SANDIA REPORT

SAND98–0507 • UC–330
Unlimited Release
Printed February 1998

Sensor-Fusion-Based Biometric Identity Verification

Dr. Jeffrey J. Carlson, Dr. Ann M. Bouchard, Dr. Gordon C. Osbourn, Rubel F. Martinez, John W. Bartholomew, Dr. Jay B. Jordan, Dr. G. M. Flachs, Dr. Zhonghao Bao, Lei Zhu

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Prices available from (615) 576-8401, FTS 626-8401

Available to the public from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Rd
Springfield, VA 22161

NTIS price codes
Printed copy: A03
Microfiche copy: A01

SAND98-0507
Unlimited Release
Printed February 1998

Distribution
Category UC-330

Sensor-Fusion-Based Biometeric Identity Verification

Dr. Jeffrey J. Carlson and Dr. Ann M. Bouchard
Security Technology Department
Sandia National Laboratories
P. O. Box 5800
Albuquerque, New Mexico 87185-0780

Dr. Gordon C. Osbourn, Rubel F. Martinez, and John W. Bartholomew
Vision Science, Pattern Recognition, and Multi-Sensor Algorithms Department
Sandia National Laboratories
P. O. Box 5800
Albuquerque, New Mexico 87185-1423

Dr. Jay B. Jordan, Dr. G. M. Flachs,
Dr. Zhonghao Bao, and Lei Zhu
Electronic Vision Research Laboratory
New Mexico State University
Las Cruces, New Mexico 88003

Abstract

Future generation automated human biometric identification and verification will require multiple features/sensors together with internal and external information sources to achieve high performance, accuracy, and reliability in uncontrolled environments. The primary objective of the proposed research is to develop a theoretical and practical basis for identifying and verifying people using standoff biometric features that can be obtained with minimal inconvenience during the verification process. The basic problem involves selecting sensors and discovering features that provide sufficient information to reliably verify a person's identity under the uncertainties caused by measurement errors and tactics of uncooperative subjects. A system was developed for discovering hand, face, ear, and voice features and fusing them to verify the identity of people. The system obtains its robustness and reliably by fusing many coarse and easily measured features into a near minimal probability of error decision algorithm.

Intentionally Left Blank

Contents

Intentionally Left Blank

1. INTRODUCTION

The basic problem under investigation is the development of a computer vision system for identifying or verifying the identity of an individual for controlling entry into secure areas or access to sensitive information. The goal is to develop a system with a high probability of detecting an unauthorized entry or access attempt with minimal inconvenience to authorized personnel. Facial, hand, ear, and voice features are investigated to evaluate their performance in terms of the complexity of the recognition problem. A number of coarse features are extracted from face and hand images and from digitized voice and acoustic ear signals. These are analyzed to establish their ability to recognize people. The thrust of the research is to analyze the feature space of the problem and to establish the features that can be fused to reduce the complexity of the recognition problem, consequently reducing the error probability and computational effort.

Our research followed two parallel paths. One path focused on a novel concept for biometric identity verification, an acoustic signature of the individual's ear canal. The concept was to emit an acoustic signal of known spectral content into the individual's ear canal and record the spectrum of the reflected signal. Under highly idealized experimental conditions, the reflectivity of different frequencies is highly sensitive to individual ear canal shape, and in principle could be used to verify an individual's identity. However, under variable conditions, such as those expected in an access control application, and using moderate-cost equipment that people find acceptable to interact with, the reproducibility of the reflected signals was poor. We therefore elected not to pursue this avenue further.

The second path involved developing an access control portal to monitor the biometric characteristics of a person entering a secure area, acquiring a feature signature database and developing information fusion algorithms to discover and fuse features into near minimal probability of error decision algorithms. Cameras are used to obtain hand and facial features, and a microphone obtains voice characteristics as a person enters the facility. A large signature database was developed of hand, facial, and voice features to provide a basis for discovering features for distinguishing people. An information fusion algorithm was developed to discover and fuse the features to perform the verification tasks. The fusion process provides the theoretical foundations for the verification system. Performance bounds are established for individual and weighted combinations of features. These performance bounds provide a basis to determine any performance degradation caused by feature measurement errors and fusion of the features into decision algorithms.

Multi-resolution wavelets are used to discover signature features, often not motivated by human insights, to distinguish signatures. The wavelet transform reduces the resolution of the signature while still retaining the basic structural and frequency components of the signature. Models are derived that can generate the signatures. Parameters are extracted from the models that maximally distinguish the signatures of interest. The result of the feature-discovery process is a multidimensional feature space containing features that can be used to design a near optimal decision module.

The information fusion algorithm selects and weights the features to establish their performance by estimating the probability of error. A genetic selection algorithm was developed to evolve a weighted feature vector to minimizes the probability of error. Genetic algorithms (GA) are very

effective at finding near optimal solutions in complex, high-dimensional problems. The properties that make the GA robust to local extrema also make it computational-intensive. Recent research has enhanced the search and adaptation mechanism by coding the features so that previous experiences can be passed along to promote more desirable crossovers. During the evolutionary process when the performance of the feature vector is increasing rapidly, the crossover rate can be increased, while during periods when the performance is stagnant, the mutation rate can be increased. These and other recent improvements have significantly reduced the search time for GAs.

Section 2 of this paper describes the mathematical framework for the feature discovery, fusion, and selection. Section 3 discusses the procedures and results of feature extraction and fusion of the hand, voice, and face for biometric identity verification. Conclusions are provided in Section 4.

2. MATHEMATICAL FRAMEWORK

A mathematical framework is presented for the feature-based information fusion problem. The problem is formulated in terms of a statistical hypothesis test. A fusion theorem is presented that ensures that the minimum probability of error cannot increase by adding more features. It is also shown that an n-dimensional feature vector can be fused into a single feature without increasing the minimum probability of error. A weighted nearest-neighbor (WNN) fusion model is used to fuse features into a near-minimum probability of error decision algorithm. Wavelets are used to assist in the discovery and extraction of features at different levels of resolution. Statistical methods are presented to evaluate features in high dimensional spaces. A genetic algorithm is used to select and weight the features to obtain a near-minimum probability of error solution.

2.1 Problem Formulation

Many decision and control problems can be formulated as either an m-ary statistical hypothesis test or a set of binary statistical hypothesis tests, one for each decision class. For mathematical convenience, the decision and control problem is described here as a binary hypothesis test. In binary hypothesis testing, a statement or claim that something is true, called the null hypothesis H_o, is tested against its alternative H_a to establish with confidence the most probable decision. Consider the features as a vector of random variables $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ used to distinguish the decision class C_j from the other classes and let $x = (x_1, x_2, \ldots, x_n)$ be a given measurement of these random variables. When the features are measured $y = (y_1, y_2, \ldots, y_n)$ from an unknown decision class and compared to measurements from a known class $x = (x_1, x_2, \ldots, x_n)$, the decision is formulated as:

H_o: The feature measurements y came from the known class;

H_a: The feature measurements y came from a different class.

The feature measurements are generally perturbed by measurement noise and random factors including many environmental and perhaps information warfare factors. The conditional probability density functions (PDFs) under the H_o and H_a hypotheses are represented as $f(x_1,x_2,\ldots,x_N \mid H_o)=f(\mathbf{x} \mid H_o)$ and $f(x_1,x_2,\ldots,x_N \mid H_a)=f(\mathbf{x} \mid H_a)$. If these PDFs are known and the

a priori probabilities of occurrence of each hypothesis are also known, an optimal hypothesis test, in terms of minimizing the probability of making an error, can be formulated. The minimum probability of error can also be computed, giving a bound on the achievable performance associated with the chosen features. The difficulty lies in estimating the PDFs.

Parameter estimation can often be used to estimate $f(\mathbf{x} \mid H_o)$. Relative frequency histograms are more appropriate for estimating $f(\mathbf{x} \mid H_a)$. A simple and often useful way to address the random perturbations is to model their overall effect as a zero mean, stationary, white random process. When a feature measurement x_i, taken from the known class, and y_i, taken from an unknown class, are affected by realizations of such random measurement processes n_j and n_k respectively, then $x_i=x_d+n_k$ and $y_i=y_d+n_j$, where x_d and y_d are the actual deterministic feature vectors. Under the H_o hypothesis, $x_d=y_d$ and the difference statistic $D = x_i-y_i$ depends only on the joint distribution of the noise components. If the joint distribution of the components are normal $N(0,\sigma_i^2)$, then the difference statistic also has a normal distribution $N(0,2\sigma_i^2)$. The related squared difference statistic $SD=(x_i-y_i)^2/(2\sigma_i^2)$ has a $\chi^2(1)$ distribution. The sum of the squared difference statistic (SSD) over n features is often used to compare features from an unknown class to a known class.

$$SSD = \sum_{i=0}^{n} \frac{(x_i - y_i)^2}{2\sigma_i^2} \tag{2-1}$$

If the features are independent, the distribution of the SSD under the H_o hypothesis is a sum of n independent $\chi^2(1)$ random variables, which is $\chi^2(n)$ distributed [7,17]. These results can be used to estimate the distribution of the SSD statistic by estimating the parameters of the distribution. The distribution under the H_a hypothesis is generally not known and relative-frequency histograms are used as estimates.

Approximating the minimum probability of error using estimates of $f(\mathbf{x} \mid H_o)$ and $f(\mathbf{x} \mid H_a)$ requires a priori knowledge of the probability of occurrence of each hypothesis $P(H_o)$ and $P(H_a)$. These a priori probabilities are often not known and an equally likely assumption is often used to estimate the minimum probability of error (MPE). The MPE is a measure of the overlap of two joint distributions in the feature measurement space that has proven to be an effective measure of the ability of a set of features to distinguish objects from different classes. For continuous random variables, where Ω is the set of all $\mathbf{x}$, the MPE is defined in terms of these joint PDFs. For discrete random variables, the MPE is defined in terms of probability mass functions. The operator $\wedge$ selects the minimum.

$$MPE(\mathbf{x}) = \int_{\Omega} P(H_o) f(\mathbf{x}|Ho) \wedge P(H_a) f(\mathbf{x}|Ha) d\mathbf{x}. \tag{2-2}$$

$$MPE(\mathbf{x}) = \sum_{\Omega} P(H_o) f(\mathbf{x}|Ho) \wedge P(H_a) f(\mathbf{x}|Ha). \tag{2-3}$$

2.2 Feature Fusion Theorem

Two theorems [11] are presented to establish mathematical foundations for our approach to feature-based fusion. The first theorem shows that increasing the number of features cannot

increase the MPE of the decision task. The second theorem provides constraints on the information fusing algorithm to maintain an MPE decision algorithm.

Theorem 2.2.1: MPE(**x**,**z**) ≤ MPE(**x**) where $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ represents an n dimensional feature vector and $\mathbf{z} = (\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_m)$ is an additional feature vector.

By the definition of the MPE

$$MPE(\mathbf{X}) = \int_{R_x} p_i \, f(\mathbf{x}|C_i) \wedge p_j \, f(\mathbf{x}|C_j) d\mathbf{x}, \text{ and} \tag{2-4}$$

$$MPE(\mathbf{X},\mathbf{Z}) = \int_{R_z} \int_{R_z} pf(\mathbf{x},\mathbf{z}|C_i) \wedge pf(\mathbf{x},\mathbf{z}|C_i) d\mathbf{z} d\mathbf{x} \tag{2-5}$$

The proof begins with the inequality

$$\int_{R_z} p_i \, f(\mathbf{x},\mathbf{z}|C_i) \wedge p_j \, f(\mathbf{x},\mathbf{z}|C_j) d\mathbf{z} \leq \int_{R_z} p_i \, f(\mathbf{x},\mathbf{z}|C_i) d\mathbf{z} = p_i \, f(\mathbf{x}|C_i) . \tag{2-6}$$

Similarly,

$$\int_{R_z} p_i \, f(\mathbf{x},\mathbf{z}|C_i) \wedge p_j \, f(\mathbf{x},\mathbf{z}|C_j) d\mathbf{z} \leq \int_{R_z} p_j \, f(\mathbf{x},\mathbf{z}|C_j) d\mathbf{z} = p_j \, f(\mathbf{x}|C_j) . \tag{2-7}$$

Hence,

$$\int_{R_z} p_i \, f(\mathbf{x},\mathbf{z}|C_i) \wedge p_j \, f(\mathbf{x},\mathbf{z}|C_j) d\mathbf{z} \leq p_i \, f(\mathbf{x}|D_i) \wedge p_j \, f(\mathbf{x}|C_j) . \tag{2-8}$$

Integrating the last inequality over the space define by R_x completes the proof

$$MPE(\mathbf{X},\mathbf{Z}) \leq \int_{R_x} p_i \, f(\mathbf{x}|C_i) \wedge p_j \, f(\mathbf{x}|C_j) d\mathbf{x} = MPE(\mathbf{X}) . \tag{2-9}$$

Consequently, if sufficient samples are used to establish the probability density functions and the decision surfaces, then MPE should not increase with additional features.

