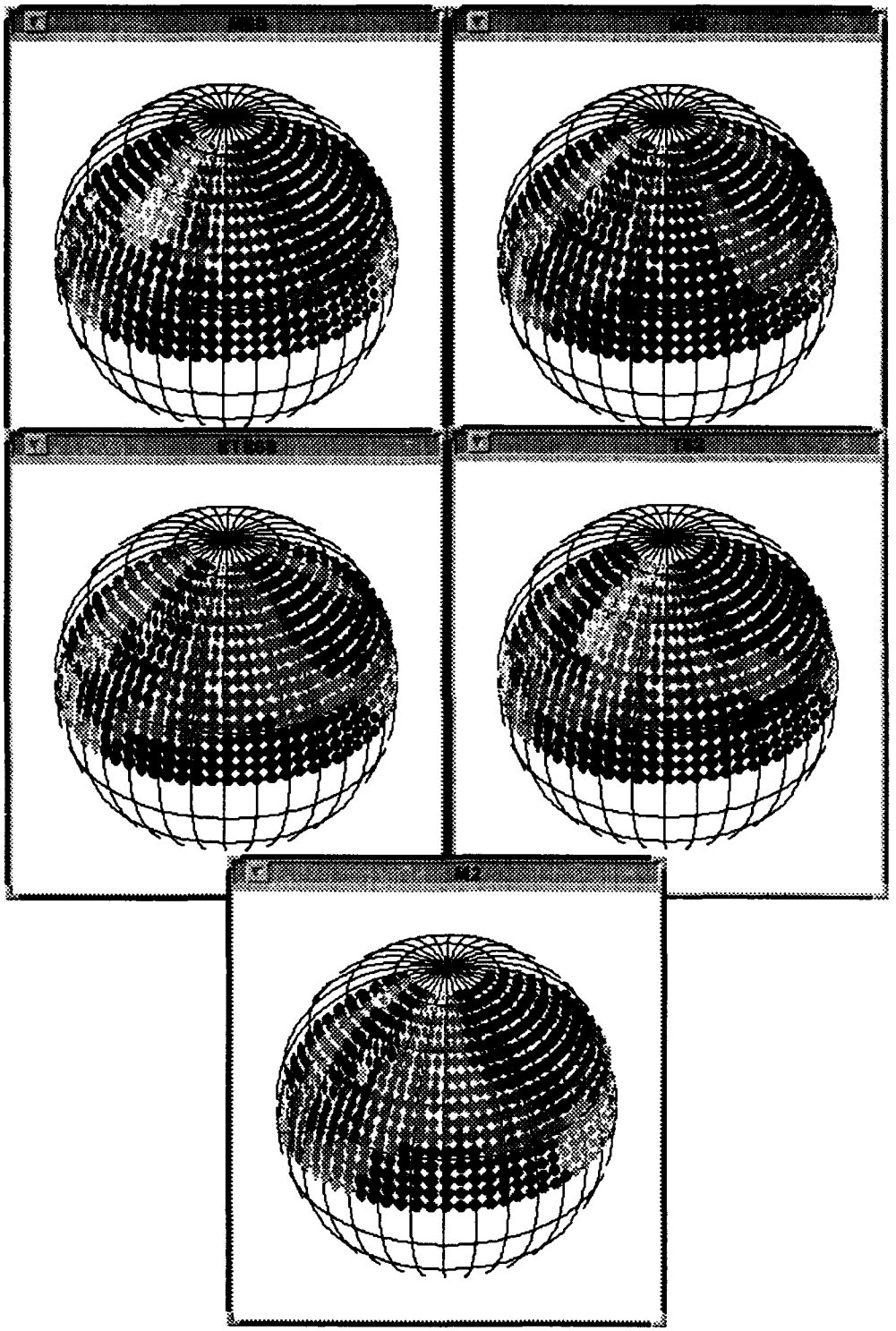


Figure 15. Aspect sphere plots for the M60, M35, BTR60, T62, and M2 vector quantizer training data. Common shades of grey represent shared aspects. (59)

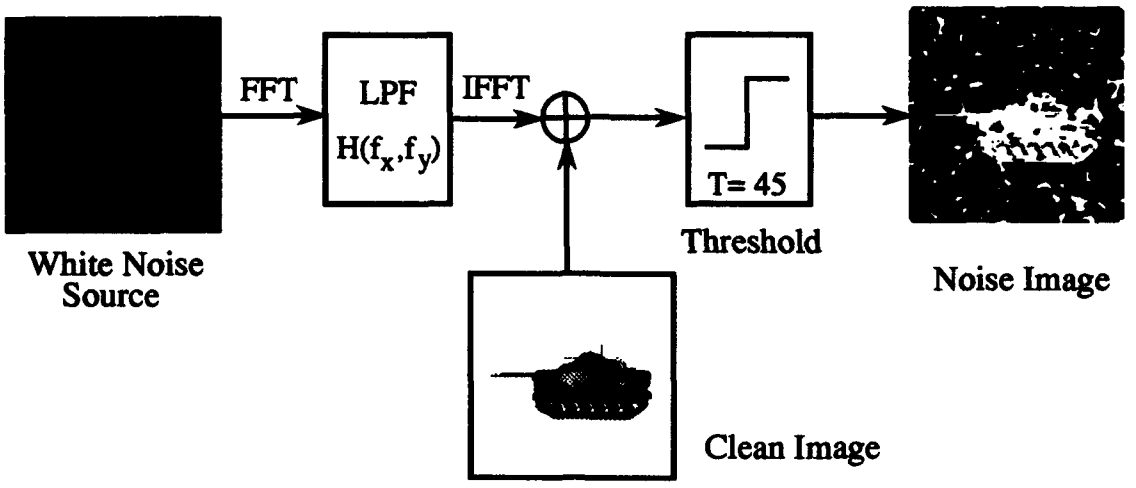


Figure 16. Method used to generated imagery with correlated Gaussian Noise. The low pass filter (LPF) is a circularly symmetric Gaussian filter.

adjusted to produce noise with a correlation level response of 2, 4, 8, and 12 pixels. A unique white noise image is used to add noise to each object image for every SNR and correlation level combination. The SNR is calculated using

$$SNR = 20\log_{10}\left(\frac{Object_{average}}{\sigma_{noise}}\right) \tag{26}$$

Because the noise is zero mean, the SNR is only a function of the variance of the original noise source and does not vary with correlation level. The effect of the correlation level is to increase the low frequency content of the noise in the thresholded image. Figure 17 shows the effects of increasing the correlation level for an M60 tank in 10dB noise.

5.3 Experimentation and Results

This section describes the experimentation and results of the five class pattern recognition problem using the Hidden Markov Model classifiers. Five separate types of observation sequences are examined. These are the left-right, right-left, vertical, acceleration, and transition-only sequences. The left-right sequence type is used to gain a basic understanding the the Hidden Markov Model classifiers. Single-look and alternate multi-frame techniques

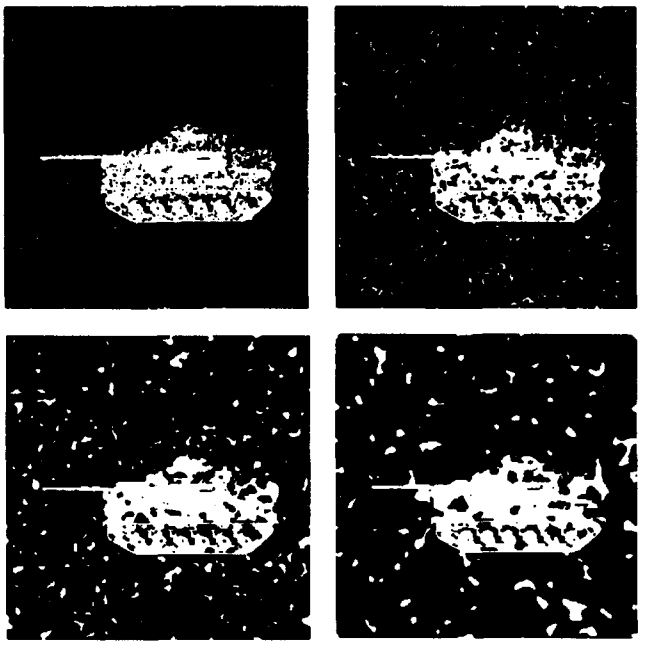


Figure 17. The M60 tank (thresholded) in 10dB noise for correlation levels (top) 2 and 4 pixels and (bottom) 8 and 12 pixels.

are also evaluated on the left-right sequences for comparison. The remaining four sequence types are investigated only with the Hidden Markov Models to test the classification robustness using sequences that differ from the training set. Because of the left-right sequences' importance, it will now be described in detail.

5.3.1 Left-Right Observation Sequences Left-right observation sequences are the major sequence type examined in this dissertation. The left-right sequence expresses *nominal* object movement for this investigation and are given the name because the azimuth angle associated with each frame in a sequence can only increase. When looking at a sphere plot, such as the one in Figure 13, azimuth increases from the left to the right hand side of the illustration. The left-right sequence should not be confused with or strictly associated with a left-right Hidden Markov Model. Although ergodic Hidden Markov Models are trained with one sufficiently long observation sequence, the left-right observation sequences can be classified using ergodic models.

The left-right sequence data set consist of 200 randomly generated sequences per class constrained to the area defined by azimuth angles 0° to 180° and elevation angles 0° to 75°. The initial observation position of a sequence is randomly chosen to begin in an azimuth angle range of 0° to 90° with any elevation in the previously described range allowed. The sequences are random in the sense that two uniform random numbers were used in determining the viewing orientation of each image in the following way. A uniform random number was used to pick an azimuth angle stepsize in the range of 4° to 6°. This range was chosen to allow a sequence to move over a significant portion of the viewing sphere in the azimuth direction. A second uniform random number is generated to choose if the elevation angle would rise, remain unchanged, or decrease. The probabilities for these three choices were 40%, 20%, and 40% respectively. The absolute elevation angle change was chosen to be 5° to obtain significant movement of the object through the sequence. For the 200 image sequences generated per class, 50 sequences each were generated with sequence lengths of 14, 16, 18, and 20 frames. An example of several random sequences generated for the M60 tank are shown in Figure 18.

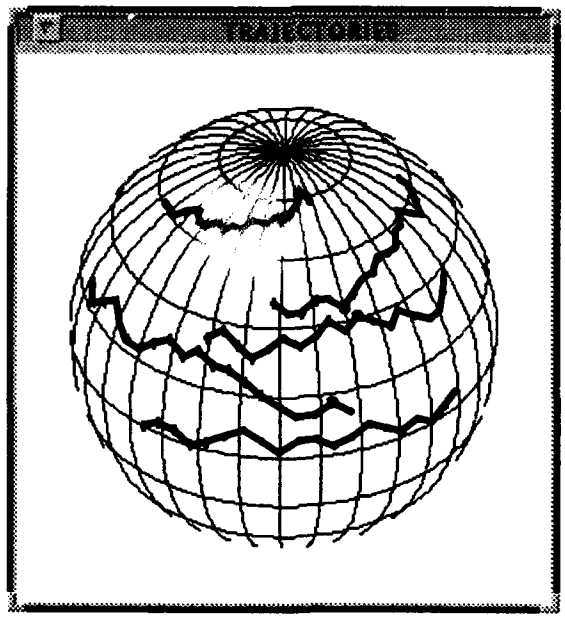


Figure 18. Example of several random sequence trajectories generated for the M60 tank.

The different training sequence lengths were used to give the Hidden Markov Model technique the robustness to recognize objects in varying length test sequences. An example of the individual images from a 12 frame sequence is shown in Figure 19.

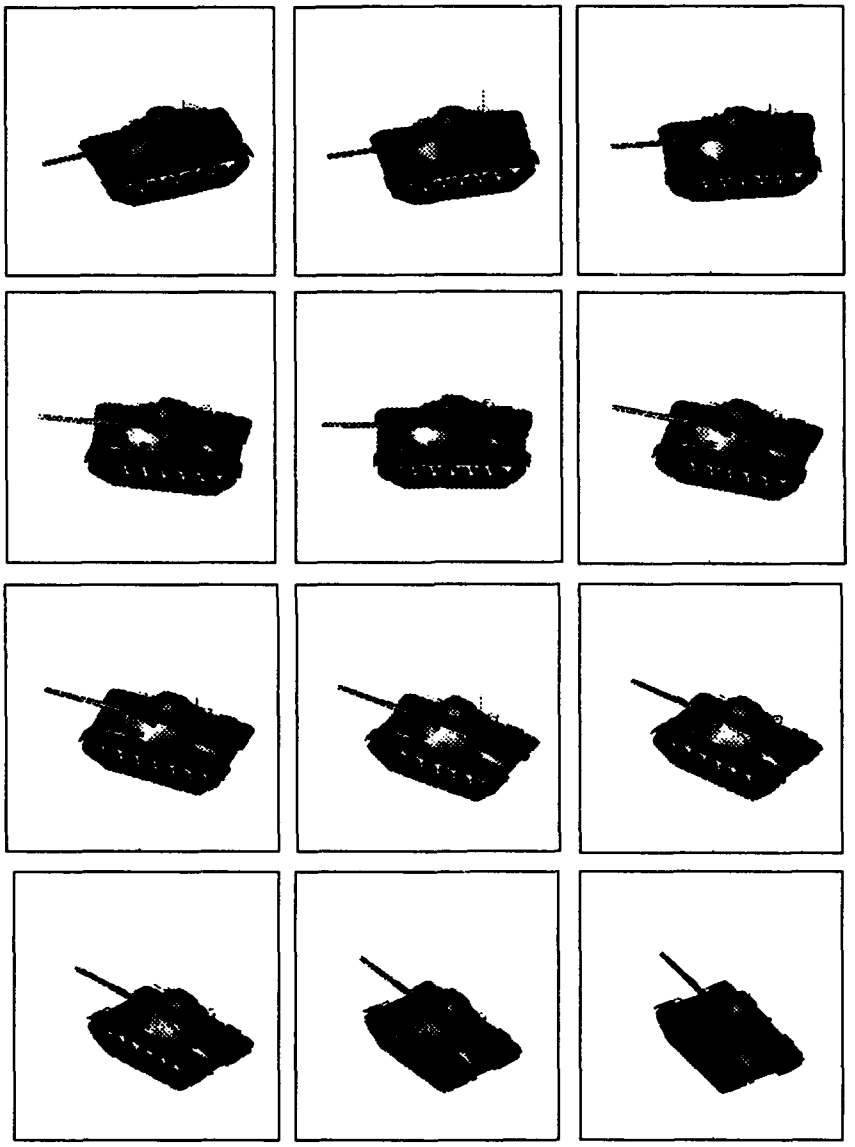


Figure 19. Example imagery from a 12 frame random sequence of an M60 Tank.

5.3.2 Left-right Discrete Hidden Markov Models. The discrete left-right Hidden Markov Model is the major type of model investigated for this five class problem. A five state

left-right model with 64 observation symbols per state is used. Five states were chosen to give the model the possibility of experiencing several self-transitions in each state based on the sequence lengths used. However, experimental evidence in a speech recognition application shows that the number of states can be varied with little effect on classification. Rabiner demonstrated that the effect of varying the number of states between 2 and 20 produced no more than a 2 percent change in the classifier error rate (55:278).

Noiseless left-right observation sequence data was prepared as previously described and the vector quantizer created. Noisy data was generated and vector quantized using the same codebook created with the noiseless data set. The classification performance was analyzed using a leave-one-out error estimation technique in the following manner. A single sequence from one class is left out of that class's 200 sequence training set. A Hidden Markov Model is then trained for that class, with separate models trained for each of the other four classes using all 200 noiseless sequences. The noisy sequences (all SNRs and correlation levels) corresponding to the sequence held out are then tested against all five classes with the results recorded. This procedure is repeated for all 200 sequences from each class. The results of this classification procedure are graphically shown in Figure 20.

