3-15-2017

Using Gaussian Mixture Model and Partial Least Squares regression classifiers for robust speaker verification with various enhancement methods

Joshua Scott Edwards
Rowan University

USING GAUSSIAN MIXTURE MODEL AND PARTIAL LEAST SQUARES REGRESSION CLASSIFIERS FOR ROBUST SPEAKER VERIFICATION WITH VARIOUS ENHANCEMENT METHODS

by

Joshua Scott Edwards

A Thesis

Submitted to the
Department of Electrical and Computer Engineering
College of Engineering
In partial fulfillment of the requirement
For the degree of
Master of Science in Electrical and Computer Engineering
at
Rowan University
August 31, 2016

Thesis Chair: Ravi. P. Ramachandran, Ph.D.

Acknowledgments

I would like to thank Dr. Ravi Ramachandran for working with me for all these years as this research came to fruition. I would also like to thank Dr. Umashanger Thayasivam and Dr. John Schmalzel for their assistance as members of my Thesis Defense Committee. Additionally, I would like to thank Russell Trafford for sticking by my side as we faced the hardships of graduate school, and for showing me all of the best tech to make writing a paper like this as painless as possible. Finally, I would like to thank my father for providing more love and support than one could reasonably hope for. My respect and appreciation for him knows no bounds.

This work was supported by the National Science Foundation via the TUES Type 2 Research Grant DUE-1122296.

Abstract

Joshua Scott Edwards
USING GAUSSIAN MIXTURE MODEL AND PARTIAL LEAST SQUARES
REGRESSION CLASSIFIERS FOR ROBUST SPEAKER VERIFICATION WITH
VARIOUS ENHANCEMENT METHODS
2015-2016
Ravi P. Ramachandran, Ph. D.
Master of Science in Electrical and Computer Engineering

In the presence of environmental noise, speaker verification systems inevitably see a decrease in performance. This thesis proposes the use of two parallel classifiers with several enhancement methods in order to improve the performance of the speaker verification system when noisy speech signals are used for authentication. Both classifiers are shown to receive statistically significant performance gains when signal-to-noise ratio estimation, affine transforms, and score-level fusion of features are all applied. These enhancement methods are validated in a large range of test conditions, from perfectly clean speech all the way down to speech where the noise is equally as loud as the speaker. After each classifier has been tuned to their best configuration, they are also fused together in different ways. In the end, the performances of the two classifiers are compared to each other and to the performances of their fusions. The fusion method where the scores of the classifiers are added together is found to be the best method.

Table of Contents

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

List of Figures

Figure Page

List of Tables

Table Page

Chapter 1

Introduction

1.1 Problem Statement

The use of speech in determining or verifying a person's identity is a promising domain of study with many applications, ranging from law enforcement to consumer-level security. In ideal conditions, speech-based verification would take place in a location with low levels of environmental noise, and be performed with high-quality audio capture devices that introduce minimal distortion to the speech signals. While high fidelity microphones are becoming more commonplace, so too are users given more potential to use these devices in locations and situations where noise levels far exceed the ideal.

In such scenarios, the performance of any speaker classification system will inevitably degrade. The addition of any type of noise causes a mismatch between the utterance that is being tested and the model that is stored for a speaker in the speaker verification system. Given a mismatch of a high enough degree, the speaker verification system will be unable to correctly verify the identity of the person from whom the test utterance originated. However, methods such as relative spectral filtering and feature warping have been developed as ways of compensating for the effects of additive noise at the stage of audio capture [1]. This thesis examines the use of two different types of speaker verification systems with multiple enhancement methods as a way to counter this problem.

1.2 Motivation

In conducting this research, two unique speaker verification systems were developed; namely, the Gaussian Mixture Model – Universal Background Model (GMM-UBM) system and the Gaussian Supervector – Partial Least Squares (GSV-PLS) classifier. These two systems were chosen for various reasons. First, the GMM-UBM system “has become the *de facto* reference method in speaker recognition” [2]. Second, fusion between generative (such as GMM-UBM) and discriminative (such as GSV-PLS) speaker verification methods has been shown to work well in past research [3]. Finally, Gaussian supervectors emerge naturally as a corollary to the GMM-UBM system, and PLS regression has been shown to be a promising method of utilizing these GSVs in a framework for speaker verification [2] [4].

The performances of these two systems for classifying speakers using speech corrupted with additive white Gaussian noise (AWGN) at signal-to-noise ratios (SNR) ranging from 0 dB to 30 dB were observed and compared. Two enhancement methods were applied to each of the systems – SNR estimation in conjunction with affine transforms, and score-level feature fusion. The selection of available affine transforms for use in test-to-training utterance mismatch reduction was viewed as a key parameter in training these speaker verification systems. Multiple of these so-called “affine resolutions” were studied in order to determine the best resolution to use for each system. Three different sets of fusion scores were generated for each system and compared with the other feature types in order to determine the most effective feature at each SNR. Each speaker verification system trained using the optimal affine resolution and feature type is considered to be the in “best configuration” for the experiments performed in this thesis.

1.2 Objectives of Thesis

The primary objectives of this thesis are:

1. *To implement both a GMM-UBM and GSV-PLS system for speaker verification.*

2. *To enhance the performance of the speaker verification systems using SNR estimation, affine transforms, and score-level fusion of feature vectors.*

3. *To investigate the effect of the "affine resolution" parameter in supplementing the robustness of the speaker verification systems.*

4. *To identify the best performing feature or fusion of features in the presence of various SNR noise-levels.*

5. *To perform a full classifier fusion of GMM-UBM and GSV-PLS in their best configurations.*

6. *To analyze the performances of each classifier and their fusion to determine whether there are statistically significant differences.*

1.3 Focus and Organization

The focus of this thesis compares two classifiers whose performances have been made more robust by the application of various enhancement methods. After examining each of the classifiers individually, the optimal configurations for each classifier were compared to each other as well as a fusion of the two systems. By analyzing the performance of these two systems – both separately and in conjunction – a recommendation is made for how to obtain robust performance in the presence of additive noise in a speech signal. The thesis is organized as follows:

Chapter 1 is an introduction, outlining the need for robust speaker verification systems and the contributions to the practice of speaker recognition made by this thesis.

Chapter 2 is a literature review and overview of all of the methods and concepts that have been utilized in the production of this thesis. This chapter contains derivations of the features that were extracted from speech utterances and whose performances at a various SNR levels were compared to one another. Also within Chapter 2 is an explanation of the two types of classifier systems whose performance was compared, GMM-UBM and PLS-GSV. Finally, the chapter contains definitions of the enhancement methods that were applied to each classifier in order to supplement their capabilities.

Chapter 3 outlines the approach taken to determine key parameters of each of the classifiers that were compared. Additionally, the methodology that was undertaken in training and testing the two classifier systems, applying the proposed enhancement methods, and analyzing system performance are expounded.

Chapter 4 presents the results obtained from each constituent of the proposed methodology. Figures and tables are included to help elucidate the significance of the results.

Chapter 5 concludes the thesis with an examination of the results that were obtained in addition to a recommendation for future work.

Chapter 2

Background

This chapter contains all of the necessary background concepts that are needed to understand the approach taken in this thesis. It begins with a complete overview of all of the feature vectors that are extracted in the process of training and testing the speaker verification system. The feature extraction process used in both classifiers is explained as well.

Next, an overview of speaker verification systems is given. This subchapter contains information about how the performance of a speaker verification system can be evaluated using different error rates. The way these error rates are calculated is explained and the effects of these error rates are discussed.

Both of the classifier types that are used in this thesis are defined and the procedure for developing them is explicated. For the GMM-UBM system, both Gaussian mixture models and the concept of universal background models is explained. Both of these topics appear again in the development of the GSV-PLS system, so the GMM-UBM system is introduced first.

Expanding upon the Gaussian mixture model concept, the GSV-PLS subsection of this chapter introduces Gaussian supervectors and explains how they can be used in a framework for speaker verification that utilizes the discriminative capabilities of PLS regression. The mathematical formulation of the PLS regression framework is also provided.

The chapter then introduces the various enhancement methods that are used in the proposed methodology of this thesis. A discussion about using vector quantization (VQ) codebooks to perform SNR estimation is included. Additionally, an explanation of how to train the affine transforms used to reduce training to testing mismatch at each of the estimated SNR levels is provided.

Finally, this chapter concludes with an examination of the statistical methods that are used to validate the results obtained through the proposed methodology.

2.1 Features

Feature extraction is an important step in speech processing that allows for the discerning of information stored within a speech signal that can be used to uniquely identify an individual. By intelligently using different digital signals processing techniques, features containing information about the physiological characteristics of an individual can be filtered out of a captured speech signal and used for classification in speaker recognition systems.

Of the four features that were chosen for the proposed methodology, three of them are based on linear prediction – namely, linear prediction cepstral coefficients (CEP), postfilter cepstrum (PFL), and adaptive component weighting cepstrum (ACW). The fourth feature, the mel-frequency cepstral coefficients (MFCC), is based on the application of a non-uniformly spaced bandpass filter bank corresponding to the mel frequency scale [2]. Each of the features will be discussed in-depth subsequently.

2.1.1 Linear prediction. The concept of linear prediction (LP) is defined by the notion that a signal can be approximated as a weighted combination of previous time-domain samples [5] [6]. To calculate a speech signal, *s(n)*, the equation used is,

$$s(n) = \sum_{k=1}^{P} a(k)s(n-k) + e(n)$$

(2.1)

where *P* is the LP order, which defines the number of previous samples being used to calculate the current sample, *a(k)* is the set of LP coefficients, and *e(n)* is the prediction error.

In speech processing applications, the LP coefficients, *a(k)*, need to be calculated from the incoming speech signal. Finding a set of LP coefficients by using the entire speech signal and a high LP order would be an insufficient way to accurately approximate the speech signal, so instead a lower LP order is used and coefficients are calculated for many short frames over the duration of the speech signal [6]. The autocorrelation method for linear predication is typically used, which is accomplished by minimizing the total mean-squared (L2) prediction error, *E*, in the equation,

$$E = \sum_{n=-\infty}^{\infty} e^2(n) = s \sum_{n=-\infty}^{\infty} \left[s(n) - \sum_{k=1}^{P} a(k)s(n-k) \right]^2$$

(2.2)

where the signal, *s(n)*, is not assumed to be causal, $e^2(n)$, is the L2 prediction error, and *E* is a function of all LP coefficients, *E(a(1), a(2), … a(P))*.

Simplifying Equation 2.2 results in,

$$E = \sum_{n=-\infty}^{\infty} s(n) - 2 \sum_{n=-\infty}^{\infty} \sum_{k=1}^{P} a(k)s(n-k)s(n)$$
$$+ \sum_{k=1}^{P} a(k) \sum_{j=1}^{P} a(j) \sum_{n=-\infty}^{\infty} s(n-k)s(n-j)$$

(2.3)

The term of two summations in Equation 2.3 can be rewritten in vector form as follows:

$$2 \sum_{n=-\infty}^{\infty} \sum_{k=1}^{P} a(k)s(n-k)s(n) = 2[a(1), a(2), \ldots a(P)] \begin{bmatrix} \sum_{n} s(n-1)s(n) \\ \sum_{n} s(n-2)s(n) \\ \vdots \\ \sum_{n} s(n-P)s(n) \end{bmatrix} = 2a^T d$$

(2.4)

The term of three summations in Equation 2.3 can also be rewritten, this time as two vectors and a matrix. This operation results in the following simplification,

$$\sum_{k=1}^{P} a(k) \sum_{j=1}^{P} a(j) \sum_{n=-\infty}^{\infty} s(n-k)s(n-j) = [a(1), \ldots a(P)][\phi] \begin{bmatrix} a(1) \\ \vdots \\ a(P) \end{bmatrix} = a^T \phi \, a$$

(2.5)

where ϕ is defined as the $P \times P$ Toeplitz autocorrelation matrix. It is assumed that every speech sample outside the frame for which the LP coefficients are being calculated is equal to zero [6]. With this assumption, each element of the matrix, ϕ, can be calculated as,

$$\phi(k,j) = \sum_{n=0}^{N-1-|k-j|} s(n-k)s(n-j)$$

(2.6)

where $1 \leq k, j \leq P$, and N is the size of the sample window [5]. By manipulating Equation 2.6 via substitution as follows, it can be shown that this is equivalent to the autocorrelation of the signal, *s(n)* [6],

$$Let\, u = n - k$$

$$\phi(k,j) = \sum_{u=0}^{N-1-|k-j|} s(u)s(u+(k-j))$$

$$Let\, m = k - j$$

$$\phi(k,j) = R(m) = \sum_{n=0}^{N-1-m} s(n)s(n+m) = \sum_{n=0}^{N-1-m} s(n)s(s-m) = R(-m)$$

(2.7)

With Equation 2.7, the vector, *d*, from Equation 2.4 and the matrix, ϕ, from Equation 2.5 can be simplified to,

$$d = \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(P) \end{bmatrix}$$

$$\phi = \begin{bmatrix} R(0) & R(1) & R(2) & \cdots & R(P-1) \\ R(1) & R(0) & R(1) & \cdots & R(P-2) \\ R(2) & R(1) & R(0) & \cdots & R(P-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R(P-1) & R(P-2) & R(P-3) & \cdots & R(0) \end{bmatrix}$$

(2.8)

Thus, what remains is the equation for the total L2 prediction error, written in terms of *s(n)*, *a*, *d*, and ϕ.

$$E = \sum_{n=0}^{N-1} s^2(n) - 2a^T d + a^T \phi\, a$$

(2.9)

To minimize this equation, the gradient with respect to *a* must be taken and set equal to zero. Not that the gradient of the first term in Equation 2.9 is equal to zero and need not be included here.

$$\frac{\partial E}{\partial a} = -2d + 2\phi a = 0$$

$$\phi a = d$$

(2.10)

All that remains is a system of equations to solve for the LP coefficients. These LP coefficients correspond to the coefficients of an IIR filter whose all-pole transfer function is given by [6],

$$H(z) = \frac{1}{A(z)} = \prod_{k=1}^{P} \frac{1}{1 - z_k z^{-1}}$$

(2.11)

The poles of this transfer function are expressed as,

$$z_k = \sigma_k e^{j\omega_k}, \quad k = 1, 2, \dots, P$$

(2.12)

Using the Levinson-Durbin algorithm to solve for the LP coefficients in Equation 2.10 is a computationally efficient way to determine the weights of each speech frame and guarantees that all of the poles of *H(z)* are found within the unit circle [6].

2.1.2 Linear prediction cepstral coefficients (CEP). The cepstrum of a signal is defined as the inverse Z-transform of the natural logarithm of the Z-transform of the speech signal [6],

$$c_s(n) = Z^{-1}\{\ln(S(z))\}$$

(2.13)

where $c_s(n)$ are the cepstral coefficients at each quefrency, and *S(z)* is the Z-transform of the speech signal, *s(n)*. For the case of an LP filter, *1/A(z)*, the causal LP cepstrum can be given as,

$$c_{LP}(n) = \begin{cases} 0, & n \leq 0 \\ \frac{1}{n}\sum_{k=1}^{P} z_k^n , & n > 0 \end{cases}$$

(2.14)

where z_k are the poles of the LP filter and P is the order of the LP filter [6]. The LP cepstral coefficients can be found in a computationally efficient way using the following recursive relationship between the LP cepstral coefficients and the LP predictor coefficients:

$$c_{LP}(n) = \begin{cases} a(n) + \sum_{k=1}^{n-1} \left(1 - \frac{k}{n}\right) a(k) c_{LP}(n-k), & 1 \le n \le P \\ \sum_{k=1}^{n-1} \left(1 - \frac{k}{n}\right) a(k) c_{LP}(n-k), & n > p \end{cases}$$

(2.15)

Note that in this equation, *a(k)* = 0 for *k* > *p.* As the duration of the LP cepstrum is infinite, usually the first *P* coefficients are taken as the feature vector. For a sufficiently large LP order, the L2 difference between a cepstrum of order *P* versus one of order *P+1* is insignificant given the decay caused by increasing *n* [6].

2.1.3 Postfilter cepstrum (PFL/PST). The postfilter cepstrum is based on a pole-zero transfer function that was designed to emphasize the peaks of the formants of a speech signal. The rationale behind this is that noise is less perceptually damaging to a speech signal in the formant regions; hence, by emphasizing these regions, the LP cepstrum extracted from the postfiltered speech should be less susceptible to error in the presence of noise [7]. The transfer function of this filter is given as,

$$H_{pf}(z) = \frac{A(\frac{z}{\beta})}{A(\frac{z}{\alpha})}, \;\; 0 < \beta < \alpha \le 1$$

(2.16)

where 1/*A(z)* is the all-pole LP transfer function, and both *β* and *α* are scaling factors for the poles of the LP filter. The cepstrum of the filter defined by the transfer function, H_{pf}, is the postfilter cepstrum (PFL/PST) that is implemented as one of the feature vectors under test in the two speaker verification systems of this thesis. To compute the postfilter cepstrum, one can simply scale the LP cepstral coefficients using the following equation:

$$c_{PFL}(n) = c_{LP}(n)[\alpha^n - \beta^n]$$

(2.17)

In the speaker verification systems implemented in this thesis, the values used to scale the LP cepstrum were α = 1 and β = 0.9.

2.1.4 Adaptive component weighted cepstrum (ACW). The adaptive component weighted cepstrum (ACW) is another feature, like PFL, that was designed to improve the robustness of speaker recognition systems in the presence of noise. It does this by normalizing the residues of the all-pole LP filter by setting them equal to 1 in order to remove variations caused by channel effects [6]. The pole-zero transfer function of the ACW system is given as,

$$H_{ACW}(z) = \frac{N(z)}{A(z)} = \sum_{k=1}^{P} \frac{1}{1 - z_k z^{-1}}$$

(2.18)

where *N(z)* is given by:

$$N(z) = \sum_{k=1}^{P} \prod_{i=1 \neq k}^{P} (1 - z_i z^{-1})$$

(2.19)

Because *N(z)* can be shown to be minimum phase, the ACW feature is guaranteed to be causal and can be defined as [6],

$$c_{ACW}(n) = \begin{cases} \log(P), & n = 0 \\ c_{LP}(n) - c_{nn}(n), & n > 0 \end{cases}$$

(2.20)

where $c_{nn}(n)$ is a component that changes with each frame to better approximate the channel effects seen in that section of the speech signal. Based on Equations 2.18 and 2.19, *N(z)* can be rewritten as,

$$N(z) = P\left(1 - \sum_{k=1}^{P-1} b_k z^{-1}\right)$$

(2.21)

where b_k are the coefficients of the polynomial in the numerator of Equation 2.18. These coefficients are used to define the subtractive component, $c_{nn}(n)$, in the same way that the LP coefficients define the LP cepstrum in Equation 2.15. Normally, finding *N(z)* would be a computationally demanding process involving polynomial root finding; however, a simpler, faster algorithm has been proposed that allows for the formulation of *N(z)* – and thus, the ACW feature – by computing b_k directly from a_k [8].

2.1.5 Mel frequency cepstral coefficients (MFCC). Mel frequency cepstral coefficients (MFCC), also sometimes called mel-warped cepstral coefficients, are not based on linear prediction as the other features have been. MFCCs are calculated via the application of a non-uniformly spaced bandpass filter bank. The filter bank is based on the so-called mel scale of frequencies, which are based on subjective pitch comparisons done by human test subjects. As such, the scale was an attempt to create a set of frequencies that more closely approximates the frequency response of the human auditory system [2] [6].

The MFCC is calculated by defining the outputs of a P-dimensional filter bank as Y_p and using the following equation,

$$c_{MFCC}(n) = \sum_{p=1}^{P} \left[\log Y_p\right] cos \left[\frac{\pi n}{P}(p - 0.5)\right]$$

(2.22)

where $c_{MFCC}(n)$ are the MFCC coefficients. Here, the outputs of the filter bank are logarithmically compressed and followed by a discrete cosine transform (DCT) in order to calculate the MFCC coefficients [2].

2.1.6 Temporal derivatives of features. In order to utilize not just the spectral properties of a speech frame, but also the temporal and transitional information contained across multiple frames, temporal derivatives of the feature vectors are taken [6] [9]. The first-order temporal derivative is approximated by,

$$d_k = \frac{\sum_{n=-m}^{m} n(c_{k+n})}{\sum_{n=-m}^{m} n^2}$$

(2.23)

where c_k and d_k are the feature vector and delta feature at frame k, respectively, and m is the number of frames the delta feature looks forward and backward. Using a value of m = 2 results in a delta feature calculated across a span of 5 frames. Replacing the feature vector with the delta feature in Equation 2.23 results in a second-order temporal derivative. By concatenating the feature vector, the delta feature, and the double delta feature, the speaker verification system is able to utilize 36-dimensional feature vectors [9].

2.2 Speaker Verification Overview

Speaker verification is the practice of determining whether a test speaker truly holds the identity that they have claimed via the analysis of their speech. Whereas a database typically contains a high number of models pertaining to each of the target speakers on which the system has been trained, the test utterance offered by the test speaker needs only to be compared to that target model of the claimed identity. Therefore, even for a speaker verification system that has been trained on potentially millions of users, the process of validating the test speaker's claim is only a 1:1 problem.

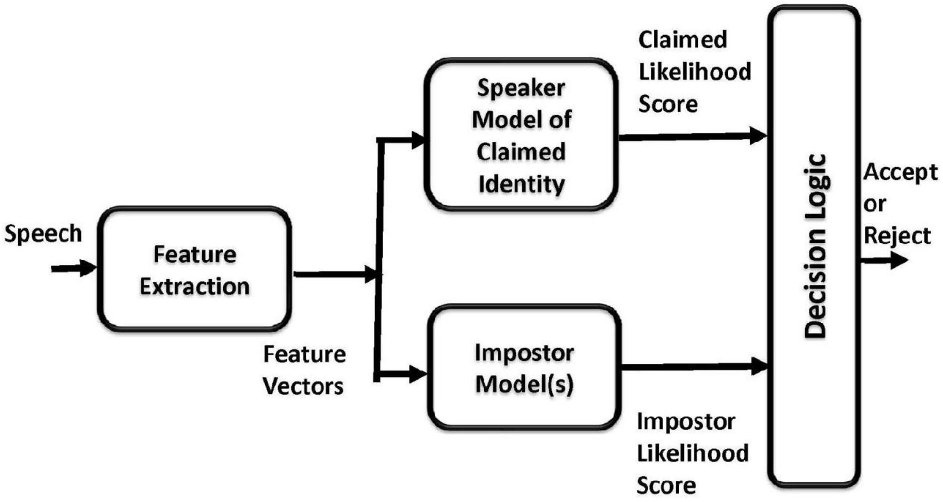

Figure 1. Speaker verification system block diagram.

In order to accomplish such a feat, the system needs to store a model that theoretically represents everyone in the world other than the target speaker. Of course, gathering speech samples from every human on Earth would be truly daunting task; however, from a statistical standpoint, a so-called universal background model is a close enough approximation for speaker verification purposes [1]. A more in-depth discussion about how the universal background model is calculated and used can be found in the following subchapter on Gaussian mixture models.

2.2.1 Performance measures. The proceeding subsections discuss the various performance measures that are used to evaluate the practical ability of a speaker verification system. Note that the subset listed within this thesis is not exhaustive; many more performance measures exist for judging the capability of a speaker recognition system, however the metrics included here are the only ones used to evaluate the implementation proposed by the methodology in this thesis.

Each of the listed performance measures are predicated upon the use of generating a large set of scores via a training set and a testing set of speech utterances. The generated scores can be classed within two subdivisions: intraclass and interclass scores. Intraclass scores, sometimes called genuine or true scores, are generated by scoring all test speakers' utterances against their true identity. Interclass scores, sometimes called impostor scores, are generated by scoring all test speakers' utterances against every identity within the system other than their true identity.

Once both of these sets of scores are calculated, one can determine the scoring threshold at which the most desirable system performance is obtained. The scoring threshold is the score that needs to be generated for a given test in order for the system to accept that the test speaker holds the claimed identity. This threshold is the key parameter for determining the performance measures outlined ahead. In order to meet system specifications, many thresholds need to be tested. Each threshold tested will generate a unique false accept rate (FAR) and false reject rate (FRR) and a threshold must be picked that balances the tradeoff between FAR and FRR in a manner that suits the application of the speaker verification system [1].

2.2.1.1 False accept rate (FAR). A false acceptance is defined as the scenario where an impostor speaker was accepted as the claimed identity by the system. The total number of times that such a scenario arises divided by the total number of acceptances in the testing phase [1]. As the threshold of the speaker verification system is lowered, necessarily the FAR will rise. This is because lowering the threshold means that impostor scores (which are, ideally, lower than genuine scores) will begin to be found above the now-too-low threshold.

In certain applications, it is desirable that the performance of the system is tuned to allow for a higher false accept rate than false reject rate. Consider a consumer electronics device that uses biometric verification to grant access to the device. If the FAR is tuned to be extremely low, balance between FAR and FRR dictates that the FRR will inevitably be much higher. In such a case, the user who owns the device would find themselves incapable of unlocking their device more often than they would deem acceptable. It would then behoove the engineers designing this system to pick a lower threshold, within reason, so that their customers do not decide to purchase a competitor's device.

2.2.1.2 False reject rate (FRR). A false rejection is defined as the situation where a true speaker fails to be validated against their own target model. The false reject rate is the total number of trials in which such a scenario occurs divided by the total number of rejections during the testing phase of the system [1]. Contrary to the FAR, a system with too high of an FRR means that the chosen scoring threshold is too high and even genuine scores are being rejected from the system.

Just as a higher FAR is desirable in some cases, so too is a higher FRR desirable in certain scenarios. Consider a biometric verification system being used to validate access to an area with highly sensitive equipment. In this case, it would be necessary to design for a higher FRR to ensure that absolutely no one without authority to enter the area is granted access by the system. Perhaps the authorized individuals would have to try multiple times to be granted access to the area, but that is less important than keeping undesirable individuals out of the area.

2.2.1.3 Equal error rate (EER). The equal error rate is the error rate at the operating point of the system at which the FAR and FRR are equal. Typically, system designers would not choose a threshold that gives an equal FAR and FRR, as discussed in previous sections; however, the EER remains a popular performance metric for researchers to compare the performance of their system to those of the past [1]. In the performance analysis of the systems implemented in the approach of this thesis, the EER is the performance metric used to determine the systems' capabilities. A lower EER indicates a better performance and is desired.

2.3 Gaussian Mixture Model – Universal Background Model (GMM-UBM)

The GMM-UBM system is one of the most common classifiers used for speaker verification. Though Gaussian mixture models alone can form a stochastic representation of a speech utterance, in order for speaker verification to be performed it must be couple with a universal background model [10]. The formulation of each of these two components is discussed hence.

2.3.1 Gaussian mixture model (GMM). In speaker recognition, a Gaussian mixture model is a method for representing a training set of feature vectors as a weighted sum of probability density functions characterized by the mean vectors, covariance weights, and mixture weights of the feature vectors [1] [2] [9] [10]. For a GMM using a weighted sum of *M* Gaussian mixtures, the probability density function is given as,

$$p(x|\lambda) = \sum_{i=1}^{M} w_i N(x_i; \mu_i, \Sigma_i)$$

(2.24)

where w_i are the mixture weights, which satisfy the condition $\sum_{i=1}^{M} w_i = 1$. For each mixture, $N(x_i; \mu_i, \Sigma_i)$ are the mixture densities, which, for D-variate Gaussian functions, are defined by,

$$N(x_i; \mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2} |\Sigma_i|^{1/2}} e^{-\frac{1}{2}(x_i - \mu_i)^T \Sigma_i^{-1} (x_i - \mu_i)}$$

(2.25)

where *D* is the dimension of the extracted feature vectors, μ_i are D-dimensional vectors, and Σ_i are D×D covariance matrices; however, typically only the diagonals of the covariance matrices are calculated in order to be more computationally efficient [9] [10].

The GMM for a given speaker is characterized by the three parameters – w_i, μ_i, and Σ_i. These parameters are iteratively refined for each utterance using the expectation maximization (EM) algorithm, which can assure monotonic convergence to optimal parameters in roughly five iterations [9] [11].

2.3.2 Universal background model (UBM). The second component of a GMM-UBM system, the universal background model, provides a contrast to the target speaker models against which each test utterance can be scored. Ideally, the universal background model should be an alternative to the claimed speaker model which represents the entire space not occupied by the target model [12]. Thus, it is crucial to use a different set of speakers in calculating the UBM than were used in calculating the target speaker GMMs.

The UBM is calculated in the same way as target speaker GMMs except that, instead of calculating one GMM per speaker, one GMM is created to represent all of the pooled speakers represented in the UBM [10]. Again, the parameters of the UBM are refined using the EM algorithm, just as with the target speaker GMMs.

Once the UBM is calculated, the speaker models are adapted from this large speaker-independent GMM using a process called *maximum a priori* (MAP) adaptation [1]. This can be done by using all of the parameters – weights, means, and covariance – to adapt the GMMs, or it can be done using only the weights. Using only the means has been shown to result in minimal degradation to system performance, so it is preferred due to the less computationally complex nature of this type of adaptation [10]. For a GMM-UBM system, the number of Gaussian mixtures used in computing the speaker GMMs must equal the number used to compute the UBM.

2.3.3 Score computation. Scoring a test utterance in a GMM-UBM system is done by comparing the likelihood of the utterance belonging to the claimed speaker to the likelihood of it belonging to the background model.

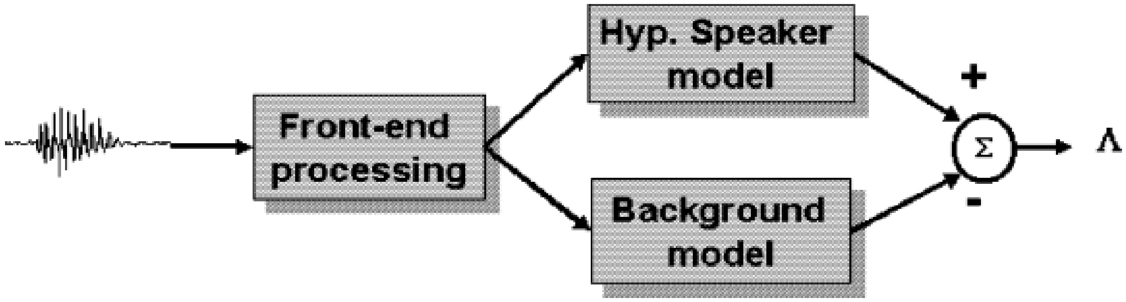

Figure 2. GMM-UBM scoring paradigm *[10].*

The score of a test utterance is calculated is a log-likelihood ratio as follows,

$$S(x) = \log p(x|\lambda_{target}) - \log p(x|\lambda_{ubm})$$

(2.26)

where *S(x)* is the score for utterance *x* and λ_{target} and λ_{ubm} are the target speaker's GMM and the UBM, respectively [10] [11]. By scoring many test utterances belonging to different target speakers in this way, the GMM-UBM system can be tuned for specific performance based on the performance metrics discussed in Section 2.2.1.

2.4 Gaussian Supervector – Partial Least Squares (GSV-PLS)

The second classifier implemented in the methodology of this thesis is the Gaussian Supervector – Partial Least Squares (GSV-PLS) system, so named for its application of Gaussian supervectors in a PLS regression framework devised for speaker verification. As explained in [2], Gaussian supervectors are essentially a byproduct of a GMM-UBM system, so it makes sense to utilize them as inputs to a classifier that is entirely unique from GMM-UBM. The calculation of Gaussian supervectors and an explanation of the PLS regression framework used is provided in the following subchapters.

2.4.1 Gaussian supervectors. By following the process outlined in Sections 2.3.1 and 2.3.2, the speaker model GMMs for the GMM-UBM system are calculated, resulting in a set of weights, means, and covariance matrices for each speaker. The Gaussian supervector is a simple concept that extrapolates the use of the adapted mean vectors of a speech utterance by concatenating each one, resulting in a single supervector for that speech utterance.

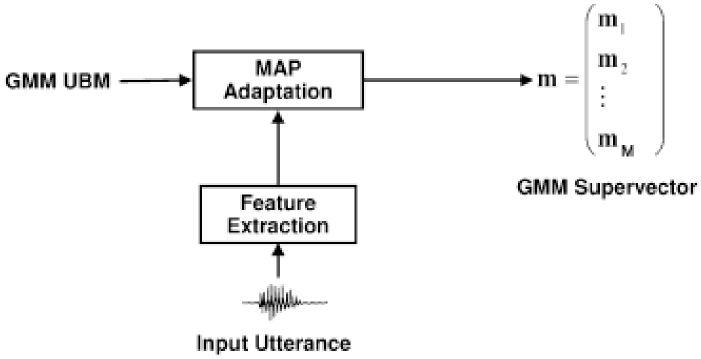

Figure 3. Gaussian supervector formulation *[13]*.

For an adapted GMM using *M* Gaussian mixtures and a 36-dimensional feature consisting of a feature concatenated with delta and double-delta features, the resulting supervector would be a 36*M* × 1 column vector. For the implementation in this thesis, 256 Gaussian mixtures were used, meaning the GSVs were 9216 × 1 in dimension.