The next theorem suggests a method to fuse or consolidate information from multiple sensors and preserve an MPE decision algorithm. The significance of this result is that large dimensional feature vectors can be fused into a single dimensional feature while preserving the MPE. A research goal in information fusion is to discover these powerful features and ways to efficiently measure them. There are many functions that can be used to combine multiple information sources into a single composite source while preserving the MPE. Some interesting fusion functions, $\mathbf{y} = g(\mathbf{x})$, that preserve MPE are:

(1) $\mathbf{y} = p_i f(\mathbf{x}|C_i) - p_j f(\mathbf{x}|C_j)$,

(2) $\mathbf{y} = p_i f(\mathbf{x}|C_i) / p_j f(\mathbf{x}|C_j)$ and

(3) $\mathbf{y} = 1$ if $p_i f(\mathbf{x}|C_i) > p_j f(\mathbf{x}|C_j)$,
$\mathbf{y} = 0$ if $p_i f(\mathbf{x}|C_i) = p_j f(\mathbf{x}|C_j)$,
$\mathbf{y} = -1$ if $p_i f(\mathbf{x}|C_i) < p_j f(\mathbf{x}|C_j)$.

All of these MPE preserving fusion functions, however, require knowledge of the multidimensional joint PDFs for each decision class.

Theorem 2.2.2: If the fusion function $\mathbf{y} = g(\mathbf{x})$ is used then MPE(Y) ≥ MPE(X) and equality holds if and only if $R(\mathbf{y}) \cap R_1 = \varnothing$ or $R(\mathbf{y}) \cap R_3 = \varnothing$ where $R(\mathbf{y}) = \{\mathbf{x} | g(\mathbf{x}) = \mathbf{y}\}$, $R_1 = \{\mathbf{x} | p_i f(\mathbf{x}|C_i) < p_j f(\mathbf{x}|C_j)\}$, $R_2 = \{\mathbf{x} | p_i f(\mathbf{x}|C_i) = p_j f(\mathbf{x}|C_j)\}$ and $R_3 = \{\mathbf{x} | p_i f(\mathbf{x}|C_i) > p_j f(\mathbf{x}|C_j)\}$.

Recall the definition of MPE(**X**)

$$MPE(\mathbf{X}) = \int_{R_x} p_i f(\mathbf{x}|C_i) \wedge p_j f(\mathbf{x}|C_j) d\mathbf{x}, \tag{2-10}$$

and observe that for $\mathbf{y} = g(\mathbf{x})$,

$$MPE(\mathbf{Y}) = \int_{R_y} p_i f(\mathbf{y}|C_i) \wedge p_j f(\mathbf{y}|C_j) d\mathbf{y} \tag{2-11}$$

where

$$f(\mathbf{y}|C_i) = \int_{R(y)} f(\mathbf{x}|C_i) d\mathbf{x}, \tag{2-12}$$

and

$$f(\mathbf{y}|C_j) = \int_{R(y)} f(\mathbf{x}|C_j) d\mathbf{x}, \tag{2-13}$$

and $R(\mathbf{y}) = \{\mathbf{x} | g(\mathbf{x})=\mathbf{y}\}$. If, over $R(\mathbf{y})$, $R(\mathbf{y}) \cap R_1 = \varnothing$ or $R(\mathbf{y}) \cap R_3 = \varnothing$ then either $p_j f(\mathbf{x}|C_j) \le p_i f(\mathbf{x}|C_i)$ or $p_i f(\mathbf{x}|C_i) \le p_j f(\mathbf{x}|C_j)$, respectively. Under either condition,

$$\int_{R(y)} p_i f(\mathbf{x}|C_i) d\mathbf{x} \wedge \int_{R(y)} p_j f(\mathbf{x}|C_j) d\mathbf{x} = \int_{R(y)} p_i f(\mathbf{x}|C_i) \wedge p_j f(\mathbf{x}|C_j) d\mathbf{x}. \tag{2-14}$$

Hence, it follows that

$$p_i f(\mathbf{y}|C_i) \wedge p_j f(\mathbf{y}|C_j) = \int_{R(y)} p_i f(\mathbf{x}|C_i) \wedge p_j f(\mathbf{x}|C_j) d\mathbf{x}. \tag{2-15}$$

If, for each $\mathbf{y} = g(\mathbf{x})$, $R(\mathbf{y}) \cap R_1 = \varnothing$ or $R(\mathbf{y}) \cap R_3 = \varnothing$ then

$$\int_{R_y} p_i\, f(\mathbf{y}|C_i) \wedge p_j\, f(\mathbf{y}|C_j)d\mathbf{y} = \int_{R_y}\int_{R(y)} p_i\, f(\mathbf{x}|C_i) \wedge p_j\, f(\mathbf{x}|C_j)d\mathbf{x}d\mathbf{y} \qquad (2\text{-}16)$$

and consequently, MPE(**Y**) = MPE(**X**). To complete the proof it is shown that if $R(\mathbf{y})\cap R_1 \neq \varnothing$ and $R(\mathbf{y})\cap R_3 \neq \varnothing$ then MPE(**Y**) > MPE(**X**). If $R(\mathbf{y})\cap R_1 \neq \varnothing$ then

$$\int_{R(y)} p_i\, f(\mathbf{x}|C_i)d\mathbf{x} > \int_{R(y)} p_i\, f(\mathbf{x}|C_i) \wedge p_j\, f(\mathbf{x}|C_j)d\mathbf{x} \qquad (2\text{-}17)$$

and if $R(\mathbf{y})\cap R_3 \neq \varnothing$ then

$$\int_{R(y)} p_j\, f(\mathbf{x}|C_j)d\mathbf{x} > \int_{R(y)} p_j\, f(\mathbf{x}|C_i) \wedge p_j\, f(\mathbf{x}|C_j)d\mathbf{x}, \qquad (2\text{-}18)$$

Hence,

$$\int_{R(y)} p_i\, f(\mathbf{x}|C_i)d\mathbf{x} \wedge \int_{R(y)} p_j\, f(\mathbf{x}|C_j)d\mathbf{x} > \int_{R(y)} p_i\, f(\mathbf{x}|C_i) \wedge p_j\, f(\mathbf{x}|C_j)d\mathbf{x}, \qquad (2\text{-}19)$$

$$p_i\, f(\mathbf{y}|C_i) \wedge p_j\, f(\mathbf{y}|C_j) > \int_{R(y)} p_i\, f(\mathbf{x}|C_i) \wedge p_j\, f(\mathbf{x}|C_j)d\mathbf{x}, \qquad (2\text{-}20)$$

and it follows that

$$\int_{R_y} p_i\, f(\mathbf{y}|C_i) \wedge p_j\, f(\mathbf{y}|C_j)d\mathbf{y} > \int_{R_y}\int_{R(y)} p_i\, f(\mathbf{x}|C_i) \wedge p_j\, f(\mathbf{x}|C_j)d\mathbf{x}d\mathbf{y}. \qquad (2\text{-}21)$$

Consequently, MPE(**Y**)>MPE(**X**) when $R(\mathbf{y})\cap R_1 \neq \varnothing$ and $R(\mathbf{y})\cap R_3 \neq \varnothing$. This result motivated a search for a fusion function that retains the MPE solution. The WNN fusion function inherits the property from the nearest-neighbor decision theory [2] that as the number of samples becomes sufficiently large the performance of the fusion algorithm is less than twice the MPE solution.

2.3 Weighted Nearest-Neighbor Fusion Model

The WNN fusion model provides a method for analyzing and fusing multiple features to design and optimize a decision process. The fusion process involves discovering the features that can be fused to obtain robust and minimum error decision algorithms. A WNN model is used to provide the mathematical framework for fusing features into near-minimum probability of error decision algorithms.

Training samples are used to guide the feature selection and fusion process. Each training sample $x = (x_1, x_2, \ldots, x_n)$ represents a point in an n dimensional space. N_j training samples are used to characterize the statistical behavior of each decision class C_j. A weighted distance d_j=WNN(y,x^j)

from an unknown sample y to the nearest neighbor x^j of class C_j is used to fuse the features and decide class membership. The WNN distance is given as

$$WNN(y,x^j)=\sum_{k=1}^{n} w_k\left(y_k-x_k^j\right)^2 \tag{2-22}$$

An unknown sample y is given C_j membership if its nearest neighbor is in class C_j and $d_j < T_j$. The thresholds T_j are chosen to achieve the desired false acceptance and rejection rates. The weights $w_k \in [0,1.0]$ are chosen to minimize the probability of error of the decision process using a genetic algorithm search process. A weight of zero effectively eliminates the feature from the decision process and indicates the feature does not contribute to the minimal error solution. The higher the weight the more the feature contributes to the decision process.

Training samples are used to estimate the one dimensional conditional probability density functions for the minimal in-class distance $f(d_j|C_j)\forall j$ and the minimal out-of-class distance $f(d_j|C_k)j\neq k$. The minimum probability of error is estimated by integrating the minimum of the conditional probability density estimates over the observation space O(d) of the in-class and out-of-class distances. The minimum probability of error (mpe) estimate is given by

$$mpe=\frac{1}{2}\sum_{O(d)} f(d_j|C_j)\wedge f(d_j|C_k)j\neq k \tag{2-23}$$

where the symbol $\wedge$ is a minimum select operator and the a priori probabilities are chosen equal $P(C_j)=P(C_k)=½$. The feature weights directly affect the distance measurements which in turn affect the conditional probability functions and the minimum probability of error. The genetic optimization method is used to select the weights to minimize mpe given a set of potential features. The net result of these operations is to select and fuse the features to achieve near-minimum probability of error performance.