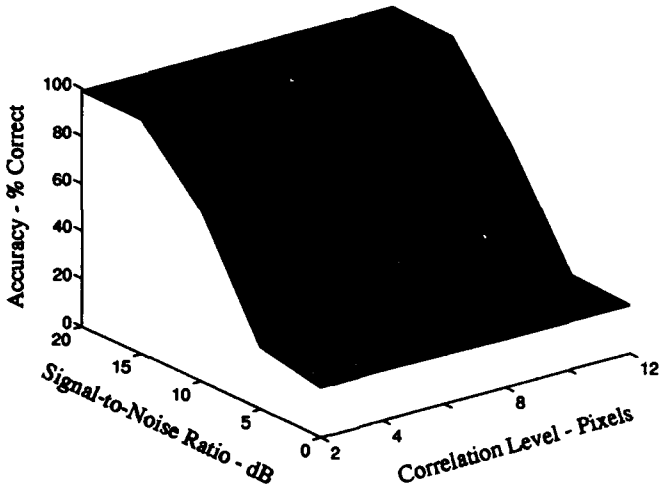


Figure 20. 3D plot of the left-right Hidden Markov Model classification accuracy.

Table 10 numerically shows the classification accuracy rates.

Table 10. Accuracy results for the left-right Hidden Markov Model with the left-right observation sequence data set.

SNR\CORR	2	4	8	12
20	98.7	98.7	98.7	98.6
15	98.4	98.4	98.2	98.1
10	71.1	68.3	66.2	64.1
5	25.6	24.7	21.8	21.3
0	20.6	20.7	20.3	20.2

Table 10 shows that the classification accuracy drops off as SNR increases as expected. This table also shows that the classification accuracy decreases slightly with increasing correlation level. This too is expected since the effect of increasing the correlation level is to increase the amount of low frequency noise that degrades the Fourier magnitude features used here.

In Chapter II, Section 2.2.1, the relationship of the Hidden Markov Model to syntactic pattern recognition was discussed. In particular, the equivalence between the Hidden Markov Model and a regular stochastic grammar was reported. Grammars were described as methods for examining strings for the presence and *ordering* of symbols belonging to a language. For this experiment, the vector quantizer codebook contains the symbols of the alphabet. Each object uses this common alphabet to construct its own language and grammar based on how the object features change when undergoing nominal movements. With a Hidden Markov Model trained to identify each language, classification decisions can be made since out of class sequences will be rejected in the parsing (classification) operation. This idea is fixed by the illustration shown in Figure 21.

One of the random sequences from the M35 truck is shown in Figure 21. Each shade of grey reflects a particular codeword at each azimuth and elevation angle. The right hand side of Figure 21 shows all of the azimuth and elevation angles of the BTR60 that are associated with the same codewords that make up the M35 sequence. It is clearly seen that non-physical movement is required for the BTR60 to pass through the regions in the correct codeword order.

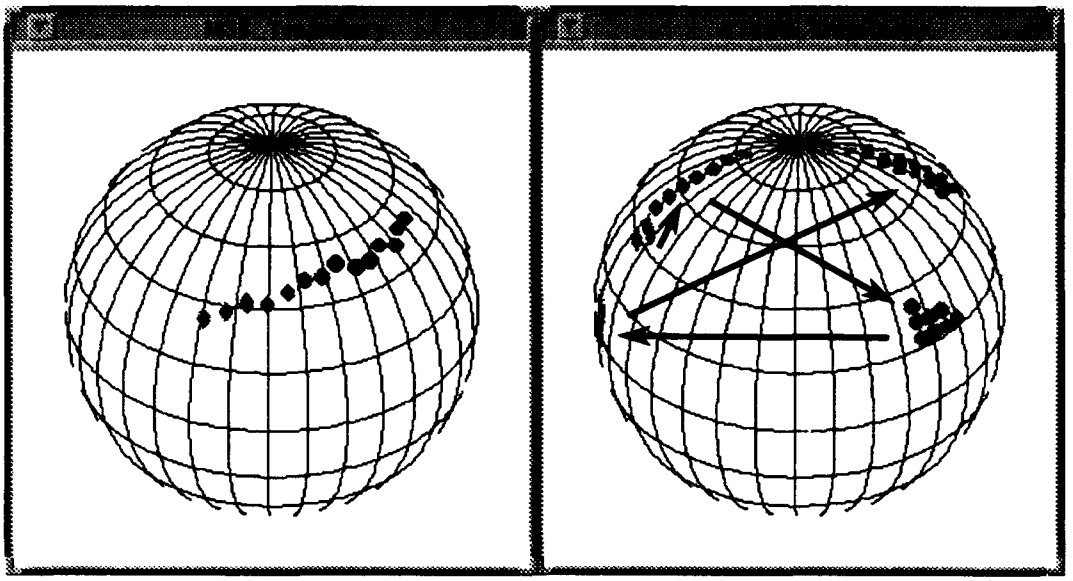


Figure 21. Mapping of an M35 trajectory (left) on the BTR60 sphere plot (right).

This example demonstrates that the Hidden Markov Model is not only learning to detect the presence of codewords, but their order of appearance as well.

5.3.2.1 Random Initialization of Model Parameters. To train a Hidden Markov Model, the state transition and state observation parameters must be initialized. The rows of both the A and B matrix must obey the stochastic constraint of summing to one. The method used to initialize the B matrix is to set each element of a row to $1/M$ where M is the number of possible observation symbols. M is 64 in this experimentation. Since the state transition matrix for a left-right Hidden Markov Model is upper triangular, the elements of each row are set to $1/\hat{N}$ where $\hat{N}$ is the number of states where a valid transition can be made.

An alternative way to initialize these parameters is to generate a random number for each element of every row of the state observation matrix and upper diagonal portion of the state transition matrix. Each row is then normalized to meet the stochastic constraints. An experiment is performed to examine the effect of initializing the Hidden Markov Models in this fashion. Using the leave-one-out error testing method, each Hidden Markov Model is

initialized with unique random numbers and normalized. Furthermore, the entire experiment is repeated 10 times. The mean and standard deviation of these 10 trials are reported in Table 11.

Table 11. Mean (left) and standard deviation (right) results from the 10 trials where the initial parameters in the state transition and observation matrices were randomly chosen.

SNR\CORR	2	4	8	12
20	99.8	98.7	98.8	98.7
15	98.7	98.2	98.6	98.8
10	71.7	63.8	63.5	64.0
5	21.0	22.1	23.4	22.1

SNR\CORR	2	4	8	12
20	0.22	0.17	0.22	0.14
15	0.28	0.10	0.22	0.17
10	0.85	1.07	1.05	0.96
5	2.34	2.52	2.73	2.76

Comparing Table 10 to the mean results in Table 11 shows that the classification accuracy is relatively insensitive to the method of initialization for this problem. It should be noted that only one trial is possible with the first initialization method. The standard deviation results show there is little difference in the results of the 10 trials for SNRs of 10 dB and above.

5.3.2.2 Alternate Classifier Performance. Because Table 10 clearly shows that the classification accuracy does not significantly change with correlation level, the comparison with the alternate classifier techniques will be performed for all SNRs at a correlation level of 2. The single-look and multiple frame vector quantizer classifiers are prepared in the manner described in Chapter II. The data used to create the vector quantizer are quantized. A frequency count of the distribution of each class in each of the clusters is made. A class label is assigned to each cluster according to which class has the most representation.

For the single-look vector quantizer classifier, classification decisions are made at the frame level. Each frame of the 200 left-right observation sequences for each class is vector quantized, associating each frame with one of the 64 codebook symbols. The codebook symbol for each frame, 3400 frames for each class, is then compared to the labeled clusters and a classification decision is made.

For the multiple frame vector quantizer classifier, each frame of a sequence is classified in the single-look fashion. A classification decision is made at the sequence level according to which class wins a plurality of the frames. Figure 22 illustrates the performance comparison between the discrete left-right Hidden Markov Model and the single-look and multiple frame vector quantizer classifiers.

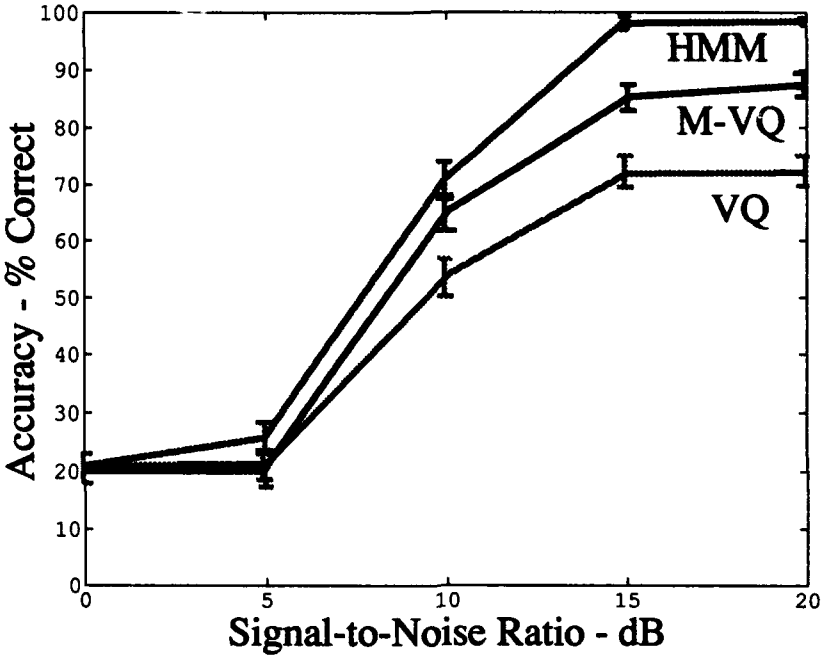


Figure 22. Performance Comparison of the discrete left-right Hidden Markov Model (D-HMM), single-look (VQ) and multiple frame (M-VQ) vector quantizer classifiers for a correlation level of 2. 95% confidence bars are shown.

Table 12 shows the same results in numerical form.

Table 12. Numerical classification results for the discrete left-right Hidden Markov Model (Hidden Markov Model), single-look (VQ) and multiple frame (M-VQ) vector quantizer classifiers for a correlation level of two.

Noise (dB)	20	15	10	5	0
D-HMM	98.7	98.4	71.1	25.6	20.6
M-VQ	87.5	85.2	64.8	20.0	20.1
VQ	72.4	72.3	53.6	21.0	20.8

Chapter II, Section 5, described a relationship that exists between the single-look frame based classification accuracies and the alternate multiple frame technique. For a multi-class problem, the multiple frame classification accuracy follows a multinomial distribution with the probabilities are derived from the single-look classifier where independence is assumed. This relationship is experimentally demonstrated here on the left-right data set for a SNR of 20dB and a correlation level of 2. The single-look and multiple-frame experiments summarized in Table 12 provided the data necessary for the analysis. It was previously shown that the multiple frame classification accuracy is a function of sequence length which is taken into account. Table 13 summarizes the verification that the multiple frame classification rate follows a multinomial distribution.

Table 13. Experimental verification the multiple frame classification rate following the multinomial distribution.

VQ - Accuracy						Multinomial					M-VQ
Object	M60	M35	BTR60	T62	M2	20	18	16	14	AVE	
M60	91.6	0.0	0.0	7.1	1.3	100.0	100.0	100.0	100.0	**100.0**	**100.0**
M35	0.5	74.3	19.7	1.5	4.0	99.3	97.9	96.8	95.6	**97.1**	**94.5**
BTR60	0.3	20.4	59.3	12.4	7.6	73.1	71.3	69.3	67.0	**70.2**	**73.5**
T62	29.3	1.3	11.9	51.4	6.1	76.9	67.1	73.2	69.8	**67.1**	**71.5**
M2	5.2	0.2	0.3	8.6	85.7	100.0	100.0	100.0	99.9	**99.9**	**100.0**

Table 13 is read in a left to right fashion for each row. The first section shows the single-look accuracy rates for the five vehicles. The middle section shows the output of the multinomial distribution for the four different sequence lengths assuming the accuracies in Section One. The final column in the second section is the average over the four sequence lengths. The third section shows the experimental multiple frame classifier accuracy. The results experimentally verify the relationship between the multinomial distribution and multiple frame classifier.

5.3.2.3 Distance Measure Results. The distance measures described in Chapter III are evaluated on the Hidden Markov Model outputs for the left-right data of 20dB SNR and correlation level of 2. The results seen here will give insight into how the Hidden

Markov Models are making classification decisions. Hidden Markov Model outputs obtained from the leave-one-out classification method are used.

The results using the new measure proposed in this dissertation are reported first. The mean and variance of the five Hidden Markov Model output distributions are found and the symmetrized Bhattacharyya distance is calculated. The results of the distance measurements are shown in Table 14.

Table 14. Distances between the 5 Hidden Markov Model classes using the new method proposed here.

Class\Class	M60	M35	BTR60	T62	M2
M60	X	85.7	88.7	2.0	5.0
M35	85.7	X	1.3	13.3	10.7
BTR60	88.7	1.3	X	2.4	5.2
T62	2.0	13.3	2.4	X	5.0
M2	5.0	10.7	5.2	5.0	X

The results show the trend that the distance between the wheeled and tracked vehicles is much larger than the distance between vehicles of the same type. For the tracked vehicles, the M60/T62 Hidden Markov Models are closer than than either tank/M2 Hidden Markov Model pair. The M35 and BTR60 are much closer to each other than any tracked vehicle. There is a high degree of correlation between the observed classification accuracy and the Hidden Markov Model distance. Table 15 shows the classification results for the 200 training sequences for SNRs of 20 and 10dB with a correlation level of 2.

Table 15. Classification results of the 200 training sequences for each object. The SNR is 20 dB (left) and 10 dB (right).

Class\Assigned	M60	M35	BTR	T62	M2	Class\Assigned	M60	M35	BTR	T62	M2
M60	199	0	0	1	0	M60	189	0	0	11	0
M35	0	192	8	0	0	M35	0	131	59	10	0
BTR60	0	4	196	0	0	BTR60	0	72	128	0	0
T62	0	0	0	200	0	T62	29	0	0	171	0
M2	0	0	0	0	200	M2	7	0	0	101	92

It is seen here that the majority of classification errors occur between vehicles within either the *wheeled* and *tracked* group. In general, classification mistakes are more likely to occur the closer the Hidden Markov Model pairs are. An analysis of the distance between the five Hidden Markov Models using Equation 15, Levinson's measure of estimation error, was performed and the results are shown in Table 16.

Table 16. Distances between the 5 Hidden Markov Model classes using the measure of estimation error.

Class\Class	M60	M35	BTR60	T62	M2
M60	X	128.8	132.3	98.1	120.1
M35	128.8	X	48.0	97.6	125.3
BTR60	132.3	48.0	X	91.0	125.4
T62	98.1	97.6	91.0	X	124.8
M2	120.1	125.3	125.4	124.8	X

Using this measure, the difference between the tracked and wheeled vehicles was not as pronounced. The ratio of the distance of the M60/wheeled vehicles and M60/tracked vehicles is not as great with this measure. The proposed measure had a 5-to-1 ratio where this measure yields slightly better than a 1-to-1 ratio.

The distance between the five Hidden Markov Models using the measure of Juang and Rabiner (35) is shown in Table 17.

Table 17. Distances between the 5 Hidden Markov Model classes using the method of Juang and Rabiner.(35)

Class\Class	M60	M35	BTR60	T62	M2
M60	X	13.2	13.0	5.0	11.3
M35	13.2	X	1.6	9.9	11.2
BTR60	13.0	1.6	X	6.7	10.1
T62	5.0	9.9	6.7	X	10.3
M2	11.3	11.2	10.1	10.3	X

The general trend using this distance measure is similar to the one proposed here. It must be noted that the five class problem considered here is separable using the Hidden Markov Model

algorithm. When the problem is not as separable, the measure proposed here will yield a more reliable measurement of the distance between two Hidden Markov Models. Again, this is due to Juang and Rabiner's (35) measure behaving as a modified mean difference calculator while the measure proposed here uses both the mean and variance in an equation derived from the upperbound of the Bayes error rate to produce a reliable measurement.