In order to use GSVs in a discriminative speaker verification system, the mean supervectors must be scaled on a mixture-by-mixture based by the covariances and weights corresponding to each mixture [13]. This is accomplished using the following equation,

$$\mu_{scaled} = \sum_{i=1}^{M} \left(\sqrt{w_i} \Sigma_i^{-1/2} \mu_i \right)$$

(2.27)

where it is assumed that only diagonal covariances are used.

2.4.2 Partial least squares regression. The key concept in partial least squares (PLS) regression is that observed data can be explained in terms of some latent factors driving the data [14]. This idea seems to mesh well with speech processing where observed data can be complicated and random, but is a result of the combination of different physiological characteristics in the human body.

Assuming that there are N speakers in the training set, whose utterances are represented by *d*-dimensional Gaussian supervectors, one can denote the concatenation of each of the GSVs as an *N* × *d* matrix, *X*. In this schema, the GSVs are concatenated such that one GSV inhabits each row. Note that for systems where more than one training utterance per speaker exists, *N* should be multiplied by the number of training utterances per speaker. A vector of labels, *Y*, can also be created that is *N* × *1* in dimension. The vector *Y* shall consist of 1's for entries corresponding to target speakers and -1's for entries corresponding to impostor speakers [4]. PLS attempts to approximate the relationship between the supervectors and the labels by projecting the vectors into latent spaces via the following decomposition,

$$X = TP^T + E$$

$$Y = UQ^T + F$$

(2.28)

where *T* and *U* are matrices containing the vectors of latent factors that govern *X* and *Y*, respectively. *Q* and *P* are loading vectors, and *E* and *F* are residual matrices [4]. The equations in 2.28 are solved using the nonlinear iterative partial least squares (NIPALS) algorithm, which attempts to maximize the covariance between the latent vectors *T* and *U* [4] [14].

$$\max[cov(t_i, u_i)]^2 = \max_{|w_i|=1}[cov(Xw_i, Y)]^2$$

(2.29)

The NIPALS algorithm results in a set of weights, $W = \{w_1, w_2, \ldots, w_p\}$, that can be used in a framework for PLS regression that can be utilized for speaker verification. This is done by substituting the weights into Equation 2.28, which results in:

$$XW = TP^T W + E$$

(2.30)

Note that the residuals need not be considered in subsequent steps. Equation 2.30 can now be rewritten in terms of T by multiplying by the inverse of $P^T W$:

$$T = XW(P^T W)^{-1}$$

(2.31)

Rewriting U in terms of T and substituting into Equation 2.28 results in,

$$Y = TDQ^T + HQ^T + F = XW(P^T W)^{-1}DQ^T + \bar{F}$$

(2.32)

where $U = TD + H$, such that D is a diagonal matrix and H is the residue [4]. Equation 2.32 can be used to define the PLS regression,

$$Y = XB + G$$

(2.33)

where,

$$B = W(P^T W)^{-1}DQ^T$$

(2.34)

and B, also referred to as the Beta matrix, is a matrix containing the PLS regression coefficients that allow for the computation of speaker verifications scores [4]. In the PLS regression framework for speaker verification, each speaker will have a unique Beta matrix. In essence, the Beta matrix can be thought of as the speaker model itself.

Scores are calculated in a one-shot process as follows,

$$S(x) = X_{test}B_{target}$$

(2.35)

where X_{test} is a concatenation of all of the supervectors representing utterances that are to be scored against the target speaker Beta matrix [3] [4]. In such a paradigm, *S(x)* will be a vector containing one score pertaining to each supervector tested against the target Beta matrix. These scores can then be analyzed the same way GMM-UBM scores are in order to determine the system performance via equal error rate.

2.5 Enhancement Methods

The enhancement methods discussed in the following subchapter have been applied to both classifiers implemented in the methodology of this thesis. The determination of the optimal application of these methods constitutes the principal contribution of this research. To begin, the subchapter explains the paradigm used for signal-to-noise ratio (SNR) estimation, beginning with a discussion of vector quantization. The SNR estimation system used in conjunction with the application of affine transforms for feature augmentation, taken as a whole is the first enhancement method used. The second enhancement method is score-level feature fusion, which can be subdivided into three different methods.

2.5.1 Signal-to-Noise Ratio (SNR) Estimation. Additive noise can affect speech signals in varying degrees of severity. The speech signal could be lightly corrupted, so minimally that the resultant is perceptually indistinguishable from clean speech; or it could be so heavily degraded that one cannot recognize it as speech at all. The more prominent the noise in a corrupted signal is compared to the clean signal, the lower the signal-to-noise ratio (SNR) of the corrupted signal is. Methods have been developed that allow for blind estimation of the SNR of a speech signal, which is a vital process in the steps for applying the proper affine transform in order to compensate for the quality loss of the signal [15].

2.5.1.1 Vector Quantization (VQ). The SNR estimation technique used in this thesis is based on a classifier known as a vector quantizer (VQ). Vector quantization is a classification paradigm that assigns labels, called codewords, to feature vectors classified as falling within the near-region of a centroid, also called a codevector [1]. The near-region of a codevector is defined by the local region in which points in the vector space have a smaller distance to that codevector than any other in the classifier. Although one could use all of the available training vectors to develop the VQ system, typically the number of codevectors are compacted via a clustering algorithm to produce a VQ codebook [1] [2].

For a set of N test feature vectors, $T = \{f_1, f_2, \ldots, f_N\}$, the score, D, known as the average distortion, against a codebook of size L, $C = \{c_1, c_2, \ldots, c_L\}$, is defined as,

$$D(T, C) = \frac{1}{N}\sum_{i=1}^{L} min_{1\le k\le K} d(f_i, c_k)$$

(2.36)

where $d(x_i, c_k)$ is a distance measure between the *i*th test feature vector and the *k*th codevector in codebook *C* [2]. In other words, for every codevector, the distance between all of the test feature vectors and that codevector is summed, resulting in *L* summed distance scores. From there, the average distortion is taken as the minimum summed distance. A lower average distortion represents a higher probability that the test feature vectors belong to the same speaker on whom the codebook was trained [2].

2.5.1.1.1 Linde-Buzo-Gray (LBG) algorithm. In the 1980s, an algorithm was devised for computing accurate codebooks based on the K-means clustering algorithm [2] [16]. The algorithm has come to be known as the Linde-Buzo-Gray algorithm, so named for the authors of the paper in which the algorithm was first proposed. The algorithm is a four stage iterative process, which begins by defining an initial codebook of size 1 as

$$c_1 = \frac{1}{N}\sum_{i=1}^{N} f_i$$

(2.37)

where f_i are feature vectors in the training set of *N* vectors, *F*. Once this codebook is calculated, find the average distortion between the training set and the codebook.

$$D = \frac{1}{N}\sum_{i=1}^{N} d(f_i, c_1)$$

(2.38)

The average distortion at each stage must be stored, and shall be denoted as D_{prev}. Next, set D_{prev} equal to D and create a binary split from the codebook by perturbing the codevector by a small factor ε. This results in a codebook with two codevectors.

$$c_2 = c_1 + \varepsilon; \qquad c_1 = c_1 - \varepsilon$$

(2.39)

Now, the Voronoi region of each codevector must be calculated. The Voronoi region, V_j, of a codevector, c_j, is defined as the set of training feature vectors that are a shorter distance away from that codevector than any other codevector in the codebook [17].

$$V_j = \{f_i \in F \mid d(f_i, c_j) \leq d(f_i, c_k), j \neq k\}$$

(2.40)

In the case where the distance from a training feature vector is equivalent between two codevectors, the training feature vector is assigned to a Voronoi region arbitrarily. With the two Voronoi regions calculated, the new average distortion must be calculated and compared to the average distortion stored as D_{prev}.

$$D = \frac{1}{N(V_1)} \sum_{i=1}^{N(V_1)} d(f_i \in V_1, c_1) + \frac{1}{N(V_2)} \sum_{i=1}^{N(V_2)} d(f_i \in V_2, c_2)$$

(2.41)

Note that the training feature vectors are only scored in the distance measure against the codevector to whose Voronoi region they belong. If the difference between the new average distortion and the previous average distortion falls below a threshold, the process can be stopped. Otherwise, the algorithm continues to iterate the process, setting *D_{prev}* equal to *D* and performing another binary split.

2.5.2 Affine transform. The affine transform is a technique borrowed from the digital image processing domain that allows for the rotation, scaling, and translation of feature vectors extracted from speech signals [6] [18]. Such transformational capability can compensate for the distortion caused by many types of noise degradation to speech, including additive noise at the speaker level. This helps to remediate the problem of training to testing mismatch when the test utterance is captured in a noisy environment, resulting in an overall increase in system performance.

For a set of feature vectors for the training condition, $\boldsymbol{x} = \{x_1, x_2, \ldots, x_p\}$, and a set of feature vectors for the testing condition, $\boldsymbol{y} = \{y_1, y_2, \ldots, y_p\}$, the affine transform mapping that relates $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined as,

$$\boldsymbol{y} = A\boldsymbol{x} + b$$

(2.42)

which can be expanded to:

$$\begin{bmatrix} y_1 \\ \vdots \\ y_p \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1p} \\ \vdots & \ddots & \vdots \\ a_{p1} & \cdots & a_{pp} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_p \end{bmatrix} + \begin{bmatrix} b_1 \\ \vdots \\ b_p \end{bmatrix}$$

(2.43)

The matrix *A* and the vector *b* and parameters learned only using the feature vectors for the training condition [6] [18] [19]. For a feature vector with *N* individual frames, the training condition feature vector is denoted as $\boldsymbol{x}^{(i)}$ and the testing condition feature vector is denoted as $\boldsymbol{y}^{(i)}$ where i = 1 to N. With these sets of feature vectors, a squared error function is composed [18]:

$$E(m) = \sum_{i=1}^{N} \left[y^{(i)}(m) - \boldsymbol{a}_m^T x^{(i)} - b(m)\right]^2$$

(2.44)

where $m = \{1,2,\ldots,p\}$, and $\boldsymbol{a}_m^T$ is the mth row of A. The error function is minimized by taking the gradient with respect to a_m and then $b(m)$, which results in the following system of equations:

$$\begin{bmatrix} \sum_{i=1}^{N} x^{(i)} x^{(i)^T} & \sum_{i=1}^{N} x^{(i)} \\ \sum_{i=1}^{N} x^{(i)^T} & N \end{bmatrix} \begin{bmatrix} \boldsymbol{a}_m \\ b(m) \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{N} y^{(i)}(m) x^{(i)} \\ \sum_{i=1}^{N} y^{(i)}(m) \end{bmatrix}$$

(2.45)

The error function is minimized for all values of m, resulting in m systems of equations $(p+1)$ in dimension [6] [18].

2.5.3 Fusion. The second enhancement method applied to the speaker verification systems in the methodology of this thesis is feature fusion. Given the four features that were implemented in this research, there are ten different possible combinations of features that could be combined using fusion; however, only the case where all four features are utilized was tested in this approach.

2.5.3.1 Score level fusion. In speaker verification, fusion is performed at the score level, meaning that each feature is evaluated independently and the scores corresponding to each trial are fused at the end stage [20]. Because the resultant scores from each feature will be highly irregular comparatively, the scores must first be normalized between 0 and 1 in order to be utilized for classification. Normalization is performed on a feature by feature basis using all of the interclass and intraclass scores pooled together. The following equation is used for mapping the scores for a feature between 0 and 1,

$$S_{norm} = \frac{S - \min(S)}{\max(S) - \min(S)}$$

(2.46)

where S_{norm} is the set of normalized scores pertaining to a specific feature and S is that feature's scores before normalization. The three fusion methods utilized in the approach proposed by this thesis are listed as follows.

2.5.3.1.1 Sum fusion. Sum fusion is performed by adding the scores generated by each feature together. For a specific trial, the sum fusion score is computed as,

$$S_{sum} = \sum_{i=1}^{N} S_{norm}^{(i)}$$

(2.47)

where S_{sum} is the sum fusion score and $S_{norm}^{(i)}$ is the normalized score for the ith feature [20]. For each of the discussed fusion methods under Section 2.5.3.1, $N = 4$ since there are four features whose scores are being fused. Additionally, Equation 2.47 generates a score for a single trial, so the process must be repeated for all genuine and impostor trials.

2.5.3.1.2 Product fusion. Product fusion is performed by multiplying the scores generated by each feature in a specific trial together as follows,

$$S_{prod} = \prod_{i=1}^{N} S_{norm}^{(i)}$$

(2.48)

where S_{prod} is the product fusion score for the trial [20].

2.5.3.1.3 Maximum fusion. Finally, maximum score fusion is performed by taking the maximum of the scores generated by each feature in a single trial as follows,

$$S_{max} = \max\left(S_{norm}^{(1)}, S_{norm}^{(2)}, \dots, S_{norm}^{(N)}\right)$$

(2.49)

where S_{max} is the maximum fusion score for the trial [20].

2.5.3.2 Classifier fusion. Classifier fusion for speaker verification also takes place on a score level using the three methods described in the previous sections. When a test utterance is input to the speaker verification system, it will be scored by each classifier independently. For a given trial, any of the above fusion methods can be applied using the same normalization technique and using $N = 2$, if two classifiers are being fused as in this thesis. In this thesis, the GMM-UBM system is a generative model for speaker recognition and the GSV-PLS system is discriminative. Such fusion between generative and discriminative systems has been shown to work well in the past. It is thought that this method works because generative and discriminative classifiers hold complementary information about the speakers [3].

2.6 Statistical Analysis

In order to determine the statistical significance of results obtained during various stages of the proposed methodology, multiple analyses of variance are performed on the datasets. When comparing two different factors (e.g. feature type and enhancement method), a two-way analysis of variance (ANOVA2) is performed. This method focuses on determining whether there is a statistically significant difference in the means of each of the factors individually and whether there is an interaction between the two [21].

In addition to the ANOVA2, a one-way analysis of variance (ANOVA) is used in the final stages of this treatment in order to compare the means of the equal error rates of the two classifiers and their fusion. Following each ANOVA, Tukey's method for multiple comparison of means is applied, using a confidence interval of 95% in order to make determinations of whether the differences in means are statistically significant [21].

Chapter 3

Methodology

In Chapter 3, a complete explanation of each stage of the proposed methodology is provided. First, a description of the speech database and the characteristics of the speech data contained therein is stated. Following that, an overview of the feature extraction process used to generate each of the features listed in Section 2.1 is included. The methods used for choosing the parameter called "affine resolution" and the optimal feature or fusion strategy are then detailed. Next, the entire process for training both the GMM-UBM and GSV-PLS classifiers is described. The chapter concludes with a discussion of the application of the proposed enhancement methods, followed by the statistical analysis used to validate the results.

3.1 Description of the TIMIT Database

A subset of the TIMIT database developed as a collaborative effort between Texas Instruments and MIT was used to develop and test the speaker verification systems in this approach. The TIMIT database contains speech samples from a total of 630 different speakers; this approach utilized 90 of those speakers for training and testing and 168 speakers taken from the 'test' portion of the original TIMIT CD for development of the background models. Each speaker in the database has 10 speech utterances recorded at a sample rate of 16 kHz; however, they have all been down-sampled to 8 kHz before utilization in the speaker verification system development. For the 90 speakers used in the training and testing phase of development, eight sentences per speaker were selected for training while the remaining two were used for testing. All ten sentences of the 168 background speakers were used in the development of the UBM.

3.2 Feature Extraction

Feature vectors are extracted on a frame-by-frame basis. Each speech utterance is broken up into frames 30 ms in length with an overlap of 20 ms. This corresponds to 240 samples in length with 160 samples overlap at a sampling rate of 8 kHz. Because each speech utterance varies in length, the total number of frames is inconstant. A Hamming window is applied to emphasize the samples near the center of the frame, and a pre-emphasis filter is applied to the frame in order to boost higher frequency formants that contain a lot of discriminant information. Following the pre-emphasis, 12th order linear prediction is performed on each frame to generate 12 LP coefficients for each frame. These coefficients are then used to compute the LP cepstral coefficients, which are then modified to calculate the ACW and PFL features. The MFCC feature is calculated by performing a short-time Fourier transform (STFT) on the speech, warping it to the Mel frequency scale, and taking the DCT on the log of the Mel spectrum to generate cepstral coefficients.

Each of the four generated features (CEP, ACW, PFL, MFCC) are analyzed to compute the delta and double-delta features as formulated in Equation 2.23. The temporal derivative is taken over a span of five frames centered on the current speech frame. The 12-dimensional feature vector and its corresponding 12-dimensional delta and double delta feature vectors are concatenated, resulting in a 36-dimensional feature vector. This is done for each of the feature types; therefore, there are four 36-dimensional feature vectors per frame of speech being processed.

Voice activity detection (VAD) based on the spectral energy of the speech signal is used [2]. Frames with low spectral energy are determined to contain little to no discriminative speech information, so they are discarded. This energy thresholding selection is done for each of the four features, so only frames containing useful speech information will be taken as part of the feature vector matrix for that utterance.

3.3 Affine Resolution

Affine resolution is a parameter that determines which affine transforms are available for use in enhancing the feature vectors of a noisy test utterance. Affine transforms were trained on the SNR levels of corrupted speech. That is, the training condition on which the affine transforms were trained was clean speech and the testing condition was the same speech corrupted at a specific decibel SNR. A total of 31 Affine transforms were trained, spanning from 0 dB to 30 dB. Each affine transform is denoted as A_s, where the subscript, $s = 0, 1, 2, \dots, 30,$ indicates what the SNR of the testing condition speech at which the affine transform was trained on was.

For an Affine resolution equal to 1, the entire set of affine transforms, $\boldsymbol{A} = \{A_0, A_1, \dots, A_{30}\}$, is available. Increasing the affine resolution decreases the available affine transforms. For an affine resolution of 5, the subset of available affine transforms is $\boldsymbol{A} = \{A_0, A_5, A_{10}, \dots, A_{30}\}$. Further increasing the affine resolution to 10 decreases the available subset to $\boldsymbol{A} = \{A_0, A_{10}, A_{20}, A_{30}\}$.

In order to determine whether a decrease in the number available affine transforms causes a decrease in system performance, affine resolutions of 1, 5, and 10 were tested in parallel for both the GMM-UBM and GSV-PLS system. Each resolution was tested using only blind SNR estimation. Test utterances corrupted at SNRs in intervals of 3 dB were used so as not to bias the experiment in favor of any of the affine resolutions. In other words, for example, if SNR intervals of 5 were used, it is likely that an affine resolution of 5 would emerge as the best option because if the SNR estimator is accurate this resolution would almost always choose the perfect affine transform. Therefore, an SNR interval of 3 dB was chosen as a compromise for testing these three resolutions.

To be able to make a comparison between the affine resolutions, all three configurations need to be tested simultaneously. When a test utterance is processed by the speaker verification system, a randomly selected noise vector from the NoiseX database is used to corrupt the speech at the selected SNR. If the three affine resolutions were tested separately, a different noise vector would be applied to the test utterance making any comparison between the obtained results invalid. Therefore, the system was designed to apply the noise vector, estimate the SNR of the feature vectors, and apply affine transforms whose availability is governed by the three different affine resolutions to the same exact feature vectors.

After scores were corresponding to each affine resolution, equal error rates were generated and compared using ANOVA2 to determine whether there was a statistically significant difference in the means of the EERs. Multiple trials for each SNR were performed using rotation of the utterances used for training and testing vectors. The first five utterances of each speaker were used for training in every rotation because these utterances were used to train the affine transforms.

Table 1

Training and testing utterance rotation schema.

Rotation Number	Training Utterances	Testing Utterances
1	8, 9, 10	6, 7
2	7, 9, 10	6, 8
3	7, 8, 10	6, 9
4	7, 8, 9	6, 10
5	6, 9, 10	7, 8
6	6, 8, 10	7, 9
7	6, 8, 9	7, 10
8	6, 7, 10	8, 9
9	6, 7, 9	8, 10
10	6, 7, 8	9, 10

This paradigm of rotation to produce multiple trials was used in every instance that ANOVA was used to validate the results obtained from experiments. After ANOVA and Tukey's method for multiple comparison of means was applied, the "best set" of affine resolutions at each SNR was decided. The best set is chosen by the taking the affine resolutions that lead to the lowest EER and whose mean was statistically proven to be different than another affine resolution. In the case that there is no statistical difference between the means of multiple affine resolutions' EERs, both are taken to be part of the best set. After speech corrupted at all SNRs had been tested, the optimal affine resolution for each system was chosen as the one that appeared in the best set most often.

3.4 Optimal Feature Selection

Optimal feature selection was performed in order to make a comparison between the best configurations of each speaker verification system in the final phase of the thesis. In this case, the three fusion strategies are included in the selection for the best feature. Each feature and fusion type was tested in 10 trials at each SNR from 0 dB to 30 dB plus clean speech. Just as with the affine resolution, rotation was performed with the training and testing utterances, followed by ANOVA and multiple comparisons of the means of the EERs corresponding to each feature or fusion. A best set of features and/or fusions was denoted for each SNR and the optimal feature was chosen as that which appeared in the best set most often. When comparing the overall performance of the GMM-UBM and GSV-PLS systems in their optimal configuration against each other and their fusion, only the optimal feature was used to calculate EERs.

3.5 GMM-UBM Classifier

The following subchapter details the process used for training and testing the GMM-UBM classifier. The classifier is comprised of four sets of 90 GMMs – one pertaining to each speaker on whom the system was trained – and four UBMs, which serves as the alternative hypothesis in the scoring phase of testing the system [10]. One classifier is trained per each of the four feature types used.

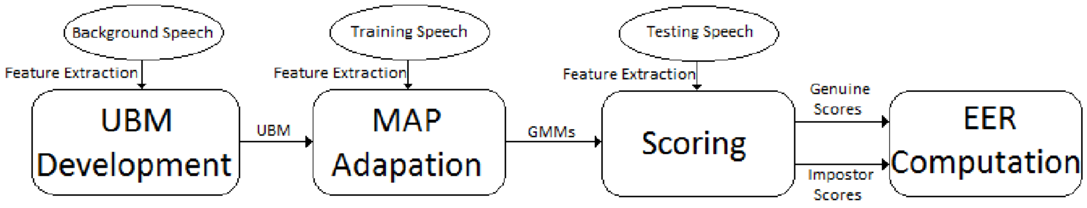

Figure 4. GMM training and testing diagram.

3.5.1 Training the GMM-UBM classifier. Before the performance of the classifier can be tested, it must be intelligently designed by choosing the correct parameters and following the appropriate steps for generating the models that serve as the basis for the speaker verification system. The training phase is divided into two partitions – namely, the creation of the universal background model (UBM) and the adaptation of all of the Gaussian mixture models (GMMs). Each of these partitions shall be elaborated upon in this subchapter.

3.5.1.1 Universal background model computation. The UBM is initialized using the K-means algorithm to randomly choose *N* centers for the pooled feature vectors of all of the background speakers. The covariance matrices are calculated as the sample covariance for the background feature vectors that are located nearest to each center. Mixture weights are determined by the proportion of feature vectors that are contained in each cluster.

This implementation uses 256 mixtures for each UBM, chosen to match the recommended configuration for the PLS system so that only one set of UBMs needed to be trained [4]. After the UBM is randomly initialized, 10 iterations of the EM algorithm are used to refine the UBM parameters [10]. This process is repeated for each of the four feature types used in this implementation.

3.5.1.2 Gaussian mixture model adaptation. The GMMs against which all of the testing feature vectors will be tested are adapted from UBM using *maximum a priori* (MAP) estimation. Four sets of 90 GMMs serve as the speaker models in the speaker verification system – one for each feature type. The GMMs are adapted using only the means, as it was shown that this outperforms adaptation using the weights, means, and covariances together in [10].

3.5.2 Testing the GMM-UBM classifier. Testing utterances are passed through the feature extraction method to generate four sets of feature vectors per utterance. If the system is being tested under additive noise conditions, a randomly selected noise vector from the NoiseX database is applied to the speech utterance prior to feature extraction. If SNR estimation and affine transform enhancement is used, the CEP feature is used to estimate the SNR of the test utterance. A set of affine transforms is selected based on their availability governed by the affine resolution and applied to each of the feature vectors. Affine transforms have been trained for each feature type specifically.

After all speech utterances have gone through feature extraction and possibly have had affine transforms applied, scores for each utterance are generated in bulk. The score for each test utterance is the log-likelihood ratio given by Equation 2.26. The scores for test utterances being tried against models for speakers to whom they do not belong are interclass scores. As two utterances are used for testing, there will be a total of 16,020 interclass scores. This comes about because each utterance (2) of every speaker (90) is tested against every other speaker's model (89). Multiplying these numbers together results in 16,020 total interclass trials. The scores for test utterances being tried against their own speaker models are intraclass scores. There are far fewer of these scores since each utterance (2) only needs to be scored against their own speaker model (90). This results in a total of 180 intraclass trials.

Inter- and intraclass scores are used to calculate FARs and FRRs at thresholds covering the range of the scores. Histograms can be created to demonstrate the score distributions of the interclass and intraclass trials. A larger overlap between the two classes of scores indicates a higher EER will be found for the system configuration used in the trials.

3.6 GSV-PLS Classifier

The following subchapter details the process used to train and test the GSV-PLS classifier. This classifier was trained after the GMM-UBM system so that it was unnecessary to generate further UBMs. Each speaker model consists of one PLS regression Beta matrix, meaning that a total of four sets of 90 Beta matrices comprise the entire system. Again, one classifier is trained per feature type.

3.6.1 Training the GSV-PLS classifier. The first steps of training the GSV-PLS classifier are similar to training the GMM-UBM classifier, but there is variation in the later steps. To begin with, the same randomly seeded UBMs with 256 mixtures and 10 EM iterations for refinement are used. It was shown that a middling number of mixtures such as 256 works well for GSVs used in a PLS regression framework since higher orders lead to overfitting of the background data, which is detrimental to a discriminative classifier [4].

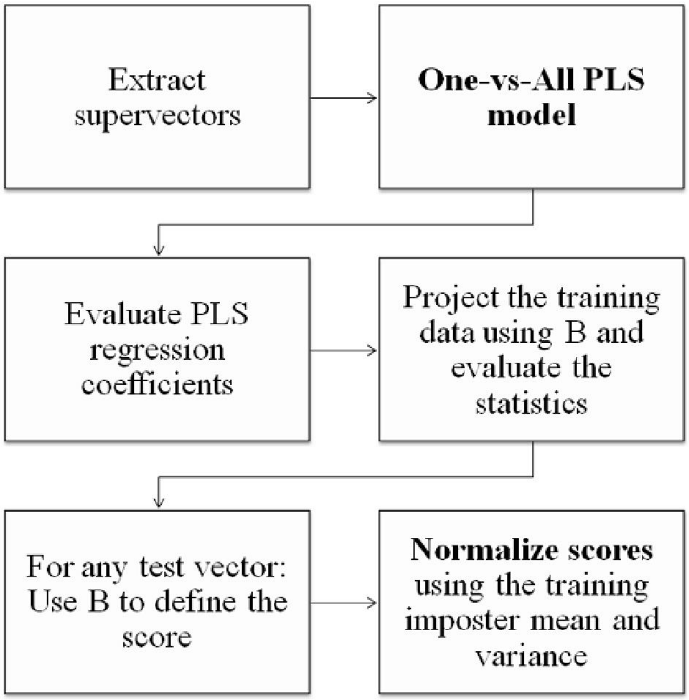
Extract supervectors
One-vs-All PLS model
Evaluate PLS regression coefficients
Project the training data using B and evaluate the statistics
For any test vector: Use B to define the score
Normalize scores using the training imposter mean and variance

Figure 5. GSV-PLS training and testing block diagram *[4]*.

3.6.1.1 Gaussian supervector computation. MAP estimation is again performed to generate GMMs, but this time one GMM per feature is created for each utterance in the training set. After each utterance for each speaker in the training set has had GMMs adapted, the mean vectors from the utterance-specific GMMs are taken and concatenated to create GSVs. With eight training utterances each, one utterance is mapped to one GSV; therefore, each speaker will have eight GSVs for a total of 720 training GSVs per feature type.

3.6.1.2 Gaussian supervector normalization. As each GSV is created, the weights and covariances of the utterance-specific GMM are used to normalize the GSVs as shown in Equation 2.27 [13]. Note that it is not explicitly stated that this normalization needs to occur in [4] where GSV-PLS framework is described; however, it was found in this research that performance was severely degraded if normalization was not performed.

3.6.1.3 Partial least squares regression framework. The feature-specific GSVs for every speech utterance are placed in a $(S \times U)$ by $(N \times D)$ matrix, where S is the number of speakers, U is the number of training utterances per speaker, N is the number of UBM mixtures, and D is the dimension of the feature vectors. $(N \times D)$ is considered the supervector dimension. In this approach, $N{\times}D = 256{\times}36 = 9216$.

The matrix is arranged such that each row contains one 1×9216 dimension GSV. This matrix is used as the X matrix in the PLS regression framework discussed in Section 2.4.2. The Y matrix is taken as a vector of labels such that $Y_i = 1$ if X_i contains a target speaker for the PLS speaker model being trained, where Y_i and X_i are the ith row of the Y

and *X* matrices, respectively. If X_i contains an impostor speaker for the PLS model being trained, then Y_i = -1. Since eight training utterances are used, each *Y* matrix will have eight entries labeled as 1, while the remaining 712 entries will be labeled as -1. In training each PLS speaker model, the same *X* matrix is used each time, while the *Y* matrix is changed to reflect the appropriate target and impostor speaker labels. The number of PLS components used for each regression is intelligently selected by first calculating the PLS regression with the maximum number of components and observing the percentage of variance in the *Y* model explained by the model. Each PLS component will consecutively contain less and less information about the variance, so the number of components used is index of the highest component found to explain at least 1% of the variance. The number of PLS components changes for each regression that is calculated.

The PLS regression is calculated for each speaker, resulting in a total of 90 Beta matrices that serve as the models against which test utterances can be scored.

3.6.2 Testing the GSV-PLS classifier. As with the GMM-UBM system, test utterances are used to perform feature extraction. If noisy speech is being tested, a randomly chosen noise vector from the NoiseX database is applied to the speech signal at the selected SNR prior to feature extraction. If SNR estimation and affine transform enhancement are enabled, the CEP feature vector is used to estimate the SNR of the speech signal. A set of affine transforms is selected from the available set that is governed by the affine resolution. The affine transforms are then applied to the four feature vectors

After each of the prior steps have been carried out, MAP estimation of the feature vectors is performed to create four GMMs for each test utterance – one per feature type. The means of these GMMs are taken and concatenated so that one GSV per feature per utterance is created. Each of these GSVs is multiplied against all of the Beta matrices that serves as the PLS speaker models to produce trial scores. Resultant scores from Beta matrix representing the speaker to whom the test utterance belongs are deemed intraclass scores. Scores from Beta matrices representing speakers to whom the test utterance did not originate are deemed interclass scores. Again, a total of 16,020 interclass and 180 intraclass scores are computed per feature type.

The GSVs used for training the PLS models are also scored against the models to generate interclass scores from which descriptive statistics can be obtained. For each feature type, the mean and standard deviation of the scores obtained from trying the training GSVs against the models are calculated. The scores contain information used to estimate speaker-specific mean and variances for the impostor distribution [22]. These are used to normalize the scores using the T-norm,

$$S_{norm} = \frac{S - \mu_i}{\sigma_i}$$

(3.1)

where S_{norm} is the normalized score, S is the initial score, and μ_i and σ_i are the mean and standard deviation of speaker *i,* respectively [4] [22]. The normalized inter- and intraclass scores are both used in calculating FARs and FRRs corresponding to each threshold that is tested. Thresholds covering the entire range of the pooled scores are tested. The final threshold is chosen to be that which generates an equal error rate.

3.7 Enhancement Methods

Two enhancement methods were examined for their efficacy in augmenting system performance for both the GMM-UBM and GSV-PLS classifiers.

3.7.1 Affine transform. Affine transforms were trained using the first five speech utterances of each speaker in the training set. These utterances were corrupted at SNRs ranging from 0 dB to 30 dB such that the affine transforms were trained on the SNR levels. A VQ SNR estimator was implemented in order to determine the SNR of unknown test utterances using a soft decision approach [15]. The SNR estimator only considered the CEP feature and did not include delta features.

Three cases were considered in determining whether the use of an affine transform is recommended. The first case is the one where no affine or SNR estimation is used to enhance the features. The second case is where blind SNR estimation was to determine which affine transform to apply to the features. The third case is the control case, where perfect SNR estimation is used by simply telling the system which affine transform to apply. Each of these cases was tested using the 10 rotations of training and testing speech at SNRs ranging from 0 dB to 30 dB plus clean speech.

3.7.2 Fusion Score-level fusion in each individual classifier was implemented separately. For both the GMM-UBM system and the GSV-PLS system, the four features were fused using maximum, sum, and product fusion [20]. These fusion strategies were tested using the 10 rotations at SNRs ranging from 0 dB to 30 dB plus clean speech.

After optimal feature selection was performed for both classifiers separately, the scores generated using only those features in each classifier were used to perform classifier fusion.

3.8 Performance Analysis

For each experiment, statistical analysis was performed by using ANOVA to determine whether the difference in the performance of the factors being tested was statistically significant. Every experiment was performed using speech corrupted at SNRs ranging from 0 dB to 30 dB in addition to clean speech. The only exception is the analysis of the affine resolution, which did not include clean speech, and only tested affine resolution at SNRs in intervals of 3 dB. For each SNR, 10 trials were performed using the rotations described in Table 1.

To determine the optimal affine resolution for each classifier to use, ANOVA2 was performed for each SNR using the EERs generated from all 10 trials. The two factors under investigation in this experiment were the feature types, not including fusion, and the affine resolution. Affine resolutions of 1, 5, and 10 were investigated to determine whether the means of the EERs was statistically different. Following this procedure, Tukey's method for multiple comparison of means was utilized to identify which affine resolutions were significantly different from the others at a 95% confidence interval [21].

The second experiment was to justify the use of the affine transform. The three cases under investigation were using no enhancement, using blind SNR estimation, and using perfect SNR estimation. The goal of this experiment was to show that using SNR estimation in conjunction with the affine transform gives significantly better results than doing no enhancement. At the same time, to ensure that the blind SNR estimation performed reasonably well, it needed to be shown that the results are not significantly different than perfect SNR estimation in most cases. Again, ANOVA2 was used to prove statistical significance between the results. The second factor under test was the features, this time including all of the fusion strategies. After ANOVA, multiple comparison was performed on the results from both factors – the first factor to prove the usefulness of the enhancement method, and the second factor to choose the optimal feature for final comparison between the classifiers.

The final experiment was to compare each of the classifiers using their best configurations against each other and against the fusion of the two using three fusion strategies. The best configurations for each classifier were those that came about as a result of proving the statistical significance of the difference between affine resolutions and feature types in the previous experiments. For this experiment, only blind SNR estimation was used, in order to simulate a practical implementation. A one-way ANOVA was used to prove the significance of these results. The factor under test was the classifier, where the three fusion methods are considered separate classifiers. After

ANOVA, multiple comparison was again used to identify the best classifier at each SNR. A recommendation was then made based on which classifier performed best most often.

Chapter 4

Results

This chapter provides a complete overview of all of the results obtained from the research presented in this thesis. The results are presented in the order that the experiments were performed, as results from earlier experiments were used to justify the choice of parameters in later experiments. To begin, the statistical analysis of the affine resolution parameter at each SNR is shown. These results were used to decide which affine resolution to use in the next experiment. A global recommendation is given for each classifier as it is impossible to perfectly determine the SNR of a test utterance in a practical scenario. The results for the experiments to justify the use of the affine transform enhancement follow. From these results, evidence that the affine transform is effective is given, and a recommendation is made for the optimal feature to use in the following stage. Finally, the results for overall system performance are detailed. A recommendation is made for the best system configuration to use as a way to counter system degradation in the presence of additive noise.

For each of the analyses of variance performed, the null hypothesis is that the means of the EERs obtained from trials of each of the factors is the same. Using the multiple comparison plots, the 95% confidence intervals can be observed to determine whether there is sufficient evidence to reject the null hypothesis. Recall that a lower EER is desirable in terms of system performance.

4.1 Affine Resolution

The multiple comparison plots for the analysis of affine resolutions at each SNR from 0 dB to 30 dB in intervals of 3 dB follow. Interpretation of each of the plots is provided, and at the end of the subchapter a summary of the best affine resolutions is tabulated.

4.1.1 GMM-UBM affine resolution analysis. The following plots are the multiple comparison plots relating to the GMM-UBM trials to determine optimal affine resolution.

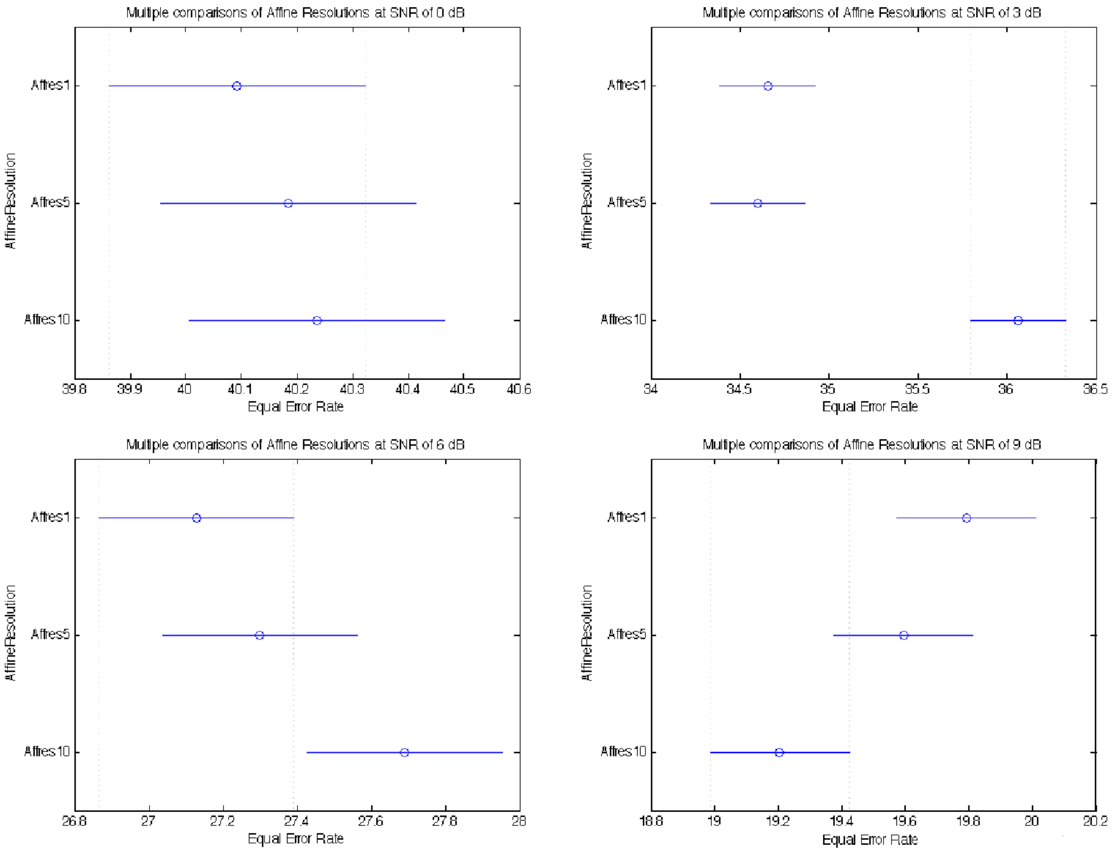

Figure 6. GMM-UBM affine resolution for 0 dB to 9 dB.

Figure 6 shows the results obtained by following the multiple comparison method to identify the best affine resolutions for trials from 0 dB to 9 dB. For each of the factors, the horizontal line represents the 95% confidence interval (CI) and the circle in the center of it represents the mean of the data. At an SNR of 0 dB, there is no statistically significant difference in the means of the EERs at any affine resolution. Therefore, the best set for this trial is considered to be all of the affine resolutions. At an SNR of 3 dB, the affine resolutions of 1 and 5 outperform an affine resolution of 10; therefore, the best set is comprised of affine resolutions of 1 and 5. At an SNR of 6 dB, the affine resolution of 1 is not considered statistically different than 5. Additionally, an affine resolution of 5 is not considered statistically different than 10; however, the difference in the means of the EERs at affine resolutions of 1 and 10 is statistically significant. For this reason, the best affine resolution at this SNR is 1. Finally, at an SNR of 9 dB, a similar scenario to 6 dB occurs, but this time an affine resolution of 10 is taken as the best.

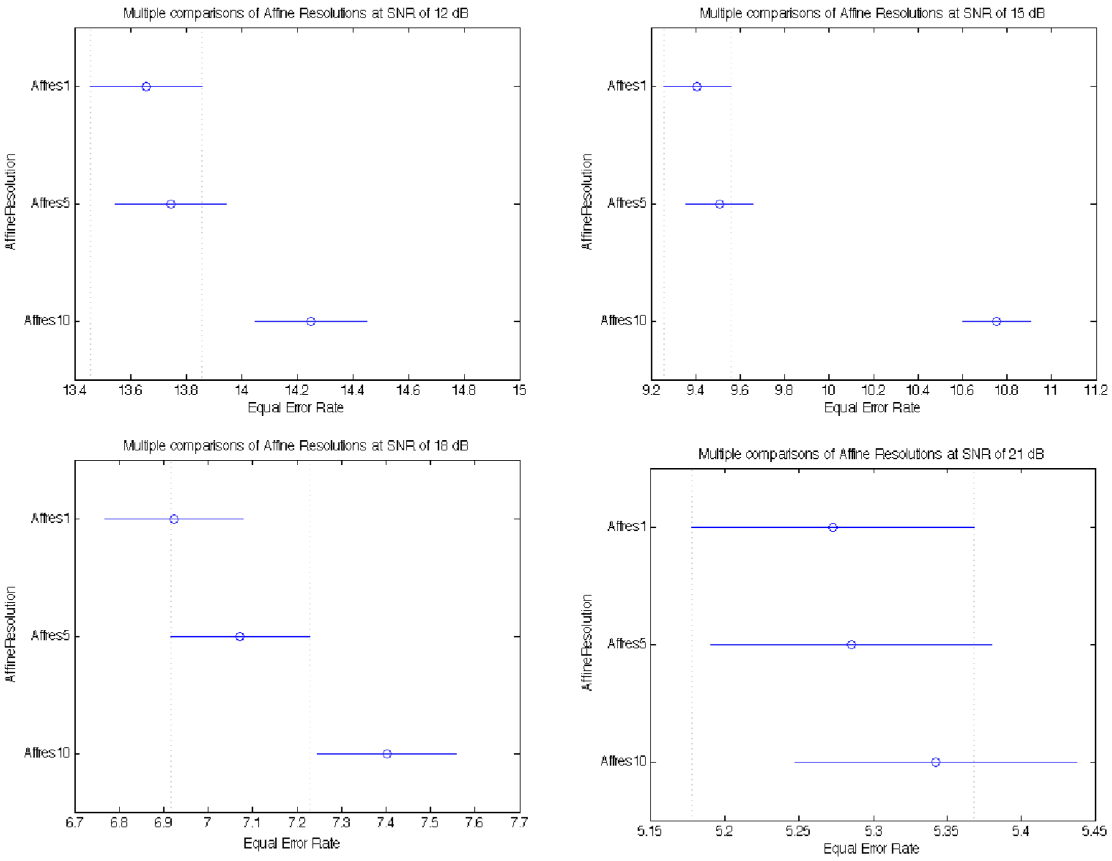

Figure 7. GMM-UBM affine resolutions for 12 dB to 21 dB.

Figure 7 shows the results for the GMM-UBM affine resolution analysis for 12 dB to 21 dB. At SNRs of 12, 15, and 18 dB, the best sets are comprised of affine resolutions of 1 and 5. At an SNR of 21 dB, the best set is comprised of affine resolutions of 1, 5, and 10 since there was no statistical difference in amongst their means.

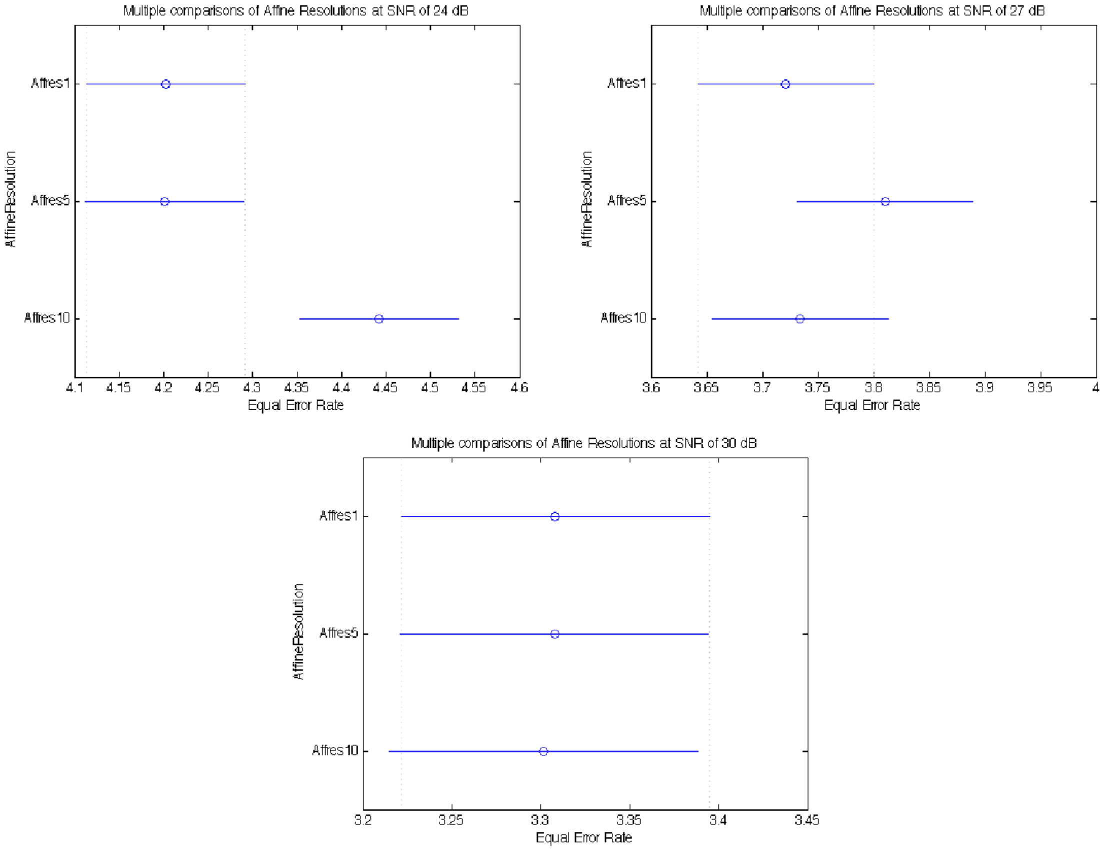

Figure 8. GMM-UBM affine resolution for 24 dB to 30 dB.

Figure 8 shows the results for the GMM-UBM affine resolution analysis for 24 dB to 30 dB. At an SNR of 24 dB, the best set is comprised of affine resolutions of 1 and 5. At SNRs of 27 dB and 30 dB, there is no statistically significant difference between any of the factors, so the best set includes all of them.

Table 2

Best sets of affine resolutions for GMM-UBM classifier.

Affine Resolution	Appeared in Best Set	Total Count
1	0 – 6 dB, 12 – 30 dB	10
5	0 dB, 3 dB, 12 – 30 dB	9
10	0 dB, 9 dB, 21 dB, 27 dB, 30 dB	5

Note: The column labeled Appeared in Best Set indicates the SNR trials for which each affine resolution was selected as one of the optimal resolutions.

The above table contains the number of times each affine resolution appeared in the best set for a given SNR. As was expected, an affine resolution of 10 only works well when the SNR of the corrupted speech signal falls near intervals of 10. Because an affine resolution of 1 appeared in the best sets most numerously, this affine resolution was chosen as the optimal value for the GMM-UBM system. The average EER for each affine resolution at each SNR taken over 10 trials is tabulated below. The results are provided for all four feature types.

Table 3

Average EERs for GMM-UBM affine resolution trials.

Test Condition	Feature	Affine Res. 1	Affine Res. 5	Affine Res. 10
0 dB SNR	ACW	43.09	43.18	43.08
	CEP	40.10	40.19	40.27
	PFL	42.23	42.43	42.41
	MFCC	34.95	34.94	35.18
3 dB SNR	ACW	37.92	37.83	38.82
	CEP	34.80	34.74	37.10
	PFL	37.41	37.22	38.56
	MFCC	28.49	28.59	29.77
6 dB SNR	ACW	31.24	31.42	31.81
	CEP	27.06	27.25	27.49
	PFL	29.05	29.29	29.81
	MFCC	21.16	21.24	21.65
9 dB SNR	ACW	23.69	23.60	23.15
	CEP	19.61	19.76	19.21
	PFL	20.68	20.21	20.15
	MFCC	15.19	14.81	14.31
12 dB SNR	ACW	16.39	16.47	17.06
	CEP	13.57	13.48	14.08
	PFL	14.58	14.60	15.05
	MFCC	10.09	10.43	10.80
15 dB SNR	ACW	10.88	10.71	12.68
	CEP	9.81	10.02	10.51
	PFL	9.88	10.11	10.86
	MFCC	7.06	7.19	8.96
18 dB SNR	ACW	7.48	7.55	7.88
	CEP	7.28	7.43	7.78
	PFL	7.28	7.38	7.79
	MFCC	5.66	5.94	6.16
21 dB SNR	ACW	5.50	5.54	5.48
	CEP	5.54	5.49	5.63
	PFL	5.75	5.77	5.88
	MFCC	4.30	4.34	4.38
24 dB SNR	ACW	4.39	4.45	4.51
	CEP	4.34	4.39	4.79
	PFL	4.58	4.52	4.85
	MFCC	3.51	3.44	3.62
27 dB SNR	ACW	3.76	3.82	3.87
	CEP	3.99	4.12	4.00
	PFL	4.07	4.20	4.04
	MFCC	3.06	3.09	3.03
30 dB SNR	ACW	3.31	3.31	3.32
	CEP	3.52	3.52	3.51
	PFL	3.57	3.56	3.55
	MFCC	2.83	2.84	2.82

4.1.2 GSV-PLS affine resolution analysis. The following plots are the multiple comparison plots relating to the GSV-PLS trials to determine optimal affine resolution.

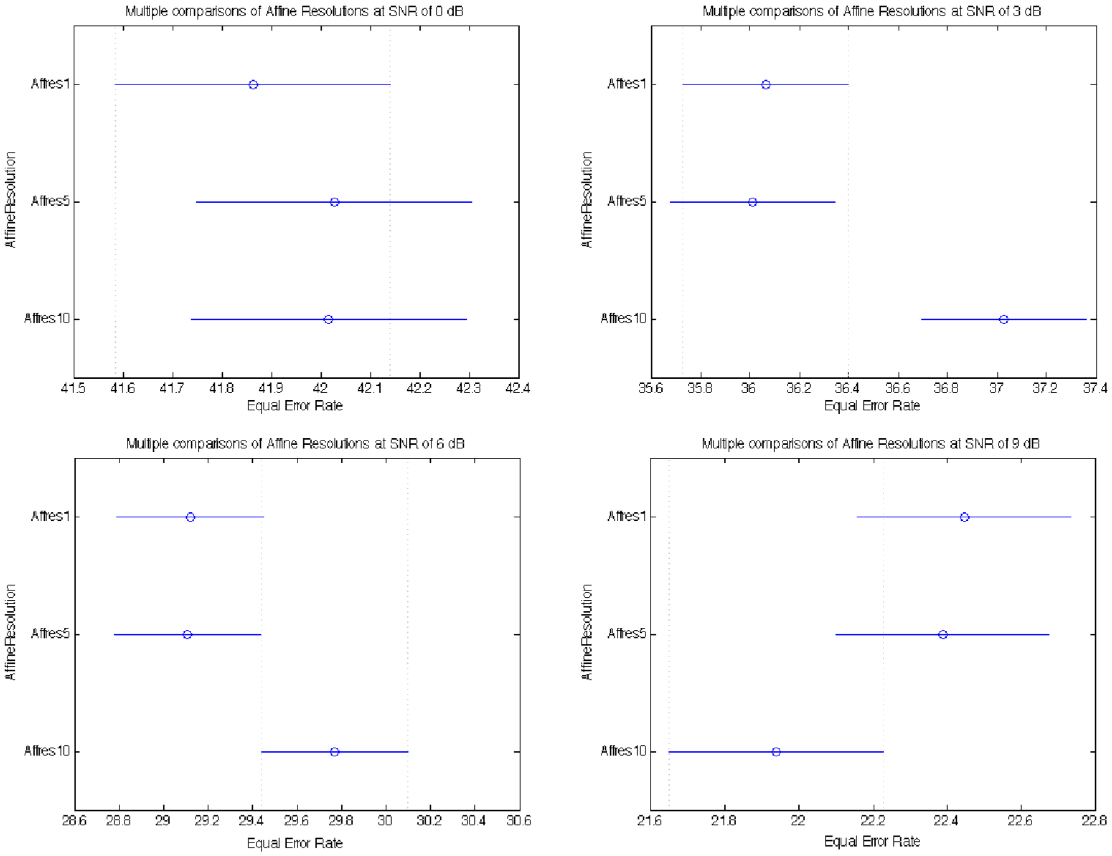

Figure 9. GSV-PLS affine resolution for 0 dB to 9 dB.

Figure 9 shows the multiple comparison plots for the affine resolution trials of the GSV-PLS classifier at SNRs of 0 dB to 9 dB. At SNRs of 0 dB and 9 dB, there is no statistically significant difference between the means of the factors, so the best set includes all affine resolutions. At an SNR of 3 dB, the best set is comprised of affine resolutions of 1 and 5. At an SNR of 6 dB, there is no statistical difference between the means of the EERs using affine resolutions of 1 and 5. Additionally, there is no statistical difference between the means of the EERs using affine resolutions of 1 and 10. However, there is a slight statistical difference in the means of the EERs using affine resolutions of 5 and 10. Therefore, the best set in this case only contains an affine resolution of 5.

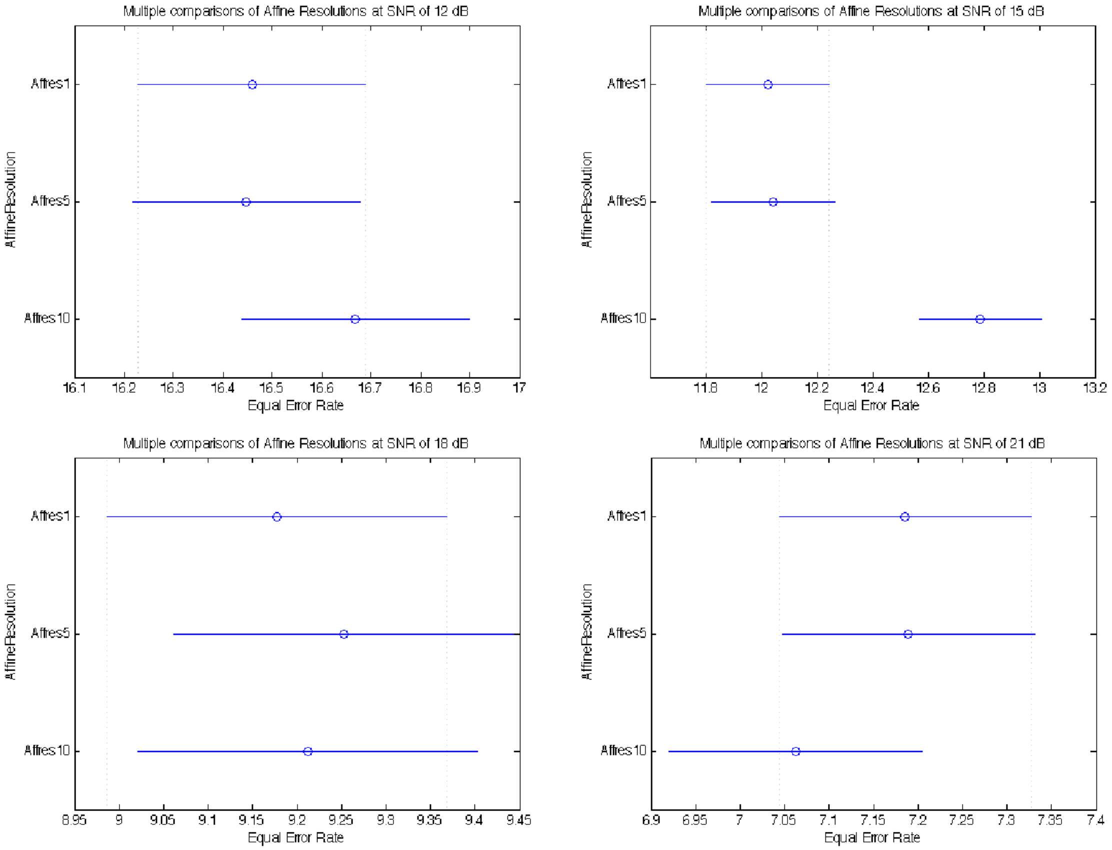

Figure 10. GSV-PLS affine resolution for 12 dB to 21 dB.

Figure 10 shows the affine resolution plots for 12 dB to 21 dB. At SNRs of 12, 18, and 21 dB, there is no statistically significant difference between any of the factors, so all are included in the best set. At an SNR of 15 dB, the best set is comprised of affine resolutions of 1 and 5.

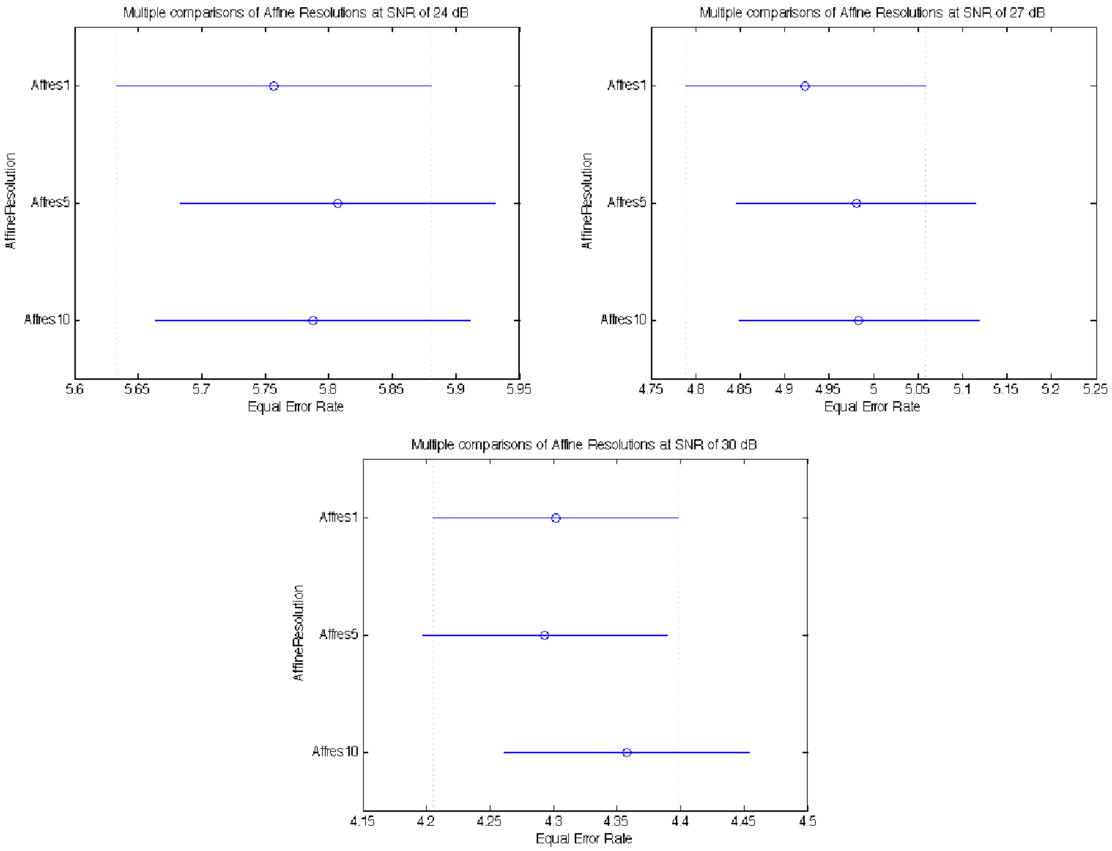

Figure 11. GSV-PLS affine resolution for 24 dB to 30 dB.

Figure 11 illustrates the affine resolution multiple comparison plots for SNRs from 27 dB to 30 dB. At each of the included SNRs, there is no statistically significant difference between any of the factors. All of them are included in the best set for each SNR.

Table 4

Best sets of affine resolutions for GSV-PLS classifier.

Affine Resolution	Appeared in Best Set	Total Count
1	0 dB, 3 dB, 9 – 30 dB	10
5	0 – 30 dB	11
10	0 dB, 9 dB, 12 dB, 18 – 30 dB	8

Note: The column labeled Appeared in Best Set indicates the SNR trials for which each affine resolution was selected as one of the optimal resolutions.

The above table contains the number of times each affine resolution appeared in the best set for a given SNR. There were more cases in the GSV-PLS trials that no statistical difference was apparent between the affine resolutions than in the GMM-UBM trials. This indicates that the affine resolution plays less of a role in determining the performance of the GSV-PLS system. Still, an affine resolution of 5 appeared in the best set at all SNRs, so this was chosen as the optimal value for this classifier.

Table 5

Average EERs for GSV-PLS affine resolution trials.

Test Condition	Feature	Affine Res. 1	Affine Res. 5	Affine Res. 10
0 dB SNR	ACW	43.94	44.02	43.95
	CEP	43.29	43.68	43.48
	PFL	41.36	41.31	41.30
	MFCC	38.85	39.08	39.33
3 dB SNR	ACW	38.53	38.53	39.42
	CEP	36.94	37.12	38.11
	PFL	36.55	36.15	37.23
	MFCC	32.24	32.24	33.34
6 dB SNR	ACW	31.66	31.66	32.08
	CEP	29.95	30.00	30.90
	PFL	28.54	28.56	29.02
	MFCC	26.32	26.21	27.08
9 dB SNR	ACW	25.07	24.74	24.44
	CEP	23.66	23.80	23.18
	PFL	21.89	21.66	21.19
	MFCC	19.17	19.35	18.93
12 dB SNR	ACW	17.99	18.31	18.45
	CEP	17.65	17.75	18.17
	PFL	16.24	16.04	16.31
	MFCC	13.96	13.70	13.74
15 dB SNR	ACW	12.91	13.04	13.74
	CEP	12.70	12.82	13.39
	PFL	12.18	12.20	12.84
	MFCC	10.29	10.10	11.17
18 dB SNR	ACW	9.63	9.62	9.54
	CEP	9.52	9.63	9.46
	PFL	9.27	9.32	9.54
	MFCC	8.29	8.45	8.30
21 dB SNR	ACW	7.18	7.02	6.96
	CEP	7.48	7.61	7.06
	PFL	7.38	7.34	7.43
	MFCC	6.71	6.79	6.80
24 dB SNR	ACW	5.49	5.62	5.47
	CEP	6.24	6.18	6.38
	PFL	5.82	5.91	5.61
	MFCC	5.48	5.51	5.68
27 dB SNR	ACW	4.61	4.77	4.74
	CEP	5.33	5.27	5.35
	PFL	5.44	5.42	5.33
	MFCC	4.32	4.47	4.51
30 dB SNR	ACW	3.85	3.89	3.95
	CEP	4.68	4.68	4.72
	PFL	5.04	4.99	4.95
	MFCC	3.64	3.61	3.81

4.2 Enhancement Methods

Using the optimal affine resolutions obtained in the previous experiment, the efficacy of the proposed enhancement methods was examined. Experiments were completed trying blind SNR estimation and perfect SNR estimation against doing nothing to corrupted speech. ANOVA2 was performed to determine the validity of using blind SNR estimation and to determine the optimal feature selection to use in the next stage. The results for both factors are detailed ahead. Average equal error rates for each of the two sets of 672 trials in this experiment are tabulated at the end of this subsection.

4.2.1 SNR estimation and affine transform. The multiple comparison plots used to analyze the performance of the three SNR estimation and affine transform configurations are shown below. These experiments were done for the GMM-UBM system and the GSV-PLS system separately.

4.2.1.1 GMM-UBM enhancement results. First, the results for the GMM-UBM enhancement trials are provided. Each plot shows the 95% confidence intervals for the three compared configurations at a specific SNR, based on performing 10 trials per configuration. A summary of the best SNR estimation modes is provided at the end of the section.

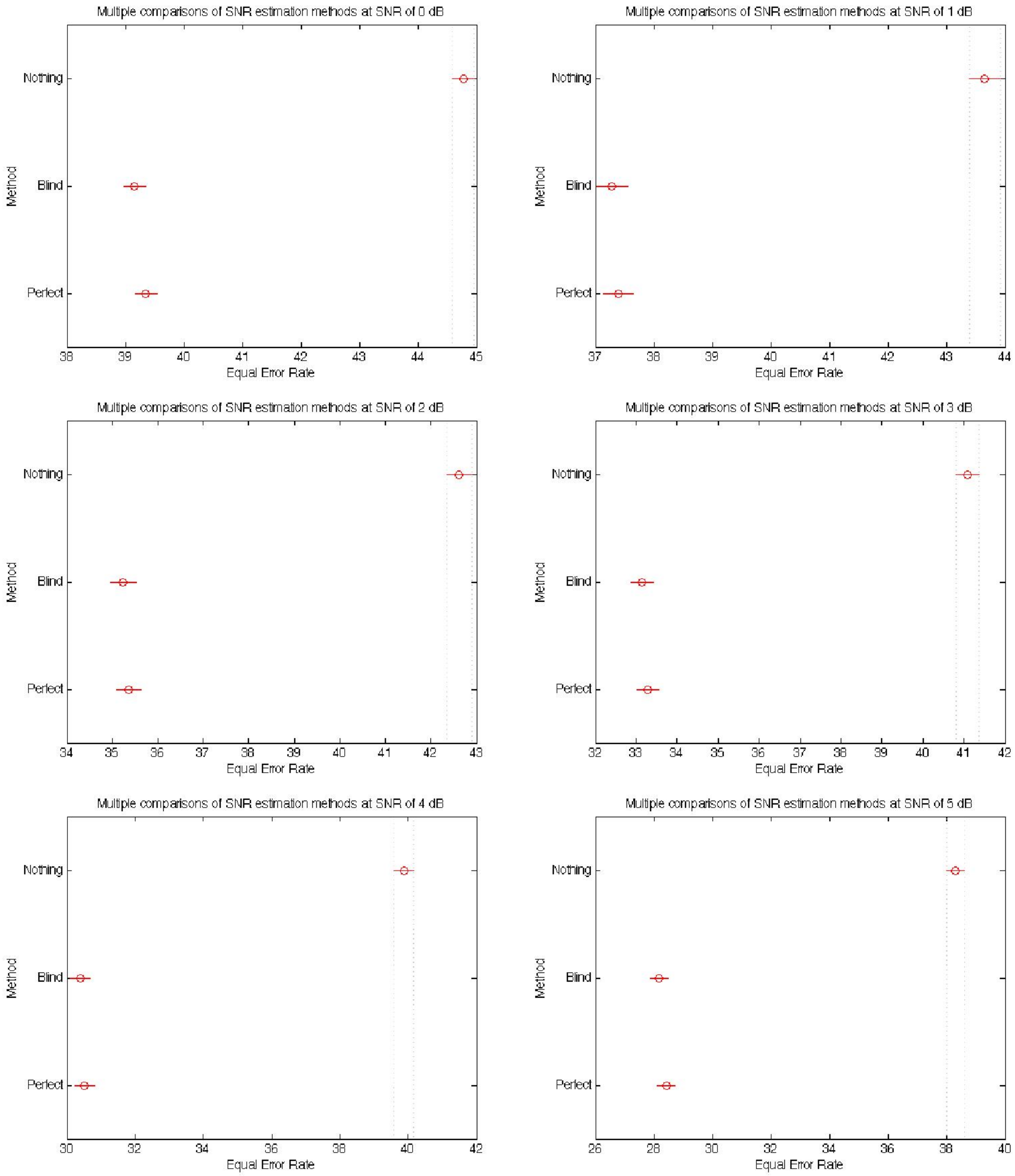

Figure 12. GMM-UBM SNR estimation mode for 0 dB to 5 dB.

Figure 12 shows the results obtained for the GMM-UBM SNR estimation mode trials at SNRs from 0 dB to 5 dB. In each of these cases, there is no statistically significant difference between the means of the EERs obtained using blind and perfect

SNR estimation; however, there is a vast difference in the performance of those two compared to doing no enhancement. At low SNRs like this, where the speech is heavily corrupted, this indicates that the affine transform does a very good job at reducing training to testing mismatch. However, even with the enhancement, the overall system performance is still beleaguered with the troubles of excessive additive noise.

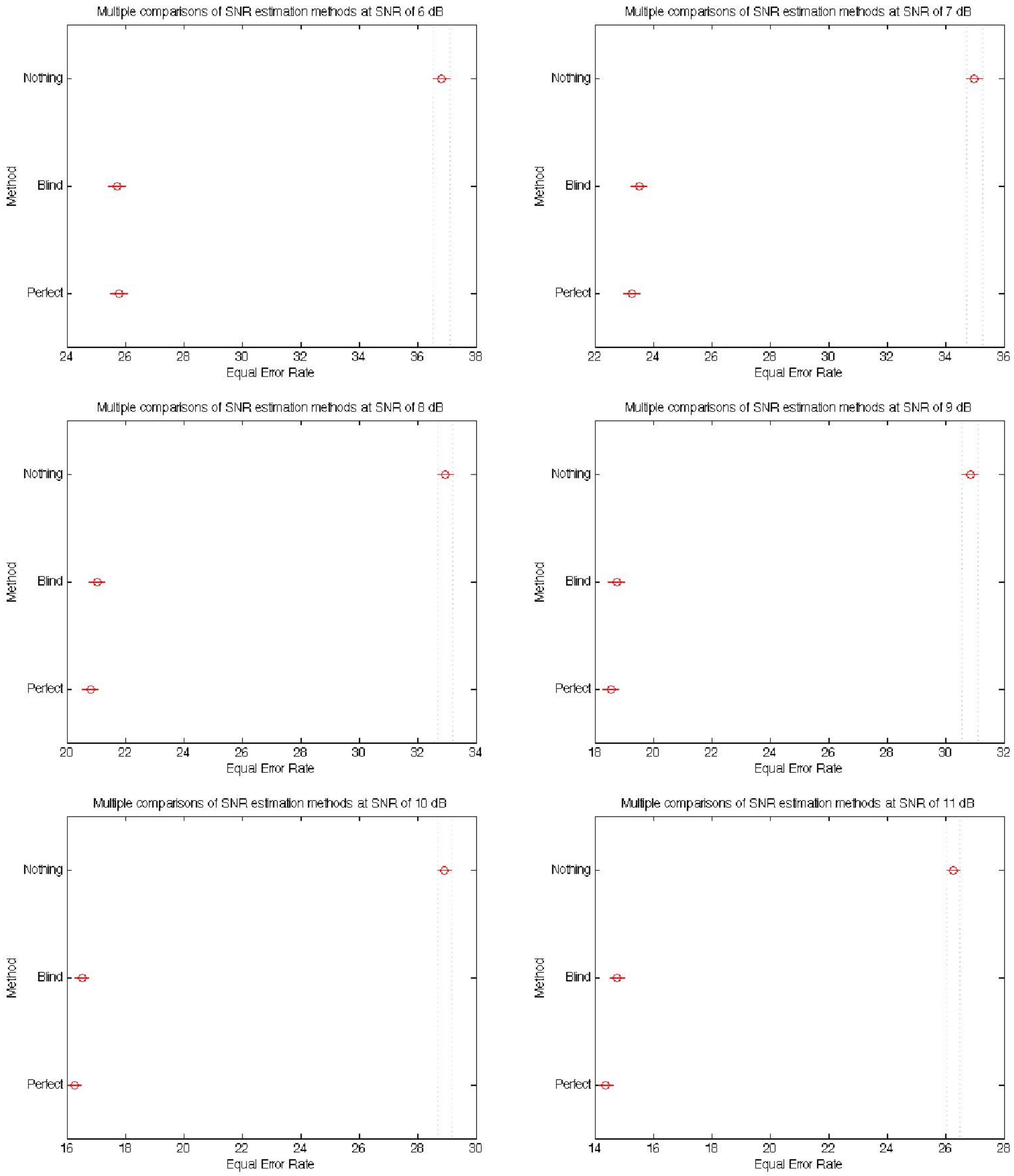

Figure 13. GMM-UBM SNR estimation mode for 6 dB to 11 dB.

Figure 13 shows the results from the GMM-UBM SNR estimation mode trials for SNRs ranging from 6 dB to 11 dB. Again, at all of these SNRs, there is no statistically significant difference between the blind and perfect SNR estimation scenarios, which always outperform the scenario where no enhancement is applied.

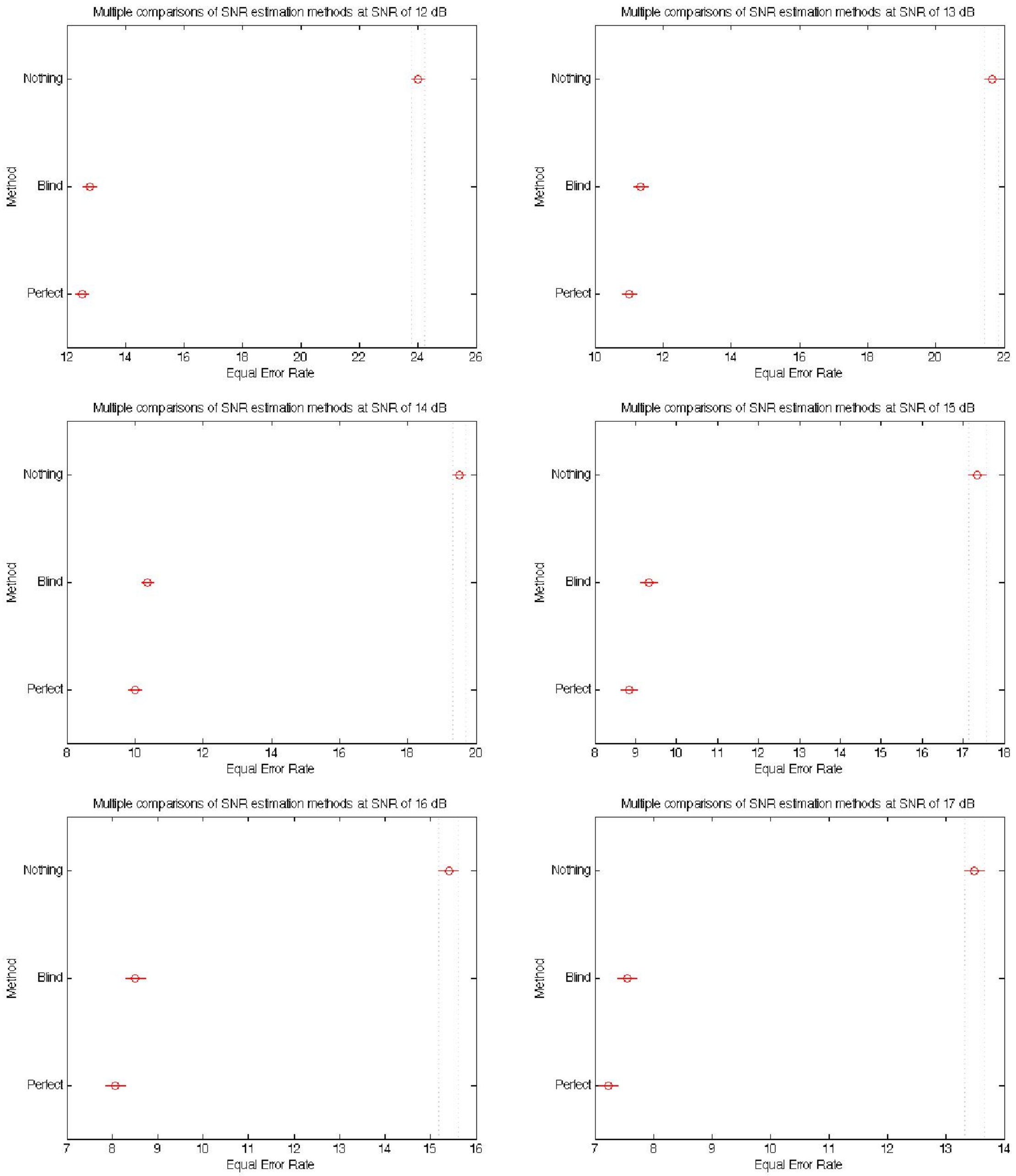

Figure 14. GMM-UBM SNR estimation mode for 12 dB to 17 dB.

Figure 14 contains the multiple comparison plots for the GMM-UBM SNR estimation mode trials for SNRs ranging from 12 dB to 17 dB. For SNRs of 12, 13, 14, and 17 dB, this same results as all of the previous trials emerge. In these cases, blind and perfect SNR estimation are not statistically different, and both far exceed the performance of doing nothing. However, for SNRs 15 dB and 16 dB, while blind and perfect SNR estimation still outperform no enhancement, there is a statistically significant difference between perfect and blind SNR estimation. Although in these cases, blind SNR estimation underperforms compared to the control scenario, such a situation is exceedingly rare for the GMM-UBM system, as will be shown towards the end of this subsection.

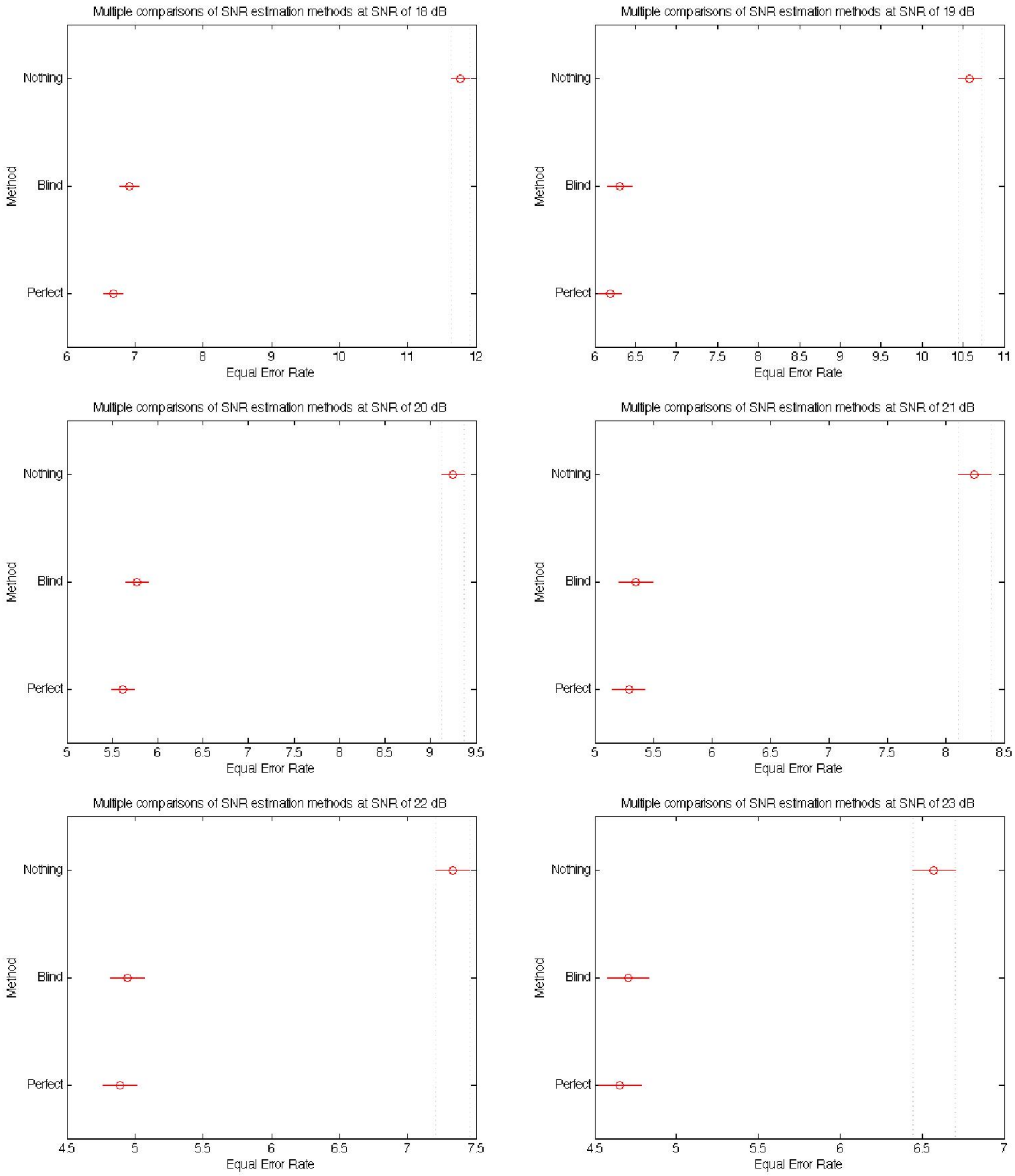

Figure 15. GMM-UBM SNR estimation mode for 18 dB to 23 dB.

Figure 15 contains the multiple comparison plots for the GMM-UBM SNR estimation mode trials for SNRs ranging from 18 dB to 23 dB. For all of these SNRs, the blind and perfect SNR estimation modes are not statistically different. Additionally, no enhancement is still inferior to using both blind and perfect SNR estimation. This indicates that even as higher SNR levels are tested, the SNR estimation and affine transform enhancement is still effective. However, it can be seen in Figure 15 that as SNR values increase, the enhancement method, while still statistically significant, does not provide as much of an increase in system performance than at lower SNRs. This is simply because there is less overhead error for the enhancement method to compensate than at lower SNRs. Whereas at lower SNR values the difference in EERs between enhancement and no enhancement ranged from about 6 to 15 percentage points, SNRs higher than about 20 dB see a difference of only about 3 to 4 percentage points. As the SNR further increases this difference will continue to decrease, but this is to be expected.

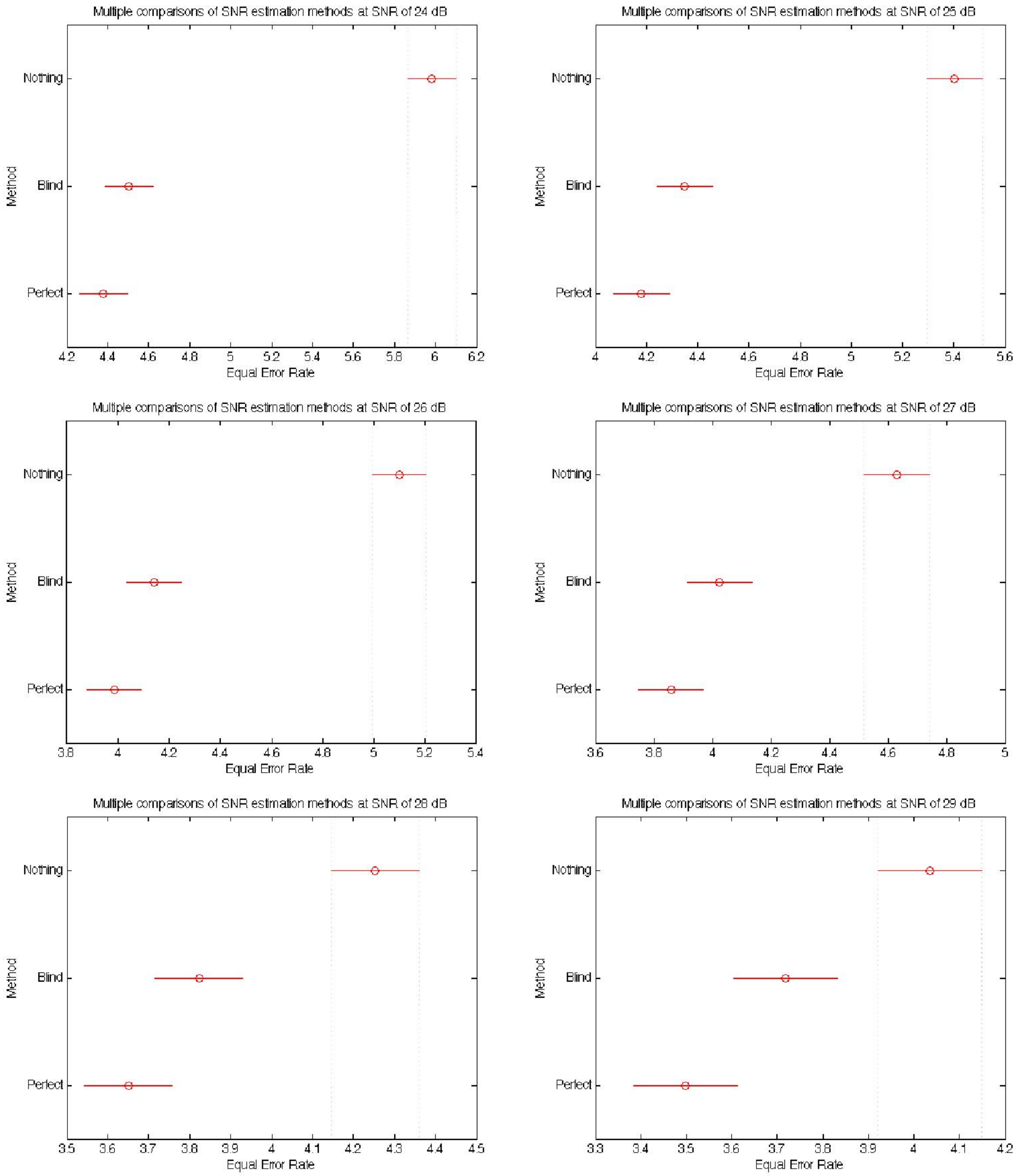

Figure 16. GMM-UBM SNR estimation mode for 24 dB to 29 dB.

Figure 16 contains the multiple comparison plots for the GMM-UBM SNR estimation mode trials for SNRs ranging from 24 dB to 29 dB. Once again, blind and perfect SNR estimation are not statistically different, and both are statistically different from no enhancement in all cases.

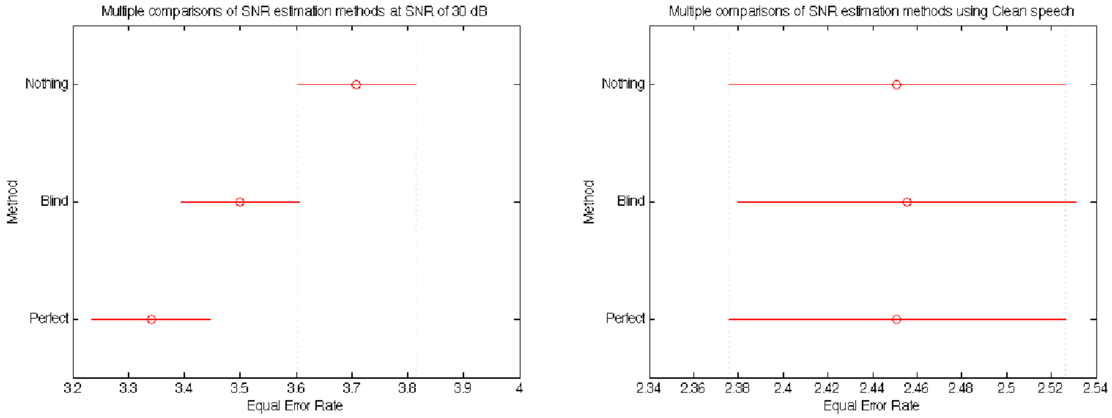

Figure 17. GMM-UBM SNR estimation mode for 30 dB and clean speech.

Finally, Figure 17 displays the multiple comparison plots for the GMM-UBM SNR estimation mode trials for speech corrupted at 30 dB and for clean speech. In the case of the 30 dB SNR trials, there is no statistically significant difference between blind and perfect SNR estimation. There is also no statistically significant difference between blind SNR estimation and no enhancement, but there is a statistically significant difference between perfect SNR estimation and no enhancement. Even though there is no statistically significant difference between blind SNR estimation and no enhancement in this case, it is still recommended to use blind SNR estimation since the two methods were statistically different at every other SNR.

When the factors were tested using clean speech, there was no statistically significant difference between any of the methods. This is both expected and desirable, as this shows that the SNR estimation technique in combination with the affine transform will not cause the performance of the system to become significantly worse when clean test utterances are used.

Table 6 summarizes the results from the GMM-UBM SNR estimation mode trials. The total number of times each SNR estimation mode was statistically shown to be one of the best configurations at a specific SNR is included. This shows that out of the 32 trials with SNRs ranging from 0 dB to 30 dB plus clean speech, blind SNR estimation was one of the best performing methods 29 times. Only in the case of clean speech was blind SNR estimation statistically shown to be no different than doing no enhancement at all. Considering all of the trials, blind SNR estimation was one of the best methods in 90.6% of the trials for the GMM-UBM system.

Table 6

Best signal to noise ratio estimation modes for GMM-UBM trials at all SNRs.

SNR Estimation Mode	Appeared in Best Set	Total Count
Nothing	Clean	1
Blind	0 – 14 dB, 17 – 29 dB, Clean	29
Perfect	0 – 30 dB, Clean	32

Note: The column labeled Appeared in Best Set indicates the SNR trials for which each SNR estimation mode was selected as one of the optimal configurations.

4.2.1.2 GSV-PLS enhancement results. The results for the GSV-PLS enhancement trials are provided. Each plot shows the 95% confidence intervals for the three compared configurations at a specific SNR, based on performing 10 trials per configuration. A summary of the best SNR estimation modes is provided at the end of the section.

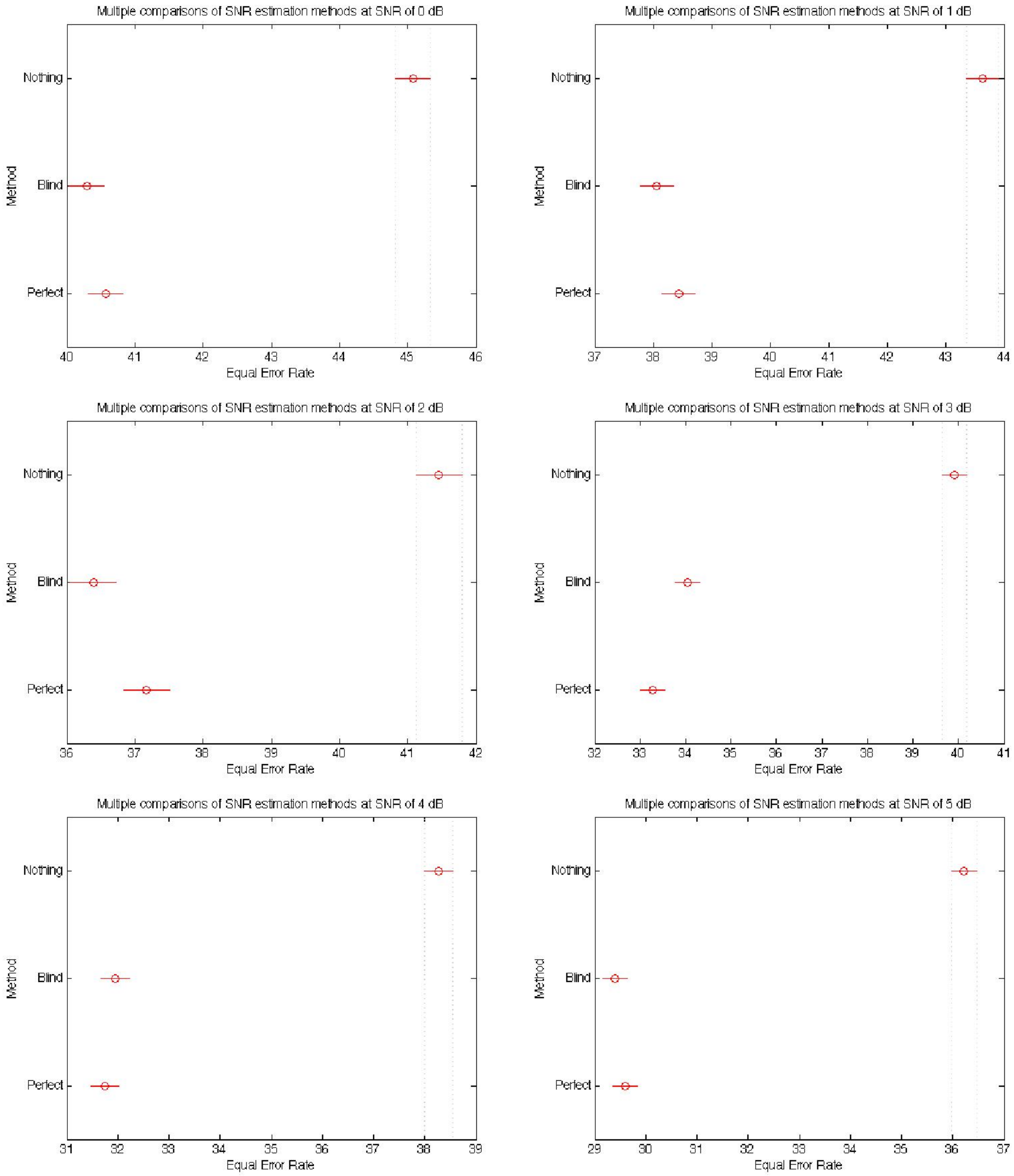

Figure 18. GSV-PLS SNR estimation mode for 0 dB to 5 dB.

Figure 18 shows the 95% confidence intervals for the GSV-PLS SNR estimation mode trials for SNRs ranging from 0 dB to 5 dB. For SNRs of 0, 1, 4, and 5 dB, the difference in the means of the EERs for the blind and perfect SNR estimation trials was not statistically significant. In each case, the difference in the means of EERs from both SNR estimation trials to the means of EERs using no enhancement were statistically significant. At an SNR of 3 dB, both SNR estimation methods are statistically superior to no enhancement, but perfect SNR estimation is shown to be better than blind SNR estimation. Such a scenario is to be expected sometimes, but it is still relatively rare (though not as rare as with the GMM-UBM classifier).

What was surprising was that for an SNR of 2 dB, blind SNR estimation is shown to be statistically better than perfect SNR estimation. Upon further investigation, such a situation can arise when the inherent error involved in applying the affine transform works in the favor of the classifier. Due to the randomness of the speech signals used to train the affine transforms, and the fact that those training utterances will vary from the testing utterances to which the transforms are applied, the affine transform will never be able to totally eliminate the training to testing mismatch. Given a large enough sample of trials, occasionally applying an affine transform not trained on the exact SNR at which the test utterance is corrupted will cause a better reduction in mismatch than the “correct” affine transform would.

To validate this idea, a single trial where the blind SNR estimation caused the “wrong” affine transform to be applied was compared against the same trial with the “correct” affine transform applied. The MFCC feature extracted from a corrupted test utterance was enhanced using both the “correct” and “wrong” affine transforms to generate two different features vectors. Both of these were tried against a single speaker model to whom they did belong and against a single speaker model to whom they did not. The result was that the “wrong” affine transform generated a higher genuine score and a lower impostor score. This means that the error in the blind SNR estimation produced a better result for this one trial than the “correct” affine transform applied using perfect SNR estimation did. Since this can happen on a trial-by-trial basis, given enough trials it is possible for the blind SNR estimation to result in a lower EER than perfect SNR estimation. In the table further on where the average EERs for all of the trials done for this experiment are tabulated, occasionally the blind SNR estimation has a lower average EER than the perfect SNR estimation; however, it is usually not a statistically significant difference. For this scenario to occur enough to cause a statistically significant increase in performance for the blind SNR estimator over the control condition is exceedingly rare, and in fact the GSV-PLS trial at 2 dB was the only time it was observed.

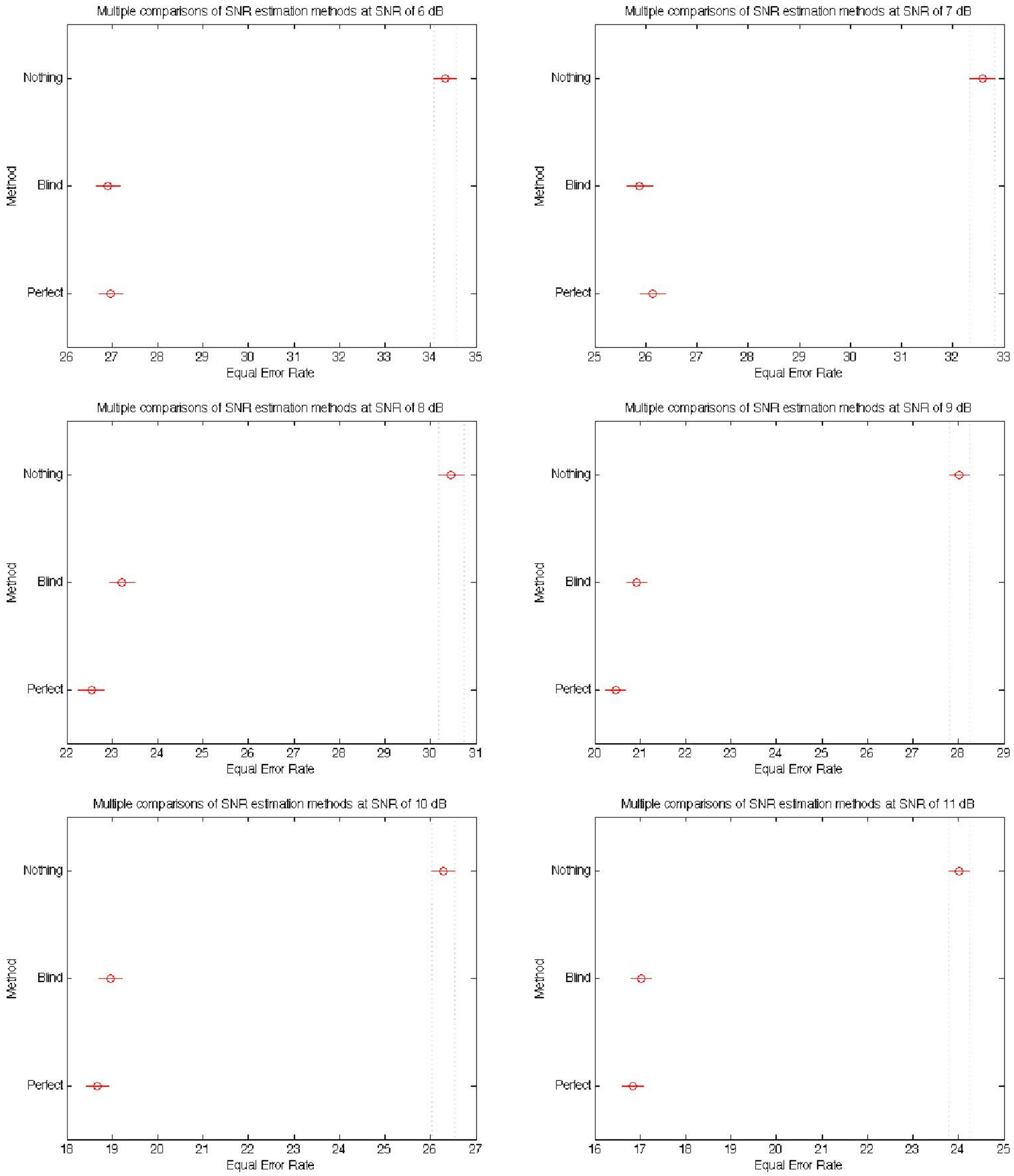

Figure 19. GSV-PLS SNR estimation mode for 6 dB to 11 dB.

Figure 19 shows the 95% confidence intervals for the GSV-PLS SNR estimation trials for SNRs ranging from 6 dB to 11 dB. For SNRs of 6, 7, 10, and 11 dB, the blind and perfect SNR estimation methods are not statistically different, but both are better than no enhancement. For SNRs of 8 dB and 9 dB, the differences in means between both SNR estimation modes and no enhancement are statistically significant. Additionally, perfect SNR estimation is statistically shown to perform better than blind SNR estimation in these two cases.

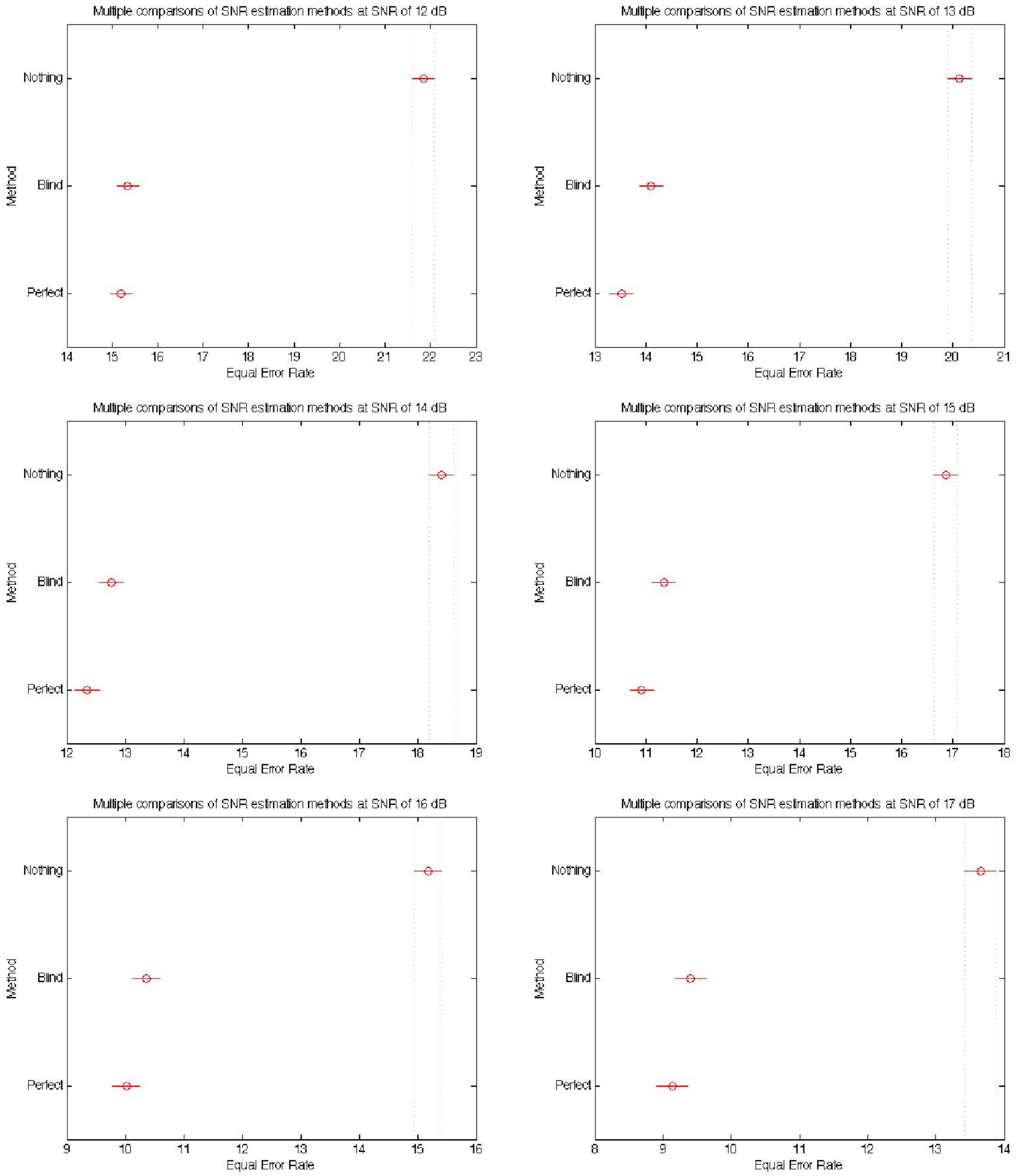

Figure 20. GSV-PLS SNR estimation mode for 12 dB to 17 dB.

Figure 20 shows the 95 % confidence intervals for the GSV-PLS SNR estimation mode trials for SNRs ranging from 12 dB to 17 dB. For SNRs of 12, 15, 16, and 17 dB, the blind and perfect SNR estimation methods are not statistically different, but both are better than no enhancement. For SNRs of 13 dB and 14 dB, the differences in means between both SNR estimation modes and no enhancement are statistically significant. Additionally, perfect SNR estimation is statistically shown to perform better than blind SNR estimation in these two cases.

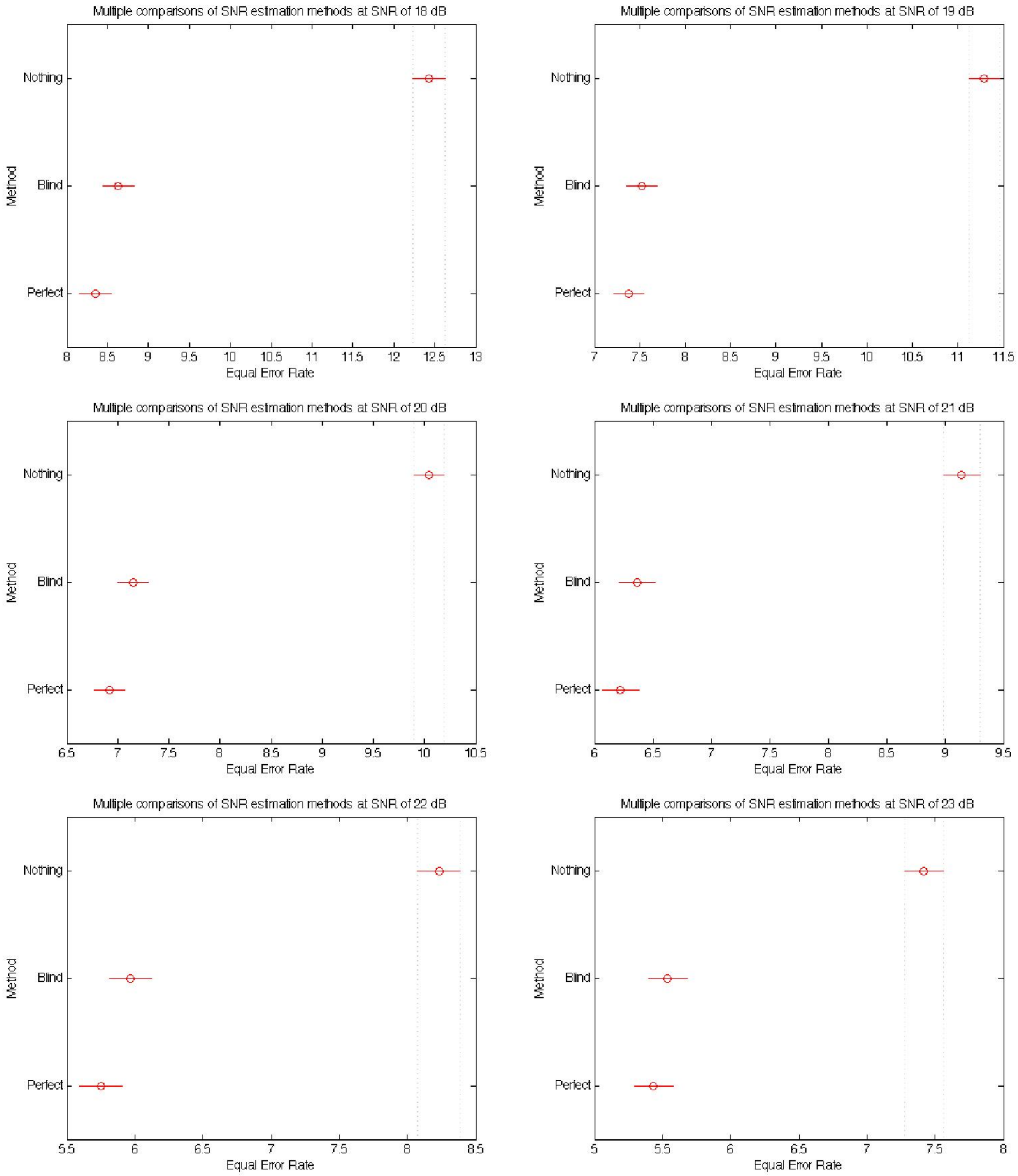

Figure 21. GSV-PLS SNR estimation mode for 18 dB to 23 dB.

Figure 21 shows the 95 % confidence intervals for the GSV-PLS SNR estimation mode trials for SNRs ranging from 18 dB to 23 dB. In all cases, the difference in the means of the EERs obtained from trials using blind SNR estimation and perfect SNR estimation are not statistically different. Additionally, in all cases, both SNR estimation methods are statistically shown to perform better than no enhancement.

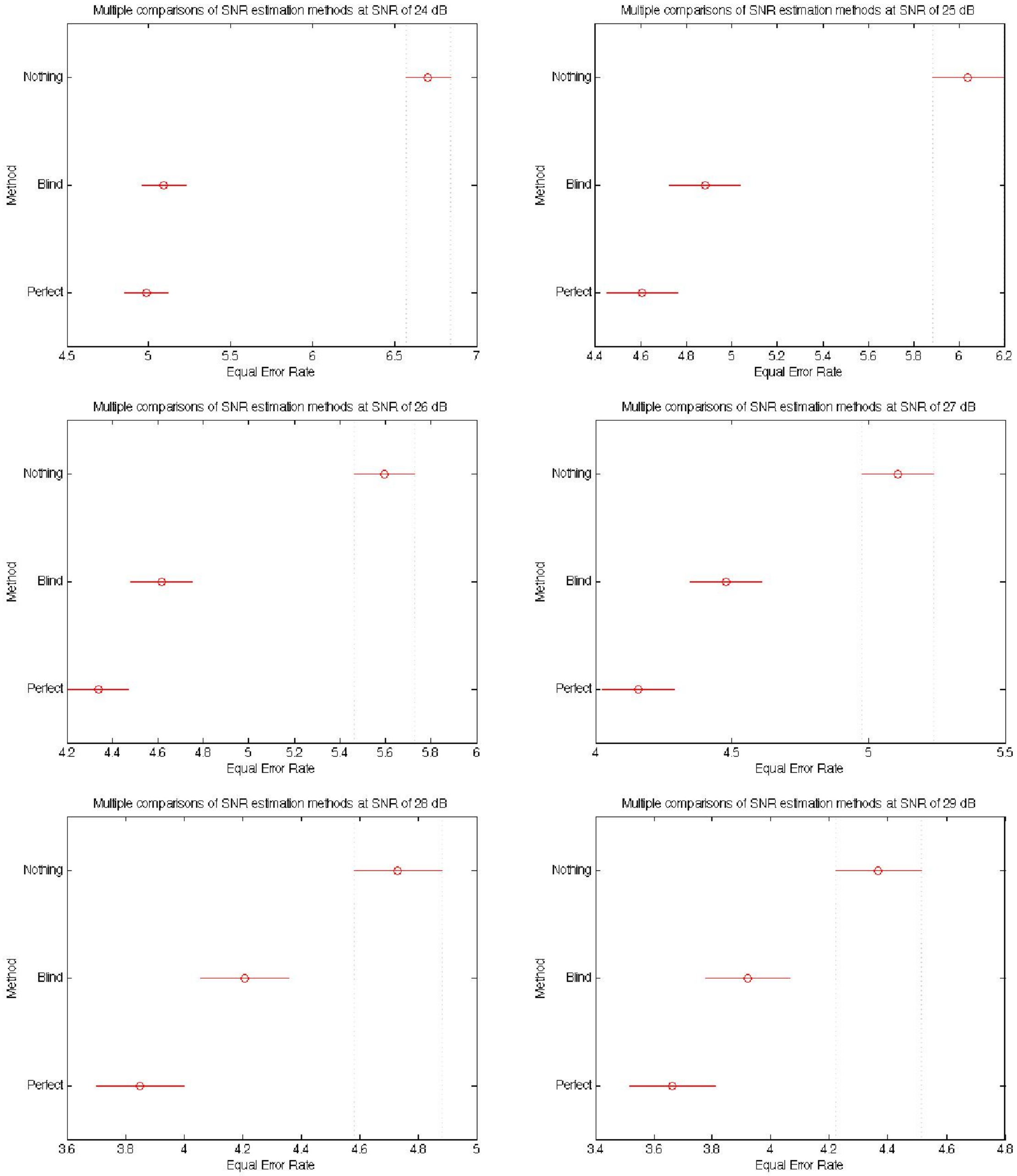

Figure 22. GSV-PLS SNR estimation mode for 24 dB to 29 dB.

Figure 22 shows the 95% confidence intervals for the GSV-PLS SNR estimation mode trials for SNRs ranging from 24 dB to 29 dB. For SNRs of 24, 25, and 29 dB, there was no statistically significant difference in the means of the EERs obtained for trials using blind and perfect SNR estimation, but both were better than doing no enhancement. For SNRs of 26, 27, and 28 dB, perfect SNR estimation was shown to be better than blind SNR estimation, and both were again better than doing no enhancement.

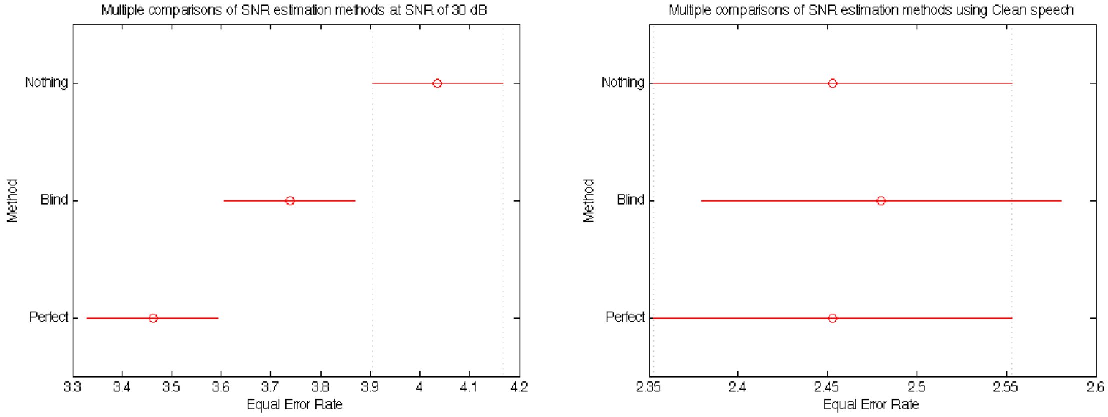

Figure 23. GSV-PLS SNR estimation mode for 30 dB and clean speech.

Finally, Figure 23 displays the 95% confidence intervals for the GSV-PLS SNR estimation mode trials for speech corrupted at 30 dB and for clean speech. In the case of the 30 dB SNR trials, both SNR estimation modes were shown to be better than doing no enhancement, and perfect SNR estimation was shown to be better than blind SNR estimation.

When the factors were tested using clean speech, there was no statistically significant difference between any of the methods. Again, this is both expected and desirable, as this shows that the SNR estimation technique in combination with the affine transform will not cause the performance of the system to become significantly worse when clean test utterances are used.

Table 7 summarizes the results from the GSV-PLS SNR estimation mode trials. The total number of times each SNR estimation mode was statistically shown to be one of the best configurations at a specific SNR is included. This shows that out of the 32 trials with SNRs ranging from 0 dB to 30 dB plus clean speech, blind SNR estimation was one of the best performing methods 23 times. Only in the case of clean speech was blind SNR estimation statistically shown to be no different than doing no enhancement at all. Considering all of the trials, blind SNR estimation was one of the best methods in 71.9% of the trials for the GSV-PLS system and performed better than the control configuration in one case.

Table 7

Best signal to noise ratio estimation modes for GSV-PLS trials at all SNRs.

SNR Estimation Mode	Appeared in Best Set	Total Count
Nothing	Clean	1
Blind	0 – 2 dB, 4 – 7 dB, 10 – 12 dB, 25 – 25 dB, 29 dB, Clean	23
Perfect	0 – 2 dB, 3 – 30 dB, Clean	31

Note: The column labeled Appeared in Best Set indicates the SNR trials for which each SNR estimation mode was selected as one of the optimal configurations.

4.2.2 Optimal feature selection. A two-way analysis of variance was performed using the EERs obtained from all of the blind vs. perfect vs. no enhancement trials. In addition to the sets of optimal enhancement methods, optimal features and fusions were analyzed using multiple comparison of means. The 95% confidence interval plots of each of the trials at every SNR are provided ahead.

4.2.2.1 GMM-UBM feature selection results. The optimal features of each SNR were analyzed for the GMM-UBM and GSV-PLS systems separately. The results for the GMM-UBM feature selection are provided first, including a summary of these results at the end of the subsection.

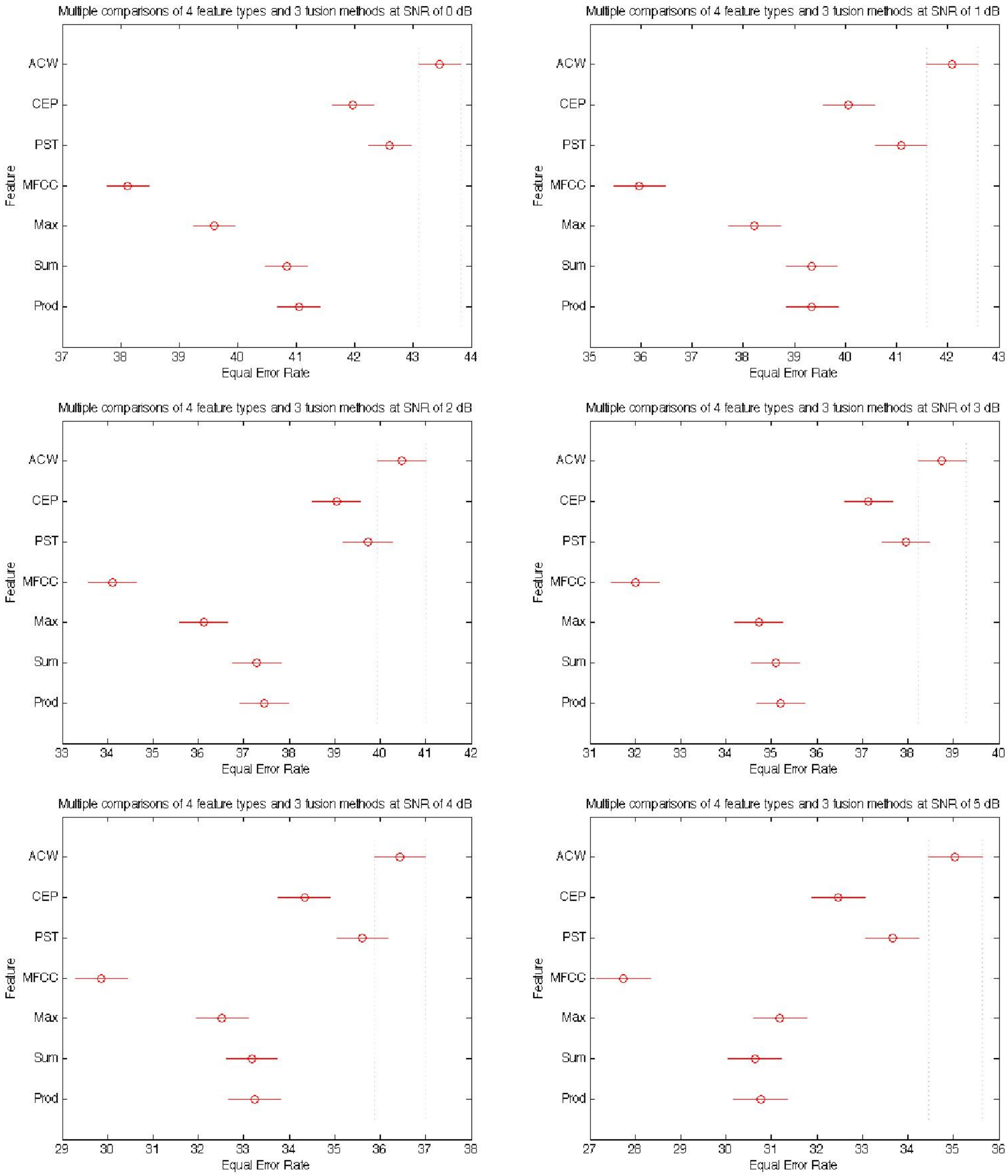

Figure 24. GMM-UBM features and fusions for 0 dB to 5 dB.

Figure 24 shows the 95% confidence intervals for all the features and fusions in the GMM-UBM trials from SNRs ranging from 0 dB to 5 dB. In every case, MFCC outperforms all of the other features and fusions. At low SNRs like this, the low performance of the other features prevents the fusions from giving good results, but this will eventually change as the SNR increases.

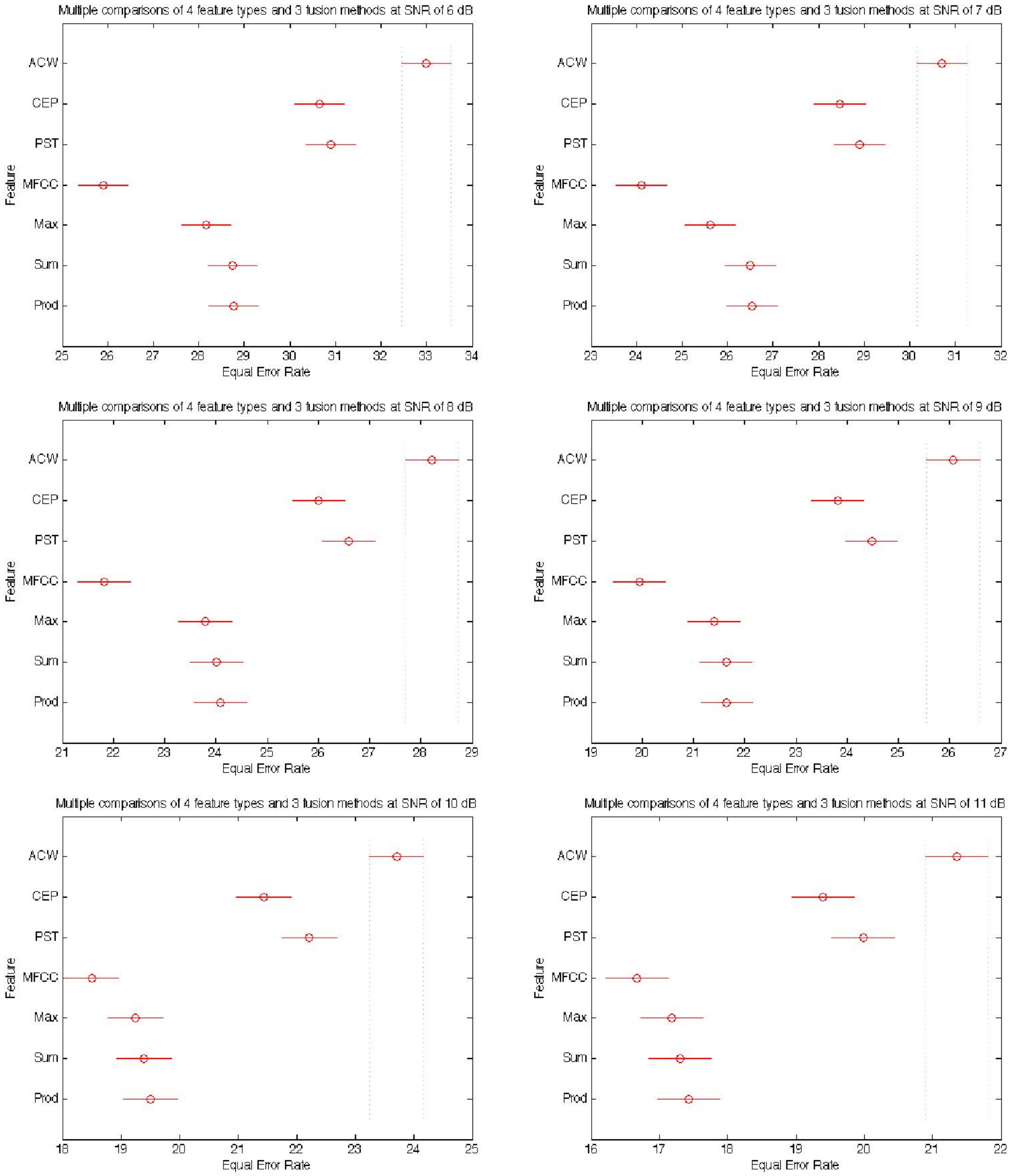

Figure 25. GMM-UBM features and fusions for 6 dB to 11 dB.

Figure 25 shows the 95% confidence intervals for all the features and fusions in the GMM-UBM trials from SNRs ranging from 6 dB to 11 dB. For SNRs ranging from 6 dB to 9 dB, MFCC again performs better than all the other features and fusions. At an SNR of 10 dB, there is no statistically significant difference between the means in the EERs obtained using MFCC, max fusion, and sum fusion. There is also no statistically significant difference between the means in the EERs obtained using any of the fusion strategies, but there is a statistically significant difference between the means in the EERs obtained using MFCC and using product fusion. Therefore, the best set at an SNR of 10 dB is taken to be MFCC, max fusion, and sum fusion. At an SNR of 11 dB, there is no statistically significant difference between the means in the EERs obtained using MFCC or any of the fusion strategies, so all are considered optimal.

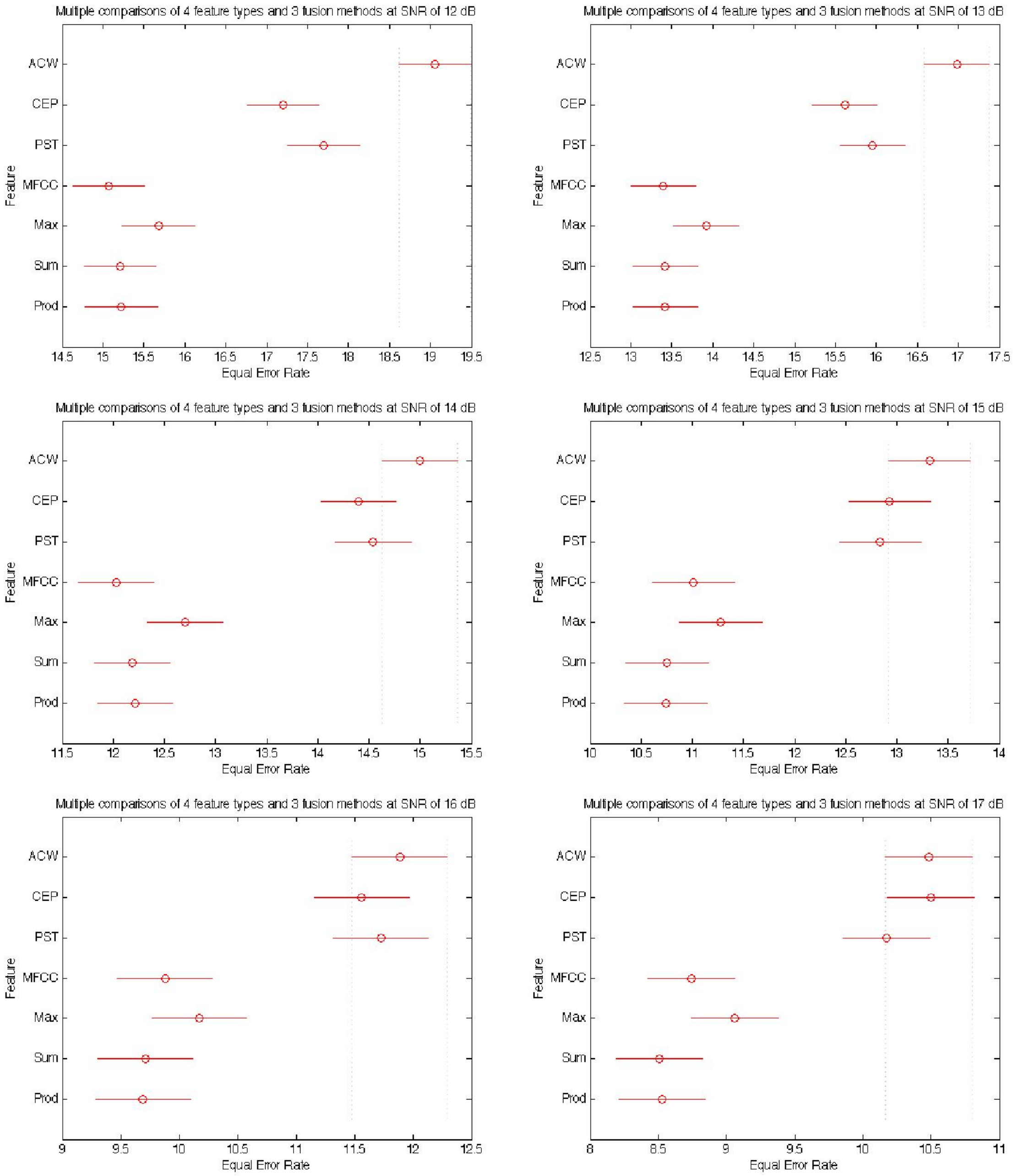

Figure 26. GMM-UBM features and fusions for 12 dB to 17 dB.

Figure 26 shows the 95% confidence intervals for all the features and fusions in the GMM-UBM trials from SNRs ranging from 12 dB to 17 dB. For each of these SNRs, there is no statistically significant difference in the means of the EERs obtained using MFCC or any of the fusion strategies, so all are considered optimal.

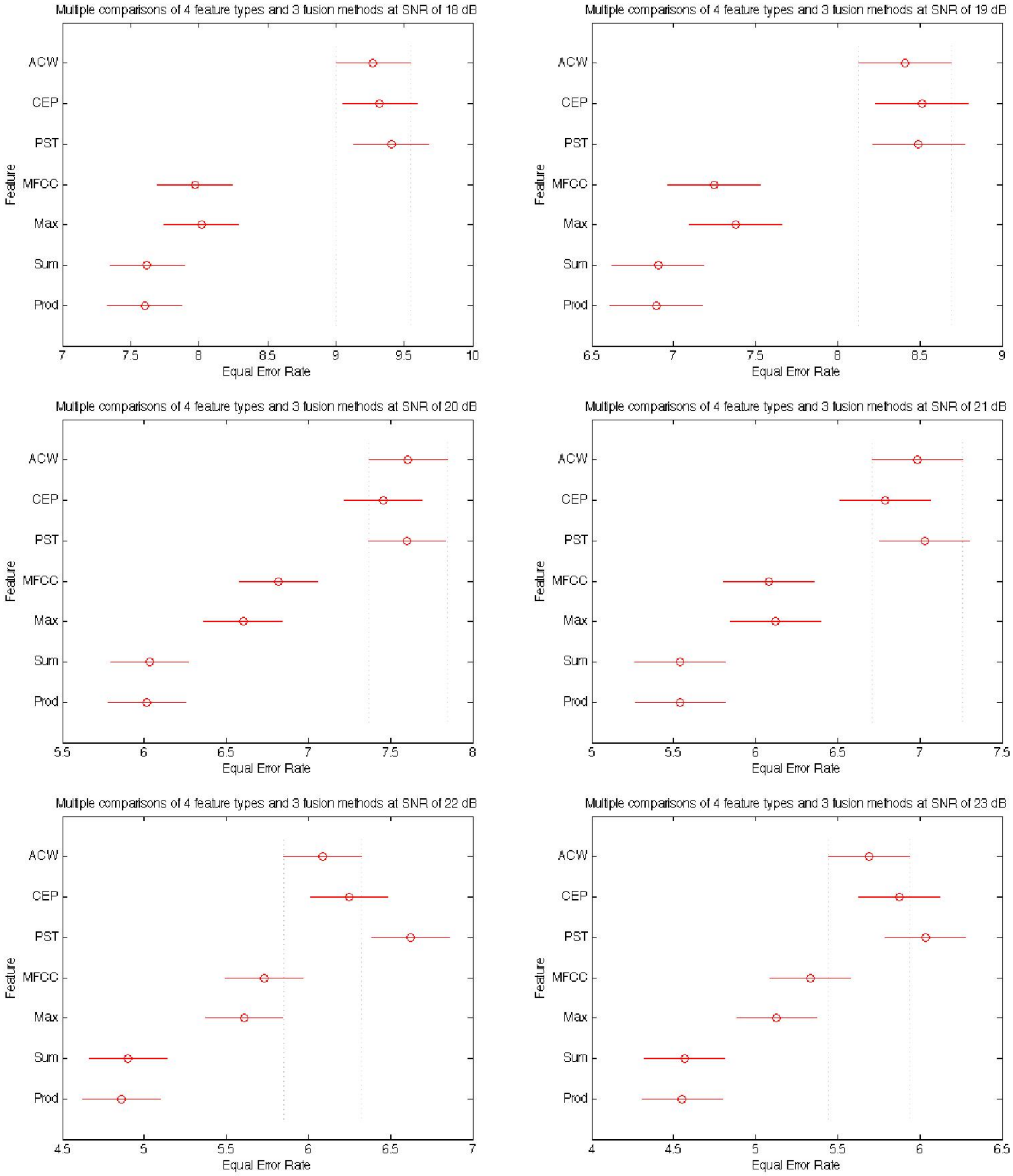

Figure 27. GMM-UBM features and fusions for 18 dB to 23 dB.

Figure 27 shows the 95% confidence intervals for all the features and fusions in the GMM-UBM trials from SNRs ranging from 18 dB to 23 dB. For SNRs of 18 dB and 19 dB, there is no statistically significant difference in the means of the EERs obtained using MFCC and all of the fusion strategies, so all are considered optimal. At an SNR of 21 dB, there is no statistically significant difference between the means of the EERs obtained using MFCC, sum fusion, and product fusion. There is also no statistically significant difference in the means of the EERs obtained using MFCC and maximum fusion; however, there is a statistically significant difference between maximum fusion and the other two fusion strategies. Therefore, sum fusion, and product fusion were chosen as the best features at an SNR of 21 dB. For SNRs of 20, 22, and 23 dB, there is no statistically significant difference in the means of the EERs obtained using sum and product fusion, but these two outperform all other features and fusion strategies and were therefore chosen as the optimal features at these SNRs. As the SNR increases and causes the CEP, ACW, and PFL features to perform better, the fusion strategies can be seen to start outperforming MFCC unlike in the lower SNR trials.

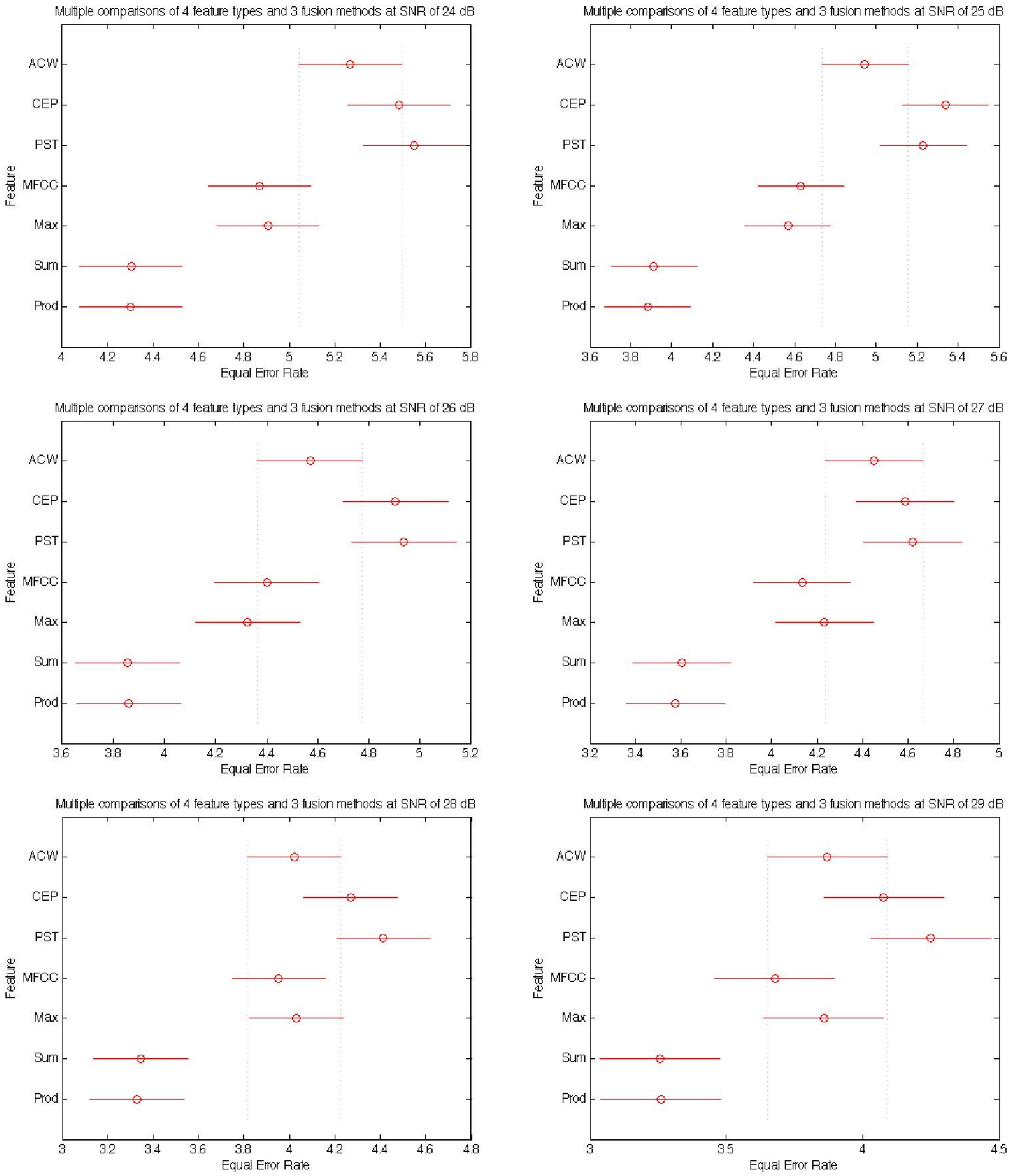

Figure 28. GMM-UBM features and fusions for 24 dB to 29 dB.

Figure 28 shows the 95% confidence intervals for all the features and fusions in the GMM-UBM trials from SNRs ranging from 24 dB to 29 dB. In all cases, sum and product fusion outperform all other features and fusions, and there is no statistically significant difference in the means of the EERs obtained using the two. At an SNR of 29 dB, there are no statistically significant differences in the means of the EERs obtained using sum fusion, product fusion, and MFCC, but there is also no statistically significant difference in the means of the EERs obtained using MFCC and maximum fusion. Because there are statistically significant differences in the means of the EERs obtained using maximum fusion and the other two fusion strategies, sum fusion and product fusion were chosen as the optimal set.

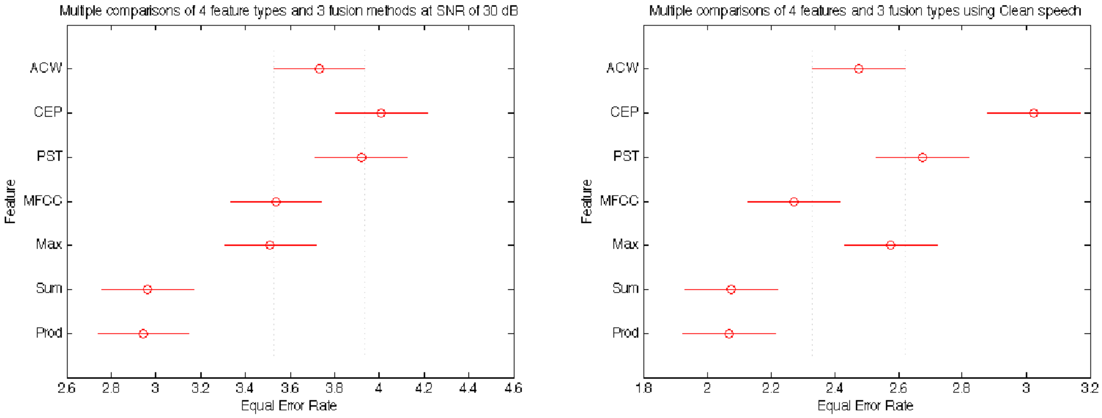

Figure 29. GMM-UBM features and fusions for 30 dB and clean speech.

Figure 29 shows the 95% confidence intervals for all the features and fusions in the GMM-UBM trials where test utterances were corrupted to a 30 dB SNR and were not corrupted at all. At an SNR of 30 dB, sum and product fusion outperform all other features and fusions, and there is no statistically significant difference in the means of the EERs obtained using the two. Using clean speech, there is no statistically significant difference in the means of the EERs obtained using MFCC, sum fusion, and product fusion. However, there was also no statistically difference in the means of the EERs obtained using MFCC and ACW. Because there were statistically significant differences in the means of the EERs obtained using sum and product fusion and using the ACW feature, sum and product fusion were selected as the optimal features using clean speech.

Table 8

Summary of optimal feature selection for GMM-UBM classifier.

Feature/Fusion	Appeared in Best Set	Total Count
ACW	None	0
CEP	None	0
PFL/PST	None	0
MFCC	0 – 19 dB	20
Max Fusion	10 – 19 dB	10
Sum Fusion	10 – 30 dB, Clean	22
Product Fusion	11 – 30, dB Clean	21

Note: The column labeled Appeared in Best Set indicates the SNR trials for which each feature was selected as one of the optimal features.

Table 8 shows that there was not much difference between the system performance obtained using MFCC, sum fusion, and product fusion. Any of these features would be a good choice to use for the GMM-UBM system. The MFCC feature performed best at low to middling SNR levels. The fusion strategies began to surpass the MFCC feature at middling to high SNR levels, where the ACW, CEP, and PFL features were also able to perform better. As sum fusion appeared in the best set for the GMM-UBM trials most numerously (68.8% of all trials), this fusion strategy was chosen as the optimal feature to use when comparing the overall system performance of the GMM-UBM and GSV-PLS classifiers.

4.2.2.2 GSV-PLS feature selection results. The optimal features selection results for the GSV-PLS system are provided henceforth. A summary of these results is included at the end of the subsection.

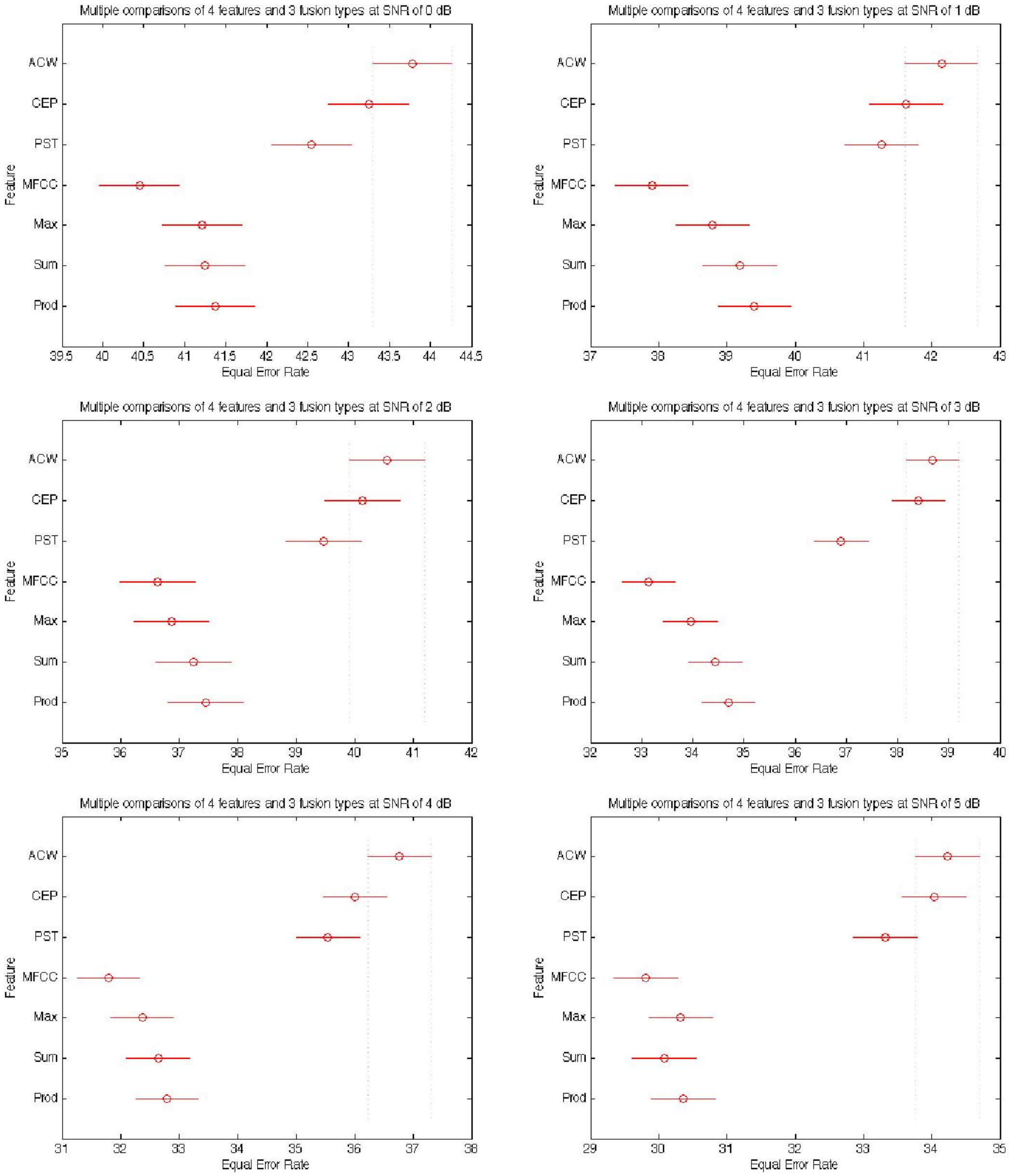

Figure 30. GSV-PLS features and fusions for 0 dB to 5 dB.

Figure 30 shows the 95% confidence intervals for all the features and fusions in the GSV-PLS trials from SNRs ranging from 0 dB to 5 dB. For each of these SNRs, there was no statistically significant difference in the means of the EERs obtained using any of the fusion methods. For SNRs of 0, 2, 4, and 5 dB, there was no statistical difference in the means of the EERs obtained using MFCC and any of the fusion strategies, so all four of these were taken as the optimal set. For SNRs of 1 dB and 3 dB, there were no statistically significant differences between the means of the EERs obtained using MFCC and the sum and maximum fusion strategies, but there was a statistically significant difference in the means of the EERs obtained using MFCC and the product fusion strategy. For these two SNRs, MFCC was taken to be the best feature. Contrary to the low SNR trials with the GMM-UBM system, the fusion strategies evidently perform almost as well as the MFCC feature even at low SNRs. This indicates that there is a smaller difference in the performance of MFCC compared to the other three features in the GSV-PLS system.

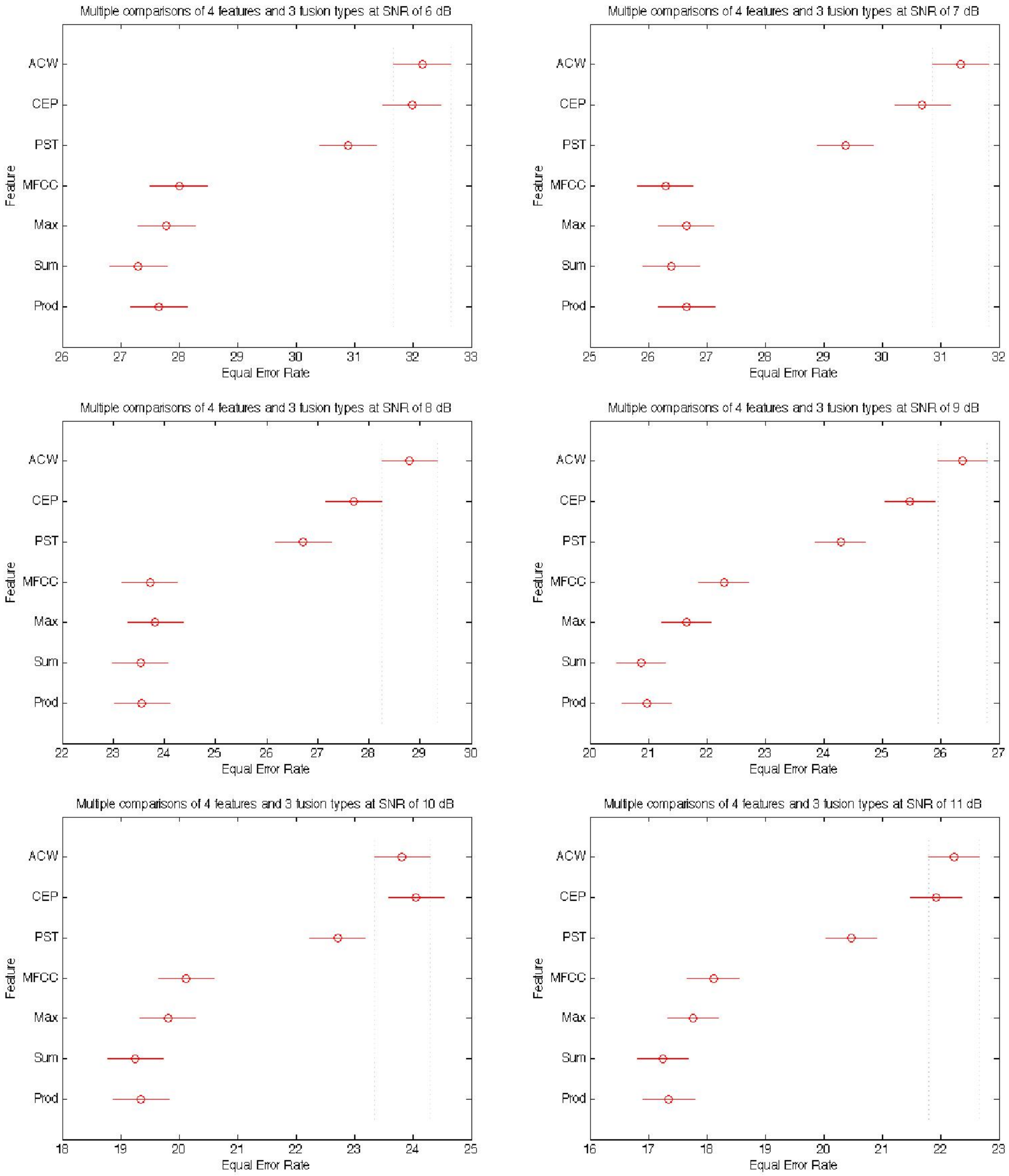

Figure 31. GSV-PLS features and fusions for 6 dB to 11 dB.

Figure 31 shows the 95% confidence intervals for all the features and fusions in the GSV-PLS trials from SNRs ranging from 6 dB to 11 dB. For SNRs of 6, 7, 8, 10, and 11 dB, there is no statistically significant difference between the means of the EERs obtained using MFCC or any of the fusion strategies; therefore, all are taken as optimal for these SNRs. At an SNR of 9 dB, there is no statistically significant difference between the means of the EERs obtained using any of the fusion strategies. There is also no statistically significant difference between the means of the EERs obtained using maximum fusion and the MFCC feature; however, there are statistically significant differences between the means of the EERs obtained using the MFCC feature and sum and product fusion. For an SNR of 9 dB, sum and product fusion were selected as the optimal features.

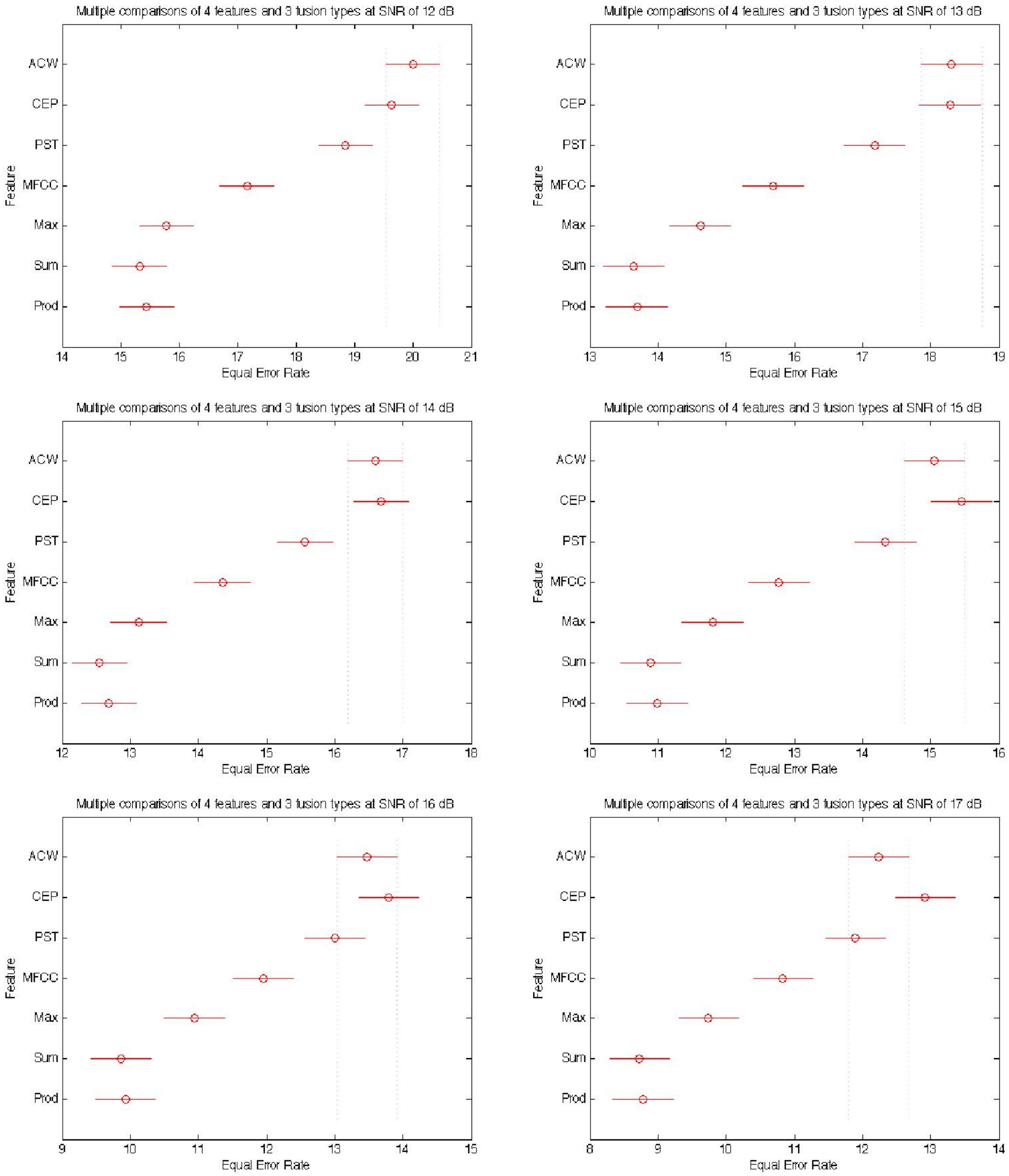

Figure 32. GSV-PLS features and fusions for 12 dB to 17 dB.

Figure 32 shows the 95% confidence intervals for all the features and fusions in the GSV-PLS trials from SNRs ranging from 12 dB to 17 dB. For SNRs of 12 dB and 14 dB, there is no statistically significant difference between the means of the EERs obtained using any of the fusion strategies, so all are taken to be optimal. At SNRs of 13, 16, and 17 dB, there is no statistically significant difference between the means of the EERs obtained using sum and product fusion, and these two methods outperform all others. At an SNR of 15 dB, there is no statistically significant difference between the means of the EERs obtained using sum and product fusion. There is also no statistically significant difference between the means of the EERs obtained using maximum and product fusion; however, because there is a statistically significant difference in the means of the EERs obtained using sum and maximum fusion, sum fusion is taken to be the best feature at an SNR of 15 dB.

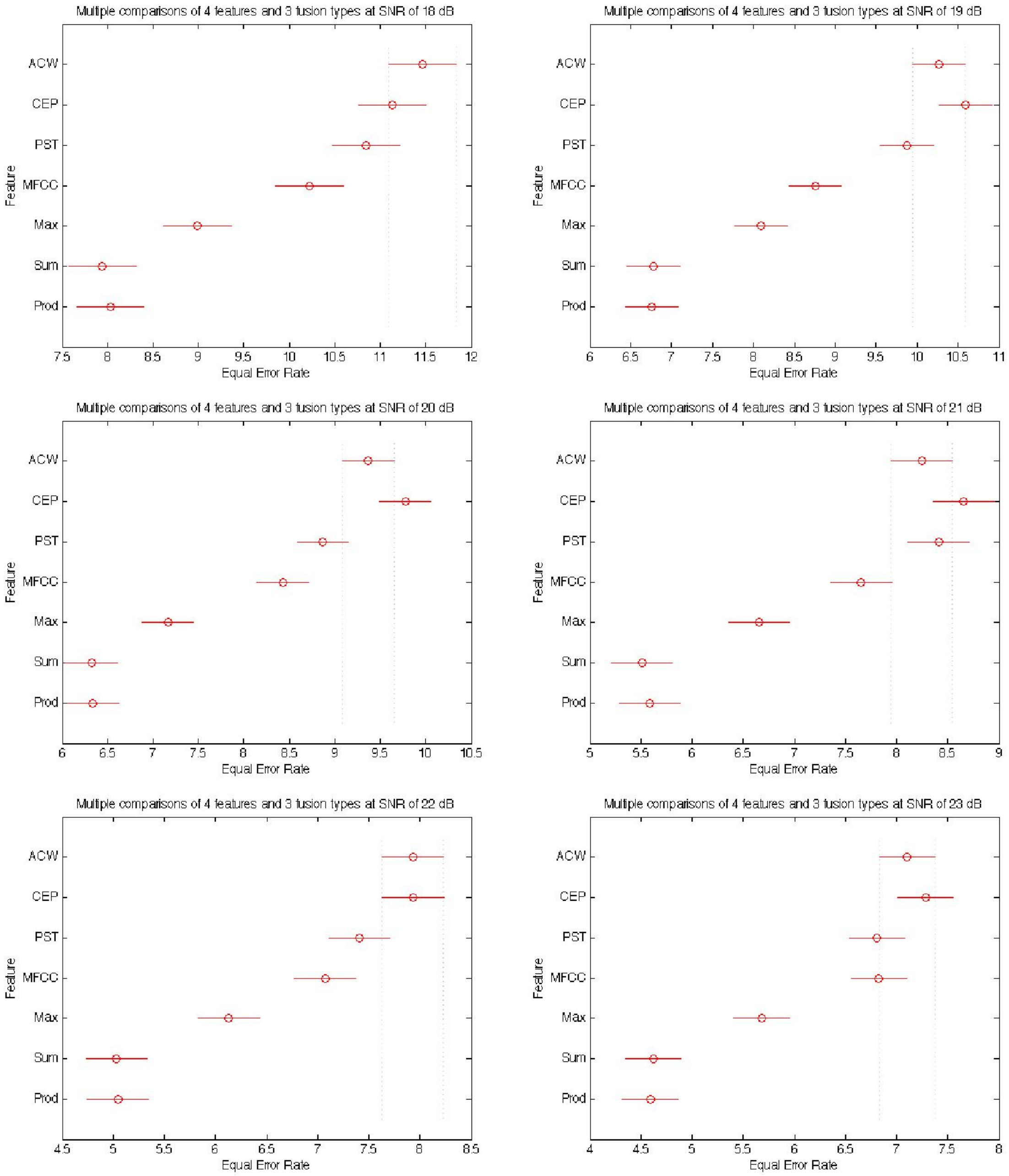

Figure 33. GSV-PLS features and fusions for 18 dB to 23 dB.

Figure 33 shows the 95% confidence intervals for all the features and fusions in the GSV-PLS trials from SNRs ranging from 18 dB to 23 dB. At each of these SNR levels, there is no statistically significant difference between the means of the EERs obtained using sum and product fusion. Both of these fusion strategies outperform all other features and fusions at each of the SNRs from 18 dB to 23 dB, and so were chosen as the optimal features.

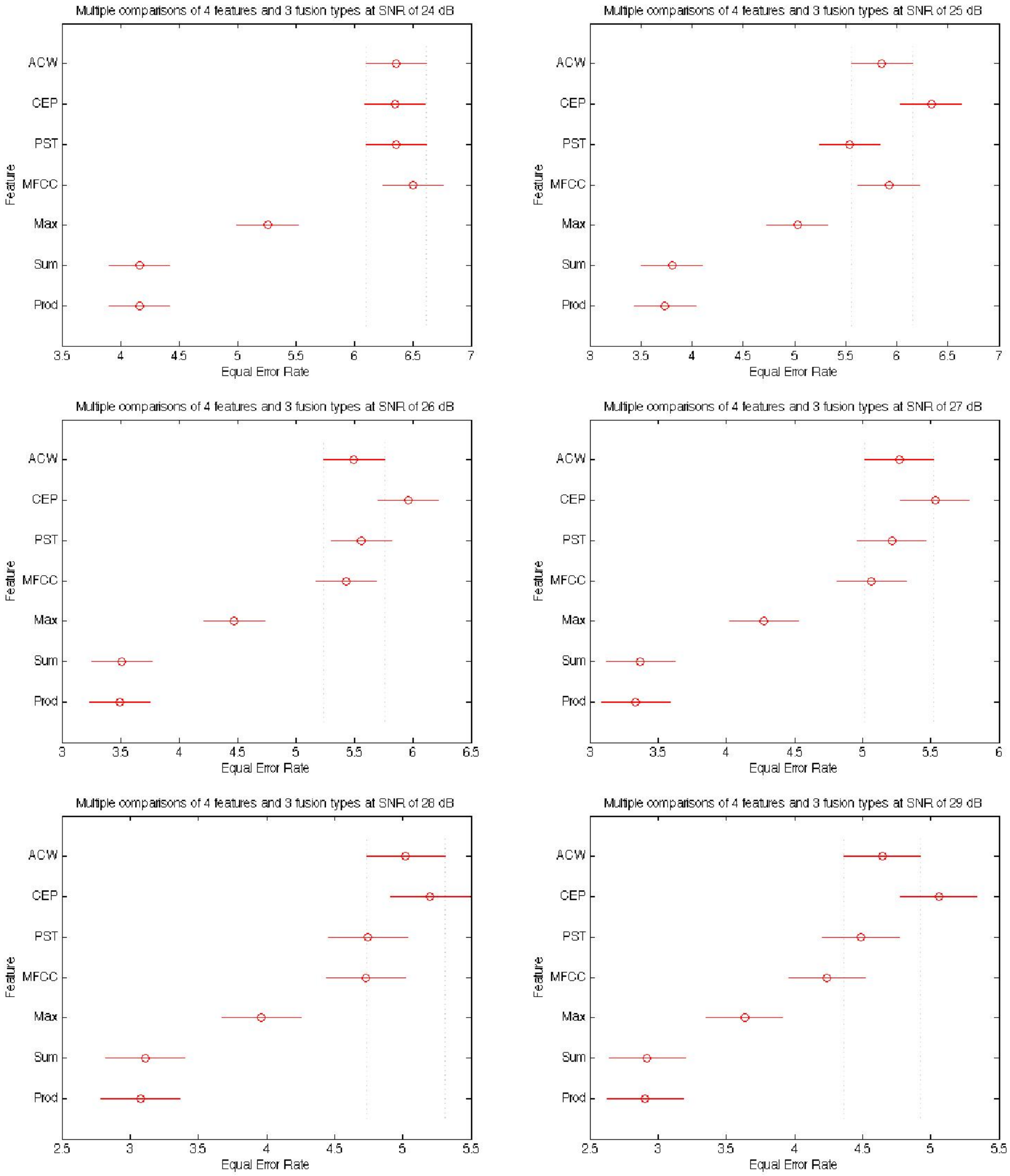

Figure 34. GSV-PLS features and fusions for 24 dB to 29 dB.

Figure 34 shows the 95% confidence intervals for all the features and fusions in the GSV-PLS trials from SNRs ranging from 24 dB to 29 dB. At each of these SNR levels, there is no statistically significant difference between the means of the EERs obtained using sum and product fusion. Both of these fusion strategies outperform all other features and fusions at each of the SNRs from 24 dB to 29 dB, and so were chosen as the optimal features.

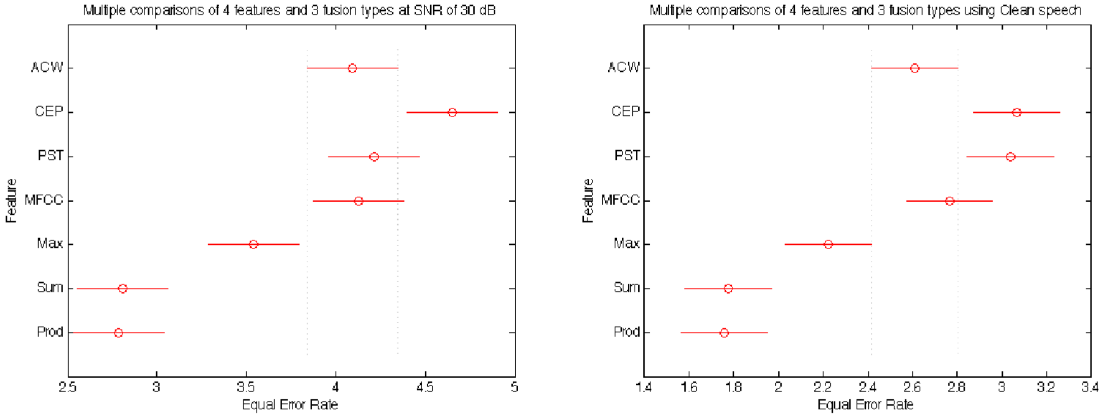

Figure 35. GSV-PLS features and fusions for 30 dB and clean speech.

Figure 35 shows the 95% confidence intervals for all of the features and fusions in the GSV-PLS trials where the test utterances were corrupted to an SNR of 30 dB and were not corrupted at all. In both cases, there is no statistically significant difference between the means of the EERs obtained using sum and product fusion, and both of these methods outperform all other features and fusions.

Table 9

Summary of optimal feature selection for GSV-PLS classifier.

Feature/Fusion	Appeared in Best Set	Total Count
ACW	None	0
CEP	None	0
PFL/PST	None	0
MFCC	0 – 8 dB, 10 dB, 11 dB	11
Max Fusion	0 dB, 2 dB, 4 – 8 dB, 10 – 12 dB, 14 dB	11
Sum Fusion	0 dB, 2 dB, 4 – 30 dB, Clean	30
Product Fusion	0 dB, 2 dB, 4 – 14 dB, 16 – 30 dB, Clean	29

Note: The column labeled Appeared in Best Set indicates the SNR trials for which each feature was selected as one of the optimal features.

Table 9 illustrates how sum fusion and product fusion outperformed all of the other features and fusions for the GSV-PLS trials. Compared to the GMM-UBM trials, sum fusion and product fusion were much more dominant, appearing in the best set in 93.8% and 90.6% of all the trials, respectively. This occurred because at lower SNR levels, there was less of a stark difference in the performance of the MFCC feature compared to ACW, CEP, and PFL. Whereas the GMM-UBM trials saw the MFCC feature performing best at lower SNRs, with the GSV-PLS trials the performance of the fusion methods was comparable and occasionally better than the MFCC feature even at lower SNRs. Because the sum fusion strategy again appeared in the optimal feature set most numerously, it was chosen for the GSV-PLS classifier's optimal configuration to be compared against the GMM-UBM classifier.

4.2.3 Tabulation of results from enhancement method trials. This subsection contains a full tabulation of the average equal error rates obtained for the enhancement method trials. Average EERs calculated over 10 rotations are provided for every test condition, every feature and fusion type, and all SNR estimation modalities. The two tables that follow pertain to the results obtained for each classifier.

Table 10

Average equal error rates for individual features, fusions, and signal to noise ratio estimation modes for GMM-UBM trials.

Test Condition	Feature/Fusion	Nothing	Blind	Perfect
0 dB SNR	ACW	45.15	42.60	42.65
	CEP	45.47	40.13	40.32
	PFL	44.90	41.38	41.55
	MFCC	43.55	35.26	35.56
	Max	44.84	36.89	37.08
	Sum	44.70	38.86	38.96
	Product	44.81	38.99	39.34
1 dB SNR	ACW	44.38	40.97	40.93
	CEP	43.92	38.09	38.18
	PFL	43.86	39.58	39.84
	MFCC	42.41	32.67	32.85
	Max	44.03	35.34	35.31
	Sum	43.48	37.16	37.37
	Product	43.53	37.22	37.30
2 dB SNR	ACW	43.20	39.10	39.14
	CEP	43.34	36.73	37.05
	PFL	42.65	38.23	38.30
	MFCC	41.62	30.23	30.47
	Max	42.69	32.83	32.81
	Sum	42.42	34.68	34.74
	Product	42.46	34.86	35.01
3 dB SNR	ACW	41.70	37.28	37.28
	CEP	41.71	34.84	34.86
	PFL	41.58	36.14	36.16
	MFCC	40.10	27.78	28.12
	Max	41.25	31.35	31.54
	Sum	40.65	32.18	32.44
	Product	40.60	32.39	32.62
4 dB SNR	ACW	40.23	34.45	34.64
	CEP	40.18	31.08	31.74
	PFL	40.43	33.21	33.18
	MFCC	38.87	25.43	25.30
	Max	39.90	28.79	28.89
	Sum	39.73	29.82	29.97
	Product	39.78	29.99	29.92
5 dB SNR	ACW	39.22	32.94	32.97
	CEP	39.11	28.93	29.37
	PFL	38.36	31.26	31.36
	MFCC	37.45	22.74	23.03
	Max	38.37	27.49	27.70
	Sum	37.90	26.88	27.13
	Product	37.77	27.06	27.43
6 dB SNR	ACW	37.65	30.81	30.52
	CEP	37.58	27.09	27.31
	PFL	36.32	28.23	28.14
	MFCC	36.22	20.79	20.72
	Max	36.97	23.75	23.75
	Sum	36.53	24.70	25.00
	Product	36.54	24.67	25.07

Table 10 (continued)

Test Condition	Feature/Fusion	Nothing	Blind	Perfect
7 dB SNR	ACW	36.01	28.34	27.79
	CEP	36.05	24.89	24.47
	PFL	35.08	25.97	25.64
	MFCC	33.95	19.21	19.16
	Max	34.85	21.10	20.91
	Sum	34.55	22.50	22.45
	Product	34.51	22.61	22.48
8 dB SNR	ACW	33.48	25.74	25.44
	CEP	33.57	22.43	22.03
	PFL	33.49	23.34	22.97
	MFCC	32.21	16.80	16.46
	Max	32.85	19.49	19.03
	Sum	32.52	19.68	19.83
	Product	32.58	19.79	19.89
9 dB SNR	ACW	31.82	23.13	23.26
	CEP	31.90	19.93	19.61
	PFL	31.02	21.27	21.15
	MFCC	29.77	15.30	14.76
	Max	30.76	16.81	16.63
	Sum	30.39	17.37	17.15
	Product	30.23	17.43	17.30
10 dB SNR	ACW	30.00	20.50	20.64
	CEP	29.66	17.44	17.21
	PFL	29.43	18.85	18.38
	MFCC	27.90	13.97	13.64
	Max	28.78	14.70	14.28
	Sum	28.34	15.02	14.81
	Product	28.39	15.21	14.94
11 dB SNR	ACW	27.47	18.37	18.22
	CEP	27.37	15.57	15.24
	PFL	26.53	16.98	16.44
	MFCC	25.71	12.30	12.00
	Max	26.05	13.05	12.43
	Sum	25.39	13.47	13.07
	Product	25.37	13.63	13.30
12 dB SNR	ACW	25.32	16.08	15.76
	CEP	25.21	13.42	12.97
	PFL	24.31	14.45	14.33
	MFCC	23.00	11.35	10.87
	Max	24.11	11.63	11.29
	Sum	23.14	11.26	11.22
	Product	23.04	11.35	11.29
13 dB SNR	ACW	22.70	14.26	14.00
	CEP	23.21	12.05	11.59
	PFL	22.31	12.98	12.56
	MFCC	20.54	9.99	9.65
	Max	21.44	10.33	10.00
	Sum	20.68	9.93	9.65
	Product	20.63	9.93	9.69

Table 10 (continued)

Test Condition	Feature/Fusion	Nothing	Blind	Perfect
14 dB SNR	ACW	20.90	12.10	11.99
	CEP	20.92	11.32	10.95
	PFL	20.05	12.06	11.51
	MFCC	18.34	9.17	8.58
	Max	19.73	9.28	9.10
	Sum	18.39	9.31	8.87
	Product	18.26	9.41	8.98
15 dB SNR	ACW	18.89	10.82	10.25
	CEP	18.52	10.38	9.88
	PFL	17.72	10.73	10.06
	MFCC	16.56	8.42	8.06
	Max	17.18	8.47	8.18
	Sum	16.34	8.20	7.71
	Product	16.27	8.22	7.74
16 dB SNR	ACW	16.70	9.86	9.11
	CEP	16.54	9.25	8.90
	PFL	15.87	9.87	9.44
	MFCC	14.56	7.70	7.39
	Max	15.18	7.86	7.49
	Sum	14.51	7.54	7.08
	Product	14.44	7.51	7.13
17 dB SNR	ACW	14.73	8.62	8.10
	CEP	14.71	8.60	8.18
	PFL	13.73	8.58	8.20
	MFCC	12.89	6.80	6.54
	Max	13.25	7.07	6.87
	Sum	12.58	6.57	6.38
	Product	12.60	6.64	6.35
18 dB SNR	ACW	13.10	7.57	7.14
	CEP	12.86	7.67	7.43
	PFL	12.27	8.09	7.86
	MFCC	11.16	6.48	6.27
	Max	11.13	6.51	6.41
	Sum	10.99	6.05	5.83
	Product	10.92	6.05	5.84
19 dB SNR	ACW	11.76	6.86	6.61
	CEP	11.67	7.08	6.79
	PFL	11.18	7.15	7.14
	MFCC	9.86	5.94	5.95
	Max	10.09	6.05	5.99
	Sum	9.75	5.55	5.42
	Product	9.77	5.52	5.40
20 dB SNR	ACW	10.36	6.28	6.18
	CEP	10.28	6.17	5.92
	PFL	10.01	6.44	6.34
	MFCC	8.72	6.01	5.72
	Max	8.78	5.69	5.34
	Sum	8.26	4.93	4.90
	Product	8.28	4.89	4.88

Table 10 (continued)

Test Condition	Feature/Fusion	Nothing	Blind	Perfect
21 dB SNR	ACW	9.55	5.72	5.67
	CEP	8.94	5.78	5.64
	PFL	8.87	6.10	6.11
	MFCC	7.77	5.24	5.22
	Max	7.76	5.34	5.25
	Sum	7.41	4.65	4.56
	Product	7.44	4.62	4.57
22 dB SNR	ACW	8.06	5.18	5.02
	CEP	8.05	5.41	5.29
	PFL	8.11	5.96	5.80
	MFCC	7.28	5.00	4.91
	Max	6.96	4.91	4.95
	Sum	6.44	4.10	4.16
	Product	6.43	4.04	4.10
23 dB SNR	ACW	7.31	4.95	4.81
	CEP	7.39	5.13	5.10
	PFL	7.05	5.53	5.52
	MFCC	6.31	4.87	4.81
	Max	6.25	4.66	4.48
	Sum	5.87	3.89	3.94
	Product	5.84	3.89	3.93
24 dB SNR	ACW	6.50	4.78	4.52
	CEP	6.67	4.97	4.80
	PFL	6.40	5.20	5.06
	MFCC	5.77	4.48	4.36
	Max	5.77	4.49	4.45
	Sum	5.35	3.82	3.74
	Product	5.40	3.79	3.72
25 dB SNR	ACW	5.94	4.55	4.34
	CEP	6.43	4.86	4.72
	PFL	6.08	4.85	4.75
	MFCC	5.15	4.61	4.13
	Max	5.02	4.40	4.29
	Sum	4.60	3.61	3.53
	Product	4.59	3.56	3.49
26 dB SNR	ACW	5.43	4.26	4.02
	CEP	5.81	4.59	4.31
	PFL	5.61	4.67	4.53
	MFCC	4.87	4.36	3.97
	Max	4.84	4.11	4.02
	Sum	4.54	3.51	3.52
	Product	4.58	3.48	3.52
27 dB SNR	ACW	5.18	4.15	4.02
	CEP	5.13	4.31	4.32
	PFL	4.95	4.50	4.41
	MFCC	4.44	4.14	3.82
	Max	4.56	4.19	3.95
	Sum	4.10	3.46	3.25
	Product	4.06	3.43	3.24

Table 10 (continued)

Test Condition	Feature/Fusion	Nothing	Blind	Perfect
28 dB SNR	ACW	4.52	3.88	3.66
	CEP	4.55	4.16	4.10
	PFL	4.68	4.34	4.22
	MFCC	4.20	3.93	3.73
	Max	4.32	4.08	3.69
	Sum	3.75	3.19	3.09
	Product	3.75	3.18	3.06
29 dB SNR	ACW	4.39	3.78	3.45
	CEP	4.37	4.03	3.84
	PFL	4.52	4.12	4.11
	MFCC	3.87	3.75	3.42
	Max	4.03	3.86	3.68
	Sum	3.52	3.25	3.00
	Product	3.56	3.24	2.99
30 dB SNR	ACW	4.18	3.57	3.45
	CEP	4.23	4.04	3.76
	PFL	4.00	3.95	3.80
	MFCC	3.56	3.60	3.45
	Max	3.73	3.50	3.30
	Sum	3.15	2.93	2.82
	Product	3.12	2.90	2.80
Clean Speech	ACW	2.49	2.46	2.49
	CEP	3.03	3.01	3.03
	PFL	2.68	2.66	2.68
	MFCC	2.25	2.32	2.25
	Max	2.55	2.62	2.55
	Sum	2.08	2.07	2.08
	Product	2.08	2.05	2.08

Note: The columns labeled Nothing, Blind, and Perfect denote the SNR estimation mode used for each trial. Nothing means that no enhancement was used.

Table 10 contains the average equal error rates taken over 10 rotations for each of the feature, fusions, SNR estimation modes, and test conditions used in the GMM-UBM trials.

Table 11

Average equal error rates for individual features, fusions, and signal to noise ratio estimation modes for GSV-PLS trials.

Test Condition	Feature/Fusion	Nothing	Blind	Perfect
0 dB SNR	ACW	45.64	42.72	42.98
	CEP	45.96	41.77	42.01
	PFL	45.79	40.71	41.14
	MFCC	44.74	38.16	38.44
	Max	44.12	39.64	39.87
	Sum	44.69	39.43	39.62
	Product	44.60	39.61	39.91
1 dB SNR	ACW	44.38	40.92	41.13
	CEP	44.42	40.15	40.31
	PFL	44.26	39.52	40.03
	MFCC	42.95	35.16	35.59
	Max	42.54	36.71	37.13
	Sum	43.31	36.88	37.38
	Product	43.60	37.11	37.51
2 dB SNR	ACW	42.00	39.35	40.30
	CEP	43.02	38.37	39.02
	PFL	42.11	37.73	38.57
	MFCC	40.82	34.21	34.90
	Max	40.15	34.86	35.61
	Sum	41.00	35.02	35.72
	Product	41.12	35.18	36.10
3 dB SNR	ACW	41.29	37.82	36.92
	CEP	41.62	37.07	36.52
	PFL	41.05	35.16	34.49
	MFCC	38.34	30.80	30.26
	Max	39.21	32.00	30.66
	Sum	38.81	32.57	31.94
	Product	39.11	32.89	32.09
4 dB SNR	ACW	39.63	35.45	35.25
	CEP	39.66	34.43	33.94
	PFL	39.84	33.46	33.34
	MFCC	37.23	29.13	29.04
	Max	37.13	30.13	29.86
	Sum	37.17	30.47	30.30
	Product	37.24	30.61	30.55
5 dB SNR	ACW	36.73	32.89	33.08
	CEP	37.90	32.13	32.08
	PFL	37.62	30.89	31.45
	MFCC	35.46	27.00	26.97
	Max	34.62	27.81	28.55
	Sum	35.52	27.38	27.37
	Product	35.71	27.66	27.72
6 dB SNR	ACW	35.52	30.40	30.56
	CEP	36.20	29.93	29.82
	PFL	35.77	28.46	28.47
	MFCC	33.50	25.15	25.37
	Max	32.89	25.09	25.39
	Sum	33.04	24.49	24.39
	Product	33.34	24.85	24.78

Table 11 (continued)

Test Condition	Feature/Fusion	Nothing	Blind	Perfect
7 dB SNR	ACW	34.10	29.76	30.18
	CEP	34.79	28.55	28.74
	PFL	33.71	27.08	27.34
	MFCC	31.26	23.79	23.83
	Max	31.42	24.38	24.14
	Sum	31.19	23.68	24.31
	Product	31.60	23.94	24.43
8 dB SNR	ACW	31.85	27.70	26.83
	CEP	32.84	25.57	24.70
	PFL	31.85	24.48	23.83
	MFCC	29.10	21.34	20.70
	Max	28.66	21.88	20.93
	Sum	29.43	20.77	20.41
	Product	29.47	20.82	20.40
9 dB SNR	ACW	30.62	24.24	24.29
	CEP	30.34	23.14	22.97
	PFL	29.13	21.98	21.76
	MFCC	26.69	20.30	19.88
	Max	26.18	20.00	18.80
	Sum	26.48	18.45	17.70
	Product	26.71	18.41	17.82
10 dB SNR	ACW	27.89	21.98	21.60
	CEP	28.87	21.72	21.59
	PFL	27.67	20.45	20.03
	MFCC	25.79	17.48	17.11
	Max	24.46	17.58	17.39
	Sum	24.62	16.69	16.44
	Product	24.71	16.81	16.55
11 dB SNR	ACW	25.84	20.53	20.32
	CEP	26.67	19.61	19.51
	PFL	25.12	18.27	18.03
	MFCC	23.34	15.59	15.41
	Max	22.07	15.66	15.57
	Sum	22.43	14.78	14.54
	Product	22.69	14.81	14.55
12 dB SNR	ACW	23.92	18.04	18.05
	CEP	24.16	17.52	17.26
	PFL	22.85	17.00	16.72
	MFCC	21.85	15.11	14.53
	Max	20.08	13.51	13.77
	Sum	19.93	13.04	13.00
	Product	20.15	13.16	13.04
13 dB SNR	ACW	21.92	16.82	16.18
	CEP	22.59	16.47	15.78
	PFL	20.84	15.79	14.89
	MFCC	20.02	13.64	13.40
	Max	18.55	12.82	12.48
	Sum	18.42	11.57	10.95
	Product	18.54	11.60	10.94

Table 11 (continued)

Test Condition	Feature/Fusion	Nothing	Blind	Perfect
14 dB SNR	ACW	20.49	14.93	14.37
	CEP	20.44	15.21	14.38
	PFL	19.27	13.95	13.48
	MFCC	18.19	12.54	12.32
	Max	16.55	11.49	11.33
	Sum	16.86	10.62	10.16
	Product	17.05	10.65	10.34
15 dB SNR	ACW	18.95	13.30	12.90
	CEP	19.01	13.96	13.39
	PFL	17.80	12.82	12.40
	MFCC	16.89	10.95	10.50
	Max	15.30	10.21	9.89
	Sum	14.92	9.12	8.63
	Product	15.16	9.11	8.70
16 dB SNR	ACW	17.15	11.75	11.51
	CEP	16.93	12.45	12.01
	PFL	16.27	11.61	11.14
	MFCC	15.20	10.47	10.19
	Max	13.75	9.72	9.35
	Sum	13.36	8.28	7.97
	Product	13.60	8.23	7.95
17 dB SNR	ACW	15.13	10.96	10.62
	CEP	15.92	11.58	11.25
	PFL	14.55	10.69	10.45
	MFCC	13.55	9.64	9.31
	Max	12.18	8.67	8.36
	Sum	12.06	7.18	6.96
	Product	12.19	7.14	7.01
18 dB SNR	ACW	13.76	10.49	10.15
	CEP	13.82	10.05	9.53
	PFL	13.46	9.79	9.29
	MFCC	12.94	8.89	8.83
	Max	11.29	7.88	7.79
	Sum	10.81	6.61	6.42
	Product	10.92	6.71	6.46
19 dB SNR	ACW	12.83	9.06	8.92
	CEP	13.09	9.51	9.17
	PFL	12.59	8.70	8.35
	MFCC	11.24	7.58	7.45
	Max	10.22	7.07	6.98
	Sum	9.50	5.40	5.43
	Product	9.58	5.34	5.34
20 dB SNR	ACW	11.46	8.45	8.20
	CEP	11.91	8.87	8.54
	PFL	10.96	7.91	7.74
	MFCC	10.12	7.65	7.51
	Max	8.96	6.39	6.13
	Sum	8.42	5.41	5.14
	Product	8.48	5.39	5.15

Table 11 (continued)

Test Condition	Feature/Fusion	Nothing	Blind	Perfect
21 dB SNR	ACW	9.91	7.59	7.24
	CEP	10.66	7.72	7.59
	PFL	10.33	7.52	7.38
	MFCC	9.52	6.86	6.57
	Max	8.40	5.70	5.87
	Sum	7.55	4.56	4.42
	Product	7.64	4.62	4.50
22 dB SNR	ACW	9.54	7.25	7.00
	CEP	9.71	7.22	6.87
	PFL	9.16	6.67	6.40
	MFCC	8.40	6.44	6.37
	Max	7.65	5.40	5.35
	Sum	6.56	4.41	4.12
	Product	6.61	4.38	4.15
23 dB SNR	ACW	8.40	6.49	6.42
	CEP	8.63	6.73	6.49
	PFL	8.14	6.19	6.09
	MFCC	8.18	6.21	6.09
	Max	6.72	5.22	5.10
	Sum	5.92	3.96	3.97
	Product	5.94	3.96	3.88
24 dB SNR	ACW	7.69	5.80	5.59
	CEP	7.69	5.80	5.56
	PFL	7.49	5.88	5.70
	MFCC	7.46	6.05	6.00
	Max	6.23	4.79	4.76
	Sum	5.16	3.67	3.67
	Product	5.22	3.65	3.63
25 dB SNR	ACW	6.79	5.55	5.23
	CEP	7.34	6.10	5.57
	PFL	6.66	5.13	4.84
	MFCC	6.52	5.71	5.54
	Max	5.95	4.66	4.48
	Sum	4.51	3.56	3.33
	Product	4.48	3.46	3.26
26 dB SNR	ACW	6.26	5.30	4.95
	CEP	6.74	5.78	5.37
	PFL	6.25	5.41	5.03
	MFCC	6.20	5.17	4.94
	Max	5.20	4.21	4.02
	Sum	4.26	3.23	3.05
	Product	4.26	3.22	3.00
27 dB SNR	ACW	5.84	5.14	4.82
	CEP	6.07	5.49	5.04
	PFL	5.86	5.15	4.63
	MFCC	5.67	4.88	4.65
	Max	4.67	4.19	3.97
	Sum	3.81	3.29	3.02
	Product	3.83	3.20	2.98

Table 11 (continued)

Test Condition	Feature/Fusion	Nothing	Blind	Perfect
28 dB SNR	ACW	5.47	4.99	4.61
	CEP	5.72	5.21	4.67
	PFL	5.39	4.61	4.23
	MFCC	5.20	4.74	4.24
	Max	4.40	3.84	3.65
	Sum	3.49	3.06	2.79
	Product	3.44	3.01	2.77
29 dB SNR	ACW	5.08	4.53	4.32
	CEP	5.46	5.08	4.64
	PFL	4.95	4.47	4.04
	MFCC	4.44	4.17	4.10
	Max	4.13	3.53	3.25
	Sum	3.27	2.81	2.67
	Product	3.24	2.85	2.63
30 dB SNR	ACW	4.41	4.11	3.77
	CEP	5.01	4.63	4.31
	PFL	4.53	4.22	3.90
	MFCC	4.29	4.17	3.92
	Max	3.86	3.55	3.21
	Sum	3.09	2.75	2.58
	Product	3.07	2.74	2.54
Clean Speech	ACW	2.59	2.65	2.59
	CEP	3.06	3.08	3.06
	PFL	3.02	3.07	3.02
	MFCC	2.77	2.76	2.77
	Max	2.20	2.27	2.20
	Sum	1.77	1.78	1.77
	Product	1.76	1.76	1.76

Note: The columns labeled Nothing, Blind, and Perfect denote the SNR estimation mode used for each trial. Nothing means that no enhancement was used.

Table 11 contains the average equal error rates taken over 10 rotations for each of the feature, fusions, SNR estimation modes, and test conditions used in the GSV-PLS trials.

4.3 Classifier Comparison

The optimal configurations for each classifier obtained via the previous experiments were used to compare the GMM-UBM and GSV-PLS systems. Ten rotations were performed in order to gather data to perform a one-way analysis of variance, followed by multiple comparison to identify the statistically best method. Only the sum fusion scores for the GMM-UBM and GSV-PLS systems were considered. The three fusion strategies were used to perform classifier fusion. The multiple comparison plots for each test condition are provided below. A summary of the results is included at the end of the subchapter.

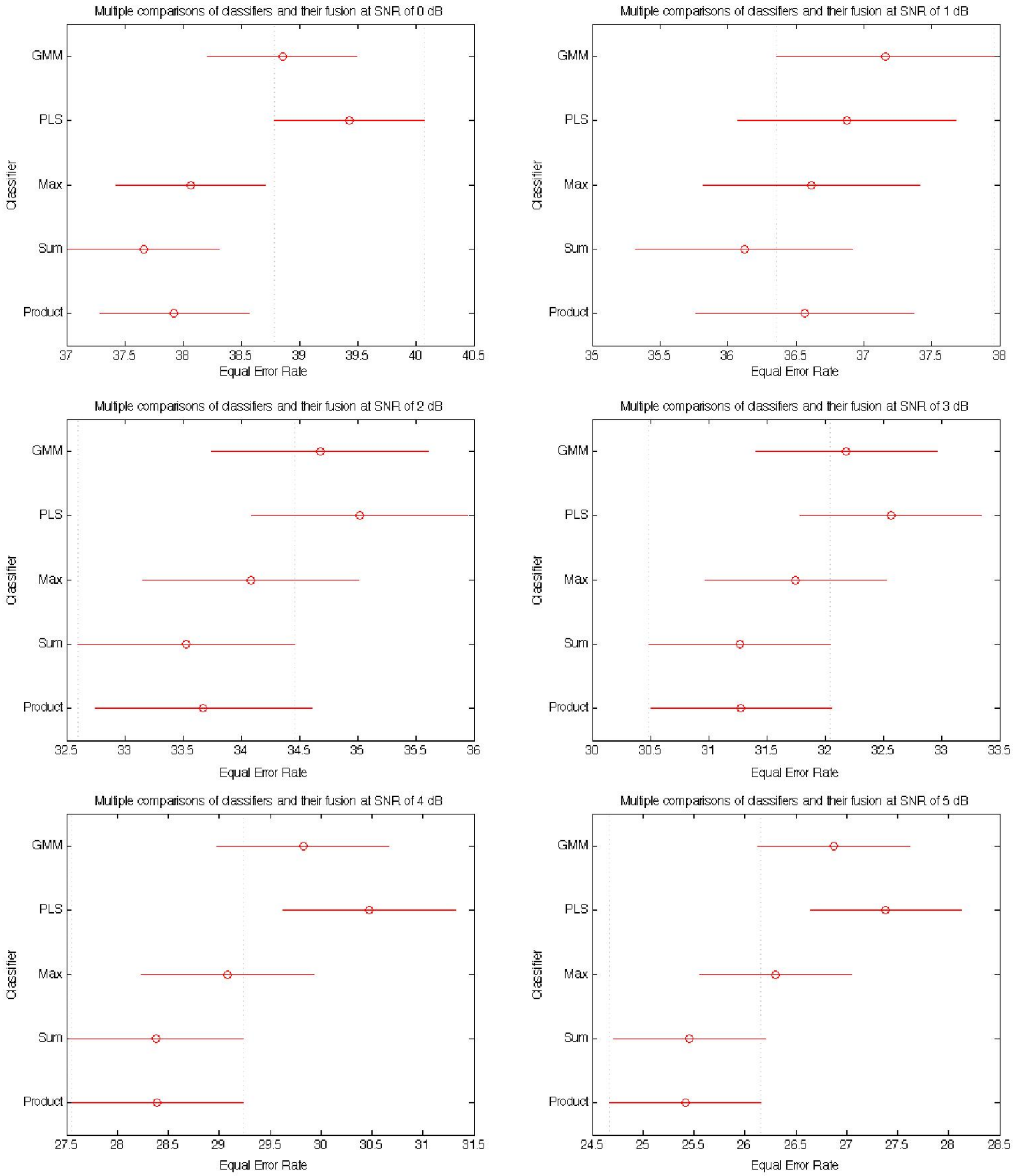

Figure 36. Comparison of classifiers and fusions for 0 dB to 5 dB.

Figure 36 shows the 95% confidence intervals for the trials comparing the GMM-UBM system, the GSV-PLS system, and the three methods used to fuse them for SNRs ranging from 0 dB to 5 dB. At an SNR of 0 dB, there is no statistically significant difference between the means of the EERs obtained from the GMM-UBM classifier and the three fusions. There is also no statistically significant difference between the means of the EERs obtained from the GMM-UBM and GSV-PLS classifiers. However, there is a statistically significant difference between the means of the EERs obtained using the GSV-PLS classifier and the three fusions; therefore, the three fusions were taken to be the best set for an SNR of 0 dB.

For SNRs of 1, 2, and 3 dB, there is no statistically significant difference between the means of the EERs obtained from any system. For SNRs of 4 dB and 5 dB, there is no statistically significant difference between the means of the EERs obtained using the GMM-UBM classifier and the three fusions. There are statistically significant differences in the means of the EERs obtained using the sum and product fusion methods and the means of the EERs obtained using the GSV-PLS classifier. For these two cases, the sum and product fusions were taken to be optimal.

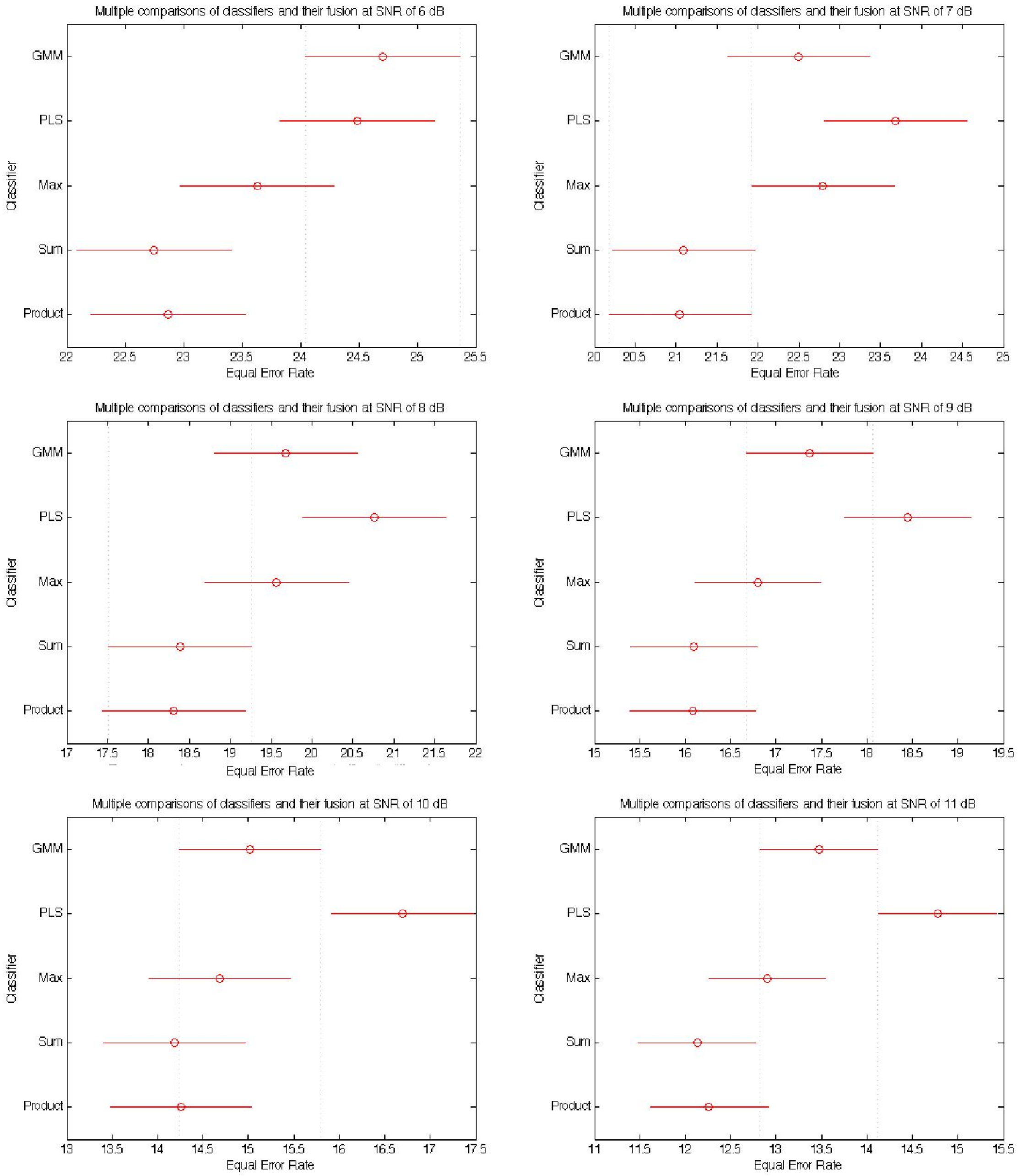

Figure 37. Comparison of classifiers and fusions for 6 dB to 11 dB.

Figure 37 shows the 95% confidence intervals for the trials comparing the GMM-UBM system, the GSV-PLS system, and the three methods used to fuse them for SNRs ranging from 6 dB to 11 dB. For all SNRs between 6 dB and 11 dB, there is no statistically significant difference between the means of the EERs obtained using sum and product fusion. At 6 dB, there is no statistically significant difference between the means of the EERs obtained using any of the fusion methods. There is a statistically significant difference between the means of the EERs obtained using sum and product fusion and the EERs obtained using the GMM-UBM and GSV-PLS classifiers alone. Therefore, for an SNR 6 dB, the best set was taken as the sum and product fusion methods.

At an SNR of 7 dB, there was no statistically significant difference between the means of the EERs obtained using sum and product and the EERs obtained using the GMM-UBM system alone; however, for only the product fusion, there were statistically significant differences in the means of the EERs obtained with that fusion method and the EERs obtained using the GSV-PLS system alone and obtained using max fusion. Therefore, for an SNR of 7 dB, product fusion of the classifiers was taken to be the optimal system.

At an SNR of 8 dB, there was no statistically significant difference in the means of the EERs obtained using any of the fusion strategies and using the GMM-UBM system alone. There were statistically significant differences between the means of the EERs obtained using sum and product fusion and the EERs obtained using the GSV-PLS system. Therefore, for an SNR of 8 dB, sum fusion and product fusion were chosen as the best set of systems.

At an SNR of 9 dB, there was no statistically significant difference in the means of the EERs obtained using any of the fusion strategies and using the GMM-UBM system alone. There were statistically significant differences between the means of the EERs obtained using all of the fusions and the EERs obtained using the GSV-PLS system. Therefore, for an SNR of 9 dB, max, sum, and product fusion were chosen as the best set of systems.

At an SNR of 10 dB, there was no statistically significant difference in the means of the EERs obtained using any of the fusion strategies and using the GMM-UBM system alone. There were statistically significant differences between the means of the EERs obtained using the GSV-PLS system and all of the other configurations. Therefore, for an SNR of 10 dB, the GMM-UBM system, max fusion, sum fusion, and product fusion were chosen as the best set of systems.

At an SNR of 11 dB, there was no statistically significant difference between the means of the EERs obtained using sum and product and the EERs obtained using maximum fusion; however, for only the sum fusion, there were statistically significant differences in the means of the EERs obtained with that fusion method and the EERs obtained using both of the classifiers alone. Therefore, for an SNR of 11 dB, sum fusion of the classifiers was taken to be the optimal system.

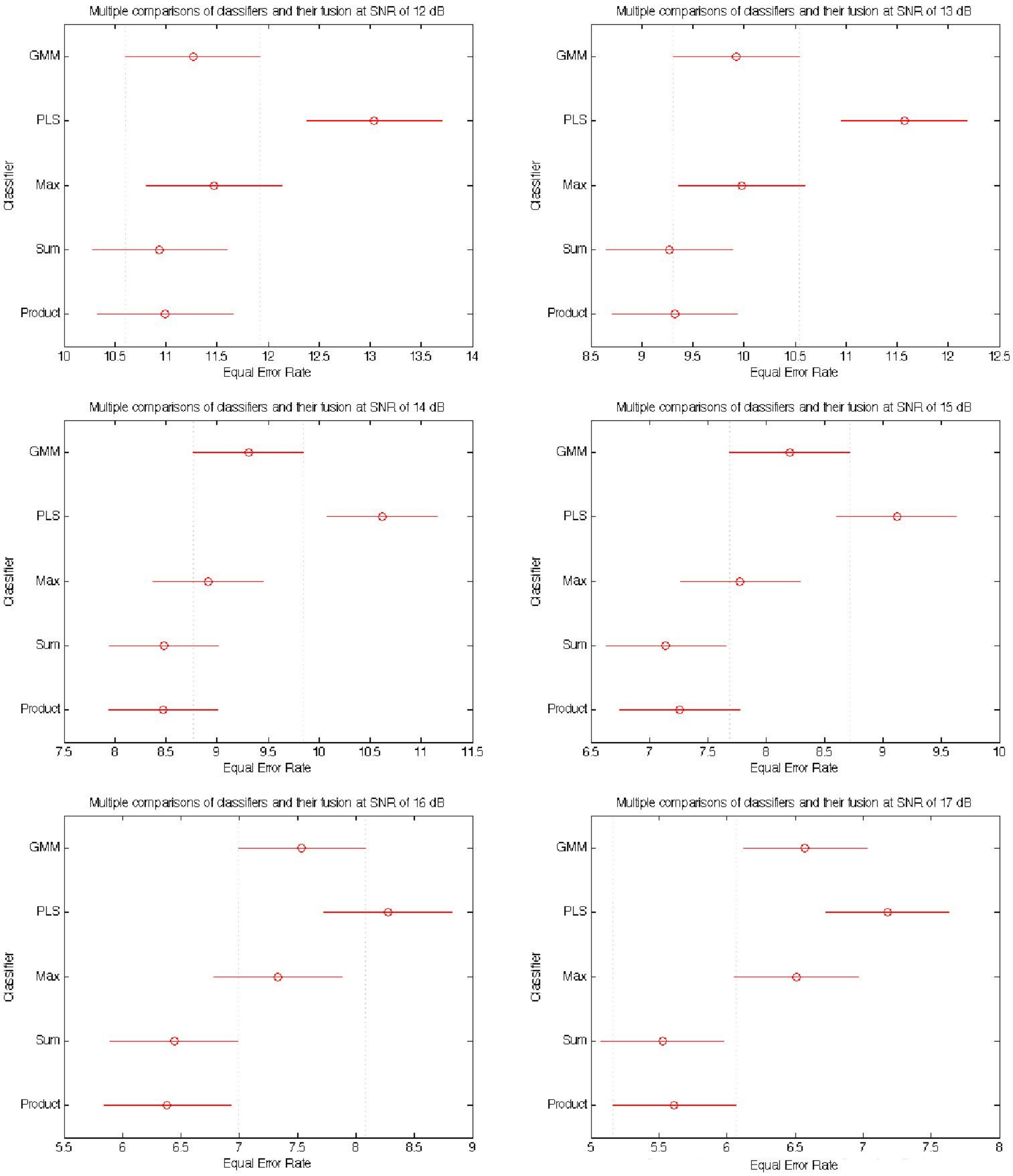

Figure 38. Comparison of classifiers and fusions for 12 dB to 17 dB.

Figure 38 shows the 95% confidence intervals for the trials comparing the GMM-UBM system, the GSV-PLS system, and the three methods used to fuse them for SNRs ranging from 12 dB to 17 dB. For SNRs of 12, 13, and 14 dB, there is no statistically significant difference in the means of the EERs obtained using the three fusion strategies and the GMM-UBM system alone; however, all four are shown to be statistically better than the GSV-PLS system. Therefore, for these three SNRs, the GMM-UBM system and the three fusion methods are taken to be the best set.

For an SNR of 15 dB, there is no statistically significant difference in the means of the EERs obtained using any of the three fusion methods. For only the sum fusion method, there are statistically significant differences in the means of the EERs obtained using that fusion method and the EERs obtained using the both classifiers without fusion. Therefore, for an SNR of 15 dB, sum fusion is considered optimal.

For an SNR of 16 dB, there is no statistically significant difference in the means of the EERs obtained using any of the three fusion methods. For sum and product fusion, there are statistically significant differences in the means of the EERs obtained using those two fusions and the EERs obtained using the two classifiers without fusion. Therefore, for an SNR of 16 dB, sum fusion and product fusion are both considered optimal.

For an SNR of 17 dB, there is no statistically significant difference in the means of the EERs obtained using sum fusion and product fusion. However, there are statistically significant differences between the means of the EERs obtained using sum fusion and the EERs for all other systems than product fusion; therefore, sum fusion is taken to be the best for this SNR.

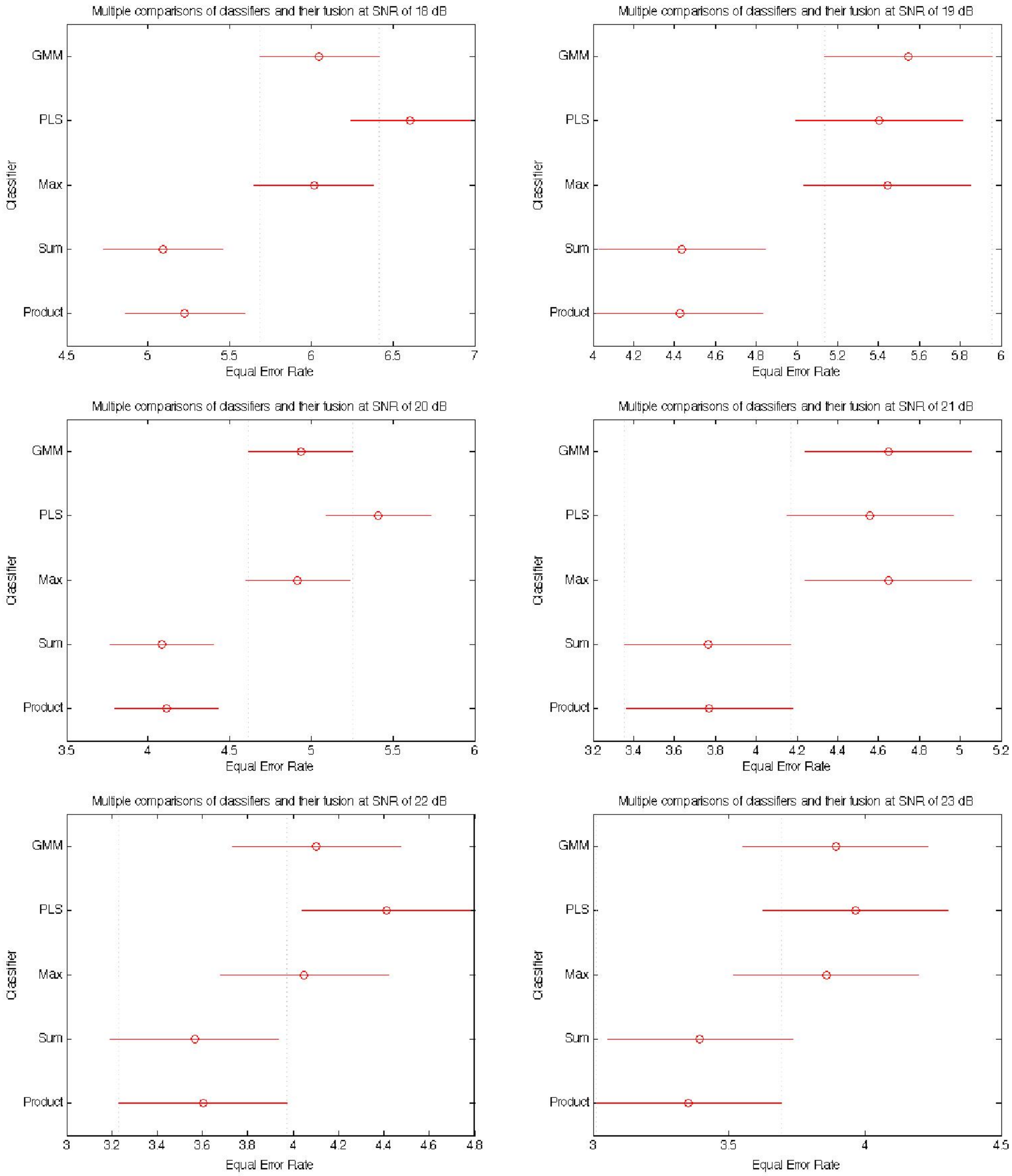

Figure 39. Comparison of classifiers and fusions for 18 dB to 23 dB.

Figure 39 shows the 95% confidence intervals for the trials comparing the GMM-UBM system, the GSV-PLS system, and the three methods used to fuse them for SNRs ranging from 18 dB to 23 dB. For SNRs of 18, 19, and 20 dB, there is no statistically significant difference between the means of the EERs obtained using sum and product fusion. Both of these fusions are shown to be statistically better than all other configurations, so they are considered optimal for these SNRs.

For an SNR of 21 dB, there is no statistically significant difference between the means of the EERs obtained using sum fusion, product fusion, and the GSV-PLS classifier with no fusion. There are statistically significant differences in the means of the EERs obtained using sum and product fusion and the EERs obtained using max fusion and the GMM-UBM system without fusion. Therefore, for an SNR of 21 dB, sum and product fusion are again taken to be optimal.

For an SNR of 22 dB, there is no statistically significant difference between the means of the EERs obtained using all of the fusions and the GMM-UBM classifier with no fusion; however, there are statistically significant difference between the means of the EERs obtained using sum and product fusion and the EERs obtained using the GSV-PLS classifier alone. Therefore, for an SNR of 22 dB, sum and product fusion are taken as optimal once again.

For an SNR of 23 dB, there is no statistically significant difference between the means of the EERs obtained using any of the systems. All systems are included in the best set of this SNR.

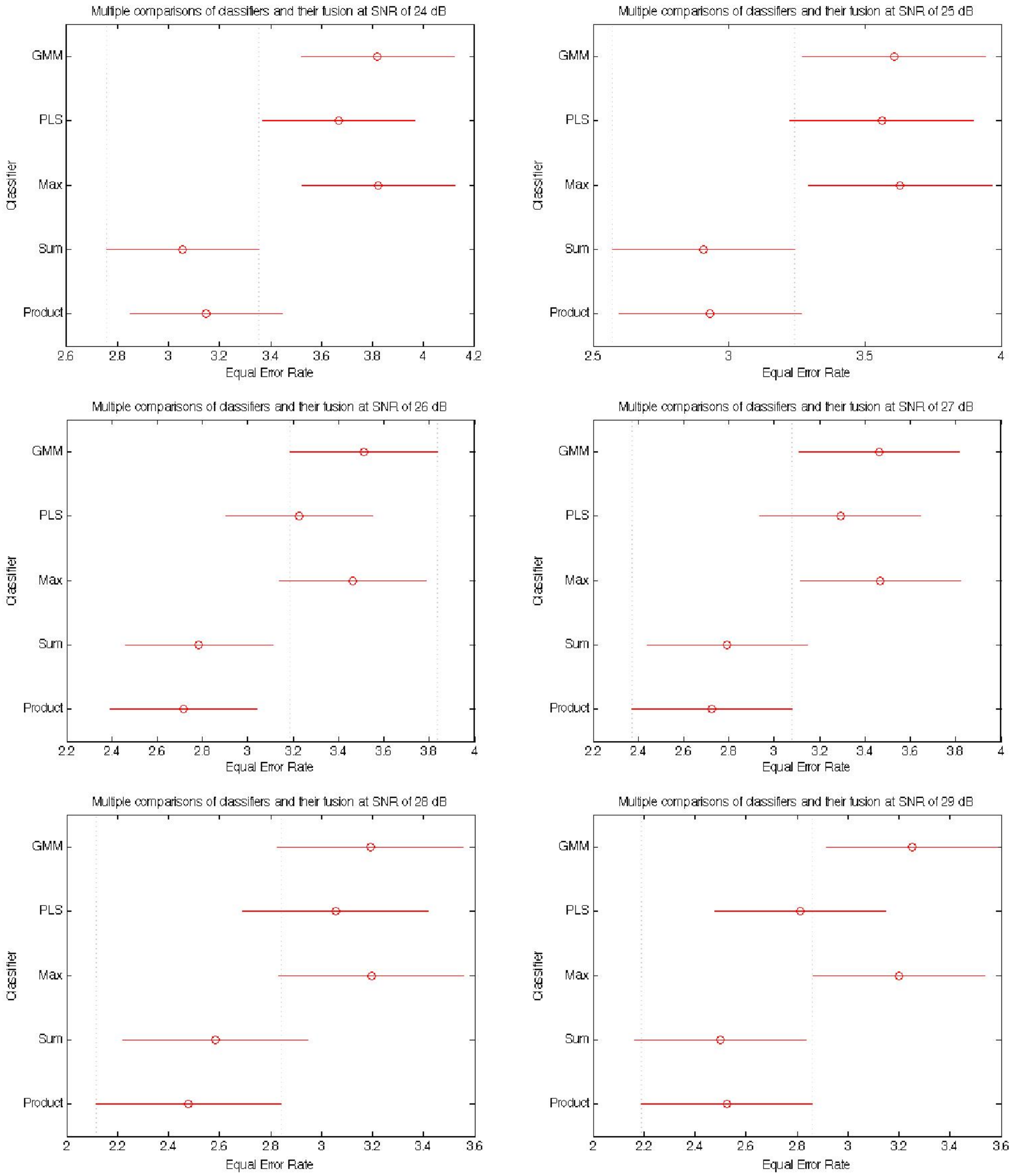

Figure 40. Comparison of classifiers and fusions for 24 dB to 29 dB.

Figure 40 shows the 95% confidence intervals for the trials comparing the GMM-UBM system, the GSV-PLS system, and the three methods used to fuse them for SNRs ranging from 24 dB to 29 dB. For an SNR of 24 dB, there is no statistically significant difference in the means of the EERs obtained using sum fusion and product fusion. However, there are statistically significant differences between the means of the EERs obtained using sum fusion and the EERs for all systems other than product fusion; therefore, sum fusion is taken to be the best for this SNR.

For SNRs of 25, 26, and 29 dB, there is no statistically significant difference between the means of the EERs obtained using sum fusion, product fusion, and the GSV-PLS classifier with no fusion. There are statistically significant differences in the means of the EERs obtained using sum and product fusion and the EERs obtained using max fusion and the GMM-UBM system without fusion. Therefore, sum and product fusion are again taken to be optimal for these SNRs.

For an SNR of 27 dB, there is no statistically significant difference between the means of the EERs obtained using product fusion, sum fusion, and the GSV-PLS classifier without fusion. However, for only the product fusion, there are statistically significant differences in the means of the EERs obtained using that fusion method and the EERs obtained using max fusion and the GMM-UBM system without fusion. Therefore, for an SNR of 27 dB, product fusion is considered optimal.

For an SNR of 28 dB, there are no statistically significant differences between the means of the EERs obtained using any of the methods. All systems are taken to be a part of the best set for this SNR.

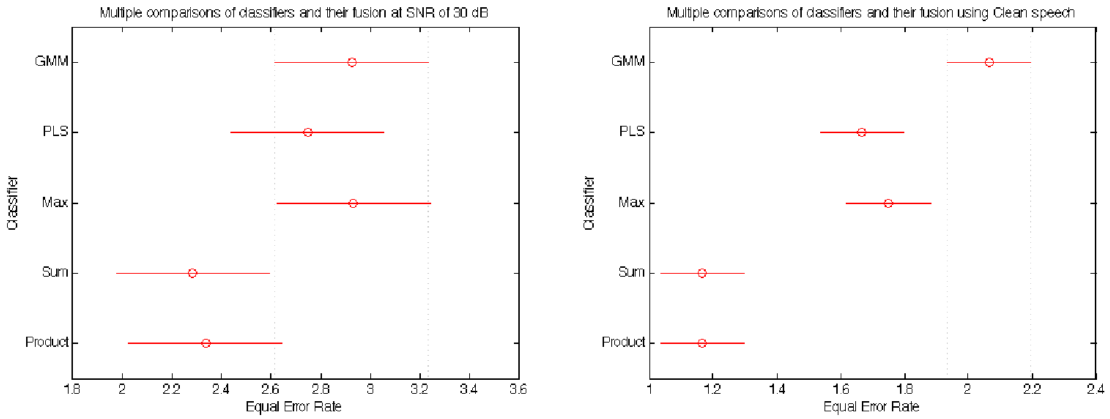

Figure 41. Comparison of classifiers and fusion for 30 dB and clean speech.

Figure 41 shows the 95% confidence intervals for the trials comparing the GMM-UBM system, the GSV-PLS system, and the three methods used to fuse them for an SNR of 30 dB and for clean test speech. For an SNR of 29 dB, there is no statistically significant difference between the means of the EERs obtained using product fusion, sum fusion, and the GSV-PLS classifier without fusion. However, for only the sum fusion, there are statistically significant differences in the means of the EERs obtained using that fusion method and the EERs obtained using max fusion and the GMM-UBM system without fusion. Therefore, for an SNR of 29 dB, sum fusion is considered optimal.

When tested on clean speech, there is no statistically significant difference between the means of the EERs obtained using sum fusion and product fusion. Both of these fusion methods are shown to be statistically superior to all of the other configurations; therefore, for clean speech, sum fusion and product fusion of the classifiers are considered optimal.

Table 12

Summary of best performing classifiers and fusions.

Classifier/Fusion	Appeared in Best Set	Total Count
GMM-UBM	1 – 3 dB, 10 dB, 12 – 14 dB, 23 dB, 28 dB	9
GSV-PLS	1 – 3 dB, 23 dB, 28 dB	5
Max Fusion	0 – 3 dB, 9 dB, 10 dB, 12 – 14 dB, 23 dB, 28 dB	11
Sum Fusion	0 – 6 dB, 8 – 26 dB, 28 – 30 dB, Clean	30
Product Fusion	0 – 10 dB, 12 – 14 dB, 16 dB, 18 – 22 dB, 25 – 29 dB, Clean	27

Note: The column labeled Appeared in Best Set indicates the SNR trials for which each classifier or fusion was selected as one of the optimal systems.

Table 12 provides a summary of the results obtained while comparing the GMM-UBM and GSV-PLS systems with their optimal configurations against each other and against their fusions. The GSV-PLS system without fusion was only included amongst the best set for a given SNR when there was no statistically significant difference between the means of the EERs in any of the systems. Additionally, it only outperformed the GMM-UBM system when tested using clean speech. Despite this evidently poor showing, it is still worthwhile to implement such a system due to how well the fusion of the GMM-UBM and GSV-PLS systems performs.

The sum and product fusion methods were counted among the best sets of classifiers in nearly every trial, and totally outperformed the systems without fusion several times. In a practical implementation, the best system to design in order to deal with additive noise at the speaker level would be the sum fusion of classifiers or the product fusion of classifiers, with sum fusion holding a slight edge.

A full tabulation of the average equal error rates obtained in these trials is provided below.

Table 13

Average equal error rates for trials comparing classifiers and their fusions.

Test Condition	GMM-UBM	GSV-PLS	Max	Sum	Product
0 dB SNR	38.86	39.43	38.07	37.67	37.93
1 dB SNR	37.16	36.88	36.62	36.12	36.57
2 dB SNR	34.68	35.02	34.09	33.53	33.68
3 dB SNR	32.18	32.57	31.75	31.27	31.28
4 dB SNR	29.82	30.47	29.08	28.38	28.39
5 dB SNR	26.88	27.38	26.30	25.46	25.41
6 dB SNR	24.70	24.49	23.63	22.75	22.87
7 dB SNR	22.50	23.68	22.80	21.09	21.05
8 dB SNR	19.68	20.77	19.57	18.39	18.30
9 dB SNR	17.37	18.45	16.80	16.10	16.09
10 dB SNR	15.02	16.69	14.69	14.18	14.26
11 dB SNR	13.47	14.78	12.90	12.13	12.26
12 dB SNR	11.26	13.04	11.47	10.94	10.99
13 dB SNR	9.93	11.57	9.98	9.27	9.32
14 dB SNR	9.31	10.62	8.91	8.48	8.47
15 dB SNR	8.20	9.12	7.78	7.14	7.26
16 dB SNR	7.54	8.28	7.33	6.44	6.39
17 dB SNR	6.57	7.18	6.51	5.53	5.61
18 dB SNR	6.05	6.61	6.02	5.09	5.22
19 dB SNR	5.55	5.40	5.44	4.44	4.43
20 dB SNR	4.93	5.41	4.92	4.08	4.11
21 dB SNR	4.65	4.56	4.65	3.76	3.77
22 dB SNR	4.10	4.41	4.05	3.56	3.60
23 dB SNR	3.89	3.96	3.86	3.39	3.35
24 dB SNR	3.82	3.67	3.82	3.06	3.15
25 dB SNR	3.61	3.56	3.63	2.91	2.93
26 dB SNR	3.51	3.23	3.46	2.78	2.72
27 dB SNR	3.46	3.29	3.47	2.79	2.72
28 dB SNR	3.19	3.06	3.20	2.58	2.48
29 dB SNR	3.25	2.81	3.20	2.50	2.53
30 dB SNR	2.93	2.75	2.93	2.28	2.34
Clean Speech	2.07	1.67	1.75	1.17	1.17

Chapter 5

Conclusions

Chapter 5 contains a recapitulation of the chapters contained within this thesis. The research accomplishments as they relate to the objectives set in Chapter 1 are reviewed. Finally, to conclude the thesis, research recommendations and ideas for future work are detailed.

5.1 Review of Thesis

Chapter 1 contains a statement of the problem that is being worked on by this thesis as well as the goals the thesis set out to accomplish. Chapter 2 gives an overview of the background information needed to understand each step of the procedure taken in the thesis. Chapter 3 contains a detailed methodology that was used to accomplish each of the research goals set in Chapter 1. Chapter 4 provided a comprehensive results obtained from each experiment performed in the proposed approach of this thesis. The results from each experiment enabled the way forward by providing enough information to identify optimal parameters for implementing each of the speaker verification systems.

5.2 Summary of Research Accomplishments

The goal of this thesis was to implement and validate methods for enhancing the performance of speaker verification systems in the presence of additive noise. The results demonstrated that using a repertoire of affine transforms in combination with a robust signal-to-noise ratio estimator significantly increased the performance of both the GMM-UBM and the GSV-PLS classifier. Further, performing score-level fusion and classifier fusion facilitated more robust classification at a range of test conditions. The objectives set in the first chapter of this thesis are now reviewed:

1. *To implement both a GMM-UBM and GSV-PLS system for speaker verification*.

- A Gaussian mixture model universal background model speaker verification system was implemented for four different formulated features. The mean vectors of the MAP adapted speaker models were used to create Gaussian supervectors. The supervectors were then used in a one vs. all partial least squares regression framework developed for speaker verification. These two classifiers were compared and analyzed to identify statistically significant performance.

2. *To enhance the performance of the speaker verification systems using SNR estimation, affine transforms, and score-level fusion of feature vectors*.

- Signal-to-noise ratio estimation was performed using VQ codebooks trained on the noise levels of training data. This SNR estimation was used to select from a repertoire of affine transforms for enhancement of the feature vectors. Score-level fusion of all of the features was performed using three different strategies. Each of these enhancements was shown to have resulted in statistically significant performance gains in numerous test conditions.

3. *To investigate the effect of the "affine resolution" parameter in supplementing the robustness of the speaker verification systems*.

- Multiple affine resolutions were examined in order to identify their effect on the performance of each classifier. For the GMM-UBM system, it was shown that an affine resolution of 1 is optimal. For the GSV-PLS system, it was shown that an affine resolution of 5 is optimal.

4. *To identify the best performing feature or fusion of features in the presence of various SNR noise-levels*.

- Statistical analysis showed that for both the GMM-UBM and the GSV-PLS classifier, sum fusion resulted in the most pronounced performance gains in the most test conditions.

5. *To perform a full classifier fusion of GMM-UBM and GSV-PLS in their best configurations*.

- The optimal configurations of each of the classifiers was determined through exhaustive experimentation with all of the identified key parameters. Fusion of the classifiers was performed using three score-level fusion techniques.

6. *To analyze the performances of each classifier and their fusion to determine whether there are statistically significant differences*.

- The performances of the GMM-UBM and GSV-PLS systems were compared against each other and against the performances of their fusions. It was shown by statistical analysis that using sum fusion of the classifiers results in the best performance under the most test conditions.

5.3 Recommendations for Future Work

All of the experiments involving different signal-to-noise ratios were performed using only additive white Gaussian noise. Testing the performance of these systems in the presence of a more diverse set of noise types may be beneficial. Additionally, the number of PLS components for the regression models were chosen based on the amount of variance explained by the model. Further experimentation with the effect of the number of PLS components on overall system performance may be warranted.

The uses of PLS regression for different aspects of speaker verification have not been exhausted. PLS regression is a powerful tool that might be successfully used in place of many different methods in the proposed approach. For example, rather than using VQ codebooks to perform SNR estimation, PLS regression might be useful in performing this task. Because of the entirely different approaches PLS and VQ would use to estimate SNRs, fusion could be attempted in order to create a more accurate SNR estimator. Furthermore, the use of PLS regression in place of affine transforms for feature enhancement as a bulwark against corrupted speech signals may be a fruitful area of research.

References

[1] H. Beigi, Fundamentals of Speaker Recognition, New York: Springer, 2011.

[2] T. Kinnunen and L. Haizhou, "An overview of text-independent speaker recognition: From features to supervectors," *Speech Communication,* vol. 52, pp. 12-40, 2010.

[3] B. V. Srinivasan, Y. Luo, D. Garcia-Romera, D. N. Zotkin and R. Duraiswami, "A Symmetric Kernel Partial Least Squares Framework for Speaker Recognition," *IEEE Transactions on Audio, Speech, and Language Processing,* vol. 21, no. 7, pp. 1451-1523, 2013.

[4] B. V. Srinivasan, D. N. Zotkin and R. Duraiswami, "A Partial Least Squares Framework for Speaker Recognition," in *IEEE International Conference on Acoustics, Speech and Speech Processing*, Prague, 2011.

[5] L. Deng and D. O'Shaughnessy, Speech Processing: A Dynamic and Optimization-Oriented Approach, Boca Raton: CRC Press, 2003.

[6] R. J. Mammone and R. P. Ramachandran, "Robust Speaker Recognition: A Feature-based Approach," *IEEE Signal Processing Magazine,* vol. 96, pp. 58-70, 1996.

[7] M. S. Zilovic, R. P. Ramachandran and R. J. Mammone, "Speaker Identification Based on the Use of Robust Cepstral Features Obtained from Pole-Zero Transfer Functions," *IEEE Transactions on Speech and Audio Processing,* vol. 6, no. 3, pp. 260-267, 1998.

[8] M. S. Zilovic, R. P. Ramachandran and R. J. Mammone, "A Fast Algorithm for Finding the Adaptive Component Weighted Cepstrum for Speaker Recognition," *IEEE Transactions on Speech and Audio Processing,* vol. 5, no. 1, pp. 84-86, 1997.

[9] R. Togneri and D. Pullella, "An Overview of Speaker Identification: Accuracy and Robustness Issues," *IEEE Circuits and Systems Magazine,* vol. 11, pp. 23-61, 2011.

[10] D. A. Reynolds, T. F. Quatieri and R. B. Dunn, "Speaker Verification Using Adapated Gaussian Mixture Models," *Digital Signal Processing,* vol. 10, pp. 19-41, 2000.

[11] C. M. Bishop, Pattern Recognition and Machine Learning, Singapore: Springer, 2006.

[12] T. Hasan and J. H. L. Hansen, "A Study on Universal Background Model Training in Speaker Verification," *IEEE Transactions on Audio, Speech, and Language Processing,* vol. 19, no. 7, pp. 1890-1899, 2011.

[13] W. M. Campbell, D. E. Sturim and D. A. Reynolds, "Support Vector Machines Using GMM Supervectors for Speaker Verification," *IEEE Signal Processing Letters,* vol. 13, no. 5, pp. 308-311, 2006.

[14] R. Rosipal and N. Krämer, "Overview and Recent Advances in Partial Least Squares," in *Subspace, Latent Structure and Feature Selection: Statistical and Optimization Perspectives Workshop*, Berlin, Springer, 2005, pp. 34-51.

[15] R. Ondusko, M. Marbach, A. McClellan, R. P. Ramachandran, L. M. Head, M. C. Huggins and B. Y. Smolenski, "Blind Determination of the Signal to Noise Ratio of Speech Signals Based on Estimation Combination of Multiple Features," in *IEEE Asia Pacific Conference on Circuits and Systems*, Singapore, 2006.

[16] Y. Linde, A. Buzo and R. M. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Transactions on Communications,* Vols. COM-28, no. 1, pp. 84-95, 1980.

[17] J. Benesty, M. M. Sondhi and Y. Huang, Handbook of Speech Processing, Berlin: Springer, 2008.

[18] K. Raval, R. P. Ramachandran, S. S. Shetty and B. Y. Smolenski, "Feature and Signal Enhancement for Robust Speaker Identification of G.729 Decoded Speech," in *International Conference on Neural Information Processing*, Doha, Qatar, 2012.

[19] R. W. Mudrowsky, R. P. Ramachandran, U. Thayasavim and S. S. Shetty, "Robust Speaker Recognition in the Presence of Speech Coding Distortion for Remote Access Applications," in *International Conference on Data Mining*, Athens, 2016.

[20] F. Răstoceanu and M. Lazăr, "Score fusion methods for text-independent speaker verification applications," in *6th International Conference on Speech Technology and Human-Computer Dialogue*, Braþov, Romania, 2011.

[21] J. L. Devore, Probability and Statistics for Engineering and the Sciences, Boston: Brooks/Cole, 2010.

[22] R. Auckenthaler, M. Carey and H. Lloyd-Thomas, "Score Normalization for Text-Independent Speaker Verification Systems," *Digital Signal Processing,* vol. 10, pp. 42-54, 2000.