A key variable in the WNN fusion process is the number of samples required to approach a minimum probability of error solution. Estimates for the sample size can be theoretically established by determining the number of samples required to estimate the parameters μ and σ of a normal distribution $N(\mu,\sigma^2)$. Letting m and s^2 represent the sample mean and variance from ns samples, it is known [3] that $m\in(\mu-c\sigma,\mu+c\sigma)$ with confidence $1-\alpha$ when the number of samples ns is given in terms of the size of the confidence interval c and a value Z_t determined from the normal distribution N(0,1)

$$ns=\frac{Z_{\alpha/2}^2}{c^2} \quad where \quad \int_{Z_t}^{\infty}\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}dx=t. \tag{2-24}$$

This results from the sample mean computed from random samples from $N(\mu,\sigma^2)$ having a normal distribution with mean μ and variance $\frac{\sigma}{\sqrt{ns}}$. Therefore, for any given confidence $1-\alpha$

$$P\left(-Z_{\alpha/2} < \frac{m-\mu}{\sigma/\sqrt{ns}} < Z_{\alpha/2} \right) = 1-\alpha . \tag{2-25}$$

Hence

$$P\left(\mu - Z_{\alpha/2}\frac{\sigma}{\sqrt{ns}} < m < \mu + Z_{\alpha/2}\frac{\sigma}{\sqrt{ns}} \right) = 1-\alpha , \tag{2-26}$$

and

$$P\left(\mu - c\sigma < m < \mu + c\sigma \right) = 1-\alpha , \tag{2-27}$$

where

$$c = \frac{Z_{\alpha/2}}{\sqrt{ns}} . \tag{2-28}$$

Solving for the sample size ns yields the result

$$ns = \left(\frac{Z_{\alpha/2}}{c} \right)^2 . \tag{2-29}$$

Consequently, a sample size of ns guarantees that the probability of an m falling in $(\mu-c\sigma,\mu+c\sigma)$ is $1-\alpha$. The sample size can also be chosen to guarantee that the variance is in a given confidence interval. Since $\frac{(ns-1)S^2}{\sigma^2}$ is χ^2 distributed with degree of freedom ns-1

$$P\left(\chi^2_{1-\alpha/2}(ns-1) < \frac{(ns-1)S^2}{\sigma^2} < \chi^2_{\alpha/2}(ns-1) \right) = 1-\alpha , \tag{2-30}$$

and

$$P\left(\frac{\chi^2_{1-\alpha/2}(ns-1)\,\sigma^2}{ns-1} < S^2 < \frac{\chi^2_{\alpha/2}(ns-1)\,\sigma^2}{ns-1} \right) = 1-\alpha . \tag{2-31}$$

for any confidence 1-α. By taking the square root we have

$$P\left(\sqrt{\frac{\chi^2_{1-\alpha/2}(ns-1)}{ns-1}}\,\sigma < S < \sqrt{\frac{\chi^2_{\alpha/2}(ns-1)}{ns-1}}\,\sigma \right) = 1-\alpha . \tag{2-32}$$

Let

$$P(\sigma - d_1\sigma < S < \sigma + d_2\sigma) = 1-\alpha , \tag{2-33}$$

where

$$d_1 = 1 - \sqrt{\frac{\chi^2_{1-\alpha/2}(ns-1)}{ns-1}} , \tag{2-34}$$

and

$$d_2 = \sqrt{\frac{\chi^2_{\alpha/2}(ns-1)}{ns-1}} - 1 . \tag{2-35}$$

When $d_1+d_2=d$, the probability of S falling in $(\sigma-d_1\sigma,\sigma+d_2)$ is greater than 1-α if

$$\sqrt{\frac{\chi^2_{\alpha/2}(ns-1)}{ns-1}} - \sqrt{\frac{\chi^2_{1-\alpha/2}(ns-1)}{ns-1}} < d . \tag{2-36}$$

Finally, since m and S are independent,

$$P\left(-Z_{\alpha/2} < \frac{m-\mu}{\sigma/\sqrt{ns}} < Z_{\alpha/2} \wedge X^2_{1-\alpha/2}(ns-2) < \frac{(ns-1)S^2}{\sigma^2} < X^2_{\alpha/2}(ns-1) \right) \tag{2-37}$$

$$= P\left(-Z_{\alpha/2} < \frac{m-\mu}{\sigma/\sqrt{ns}} < Z_{\alpha/2} \right) \bullet P\left(X^2_{1-\alpha/2}(ns-1) < \frac{(ns-1)S^2}{\sigma^2} < X^2_{\alpha/2}(ns-1) \right) \tag{2-38}$$

$$= (1-\alpha)^2$$

the probability of both m falling in $(\mu-c\sigma,\mu+c\sigma)$ and S falling in $(\sigma-d_1\sigma,\sigma+d_2)$ is $(1-\alpha)^2$. Hence, by selecting a sample size larger than that required for m and S yields a confidence of $(1-\alpha)^2$. For example, if the confidence interval size is c=0.52, and if the confidence is $(1-\alpha)^2=0.9$, then the number of required samples ns=10. This result holds only when the number of features n=1. If the number of features is n and they are independent and identically distributed $N(\mu,\sigma^2)$, then the number of samples required is given by

$$ns = \left(\frac{Z_{\frac{\alpha}{2}}}{c} \right)^{2n} \tag{2-39}$$

where $a = \frac{Z_{\frac{\alpha}{2}}}{c} > 1$. For example, if the number of features is n=4, if the desired confidence is 1-α=.9 and if the confidence interval size is c=0.52 then a=3.163 and the required number of samples ns=1002. For many problems with many features the number of required samples is an unacceptable large number. In our experiments with the human identity verification problem, the number of features is on the order of n=40, which leads to an unrealistic number of samples. These estimates are based on estimating the parameters of the joint feature distributions and determining the minimum probability of error decision surfaces.

Our experience with the WNN decision process, however, indicates that far fewer samples are required to approach the minimum probability of error solution. To illustrate the benefit of the WNN decision process, consider a two-class decision problem with n=40 features which would require a very large number of samples to estimate the joint distributions. If the features are independent with the same variance and normally distributed then only two samples are required (at the expected values of these two distributions) to define the optimal WNN decision surface. Any two samples co-aligned with the peaks of the distributions will form the optimal decision boundary. In our studies with the human verification problem with 44 features only five to ten samples for each person has produces excellent results.

To further illustrate the number of training samples required to produce near-minimum probability of error solutions, a large database of 13 hand features (n=13) was used to establish an experimental relationship between the number of samples and the probability of error. A distribution of pe was obtained for a sample size of ns=2 by random sampling the training data and determining the pe from the remaining data. The experiment was repeated for ns=5, 10, and 20. These distributions are given in Figure 2–1. As the number of samples is increased the variance of the distributions decreases and the means decrease toward the minimum probability of error solution.

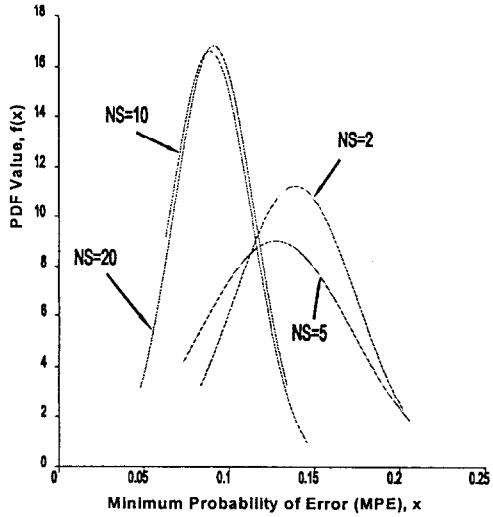


Figure 2–1. Distribution of error probabilities

2.4 Feature Discovery Module

The goal of the feature discovery module is to assist in the discovery of multiscale features to distinguish object classes, as shown in Figure 2–2.

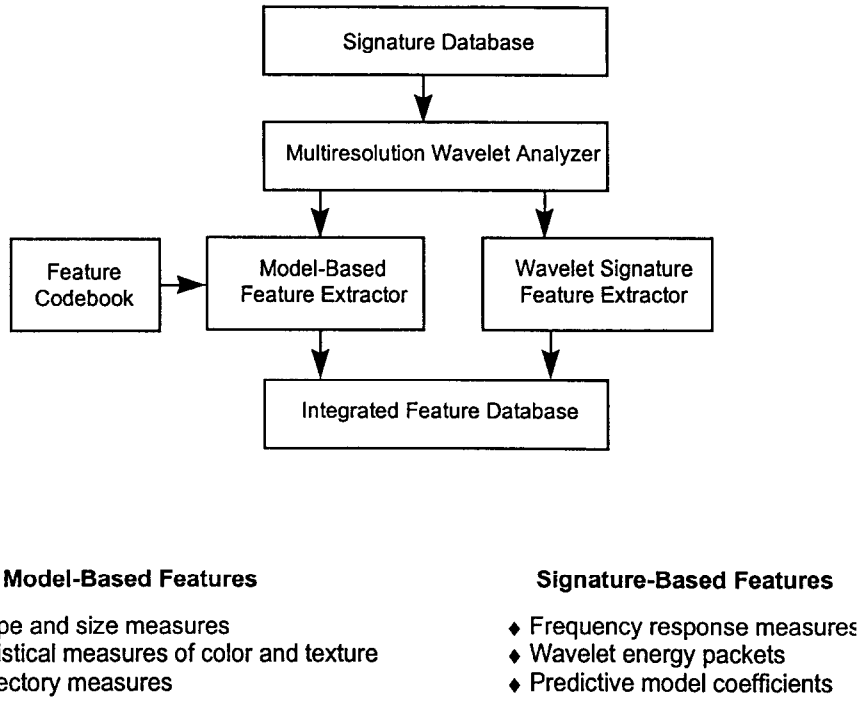
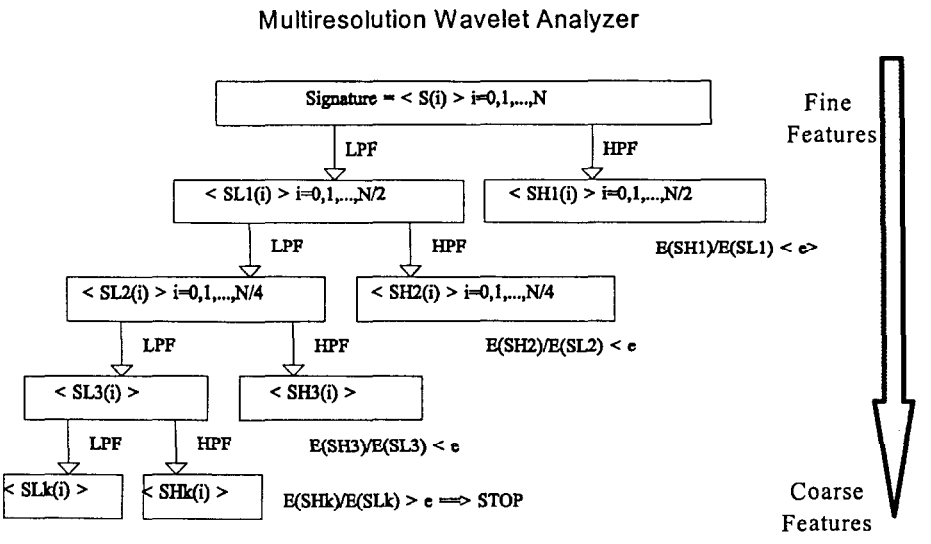


Figure 2–2. Feature discovery module

The wavelet transform provides a method to discover model- and signature-based features at different levels of resolution. Model-based features include the geometric features that are used by humans and features that are developed by modeling the fundamental laws governing the sensors, atmosphere, and objects to be distinguished. The signature-based features include the subtle spectral features that often do not have a geometric interpretation. The wavelet transform is used to generate a hierarchical pyramid representation where features can be extracted with different resolution levels, ranging from coarse to fine. By using the wavelet filter/decimation process, signatures are reduced in resolution while the low-pass component contains the basic spectral-frequency characteristics. At each level, features can be extracted and analyzed in terms of their ability to separate the classes. When features are proven effective, they are added to the feature codebook. The feature codebook provides a dictionary of the features that have been shown to be effective in distinguishing the object classes. This dictionary represents an accumulation of knowledge obtained from experience and from detailed system models based upon fundamental laws governing the ability of the sensor to distinguish the objects and backgrounds of interest.



The wavelet generates a hierarchical pyramid representation of a signature at different resolutions for feature discover.

Figure 2–3. Filter-decimation operation diagram

The wavelet transform is used to reduce signals to their significant features. Let h(n) be a low pass filter with finite impulse response as described by Daubechies[13]. Let g(n) be a band pass or a high pass filter with finite impulse response defined by: $g(n)=(-1)^{-1}h(1-n)$.

Let x(n) be a signal sequence with data size of N where N is a power of 2. The wavelet transform involves applying the filter pair (h(n), g(n)) to x(n):

$$x_h(n) = \sum_k x(k)h(2n-k), \tag{2-40}$$

$$x_g(n) = \sum_k x(k)g(2n-k), \tag{2-41}$$

followed by a decimation of 2 (reduction in the spatial resolution by 2). The data size of $x_h(n)$ and $x_g(n)$ is one-half of the data size of x(n). Using the operation repeatedly, the original input signal sequence with N points is partitioned into its low-frequency and high-frequency components. This procedure is also known as filter-decimation operation. A filter-decimation example is given in Figure 2–3 where the high- and low-frequency parts are placed in bins. At each level of the filter-decimation operation, only one-half of the samples are kept in the same space and frequency domain. A well selected filter pair separates different frequency components so that there is little information overlap in the low- and high-frequency parts at each level of the transform.