The distance between the five Hidden Markov Models using the difference between the means is shown in Table 18.

Table 18. Distances between the 5 Hidden Markov Model classes using the difference of means.

Class\Class	M60	M35	BTR60	T62	M2
M60	X	13.1	13.0	5.0	11.3
M35	13.1	X	1.6	9.9	11.2
BTR60	13.0	1.6	X	6.7	10.1
T62	5.0	9.9	6.7	X	10.3
M2	11.3	11.2	10.1	10.3	X

These results were very similar to those obtained using Juang and Rabiner's (35) method with differences in most entries showing in the third or fourth decimal location. Note that Equation 17 and Equation 19 are identical if T_n is fixed for all n. These equations will also converge if the number of sequences is much larger than the differences in T_n. In the case examined here, $N = 200$ with 50 sequences each having a T_n value of 14, 16, 18 or 20.

The distance results using D'Orta's measure are shown in Table 19.

Table 19. Distances between the 5 Hidden Markov Model classes using the method of D'Orta.

Class\Class	M60	M35	BTR60	T62	M2
M60	X	223.7	221.5	85.6	192.4
M35	223.7	X	27.3	167.7	190.9
BTR60	221.5	27.3	X	115.2	172.4
T62	85.6	16.7	115.2	X	132.7
M2	192.4	190.9	172.4	132.7	X

These results have the same tendency as the measure of estimation error. The differences between the tracked and wheeled vehicles are not as pronounced. The difference in Table 18 and Table 19 demonstrates the effect of not normalizing each Hidden Markov Model output by the sequence length.

5.3.2.4 Right-Left Data Sequences. An examination of the directional sensitivity of the left-right Hidden Markov Model classifiers is accomplished with the right-left observation sequence type. The right-left observation sequences are constructed by reversing the frame ordering of the left-right observation sequences. The right-left data sequence begins at high azimuth angles which decreases with each frame in the sequence. Five additional Hidden Markov Models were trained for the right-left observation sequences creating a 10 class problem. The results of the leave-one-out error estimation procedure for a SNR of 20 dB and correlation level of two are shown in Table 20.

Table 20. Right-left Sequence Testing. The number of errors from each class for the two sequence sets are reported.

Sequence Type	M60	M35	BTR60	T62	M2
Left-Right	3	1	6	17	2
Right-Left	1	1	3	2	2

Of the 38 total errors (out of 2000) shown in Table 20, 11 errors are associated with an out of class object while the remaining 27 errors are misclassified by the direction of motion. The 27 sequences misclassified by direction were all moving at low elevation angle where there were between one and three distinct codewords in the entire sequence. The codeword ordering looked almost identical read forward or backward. This condition can be seen in the sphere plots of Figure 15. This result verifies that left-right Hidden Markov Models are tuned to represent the objects undergoing the type of motion in the training sequences and can be thought of as directional movement filters. This is readily understood in terms of the model architecture and the results of the vector quantization process. For any given sequence, the codebook values change from the beginning to end with some duplicate

codewords represented between adjacent time indices. Considering the left-right Hidden Markov Model architecture, codewords at the beginning of a sequence can be associated with state one while codewords at the end of a sequence can be associated with state five. Association between intermediate codewords and states cannot be explicitly made but will be distributed is some manner. Therefore, the probability of being in state one and seeing codewords toward the end of the sequence will be extremely low. Likewise, the probability of being in the last state and observing codewords at the beginning of a sequence will be low as well. This line of reasoning is why the left-right and right-left Hidden Markov Model classifiers exhibit directional selectivity. Left-right sequence trained Hidden Markov Models measure little association with the right-left observation sequences because the sequence is *out of order* from what it was trained to expect. This is equivalent to attempting to understand an individual *talking backwards*.

5.3.2.5 Vertical Data Sequences. An examination of directional sensitivity using a data set not as dramatically different as the right-left data set is accomplished with vertical data sequences. For each object, nine vertically moving image sequences spaced in 22.5° increments for azimuth angles in the range of 0° to 180° are generated. Each sequence moves vertically from elevation angle 0° to 75° in 5° steps. The vertical trajectories are shown in Figure 23.

These sequences are classified with Hidden Markov Models trained on the left-right observation sequences and are considered to be an independent test set. A classification accuracy rate of 68.8% was found across the five objects. Table 21 shows the classification results for the vertically moving sequences.

Most of the errors occurred at low and high azimuth angles. The 0° and 180° azimuth views for all objects are somewhat confusable in the low frequency Fourier magnitude space. The T62 was the only class to be correctly identified at the extreme azimuth angles. The M2 was the only object misclassified at azimuths other than 0° or 180°. The M2 was incorrectly associated with the M35 or the T62 at these non-extreme azimuth angles. The error rate for

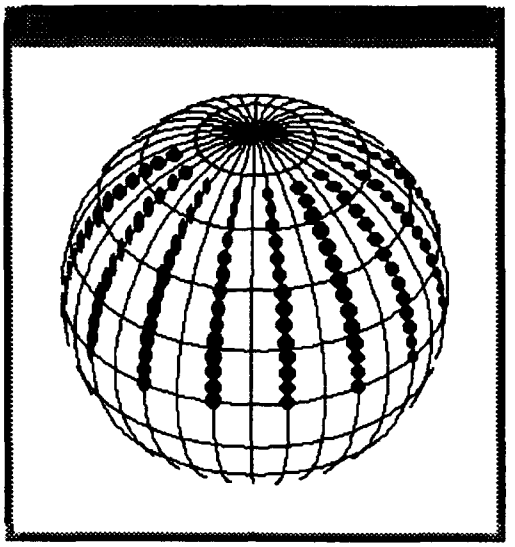


Figure 23. Vertical Sequence Trajectories.

Table 21. Test results for sequences with vertical motion. O represents a correct classification, X represents a miss.

Class\angle	0°	22.5°	45.0°	67.5°	90.0°	112.5°	135.0°	157.5°	180.0°
M60	X	O	O	O	O	O	O	O	X
M35	X	O	O	O	O	O	O	O	X
BTR60	X	O	O	O	O	O	O	O	X
T62	O	O	O	O	O	O	O	O	O
M2	X	O	X	X	X	O	O	X	X

the vertical sequences is not as severe as the rate for the right-left sequences when tested with the left-right trained Hidden Markov Model classifiers. There is enough information about vertical movement in the left-right observation sequence data set to allow reasonable recognition of the vertical data sequences.

5.3.2.6 Velocity Changes. An examination of the sensitivity of the left-right sequence trained Hidden Markov Models to changes in object velocity is made. Each velocity data set that will be described is considered to be an independent test set for classification testing purposes. Testing is performed for all SNRs at a correlation level of two. The first data set tested is where the velocity of the object is assumed to decrease by a factor of two. It

is also assumed that the sensor tracks the object for the same amount of time. Therefore, to create the test observation sequences, each codeword in each training sequence is duplicated. This process yields sequences of lengths 28, 32, 36, and 40. Each of these test sequences is evaluated against the previously trained left-right Hidden Markov Models. The accuracy of the original left-right sequences, the velocity test sequences, and the change in accuracy is shown in Table 22.

Table 22. The accuracy of the original left-right sequences, the velocity test sequences, and the change for a velocity decrease of a factor of two.

Noise (dB)	20	15	10	5	0
Original	98.7	98.4	71.1	25.6	20.6
Velocity Test	98.9	98.7	71.2	25.8	20.5
Change	+0.2	+0.3	+0.1	+0.2	-0.1

Decreasing the velocity by a factor of five implies duplicating each codeword five times, increasing the overall sequence lengths. Table 23 shows the classification results.

Table 23. The accuracy of the original left-right sequences, the velocity test sequences, and the change for a velocity decrease of a factor of five.

Noise (dB)	20	15	10	5	0
Original	98.7	98.4	71.1	25.6	20.6
Velocity Test	98.8	98.6	71.5	25.7	20.6
Change	+0.1	+0.2	+0.4	+0.1	0

Next, testing with sequences from objects with increased velocity is performed. The first case considered is for a velocity increase of a factor of two. The test sequences are created by eliminating every other codeword from the original left-right sequences, shortening each sequence by a factor of two. Table 24 shows the classification results.

Increasing the velocity by a factor of four implies keeping every fourth codeword in the original sequence. Final sequence lengths were three to five codewords long. Table 25 shows the classification results.

Table 24. The accuracy of the original left-right sequences, the velocity test sequences, and the change for a velocity increase of a factor of two.

Noise (dB)	20	15	10	5	0
Original	98.7	98.4	71.1	25.6	20.6
Velocity Test	98.4	97.9	70.7	25.5	20.3
Change	-0.3	-0.5	-0.4	-0.1	-0.3

Table 25. The accuracy of the original left-right sequences, the velocity test sequences, and the change for a velocity increase of a factor of four.

Noise (dB)	20	15	10	5	0
Original	98.7	98.4	71.1	25.6	20.6
Velocity Test	93.3	92.9	66.3	20.4	20.2
Change	-5.4	-5.5	-4.8	-5.2	-0.4

The overall results of the velocity testing on the left-right sequences again demonstrates the fact that the left-right Hidden Markov Model classifiers are searching for sequences of codewords that are arranged in the proper order. The overall $P(\mathrm{O}|\lambda)$ for each of the different velocities can be significantly different than the original length sequences. However, classification is only based on which Hidden Markov Model produces the highest probability of association.

5.3.2.7 Deceleration Changes. Testing with sequences derived from the object undergoing deceleration is also examined. Each training sequence was divided into 5 nearly equal length sections. The velocity in the first section is assumed to be the original velocity. The velocity of in the second section is decreased by a factor of two, the third section by a factor of three, the fourth section a factor of four, and the fifth section by a factor of five. This results in a stepwise deceleration from the original velocity to $1/5$ of the original velocity over the duration of the sequence. The classification results of the 20 dB SNR and correlation level of two data set is shown in Table 26.

Table 26. Deceleration testing with left-right military vehicle data set.

Noise (dB)	20	15	10	5	0
Original	98.7	98.4	71.1	25.6	20.6
Test	96.4	95.8	66.9	25.8	20.5
Change	-2.3	-2.6	-4.2	+0.2	-0.1

5.3.3 Continuous Left-Right Hidden Markov Models. The continuous left-right Hidden Markov Model is the second type of classifier examined in this dissertation. This type of model does not require the creation of a vector quantizer since it is able to process the full 28 dimensional feature vectors. Thus, the major difference with the continuous model is that the state observation probabilities are modeled with a Gaussian Mixture rather than the probability mass functions of the discrete version. The training procedure for the continuous Hidden Markov Model is modified and the reader is referred to Rabiner (55:267) for the details.

The continuous left-right Hidden Markov training and recognition steps are accomplished using Entropic's HTK–Hidden Markov Model Toolkit (23). There are many choices to be made in determining a model architecture. Among the most important of these are the number of states, the number of Gaussian mixtures used to model the observation probabilities, and the type of covariance matrix relationship to use with the Gaussian mixtures. As with the discrete Hidden Markov Model, there is no known way to optimally pick the model parameters. A trial and error search is required to find a parameter set that works well. A five state model is used to be consistent with the discrete left-right Hidden Markov Models described above. Four Gaussian Mixtures with common diagonal covariance matrices were chosen to model the state observation probabilities.

The continuous Hidden Markov Model experimentation will be performed on the left-right sequence set for all SNRs at a correlation level of two. The left-right sequence data set is prepared for training and testing by converting it to the HTK format. A master-control program written in the *C* language uses the HTK library procedures to perform a classification evaluation using the leave-one-out error method. The results are shown in Table 27.

Table 27. Results from the continuous Hidden Markov Model classifier. The left-right sequence data set for all SNRs at a correlation level of 2 is used.

Noise (dB)	20	15	10	5	0
C-HMM	100.0	100.0	63.5	36.7	21.5

Compared to the discrete Hidden Markov Model classifier, the classification accuracy is slightly higher for the 20, 15, 5, and 0 dB SNR levels. However, it is about 8 percent less for the 10 dB case. A distinct advantage or disadvantage of using the continuous Hidden Markov Model cannot be discerned from these results.

5.3.3.1 Alternate Classifier Performance. The alternate classifiers examined for the continuous sequential left-right data set are the one nearest neighbor (1-NN) and multiple frame 1-NN (M-NN) classifiers. The prototype set for the 1-NN classifier consists of the features from the individual image frames that make up the noiseless left-right data sequences. For the 200 sequences generated per class there are 3400 individual image frames. Error testing is performed with the leave-one-out method. When an individual frame is to be tested, its feature vector is deleted from the prototype set. The closest feature vector from all 5 classes is then determined using a Euclidean distance measure. If the closest feature vector has the same class label as the test vector, a correct classification is observed.

The multiple frame 1-NN classifier operates in a similar fashion to the multiple frame vector quantizer classifier described above. Each frame of a sequence is classified using the 1-NN single look classifier. A classification decision is made at the sequence level according to which class wins a plurality of the frames for a given sequence. Figure 24 illustrates the performance comparison between the continuous left-right Hidden Markov Model and the single-look and multiple frame nearest neighbor classifiers. Table 28 shows the same results in numerical form.

The results of the three classifiers are very similar for the 20, 15, and 0 dB SNRs. At 10 dB noise, the continuous Hidden Markov Model performs better than the 1-NN single look algorithm but almost 9 percent worse that the multiple frame nearest neighbor technique. This

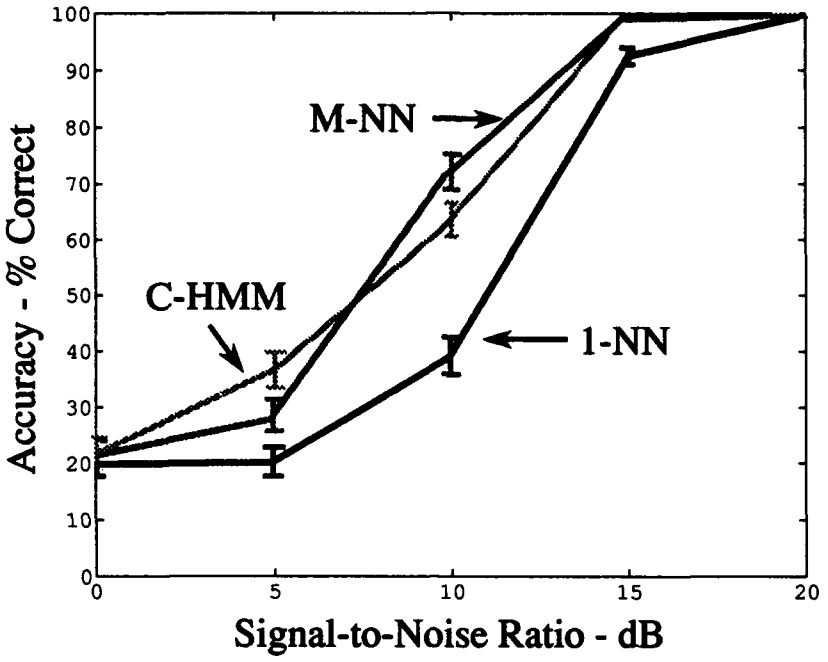


Figure 24. Performance Comparison of the continuous left-right Hidden Markov Model (C-HMM), single-look 1-NN and multiple frame (M-NN) classifiers for a correlation level of 2. 95% confidence bars are shown.

is surprising since the multiple frame technique does not directly use the temporal information in the sequence. This result may be reversed by adjusting the number of Gaussian Mixtures used to model the state observation probability creating a better model.