When the wavelet transform is used to reduce the data size for a signal sequence, a major consideration is defining the level for stopping the decimation procedure. The objective is to reduce the resolution while keeping the significant properties of the signal. An energy map is used to determine at which level to stop the wavelet transformation and establish the bins that contain the significant information for a feature space. The bin location within a decomposition tree is represented by using the notations $SL_j(i)$ and $SH_j(i)$ to represent the low-frequency part and the high-frequency part, respectively. The wavelet transform reduces the high-frequency noise while retaining the signal structural information. Hence, it is possible to represent the signal efficiently with fewer samples. In other words, the signal may be classified by using the lower resolution information. The energy of the signal in each resolution also provides important information for recognition purposes. The energies in the low-frequency part EL_j and in the high-frequency part EH_j in level j are calculated using

$$EL_j = \sum_i A_{2^j}(i) * A_{2^j}(i) \quad and \quad EH_j = \sum_i D_{2^j}(i) * D_{2^j}(i). \tag{2-42}$$

The energy ratio of the high and low-frequency parts (PR_j) is given in percent by

$$PR_j\% = \frac{EH_j}{EL_j} * 100. \tag{2-43}$$

Higher energy signals contain the more significant structural information than do lower energy signals. The energy ratio is used for extracting the important features. If PR_j is less than a threshold *e*, then the information contained in the high-frequency part is small compared to the low-frequency part. When the PR_j is greater than *e*, the filtering process is stopped because the structural features of the signal are going into the high-frequency bin. The effective features are extracted from the low-frequency bin of the previous level and used for recognition.

2.5 Feature Selection Module

The feature selection module, Figure 2–4, selects and establishes weights for the features to form a weighted-feature vector that minimizes the probability of error. A GSA selects the features from the possible feature set and establishes the weights to minimize the probability of error. The result of this process is a weighted-feature vector that can be partitioned to meet processing constraints and minimize the probability of error.

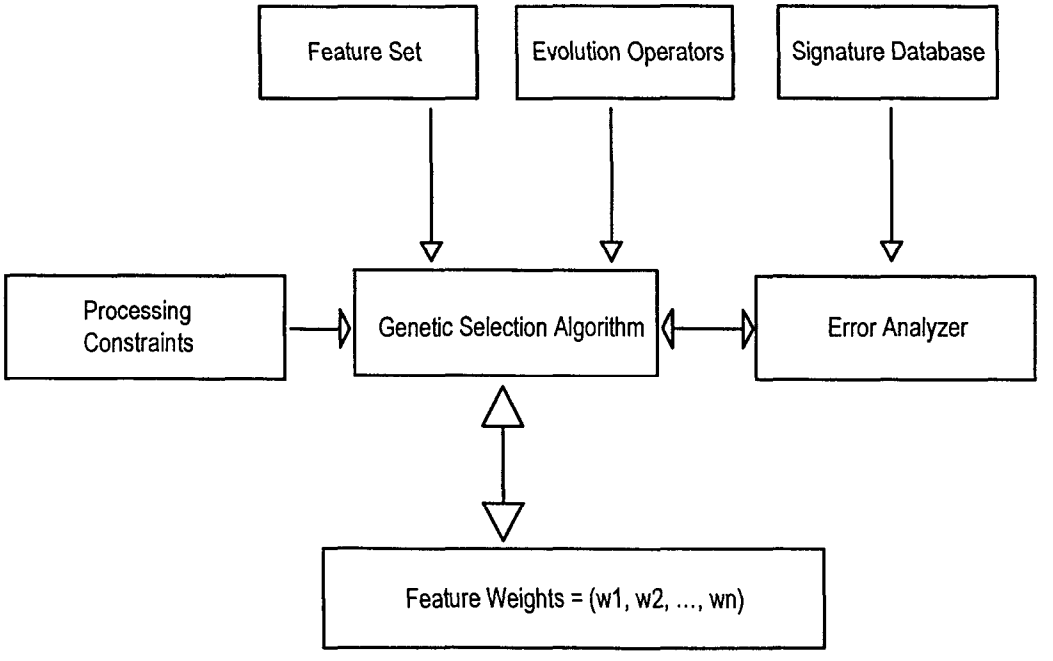


Figure 2–4. Feature selection module

The central element of the feature selection module is the GSA. It selects combinations of features from the feature set generated by the discovery module and utilizes the evolution operators to weight the features to obtain a near-minimum probability of error solution. If any processing constraints are imposed, the GSA ensures that they are satisfied. Given the selected features and weights, the error analyzer uses the training samples to estimate the conditional probability density functions for the minimal nearest-neighbor in-class distance $f(d_j|C_j)\ \forall\ j$ and the minimal nearest-neighbor out-of-class distance $f(d_j|C_k)j\neq k$. The minimum probability of error is estimated by integrating the minimum of the conditional probability density functions over the observation space O(d) of the in-class and out-of-class distances. For discrete probability density functions the minimum probability of error (mpe) is given by

$$mpe = \frac{1}{2}\sum_{O(d)} f(d_j|C_j) \wedge f(d_j|C_k)\, j \neq k \tag{2-44}$$

where the symbol $\wedge$ is a minimum select operator. The features selected and the feature weights directly affect the distance measurements, which in turn affect the conditional probability functions and the probability of error. The genetic optimization problem is used to select the weights to minimize the mpe given a set of potential features. The net result of these operations is the selection and fusion of the features to achieve a near-minimal probability of error decision algorithm.

The purpose of GSAs is to establish an initial population (P) of NP possible solutions (members) to a given problem. For the current problem, a member is an ordered collection of features and their corresponding weights. These population members compete, mate, and mutate to generate a new population of NP members that are better qualified to solve the problem. This process continues until a desired performance level is reached or a limit on the number of generations (NG) is reached.

The initial P is formed by establishing the performance of each individual feature in the given feature set. A weighted list of the features is formed to generate a weighted random selection of NP members based upon their individual performance. Weights are assigned for each feature selected based upon the relative performance of the features selected. After the initial population is selected the mutation, reproduction, and mating (crossover) operators generate the natural selection of the next population. The mutation operator selects a random member from the population, selects a random pointer to a feature in the member, performs a weighted random selection of a new feature not already in the member (could be the same feature) and randomly assigns weights to the new features and all features above the pointer. This allows new features to enter the population with randomly assigned weights. The reproduction operator performs a weighted random select of a member from the population and randomly perturbs the feature weights to generate a new member with similar performance. This process promotes new members with characteristics similar to members with good performance. The crossover operator performs two weighted random selections of two members based on their performance. Two random pointers are selected in the feature string, and the common features and fuzzified weights are exchanged between the pointers to prevent duplication of features. After the crossover process is completed, the new solutions are evaluated and only the best are introduced into the next generation. Using these operators, each old member is replaced by randomly selecting an operator with the probabilities p_m=0.2 for mutation, p_r=0.1 for reproduction, and p_c=0.7 for crossover and generating a new member.

The GSA has demonstrated the ability to quickly generate good solutions that compare well with solutions obtained with neural nets that take much longer to converge. For the human verification problem with n=44 possible features excellent solutions are obtained with only five generations (NG=5) with a population size of NP=40. The training time on a 90MHz personal computer is only a few minutes. A comparable solution using a neural network solution requires several hours of training time.

2.6 Decision Module

A common mathematical foundation for the comparative analysis of statistical, fuzzy, and artificial neural pattern recognition or decision-making systems was developed [15] using abstract algebraic techniques. These techniques characterize the functions generating decision surfaces and the learning/training processes involved in each technique.

Abstract algebra is built on the most basic of foundations: sets, operations on sets, and mappings from set to set. Viewed from this level, all pattern recognition methods are strikingly similar. So much so, in fact, that an abstract pattern recognizer (APR), Figure 2–5, can be defined such that all others are its subsets or special cases. The basic structure of the APR was developed during this project. Typical popular statistical, fuzzy, and artificial neural pattern recognizers are

characterized in the context of the APR [15]. Each of the modern methods has strengths and weaknesses, advantages and disadvantages. Using the concept of an APR, it is possible to design and build hybrid systems that capitalize on the advantages and strengths of all previous methodologies. Each of the modern pattern recognition areas has its own terminology and set of notations. As much as possible, the standard notation for each discipline was preserved and described as it relates to the standard notation of abstract algebra.

The study of pattern recognition has many important goals, such as the development of methodologies for designing and building machines capable of recognizing and learning to recognize objects in natural or unstructured environments. The common task of any pattern recognizer is to obtain a set of observations or measurements from an unknown (or unclassified) pattern and to decide which class the pattern belongs based on the values observed. From an abstract point of view, once pattern recognition is clearly defined within a mathematical framework, the pattern recognition task becomes a simple mathematical problem.

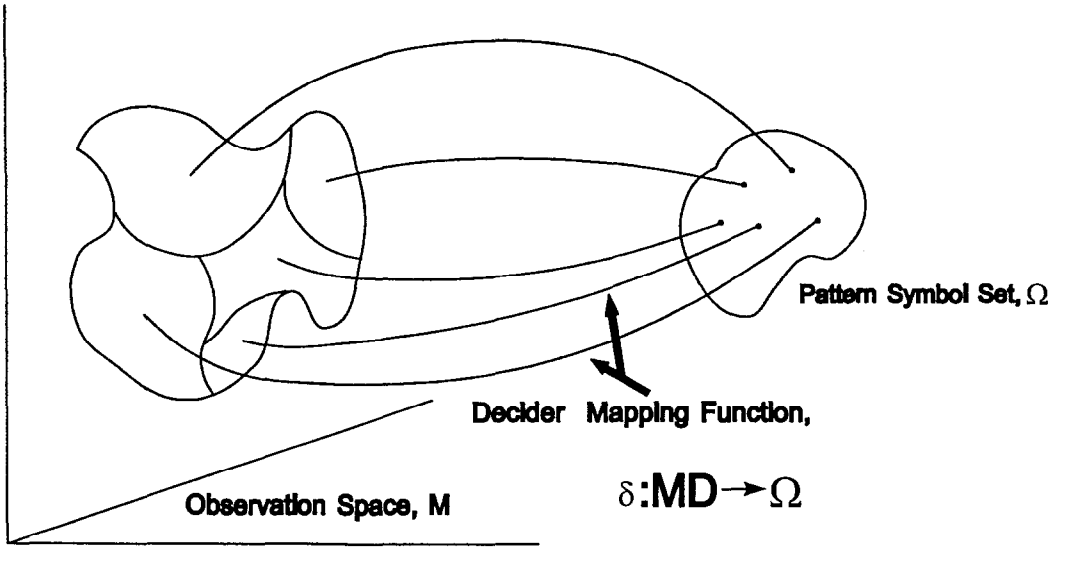


Figure 2–5. Conceptual view of abstract pattern recognizer

A pattern recognizer is a decision maker or decider, D, that can be viewed as a mapping from an arbitrary observation space M to a finite set of symbols representing decisions or object classes, Ω. D is therefore a triple (M,Ω,δ); M is a space (the largest set from which subsets of observations are taken); Ω is a finite set of symbols; and δ is a mapping from M to Ω, $\delta:M\rightarrow\Omega$. The decider is meaningful only if Ω contains two or more elements and M contains at least as many elements as Ω. For most deciders, the mapping δ is many-to-one with the number of elements in M quite large, often not finite. In these cases, the mechanism creating δ defines a partition of M with nonoverlapping subsets of M mapping into elements of Ω. Note that members of the special subset of deciders characterized by finite M are called combinatoric

machines in automata theory with M being a finite set of input symbols and Ω a finite set of output symbols.

There are three basic types of deciders: fixed, trainable, and learning. The mapping δ in a fixed decider is static. It is developed prior to the use of the decider and does not change during decider operation. The trainable decider has a training mode wherein the mapping can be altered in response to one or more sets of measurements whose classification is known a priori. The mapping of a learning decider develops and evolves as the decider encounters and classifies new patterns of unknown classes. There are, of course, many variations of these basic types.