5.3.4 Ergodic Hidden Markov Models. The discrete ergodic Hidden Markov Model is the final type of classifier investigated on the military vehicle data set. The left-right,

Table 28. Numerical classification results for the continuous left-right Hidden Markov Model (Hidden Markov Model), single-look 1-NN and multiple frame (M-NN) classifiers for a correlation level of 2.

Noise (dB)	20	15	10	5	0
C-HMM	100.0	100.0	63.5	36.7	21.5
M-NN	100.0	100.0	72.2	28.4	21.4
1-NN	100.0	92.9	39.2	20.1	20.0

right-left, vertical, and transition-only observation sequence data sets are evaluated. The model architecture consists of 5 states with 64 observation symbols. All state transitions are possible with the ergodic Hidden Markov Model requiring a full state transition matrix. One of the major differences with the ergodic model is that it is trained with one long observation sequence rather than the multiple short sequences used for the left-right model. The composition and length of the training sequence is described next.

5.3.4.1 Left-Right Observation Sequences. The first data set classified with the ergodic models is the left-right observation sequence data set. Before classification can begin, the ergodic Hidden Markov Model for each class must be trained using a single long observation sequence of vector quantized data. The training sequence is created by generating a random trajectory over the region of interest. The azimuth and elevation coordinates of the random trajectory are created using a C language routine provided by Seibert and Waxman (59). The initial observation position is azimuth 30 degrees and elevation 20 degrees. The observation position of each additional frame is determined by generating two Gaussian random numbers controlling the azimuth and elevation stepsize. The program can handle boundary encounters by gracefully moving in a different direction. Once the routine has generated the random trajectory it is given access to the *master* data set from each class that was used to create the vector quantizer for the left-right Hidden Markov Models. This is the codebooked data generated for every 5° degrees in azimuth and elevation. For a particular class, the codeword associated with each observation position in the random trajectory is found by an interpolation routine that operates on azimuth and elevation angles from that class's *master* data set. All 5 class training sequences will follow the same random trajectory.

To determine the length of the training sequences, many sample training sequences of different lengths are generated. An ergodic Hidden Markov Model is trained for each class for each sequence length. The left-right observation sequence data for 20 dB SNR and correlation level of 2 is classified. The effect of the different length sequences on the classification accuracy is shown is Table 29.

Table 29. Effects of Training Sequence length on Ergodic Hidden Markov Model Classification Accuracy

Sequence Length	100	500	1000	1500	2000	2500	3000
Accuracy -%	85.9	96.1	96.8	97.0	97.7	98.8	99.3

Based on these results, a training sequence length of 3000 is chosen for this experimentation. An illustration of the 3000 frame random trajectory is shown in Figure 25.

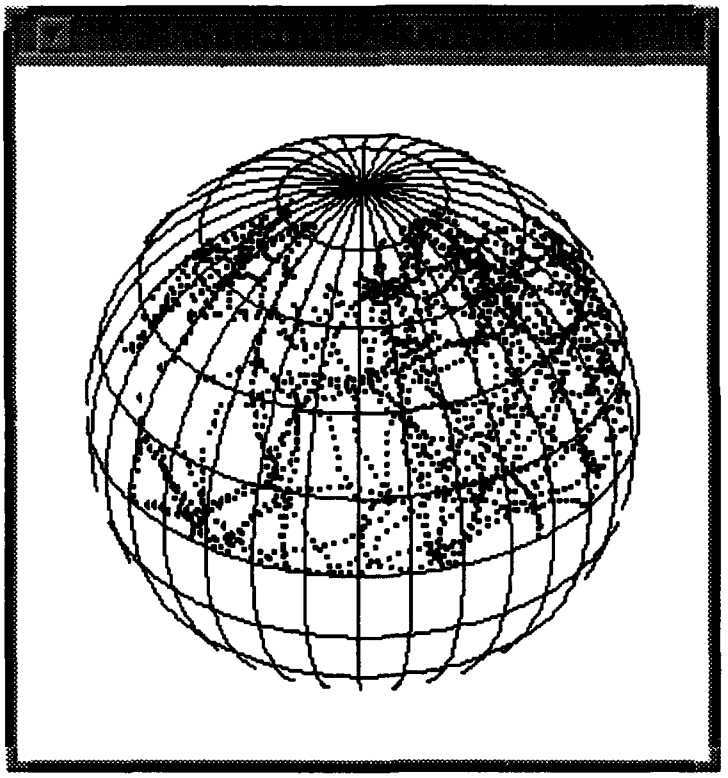


Figure 25. 3000 frame random trajectory used to train the ergodic Hidden Markov Models. The same trajectory is used for all five classes.

An ergodic Hidden Markov Model is trained for each class using the 3000 frame training sequences. The left-right observation sequence data is treated as an independent test set for this classification experiment. The classification results are shown in Table 30.

The classification accuracy of these classifiers for all noise conditions are slightly higher but consistent with the discrete left-right Hidden Markov Model results reported in Table 10.

Table 30. Classification results for the ergodic Hidden Markov Model classifiers using the left-right observation data set.

SNR\CORR	2	4	8	12
20	99.3	99.2	98.9	99.0
15	99.2	98.9	99.2	98.9
10	77.6	68.8	69.6	71.4
5	25.5	25.8	27.0	24.3
0	21.7	21.7	21.4	21.3

5.3.4.2 Right-Left Observation Sequences. The ergodic Hidden Markov Model does not encode the ordering of the sequences as strongly as the left-right model. It tends to learn the adjacency, or boundary relationships, of the clusters from the vector quantizer. This difference is explored by examining the classification accuracy of the right-left observation sequence data on the ergodic model classifiers. The classification results for the right-left observation sequences are shown in Table 31.

Table 31. Classification results for the ergodic Hidden Markov Model classifiers using the right-left observation data set.

SNR\CORR	2	4	8	12
20	99.3	99.2	99.0	98.9
15	99.1	99.0	99.2	99.0
10	77.5	69.2	69.7	71.0
5	25.5	25.8	26.9	24.3
0	21.6	21.8	21.4	20.8

The results show there is not a great deal of difference in the classification accuracies of the two data sets. The ergodic Hidden Markov Model is not as directionally sensitive as the left-right variety. One could envision a pattern recognition system that uses ergodic Hidden Markov Models to sense the presence of desired target in sensor imagery. Once the target has been detected, a bank of left-right Hidden Markov Models could track the movement of the object in a precise manner.

5.3.4.3 Vertical Sequence Testing. The sequence encoding strength of the ergodic model is further examined by determining the classification accuracy of the vertical data set. The classification results using the vertical data set are shown in Table 32.

Table 32. Test results for sequences with vertical motion using ergodic Hidden Markov Models. O represents a correct classification, X represents a miss.

Class\angle	0°	22.5°	45.0°	67.5°	90.0°	112.5°	135.0°	157.5°	180.0°
M60	X	O	O	O	O	O	O	O	X
M35	O	X	O	O	O	O	O	O	O
BTR60	X	O	O	O	O	O	O	O	X
T62	O	O	O	O	O	O	O	O	O
M2	X	O	X	X	X	O	O	O	O

A classification accuracy rate of 75.5% was found across the five objects. This is approximately an 8% increase over the 68.8% classification accuracy found using the left-right Hidden Markov Models. Again, this classification rate increase is attributed to the ergodic model not as strongly encoding the ordering of the training sequence.

5.3.4.4 Transition-only Observation Sequences. The final experiment performed on the ergodic Hidden Markov Models is a classification analysis using *transition-only* observation sequences. Transition-only observation sequence have codeword duplications for adjacent time indices deleted from the data set. Therefore, only codewords that mark transitions between aspects or characteristic views compose the sequence. This procedure is similar to the experimentation of Seibert and Waxman who only examined aspect transitions in their classification system (59).

The first experiment uses the full 3000 frame sequences to train the ergodic models and test with transition-only left-right observation sequences. The transition-only classification results are shown in Table 33. The results from testing with the full left-right data set reported Table 31 are repeated for comparison.

An experiment where the ergodic Hidden Markov Model was trained using a transition-only sequence was performed. Because of the number of codewords eliminated in creating

Table 33. The classification results from Table 31 (left) and the transition-only left-right data set classification results (right).

SNR\CORR	2	4	8	12
20	99.3	99.2	98.9	99.0
15	99.2	98.9	99.2	98.9
10	77.6	68.8	69.6	71.4
5	25.5	25.8	27.0	24.3
0	21.7	21.7	21.4	21.3

SNR\CORR	2	4	8	12
20	91.6	90.8	90.9	90.7
15	90.1	90.8	91.1	90.7
10	78.4	69.3	67.9	67.8
5	24.7	25.6	25.0	28.6
0	22.1	21.8	21.8	21.7

the transition-only sequence, a 10,000 frame training sequence is generated for each class and processed. The resulting transition-only training sequence for each class contained between 1200 and 1500 codewords. An ergodic Hidden Markov Model was trained for each class. Classification testing was performed on the full and transition-only left-right observation sequence data sets. This result is shown in Table 34

Table 34. Classification results of the transition-only trained ergodic HMMs for the full left-right data set (left) and the transition-only left-right data set (right).

SNR\CORR	2	4	8	12
20	97.7	97.9	97.8	97.8
15	97.5	97.6	97.6	98.2
10	73.0	68.4	68.5	69.2
5	25.3	25.5	26.4	23.7
0	20.8	21.0	20.9	20.8

SNR\CORR	2	4	8	12
20	97.8	97.8	97.5	97.5
15	97.4	97.3	97.1	97.4
10	72.4	65.3	66.8	71.1
5	24.4	25.3	26.0	23.6
0	21.2	21.3	20.8	20.5

The results from the last two tables indicate that transition-only sequences contain enough spatio-temporal information about the objects to produce good classification results when used for training or testing.

5.4 Conclusion

This chapter has detailed the experimentation and results for the proposed spatio-temporal pattern recognition technique based on the Hidden Markov Model. The classification performance of three types of Hidden Markov Models were investigated using a five class

problem of selected tactical military ground vehicles. The discrete and continuous left-right and discrete ergodic Hidden Markov Models performed extremely well in identifying the 3D objects in sequences of 2D imagery. A significant performance improvement was observed over the single-look and alternate multiple frame classifiers. The results demonstrate the advantage the Hidden Markov Model technique has in accessing the temporal information contained in the image sequences.

Testing with sequences where the object moved in a different manner than the training sequences showed that the left-right Hidden Markov Model is directionally sensitive and can be thought of as a directional movement filter. In contrast, the ergodic model displayed the tendency to discriminate the class of the targets while being rather insensitive to the type of motion. This is due to the ergodic model learning the adjacency of the clusters of the vector quantizer rather than the strict ordering of codewords as with the left-right models.

Finally, the new distance measure proposed here was experimentally shown to produce results superior to the other five measures discussed. Using the mean and the variance of the output distributions of the Hidden Markov Model classifier to measure the distance between model pairs is an excellent tool for judging the expected classification accuracy in a multiclass pattern recognition problem. The new measure was also shown to give insight into the nature of the classification errors by comparing the computed distances with the classification of each object's training sequences. Errors occurred most often with class pairs whose computed distance is smallest.

VI. Spatio-temporal Automatic Target Recognition System

6.1 Introduction

Accurately identifying *real world* objects using classifiers trained with *synthetic* imagery is the *Holy Grail* of pattern recognition. This chapter describes the development of such a pattern recognition system designed to analyze the performance of the Hidden Markov Model classifier identifying objects in real image sequences. The three components comprising this pattern recognition system are segmentation, feature extraction, and classification. These components are illustrated in Figure 26.

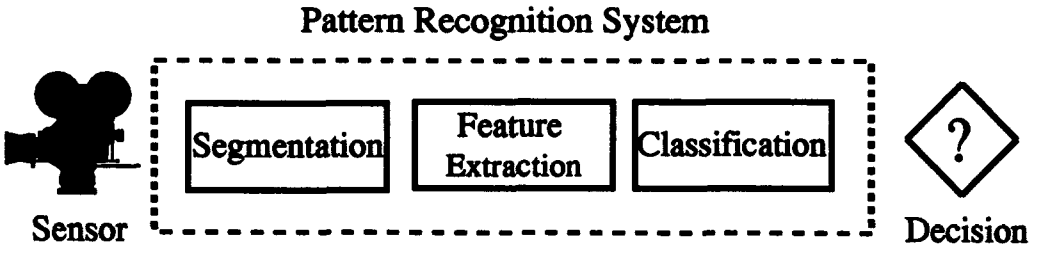


Figure 26. A typical pattern recognition system can be broken into three processes known as segmentation, feature extraction, and classification. The system uses data from a sensor to produce information to which an action/decision process is applied.

The successful classification of the real image sequences is accomplished using the discrete left-right, discrete ergodic, and continuous left-right Hidden Markov Models.

6.2 Segmentation

Segmentation is the component of the pattern recognition system that is, perhaps, the most vital and difficult to implement. The purpose of segmentation is to remove as many non-object pixels from an image as possible while leaving the object itself intact. Removing the non-object pixels, also known as background or clutter, is an essential processing step necessary for the extraction of good object features. Failure to extract good object features that are representative of the true object results in poor classification performance.

In this dissertation, the segmentation process will be applied to a sequence of real video imagery of the M60 tank and M35 truck. Features will be extracted from each image frame with the sequence classified using the three Hidden Markov Model classifiers trained on synthetically generated data. Two types of segmentation processes are investigated. These two types are manual hand segmentation and an automated technique using a neural network trained to recognize object and non-object RGB color values.

6.2.1 Hand Segmentation. The first type of segmentation investigated is hand segmentation. Video sequences of the M60 tank and M35 truck are recorded using a super VHS camcorder. The camcorder was moved in a circular motion around the objects consistent with the view centered approach previously used to generate the synthetic observation sequence training data. An IBM PC based framegrabber was used to capture sequences of 512×480 pixel images for each object. The images were reduced to 256×256 pixels. The imagery associated with each frame was transferred to a Macintosh computer for background pixel removal. Using an image processing software package, the author zeroed out the background pixels in each frame of the sequences. This process is equivalent to an *ideal* segmentation process. An example of the hand segmentation process applied to the M60 and M35 is shown in Figure 27 and Figure 28.

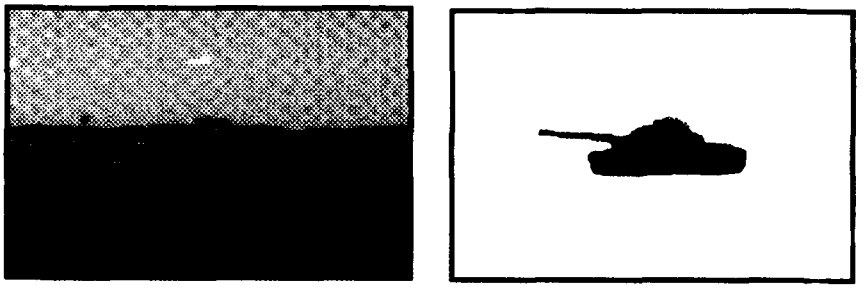


Figure 27. Real image of an M60 tank (left) and hand segmented image (right).

An 11 frame sequence of the M60 tank moving in the azimuth angle range of 80° to 180° at an elevation angle of 0° was generated. The 10 frame M35 image sequence moved between azimuth angles 0° to 50° at an elevation angle of 0°.

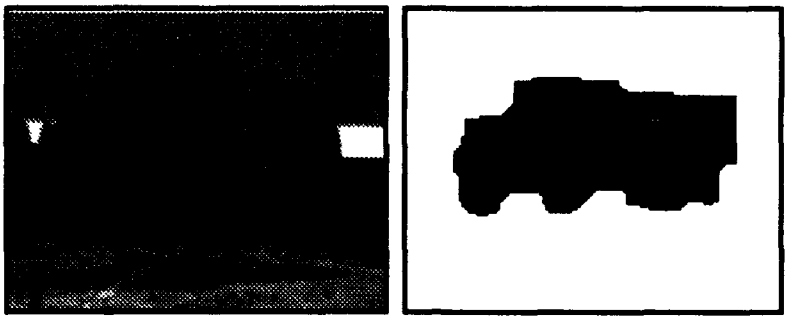


Figure 28. Real image of an M35 tank (left) and hand segmented image (right).

6.2.2 Color Segmentation. The second method of segmentation is based on a technique of identifying object and background pixels through their red, green, and blue (RGB) color values (48). To implement this technique, sample pixels from the background and object are extracted from the imagery. Labeled data is passed to a feedforward neural network for training. Once the network is trained, each individual pixel of a test image is evaluated and labeled as object or background. All background pixels are subsequently given an RGB value of (0,0,0).

The color segmentation process is tested on the 10 frame image sequence of the M35 truck. The neural network was implemented in the LNKnet software package and trained using the standard backpropagation paradigm (45). The neural network architecture consists of 3 input nodes, 25 hidden nodes, and 2 output nodes. A learning rate of 0.1 is used with no momentum. The training data set consists of 20,450 object vectors and 24,636 background vectors. The test on training data error rate is 8.0%.

The majority of object and background training vectors were those previously used to segment the M60 tank imagery (48). The M60 imagery was obtained two months prior to the M35 sequence. The video imagery for both vehicles were taken in the same general location under similar weather conditions. Example pixels from the M35 truck and some background objects not present in the M60 imagery were added to the training set. After training, each pixel in the 10 image sequence was evaluated to determine its class, object or non-object. Before viewing the results of the segmentation process, it is interesting to observe the three-space

distribution of the RGB pixel values from the object and background. Figure 29 illustrates the distribution of 200 representative feature vectors from the two classes. Figure 29 reveals the

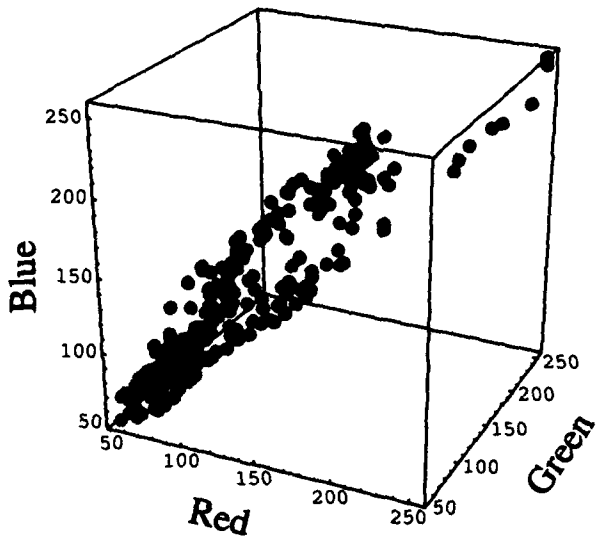


Figure 29. 3D plot of 200 RGB data vectors from the background (dark) and M35 truck (light).

situation that the two classes are non-linearly separable and thus a good candidate classification problem for the feedforward neural network. The original and segmented images of the 10 individual frames of the M35 sequence are shown in Figure 30, Figure 31, and Figure 32.

The results of the segmentation algorithm were excellent. An average of 94% of the background pixels were removed from the images while only 4.4% of the object pixels were removed. An analysis of the before and after segmentation signal-to-clutter ratio (SCR) was accomplished. The SCR is defined as

$$SCR = 20\log_{10}\left(\frac{Object_{average}}{\sigma_{clutter}}\right) \tag{27}$$

All imagery is converted from 24-bit RGB color to 256 level greyscale imagery for the SCR analysis. The average SCR of the original imagery is 2.4 dB. The average SCR of the segmented imagery is 12.2 dB. Although the color based segmentation process performed

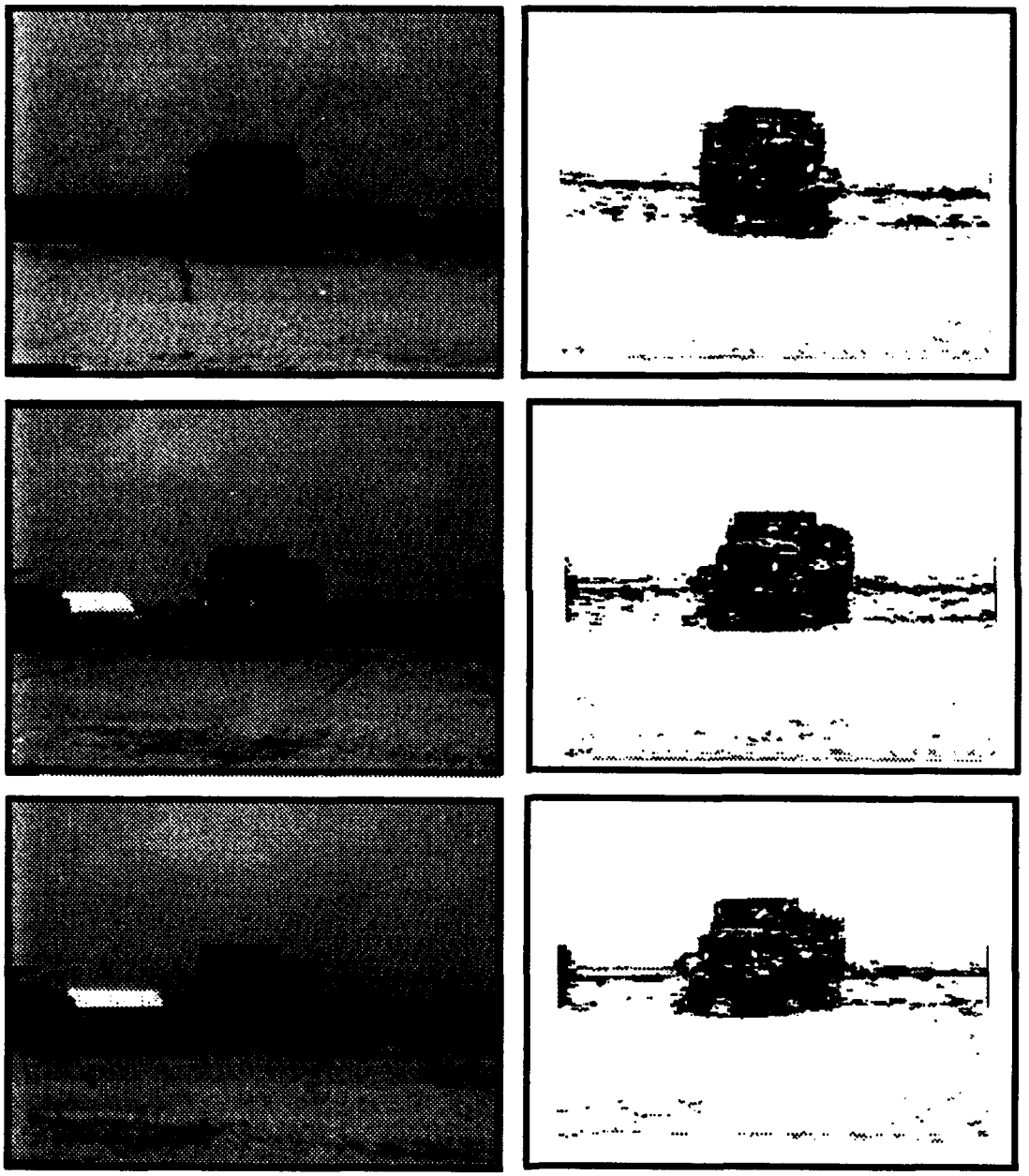


Figure 30. Original and segmented imagery for the first three frames of the M35 truck sequence.

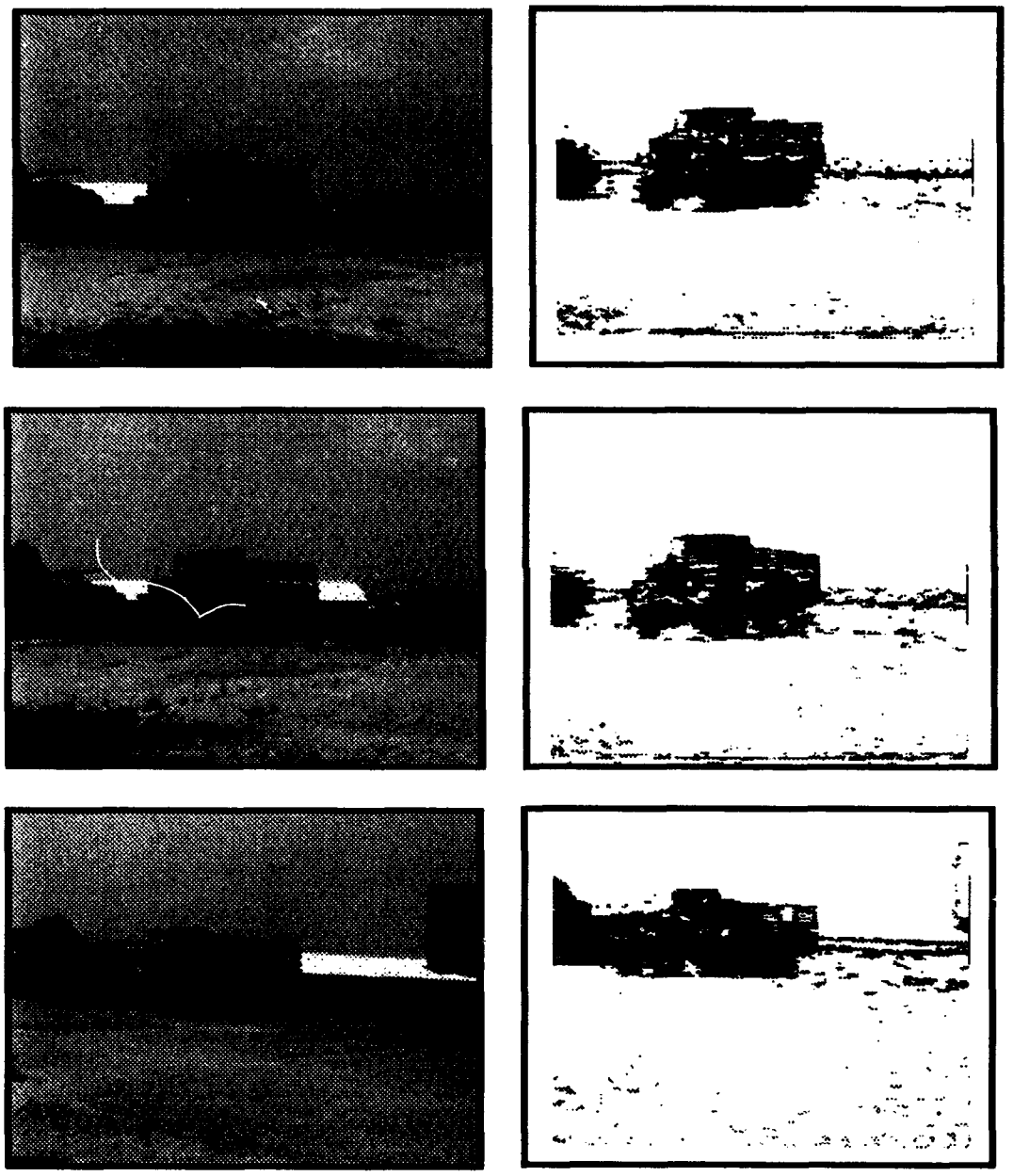


Figure 31. Frames 4, 5, and 6 of the original and segmented M35 truck sequence.

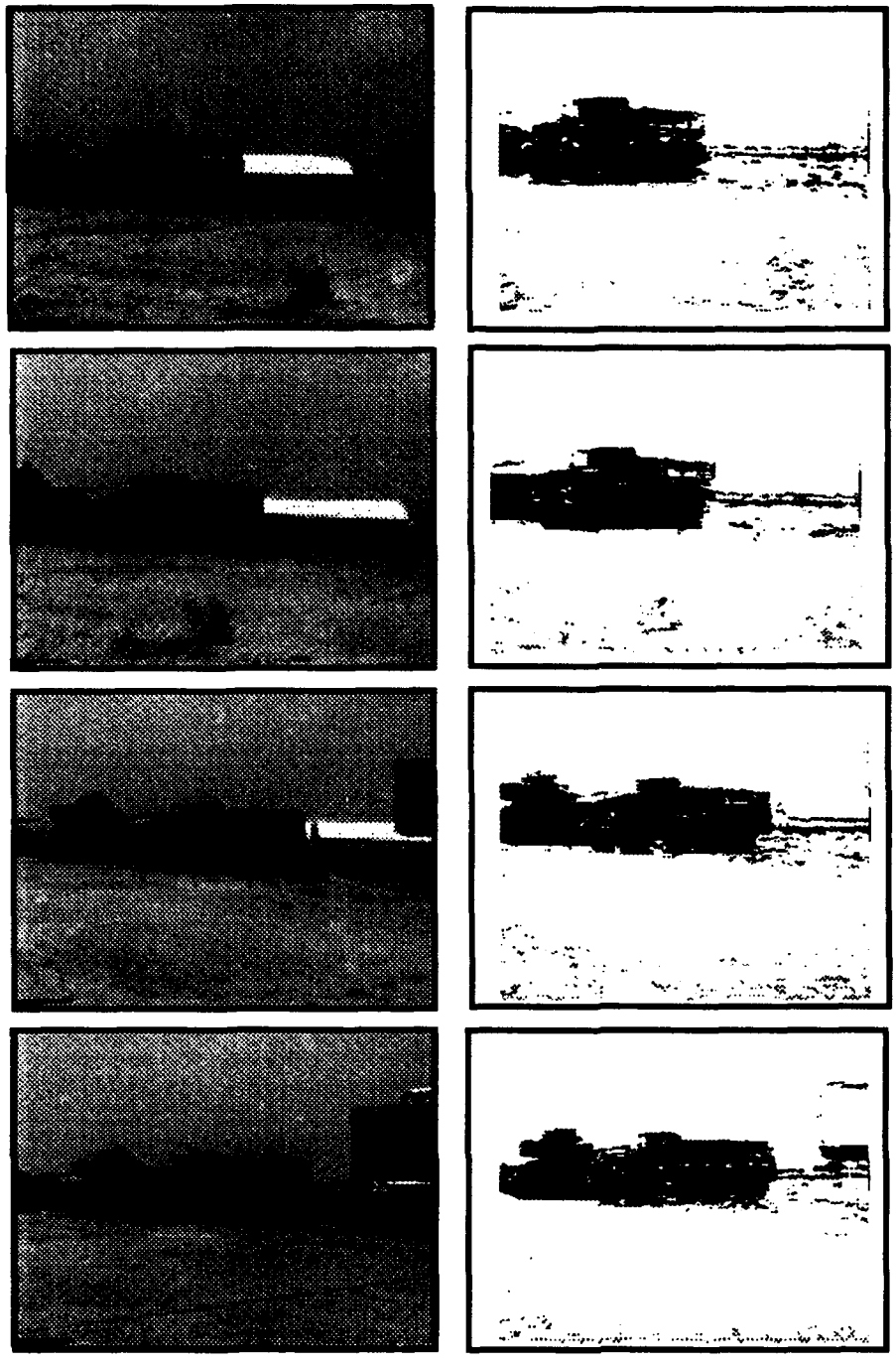


Figure 32. Frames 7, 8, 9, and 10 of the original and segmented M35 truck sequence.

well, it is clear that additional post-processing of the segmented imagery would improve the results found here.