Regardless of the type or implementation method, all pattern recognizers are deciders as defined above. Once a specific pattern recognition task has been defined, the sets M and Ω are the same whether the pattern recognizer be classical deterministic, statistical, fuzzy, artificial neural, or something altogether new. What differs for each is the mapping δ and its generation. For typical examples of each of the pattern recognition methods chosen for comparative analysis, the composition of δ is derived and used as a basis for comparison and contrast. In each of these methods, the mapping δ is expressed as a composition of a decision rule and another set of mappings classically termed discriminant or decision functions. The three methods chosen for analysis are also based on the implicit assumption that M and the subsets of M from which measurements are taken is a measurable space. In this development, M was arbitrarily restricted to be a vector space.

These restrictions were for convenience only and are not contained in the fundamental definition of a decider. Without these restrictions, extraordinarily rich pattern recognizers can be developed using nonmeasurable spaces containing measurable subspaces. The use of a nonmeasurable space M provides a mechanism for incorporating nonmeasurable information dimensions and nonmeasurable a priori knowledge into a pattern recognizer.

Deciders can be considered optimal if they minimize some cost function that rewards correct decisions and penalizes errors. It general, there may be many decision surfaces that minimize a cost function. It is expected that for a given problem, optimal decision surfaces obtained by any of the methods will be similar. The probability of error is an accepted measure of the performance of a decision maker. The optimality criterion is the minimization of the probability of error (MPE). The number of training samples (ns), the amount of time required for training (tt) and the amount of time required to make a decision (td) are also important factors in evaluating a decision maker. The WNN decision maker is a good choice for many applications. The power of the nearest-neighbor decision process reduces the number of training samples, the GSA reduces the training time and if the number of decision-making samples (nds) is not too large the time to make a decision is favorable. Normally the number of decision making samples is much less than the number of training samples (nds<<ns) by eliminating samples that do not significantly affect the decision surfaces as shown in Figure 2–6.

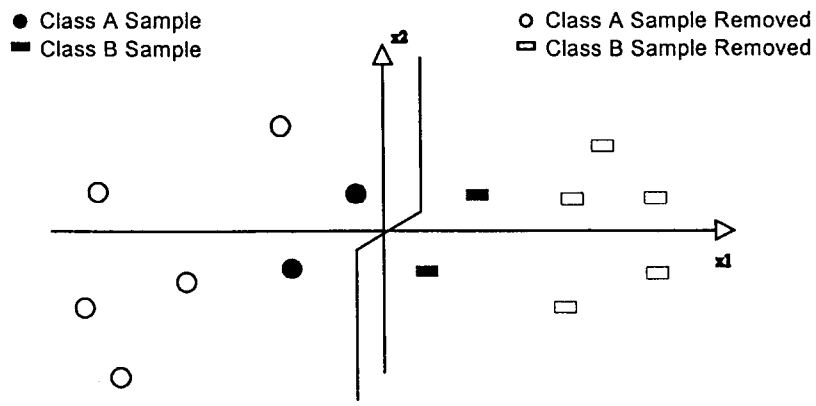


Figure 2–6. Weighted nearest-neighbor decision process

The WNN decision process involves computing the minimum in-class distances d_j and comparing to the class thresholds T_j. It is important to observe that the genetic training algorithm selects the features and establishes the class thresholds T_j during the training phase. Hence, the real time decision process is very fast.

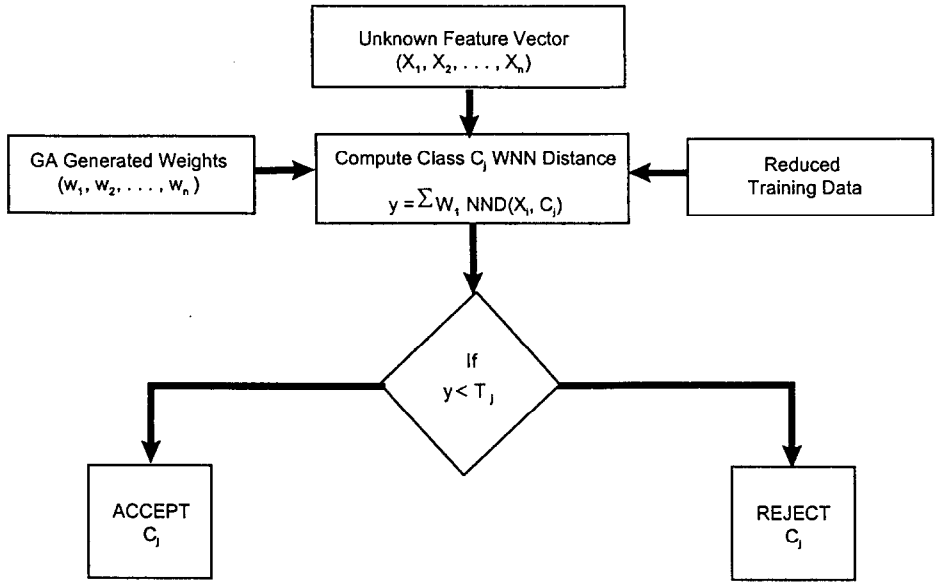


Figure 2–7. Weighted nearest-neighbor decision process

The WNN training and decision performance compares well with neural network performance. The training time of the WNN decision process, however, is much faster and achieves acceptable performance very quickly with the genetic optimization procedure.

3. HUMAN IDENTITY VERIFICATION PROBLEM

A detailed, practical application of the sensor fusion methodology is a solution to the human identity verification problem. The basic problem under investigation is the development of a computer vision system for identifying or verifying the identity of an individual for controlling entry into secure areas or access to sensitive information. The goal is to develop a system with a high probability of detecting an unauthorized entry or access attempt with minimal inconvenience to authorized personnel. Facial, hand and voice features are investigated to evaluate their performance in terms of the complexity of the recognition problem. Several coarse features are extracted from face and hand images and from digitized words. These are analyzed to establish their ability to recognize people. The objective of this research is to analyze the feature space of the problem and to establish the features that can be fused to reduce the complexity of the recognition problem, consequently reducing the error probability and computational effort.

Results of initial experiments were reported by Flachs et al [5] and Carlson et al [6]. The early results were obtained from a Sandia National Laboratories (SNL) database and an early New Mexico State University (NMSU) database. The SNL data set was acquired under ideal conditions. The early NMSU data set was produced under somewhat more relaxed conditions. The results presented here are based on data obtained from an operational access control/portal control system under what would be considered normal operating conditions. The operators of the system and the participants were essentially untrained and, for the most part, had little image processing/pattern recognition experience. Many of the subjects actually tried to defeat the system by quickly moving their hands below the hand sensor or "making faces" in an attempt to fool the facial sensors. The main purpose of the experiment was to test the robustness of the algorithms developed earlier and to gain insight into their operational utility. These tests demonstrate that good features possess the ability to distinguish people and they can be reliably computed.

The configuration of the human biometrics data acquisition system is shown in Figure 3–1. A person attempting to enter a secure area must present his or her right hand, palm-down, over the hand imaging area. A television monitor in front of the hand imaging area shows the user how the hand looks to the system. This monitor also serves the purpose of causing the user to orient his or her face appropriately for facial front and side view cameras. The system automatically senses a person's hand in the hand imaging area and acquires the hand and face images. The system then prompts the user to repeat a sequence of digits flashed one at a time on the monitor. During this time, the voice digitizer captures the spoken numbers.

In the side view image, the facial profile forms a one-dimensional signal for side view features. A gray-level projection along the rows of the front view image forms a one-dimensional signal for front view features. The wavelet feature discovery algorithm is used to reduce the signatures for the side and front views to 12-dimensional feature vectors. In the hand image, the hand is segmented from the background and the length and width of the fingers are combined with three hand-shape features to form an 11-dimensional hand feature vector. Each person is asked to say three numbers with strong vowel sound as they enter the verification room. The voice signature for each number is analyzed with match filters to obtain a pitch frequency profile and the fast Fourier transform is used to obtain the frequency spectrum. The feature discovery process

reduced the pitch frequency profile to six features and the frequency spectrum to eight features. Consequently, 14 features are obtained for each spoken number, forming a 42-dimensional feature vector for the three spoken numbers. Statistical decision techniques are used to evaluate the facial, hand, and voice features in terms of their ability to discriminate humans. The WNN method is used to reliably fuse the features from all of the different sensors.

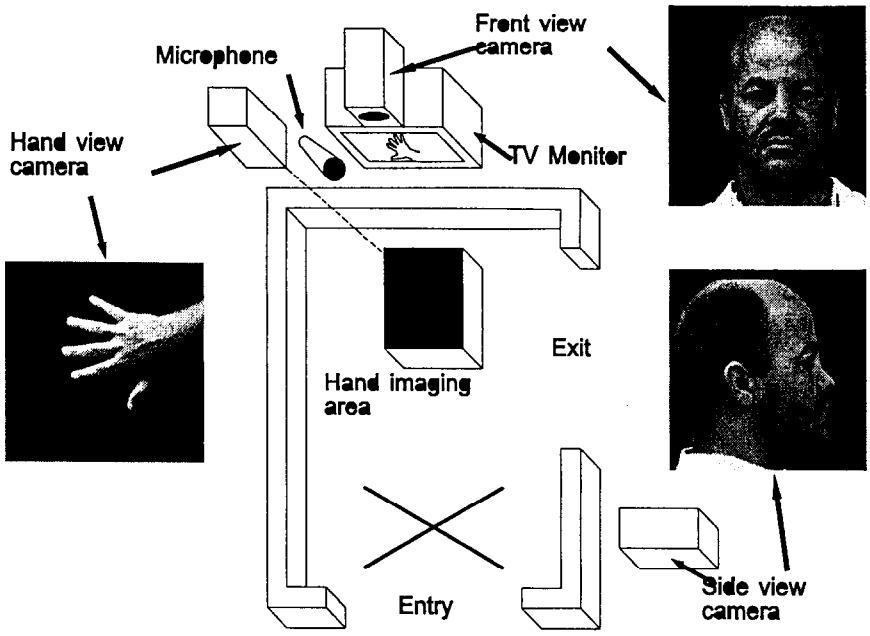


Figure 3–1. System configuration

3.1 Facial Features

For both the front and side view face images, the top of the head and the sides of the face are located using edge detection. This restricts the computations to the face regions and removes much of the positional variation from trial to trial. In the side view image, an edge detector is used to highlight and extract the profile of the front of the face. The extracted facial profile forms a one-dimensional signal for side view features. A typical side view profile signal is shown in Figure 3–2. The average gray level of each row in the front view image forms a one-dimensional signal for front view features. Since the two-dimensional image is projected along the rows onto a one-dimensional array, this front view signal is termed a gray level row projection of the image. A typical front face view row projection is shown in Figure 3–3. These signals from the side and front of the face contain considerable discriminating information. The side and front view cameras are not aligned exactly so the same features do not occur in corresponding rows. Because of the way features are extracted and fused from these signals, alignment of the cameras is not necessary, thus simplifying system setup.

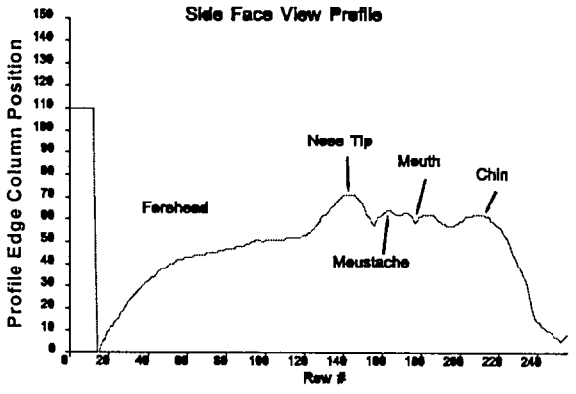


Figure 3–2. Signal for side view features

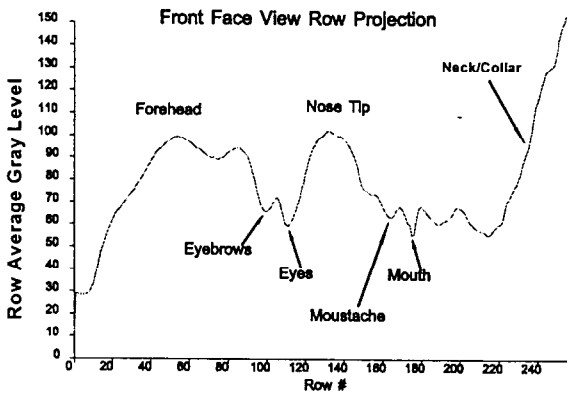


Figure 3–3. Signal for frontal view features

The feature discovery algorithms utilize the wavelet transforms of the two signals to extract potential features for use in the system. Initial experiments were performed on face side view images from a Sandia National Laboratories database containing 28 different people with 5 trials from each person. In early experiments, the full discrete wavelet transform (DWT) using the Daubechies[10] r=3 wavelet was applied to the profiles. Energy distributions and sample autocorrelations of the wavelet coefficients were examined. It was determined that only the coefficients representing the larger scale phenomena were useful in identity discrimination. The high pass portion of the DWT was not selected, since it mostly contained the measurement and environmental noise factors. The low-pass portion was replaced by a unit sample response derived from a windowed ideal low pass filter. The filter used is $h(n) = 0.0234375\delta(n-4) -0.046875\delta(n-3) -0.125\delta(n-2) +0.296875\delta(n-1) +0.703125\delta(n) +0.296875\delta(n+1) -0.125\delta(n+2) -0.046875\delta(n+3) +0.0234375\delta(n+4)$.

One way of assessing the quality of a feature set is by examining how well the set can reconstruct the object it represents. The full DWT of a signal using an orthonormal basis wavelet set has a unique inverse that is the original signal. Reconstructing the signal from only a few of the DWT coefficients results in only an approximation of the signal. Reconstruction of these facial signals using only the wavelet coefficients from the 2^3 level preserves most of the important structural and frequency features of the signals. The reconstructed signals corresponding to those shown in Figures 3–2 and 3–3 are shown in Figures 3–4 and 3–5 respectively.

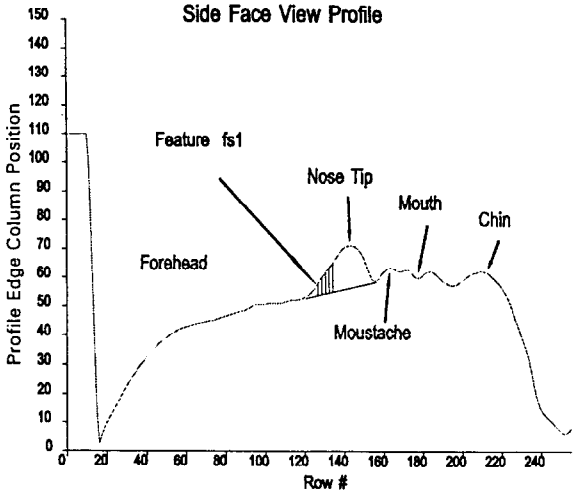


Figure 3–4. Reconstructed side view signal

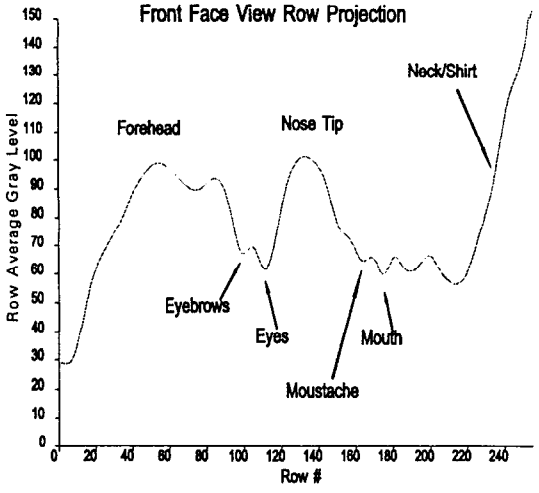


Figure 3–5. Reconstructed frontal view signal

The side view and front facial projection signals each have 256 components. In the side view image, a correction is made for the tilt of the head by finding the nose profile and rotating the profile so that the line under the nose shown in Figure 3–5 becomes horizontal. After correction, the wavelet coefficients are computed, and the feature discovery module selects 12 of the 32 coefficients from the 2^3 scale in the vicinity of the nose region to keep by the feature discovery algorithms as the side view features. The position of the nose in the image is also kept as a feature. This is a rough measure of a person's height. The top of the head position and eleven wavelet coefficients are kept as features from the front view projection. The wavelet coefficients from the top and bottom portions of the row projection are omitted because they fall in the highly variable hair and neck/collar regions respectively. These features are evaluated by the mpe statistic to determine their discriminating power. The ones that indicate the greatest discriminating ability are selected by the feature discovery module for use in the system.

The SNL and the early NMSU database images were used to develop the feature discovery algorithms and for preliminary feature evaluation. For final feature evaluation, these images were not used because they were obtained under controlled conditions and do not contain the levels of variation that would be expected in an operational system. Students in the NMSU Electrical and Computer Engineering Department acquired a large database of images using the prototype access control portal described earlier. These images contain the large variations that would be expected in an operational environment. Images from 25 different people in the new NMSU database are used in the evaluation of the feature discovery algorithms.

The facial features were evaluated by estimating the conditional probability density functions (PDFs) $f_X(\mathbf{x} \mid H_o)$ and $f_X(\mathbf{x} \mid H_a)$ for the features under the H_o and H_a hypotheses. Estimates of the PDFs were computed for each feature by using the nearest-neighbor minimal in-class (H_o) and the minimal out-class (H_a) distance statistics to compare measurements from one person to himself or herself and to compare measurements from different people. The PDF estimators are smoothed relative-frequency histograms of all combinations of measurements for each hypothesis. The MPE statistic is used to evaluate and rank the 32 facial features from the two

views. An MPE value 0 implies the feature completely separates the classes and an MPE value of 0.5 implies the feature provides no information to separate the classes.

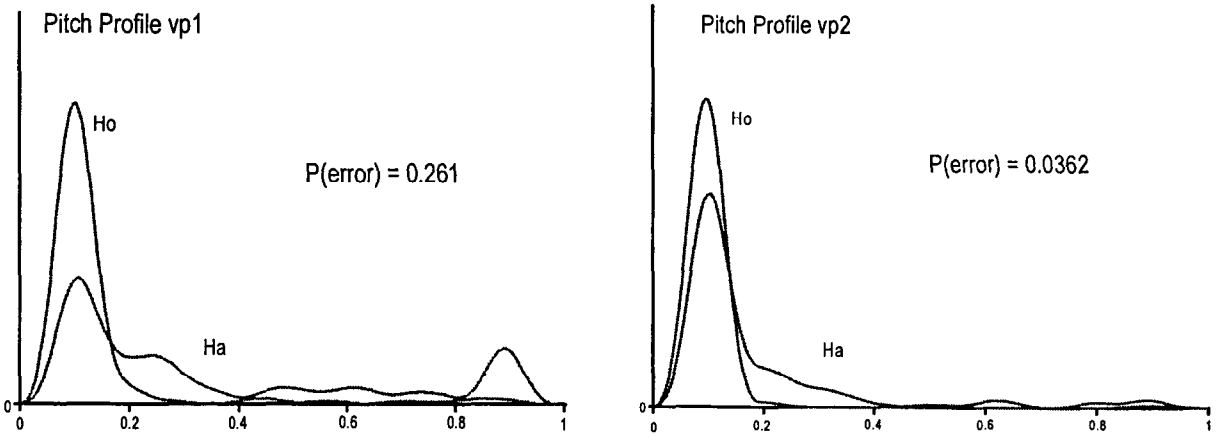


Figure 3–6. Complexities of selected individual facial features

The estimated PDFs for $f_X(\mathbf{x} \mid H_o)$ and $f_X(\mathbf{x} \mid H_a)$ for a side view feature and a front view feature together with the estimated probability of error for these individual features are shown in Figure 3–6. From these graphs, it is clear that these features have relatively high probabilities of error. The other features have similar graphs and are summarized in Tables 3–1 and 3–2.

Table 3.1. Side Facial Feature Performance

Feature (fs)	P(error)
0	0.46
1	0.38
2	0.39
3	0.31
4	0.27
5	0.31
6	0.28
7	0.34
8	0.41
9	0.22
10	0.41
11	0.27

Table 3.2. Front Facial Feature Performance

Feature (ff)	P(error)
0	0.37
1	0.28
2	0.42
3	0.40
4	0.26
5	0.31
6	0.36
7	0.30
8	0.35
9	0.45
10	0.39
11	0.30

The genetic optimization algorithm was used to weight the features to obtain a near-minimum probability of error when using the WNN fusion and classification algorithm. The weights increase the effects of good features and decrease the effects of the poor ones. After the minimization procedure, the WNN fusion algorithm uses the weights to account for the differences in the discriminating abilities of the features. The weights assigned to the front view features are {0.230, 0.010, 0.148, 0.400, 0.160, 0.128, 0.368, 0.142, 0.412, 0.038, 0.014, .114}. The weights assigned to the side view features are {1.000, 0.100, 0.386, 0.200, 0.211, 0.050, 0.121, 0.100, 0.227, 0.000, 0.208 ,0.029}. The features with the higher weights are more important in terms of minimizing misclassification error. The first feature—height—is given a large weight by the generic algorithm fusion algorithm. Figure 3–7 shows the dramatic improvement when the WNN fusion algorithm is used to fuse the front features and Figure 3–8 shows the performance of the fused side features. The fused front features perform better (pe = 0.116) than the fused side features (pe = 0.164).

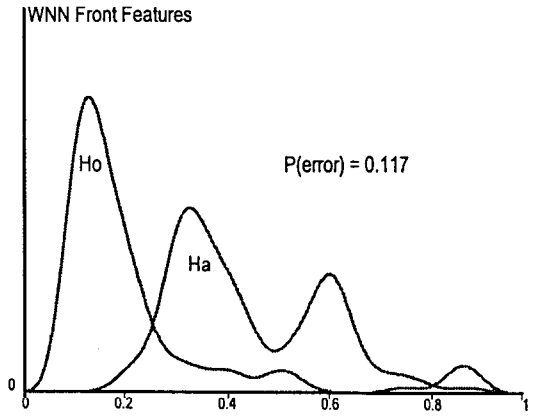


Figure 3–7. WNN-fused front facial features

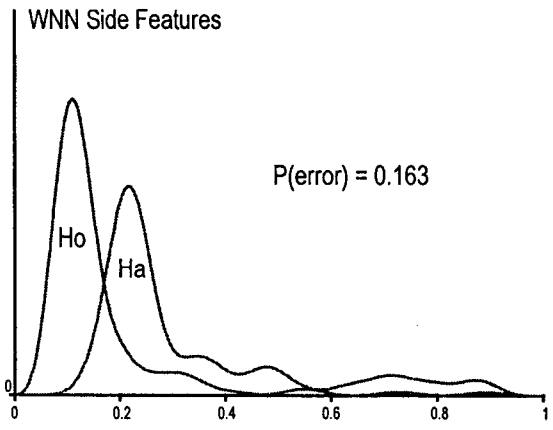


Figure 3–8. WNN-fused side-face features

3.2 Hand Features

Finger lengths and widths are relatively stable and easily computed measurements that provide coarse information about the structure of the hand. The lengths and widths of the four digits are extracted from the hand image using a combined process of segmentation, border tracking, and region filling. The segmentation is accomplished by taking the histogram of the entire image and setting a threshold according to the histogram. This threshold is used to make a binary image consisting of background and target. A finite state machine is then implemented to track the border of the segmented image to generate the hand curve. During this tracking stage, the maximal points of curvature are determined by locating the point where the angles of curvature are maximum.

The high curvature points are used to locate the tips of the fingers and the valleys between each pair. The valleys are marked in Figure 3–9 as points *b*,*c* and *d*. The line *cb* is extended to point *a* to isolate the index finger. The line *cd* is extended to *e* to isolate the little finger. The features of

the hand used in the recognition system are the length of the fingers, fl_i $(i=0,...,3)$, and the widths of the fingers fw_i $(i=0,...,3)$ also shown marked in Figure 3–9. The lengths are the distances from the midpoints of the line segments to the finger tips. The finger widths are measured at the midpoint between the finger tips and the finger valleys as the minimal distance across the finger as shown in Figure 3–9. The three points labeled *b*, *c*, and *d* form a triangle whose sides $hg_0=bc$, $hg_1=cd$, and $hg_2=bd$ are also used as features, forming an 11-dimensional hand-feature space. The side *bd* provides a measure of hand width, and the sides provide additional hand distinguishing characteristics.

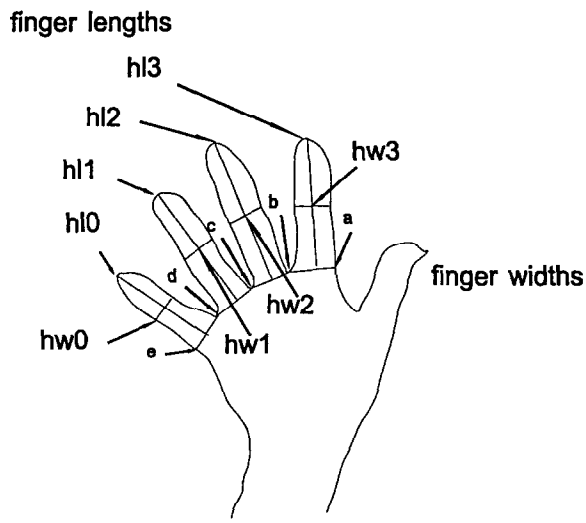


Figure 3–9. Hand geometry features

The new NMSU hand image database was used to study the effectiveness of these hand features. The hand database includes ten hand images of 25 different people. As with the features from the face images, the hand features are individually analyzed with the MPE statistic to establish their effectiveness in hand recognition. The performance of typical individual hand features are shown in Figure 3–10. A summary of the probability of error evaluation of the hand features is given in Table 3–3. These results indicate that all the individual hand features have considerable probability of error.

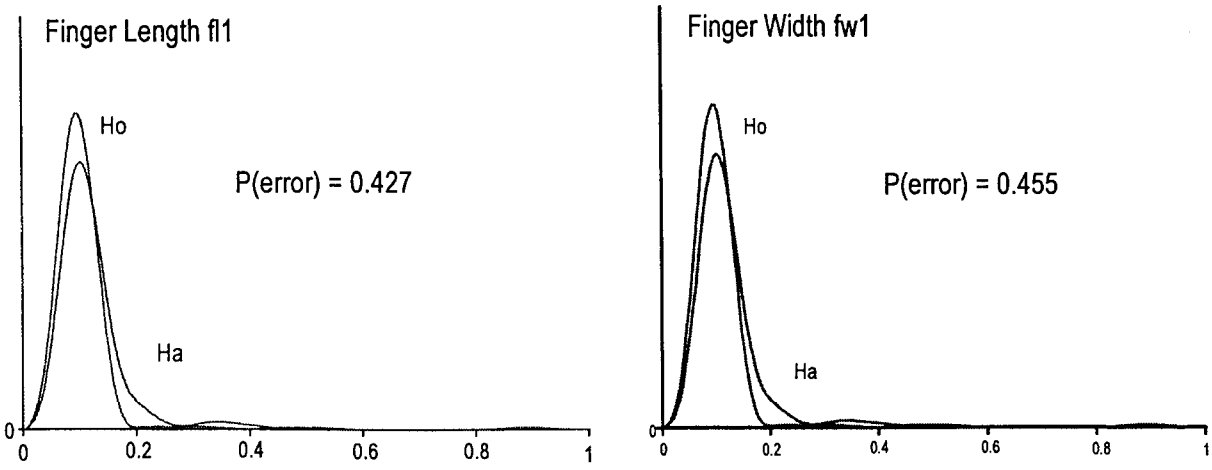


Figure 3–10. Performance of selected individual hand features

Table 3–3. Hand Feature Performance

Feature	P(error)
hl0	0.42
hl1	0.45
hl2	0.45
hl3	0.45
hw0	0.45
hw1	0.38
hw2	0.40
hw3	0.35
hg0	0.33
hg1	0.38
hg2	0.39

The genetic algorithm was used to select the weights using the WNN fusion model to obtain a near-minimum probability of error performance. The weights for the hand features were determined to be {0.750, 0.370, 1.000, 0.985, 0.950, 0.542, 0.700, 0.210, 0.493, 0.225, 1.000}. The performance of the WNN fusion process is shown in Figure 3–11. This indicates that the

expected probability of error of the fused hand features is pe=0.093, which again shows the performance improvement by using the fusion process.

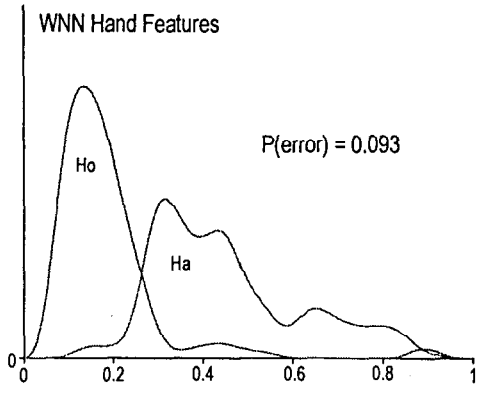


Figure 3–11. WNN-fused hand features

3.3 Voice Features

Since most of the vocal apparatus is consciously controllable, it is difficult to obtain features from the voice that reliably distinguish individuals. The portion of the vocal tract that is not under conscious control by the individual is the nasal tract. This portion is subject to significant variations with health (allergies, colds, etc.). Rather than attempting to find a few intricate vocal features that are invariant with respect to health and conscious effort, many simple features are used that are habitual in nature. These features are useful because it is difficult to control many habitual features simultaneously. That is, out of habit, the person vocalizes certain phonemes and combinations of phonemes in consistent and predictable ways. These vocalizations can be altered but only with much conscious effort. When many such features are used, it is very difficult for an impostor to alter enough of them in a sufficiently controlled fashion to mimic the voice of another person.

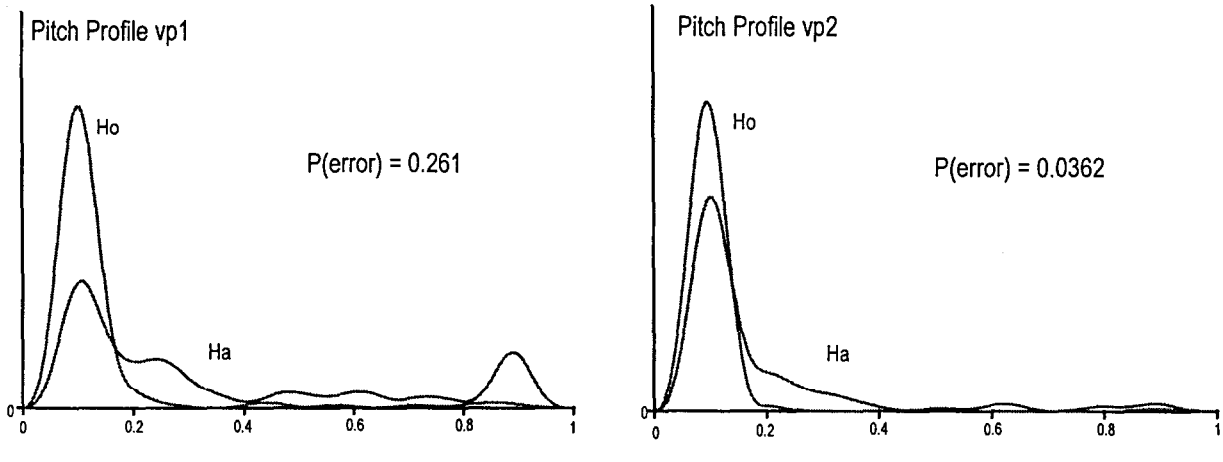


Figure 3–12. Performance of selected individual voice features

For this study, several experiments on isolated digits from *zero* to *nine* indicated that the vowel, diphthong and nasal phonemes in the combination *three-nine-five* carried considerable information for distinguishing individuals. The simple voice features extracted from these signatures are the pitch frequency profile and the frequency spectrum. The pitch frequency varies as one speaks the numbers and this variation is a good speaker-verification characteristic. The frequency spectrum for each number also provides significant information to identify speakers.

A digital correlator is used to discover the presence of near uniformly spaced pitch pulses to determine the location of the voiced phonemes. The pitch frequency is tracked before and after the center of the voiced numbers, forming a pitch-frequency profile signature. This pitch-frequency profile provides significant speaker recognition features. The profile is reduced by the feature discovery module to an average pitch frequency and five features that represent the beginning, middle and ending of the profile (vpi,i=0,...,5). In addition to finding the pitch frequency profile features, a fast Fourier transform is used to establish the frequency spectrum of the middle segment of the voiced numbers. The feature discovery module reduces the frequency spectrum to eight features $(vs_i, i=0,...,7)$, representing the major peaks in the spectrum.

The frequency profile and spectrum features give an indication of the habitual speaking characteristics and prosody (rhythm) of the speaker. Fourteen features are measured for each voiced number. Consequently, a total of 42 features are measured for the three-number sequence *three-nine-five.* Performance of the average-pitch frequency for the number *three* and the front-porch of the number *three* frequency profile are given in Figures 3–12a and 3–12b. As with the coarse features from the face and hand, each of the voice features individually is not sufficiently reliable for identity verification. However, after the *WNN* algorithm was used to fuse the forty-two features, a probability of error pe=0.147 was obtained as shown in Figure 3–13.

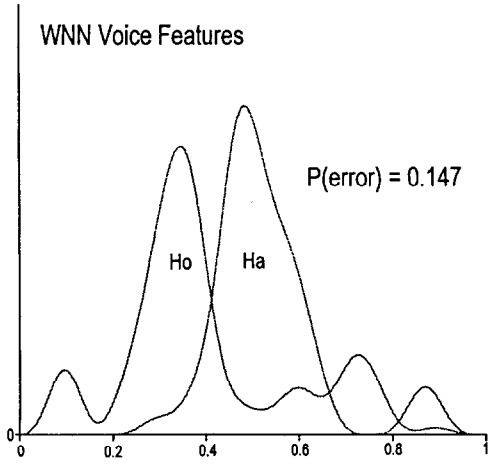


Figure 3–13. WNN fused voice features

3.4 Fusion of Facial, Hand, and Voice Features

The coarse features from the face, hand and voice have demonstrated the ability to distinguish people with a probability of error of less than 1 percent. Groups of features taken together tend to lower the probability of error. An important research problem addressed here is to establish a method for combining some or all of the available features to improve the error performance without greatly increasing the computational cost. This problem is called the feature fusion problem. The approach to this problem presented here involves selecting reliable, simple and nearly independent features and using a near optimal algorithm to fuse all selected features to solve the discrimination problem.

Since the hand and front facial features performed the best an augmented feature vector is assembled that contains the hand and facial front features (23 in all). The genetic optimization algorithm and the WNN fusion model described earlier is used to weight the features to obtain a near-minimum probability of error. The performance of the combined hand and facial front features is shown in Figure 3–14 with the hand features weighted by {0.800, 0.370, 1.000, 1.000, 1.000, 0.492, 0.750, 0.260, 0.143, 0.225, 0.850} and the front features weighted by {0.000, 0.050, 0.150, 0.050, 0.100, 0.000, 0.000, 0.050, 0.150, 0.200, 0.100, 0.000}. The probability of error was significantly reduced from the hand performance (pe_hand=0.093) and the facial front performance (pe_front=0.116) to a combined performance of pe_hf=0.042). The hand, facial front, and facial side features are also fused together using the WNN fusion algorithm. The fused hand-feature weights are {0.900, 0.520, 1.000, 0.200, 1.000, 0.442, 0.250, 0.160, 0.143, 0.075, 0.700}; the fused front feature weights are {0.000, 0.050, 0.350, 0.150, 0.200, 0.000, 0.000, 0.050, 0.150, 0.200, 0.100, 0.000}; and the fused side weights are {1.000, 0.000, 0.000, 0.050, 0.050, 0.000, 0.000, 0.050, 0.000, 0.000, 0.000, 0.000}. The results are shown in Fig. 4–15. Again the probability of error was significantly reduced from the combined hand/front performance (pe_hf=0.042) to pe_hfs=0.013. All the features from the front, hand, side, and nine of the most significant voice features are fused to form a 44-dimensional space to obtain the performance pe_fhsv=0.008 as shown in Figure 3–16. The nine voice features selected are {vp0, vp4, vp5} from the pitch profile of the number *three*, {vp0, vp4, vp5, vs7} from the number *nine* and {vp4, vp5} from the number *five*. The fused weights for the hand features are {1.000, 0.620, 1.000, 0.200, 0.950, 0.442, 0.250, 0.160, 0.043, 0.125, 0.750}, the fused front feature weights are {0.000, 0.000, 0.300, 0.300, 0.200, 0.000, 0.000, 0.050, 0.100, 0.200, 0.150, 0.000}, the fused side weights are {1.000, 0.000, 0.000, 0.050, 0.050, 0.000, 0.000, 0.050, 0.000, 0.050, 0.000, 0.000} and the fused voice weights are {0.050, 0.000, 0.000, 0.250, 0.050, 0.000, 0.000, 0.100, 0.000}. To illustrate the effectiveness of the feature-weighting process, all features were given equal weights, and the WNN fusion process resulted in a probability of error of pe=0.068, as shown in Figure 3–17. This clearly shows the power of the weighting process.

The hand and facial side features are also fused together using the WNN fusion algorithm. The results are shown in Figure 3–15. Again the probability of error was significantly reduced from the combined front/hand performance (pe_fh=0.028) to pe_fhs=0.019. All the features from the front, hand, side and voice are fused to form a 44 dimensional space to obtain the performance pe_fhsv=0.016 as shown in Figure 3–16. To demonstrate the effectiveness of the feature weighting process, all features were given equal weights and the WNN fusion process resulted in

a probability of error of pe=0.144, as shown in Figure 3–17. This clearly shows the power of the weighting process.

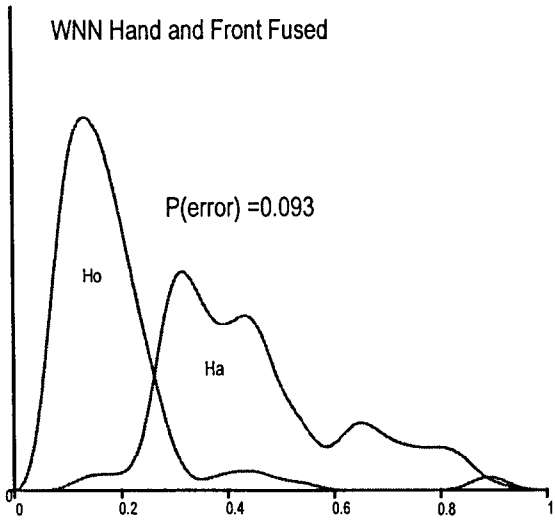


Figure 3–14. WNN-fused hand/front

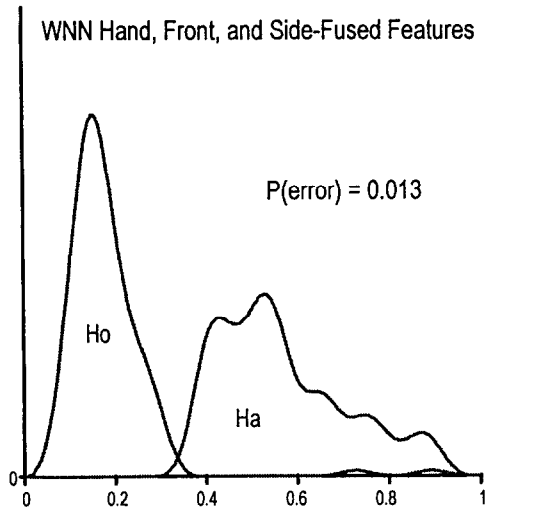


Figure 3–15. WNN-fused hand/front/side

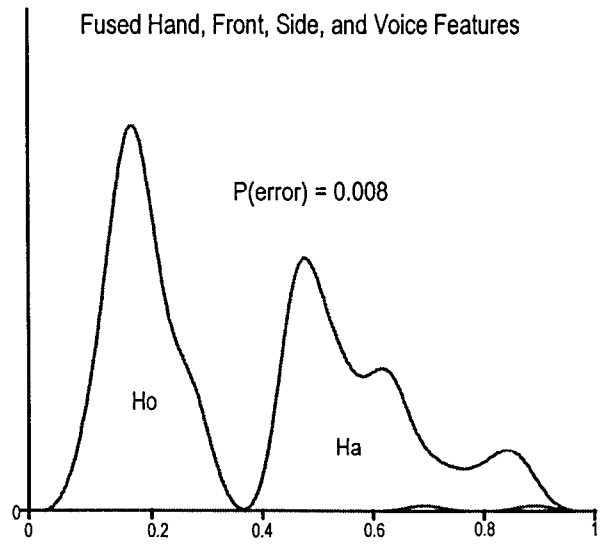


Figure 3–16. WNN hand/front/side/voice

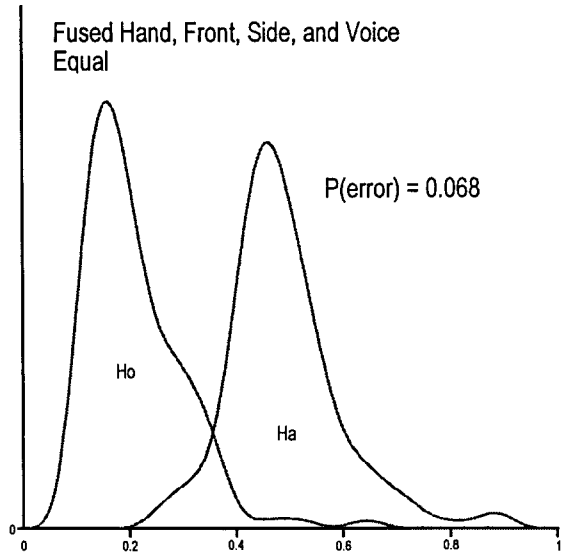


Figure 3–17. Fused with equal weights

The WNN method is demonstrated to be a good tool for feature fusion. A well designed system can accurately verify the identity of individuals by fusing numerous simple biological features extracted from remotely acquired digital images and voice data. Although the individual simple

features do not provide quite the performance of some of the more complex human biometrics features, the time to extract the 44 simple features is significantly less than that required for some of the more complex features. Furthermore, the results demonstrate the important theorem in feature fusion that the performance of an optimal feature fusion system will not degrade with the addition of new features and may improve.

For comparison purposes an artificial neural network was trained to fuse the features. Once trained the neural network's performance was similar to the WNN fusion model. The main disadvantage of the neural network fusion was the very long training time required each time a new person was added to the database. The training time for the WNN involves selecting the training samples and determining the feature weights to obtain a near-minimal probability of error.

4. CONCLUSIONS

The results support the conclusion that the identity of individuals can be reliably verified by using a large number of relatively simple features extracted from digital images of the face and hand and from digitized speech. The performance of each feature is established in terms of its probability of error. Individually, the coarse features have a relatively high probability of error. However, when fused together in a higher dimensional space their combined performance is much improved. The basic idea is that a few features can be easily confused but it is difficult to confuse a large number of features from different sensors simultaneously. This concept is illustrated by using the WNN method to fuse and evaluate the joint performance of the features.

The use of data acquired under normal operating conditions was of great help in evaluating the robustness of the features chosen for the experiment and the feature extraction algorithms. Feature performance is closely related to the ability of the extraction algorithms to reliably measure the features. Errors in the measurement of the features cause decision errors. However, the fusion of many features reduces the effect of any given feature measurement error. For example, the hand feature extraction algorithm performed very well on subjects whose fingers were spread apart. It often failed when two or more fingers were touching. The hand feature extraction algorithm was modified to correctly detect this failure mode and prompt the person to spread their fingers. This modification significantly enhances the performance of the system. Several of the subjects tried to foil the system with some success by moving their hands very quickly through the scanning area or making unnatural faces. The digitized hand images were somewhat blurred and often their fingers were out of the scanning area before all calculations were complete. Uncooperative subjects accounted for many of the system errors. These subjects can also be seen in the H_0 probability of density functions as minor modes under the H_a probability of density functions. The voice feature extraction algorithm also performed well on some subjects but not on others. The major cause of the failures in the voice feature extraction was determined to be variation in loudness from person to person. Perhaps a better microphone with a better dynamic range and better signal-to-noise ratio would significantly improve the performance of the voice recognition.

5. REFERENCES

1. D. R. Scott, The k-nearest neighbor statistic with applications to electronic vision systems, Ph.D. dissertation, New Mexico State University, December 1990.

2. T. Cover and P. Hart, *Nearest Neighbor Pattern Classification*, IEEE Trans. on Information Theory, 1963.

3. McClave and Scheaffer, *Probability and Statistics for Engineers*, Duxbury Press, 1995.

4. Cynthia Beer, The Tie Statistic and Texture Recognition, Ph.D. dissertation, New Mexico State University, December 1989.

5. G. M. Flachs, Qiang Meng, Xiaohua Niu, Wei Wang and Zhonghao Bao, *Entry Control Project Final Report* for Sandia National Laboratories contract, New Mexico State University, 1994.

6. J. Carlson, J. Jordan, G. Flachs, Q. Meng, X. Niu, W. Wang, Z. Bao, D. Marten, "High Confidence Identity Verification Using Multiple, Coarse Features Acquired from the Face, Hand and Voice," *Proceedings of the 11th Annual Security Technology Symposium,* Virginia Beach, Virginia, June 19–22, pp. 249–256, 1995.

7. H. T. Nguyen and G. S. Rogers, *Fundamentals of Mathematical Statistics*, Vol. II Statistical Inference, New York: Springer-Verlag, 1989.

8. A. M. Mood, F. A. Graybill, and D. C. Boes, *Introduction to the Theory of Statistics*, New York: McGraw-Hill, 1974.

9. A. Papoulis, *Probability, Random Variables and Stochastic Processes,* New York: McGraw-Hill, 1984.

10. R. V. Hogg and A. T. Craig, *Introduction to Mathematical Statistics*, New York: Macmillan, 1978.

11. Jeffrey J. Carlson, Decision-Making Complexity with Applications in Electronic Vision, Ph.D. dissertation, New Mexico State University, September 1988.

12. Cynthia Beer, The Tie Statistic and Texture Recognition, Ph.D. dissertation, New Mexico State University, December 1989.

13. Ingrid Daubechies, "Orthonormal Bases of Compactly Supported Wavelets," *Communications on Pure and Applied Mathematics,* Vol. 41, No. 7, pp. 909–996, 1988.

14. G. M. Flachs, J. B. Jordan, C. L. Beer, and D. R. Scott, "Feature Space Mapping for Sensor Fusion," *Journal of Robotic Systems*, Vol. 7, No. 3, pp. 373-393, June, 1990.

15. Howon Choe, A Comparative Analysis of Statistical, Fuzzy, and Artificial Neural Pattern Recognition Techniques, Ph.D. dissertation, New Mexico State University, December 1992.

16. J. B. Jordan and H. Choe, "A comparative analysis of statistical, fuzzy, and artificial neural pattern recognition techniques," *Proceedings of SPIE Signal Processing, Sensor Fusion, and Target Recognition*, Vol. 1699, pp. 166–176.

17. N. I. Johnson, and S. Kotz, *Continuous Univariaate Distributions-1*, Boston: Houghton Mifflin, 1970.

Distribution

10	Jeffrey Carlson, 5838	MS 0780
1	Ann Bouchard, 5838	0780
1	Steve Ortiz, 5838	0780
1	Gordon Osbourn, 1155	MS 1423
1	Rubel Martinez, 1155	1423
1	John Bartholomew, 1155	1423
1	Dennis Miyoshi, 5800	MS 0769
1	Basil Steele, 5804	MS 0768
1	Judy Moore, 6234	MS 0449
2	Chuck Meyers, 4523	MS 0149
1	Central Technical Files, 8940-2	MS 9018
2	Technical Library, 4916	MS 0899
2	Review & Approval Desk, 12690 For DOE/OSTI	MS 0619