6.3 Feature Extraction

With the segmentation process completed, each image is converted from 24-bit RGB color to a 256 level greyscale format. The 28 dimensional Fourier Magnitude features were then computed from the two hand segmented and single color segmented image sequences. Classification using the two types of discrete Hidden Markov Models requires that the multi-dimensional features be vector quantized. The three segmented sequences were codebooked using the vector quantizer constructed for the synthetically generated five class tactical military vehicle data set. The continuous sequence data was prepared for classification using the HTK Hidden Markov Model Toolkit (23).

6.4 Classification

The discrete and continuous versions of the three segmented image sequences are classified using the discrete left-right, discrete ergodic, and continuous left-right Hidden Markov classifiers. These classifiers use the same architecture as those described in Chapter V. The left-right Hidden Markov Models are trained on the left-right observation sequence data set described in Chapter V, Section 3.1. The continuous left-right Hidden Markov Models are trained with the 28 dimensional left-right observation sequence data set. The ergodic Hidden Markov Models are trained on the 3000 frame observation sequence discussed in Chapter V, Section 3.4.

The results of testing the hand segmented M60 tank sequence on the three Hidden Markov Model classifiers are shown in Table 35. The output response of the Hidden Markov Model designed for each class is given in terms of $\log\left[P(\mathbf{O}|\lambda)\right]$.

The M60 classifier displayed the strongest match for all three types of Hidden Markov Models. For the discrete left-right model, the match was approximately 140 orders of magnitude

Table 35. M60 hand segmented real image sequence classification results. The output of the Hidden Markov Model for each class is given in terms of $\log\left[P(O|\lambda)\right]$.

	M60	M35	BTR60	T62	M2
D-HMM	-4.7	-170.2	-184.7	-149.2	-168.4
C-HMM	-35.4	-94.8	-72.1	-48.0	-173.5
D-Ergodic	-3.8	-110.0	-110.0	-17.1	-110.0

stronger than the nearest out of class model. The difference was almost 13 orders of magnitude for the continuous left-right and discrete ergodic models.

The classification results using the hand segmented M35 truck sequence are shown in Table 36.

Table 36. M35 hand segmented real image sequence classification results. The output of the Hidden Markov Model for each class is given in terms of $\log\left[P(O|\lambda)\right]$.

	M60	M35	BTR60	T62	M2
D-HMM	-31.7	-20.6	-34.5	-28.3	-32.2
C-HMM	-79.7	-75.4	-89.7	-101.1	-119.1
D-Ergodic	-47.1	-9.8	-10.5	-29.8	-17.7

Here, the M35 discrete left-right classifier response was eight orders of magnitude greater that the nearest out of class object. The difference was four orders of magnitude for the continuous left-right model and approximately one order of magnitude for the discrete ergodic version.

Classification results of the M35 color segmented imagery are displayed in Table 37.

Table 37. M35 color segmented real image sequence classification results. The output of the Hidden Markov Model for each class is given in terms of $\log\left[P(O|\lambda)\right]$.

	M60	M35	BTR60	T62	M2
D-HMM	-143.2	-34.1	-37.2	-48.7	-104.6
C-HMM	-226.2	-161.3	-170.5	-182.3	-245.8
D-Ergodic	-91.9	-13.1	-15.2	-70.1	-64.5
1-NN	10	0	0	0	0

The M35 discrete left-right classifier response was three orders of magnitude greater than the BTR60. The M35 continuous left-right classifier response was nine orders of magnitude higher than the nearest competitor. The M35 ergodic response was two orders of magnitude higher than the BTR60 classifier. It is interesting to examine the last row of Table 37. The 28 dimensional feature vectors from each frame of the color segmented image sequence were classified using the 1-NN single look classifier. All 10 frames were classified by the 1-NN technique as an M60 tank. The Hidden Markov Model based classifier, however, was able to correctly, and quite strongly, identify the sequence.

6.5 Conclusion

This chapter investigated the performance of a Hidden Markov Model based pattern recognition system used to classify objects in real world image sequences. This demonstration is particularly interesting because training classifiers on synthetic data to recognize objects in real image sequences is a difficult task. The three components of the pattern recognition system (segmentation, feature extraction, and classification) were described. Image sequences were segmented using manual hand segmentation and an automated technique based on identifying object and clutter RGB color values. The image sequences were evaluated with the discrete left-right, discrete ergodic, and continuous left-right Hidden Markov Models. A strong, correct classification was obtained for each sequence with the three classifier types. The encouraging results reported here are, however, anecdotal. An examination using numerous real image sequences depicting motion over the entire region of interest is required to generalize the results found here.

VII. Recommendations and Conclusions

7.1 Recommendations

The research explored in this dissertation has opened the door to several new questions whose answers would provide valuable information for future research in the area of identifying 3D objects in 2D image sequences. It is recommended that the following research areas be explored.

- **Determining Optimum Hidden Markov Model Parameters.** Choosing Hidden Markov Model parameters such as the number of states, the number of symbols for the discrete model state observation probabilities, and the number of Gaussian Mixtures for the continuous state observation densities is more of an art than science. A trial and error approach is currently used to find model parameters that work well with a particular data set. An investigation into techniques aimed at the automated selection of the optimum model parameters for specific data sets should be undertaken.
- **Bayes Classification.** Investigating the theoretical conditions for Bayesian classification using image sequences should be accomplished. A portion of this investigation should focus on the conditions where the Hidden Markov Model based classifiers are functioning according to the Bayes decision rule. An investigation on bounding the Bayes error rate using resubstitution and leave-one-out error testing, similar to the work of Fukunaga and Hummels (27), is needed.
- **Continuous or Discrete Models.** Several researchers have expressed opinions on which type of Hidden Markov Model, discrete or continuous, has better performance. Theoretically, the continuous model can explicitly represent any state observation probability density given enough Gaussian Mixtures. A large number of Gaussian Mixtures, however, would require an enormous amount of training data for the accurate representation of the model parameters. It may be the case that a discrete model may have nearly the same performance as a continuous model with much less computational bur-

den. A tradeoff analysis on the use of the continuous and discrete model is required for optimizing the design of future Hidden Markov Model based classifiers.

- **Ergodic vs Left-right Models.** In Chapter V, it was observed that the left-right Hidden Markov Model is a directionally sensitive motion filter. That is, the left-right model could identify objects moving in a certain manner while rejecting the same object moving differently. The ergodic model, on the other hand, has the property that it can identify the objects over a wide range of motion. An investigation into the conditions where each model yields superior performance is needed. This knowledge would allow the optimization of Hidden Markov Model based classifiers where an ergodic or left-right model would be employed according to conditions.
- **Region of Interest.** In this dissertation, sample sequences over a large region of interest were used to demonstrate the superiority of the Hidden Markov Model classifier technique. Would several Hidden Markov Models *looking* at smaller regions of interest have superior classification accuracy? The knowledge of the tradeoff in the size of the region of interest versus problem scenario is required for the proper design of an aspect independent 3D object recognition system.

7.2 Conclusions

Identifying 3D objects moving in 2D image sequences is now a *solved* problem that has many interesting military and industrial applications. Current systems that identify objects in 2D imagery, generally, receive the imagery from visual or infrared sensors and perform a particular technique on a single image frame at a time or they assume independence. These systems are therefore said to perform *single look* pattern recognition. Biological studies, however, have shown that many animals, including humans, use object motion information in the identification process. The identity of objects as well as a description of their movement is discerned through an analysis of the spatial and temporal behavior of the object features extracted by the eye and brain. Therefore, this research developed a method of incorporating both spatial and temporal object information in the automatic target recognition process. It

was hypothesized that incorporating the spatial and temporal object information through an analysis of a time-indexed sequence of images will lead to a system with a substantially higher classification performance than single look methods. This goal was achieved through a new application of the technique of Hidden Markov Models to learn how features derived from moving objects change over time. A challenging demonstration of the Hidden Markov Model classifier was successfully performed on a three class moving light display problem. Another successful demonstration of this spatio-temporal procedure was performed on a five class problem of recognizing tactical military vehicles. The contributions to the state-of-the-art of pattern recognition developed in this dissertation are now described.

7.3 Contributions

- **An Information Theoretic Argument For Sequence Processing.** A new argument advocating the use of sequence, rather than single look, processing was developed. The argument is based on Shannon's definition of information and its relationship with entropy (61). A pattern recognition system can be thought of as being on the receiving end of a communication channel. From the receiver's point of view, information is equal to a reduction in the entropy, or uncertainty, of the message sent. In this application the message is the object itself viewed by the pattern recognition system. Mandating constraints on the signal (object features) reduces the associated entropy. The particular constraint applied in this dissertation is to process a sequence of messages at a single time. It was shown by proof that the entropy of a sequence of observations is less than the entropy of the individual events. It was also shown by proof that the more restrictive the dependency of the individual events, the more the entropy is reduced. In this application, the reduction of the entropy of an image sequence implies there is less uncertainty about the object's identity. To take advantage of this reduction of entropy, a sequence processing technique known a the Hidden Markov Model is employed as the pattern recognition algorithm.

- **Hidden Markov Model Distance Measure.** A new method for analyzing the distance between a pair of Hidden Markov Models was proposed. The distance measure between pairs of Hidden Markov Models gives insight into the sensitivity of the model to changes in parameters. Additionally, the distance measure is an important tool for analyzing the performance of Hidden Markov Models in a multi-class pattern recognition problem. The proposed method uses second order statistics, the mean and variance of the Hidden Markov Model output distributions, and the Bhattacharyya distance measure to find the distance between each Hidden Markov Model pair. Previously reported methods essentially only measure the distance between the means of the output distributions. Comparing the proposed method with those previously reported using a *worst case* example has demonstrated that the new method, which accesses the information in the output distribution variance, is a superior approach yielding a realistic distance measurement between pairs of Hidden Markov Models.
- **Identification of Moving Light Displays.** This dissertation reports the first known pattern recognition algorithm applied to the identification of objects from a class of imagery known as moving light displays. All previously known automated techniques attempt to uncover the type of motion the moving light display object is undergoing. Individual frames of a MLD image sequence contain very little spatial information. The information content is highly temporal in that sense that image sequences are required for object identification. Moving Light Display sequences of a cube, sphere, and pyramid were generated for experimentation. The single look classification rate for the moving light display imagery was observed to be near 50%. In contrast, the Hidden Markov Model classification rate was above 93%. The alternate nearest neighbor multiple frame technique classification rate was at least 20% below the Hidden Markov Models. A one sided t-test revealed a highly statistically significant difference between the Hidden Markov Model and multiple frame technique at a 0.01 level of significance. The ability to accurately identify this difficult class of imagery is clearly a testament to the power and robustness of the spatio-temporal technique proposed in this dissertation.

- **Use of Hidden Markov Models as a Spatio-temporal Classifier.** This dissertation presents a new application of the Hidden Markov Model technique. The Hidden Markov Model is perhaps the preeminent technique used in speech recognition. This dissertation uses the Hidden Markov Model as a spatio-temporal pattern recognition algorithm to identify 3D objects contained in 2D image sequences. Here, the Hidden Markov Model learns to recognize the temporal changes object features undergo during movement. The discrete left-right and ergodic as well as the continuous left-right Hidden Markov Models are examined as spatio-temporal sequence processors. Experimentation using a five class problem tactical military vehicles demonstrates the theoretical advantages of recognizing objects using image sequences. The Hidden Markov Model performance was substantially superior to a 1-NN single look and alternate multiple frame technique.
- **Identification of Real Imagery.** Identifying objects in real sensor imagery using classifiers trained on synthetic data is one of the most highly desired characteristics of a pattern recognition system. This characteristic, however, is seldom seen. This dissertation demonstrates such a system where real video image sequences of the M60 tank and M35 truck are successfully classified. The individual frames of the sequences were hand segmented or passed through a neural network based segmentation algorithm that identified object and background pixels based on their red, green, and blue (RGB) color values. The sequence of features obtained from the segmented imagery were correctly identified by the three types of Hidden Markov Model classifiers that were trained on the synthetic data generated from BRL-CAD.

7.4 To the Future

The research described in this dissertation represents an important evolution in the state-of-the-art of automatic pattern recognition. The information theoretic argument for sequence processing clearly demonstrates the benefits over single look techniques. The novel and successful use of the Hidden Markov Model classification technique using the moving light display and military vehicle data sets opens the door to what will be exciting and widespread

research in the processing of image sequences for object identification. The excellent results identifying objects in real image sequences using classifiers trained on synthetic data highlights the capability of the Hidden Markov Model based spatio-temporal classification technique to fulfill the object recognition tasks for the military and industry.

Appendix A. Derivation of Equations

A.1 Derivation of Forward and Backward Algorithm

The first section of Appendix A details the derivation of the Forward and Backward algorithm provided by Ruck (57). Let

$$\alpha_t(i) = Pr(O_1 \cdots O_t, i_t = q_i|\lambda)$$

That is, the probability of the observation $O_1 \cdots O_t$ and being in state $i_t = q_i$ at time t given the model λ. The following derivation for $\alpha_t(i)$ is inductive.

1) Initial condition:

$$\begin{aligned} \alpha_1(i) &= Pr(O_1, i_1 = q_i|\lambda) \\ &= Pr(O_1|i_1 = q_i, \lambda)Pr(i_1 = q_i|\lambda) \\ &= b_i(O_1)\pi_i \end{aligned}$$

which is valid for $1 \leq i \leq N$.

2) Given α_t find α_{t+1}

$$\begin{aligned} \alpha_{t+1}(j) &= Pr(O_1 \cdots O_{t+1}, i_{t+1} = q_j|\lambda) \\ &= Pr(O_{t+1}|O_1 \cdots O_t, i_{t+1} = q_j, \lambda)Pr(O_1 \cdots O_t, i_{t+1} = q_j|\lambda) \\ &= b_j(O_{t+1})Pr(O_1 \cdots O_t, i_{t+1} = q_j|\lambda) \end{aligned}$$

Now

$$
\begin{aligned}
Pr(O_1 \cdots O_t, i_{t+1} = q_j|\lambda) &= \sum_{i=1}^{N} Pr(O_1 \cdots O_t, i_{t+1} = q_j, i_t = q_i|\lambda) \\
&= \sum_{i} Pr(i_{t+1} = q_j|O_1 \cdots O_t, i_t = q_i, \lambda) Pr(O_1 \cdots O_t, i_t = q_i|\lambda) \\
&= \sum_{i} a_{ij}\alpha_t(i)
\end{aligned}
$$

Hence

$$
\alpha_{t+1}(j) = b_j(O_{t+1}) \sum_{i=1}^{N} a_{ij}\alpha_t(i)
$$

and this is valid for $1 \leq t \leq T-1$ and $1 \leq j \leq N$.

3) For $t = T$ the total probability is given as

$$
\begin{aligned}
Pr(O_1 \cdots O_T|\lambda) &= \sum_{i=1}^{N} Pr(O_1 \cdots O_T, i_T = q_i|\lambda) \\
&= \sum_{i=1}^{N} \alpha_T(i)
\end{aligned}
$$

The Forward algorithm is summarized as follows:

1. Initialization. Compute

$$
\alpha_1(i) = \pi_i b_i(O_1)
$$

for all $1 \leq i \leq N$

2. Compute successively

$$
\alpha_{t+1}(j) = b_j(O_{t+1}) \sum_{i=1}^{N} \alpha_t(i) a_{ij}
$$

for all $t = 1, 2, \ldots, T-1$ and $j = 1, 2, \ldots, N$.

3. Compute

$$Pr(O_1 \cdots O_T|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

The Backward Algorithm is derived in a similar manner. Let

$$\beta_t(i) = Pr(O_{t+1} \cdots O_T|i_t = q_i, \lambda)$$

That is, the probability of observing the partial sequence $O_{t+1} \cdots O_T$ given the current state $i_t = q_i$ and the model λ. Again the derivation proceeds inductively.

1) Define

$$\beta_T(i) = Pr(O_{T+1} \cdots O_T|i_t = q_i, \lambda) = 1$$

for $1 \leq i \leq N$.

2) Given β_{t+1} find β_t

$$\begin{aligned}
\beta_t(i) &= Pr(O_{t+1} \cdots O_T|i_t = q_i, \lambda) \\
&= \sum_{j=1}^{N} Pr(O_{t+1} \cdots O_T, i_{t+1} = q_j|i_t = q_i, \lambda) \\
&= \sum_{j} Pr(O_{t+1} \cdots O_T|i_{t+1} = q_j, i_t = q_i, \lambda) Pr(i_{t+1} = q_j|i_t = q_i, \lambda) \\
&= \sum_{j} a_{ij} Pr(O_{t+1}|O_{t+2} \cdots O_T, i_{t+1} = q_j, i_t = q_i, \lambda) Pr(O_{t+2} \cdots O_T|i_{t+1} = q_j, i_t = q_i, \lambda) \\
&= \sum_{j} a_{ij} b_j(O_{t+1}) Pr(O_{t+2} \cdots O_T|i_{t+1} = q_j, i_t = q_i, \lambda)
\end{aligned}$$

Note $O_{t+2} \cdots O_T$ is independent of $i_t = q_i$ by Markov property

$$\begin{aligned}
\beta_t(i) &= \sum_{j} a_{ij} b_j(O_{t+1}) Pr(O_{t+2} \cdots O_T|i_{t+1} = q_j, \lambda) \\
&= \sum_{j} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)
\end{aligned}$$

Hence,

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$$

for $t = T-1, T-2, \ldots, 1$ and $1 \leq i \leq N$.

3) For $t = 1$ the total probability is calculated as

$$\begin{aligned} Pr(O_1 \cdots O_T|\lambda) &= \sum_{i=1}^{N} Pr(O_1 \cdots O_T, i_1 = q_i|\lambda) \\ &= \sum_{i} Pr(O_1 \cdots O_T|i_1 = q_i, \lambda) Pr(i_1 = q_i|\lambda) \\ &= \sum_{i} \pi_i Pr(O_1|O_2 \cdots O_T, i_1 = q_i, \lambda) Pr(O_2 \cdots O_T|i_1 = q_i, \lambda) \\ &= \sum_{i=1}^{N} \pi_i b_i(O_1) \beta_1(i) \end{aligned}$$

Summary of Backward Algorithm

1. Initialization. Set

$$\beta_T(i) = 1$$

for all $1 \leq i \leq N$.

2. Compute successively

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1} \beta_{t+1}(j)$$

for $t = T-1, T-2, \ldots, 1$ and $1 \leq i \leq N$.

3. Compute

$$Pr(O_1 \cdots O_T|\lambda) = \sum_{i=1}^{N} \pi_i b_i(O_1) \beta_1(i)$$

A.2 Derivation of the Baum-Welch Re-estimation Formula

The theoretical development of the re-estimation formulas revolve around the work of Leonard E. Baum from the Institute of Defense Analysis and several colleagues. In the 1966

paper by Baum and Eagon (8), it is proved that if there exists a homogeneous polynomial, $P(x) = P(x_{ij})$, of degree d, with nonnegative coefficients and a transformation of the form

$$\mathcal{T}(x)_{ij} = \frac{x_{ij}\frac{\partial P}{\partial x_{ij}}}{\sum_j x_{ij}\frac{\partial P}{\partial x_{ij}}} \tag{28}$$

then $P(\mathcal{T}(x)) \geq P(x)$. Baum and Eagon (8:362) have shown that the Hidden Markov Model probability, $P(O|\lambda)$ - the probability that an observation sequence O is generated by the particular Hidden Markov Model λ where $\lambda = (\Pi, A, B)$, is a homogeneous polynomial with non-negative coefficients. The degree of the homogeneous polynomial is $2T + 1$ where T is the length of the observation sequences (7:3) (8:262). The fact that $P(O|\lambda)$ is a homogeneous polynomial with non-negative coefficients suggests that the individual elements of the model parameters $\lambda = (\Pi, A, B)$ can be maximized through an iterative application of the transformation shown in Equation 28. $P(O|\lambda)$ is usually written as (55:262)

$$P(O|\lambda) = \sum_{q_1,q_2\cdots,q_T} \pi_{q_1} b_{q_1}(O_1) a_{q_1q_2} b_{q_2}(O_2) \cdots a_{q_{T-1}q_T} b_{q_T}(O_T) \tag{29}$$

Equation 29 may be re-written in the form (42:1040)

$$P(O|\lambda) = \Pi^t B_1 A B_2 A \cdots A B_{T-1} A B_T \tag{30}$$

or equivalently as (42:1069)

$$P(O|\lambda) = 1^t B_T A^t B_{T-1} A^t \cdots B_2 A^t B_1 \Pi \tag{31}$$

where the superscript t implies transposition. Equation 30 is valid since we consider Π and 1 to be a column vectors and A and B to be arrays where

$$\begin{aligned} \Pi &= (\pi_i \ni 1 \leq i \leq N) \\ A &= (a_{ij} \ni 1 \leq i, j \leq N) \end{aligned}$$

$$B = (b_j(k) \ni 1 \leq j \leq N \text{ and } 1 \leq k \leq M)$$

$$1 = \text{unity column vector of length N}$$

The A array is formed in the usual way, however, the B array has the form

$$B_t = \begin{pmatrix} b_1(O_t) & & O \\ & b_2(O_t) & \\ & & \ddots \\ O & \cdots & b_N(O_t) \end{pmatrix}$$

The dimensionality of $P(O|\lambda)$ in Equation 30 is verified by noticing

$$(1xN)(NxN)(NxN)\cdots(NxN)(NxN)(Nx1) = 1$$

Levinson writes the forward and backward variables in matrix form as (42:1069)

$$\alpha_{t+1} = B_{t+1}A^t\alpha_t = B_{t+1}A\alpha_t' \qquad t = 1, 2, \ldots T-1$$

$$\beta_t = AB_{t+1}\beta_{t+1} \qquad t = T-1, T-2, \ldots 1$$

where α_t and β_t are represented by column vectors that follow

$$\alpha_t = (\alpha_t(i) \ni 1 \leq i \leq N)$$

$$\beta_t = (\beta_t(i) \ni 1 \leq i \leq N)$$

Notice that the second portion of Equation 30 can now be rewritten as

$$P(O|\lambda) = \beta_t'\alpha_t = \alpha_t'\beta_t \quad \text{for any } t \text{ in } (1, T)$$

where the superscript again implies transposition.

To derive the re-estimation formulas for the Hidden Markov Model, define the gradient with respect to each of the three independent model parameters, (Π, A, B), as

$$\begin{aligned}
\nabla_{\Pi} P(O|\lambda) &= \frac{\partial P(O|\lambda)}{\partial \pi_i} \\
\nabla_{A} P(O|\lambda) &= \frac{\partial P(O|\lambda)}{\partial a_{ij}} \\
\nabla_{B} P(O|\lambda) &= \frac{\partial P(O|\lambda)}{\partial b_j(k)}
\end{aligned}$$

Now calculate each of the gradients above and show the re-estimation formulas.

A.2.1 Re-estimate of A. Looking at Equation 30 and evaluating the gradient for the parameter A, noting that Π and B are independent of A, we get the following $T-1$ terms

$$\begin{aligned}
\nabla_{A} P(O|\lambda) &= \left(\Pi' B_1 \frac{\partial A}{\partial A} B_2 A \cdots B_{T-1} A B_T 1\right) + \left(\Pi_t B_1 A B_2 \frac{\partial A}{\partial A} \cdots B_{T-1} A B_T 1\right) \\
&+ \cdots + \left(\Pi' B_1 A B_2 \cdots B_{T-1} \frac{\partial A}{\partial A} B_T 1\right) \qquad (32)
\end{aligned}$$

As an example, the term of Equation 32 for $t = 4$ is

$$\left(\Pi' B_1 A B_2 A B_3 A B_4 \frac{\partial A}{\partial A} B_5 A B_6 \cdots A B_{T-1} A B_T 1\right) \qquad (33)$$

and can be grouped in the following way

$$\underbrace{\Pi' B_1 A B_2 A B_3 A B_4}_{\alpha_t} \frac{\partial A}{\partial A} B_5 \underbrace{A B_6 \cdots A B_{T-1} A B_T 1}_{\beta_{t+1}}$$

Using the definitions in Equation 32 we get

$$\begin{aligned}
\alpha_t &= \alpha_4 = B_4 \alpha_3^t A \\
\beta_{t+1} &= \beta_5 = A B_6 \beta_6
\end{aligned}$$

also

$$\frac{\partial A}{\partial A} B_5 = B_5$$

Therefore, Equation 33 can be written as

$$\alpha_t \beta'_{t+1} B_{t+1}$$

Adding up all $T - 1$ terms, the total probability gradient for A is

$$\nabla_A P(O|\lambda) = \sum_{t=1}^{T-1} \alpha_t \beta'_{t+1} B_{t+1}$$

The derivative of a specific element of the A matrix is now

$$\frac{\partial P}{\partial a_{ij}} = \sum_{t=1}^{T-1} \alpha_t(i) \beta_{t+1}(j) b_j(O_{t+1})$$

The re-estimation formula for a specific element of the A matrix following the transformation in Equation 28 is

$$\bar{a}_{ij} = \frac{a_{ij} \frac{\partial P}{\partial a_{ij}}}{\sum_j a_{ij} \frac{\partial P}{\partial a_{ij}}}$$

which can now be written as

$$\bar{a}_{ij} = \frac{a_{ij} \sum_{t=1}^{T-1} \alpha_t(i) \beta_{t+1}(j) b_j(O_{t+1}}{\sum_{j=1}^{N} a_{ij} \sum_{t=1}^{T-1} \alpha_t(i) \beta_{t+1}(j) b_j(O_{t+1}}$$

or in the more compact form

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \alpha_t(i) a_{ij} \beta_{t+1}(j) b_j(O_{t+1}}{\sum_{t=1}^{T-1} \alpha_t(i) \beta_t(i)}$$

A.2.2 Re-estimate of Π. For the parameter Π, use $P(\mathrm{O}|\lambda)$ in Equation 31. There is only one term in the gradient, namely

$$\nabla_{\Pi} P(0|\lambda) = \frac{\partial \Pi}{\partial \Pi} \left(1 B_T A' B_{T-1} A' \cdots B_2 A' B_1\right) \;=\; B_1 \beta_1$$

The derivative for a specific π_i can be written as

$$\frac{\partial P}{\partial \pi_i} = b_i(0_1)\beta_1(i) \;=\; \sum_{j=1}^{N} b_i(0_1) a_{ij} b_j(0_2) \beta_2(j)$$

The re-estimation formula for a specific element of the Π vector following the transformation in Equation 28 is

$$\bar{\pi}_i = \frac{\pi_i \frac{\partial P}{\partial \pi_i}}{\sum_j \pi_j \frac{\partial P}{\partial \pi_j}}$$

can now be written as

$$\bar{\pi}_i = \frac{\pi_i \sum_{j=1}^{N} b_i(0_1) a_{ij} b_j(0_2) \beta_2(j)}{\sum_{j=1}^{N} \pi_j \sum_{j=1}^{N} b_i(0_1) a_{ij} b_j(0_2) \beta_2(j)}$$

A.2.3 Re-estimate of B. The re-estimation formula for the B matrix is found by applying the gradient to Equation 30, yielding T terms of the form

$$\begin{aligned}
(\nabla_B P(O|\lambda))_{(jk)} \;&=\; \left(\Pi' \frac{\partial B_1}{\partial B} A B_{T-1} A B_2 A \cdots B_{T-1} A B_T 1\right) \\
&+\; \left(\Pi' B_1 A \frac{\partial B_2}{\partial B} \cdots B_3 A \cdots B_{T-1} A B_T 1\right) \\
&+\; \cdots + \left(\Pi' B_1 A B_2 \cdots B_{T-1} A \frac{\partial B_T}{\partial B} 1\right) \qquad (34)
\end{aligned}$$

As an example, the term for $t = 4$ in Equation 34 is

$$\left(\Pi' B_1 A B_2 A B_3 A \frac{\partial B_4}{\partial B} A B_5 \cdots B_{T-1} A B_T 1\right)$$

and can be grouped in the following way

$$\underbrace{\Pi^t B_1 A B_2 A B_3}_{\alpha_{t-1}} A \frac{\partial B_4}{\partial B} \underbrace{A B_5 \cdots B_{T-1} A B_T 1}_{\beta_t} \tag{35}$$

Equation 35 can now be written compactly as

$$\left(\alpha_{t-1}^t A\right)_j (\beta_t)_j = \left(A^t \alpha_{t-1}\right)_j (\beta_t)_j$$

Summing the T terms gives

$$\nabla_B P(O|\lambda) = \sum_{t=1, O_t=k}^{T} \left(A^t \alpha_{t-1}\right)_j (\beta_t)_j$$

For a specific j

$$\left(A^t \alpha_{t-1}\right)_j = \sum_{i=1}^{N} a_{ij} \alpha_{t-1}(i)$$

Therefore, the derivative for a particular element of the B matrix is

$$\frac{\partial P}{\partial b_j(k)} = \sum_{t=1, O_t=v_k}^{T} \left[\sum_{i=1}^{N} a_{ij} \alpha_{t-1}(i)\right] \beta_t(j)$$

The re-estimation formula for a specific element of the B matrix following the transformation in Equation 28 is

$$\bar{b}_j(k) = \frac{b_j(k) \frac{\partial P}{\partial b_j(k)}}{\sum_l b_j(l) \frac{\partial P}{\partial b_j(l)}}$$

and can now be written as

$$\bar{b}_j(k) = \frac{b_j(k) \sum_{t-1, O_t=v_k}^{T} \left[\sum_{i=1}^{N} a_{ij} \alpha_{t-1}(i)\right] \beta_t(j)}{\sum_{l=1}^{M} b_j(l) \sum_{t=1, O_t=v_l}^{T} \left[\sum_{i=1}^{N} a_{ij} \alpha_{t-1}(i)\right] \beta_t(j)}$$

or in the more compact form

$$\bar{b}_j(k) = \frac{\sum_{t-1, O_t=v_k}^{T} \alpha_t(j)\beta_t(j)}{\sum_{t=1}^{T-1} \alpha_t(j)\beta_t(j)}$$

A.3 Logarithmic Form of the Baum-Welch Re-estimation Formula

The forward and backward variables are written as

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i)a_{ij}\right] b_j(O_{t+1})$$

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij}b_j(O_{t+1})\beta_{t+1}(j)$$

The Baum-Welch re-estimation formula for elements of the A and B matrices are

$$a_{ij} = \frac{\sum_{i=1}^{T-1} \alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\sum_{t=1}^{T-1} \alpha_t(i)\beta_t(i)}$$

$$b_j(l) = \frac{\sum_{t=1, O_t=l}^{T} \alpha_t(j)\beta_t(j)}{\sum_{t=1}^{T} \alpha_t(j)\beta_t(j)}$$

The Logarithmic version of the forward and backward variables are

$$\log_{10}(\alpha_{t+1}(j)) = \log_{10}\left[\sum_{i=1}^{N} \alpha_t(i)a_{ij}\right] + \log_{10}(b_j(O_{t+1}))$$

$$\log_{10}(\beta_t(i)) = \log_{10}\left[\sum_{j=1}^{N} a_{ij}b_j(O_{t+1})\beta_{t+1}(j)\right]$$

Likewise, the logarithmic version of the Baum-Welch re-estimation formula become

$$\log_{10}(a_{ij}) = \log_{10}\left[\sum_{t=1}^{T-1} \alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)\right] - \log_{10}\left[\sum_{t=1}^{T-1} \alpha_t(i)\beta_t(i)\right]$$

$$\log_{10}(b_j(l)) = \log_{10}\left[\sum_{t=1, O_t=l}^{T} \alpha_t(j)\beta_t(j)\right] - \log_{10}\left[\sum_{t=1}^{T} \alpha_t(j)\beta_t(j)\right]$$

The re-estimation formula for the left-right Hidden Markov Model, trained with multiple observation sequences, are given as

$$\log_{10}(\alpha_{t+1}^k(j)) = \log_{10}\left[\sum_{i=1}^{N} \alpha_t^k(i)a_{ij}\right] + \log_{10}(b_j(O_{t+1}^k))$$

$$\log_{10}(\beta_t^k(i)) = \log_{10}\left[\sum_{j=1}^{N} a_{ij}b_j(O_{t+1}^k)\beta_{t+1}^k(j)\right]$$

and

$$\begin{aligned}\log_{10}(a_{ij}) &= \log_{10}\left[\sum_{k=1}^{K}\frac{1}{P_k}\sum_{t=1}^{T-1}\alpha_t^k(i)a_{ij}b_j(O_{t+1}^k)\beta_{t+1}^k(j)\right] \\ &- \log_{10}\left[\sum_{k=1}^{k}\frac{1}{P_k}\sum_{t=1}^{T-1}\alpha_t^k(i)\beta_t^k(i)\right]\end{aligned}$$

$$\begin{aligned}\log_{10}(b_j(l)) &= \log_{10}\left[\sum_{k=1}^{K}\frac{1}{P_k}\sum_{t=1, O_t=l}^{T}\alpha_t^k(j)\beta_t^k(j)\right] \\ &- \log_{10}\left[\sum_{k=1}^{K}\frac{1}{F_k}\sum_{t=1}^{T}\alpha_t^k(j)\beta_t^k(j)\right]\end{aligned}$$

Bibliography

1. Norman Abramson. *Information and Coding Theory.* McGraw-Hill, New York, 1963.
2. C. Anderson and D. VanEssen. Processing of visual information in primate brains. In *NASA Technical Brief vol 15 n 3*, 1991.
3. R. Bakis. Continuous speech word recognition via centi-second acoustic states. Technical Report RC-4788, IBM Thomas J. Watson Research Center, April 1974.
4. Dana H. Ballard and Christopher M. Brown. *Computer Vision.* Prentice-Hall, New Jersey, 1982.
5. H. B. Barlow and W. R. Levick. The mechanism of directionally selective units in rabbit's retina. *Journal of Physiology (London)*, 178:477-504, 1965.
6. L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical snalysis of probabilistic functions of markov chains. *Annals of Mathematical Statistics*, 41(1):164-171, 1970.
7. Leonard E. Baum. An inequality and associated maximization technique is statistical estimation for probabilistic functions of markov process. *Inequalities*, 3:1-8, 1972.
8. Leonard E. Baum and J. A. Egon. An inequality with applications to statistical estimation for probabilistic function of a markov process and to a model for ecology. *Bulletin of the Americal Meteorological Society*, 73:360-363, 1967.
9. Leonard E. Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state markov chains. *Annals of Mathematical Statistics*, 37:1554-1563, 1966.
10. Leonard E. Baum and George R. Sell. Growth transformations for functions on manifolds. *Pacific Journal of Mathematics*, 27(2):211-227, 1968.
11. Richard E. Blahut. *Principles and Practice of Information Theory.* Addison-Wesley, Reading MA, 1987.
12. A. J. Bulpitt and N. M. Allinson. Motion perception and recognition using moving light displays. In *Second International Conference on Artifical Neural Networks*, pages 91-94, November 1991.
13. J. Callahan and R. Weiss. A model for describing surface shape. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 240-245, 1985.
14. P.A. Chou. Recognition of equations using a two-dimensional stochastic context-free grammar. In *Proceedings of the SPIE Vol. 1199: Visual Communications and Image Processing IV*, pages 852-863, 1989.
15. Charles Cole. Shannon revisited: Informaiton in terms of uncertainty. *Journal of the American Society for Information Science*, 44(4):204-211, May 1993.
16. Thomas M. Cover. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Transactions on Electronic Computers*, EC-14(14):326-334, June 1965.

17. J. E. Cutting and L. T. Kozlowski. Recognizing friends by their walk: Gait perception without familiarity clues. *Bulletin of the Psychonometric Society*, 9(5):353–356, 1977.

18. Pierre A. Devijver and Josef Kittler. *Pattern Recognition: A Statistical Approach.* Prentice-Hall, New Jersey, 19.

19. Mark R. Dewitt. High range resulution radar target identification using the prony model and hidden markov models. Master's thesis, Air Force Institute of Technology, Wright-Patterson AFB, OH 45433, December 1992.

20. P. D'Orta, M. Ferretti, and S. Scarci. Phoneme classification for real time speech recognition of italian. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 81–84, 1987.

21. Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis.* John Wiley and Sons, New York, 1973.

22. Bradley Efron. *The jackknife, the bootstrap, and other resampling plans.* Society for Industrial and Applied Mathematics, Philadelphia, 1982.

23. Entropic Research Laboratory, Inc., Washington,DC. *HTK - Hidden Markov Model Toolkit*, 1st edition, 1993.

24. Yariv Ephraim and Lawrence E. Rabiner. On the relations between modeling approaches for speech recognition. *IEEE Transactions on Information Theory*, 36(2):372–380, March 1990.

25. Donald H. Foley. Considerations of sample and feature size. *IEEE Trans. on Information Theory*, IT-18:618–626, September 1972.

26. Keinosuke Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press, New York, second edition, 1990.

27. Keinosuke Fukunaga and Donald M. Hummels. Bayes error estimation using Parzen and k-nn procedures. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, PAMI-9(5):634–643, September 1987.

28. Keinosuke Fukunaga and Donald M. Hummels. Leave-one-out procedures for non-parametric error estimates. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, PAMI-11(9):421–423, April 1989.

29. Nigel H. Goddard. The interpretation of visual motion: Recognizing moving light displays. In *Proceedings of the Workshop on Visual Motion*, pages 212–220, Washington D. C., 1988.

30. Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing.* Addison-Wesley, Reading, Mass, 2nd edition, 1987.

31. M. Gray. Recognition planning from solid models. In *Proceedings of the Alvey Computer Vision and Image Processing Meeting, Bristol*, pages 41–43, September 1986.

32. Richard W. Hamming. *Coding and Information Theory.* Prentice-Hall, New Jersey, 2nd edition, 1980.

33. W. H. Highleyman. The design and analysis of pattern recognition experiments. *Bell Systems Technical Journal*, 41:723–744, March 1962.

34. Gunnar Johansson. Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, 14(2):201–211, 1973.

35. B. H. Juang and L. R. Rabiner. A probabilistic distance measure for hidden markov models. *AT&T Technical Journal*, 64(2):391–408, February 1985.

36. J. J. Koenderink and A. J. van Doorn. The internal representation of solid shape with respect to vision. *Biological Cybernetics*, 32:211–216, 1979.

37. L. T. Kozlowski and J. E. Cutting. Recognizing the sex of a walker from a dynamic point-light display. *Perception & Psychophysics*, 21(6):575–580, 1977.

38. Peter A. Lachenbruch. An almost unbiased method of obtaining confidence intervals for the probability of misclassification in discriminant analysis. *Biometrics*, 23:639–645, December 1967.

39. Peter A. Lachenbruch and M. Ray Mickey. Estimation of error rates in discriminant analysis. *Technometrics*, 10:1–11, 1967.

40. Francois Le Chevalier, Gerard Bobillot, and Cecile Fugier-Garrel. Radar target and aspect angle identification. In *Proceedings of the IEEE 1978 International Conference on Pattern Recognition*, pages 398–400, 1978.

41. Mun K. Leung and Thomas S. Huang. An integrated approach to 3-d motion analysis and object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10):1075–1084, 1991.

42. S. E. Levinson, L. R. Rabiner, and M. M. Sondhi. An introduction to the applications of the theory of probabilistic functions of a markov process to automatic speech recogniton. *The Bell System Technical Journal*, 62(4):1035–1074, 1983.

43. Edumnd W. Libby. *Application of Sequence Comparison Methods to Multisensor Data Fusion and Target Recognition.* Ph.D. Dissertation, Air Force Institute of Technology, Wright-Patterson AFB, OH 45433, July 1993.

44. Yoseph Linde, Andres Buzo, and Robert M. Gray. An algorithm for vector quantizer design. *IEEE Transactions on Communications*, COM-28(1):84–94, 1986.

45. Richard P. Lippmann, Linda Kukolich, and Elliot Singer. Lnknet: Neural network machine-learning and statistical software for pattern classifiction. *The Lincoln Laboratory Journal*, 6(2):249–267, 1993.

46. Cheng-Hsiung Liu and Wen-Hsiang Tsai. 3d curved object recognition from multiple 2d camera views. *Computer Vision, Graphics, and Image Processing*, 50:177–187, 1990.

47. A. J. Lunts and V. L. Brailovski. Evaluation of attributes obtained in statistical decision rule. *Engineering Cybernetics (USSR)*, 3:98–109, 1967.

48. K.A. McCrae, D.W. Ruck, S.K. Rogers, and M.E. Oxley. Color image segmentation. In *Applications of Artificial Neural Networks V, Proc. SPIE 2243*, April 1994.

49. Hans-Hellmut Nagel. On the estimation of optical flow: Relations between different approaches and some new results. *Artificial Intelligence*, 33:299–324, 1987.

50. D. I. Perrett, M. H. Harries, R. Bevan, S. Thomas, P. J. Benson, A. J. Mistlin, A. J. Chitty, J.K. Hietanen, and J. E. Ortega. Frameworks of analysis for the neural representations of animate objects and actions. *Journal of Experimental Biology*, 146:87–113, 1989.

51. David I. Perrett, A. J. Mistlin, and A. J. Chitty. Visual neurones responsive to faces. *TINS*, 10(9):358–364, 1987.

52. Alan B. Poritz. Hidden markov models: A guided tour. In *Proceedings of the ICASSP*, pages 7–13, 1988.

53. L. R. Rabiner and B. H. Juang. An introduction to hidden markov models. *IEEE ASSP Magazine*, pages 4–16, 1986.

54. L. R. Rabiner, B. H. Juang, S. E. Levinson, and M.M. Sondhi. Recognition of isolated digits using hidden markov models with continuous mixture densities. *AT&T Technical Journal*, 64(6):1211–1222, December 1986.

55. Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

56. Richard F. Rashid. Towards a system for the interpretation of moving light displays. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-2(6):574–581, 1980.

57. Dennis W. Ruck. Class lecture notes, eeng 621. Air Force Institute of Technology, 1993.

58. H. M. Sakai and K-I Naka. *Neuron Network in Catfish Retina:1968-1987 in Progress in Retinal Research vol 7.* Pergamon Press, N.N. Osborn and G.J. Chander eds., Oxford, 1987.

59. Michael Seibert and Allen M. Waxman. Adaptive 3-d object recognition from multiple views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):107–124, 1992.

60. K. Sam Shanmugan and A. M. Breipohl. *Random Signals: Detection, Estimation, and Data Analysis.* John Wiley & Sons, New York, 1988.

61. Claude E. Shannon. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379–423,623–656, 1948.

62. Claude E. Shannon and Warren Weaver. *The Mathematical Theory of Communication.* University of Illinois Press, Urbana, 1964.

63. Murray R. Spiegel. *Schaum's Outline of Theory and Problems of Statistics.* McGraw-Hill, New York, 1961.

64. Oliver Tallman and Matthew Kabrisky. A model for the classifiction of visual images. *International Journal of Biomedical Computation*, 1(1):1035–1039, 1969.

65. Julius T. Tou and Rafael C. Gonzalez. *Pattern Recognition Principles.* Addison-Wesley Pubishing Company, 1st edition, 1974.

66. Godfried T. Toussaint. Bibliography on estimation of misclassification. *IEEE Transactions on Information Theory*, IT-20(4):472–479, July 1974.

67. U.S. Army Ballistic Research Laboratory, Aberdeen Proving Ground, Maryland 21005-5066. *BRL-CAD*, 4.0 edition, December 1991.

68. Y. F. Wang, M. J. Magee, and J. K. Aggarwal. Matching three-dimensional objects using sillouetts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(4):513–518, 1984.

69. Andrew B. Watson and Albert J. Ahumada. Model of human visual-motion sensing. *Journal of the Optical Society of America A*, 2(2):322–341, 1985.

70. Arthur R. Weeks, Harley R. Myler, and Holly G. Wenaas. Computer-generated noise images for the evaluation of image processing algorithms. *Optical Engineering*, 32(5):982–992, May 1993.

71. Sholom M. Weiss. Small sample error rate estimataion for k-nn classifiers. *IEEE Transactions of Pattern Analysis and Machine Intelligen*, 13(3):285–289, March 1991.

72. G. Barrie Wetherill. *Sequential Methods in Statistics*. John Wiley & Sons, New York, 1966.

Vita

Captain Kenneth H. Fielding was born on 5 January, 1961 in Columbia, Missouri. He graduated from Jacksonville High School in Jacksonville, Arkansas in 1979. He attended the University of Central Arkansas where he received a B.S. in Physics in 1983. He then entered the U.S. Air Force where he first attended the University of New Mexico, earning a B.S.E.E. degree in 1985. He was assigned to the Tactical Air Warfare Center for two years where he served as an electronic warfare engineer. In 1988 he received an M.S.E.E. degree specializing in electro-optics from the Air Force Institute of Technology (AFIT). His next assignment was to Rome Laboratory where, as chief electro-optics engineer, he performed research in optical automatic target recognition systems. Capt Fielding returned to AFIT in 1990 to pursue his Ph.D. His doctoral research focuses on the identification of 3-D objects using sequences of 2-D imagery. He is a member of SPIE and IEEE.

Permanent address: 301 Red Bud Lane
WPAFB, OH 45433

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	June 1994	*Doctoral Dissertation*

4. TITLE AND SUBTITLE
Spatio-temporal Pattern Recognition Using Hidden Markov Models

5. FUNDING NUMBERS

6. AUTHOR(S)
Kenneth H. Fielding, Capt, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology, WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/DS/ENG/94J-02

9. SPONSORING MONITORING AGENCY NAME(S) AND ADDRESS(ES)
WL/AARA (Greg Power)
2010 5th St.
WPAFB, OH 45433-7001

10. SPONSORING MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for Public Release; Distribution Unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

A new spatio-temporal method for identifying 3D objects found in 2D image sequences is presented. The Hidden Markov Model technique is used as a spatio-temporal classification algorithm to identify 3D objects by the temporal changes in observed shape features. A new information theoretic argument is developed that proves identifying objects based on image sequences can lead to higher classification accuracies than single look methods. A new distance measure is proposed that analyzes the performance of Hidden Markov Models in a multi-class pattern recognition problem. A three class problem identifying moving light display objects provides experimental verification of the sequence processing argument. Individual frames of a MLD image sequence contain very little spatial information. The single look classification rate for the moving light display imagery was observed to be near 50%. In contrast, the Hidden Markov Model classification rate was above 93%. The alternate nearest neighbor multiple frame technique classification rate was 20% below the Hidden Markov Models. A one sided *t*-test revealed a highly statistically significant difference between the Hidden Markov Model and multiple frame technique at a 0.01 level of significance. A five class problem consisting of tactical military ground vehicles is considered to provide verification using imagery with both spatial and temporal information. Results confirmed the new spatio-temporal pattern recognition method produces superior results by accessing the temporal information in the image sequences. A prototype automatic target recognition system is demonstrated. Objects in real video imagery are correctly identified by the spatio-temporal Hidden Markov Model classifiers trained on synthetic data.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Hidden Markov Model,Pattern Recognition, Motion Analysis, Signal Processing	133
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL