Sonification

and augmented data sets in binary classification

Preface

This master thesis serves as documentation for the final assignment in the requirements to achieve the degree Master of Science in Engineering. The work has been carried out in the period from the 19th of January 2005 to the 19th of August 2005 at the Institute of Informatics and Mathematical Modeling at the Technical University of Denmark. The work has been supervised by Lars Kai Hansen and with the help of the Ph.D. students at IMM.

Kgs. Lyngby, August 19, 2005

Fares El-Azm s970581

Abstract

In the thesis an auditory browser based on granular synthesis is designed and implemented to aid in browsing through long EEG time courses. This application can be used when ICA is applied to EEG signals as a means of decontamination and is intended to accelerate the identification of artifactual time courses, though this was not confirmed through testing. Furthermore, an introduction to the rather young field of sonification and EEG sonification is presented, also including introductory chapters on auditory perception and sound synthesis. Concepts in classification are introduced and the idea of augmented data sets using PCA and ICA is investigated. It is shown that augmenting data sets can "supervise" PCA and ICA, though this was seen to be especially true for PCA.

Keywords: Sonification, classification, EEG, granular synthesis, auditory perception, sound synthesis, augmented data sets, ICA, PCA.

Resumé

I denne afhandling blev en lyd browser baseret på granular syntese konstrueret og implementeret til at assistere i browsing gennem lange EEG aktiveringer. Denne applikation kan bruges når ICA er benyttet til rensning af EEG signaler og er beregnet til at accelerere identifikationen af artefaktiske aktiveringer, dog var dette ikke bekræftet gennem afprøvning. Derudover blev sonifikation og EEG sonifikation introduceret, samt indledende kapitler om lyd opfattelse og lyd syntese. Begreber indenfor klassifikation introduceres, samt begreber indenfor supplerede dataset ved brug af PCA og ICA undersøges. Der vises, at de supplerede dataset kan "assistere" PCA og ICA, dog gælder dette især for PCA.

Nøgleord: Sonifikation, klassifikation, EEG, granular syntese, lyd opfattelse, lyd syntese, supplerede dataset, ICA, PCA.

Acknowledgements

I wish to thank Nanna for being patient with me and aiding me through the process of this thesis. I would also like to thank my mother and my sister, Iben. Last but not least, I wish to thank Lars and all the people at IMM who made the 7 months I spent writing my thesis a pleasant and great learning experience.

Chapter 1

1 Introduction

This thesis focuses on sonification and data analysis, and it was originally inspired by a project description by the name of, "The sound of the experienced self: A study of subjectivity disturbances and brain activity", written by senior researcher Sidse Arnfred at Hvidovre Hospital. A subset of that project was to sonify event related EEG activity for novel ways of interpreting EEG data. Due to the fact that no previous investigations of sonification, or sonification of EEG data, had been conducted at IMM, an overview of the field of sonification was requested in this connection. Furthermore, EEG sonification is quite new and very few scientists are actively involved in this research area, and no well-documented or tested techniques were readily available for the requested task, which also required a study of the suggested techniques for analyzing EEG data through sound.

The field of sonification is rather young, though the use of sound to convey information is not, for example Morse code and Geiger counters. Since the establishment of the annual international conference on auditory display in 1992, sonification seems to have become a more accepted discipline in exploring and presenting data, though the field is still in its early steps. As will be made clearer, the designing of auditory displays is interdisciplinary in nature and touches on many aspects such as data mining, computer science, human factors, signal processing, acoustics, psychology, auditory perception and more. In this thesis the main focus will be on auditory perception and sound synthesis techniques, which the author feels are important aspects when designing and implementing sonifications. As mentioned, an overview of the field of sonification and an overview of using auditory displays for analyzing EEG data will be given. Furthermore, an application for browsing through time courses will be designed and implemented using presented techniques. A usability test of this application is out of the scope of this thesis, but is a crucial part when determining the possible advantage of using sonifications compared to the more traditional visual displays.

The thesis also focuses on data analysis, which also is an important part of exploring and analyzing data. An investigation of augmenting labeled data sets will be conducted. The main idea behind augmented data sets is that unsupervised techniques such as PCA and ICA could to some extent become supervised. Experiments will be conducted on augmenting data sets using PCA and initial experiments will be conducted on ICA (infomax). The investigation of augmenting data sets using unsupervised techniques was to reveal the possibilities of this technique, and since this is “new territory” this thesis only presents a preliminary heuristic assessment of this technique.

Chapter 2 in this thesis will give an introduction to pattern recognition and classification, including the investigations of augmented data sets. Chapter 3 deals with auditory perception and gives an introduction to the human ear and psychoacoustics. Subsequently, chapter 4 concerns the field of sonification and gives an introduction to the present techniques and the issues in designing sonification. Chapter 5 gives a brief introduction to the sound synthesis techniques where the weight is put on the granular synthesis technique. Chapter 6 will focus on EEG sonification and presents the auditory browser designed in the course of this project. Thereafter, a short discussion on sonification and the auditory browser is presented in chapter 7, and finally, the conclusion is made in chapter 8.

The project should be read as an introduction to sonification and EEG sonification, focusing on monitoring and browsing through classified states in time courses. Classification is introduced in this regard and an investigation is made on augmented data sets in the field of feature extraction and classification in binary classification problems.

Chapter 2

2 Pattern Recognition and Classification

In this chapter topics of pattern recognition, and especially classification, are presented. These include Bayes' theorem that lets the posterior probability be expressed in terms of attainable quantities. This theorem is then used to deduce optimal decision boundaries and discriminant functions, which are central in classification problems. The concept of generalization is presented, which ensures that the inferred parameters are the optimal in a general sense. Furthermore, performance measures in binary classification problems are introduced, where the confusion matrix and the receiver operating characteristic curve are discussed. Then, a brief review of the Principal Component Analysis and the Independent Component Analysis, together with Fisher's linear discriminant is presented. Finally, the concept of augmenting data sets is introduced followed by a heuristic investigation when this technique is used in concert with the principal component analysis and the independent component analysis. First of all, an introduction to classification is presented.

2.1 Introduction to Classification

Pattern recognition can be viewed as the process of assigning a label or input to an observation or output, as illustrated in Figure 1. In *classification* problems the task is to assign new inputs to one of a number of discrete classes or categories, the outputs. However, there is another pattern recognition task referred to as *regression*, where the outputs represent the values of continuous variables. Both regression and classification problems can be viewed as particular cases of *function approximation*. In regression problems it is the regression functions which we seek to approximate, while for classification the functions we seek to approximate are the probabilities of membership of the different classes [Bishop 1995 p. 6].

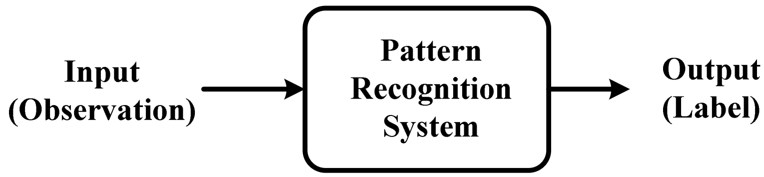


Figure 1. A schematic illustration of pattern recognition.

The outcome of the classification can be represented in terms of a variable y, which takes the value 1 when a data point is classified as C_1, and the value 0 if it is classified as C_2. Thus, the overall system can be viewed as a *mapping* from a set of input variables $x_1,\dots,x_d$, to an output variable y representing the class label. In more complex problems there may be several output variables, which one can denote by y_k where $k = 1,\dots,c$.

In general, it will not be possible to determine a suitable form for the required mapping, except with the help of a data set of examples. The mapping is therefore modeled in terms of some mathematical function which contains a number of adjustable parameters, whose values are determined with the help of the data **x**. In [Bishop 1995 p. 5] the functions are written in the general form

$$y_k = y_k(\mathbf{x};\mathbf{w}) \qquad 2.1$$

where **w** is the parameter vector. In this case, the parameters in **w** are often called *weights*. The method of determining the values for the adjustable parameters on the basis of the data set is called *learning* or *training*, and therefore the data set of examples is referred to as a *training set*. Neural network models (non-linear), as well as many conventional approaches to statistical pattern recognition (mostly linear), can be viewed as specific choices for the functional forms used to represent the mapping in equation 2.1. The determining of these parameters in the classification process is called *inference*.

In statistical pattern recognition applications the original set of input variables $x_1,\dots, x_d$ are usually transformed with the help of an important *pre-processing* stage before being fed into the statistical model or classifier. The removal of irrelevant information and extraction of key features to simplify a pattern recognition problem is referred to as pre-processing [Kennedy 1997 p. 1.17]. Any object or pattern which can be recognized and classified possesses a number of discriminatory properties or *features*. Transforming the original input variables, given by x_i, to a single variable x_1' possessing the discriminatory attributes, is an example of pre-processing referred to as *feature extraction* [Bishop 1995 p. 6]. This is illustrated in Figure 2. The use of pre-processing can often greatly improve the performance of a pattern recognition system, the reason for this is:

1. Incorporating *prior knowledge*
2. Reducing the *dimensionality*

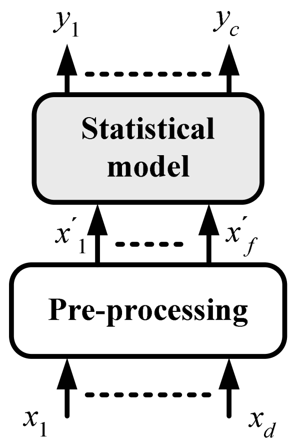


Figure 2. For a large part of pattern recognition applications the original input variables $x_1,\ldots, x_d$ are transformed by some form of pre-processing to give a new set of variables $x'_1,\ldots, x'_f$. These are then treated as the inputs to the statistical model, whose outputs are denoted by $y_1,\ldots, y_c$. This figure is redrawn from [Bishop 1995 p. 7]

Prior knowledge is information which we possess about the desired form of the solution and is additional to the information provided by the training data [Bishop 1995 p. 295]. Secondly, by increasing the dimensionality of a model one also needs to exponentially increase the size of the training data in order for the data to have the same density in feature space [Bishop 1995 p. 7]. This phenomenon has been termed the *curse of dimensionality* and in statistics it relates to the fact that the convergence of any estimator to the true value of a smooth function defined on a space of high dimension is very slow. In practice we are forced to work with limited quantity of data, then increasing the dimensionality of the space rapidly leads to the point where the data is very sparse, in which case it provides a very poor representation of the mapping. Some statistical methods for features extraction are Principal Component Analysis, Factor analysis, Fisher's linear discriminant and Independent Component Analysis, some of which will be discussed later in the text.

It is important to distinguish between two separate stages in the classification process as illustrated in Figure 3. The first is inference whereby (known/measured) data is used to determine the optimal parameter values. These are then used in the second stage, which is *decision-making*, in which the parameters are used to make decisions such as assigning a new data point to one of the possible classes [Bishop 1995 p. 20].

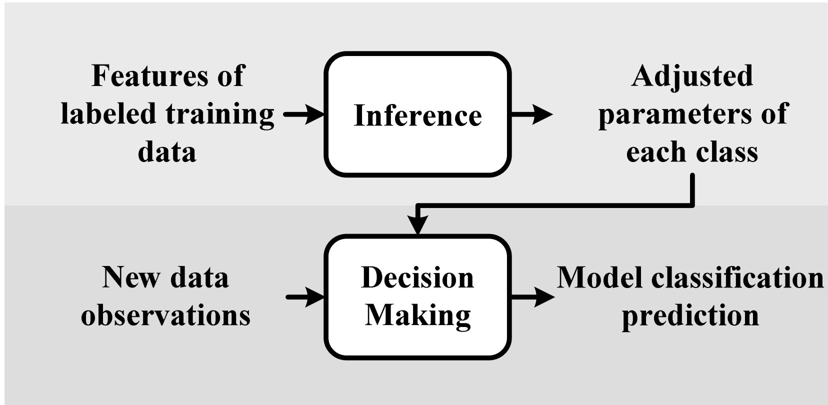


Figure 3. A schematic illustration of the two stages in the classification process. First inference is made about the labeled training resulting in the estimated parameters or weights, which then are used in the decision making stage to classify new data observations.

As mentioned, for classification problems the functions which we seek to approximate are the probabilities of membership of the different classes expressed as functions of the input variables. The goal in classification is, thus to classify the data in such a way as to minimize the probability of misclassification. The ability of a model to generalize to new data and not over-fit or imitate the training data, is an important goal for any function or model approximation. This can be ensured by *cross validation* which will be presented in section 2.5.

2.2 Bayes' Theorem

Bayes' theorem is the starting point for inference problems using probability as logic. *Bayesian* approaches maintain that rational belief is governed by the laws of probability, lean heavily on conditional probabilities in their theories of evidence and their models of empirical learning. For continuous variables Bayes' theorem can be expressed as [Bishop 1995 p. 23]

$$P(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k) \cdot P(C_k)}{p(\mathbf{x})} \qquad 2.2$$

where $P(C_k|\mathbf{x})$ is called the *posterior* probability, since it gives the probability that the class is C_k given a measurement of $\mathbf{x}$. Bayes' theorem lets the posterior probability be expressed in terms of the prior probability $P(C_k)$, together with the quantity $p(\mathbf{x}|C_k)$ which is called the *class-conditional* probability density function of $\mathbf{x}$ for class C_k. The denominator $p(\mathbf{x})$ plays the role of a *normalization factor*, and ensures that the posterior probabilities sum to unity. The density $p(\mathbf{x})$ for c distinct classes is given by

$$p(\mathbf{x}) = \sum_{k=1}^{c} p(\mathbf{x}|C_k) \cdot P(C_k) \qquad 2.3$$

thus ensuring that the posterior probabilities sum to unity. The class-conditional densities $p(\mathbf{x}|C_k)$ can be assumed Gaussian distributed, which are modeled by parameterized functional forms as seen in equation 2.20. When viewed as functions of the parameters they are referred to as *likelihood functions*, of class C_k for the observed value of $\mathbf{x}$. Bayes' theorem can therefore be summarized in the textual form

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{normalization factor}} \tag{2.4}$$

The joint probabilities density function of $\mathbf{x}$ and C_k occurring simultaneously is given by

$$p(\mathbf{x}, C_k) = p(\mathbf{x}|C_k) \cdot P(C_k) \tag{2.5}$$

2.3 Decision boundaries

As mentioned, the posterior probability $P(C_k|\mathbf{x})$ gives the probability of the pattern belonging to class C_k given a feature vector $\mathbf{x}$. The probability of misclassification is minimized by selecting the class C_k having the largest posterior probability, so that a feature vector $\mathbf{x}$ is assigned to class C_k if [Bishop 1995 p. 23]

$$P(C_k|\mathbf{x}) > P(C_j|\mathbf{x}), \qquad \text{for all } j \neq k \tag{2.6}$$

Since the density $p(\mathbf{x})$ is independent of the class, it can be left out from the Bayes' formula for the purposes of comparing posterior probabilities. Equation 2.2 can then be used to write the criterion in equation 2.6 to

$$p(\mathbf{x}|C_k) \cdot P(C_k) > p(\mathbf{x}|C_j) \cdot P(C_j), \qquad \text{for all } j \neq k \tag{2.7}$$

A pattern classifier provides a rule for assigning each point of feature space to one of c classes. The feature space can be regarded as being divided into *c decision regions* $\mathcal{R}_1, \ldots, \mathcal{R}_c$ such that a point falling in region $\mathcal{R}_k$ is assigned to class C_k. The boundaries between these regions are known as *decision surfaces* or *decision boundaries*.

In order to find the optimal criterion for placement of decision boundaries, consider the simple case of a one-dimensional feature space x and two classes C_1 and C_2. As illustrated in Figure 4 the decision boundary tries to minimize the probability of misclassification. Assigning a new pattern to class C_1 when in it belongs to class C_2, or vice versa leads to a misclassification error. The total probability of an error of either kind occurring can be calculated by the following [Bishop 1995 p. 24]

$$\begin{aligned} P(\text{error}) &= P(x \in \mathcal{R}_2, C_1) + P(x \in \mathcal{R}_1, C_2) \\ &= P(x \in \mathcal{R}_2|C_1)P(C_1) + P(x \in \mathcal{R}_1|C_2)P(C_2) \\ &= \int_{\mathcal{R}_2} p(x|C_1)P(C_1)dx + \int_{\mathcal{R}_1} p(x|C_2)P(C_2)dx \end{aligned} \tag{2.8}$$

where $P(x \in \mathbf{R}_1, C_2)$ is the joint probability of x being assigned to class C_1 and the true class being C_2. Thus, if $p(x|C_1)P(C_1) > p(x|C_2)P(C_2)$ for a given x, one should assign x to $\mathbf{R}_1$, since this gives a smaller contribution to the error. By choosing the decision boundary to coincide with the value x where the two distributions cross (shown by the arrow in Figure 4) one minimizes the area of the shaded region, illustrating the classification error, and therefore minimizes the probability of misclassification. This corresponds to classifying each new pattern x using equation 2.7, which is equivalent to assigning each pattern to the class having the largest posterior distribution. This can naturally be extended into the general case of c classes and d-dimensional feature vectors. Binary classification, however, is not an oversimplified method only used in textbook examples, it has its applications and is typically used in [Fawcett 2003]

- ***Medical testing***. To determine if a patient has certain disease or not;
- ***Psychophysical testing***. To determine thresholds e.g. of human listeners;
- ***Quality control in factories***. To decide whether a new product is good enough to be sold, or if it should be discarded

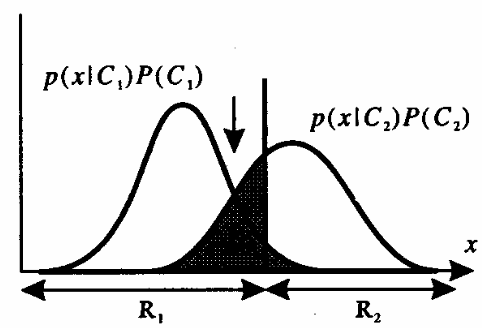


Figure 4 is a schematic illustration of $p(x|C_k)P(C_k)$, also known as the joint probability densities, as a function of a feature value x, for two class C_1 and C_2. If the vertical line is used as the decision boundary then the classification errors arise from the shaded region. By placing the decision boundary at the point where the two probability density curves cross (shown by the arrow), the probability of misclassification is minimized. This figure is taken from [Bishop 1995 p. 25].

2.4 Discriminant functions

The focus of the previous section was on probability distribution functions, where the decision on class membership in the classifier was solely based on the relative sizes of the probabilities. The classification problem can be reformulated in terms of a set of *discriminant functions* $y_1(\mathbf{x}),\ldots, y_c(\mathbf{x})$ such that an input vector $\mathbf{x}$ is assigned to class C_k if

$$y_k(\mathbf{x}) > y_j(\mathbf{x}), \qquad \text{for all } j \neq k \qquad 2.10$$

The decision rule for minimizing the probability of misclassification may easily be cast in terms of discriminant functions, simply by choosing

$$y_k(\mathbf{x}) = P(C_k|\mathbf{x}). \qquad 2.11$$

which leads to

$$y_k(\mathbf{x}) = p(\mathbf{x}|C_k)P(C_k). \qquad 2.12$$

Since it is only the relative magnitudes of the discriminant function which are important in determining the class, we can replace $y_k(\mathbf{x})$ by $g(y_k(\mathbf{x}))$, where $g()$ is any monotonic function, and the decisions of the classifier will not be affected. A monotonous function usually applied in this case is the natural logarithm function, which lead to the discriminant functions in the form

$$y_k(\mathbf{x}) = \ln p(\mathbf{x}|C_k) + \ln P(C_k) \qquad 2.13$$

In general, the decision boundaries are given by the regions where the discriminant functions are equal, so that if $\boldsymbol{R}_k$ and $\boldsymbol{R}_j$ are adjacent, then the decision boundary separating them is given by

$$y_k(\mathbf{x}) = y_j(\mathbf{x}) \qquad 2.14$$

The location of the decision boundaries are therefore unaffected by monotonic transformations of the discriminant functions.

Discriminant functions for two-class decision problems can be written in a more compact from. Instead of using two discriminant functions $y_1(\mathbf{x})$ and $y_2(\mathbf{x})$, a single discriminant function can be obtained

$$y(\mathbf{x}) = y_1(\mathbf{x}) - y_2(\mathbf{x}) \qquad 2.15$$

$$= \begin{cases} y(\mathbf{x}) > 0, & \mathbf{x} \in C_1 \\ y(\mathbf{x}) < 0, & \mathbf{x} \in C_2 \end{cases} \qquad 2.16$$

From equation 2.11 and equation 2.15 it follows that

$$y(\mathbf{x}) = P(C_1|\mathbf{x}) - P(C_2|\mathbf{x}) \qquad 2.17$$

or alternatively, from equation 2.13 and equation 2.15 it follows that

$$y(\mathbf{x}) = \ln \frac{p(\mathbf{x}|C_1)}{p(\mathbf{x}|C_2)} + \ln \frac{P(C_1)}{P(C_2)} \qquad 2.18$$

By relating the discriminant functions to the probabilities, one retains the link to the optimal criteria of decision theory introduced above.

As suggested, in a practical application of discriminant functions, specific parameterized functional forms are chosen, and the values of the parameters are then determined from a set of training data by means of inference. The simplest choice of a discriminant function consists of a linear combination of the input variables in which the coefficients in the linear combination are the parameters of the model, and has been considered widely in the literature on conventional approaches to pattern recognition, and has the well known functional form

$$y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x} + w_0 \qquad 2.19$$

2.4.1 Discrimination between Two Normal Distributions

The general multivariate normal probability density in d dimensions, as defined in [Bishop 1995 p. 35], can be written as

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{1/d}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu)^{\mathrm{T}}\Sigma^{-1}(\mathbf{x}-\mu)\right\} \qquad 2.20$$

where the mean μ is a d-dimensional vector and the covariance Σ is $d \times d$ matrix. Thus, the normal probability density is governed by the parameters μ and Σ and is usually specified as $N_d(\mu, \Sigma)$.

Considering a two-class problem in which the class-conditional probability densities are Gaussian distributions with equal covariance matrix, they can then be expressed as

$$p(\mathbf{x}|C_k) = \frac{1}{(2\pi)^{1/d}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu_k)^{\mathrm{T}}\Sigma^{-1}(\mathbf{x}-\mu_k)\right\} \qquad 2.21$$

A graphical illustration of two classes in two dimensions is given in Figure 7. Inserting equation 2.21 into equation 2.18 gives

$$y(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\Sigma^{-1}(\mu_1-\mu_2) - \frac{1}{2}\mu_1\Sigma^{-1}\mu_1 + \frac{1}{2}\mu_2\Sigma^{-1}\mu_2 + \ln\frac{P(C_1)}{P(C_2)} \qquad 2.22$$

which can be written in the form of equation 2.19, where

$$\mathbf{w} = \Sigma^{-1}(\mu_1-\mu_2) \qquad 2.23$$

$$w_0 = -\frac{1}{2}\mu_1^{\mathrm{T}}\Sigma^{-1}\mu_1 + \frac{1}{2}\mu_2^{\mathrm{T}}\Sigma^{-1}\mu_2 + \ln\frac{P(C_1)}{P(C_2)}. \qquad 2.24$$

It is clear to see that the discriminant function is linear in the components of **x**, where **w** is referred to as the d-dimensional *weight vector* and the parameter w_0 is called the *bias*.

In practice, the parameters of the Gaussian distributions of the class-conditional probability densities need to be estimated. The estimation of the parameters, as defined in [Bishop 1995 p. 41], are given by

$$\hat{\mu} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}^n \qquad 2.25$$

and

$$\hat{\Sigma} = \frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{x}^n - \hat{\mu}\right)\left(\mathbf{x}^n - \hat{\mu}\right)^{\mathrm{T}} \qquad 2.26$$

where N is given by number of samples in $\mathbf{x}$. If one assumes equal covariance matrices then the estimated pooled covariance matrix can be used [Ersbøll and Conradsen, 2003], and for a two class problem it is defined by

$$\begin{aligned}\hat{\Sigma}_p &= \frac{1}{N_1+N_2-2}\left(\sum_i\left(\mathbf{x}_1^n-\hat{\mu}_1\right)\left(\mathbf{x}_1^n-\hat{\mu}_1\right)^{\mathrm{T}}+\sum_i\left(\mathbf{x}_2^n-\hat{\mu}_2\right)\left(\mathbf{x}_2^n-\hat{\mu}_2\right)^{\mathrm{T}}\right)\\ &= \frac{1}{N_1+N_2-2}\left(\left(N_1-1\right)\cdot\hat{\Sigma}_1+\left(N_2-1\right)\cdot\hat{\Sigma}_2\right)\end{aligned} \qquad 2.27$$

where N_1 and N_2 are the number of samples in each class, and $\hat{\mu}_1$ and $\hat{\mu}_2$ are the estimated expectations or means in each class, respectively. If one assumes two distinct covariance matrices then the resulting classifier is no longer linear, but rather *quadratic* and is given by

$$\begin{aligned}y(\mathbf{x}) = &-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\left(\Sigma_1^{-1}-\Sigma_2^{-1}\right)\mathbf{x}-2\mathbf{x}^{\mathrm{T}}\left(\Sigma_1^{-1}\mu_1-\Sigma_2^{-1}\mu_2\right)\\ &-\mu_1^{\mathrm{T}}\Sigma^{-1}\mu_1+\mu_2^{\mathrm{T}}\Sigma^{-1}\mu_2+\ln\frac{P(C_1)}{P(C_2)}+\frac{1}{2}\ln\frac{|\Sigma_2|}{|\Sigma_1|}.\end{aligned} \qquad 2.28$$

The above function is known as the quadratic discriminant function and as its name states the decision boundary is of a quadratic form and therefore has the potential of a more precise discrimination than the former linear discriminant function. In the above analysis a two-class problem was assumed, but this can easily be extended to several-class problem, which, if one is interested, can be found in [Bishop, 1995] and in [Ersbøll and Conradsen, 2003] to mention a few.

The linear discriminant functions, as well as the quadratic, can be regarded as *supervised learning techniques* due to the fact that they take the target data into account, thus giving substantially more optimal results than compared with *unsupervised techniques*.

2.5 Generalization through Cross-Validation

As previously mentioned, the goal of model training is not to learn an exact representation of the training data itself, but rather to build a statistical model of the process which generates the data. This is important if the model is to exhibit good *generalization*, i.e. to make good predictions for new inputs. In practical applications, one seeks to find the best overall performing model and the most important technique for doing this is called *cross-validation* [Bishop 1995 p. 332].

2.5.1 Cross-validation

Since the goal is to find the network having the best performance on new data, the simplest approach to the comparison of different networks is to evaluate the error function using data which is independent of that used in the training process. Various models are trained by minimization of an error function defined with respect to a *training* data set. In this projects context the accuracy or the area under the ROC curve is maximized and therefore the classification error is minimized. The performance of the model is then compared by evaluating the error function using an independent *validation* set, and the model having the smallest classification error with respect to the validation set is selected. This approach is called the *hold out method*. Since this procedure can itself lead to some over-fitting to the validation set, the performance of the selected network should be confirmed by measuring its performance on a third independent set of data called a *test* set. Though if one has access to large data sets, e.g. experimental data, then it is enough to perform training and testing.

In practice, though, the availability of labeled data may be limited. In such cases the procedure of cross-validation can be adopted. This procedure divides the training set at random into S distinct segments, then trains a network using data from $S - 1$ of the segments and tests its performance, by evaluating the error function, using the remaining segment. This process is repeated for each of the S possible choices for the segment which is omitted from the training process, and the test errors averaged over all S results. Such a procedure allows the use of high proportion of the available data (a fraction $1 - 1/S$) to train the networks, while also making use of all data points in evaluating the cross-validation error. The disadvantage of such an approach is that it requires the training process to be repeated S times which in some circumstances could lead to a requirement for large amounts of processing time. Data can in extreme cases be very scarce, in this case one should set $S = N$ for a data set with N data points, which involves N separate training runs per network, each using $(N - 1)$ data points. This limit is known as the *leave-one-out* method and will be used in chapter 6.

2.6 Performance Measures for Binary Classifications

During the process of cross-validation one has to evaluate the overall performance of the classifier or model. To measure the performance of a classification the estimation of the *confusion matrix* reveals relevant information. A confusion matrix contains information about actual and predicted classifications done by a classification system. Performances of such systems are commonly evaluated using the data in the matrix. Each of the possible output classes are represented both along the *x* and the *y*-axis. Each cell of the matrix shows the number of patterns from the class on the *y*-axis that was mapped into the class on the *x*-axis. Diagonal entries represent patterns that were correctly classified, whereas misclassifications are displayed off the diagonal. The confusion matrix can help identify which classes are being learned well and which classes are being confused. This may identify additional features that can be added in order to help the classifier differentiate among confused classes [Fawcett 2003].

		Truth	
		Positive	**Negative**
Model	**Positive**	***a***	***c***
	Negative	***b***	***d***

Figure 5. A schematic illustration of a confusion matrix for a binary problem.

In Figure 5 a schematic illustration of a confusion matrix for a binary classification problem is shown. The lower-case letters in the matrix have the following meaning;

- ***a*** is the number of correct classifications that an observation is positive. This number is also called the number of *hits*.
- ***b*** is the number of erroneous classifications that an observation is negative This number is also called the number of *misses*.
- ***c*** is the number of erroneous classifications that an observation is positive This number is also called the number of *false alarms*.
- ***d*** is the number of correct classifications that an observation is negative This number is also called the number of *correct rejections*.

There has been defined several standard terms of the combinations of the elements in the confusion matrix for binary problems. In the following the terms used in this project are presented. The *accuracy* (AC) is the proportion of the total number of predictions that are correct. It is determined using the using the following equation;

$$\text{AC} = \frac{a+d}{a+b+c+d} \qquad 2.29$$

The *true positive rate* (TP) is the proportion of positive cases that were correctly identified, is calculated using the following equation;

$$\text{TP} = \frac{a}{a+b} \qquad 2.30$$

The *false positive rate* (FP) is the proportion of negatives cases that were incorrectly classified as positive, is calculated using the following equation;

$$\text{FP} = \frac{c}{c+d} \qquad 2.31$$

The accuracy determined using equation 2.29 may not be an adequate performance measure when the number of negative cases is much greater than the number of positive cases [Fawcett 2003]. Suppose there are 1000 cases, 995 of which are negative cases and 5 of which are positive cases. If the system classifies them all as negative, the accuracy would be 99.5%, even though the classifier missed all positive cases. A way of examining the performance of a classifier is through the *receiver operating characteristic* (ROC) graph.

In signal detection theory, a ROC is a graphical plot of the FP as a function of TP as its decision boundary, also called *discrimination threshold* or *criterion*, is varied across the range of all possible observation values. The ROC can also be represented equivalently by plotting the 1-*specificty* as a function of the *sensitivity*. A completely random predictor would yield a straight line from the origin to the upper right corner, as illustrated by the solid line in Figure 6. This is due to the fact that as the decision boundary is moved, equal numbers of true and false positives are registered. This would correspond to a total overlap of two probability density functions. In the case where there is separation, the ROC curve becomes bowed, and for larger separations the more bowed the curve becomes. In Figure 6 a schematic illustration of two classes with a large and a small overlap is shown together with their corresponding ROC curves. In terms of noise, the large overlap is equivalent to large noise levels and the small overlap to low noise levels in a system. As it can be seen the area under the ROC curve increases from 0.5 for a random classifier to 1 for an ideal classifier: thus the area can be used as a measure of the discriminability of the two classes [Fawcett 2003]. This measure together with the accuracy measure will be used to compare results from real and experimental data, respectively.

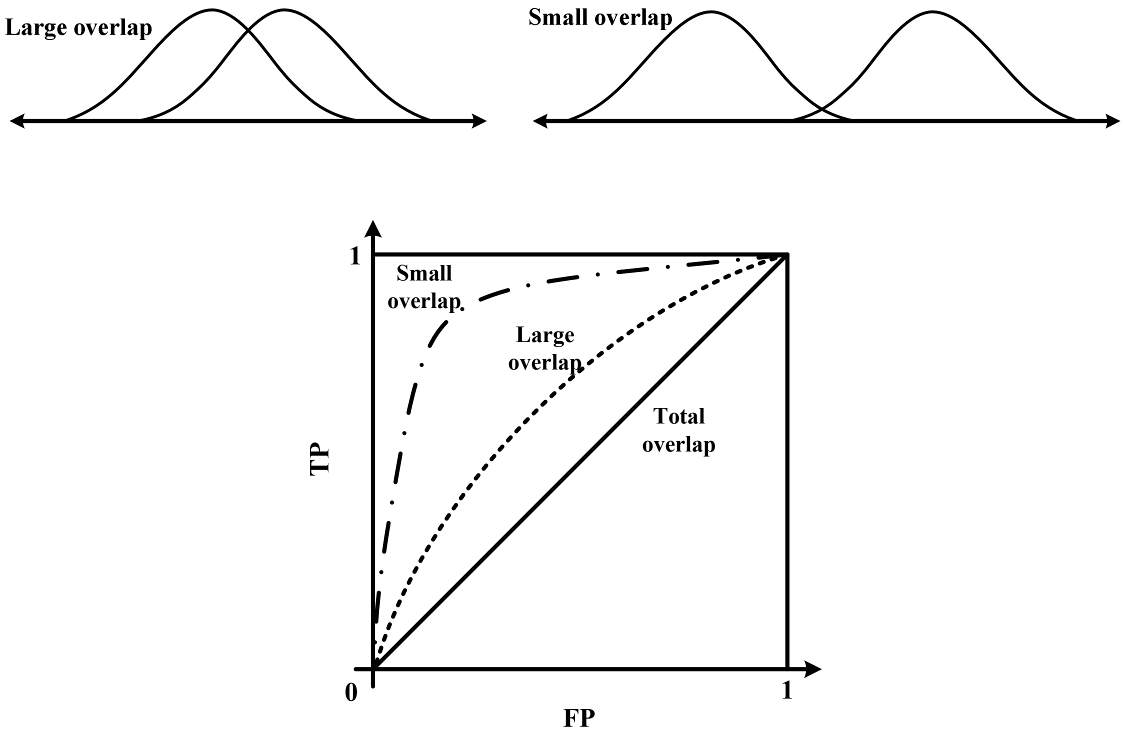


Figure 6. A Schematic illustration of two classes with large overlap and the small overlap, corresponding to the dotted and the dash-dotted lines in the ROC curve, respectively. The solid line is, as explained in the text, a completely random predictor, representing total overlap.

2.7 Principal component analysis (PCA) and Singular Value Decomposition (SVD)

The goal in dimensionality reduction is to preserve as much of the relevant information as possible, and to make the essential structure in the data more visible or accessible. The procedures discussed in the following relies entirely on the input data itself without reference to the corresponding target data, and can be regarded as a form of *unsupervised* learning. While they are of great practical significance, the neglect of the target data information implies they can also be significantly sub-optimal, as mentioned previously. The current and the following section will focus on unsupervised techniques using linear transformations for dimensionality reduction.

The goal in dimensionality reduction using PCA is to map vectors $\mathbf{x}^n$ in a d-dimensional space ($x_1,\ldots, x_d$) onto vectors $\mathbf{z}^n$ in an M-dimensional space ($z_1,\ldots, z_M$), where $M < d$. The vector $\mathbf{x}$ can be represented, without loss of generality, as a linear combination of a set of d orthonormal vectors $\mathbf{u}_i$

$$\mathbf{x} = \sum_{i=1}^{d} z_i \cdot \mathbf{u}_i \qquad 2.32$$

where the vectors $\mathbf{u}_i$ satisfy the *orthonormality* relation (orthogonal and unit length vectors)

$$\mathbf{u}_i^{\mathrm{T}}\mathbf{u}_j = \delta_{ij} \qquad 2.33$$

Explicit expressions for the coefficients z_i in equation 2.32 can be found by using equation 2.33 to give

$$z_i = \mathbf{u}_i^{\mathrm{T}}\mathbf{x} \qquad 2.34$$

which can be regarded as a simple rotation of the coordinate system from the original x's to a new coordinate given by the z's. Reducing the dimensionality is equivalent to retaining only a subset $M < d$ of the basis vectors $\mathbf{u}_i$, so that only M coefficients are used to create z_i. The optimal linear dimensionality reduction procedure (in the sense of least squares) is determined by minimization of the following error function:

$$\begin{aligned} E_M &= \frac{1}{2}\sum_{i=M+1}^{d}\sum_{n}\left\{\mathbf{u}_i^{\mathrm{T}}\left(\mathbf{x}^n - \hat{\mu}\right)\right\}^2 \\ &= \frac{1}{2}\sum_{i=M+1}^{d}\mathbf{u}_i^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{u}_i \\ &= \frac{1}{2}\sum_{i=M+1}^{d}\lambda_i \end{aligned} \qquad 2.35$$

where Σ is the covariance of the set of vectors $\{x^n\}$ and $\hat{\mu}$ is the mean. Thus, the minimum error is obtained by choosing the $d - M$ smallest eigenvalues, and their corresponding eigenvectors, to be discarded. It can be shown that the minimum of the error function occurs when the basis vectors $\mathbf{u}_i$ satisfy

$$\boldsymbol{\Sigma}\mathbf{u}_i = \lambda_i\mathbf{u}_i \qquad 2.36$$

thus being the eigenvectors of the covariance matrix.

The first principal component is the linear combination (with normed coefficients) of the original variables which has the largest variance. The m'th principal component is the linear combination (with normed coefficients) of the original variables which is uncorrelated with the $m - 1$ first principal components and has the largest variance.

An efficient way of calculating the principal components is by using the singular value decomposition. The equation for SVD of a matrix $\mathbf{X}$ ($m \times n$) is the following;

$$\mathbf{X} = \mathbf{USV}^{\mathrm{T}} \qquad 2.37$$

where $\mathbf{U}$ ($m \times m$) and $\mathbf{V}$ ($n \times n$) are orthogonal matrices with the *left* and *right*, respectfully, *singular vectors*, and $\mathbf{S}$ ($n \times n$) is a diagonal matrix with *singular values*. By convention, the ordering of the singular vectors is determined by high-to-low sorting of singular values, with the highest singular value in the upper left index of the **S** matrix. For a square, symmetric matrix **X**, SVD is equivalent to the solution of the eigenvalues problem;

$$\mathbf{X} = \mathbf{U}\Lambda\mathbf{U}^{\mathrm{T}} \qquad 2.38$$

$$\mathbf{XU} = \Lambda\mathbf{U} \qquad 2.39$$

where Λ is a diagonal matrix with eigenvalues of **X**. **V** and **U** are now equal and hold the corresponding eigenvectors. Equation 2.39 is equivalent to equation 2.36, when **X** is substituted by a covariance matrix Σ.

The classic use of PCA is usually described as an unsupervised dimensionality reduction technique, as was presented above. Although in this report, there will be examined to what extent the PCA can be made supervised or how prior knowledge is included by augmenting the data $\mathbf{x}^n$ by a class label vector. This technique will be described later in section 2.10.

2.8 Independent component analysis (ICA)

ICA distinguishes itself from other methods, such as PCA, in the sense that it looks for components that are both statistically independent, and non-Gaussian. In practical situations, one cannot in general find a representation where the components are really independent, but one can at least find components that are as independent as possible.

The general model for ICA is that the sources are generated through a linear basis transformation, where additive noise can be present. In the following, the noiseless model is considered, which has the form,

$$\mathbf{X} = \mathbf{AS}\,, \qquad \mathbf{X}_{m,n} = \sum_{k=1}^{N_k} \mathbf{A}_{m,k} \cdot \mathbf{S}_{k,n} \qquad 2.40$$

where **X** is the matrix holding the N_m mixed or observed signals in each row with N samples, **A** is the $N_m \times N_k$ basis transformation or *mixing matrix*, and **S** is the matrix holding the N_k independent source signals in rows of N samples. In the special case when assuming that the mixing matrix is an invertible square matrix and that no noise is present, the infomax solution is achieved [Kolenda 1998].
The estimation of **S**, will be called **Y**, and is found by solving

$$\mathbf{Y} = \mathbf{WX} = \mathbf{A}^{-1}\mathbf{X} \qquad 2.41$$

which is equivalent to unmixing the observed signals with the inverse of the estimated mixing matrix **W**. Thus the goal of ICA algorithm is to estimate **W** by assuming independent and non-Gaussian sources.

2.8.1 Independent sources

The fundamental principle in ICA is that the sources are independent of each other. When the distribution of **s** can be written as the product of the distributions for each of the components separately in the form

$$p(\mathbf{s}) = \prod_{k=1}^{N_k} p(s_k) \qquad 2.42$$

then **s** is said to be statistically independent. If the signals are uncorrelated for all moments, including the higher order moments, then they are considered independent. To achieve independence, however, it is sufficient to estimate no more than fourth order moments [Kolenda 1998]. The fourth order moment can be expressed as the signal's kurtosis γ, and describes the "top-steepness" of a signal. Since ICA is an unsupervised algorithm, the estimated sources will converge to a false optimum if the true sources are not independent.

2.8.2 Source probability distribution

Recovering the source signals involves more or less directly the source signals probability distributions. In the case of zero mean probability distributions, the error made by not matching the source distributions (if not too gross) results in merely a scaling of the estimated signals [Kolenda 1998]. The basic properties of the underlying distributions need therefore to be respected, although it might not make the optimization of the ICA algorithm unstable.

2.8.3 Mixing matrix

The mixing matrix **A** can be thought of as being a non-orthogonal transformation basis, as opposed to PCA. The columns in **A** are linearly independent and must have full rank [Kolenda 1998]. The matrix can at least be recovered from the true mixing matrix up to a scaling and permutation of the matrix rows. The number of sources, hence columns in **A**, are generally not known and must be estimated. In the case where the number of sources and number of observed signals are the same, the problem simplifies and the un-mixing matrix can be found as the inverse of **A**.

2.9 Fisher's Linear Discriminant (FD)

The Fisher's linear discriminant aims to achieve an optimal linear dimensionality reduction with respect to prior class information, and similar to linear discriminant analysis can be regarded as a form of supervised learning technique, as opposed to the two latter techniques. The FD uses a linear projection of the data onto a one-dimensional space, so that an input vector **x** is projected onto a value y given by

$$y = \mathbf{w}^{\mathrm{T}}\mathbf{x} \tag{2.43}$$

where **w** is a vector of adjustable weight parameters. In general, the projection onto one dimension leads to a considerable loss of information and classes which are well separated in the original d-dimensional space may become strongly overlapping in one dimension. However, by adjusting the components of the weight vector **w** we can select a projection which maximizes the class separation.

In the following a two-class problem is considered, this can of course be extended to several classes [Bishop 1995 p. 100]. There are N_1 points of class C_1 and N_2 points of class C_2. The mean vectors of the two classes are estimated by

$$\hat{\mu}_1 = \frac{1}{N_1}\sum_{n \in C_1}\mathbf{x}^n , \qquad \hat{\mu}_2 = \frac{1}{N_2}\sum_{n \in C_2}\mathbf{x}^n \tag{2.44}$$

The resolution proposed by Fisher is to maximize a function which represents the difference between the projected class means, normalized by a measure of the *within-class* scatter along the direction of **w**.

The projection in equation 2.43 transforms the set of labeled data points in **x** into a labeled set in the one-dimensional space y. The within-class scatter of the transformed data from class C_k is described by the within-class covariance, given by

$$\mathbf{S}_k^2 = \sum_{n \in C_k}\left(\mathbf{x}^n - \hat{\mu}_k\right)\left(\mathbf{x}^n - \hat{\mu}_k\right)^{\mathrm{T}} \tag{2.45}$$

and we can define the total within-class covariance for the whole data set to simply be

$$\mathbf{S}_W = \mathbf{S}_1^2 + \mathbf{S}_2^2 \tag{2.46}$$

It can be shown that FD is given by

$$\mathbf{w} \propto \mathbf{S}_W^{-1}\left(\hat{\mu}_2 - \hat{\mu}_1\right) \tag{2.47}$$

As mentioned, this is not a discriminant but rather a specific choice of direction for projection of the data down to one dimension.

2.10 Augmented data sets

In this chapter, the concept of augmenting data sets with a class label vector and then performing classical statistical analysis is investigated. The idea behind augmented data sets is to improve the statistical analysis by enforcing the concept of classes on the data set.

The motivation for this approach was found in [Meinicke *et al.*, 2004]. Here they propose an extension to the techniques that are based on an augmentation of the data space by additional dimensions for encoding class-membership, such as the ICA-FX method reported Kwak and Choi, most recently in [Kwak and Choi 2003]. Thus, unlike most EEG and MEG applications of ICA which aim at source separation [Jung *et al.* 2000], [Jung *et al.* 2001], e.g. for isolation of muscle artifacts, the latter article proposes an ICA technique as an analysis tool in order to identify discriminative features in a two class problem.

In the following analysis, there are two different objectives. The first one being to investigate how PCA is affected by augmented data sets, and the second being whether a standard ICA technique (Infomax) can be used when augmenting data sets or if it is necessary to utilize an ICA technique proposed in [Meinicke *et al.*, 2001], which allows for non-parametric source models. These, supposedly, give added flexibility and are important for capturing features with multimodal distributions, which can occur in augmented data techniques or are inherent for the measured distributions themselves [Kwak and Choi 2003].

In the following section, the testing method for this investigation is described. In section 2.10.3, it will be shown that augmenting data sets can further improve the discriminative features of the PCA procedure. The results will show that the augmented PCA (APCA) procedure generally gives just as good results as the Fischer's linear discriminant (FD).

2.10.1 The General Concept

Consider a data set $\mathbf{X}$ consisting of two distinct classes in d-dimensions, where each data point is known to belong to either C_1 or C_2. The class label set $\boldsymbol{g}$ is a binary vector with the same length as $\mathbf{X}$, which indicate the class membership of the corresponding data points. When speaking about augmented data sets it is simply intended that $\mathbf{X}$ is augmented with $\boldsymbol{g}$, giving

$$\tilde{\mathbf{X}} = \begin{bmatrix} \mathbf{X} \\ \boldsymbol{g} \end{bmatrix} \qquad 2.48$$

The augmented data set $\tilde{\mathbf{X}}$ now has the target values incorporated into the data set and this results in a $d + 1$ dimensional data set. The idea is now that unsupervised learning techniques, i.e. PCA or ICA, in some way become supervised due to the fact that the target values are taken into account, even though the information is embedded directly into the data set itself.

2.10.2 Augmented PCA

Having augmented the data set as described in the previous section, one can now perform the standard PCA on the augmented data set. As mentioned, this can be done by estimating the covariance matrix of the augmented data $\Sigma_{\tilde{\mathbf{X}}}$, which results in a d + 1 × d + 1 matrix. Performing SVD on the $\Sigma_{\tilde{\mathbf{X}}}$ results in

$$\Sigma_{\tilde{\mathbf{X}}} = \mathbf{U}\Lambda\mathbf{U}^{\mathrm{T}} \qquad 2.49$$

where **U** correspond to the d + 1 eigenvectors.

Performing PCA on $\tilde{\mathbf{X}}$ yields d + 1 eigenvectors of length d + 1 ($\mathbf{u}_i$), and considering that **X** is to be projected into the original data space, the last d + 1 coefficient in each eigenvector, corresponding to the class label space, is removed and the d + 1 eigenvectors ($\tilde{\mathbf{u}}_i$) are now of length d. After this deletion, vectors in $\tilde{\mathbf{u}}_i$ are no longer of unit length, thus the vectors in $\tilde{\mathbf{u}}_i$ have to be rescaled, giving $\mathbf{u}_i' = \tilde{\mathbf{u}}_i / \|\tilde{\mathbf{u}}_i\|_2$. This gives,

$$\mathbf{Z}' = \mathbf{U}'^{\mathrm{T}} \cdot \mathbf{X} \qquad 2.50$$

It was observed through extensive testing of this method, that the eigenvector corresponding to the smallest eigenvalue in the $\tilde{\mathbf{X}}$ space or the d + 1 eigenvector generally gave the best discrimination, when the class label was in an interval of]0; g_{max}]. Thus, the optimal discriminant projection is given by;

$$z_{d+1}' = u_{d+1}'^{\mathrm{T}} \cdot \mathbf{X} \qquad 2.51$$

and for a specific value of the class label.

It was also observed that when varying the class label value the accuracy would vary and at some value a maximal accuracy was achieved, which can be observed in Figure 9 and Figure 10. When compared to the Fisher's linear discriminant (FD) and other linear discriminant methods (linear discriminant and conditional distributions), it was seen that the APCA method gave results corresponding to these methods. These statements will be supported by the results of the experiments which follow in the next section.

2.10.3 Investigations and Observations of APCA

In the following investigations, a data set consisting of two Gaussian distributed classes, were used. Two main distributions were used;

1. Distribution 1: a distribution consisting of two different means ($\boldsymbol{\mu}_1 \neq \boldsymbol{\mu}_2$), but with equal covariance matrices ($\Sigma_1 = \Sigma_2$), and
2. Distribution 2: a distribution consisting of two different means ($\boldsymbol{\mu}_1 \neq \boldsymbol{\mu}_2$), and covariance matrices ($\Sigma_1 \neq \Sigma_2$),.

A 2 dimensional schematic illustration of the general form of the distributions are shown in Figure 7.

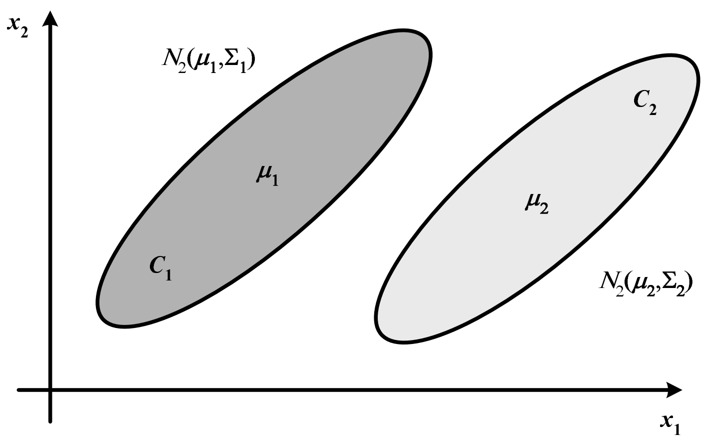


Figure 7. A schematic illustration of a two dimensional two class, C_1 and C_2 respectfully, Gaussian distribution with their corresponding parameters. This illustration corresponds to equal covariance matrices.

For the preliminary investigations, a two class problem consisting of a simple two dimensional Gaussian distribution with non-equal fixed means and equal covariance matrices were used. This made it less complex and therefore easier to achieve analyzable results. Though, after examination of the preliminary results, the data sets were extended to N dimensional 2 class Gaussian distributions with randomly generated means and covariance matrices. This was introduced to test the flexibility and robustness of the technique, particularly for higher dimensions and for as many as possible different Gaussian distributions.

In the preliminary experiments the effect of varying the size of the class label was investigated by testing the discriminatory value of the resulting projections in $\mathbf{U}'$ in equation 2.50. These where compared to the discriminatory value of PCA and FD. For the class label vector $\boldsymbol{g}$ one class, usually the class with the most points, is assigned the label 0, while the other class was assigned a number lying in the interval [1e-5; 10]. In Figure 8, the 2 + 1 dimensional case is schematically illustrated. In can be seen that the distribution, which is assigned the non-zero class label, is moved parallel to the class label axis when the value is changed and therefore the structure of the data is changed resulting in a change in the principal components, as described in the latter section.

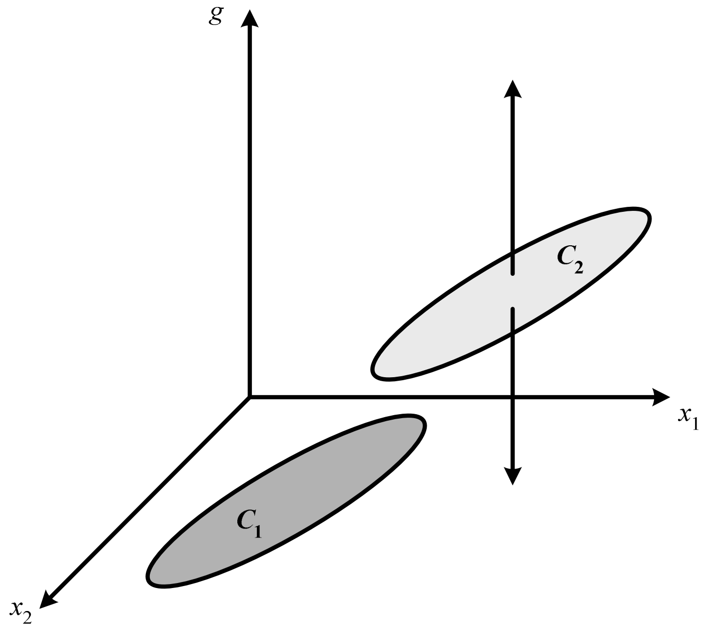


Figure 8. A schematic illustration of two two dimensional Gaussian classes in 2 + 1 dimensions when varying the class label *g* of the APCA. As *g* is varied the two dimensional Gaussian class C_2 is moved parallel to the class label axis.

The cross-validation procedure used in the following resembles the hold out method described in section 2.5.1. First a training set is created, which is used to find all projections of the PCA, APCA and FD, and, their optimal decision boundaries. With this found, a large test set is created from the same distribution as the training set, which then is used to assess the generality of the found projections and decision boundaries. In this project the accuracy is used to compare the results of the various experimental results due to the fact that the numbers of samples in each class were kept equal, i.e. $N_1 = N_2$. Figure 9 and Figure 10, shows the results of the initial experiment for distribution 1 and distribution 2.

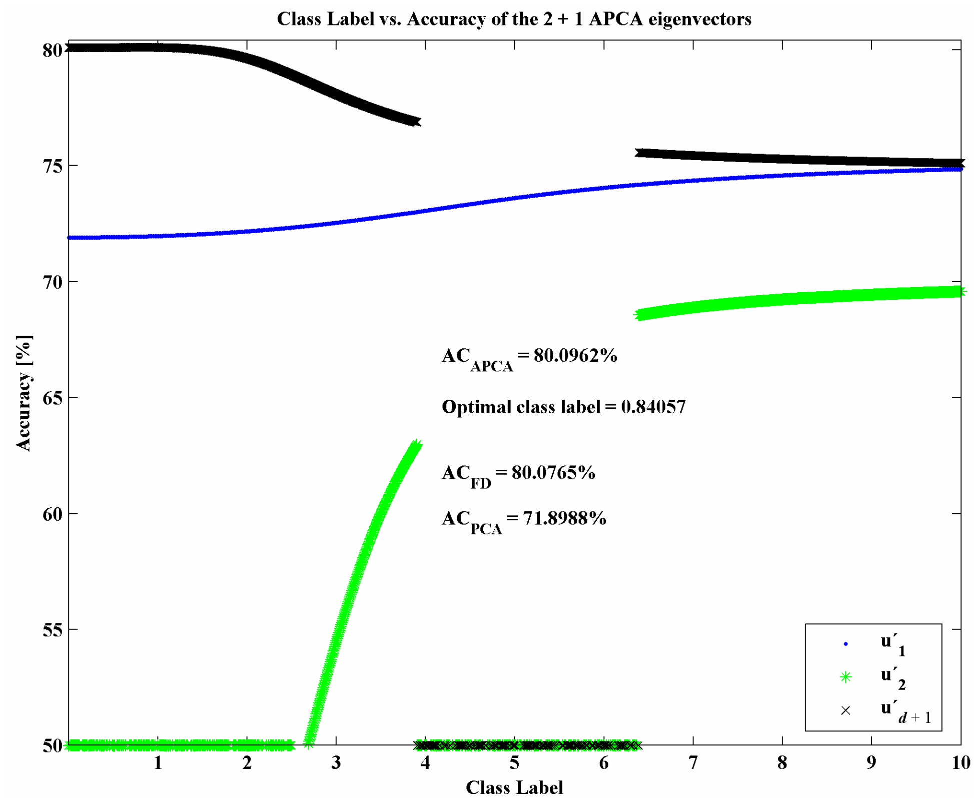


Figure 9. The results showing the discriminatory value expressed in accuracy of the APCA vectors $\mathbf{u}_i$, these are also compared to the best performing PCA eigenvector and Fisher's linear discriminant. The black line corresponds to the AC of the u'_{d+1} vector and clearly this vector seems to give larger AC for smaller class labels. Distribution 1 is used in the test.

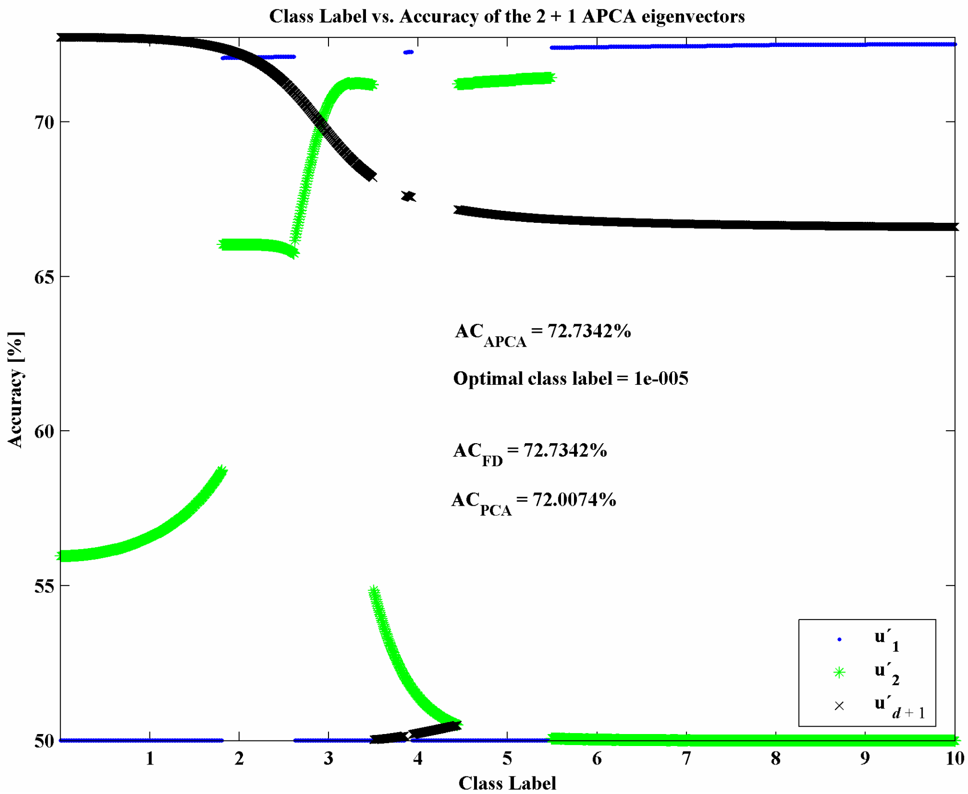


Figure 10. The results showing the discriminatory value expressed in accuracy of the APCA vectors u$_i$, these are also compared to the best performing PCA eigenvector and Fisher's linear discriminant. The black line corresponds to the AC of the u'_{d+1} vector and clearly this vector seems to give larger AC for smaller class labels. Distribution 2 is used in the test.

These are very fascinating results and although nothing general can be said about this procedure yet, it can be seen that the discriminatory value of the PCA is improved by augmenting the data, the class label value has a clear effect, and it also seems that the discrimination is comparable to FD, which could indicate that this procedures discriminatory value is equivalent to linear discriminant functions. Furthermore, as mentioned above, it can be seen that u'^{T}_{d+1} in both cases give the best overall results, though it was observed that in a few cases, for reasons yet unknown, other eigenvectors gave better results.

A Hotelling's T^2 test was performed on difference values, i.e. accuracy difference between APCA and FD, for 100 trails of distribution 1 and distribution 2 for randomly generated distributions. The results can be seen in appendix A and show that for distribution 1, APCA and FD give very similar results, whereas for distribution 2 the results differ in favor of APCA, which generally give slightly better results than FD.

To be able to talk more generally about these observed phenomena the initial tests were extended into higher dimensions with randomly generated parameters, i.e. mean and covariance. The results of these tests are shown in appendix C1 and C2 for distribution 1 and 2, respectively. Figure 11, Figure 12 and Figure 13 are a summary of these tests and will be described in the following.

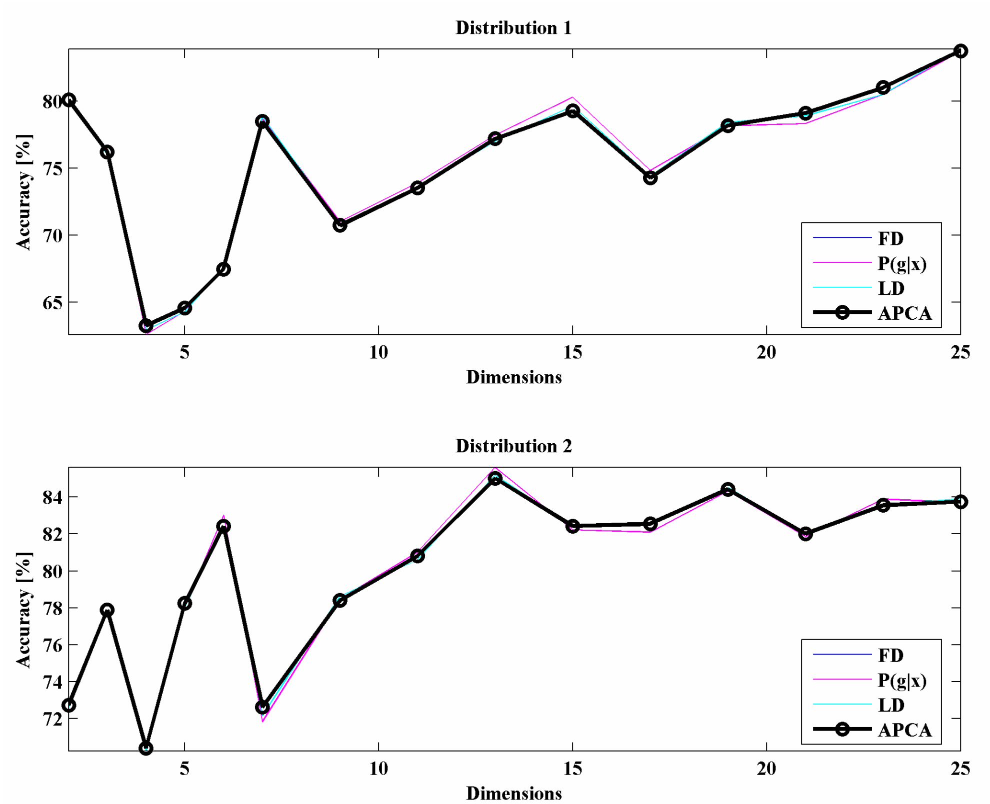


Figure 11 shows the discriminatory value of APCA, linear discriminant function (LD), Fisher's linear discriminant (FD) and conditional distributions (*P*(g|x)) for various dimensions and for both distributions. It is clear from the figure that the APCA performs just as well as the other linear methods.

Figure 11 shows a comparison of the discriminatory values of the APCA method with some standard linear discriminant functions, i.e. linear discriminant analysis, Fisher's linear discriminant and conditional distributions. It can be seen that the APCA method lies extremely close to the other methods and this supports the fact that the APCA method is equivalent to a linear discriminant even for higher dimensions.

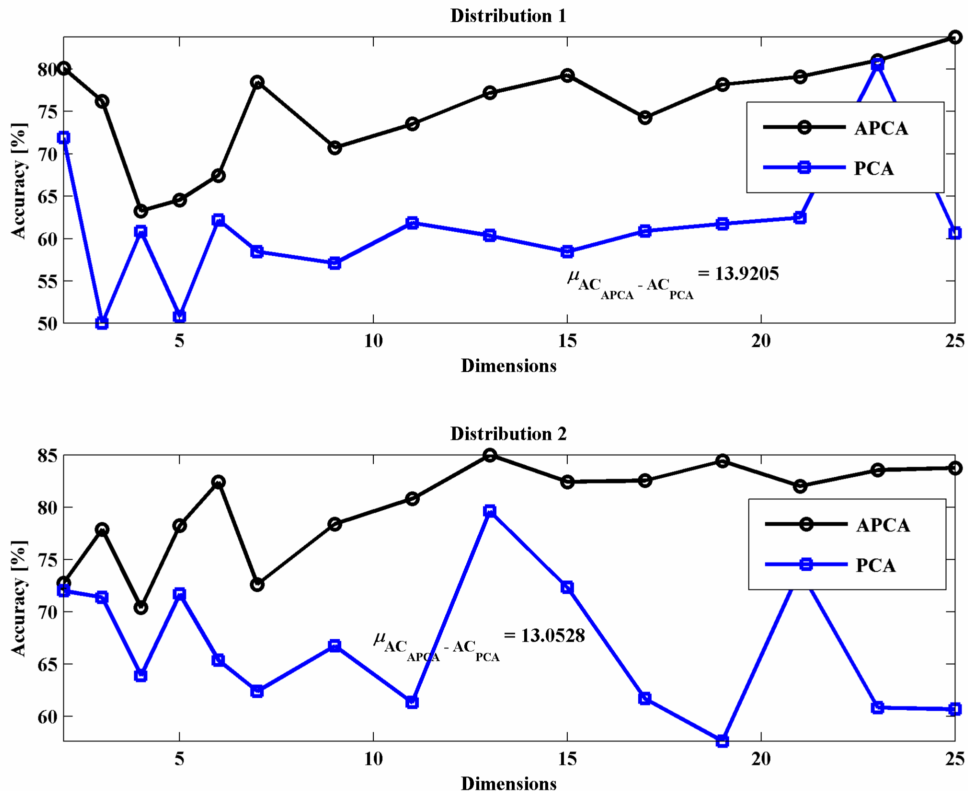


Figure 12. A plot of the comparison of the APCA method and the standard PCA method for both distributions. It is clear to see that the APCA generally performs better than the original PCA method, and in some instances up to 25% better, e.g. for distribution 2 and dimension 19.

The augmented data clearly aids the standard PCA in finding more optimal transformations of the data. Generally APCA out performs the standard PCA with a considerable accuracy margin.

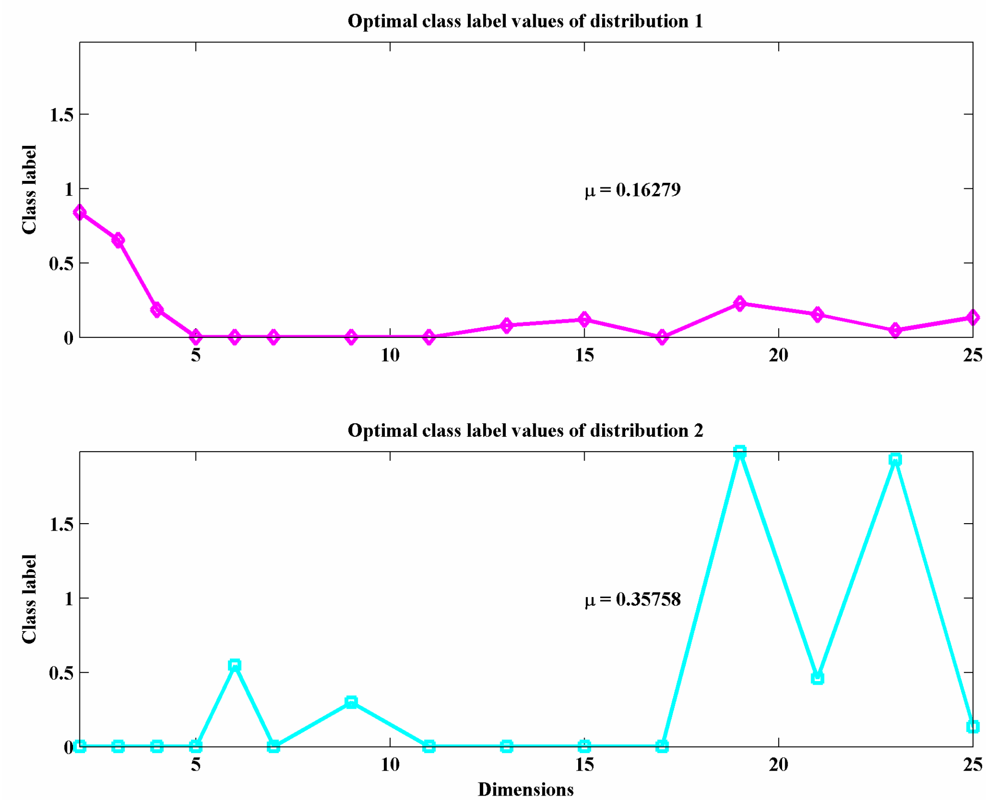


Figure 13. The class label values giving the best discriminatory value for the two distributions. It can be seen that the mean class label value for the first distribution is slightly lower than that of the second distribution. This could support the fact that for data with the same covariance matrices, the optimal class labels tends to be generally smaller than those for non alike covariance matrices.

As observed in the preliminary two dimensional experiments and again in Figure 13, the optimal class label seems to be larger for distributions with non alike covariance matrices. To sum up, it was observed through the above experiments that;

1. the discriminatory value of the standard PCA was increased by using augmented data sets,
2. the discriminatory value of the APCA is comparable with linear discriminant functions, even for higher dimensions, and that
3. u'_{d+1} generally seems to give the best results, though it was observed that for some data sets with little separation between classes, that the "first eigenvector", corresponding to u'_1, out performed u'_{d+1},
4. the value of the class label has influence on the discriminatory value of the vectors in **U**, furthermore, the optimal class labels for distribution 1 are slightly lower than those for distribution 2.

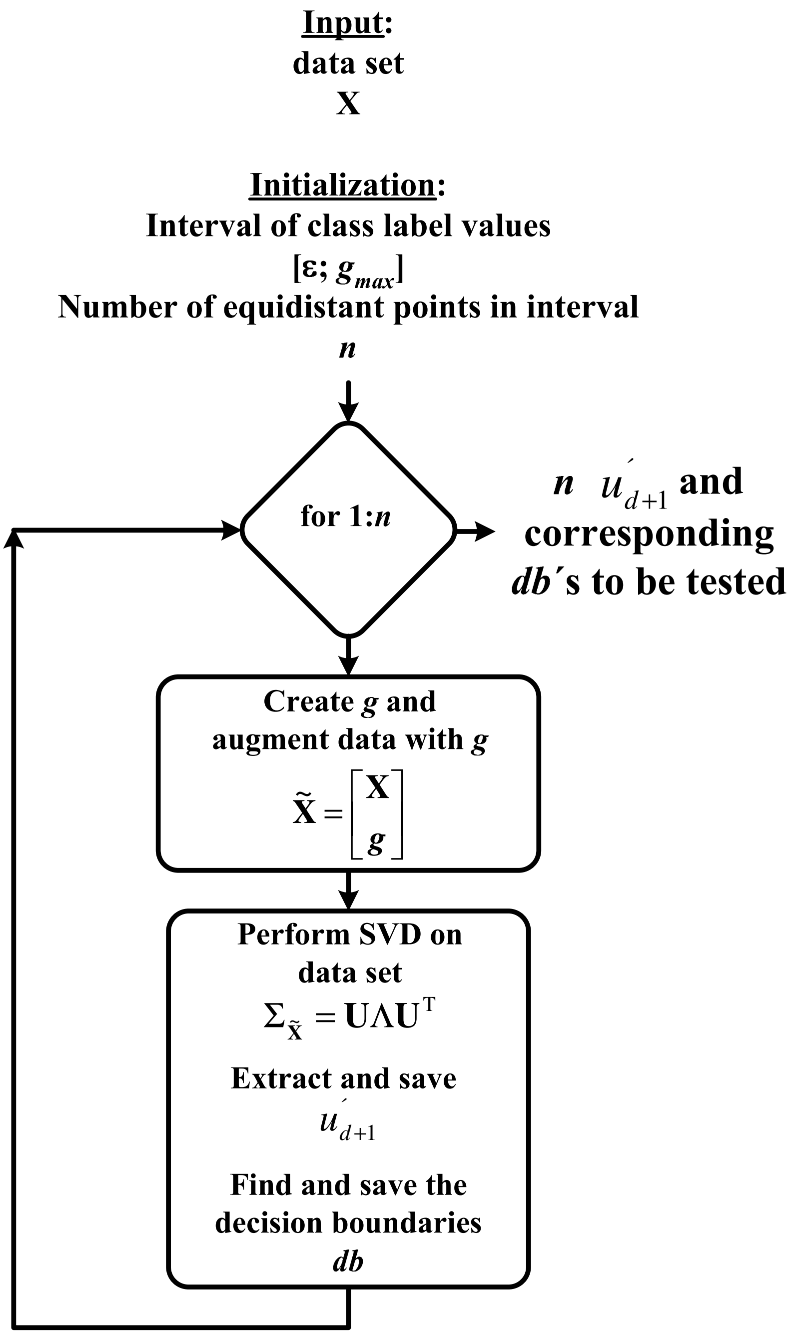


Figure 14. A flow diagram of the training of the APCA. The training procedure results in *n* vectors and their corresponding decision boundaries to be tested making this a computationally inefficient method. In this thesis ε = 1e-5.

Finally, it should be said that the APCA method is computationally ineffective due to the fact that in the training process one needs to run through an interval [ε; g_{max}] of n points, resulting in n vectors that need to be tested, which is just as inefficient. An illustration of the training procedure is shown in Figure 14.

2.10.4 Augmented ICA

The augmented ICA (AICA) method is somewhat equivalent to APCA, although instead of performing PCA on the augmented data, ICA is performed. Due to the fact of the specific nature of ICA, there is no specific vector in the mixing matrix **A** that gives the best discrimination as was observed with APCA. Furthermore, a change in the value of the class label had no apparent effect on the discrimination.

The testing of AICA is not as extensive as for APCA and this analysis leaves many questions uncovered. This section tests whether augmenting a data set with class information also affects the discriminatory value of ICA and to observe the directions found by ICA and AICA. In the following two experiments with different data types are performed, where;

1. Data type 1 has the same distribution as the first experiment in the previous section, i.e. a simple 2 class Gaussian distribution with randomly generated means and covariance matrices, and
2. Data type 2 is a distribution consisting of 2 dimensional 2 class super-Gaussian non-orthogonal data with zero mean, as illustrated in Figure 15.

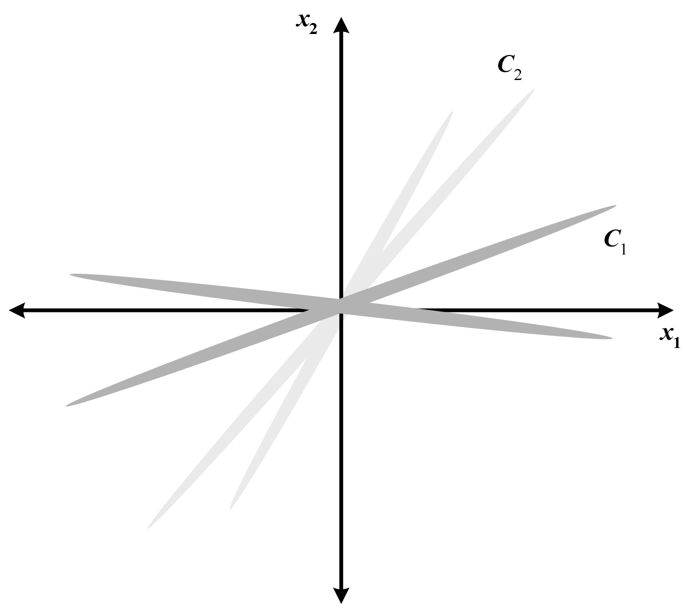


Figure 15. A schematic illustration of a 2 dimensional 2 class super Gaussian non-orthogonal data.

The reason for using the latter data type is due to the fact that the ICA assumes data is non-Gaussian and non-orthogonal (independent), and that data of this type is typically analyzed by ICA.

As mentioned, initial tests showed that the size of the class label had no effect on the discriminatory value, thus throughout the following tests the class label is kept constant to one and zero. Figure 16 shows the result of 100 trials of training and testing the AICA and ICA on different 2 class Gaussian distributions. It can be seen that the discriminatory value of AICA is not as greatly improved by augmenting data as was seen with the APCA compared to PCA, though in some cases a considerable improvement is seen.

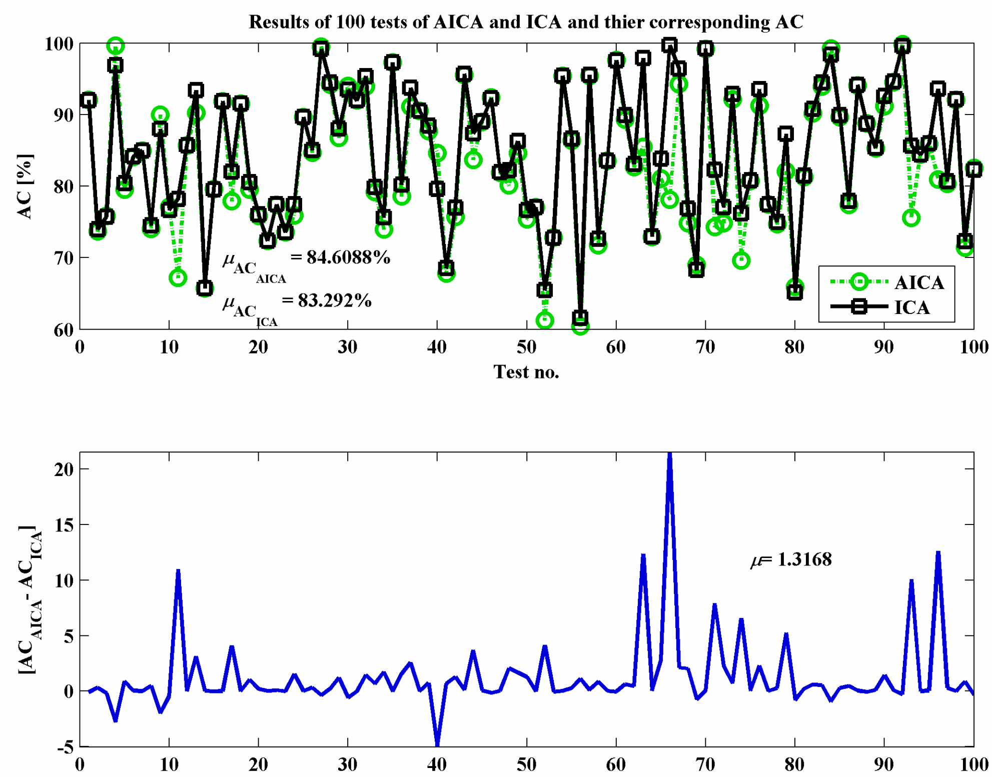


Figure 16. The results of training and testing 100 random 2 dimensional 2 class Gaussian distributions. It is clear that AICA in some cases performs up to 10-20% better than the standard ICA, though the mean difference over all 100 trials shows that AICA only gives a general increase of around 1.3%.

A Hotelling's T^2 test performed on difference values, i.e. accuracy difference between AICA and ICA, for 100 trails of data type 1, which show that the results favor AICA, which generally performed slightly better than the ICA. This can be seen in appendix B.

2.10.5 Investigations and Observations of AICA

In the following experiment a 2 class 2 dimensional super-Gaussian distribution was created and the directions found in **A** for ICA and AICA were examined and compared. Figure 17 shows the directions found by ICA in a single run. The two colored lines represent the directions of the vectors in the resulting 2×2 mixing matrix. Equivalent to APCA, the AICA resulting mixing matrix is a 3×3 matrix where the coefficients corresponding to the class label dimension are deleted, resulting in 3 two dimensional directions. These are displayed in Figure 18. It was observed that the $d + 1$ direction was different for each time AICA was run and in some instances seemed to have random direction. In Figure 18, AICA was run 750 times for the same data set (shown in Figure 17) and for the two first directions (AIC1, AIC2) the directions are consistent with the data directions.

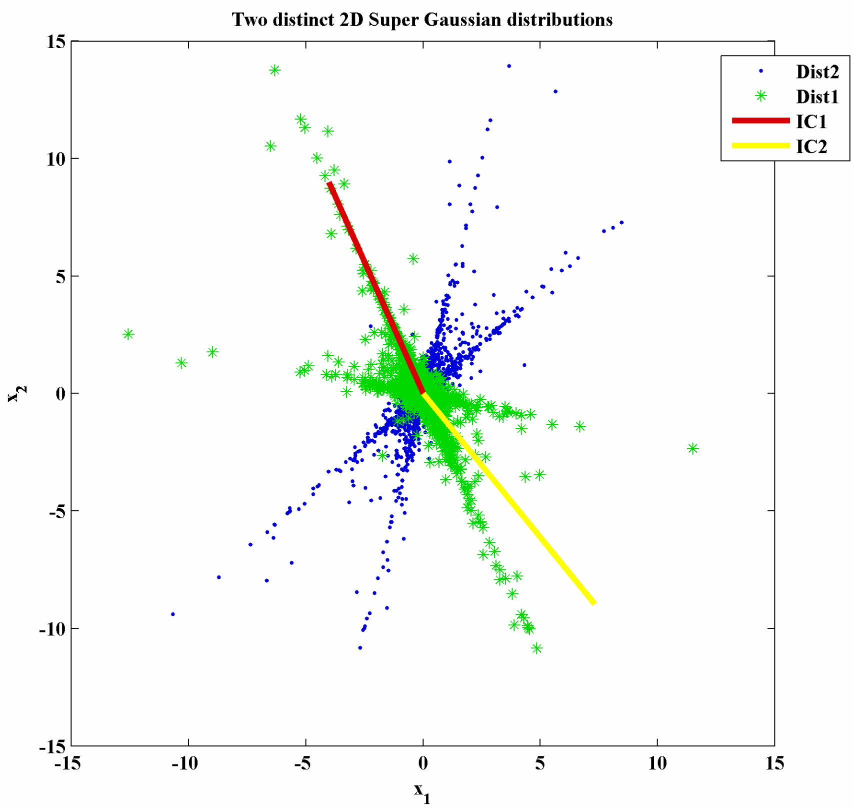


Figure 17. The blue and green points are 2 distinct super-Gaussian distributions in a data set. The directions of A (the mixing matrix) are shown as red and yellow lines superimposed on the distributions. The first independent component (IC1) is in the direction of the largest tail of the green distribution, and the second independent component (IC2) direction is not aligned with any of the other main directions.

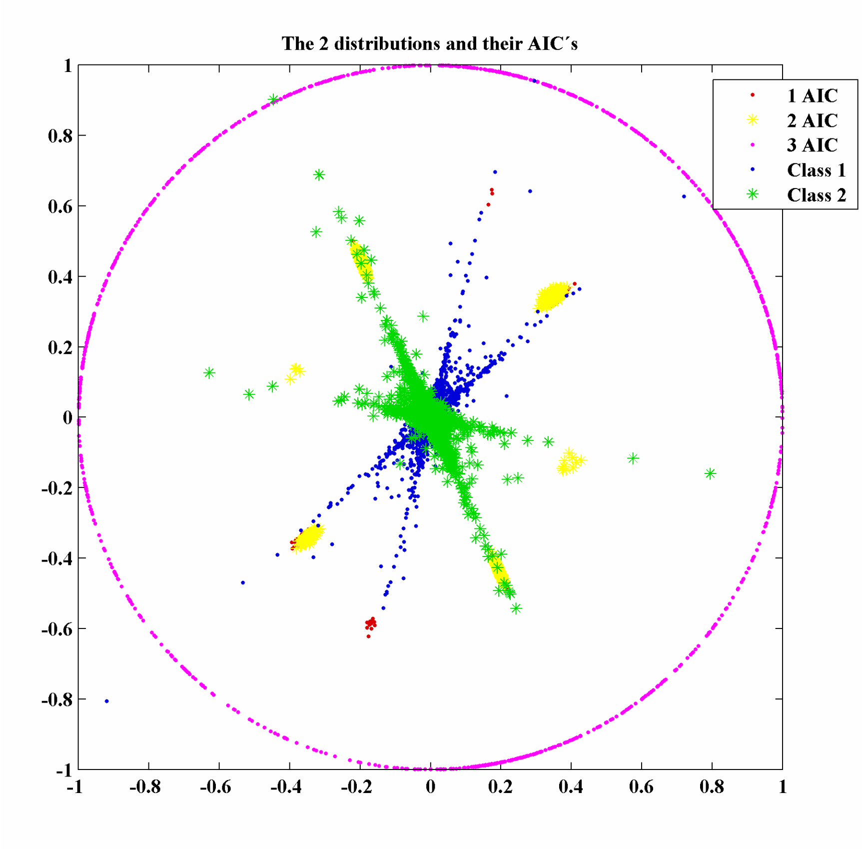


Figure 18. A scaled version of the 2 distributions in Figure 7 is seen with the augmented independent component (AIC) directions. Compared to Figure 7, where the standard ICA was used, the directions seem better tuned to the main tails of the two distributions. This is a very interesting result and supports the fact that the augmented data sets do have a positive effect on ICA. The pink dots that form a circle are the directions of AIC3, and in contrast to the other two directions, AIC3 does not seem to have any fixed direction.

To sum up, it was observed through the above experiments that;

1. although for some instances, the overall improvement using augmented data sets in combination with ICA was not as significant as for APCA using data type 1,
2. when using data type 2, it was observed that the directions of the mixing matrix **A** are affected positively by the augmented data set.

2.11 Conclusion

In this section pattern recognition was introduced, specifically for classification of two class problems; however, all of the procedures can be extended to an arbitrary number of class problems. There was given a brief description of Bayes' theorem, inference and decision making techniques, and generalization. Furthermore, the standard PCA, ICA (infomax), and FD were presented.

The concept of augmented data sets was presented and several results of the preliminary heuristic investiigations using this technique were also presented. Using augmented data sets in combination with PCA yielded very interesting results. The most interesting being, that the discriminatory value of PCA can be increased to the level of linear discriminant functions by using this technique, without estimating any statistical parameters, i.e. mean and variance-covariance. Although, it should be said that the implementation of this procedure is computational inefficient compared to the other linear discriminant techniques presented in this chapter. It is clear that further investigation of this method is necessary to give a lucid and precise explanation of what is going on.

Augmented data sets in combination with ICA were also examined, though the testing, due to limited time, was not as extensive as for APCA. The augmented data sets seem to have little effect, although the accuracy percentage in some case was significantly larger for AICA, and that the directions in **A**, as seen in Figure 18, seem to correlate better with the data set. As for APCA, the method seems promising, though further investigation is needed. The minimal effect of the augmented data sets could be due to the fact highlighted in section 2.10 and mentioned in [Meinicke *et al.*, 2001], i.e. the need for more flexible source models to capture the bimodal nature of the class label vector.

Chapter 3

3 Auditory Perception

In this chapter several aspects of auditory perception is presented. The following aspects will help to understand how to predict and possibly enhance the understandability of a sonification. This chapter is a conglomerate of chapters on the human ear and information processing in the auditory system taken from [Zwicker and Fastl, 1999], [Poulsen 2003], [Hartmann 2000], [Hermann 2002], and [Moore 2003].

The chapter begins with a brief introduction to the human ear and how it works. Thereafter, aspects of psychoacoustics are presented, which contributes important knowledge about the functional mapping from stimuli to sensations. Furthermore, sound processing such as sound segregation into different auditory streams are topics of auditory scene analysis.

3.1 The Human Ear

The ear can be divided into four main parts: the outer ear, the middle ear, the inner ear and the nerve connection to the brain. The first three parts are the peripheral parts of the auditory system, and are shown in Figure 19. The sound will reach the outer ear, progress through the outer ear canal, reach the *tympanic membrane* (the eardrum), transmit the movements to the bones in the middle ear, and further transmit the movements to the fluid in the inner ear. The fluid movements will be transformed to nerve impulses in the inner ear and the impulses are transmitted to the brain through the auditory nerve.

The outer ear, composed of the *pinna* and the auditory canal, influence the sound pressure level in front of the eardrum, though the shoulders as well as the head have also shown to play a crucial part in this. Such signal distortions are used by the auditory system to localize the sound source [Moore 2003]. The auditory system uses differences in timing and level, and spectral profiles between sound signals arriving at the left and right ear to conclude from this information the location of the sound source.

The sound affecting the outer ear consists of oscillations of air particles, whereas the inner ear contains fluids that surround the sensory cells. In order to excite, these cells, it is necessary to produce oscillations in the fluids. Thus, the major function of the middle ear is to ensure the efficient transfer of sound from the air to vibrations in the fluids in the inner ear. As the impedance is much higher in fluid than in air, it is necessary to match the impedance. The impedance matching is achieved by; the difference in the dimensions of the tympanic membrane and that of the *oval window*, and the three small bones, the malleus, incus and stapes, amplify the vibration due to the leverage effect. Transmission of sound through the middle ear is most efficient at middle frequencies (500 – 4000 Hz) [Moore 2003].

The *cochlea* (inner ear) is a crucial part of the ear and provides a key to many aspects of auditory perception, e.g. masking, loudness, and pitch perception. The cochlea is shaped like a snail and is embedded in the extremely hard temporal bone. The cochlea is filled with lymph and is closely connected to the balance organ that contains the three semicircular canals that controls our sense of balance. There are approximately 2.5 turns in the snail shell, and the total length from the base (*basis*) to the top (*apex*) is about 32 mm. As mentioned, the function of the cochlea is to convert the vibrations into nerve impulses in the auditory nerve. Vibrations arriving at the oval window lead to fluid waves that travel from the oval window to the *apical* end. These waves lead to vibrations of the *Reissner membrane* resulting in a relative motion of the *organ of Corti* with its sensory cells. The function of the organ of Corti, which is located on the *basilar membrane*, is the transformation of the mechanical oscillations in the inner ear, into a signal that can be processed by the nervous system. The most important sensory cells are the inner and outer hairs cells, whose electrical potential changes depending on their deviation from the equilibrium position. The potential changes are passed on to the nerve fibers and may at this point cause neuronal pulses.

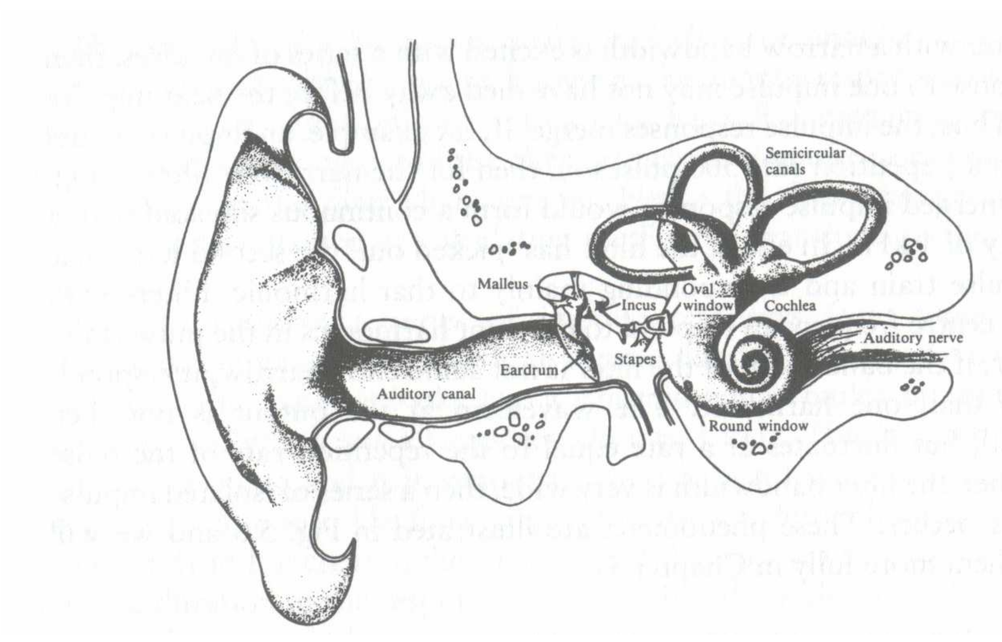


Figure 19. An illustration of the structure of the peripheral auditory system showing the outer, middle and inner ear. This figure is taken from [Moore 2003].

The function of the basilar membrane is very important for the understanding of the function of the ear. The basilar acts like a frequency decomposer. When the ear is exposed to a pure tone a traveling wave propagates along the basilar membrane, and the movement of the basilar membrane will show a certain pattern and the pattern is connected to a certain position on the basilar membrane. If the frequency is changed, the general pattern will not change much, but the position of the pattern will move along the basilar membrane. The basic form of the wave is illustrated in Figure 20, which shows the instantaneous displacement of the basilar membrane (derived from a cochlear model) for two successive instants in time in response to a 200-Hz sinusoid. As a consequence, hair cells along the basilar membrane respond selectively to specific frequencies. The frequency that gives maximum response at a particular point on the basilar membrane is known as the *characteristic frequency*.

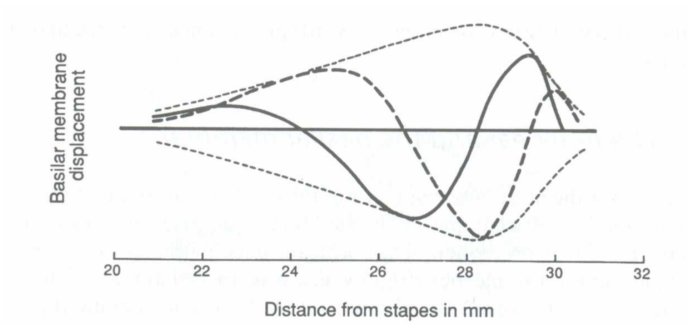


Figure 20. The instantaneous displacement of the basilar membrane at two successive instants in time. The pattern moves from left to right, building up gradually with distance, and decaying rapidly beyond the point of maximal displacement. The dotted line represents the envelope of the waveform. This figure is taken from [Moore 2003].

As stated, the hair cells along the basilar membrane have excitation characteristics that are highly frequency selective. This selectivity, or analytical ability, can be seen in *tuning curves* of primary afferent fibers in the auditory nerve that are driven by the hair cells. A tuning curve is a threshold-of-hearing curve for a single neuron in the auditory nerve, using a sine-tone stimulus. The tuning seen in the neurons resembles non linear filters, which for low pressure levels are very selective, though for higher exposure levels the bandwidth broadens and where the characteristic frequency slightly shifts. This results in a nonlinear perception of sound pressure level, which shall be discussed in the next section. The tuning curve of a hair cell for two levels of stimuli is shown in Figure 21.

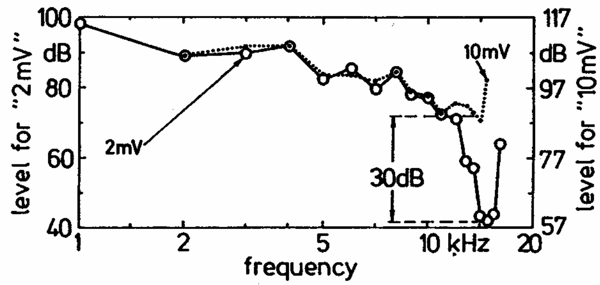


Figure 21. Tuning curve of a hair cell. It shows the sound pressure level of a tone necessary to produce a certain DC receptor potential (circles 2mV, dots 10mV) as a function of frequency. The 10mV is shifted downwards by 17dB so that the two curves are superimposed at low frequencies. This figure is taken from [Zwicker and Fastl 1999].

3.2 Psychoacoustics

Psychoacoustics is a subset of psychophysics, which is the study of the relationship between the magnitude of sensation and the magnitude of a stimulus as measured in conventional physical units [Hartmann 2000]. The only relevant measuring instrument in this connection is a human, i.e. a test subject. Psychoacoustics is about human perception of sound and assesses the relation between stimuli and the hearing sensations they cause. The stimuli are described by physical properties of the sound, e.g. sound pressure, frequency, and location, and can be measured and controlled exactly. Corresponding subjective sensations, such as loudness, pitch, lateralization or localization, cannot be measured as easily as the physical counterpart because they depend on the listener. One goal in psychoacoustics, through psychophysical experiments, is to create sensation magnitudes and to determine their functional dependencies on the stimuli. This is made difficult by the fact that several stimuli may influence a single hearing sensation. For instance, although perceived pitch depends mainly on the frequency, sound pressure level also has a small effect on pitch perception. Perceptual interactions or coupled perceptual parameters are sometimes referred to as the *lack of orthogonality* [Kramer 1994].

Sensations are caused by a stimulus if it exceeds a *perceptual threshold*. Typical for psychoacoustic tasks are to determine the *absolute threshold*, which is the stimulus magnitude for which the corresponding sensation is audible for 50% of the listeners and the *difference threshold*; this is the stimulus increment by which 50% of the listeners have a difference in sensation.

In the following subsections, some psychoacoustic findings of the most important hearing sensations or perceptual parameters are presented. The hearing sensations, which will be presented, are loudness, pitch, and timbre. In [Zwicker and Fastl 1999], other sensations such as roughness, sharpness and fluctuation strength are described, though these are – to the author – a subset of timbre due to its ASA definition, see section 3.2.3. Finally, the a short introduction to the basic concepts of auditory organization, called *auditory scene analysis*, are presented.

3.2.1 Loudness

The psychophysical sensation that corresponds to sound intensity of the stimulus is *loudness*. As explained earlier, the hair cells in the inner ear are excited when a perceivable sound is heard. The more the amplitude of an input sound increases the more hair cells are excited. The intensity of this excitation is perceived as loudness. Loudness is defined as that attribute of auditory sensation in terms of which sounds can be ordered on a scale extending from quiet to loud [Moore 2003]. The stimulus-sensation relation is constructed by results from measurements such as *magnitude estimation* or *loudness comparisons*.

Loudness comparisons gave rise to the *loudness level* measure, which was created to characterize the loudness sensation of any sound. *Loudness level* of a sound is the sound pressure level of a 1 kHz tone in a plane wave and frontal incident that is as loud as the sound; its unit is *phon* [Zwicker and Fastl 1999]. Best known are the loudness levels for different frequencies of pure tones. Lines which connect points of equal loudness in the hearing area are often called *equal-loudness contours* or *isophones*, which are presented in Figure 23. Furthermore the absolute threshold, also known as the *threshold in quiet*, can bee seen to have the value 3 phon, corresponding to 1 kHz tone at 3 dB as being the lowest perceptible loudness level. It is also clear that the perception of loudness depends strongly on the frequency. The line at the top shows the limit of damage risk.

The phon scale does not provide information about the quantative relations between the loudness of two tones. Therefore the *sone* scale is used, which bases on experiments where loudness ratios, such as doubling and halving, are adjusted by human subjects. The loudness of a pure 1 kHz tone of 40 dB is set to 1 sone. Sound with a loudness of 4 sone is thus perceived as four times as loud, also illustrated in Figure 23. This measure is used to estimate the loudness function normally given for the 1 kHz tone, which is illustrated in Figure 22.

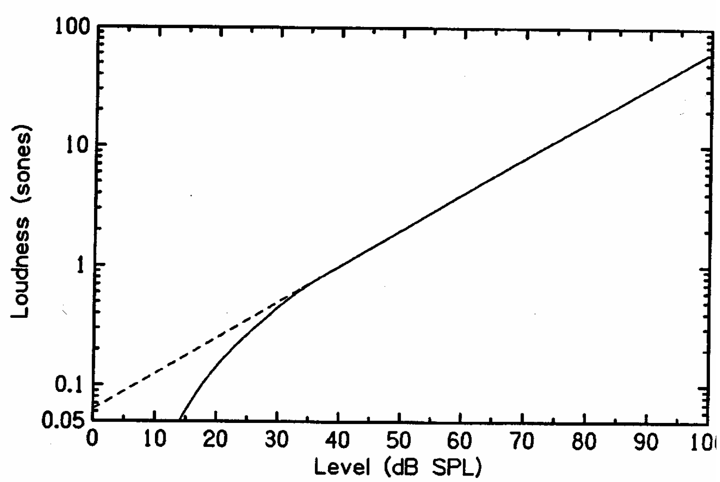


Figure 22. The measured loudness of a 1 kHz tone. The solid line shows the idealized loudness by Stevens and Davis and Zwicker and Fastl [Hartmann 1998]. This figure is taken from [Hartmann 1998].

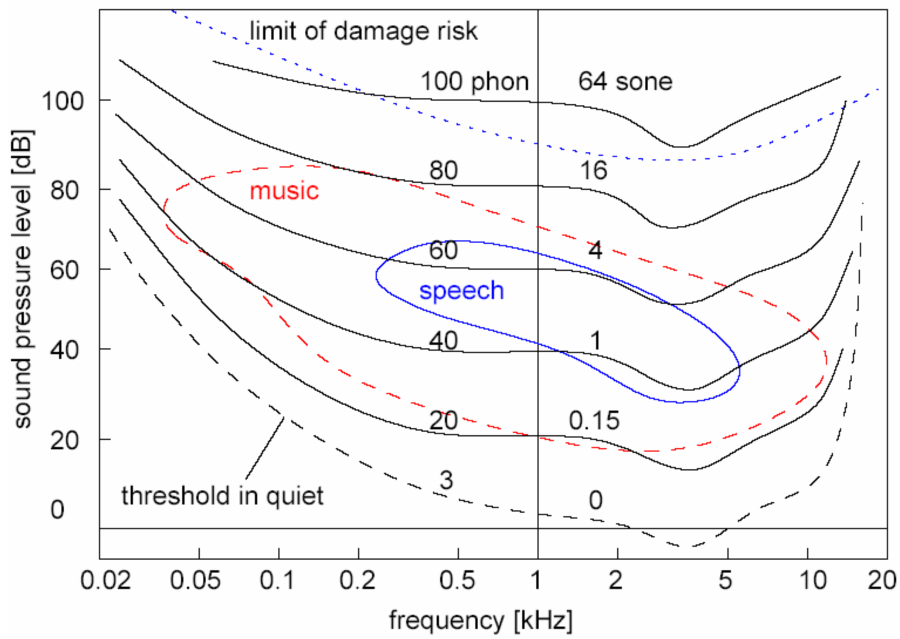


Figure 23. Hearing area. Threshold in quiet, limit of damage risk and equal-loudness contours are shown. Typical regions for music and speech are outlined. This figure is taken from [Hermann 2002].

The above magnitudes are defined for pure tones occurring at one frequency at a time and finding a definition of loudness for pure tones appearing simultaneously and for complex tones is much more difficult. The spectral distribution of a sound can either be narrow (e.g. a sinusoidal tone) or broad (e.g. white noise). A comparison between the loudness level of a pure tone and the level of white noise makes it clear that white noise is perceived much louder than a 1 kHz tone at the same sound pressure level [Zwicker and Fastl 1999]. An important effect when dealing with simultaneous presented stimuli – as in the real world – is the *masking* effect.

When two pure tones with different frequencies or a pure tone and a broad band signal are presented simultaneously the masking effect can be observed. The masking effect can simplest be explained by an everyday situation: imagine having a conversation at a reasonable level by a road, when the traffic becomes heavier conversation becomes unclear and the conversion level is raised. The reason for raising the conversation level is due to the fact that the traffic noise masks your conversation, thus raising the level of the threshold in quiet, called the *masked threshold*. Masking occurs both spectrally and temporally. For a more precise and thorough presentation of masking please see [Zwicker and Fastl 1999], [Moore 2003].

When sound bursts are reduced in duration to less than about 200ms, their level must be raised to remain audible. This dependence on threshold in quiet and masked threshold on the sound duration corresponds to a *temporal integration* of the sound intensity within a time window of 200ms. This is why the effect is frequently called temporal integration or loudness integration [Zwicker and Fastl 1999]. The reason for mentioning this is due to the fact that many auditory display researchers [Barrass and Kramer 1999], [Hermann 2002] use this ability of the ear to detect very short bursts of sound as an advantage over the visual system.

3.2.2 Pitch

Pitch is related to the repetition rate of the waveform of a sound; for a pure tone this corresponds to the frequency. In contrast to the perception of loudness, which is an intensity perception, pitch is a positional perception of the traveling waves on the basilar membrane. Assigning a pitch value to a sound is generally understood to mean specifying the frequency of a pure tone having the same subjective pitch as the sound [Moore 2003]. The property of pitch being a positional perception has consequences on the possible methods for measuring perceptional functions: positional perception functions can be either determined by the *stimulus steps*, or from the measurements of *pitch ratio*.

It has been found, that in the frequency range up to about 1 kHz, the perceived pitch is doubled with a doubling of the frequency. This gradually becomes less true for frequencies above 1 kHz as can be seen in Figure 24. The figure shows that an interval of one octave at low frequency is perceptually smaller than an interval of one octave at high frequency. Ratio pitch was assigned the unit *mel* as it is related to our sensations of melody [Zwicker and Fastl 1999], though their mel scale is slightly different, with 125 mels set to equal to 125 Hz.

The pitch of pure tones depends not only on frequency, but also on other parameters such as sound pressure level, although this dependence is small. The dependence of the pitch of pure tones on the level is displayed in Figure 25. Pitch shifts of pure tones can also occur if additional sounds that produce partial masking are presented.

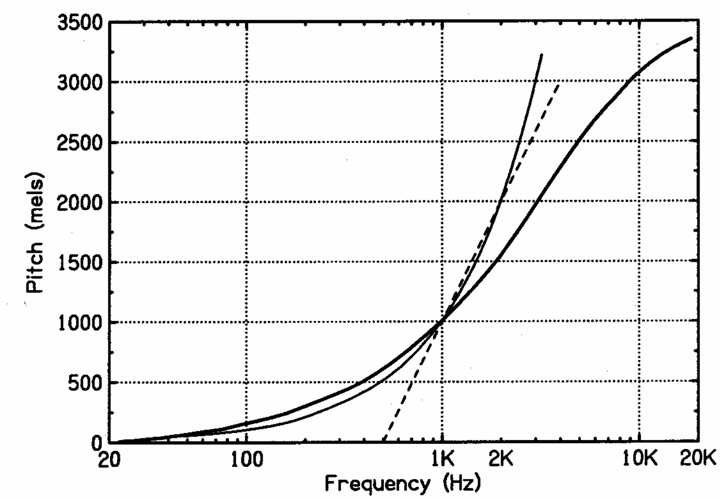


Figure 24. The heavy line is the mel scale from Stevens and Volkman (1940). The thinner line corresponds to pitch given by frequency in Hz, and the dashed line shows pitch proportional to octave number (the musical scale). This figure is taken from [Hartmann 2000].

Complex tones can be regarded as the sum of several pure tones. Although complex tones contain many pure tones, they do not usually produce many pitches, rather one single or perhaps a prominent pitch; the *fundamental frequency*. The *pitch strength* of stimuli, describes the sensation perceiving a sound as having distinct or faint pitch. It can be stated that sounds with line spectra generally elicit relatively large pitch strength, whereas sounds with continuous spectra produce only small values of pitch strength [Zwicker and Fastl 1999].

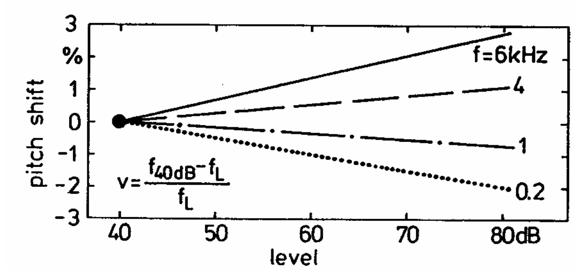


Figure 25 Pitch shift as a function of sound pressure level. This figure is taken from [Zwicker and Fastl 1999].

3.2.3 Timbre

Timbre is defined in ASA (American Standard Association) as that quality which distinguishes two sounds with the same pitch, loudness and duration – spatial position is sometimes also included in the definition. Timbre is generally assumed to be multidimensional, where some of the dimensions have to do with the spectral envelope, the amplitude envelope, etc. The difficulty of timbre identity research is often increased by the fact that many timbre parameters are more similar for different instrument sounds with the same pitch, than for sounds from the same instrument with different pitch. Nevertheless, human perception or cognition generally identifies the instrument correctly [Jensen 2001]. Timbre is related to sensations like brightness, roughness, harshness, sonority, hardness, sharpness, etc. Experimentally, timbre is approached by doing dissimilarity tests. Asking subjects to judge the dissimilarity of a number of sounds and analyzing the results is the essence of the dissimilarity tests. Statistical analysis is used on the dissimilarity scores, and the resulting dimensions are analyzed to find the relevant timbre qualities. In [Jensen 2001], Jensen examines the results from several previous researchers on determination of the dimensions of timbre. He concludes that no clear consensus has emerged, though the most common dimensions seem to be spectral envelope associated with brightness and the resonances of the sounds, temporal/amplitude envelope associated with attack and delay time of the sounds, and irregularities, which are acoustic properties that cannot be uniquely associated with either of the two previous dimensions. The irregularities are divided into periodicity and non-periodic noises. The noise on the amplitude and frequency envelope is, as defined by Jensen, shimmer and jitter, respectively.

Timbre modeling consists in designing synthesis methods to generate sounds under perceptual constraints, reconstructing a given natural sound using algorithmic techniques. Due to the fact that both the spectral and the amplitude envelopes evolution over time play such an important role, many have approached timbre modeling with some sort of time-frequency analysis method [Kronland-Martinet *et al.* 2002]. Timbre model synthesis techniques can therefore be realized by all types of synthesis techniques, though each technique having its advantages and disadvantages in fitting the perceptual parameters to the synthesis parameters. This is an extremely exciting field, though out of the scope of the project.

3.2.4 Auditory Scene Analysis

It has been suggested that it is useful to make a distinction between two concepts: *source* and *stream* [Bregmann 1990]. A source is some physical item which gives rise to acoustic pressure waves. An auditory stream, on the other hand, is the percept of a group of successive and/or simultaneous sound elements as a coherent whole, appearing to emanate from a single source. It is hardly ever the case that the sound reaching our ears comes from a single source, though generally we appear to have little difficulty in hearing out individual sources, e.g. listening to the melody of one instrument in a piece of music or a person talking at a cocktail party. The process of assigning multiple sources their own corresponding distinct streams is often called *perceptual grouping*, *parsing* or *auditory scene analysis*. The process of separating the elements arising from two different sources is sometimes called *audio stream segregation* [Moore 2003]. In short it can be said that, auditory scene analysis aims at understanding the process of decoding the auditory scene into separate auditory streams.

It has been suggested that the grouping of auditory streams depends on the focus of attention. In [Hermann 2002] a distinction is made between analytic and synthetic listening. *Analytical perception* aims at focusing on the maximal information of one stream, e.g. following the voice of one instrument in a band. *Synthetic perception* (also known as holistic perception) aims at perceiving the auditory scene as a whole, e.g. following the piece of music instead of one instrument alone.

Cues such as fundamental frequency, onset, change detection, correlated changes in amplitude or frequency (e.g. rhythm), and sound location are important in assigning sound components to their appropriate sources. *Gestalt* psychology, a theory of psychology that emphasizes the importance of configurational properties, identifies features that promote the binding of signal parts together. Gestalt principles like similarity, good continuation, common fate, disjoint allocation, and closure have been investigated mainly for the purpose of vision research, though these principles can also be carried over to the auditory domain [Moore 2003]. The most important gestalt principles in the auditory domain are:

- ***Similarity***. Components are perceived as related if they share the same attributes, usually implying closeness of timbre, pitch, loudness, or subjective location.
- ***Good continuation***. This principle exploits a physical property of sound sources, that changes in frequency, intensity, location or spectrum tend to be smooth and continuous, rather than suddenly. Hence a smooth change in any aspects indicates a change within a single source, whereas an abrupt change indicates that a new source has been activated.

- ***Common fate***. If two or more components in a complex sound undergo the same kinds of changes at the same time, then they are grouped and perceived as part of the same source.
- ***Disjoint allocation***. This principle, also known as *belongingness*, is that a single component in a sound can only be assigned to one source at a time. In other words, once a component has been used in the formation of one stream, it cannot be used in the formation of a second stream.
- ***Closure***. Incomplete forms tend to be completed. The perception of virtual pitch is an example: the pitch of the fundamental frequency is perceived in a mixture of overtones even if the fundamental frequency does not exist within the spectrum.

For an in-depth knowledge on these topics please see [Bregmann 1990].

3.3 Conclusion

In this chapter a brief introduction of the human ear was given, together with a presentation of psychoacoustics and some of the most prominent hearing sensations. Finally, a short introduction to the area of auditory scene analysis was given.

As will be made clear later on in the thesis, the understanding of the limits, the non linearities and the perceptual grouping of the auditory system will aid the design of auditory displays. Though, some preliminary conclusions can be made if one has in mind of mapping data to the above mentioned hearing sensations.

The linear mapping of data to frequency or to sound pressure level are not recommended due to the fact that the auditory system perceives these, more or less, logarithmically. Although, no clear consensus has emerged on the dimensions of timbre, amplitude and spectral envelope have been found to be amongst the most prominent. Furthermore, perceptual parameter interactions occur between all the presented sensations and do so in a non linear fashion (lack of orthogonality) mirroring the non linearity of the auditory system. This could pose a problem and result in a blurring of information when this information is presented through the perceptual parameters. This will be made discussed further in the next chapter.

Chapter 4

4 Sonification

This chapter will give an introduction to the field of sonification. In the following, the most accepted definitions will be presented and briefly discussed, subsequently; a presentation of the broad field of research involved in the process of sonification is reviewed to give an idea of the interdisciplinary nature of this area. A brief history of the field is presented to give an idea of how sound has been used to convey information in the past and which communities at this moment are active in the research. This leads to a more in-depth, though introductory, section that lists the already present application fields of sonification.

The most accepted sonification techniques are presented and discussed to give an idea of the possible methods of realizing sonifications and how to use the different realization methods most effectively. Finally, some issues in designing sonifications are presented. The main focus in this section is on using perceptual knowledge, considerations of the data, knowledge of the task at hand, and the evaluating the usability when using sonifications.

4.1 Definition

The most accepted definition is given [Kramer *et al.*, 1999] states that, "Sonification is defined as the use of non-speech audio to convey information." More specifically sonification is defined as, "the transformation of data relations into perceived relations in an acoustic signal for the purposes of facilitating communication or interpretation." Sonification is also referred to as auditory display.

The first part of the definition restricts sonification to the use of non-speech sound to discriminate it from speech interfaces. However, speech can provide explanations in auditory displays without changing the media, and furthermore, data-driven use of speech-like sounds should also be called sonifications [Hermann 2002]. The second, and more general definition, emphasizes the purpose of sonification: the communication or interpretation of data in any given domain of study.

In [Hermann 2002], Hermann summarizes the requirements for a sound to be called a sonification as:

- the sound is synthesized depending upon the data of the domain under study, and
- the intention for generating the sound is to learn something about the data by listening to it. The sound is only regarded as the medium of communication.

4.2 Research Field of Sonification

In most introductions to sonification it is stressed that the research field is very interdisciplinary. Several research disciplines contribute to the implementation and understanding of the involved processes. Figure 26 shows a typical information flow in a sonification system.

For the development of a sonification system, it must first be understood what data are available and what the measurements mean. In the case of sonifying high-dimensional data, statistics and data mining, in cooperation with domain expertise, contribute techniques for an intelligent data preprocessing, e.g. for feature extraction and modeling. The discipline of human computer interaction (HCI) is concerned with many aspects of sonification systems. HCI topics include design guidelines for tools, human information processing, ergonomics, system design and usability. HCI contributes valuable insights into how such topics may be analyzed and evaluated. Computer science contributes to the realization of a sonification system in different aspects: software engineering copes with how to program the interface and how to implement the rendering of the sonifications from the data, signal processing provides techniques to manipulate sound signals. The field of acoustics and physics are an important part in sound generation. Examining the physics of sound generating processes can be inspiring for the selection of sound synthesis techniques to represent data, e.g. physical modeling synthesis. The transformation from the digital representation of a sonification to sound waves in air is preformed by soundcards or synthesizers, amplifiers and, loudspeakers or headphones. Depending on the needs, solutions from a mono loudspeaker system to complex multi-speaker arrays or binaural implementations for high-resolution spatialization of the sound are used. Sound engineering is concerned with the technical realization and the sound signal changes due to reflections in the listening room. The disciplines of physiology and neurobiology are concerned with the processing of the sound signal after it reaches the ear (biological perspectives of signal processing). Psychology, psychoacoustics and auditory perception are concerned with higher-level perceptual processing which take place in the auditory brain. Guidelines like the auditory scene analysis, briefly described in chapter 3 section 3.2.3, provide guidelines for the usage of sound in sonifications. Musicology contributes to understanding different aspects of sound: it provides a framework to organize acoustic material concerning its rhythm, measure, harmony and it delivers tools for documentation (e.g. a score) and analysis of musical pieces. Finally cognition focuses on various aspects of the listener like acoustic memory, processing speed for auditory signals and the coupling of sound and emotional states.

In [Kramer *et al.* 1999], Kramer divides all of these areas into three main sonification components; perception, research and development tools (i.e. sound hardware and software), and sonification design and application.

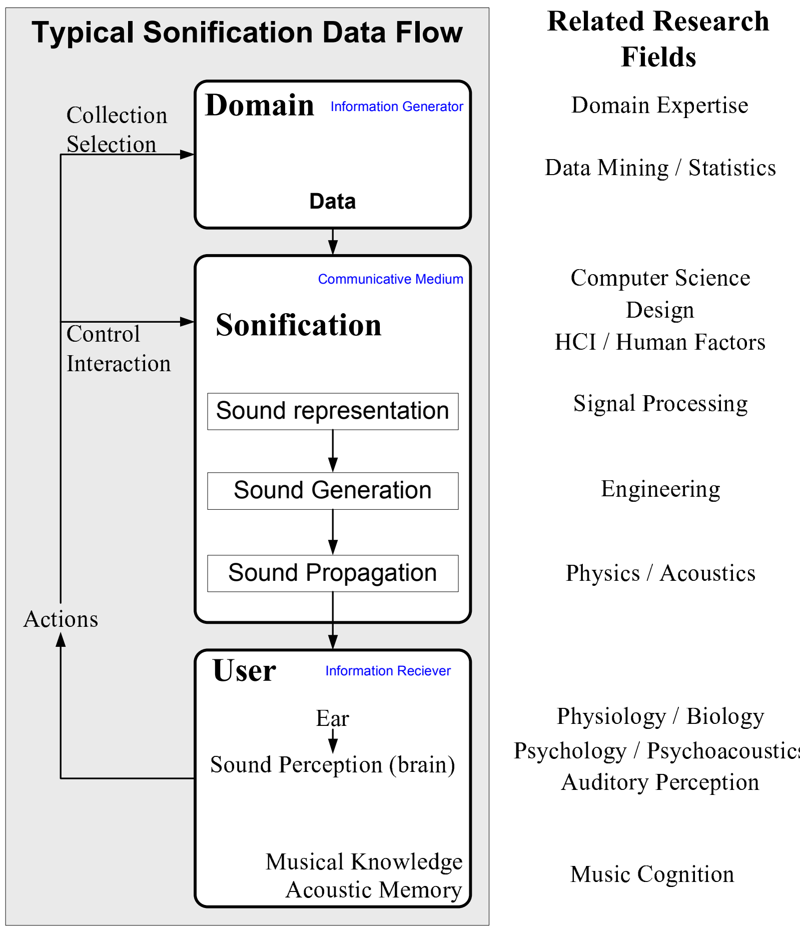


Figure 26. A Schematic illustration of the information flow in a typical auditory display system. Related research disciplines are indicated on the right side. This figure is redrawn from [Hermann 2002].

4.3 A Brief History of Sonification

Even though sonification is a somewhat young discipline, there are plenty examples throughout history of sound being used to convey information. Some examples of these are: sonar, Morse code, the telephone bell, alarms or sirens, pulsoximeter, Geiger counter, auditory thermometer, and metal detectors [Hermann 2002], [Kramer *et al.* 1999], [Barrass and Kramer 1999]. Although this list is not extensive, it shows that humans frequently use sound to convey information and understand the world around them, and most of the times doing this effortlessly. In many domains *audition* is used to gain insight to a system. Car mechanics listen to the sound of an automobile engine in order to draw conclusions about causes of malfunction. Staying in the same groove, physicians apply their stethoscope to diagnose disease from *auscultation* (medical listening) to sounds of lungs, the heart and other parts of the body. These applications show that humans are very capable of learning to interpret sounds and to use acoustic clues.

The first recognized efforts in sonification were made in 1954 by Pollack and Ficks, who investigated the usage of abstract auditory variables (i.e. perceptual parameters) for the presentation of quantative information. Their display consisted of alternating tone and noise bursts, using, amongst others, attributes like loudness, pitch, relative duration, total duration, and stereo location. An important contribution in integrating audio signals with general computer interfaces was Gaver's SonicFinder in 1985 for the Apple Macintosh. In [Kramer 1994] there is an extensive review of the early research results in the field of auditory display and sonification.

In 1992, the International Community of Auditory Display (ICAD) was established, and as a by-product of this meeting, the book on auditory display [Kramer 1994] was published, which is until now an important and much cited book. Since the founding of ICAD, auditory display research has grown steadily. However, due to the fact that sonification is interdisciplinary in nature, ICAD is by far the only community concerned with auditory displays, and communities concerned with perception, psychology, visualization, computer music, HCI, multi-media systems, electrical engineering, and acoustics are also a source of inspiration for related publications. For this reason, it can be time consuming and problematic to find relevant information in the area of sonification.

4.4 Application Fields

Auditory displays find applications in very different contexts. Besides their utility to replace visual displays where visual displays are not possible (blind people) or unavailable (radio programs), they offer an additional information channel to extend visual displays. The main application fields can be listed as:

- ***Alarm Systems***. Alarms are the oldest application of auditory display and are associated with an urgent situation and alarm sounds are designed to stand out in the main acoustic environment. They usually have a strong effect of drawing the listener's attention and eyes towards them, due to the fact that we cannot ignore them. More elaborate forms of alarms, which also represent data, are applied in airplane cockpits or during surgeries (e.g. pulsoximeter).
- ***Auditory display for visually impaired people***. Auditory display may provide location-based information, be the means for inspecting visual scenery, or to convey information about the structure or layout of a document (e.g. while browsing websites on the internet) as non-verbal auditory streams.
- ***Browsing Data***. Any quick browsing of information can be supported by sonification, e.g. traffic data, log files, virtual data spaces of sounds or searching in long-term EEGs. Since auditory perception offers a high temporal resolution, i.e. small detection times, data can be presented temporally very compressed.
- ***Human-computer interaction***. Sonification may be used to support the interaction with devices/computers, e.g. mobile phones. Many operating systems use sound to portray information about actions, like a deletion sound when dragging a symbol to the trashcan symbol. Auditory displays can for instance increase awareness by connecting the sound characteristics with data characteristics (e.g. size of the deleted file).

- ***Virtual reality***. Interactions in the real-world cause sounds. In virtual reality systems, similar impacts between objects occur. Sonification can here increase the immersion effect, and communicate information about the interacting objects, thus augmenting the experience of the visual domain.
- ***Process Monitoring***. Sonifications are useful for monitoring processes because of two reasons: firstly auditory monitoring is eyes-free. The user may therefore perform other tasks at the same time. Secondly, the auditory system can give certain sounds a low priority of attention while maintaining awareness and drawing the listener's attention if these sounds significantly changes. This is also known as *backgrounding* [Kramer 1994]. Listeners can familiarize themselves to a sound pattern (e.g. the engine sound of their own car) but remain attentive to even subtle changes. In addition, the ear is able to merge many acoustic streams to an auditory scenery. This makes the ear a suited instrument for process monitoring applications, from stock market, medical monitors, and power plants to complex robotics systems.
- ***Online-Feedback***. Situations where an auditory feedback is rendered to give a direct feedback on the action, either to analyze or track processes (real-time monitoring), but also to enable an interactive refinement of own actions (e.g. using sonification in rehabilitation, to retain coordinated muscle movements after a stroke). An examples of this form of sonification is given in [Williamson and Murray-Smith 2002].
- ***Exploratory Data Analysis***. Exploratory data analysis is the application of listening to learn about an unknown system. Wherever data is available, sonification may just provide a new view on the data, and may be the factor for detecting the unexpected, for discovering new regularities or features in the data. The ear has been used for such analysis tasks in many scientific contexts, ranging from physics, neurophysiology, and medicine to geology. The high-developed skills of human listeners, e.g. in interpreting even very noisy sounds, or detecting spectral or rhythmical changes in sound make it a promising channel for this application. A successful example, which often is mentioned in this context, is the "Quantum Whistle" and the Voyager 2 problem around Saturn [Kramer *et al.* 1999]. The physicists Davis and Packard attributed an important discovery, which they have called the "Quantum Whistle" to their use of a sonification technique. After months of unsuccessful study of visual oscilloscope traces for evidence of an oscillation predicted by quantum theory, Davis and Packard decided to listen to their experiment. The resulting sound was a faint whistling – the first evidence that these oscillations do actually occur. The second example was during the Voyager space mission where there was a problem with the spacecraft as it began its traversal of the rings of Saturn. The controllers were unable to pinpoint the problem using visual displays, which showed a lot of noise. When the data was played through a music synthesizer, a "machine gun" sound was heard during a critical period, leading to the discovery that the problem was caused by high-speed collisions with electromagnetic charged micrometeoroids.
- ***Educational applications***. It has been suggested that using sonification to present information to students in primary and secondary schools can provide a more engaging learning experience [Kramer 1994]. Rhythm and music are used as a

mnemonic (method of aiding memory) device for teaching young students concepts such as the alphabet. Likewise, it may be possible to harness the underlying components of this learning dynamic to assist students in grasping more sophisticated concepts, e.g. in calculus or statistics. Representing concepts and data through sound provides a means of capitalizing on strengths of individual learning styles, some of which may be more compatible with auditory representations than more traditional verbal and graphical representations.

4.5 Sonification techniques

In this section the existing techniques that map data into an acoustic signal are discussed. A technique for searching a stereo sound scene is presented and two categorization methods of the sonification techniques are presented. The main sonification techniques are:

- ***Audification***
- ***Earcons*** [Blattner *et al.* 1989]
- (Parameterized) ***Auditory Icons*** [Gaver 1994]
- ***Parameter Mapping*** [Kramer 1994]
- ***Model-based Sonification*** [Hermann and Ritter 1999], [Hermann 2002]

In addition to the above mentioned techniques, there also is a branch of sonification that puts a particular focus on sonification systems where the human user is closely integrated into an interactive loop; this is referred to as *interactive sonification*. This form of sonification, can with varying degrees of ease, be integrated into the above listed techniques.

4.5.1 Audification

Audification is the most direct transformation of data values into sound: the sound samples (instantaneous sound pressure levels) are directly obtained from the data values. That means that ordered lists of numbers, e.g. seismic data are directly taken as PCM (Pulse Code Modulation) data for a sound. There are a couple of interesting transformations like re-sampling, time stretching, pitch scaling, dynamic compression, filtering, etc., which allow to adapt the resulting sound better to the preferred frequency range of the ear. There exist domains where audification is very suited, i.e. where the data itself stems from a physical process (e.g. waves propagating through material) [Barras and Kramer 1999]. The advantages of audification are:

1. ***Ease of production***. Any data set can easily be heard by playing them as a standard sound file;
2. ***Compressed information***. Using standard sampling rates can compress 24 hours of very low frequency data into few minutes of sonified data, thus having the potential to save time in the analysis stage.

The disadvantages of audification are:

1. ***Large data sets are needed***. The standard sampling rates, as mentioned above, require large data sets to produce a sound of analyzable duration;
2. ***Limited control.*** There is only limited independent control over temporal and spectral organization. Using the transformations mentioned above require some understanding to manipulate the audification in a constructive way.

4.5.2 Earcons

Earcons were developed to provide feedback about activities in a GUI. They are constructed by combining a lexicon of simple sounds to build more complex meanings, similarly as words can be combined to form phrases. The lexicon may have elements that vary in rhythm, pitch, timbre, register, and dynamics. An example of this was presented in [Blattner *et al.* 1989]. Consider tone "A" with pitch 440 Hz is given the meaning of "file" and, tone "B" with pitch 600 Hz is given the meaning "deleted". Then combining A and B in series produces a rising tone "AB" that means "file deleted". The advantages of earcons are:

1. ***Ease of production***. Earcons can be easily constructed and produced on almost any computer with tools that already exist for music and audio manipulation;
2. ***Abstract representation***. Earcon sounds do not have to correspond to the objects they represent, so objects that either make no or an unpleasant sound can still be represented [Barrass and Kramer 1999].

The disadvantage with earcons is *learnability*. Novices are able to learn up to 7 symbolic sounds within minutes, which can be linked to the limits of our short term memory, but further learning of up to ten symbols can take hours. Beyond ten, the process is prolonged and some listeners may never learn the catalog (of earcons) completely [Patterson 1982].

4.5.3 Auditory Icons

Auditory icons were also originally designed to provide feedback about activities in a graphical user interface. The auditory icon approach is to map objects and events in the interface onto everyday sounds that represent reminiscent or conceptually related objects and events [Gaver 1994]. The meaning of the sound shall be connected to the information by metaphorical association. For example, when dragging a file symbol on the computer desktop to the trashcan symbol, a crushing sound could represent the deletion action. If the sound level or complexity would depend on the file size being deleted, this would be a parameterized auditory icon. Similar to their visual counterparts, auditory icons rely on the analogy between the everyday world and the model world, and the more intuitive the analogy is, the easier the icons are understood. Besides low-level use in computer desktop interaction, auditory icons can be used for interacting with data in exploratory data analysis, e.g. for categorization or classification. The advantages of auditory icons are:

1. ***Familiarity***. Everyday sounds are already familiar and may be understood very quickly;
2. ***Directness***. Everyday sounds can allow direct comparisons of length or size or other quantities [Barrass and Kramer 1999].

The disadvantages of auditory icons are:
1. ***Learnability***. Representing a virtual event, such as a software operation, with a sound from a mechanical event, is a conceptual mapping that may invoke learning demands similar to those of earcons;
2. ***Experience of the listener***. The cultural experience of the listener may have significant effects on the recognition of recorded everyday sounds;
3. ***Shortage of compelling sonic representations***. Limited everyday sounds available that can be used to give the listener an intuitive idea of the information being conveyed.

4.5.4 Parameter Mapping

Parameter mapping is the widest used sonification technique for representing high-dimensional data as sound. Typically, a data dimension is mapped onto an auditory parameter such as onset, duration, pitch, pitch variation, loudness, position (spatial cues), reverberation, brightness, etc. Different data variables can be mapped to different auditory parameters at the same time to produce a complex sound. For this reason, high-dimensional data displays can be obtained. To formalize the parameter mapping let there be given a d-dimensional data point $\mathbf{x} = (x_1, \ldots, x_d)^T$. Simple parameter mappings, map a single data variable x_j to values of an acoustic attribute p_i. Such a mapping can be written as

$$p_i = h_i(x_i), \qquad i \leq d \qquad 4.1$$

The functions $h_i()$ provide a mapping of data values to attribute values. Usually monotonous functions or constant values are used. Figure 27 shows some frequently applied mapping functions.

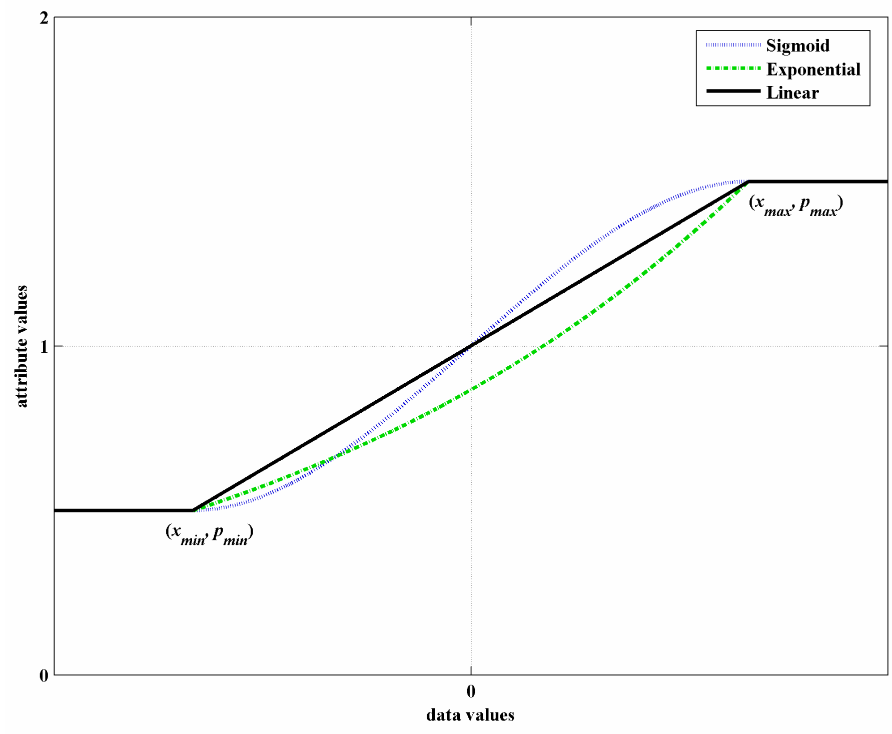


Figure 27 Typical transfer functions for parameter mapping. The piecewise linear transfer function (black line) is described by equation 3.3. The blue and green dashed lines are respectively sigmoid and exponential transfer functions. This figure is modified and expanded from [Hermann 2002].

The linear mapping with a clipping to min/max values in the attribute domain is very commonly used, and Hermann uses the following notation to clarify it:

$$p(x) = \mathrm{map}(x, [x_{min}, x_{max}], [p_{min}, p_{max}]) \qquad 4.2$$

$$= \begin{cases} p_{min} : x \le x_{min} \\ p_{min} + \dfrac{p_{max} - p_{min}}{x_{max} - x_{min}}(x - x_{min}) : x_{min} < x < x_{max} \\ p_{min} : x \ge x_{max} \end{cases} \qquad 4.3$$

The parameter mapping sonification technique is also sometimes referred to as sonic scatter plots or *n*th order parameter mapping. This technique has the following advantages:

1. ***Ease of production***. Existing tools (instrument sounds synthesized by efficient algorithms) allow almost real-time mappings to many auditory parameters;
2. ***Flexible multivariate representations***. Many data dimensions can be listened to simultaneously and the mapping choices can be changed using the same data giving different views of the same data.

The disadvantages of the parameter mapping approach are the following:

1. ***Unpleasantness of produced sounds***. The sounds that are produced using this method can become unpleasant. For example, if a data dimension is mapped to loudness and the data dimension encounters unexpected large values, the resulting sonification can become unpleasantly loud;
2. ***Linear changes in one domain produce non-linear effects in the auditory domain***. Linear changes in multivariate synthesis parameters can have complex, non-linear perceptual effects, and the range of the variation can differ considerably with different parameters and synthesis techniques. These perceptual interactions (coupled perceptual parameters) between parameters can obscure data relations and confuse the listener, and a truly balanced multivariate auditory display may not be possible in practice [Kramer 1994], due to the fact reason of lack of orthogonality, as mentioned in chapter 3.
3. ***No Unique Mapping.*** There is no unique mapping from data to acoustic attributes, and therefore manual assignment is necessary making this a heuristically governed approach.
4. ***Interpretability.*** Each mapping choice sounds different using the same data, which makes learning and adapting to these sonifications difficult, though this can also be viewed as the flexibility of this technique.

4.5.5 Model-based Sonification

Model-based sonification has been proposed as an alternative framework for computing data driven sound in [Hermann and Ritter 1999], [Hermann 2002]. The starting point is that sound in the real-world is the by-product of physical processes and the complex sound field encodes in a holistic way source-properties in its temporal evolution. However, since the extraction of source-related information from the sound has been of high importance in the real-world (e.g. to recognize arriving predators early) evolution has lead to an optimization of these sound processing skills, including the processing hardware, the brain. In addition, one often learns about the world by interacting with the world and interpreting the acoustic feedback (think about shaking a present at Christmas). To carry these concepts over to the domain of data exploration, a sonification model defines a kind of "virtual acoustic object", whose setup might be driven by the dataset under analysis. Laws of dynamics (corresponding to the laws of physics in the real world) determine the temporal evolution of a sonification model. The advantages of model-based sonifications are:

1. ***Source-related information***. It is argued that by evolution the human auditory system is optimized to extract source-related information from sounds that result from a dynamic process. The sound of dynamic processes differ for different interactions, but the sound shares the same typical properties;
2. ***Dissipative process***. Model-based sonifications are typically a dissipative process, converging to a state of equilibrium, corresponding to silence.

The disadvantages of model-based sonifications are:

1. ***Model selection***. It is unclear what the best models are for certain analysis tasks at hand.
2. ***Many parameters in physical models.*** When designing physical models many parameters are available and no generic transformation from the data dimensions to the physical dimension, equivalent to the problems faced with the parameter mapping technique.
3. ***Computationally expensive.*** Model-based synthesis may be computationally more expensive than parameter mappings.
4. ***Boundary conditions or physical limits.*** The model, that is used to generate the sound, may contain critical borders that yield unexpected large changes of the sound even on smooth changes in the dataset.

4.5.6 Interactive Sonification

Different from offline sonifications, which are rendered without any interaction by the user and then consumed by uninterrupted listening (like to a piece of music), interactive sonification considers settings where the sonification is directly controlled (e.g. navigated, manipulated or excited) by the user. Interaction provides user-centered views on objects – like several visual views support understanding of visual objects (e.g. 3D-shape), interaction with acoustic systems supports understanding the sound source, which is in the case of sonification the underlying data or generation process that involves the data. An interesting example of interactive sonification can be found in [Williamson and Murray-Smith 2002]. Here a general framework for formative audio feedback for gesture recognition is presented. A disadvantage at this moment is how to compare and evaluate displays that rely so heavily on interactive exploration processes [Hermann 2002].

The author believes that sonification toolboxes should to some degree be interactive to allow the user to tune parameters that can help improve the clarity of his or her display. This can be compared to changing the axes on a plot, changing the color or structure (dashed/dotted) of a line to enhance the information one is trying to present.

4.5.7 Techniques for Spatialized Sonifications

An interesting technique, which is worth mentioning, is a technique for searching or browsing through an auditory scene, be that stereo or 3D. The *Aura* technique was presented in [Benford and Greenhalgh 1997] and later used as a component for sonic browsing in [Fernström and McNamara 1998]. In the former article the aura is described as a device through which users perceive the world and takes the form of a scope or area of interest. The idea behind this, as stated by Fernström and McNamara, was to exploit our ability to single out sounds in a sometimes sonically dense environment, i.e. the *cocktail party problem*. Browsing through sounds of interest that are either panned out in a stereo field or located in 3D space with the help of the aura acting as a magnifying glass cursor in the auditory domain, one is able to focus ones attention fully on a subset of the original sound field. The aura can be user controlled and can be adjusted in size. By increasing the aura one expands the area of listening and by decreasing the size one zooms in on the sound(s) of interest.

4.5.8 Categorization of Auditory Display Techniques

In this section the two main methods of classifying auditory display techniques are presented. This was found important to include, due to the fact that the different perspectives on the techniques can expand the understanding of sonifications and further clarify which sonification technique best suits a given tasks. The two categorization techniques presented here are:

- The Semiotic Categorization
- The Analogic-Symbolic continuum

Semiotics is the theory of signs and their meaning and it can be used to analyze communication media. The best known auditory display techniques were classified using the semiotic categorization by Blattner *et al.* in 1994. The semiotic distinctions are: *syntactic*, *semantic*, and *lexical*. Earcons focus on syntactic organization of acoustic material to communicate messages. The sounds are symbols to the signified, i.e. receiver of the signs. Auditory icons are an example of the semantic approach, i.e. the meaning is associated to the sound by metaphorical or iconic association. Parameter mapping is a lexical approach, i.e. the signs are created from the data [Barrass and Kramer 1999].

In [Kramer 1994] the ideas of A. Sloman about analogical representations are related to auditory display. A *symbolic* representation is a categorical representation of what is being represented. The information being represented is clustered in categories and the relationships between the representations do not reflect intrinsic relationships between the elements being represented. For example, words are typical examples for symbolic representations. In an *analogical* representation an immediate and intrinsic correspondence between the represented item and the representation is given, i.e. changes in the represented item map to similar changes in the representation, even though the representation can be a simplification of the represented item. A typical example, is a thermometer, i.e. the height of the thermometer column analogically represents the temperature.

In Figure 28 the main sonification techniques are presented along the analogic-symbolic continuum. In [Hermann 2002] it is stated, that the model-based sonification technique is difficult to locate on this scale, due to the fact that model dynamics associate sounds to a dataset, thus the model may contain critical borders which yield to large changes of the sound even on smooth changes in the dataset, as mentioned earlier.

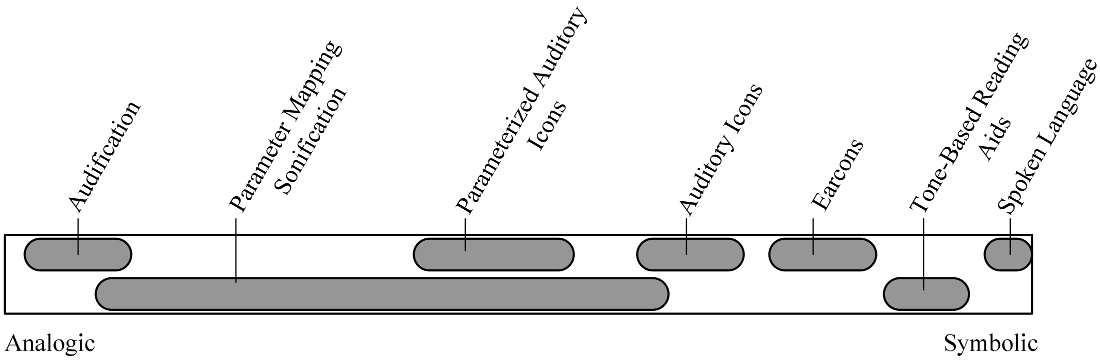


Figure 28 The Analogic-Symbolic continuum. This figure is redrawn from [Hermann 2002].

The presented categorization techniques are by no means the only forms of categorization techniques present. One could choose to classify the sonification techniques by applied sound synthesis techniques, data domain, application etc.

4.6 Issues in Designing Sonification

The range of sonification techniques with different advantages and disadvantages lead to the question of which to choose, and there is no known method for determining the best way to map data relations into sound, though some general guidelines have been obtained.

Knowledge of auditory perception can allow the designer to predict how the sonification will be heard by a human listener, and enables a theoretical evaluation of new untried designs. However, psychoacoustic theories do not involve issues of representation that are central in sonification, where the listener needs to correctly understand data relations from the sounds, though some experiments have shown possible perceptual observations that should be taken into consideration.

In [Flowers 2005] a reflection is made on the past thirteen years of the history of auditory graphing, and mentions some of the strategies that work well and some that do not. He states that, using loudness changes to represent an important continuous variable can have its pitfalls. He writes, “Even with isolated presentation of a single auditory stream of constant pitch, ability to discriminate different loudness levels for reliable mapping to numeric values is far more limited than for pitch (log frequency) mapping to quantity, or temporal auditory changes such as modulation rate, pulse or note rate, etc. In addition, there is a major non-perceptual factor that makes loudness unsuitable for carrying fine-grained quantitative information – limitations of sound reproduction equipment, and differences in the dynamic ranges and general quality of such equipment from setting to setting.” Though he writes that, “Temporal or rhythmic patterning of loudness levels, especially when integrated into pitch and timbre defined data streams may be highly useful” to provide contextual cues and signal critical events. In [Neuhoff *et al.* 1999], they found that when people listened to a change in loudness with a rising pitch, they perceived the change to be greater than when they heard the same degree of change in loudness with falling pitch.

In [Walker and Ehrenstein 1997], it was investigated whether it is possible to attend to *relative pitch* while ignoring changes in pitch, and whether changes in pitch could be assessed independently of the overall pitch of a dynamic auditory stimulus. They define their stimuli as:

- ***Congruent***. High pitch stimulus that became higher in pitch, or low pitch that became lower;
- ***Incongruent***. High pitch stimulus that became lower in pitch or high pitch that became higher.

Walker and Ehrenstien conclude that responses were almost always faster when the direction of pitch change matched the onset pitch position, i.e. for congruent stimuli. Similar effects in vision are also observed; the dimension of position influences the perception of direction and vice versa. Furthermore, the intrusion of the information of one dimension onto judgments regarding the other dimension was not symmetrical, in that pitch information had a greater influence on responses to direction of pitch change than direction information had on pitch judgments. The selective attention between pitch and pitch change was found not to be perfect. Though, it is suggested that auditory display designers should take advantage of this *congruency effect* when a crucial distinction must made between high and low pitches.

In [Flowers 2005], Flowers writes that, "Pitch profiles are a compelling dimension for representing changes in numeric values. Mapping pitch height (essentially log frequency) to numeric magnitude affords perception of function shape or data profile changes, even for relatively untrained observers." Though he writes that, "Listening to simultaneously plotted multiple continuous pitch mapped data streams, even when attention is given to timbre choice for different variables to reduce unwanted grouping, is probably not productive… it is generally the case that attending to three or more continuous streams of sonified data is extremely difficult." This can be compared to listening to and understanding three conversations simultaneously, which is for the author an impossible task.

As suggested, timbre differences can be useful for minimizing unwanted perceptual grouping of separate continuous data streams when multiple continuous variables are required to be plotted. Flowers further writes that, "Timbre changes due to onset envelope differences in note streams probably allow better separation than timbre differences due to harmonic content per se." He also highlights that avoiding confusions between simultaneous data events and streams is important, and states that there is little basic psychoacoustic research that directly relates to the attention and perceptual demands of listening to auditory mappings of data, even though auditory scene analysis describes the basic concepts perceptual organization, more empirical research needs to be conducted.

The need to *consider the data structure* in mapping data relations into auditory relations is found in Chris Hayward's description of why audification techniques work well for seismic data [Barrass and Kramer 1999]. He explains that seismic data consists of large data sets and that a seismic audification will sound like a recording of natural environmental sounds, because sounds transmitted through air (acoustic waves) have similar physics to seismic vibrations transmitted through the earth (elastic waves). The direct physically consistent playback can take advantage of human experience with natural sounds. It has been stated, that an arbitrary mapping from data to sound parameters often results in unpleasant sounds that lack any natural connection to the data represented. Gary Kendall, as written in [Barrass and Kramer 1999], proposed an approach that links the structure of the data with the structure of heard sounds. He observed that categorical data relations should sound categorical, and ordered data relations should sound ordered. He also states that, "Relevant changes in data should insure a change in what is perceived. Changes in what is perceived should signify meaningful changes in the data."

In some instances the "relevant changes in data" are unknown; therefore Kramer suggests that two broad types of tasks are important in auditory display [Kramer 1994 p.15]:

1. ***Analysis***. Tasks where the user cannot anticipate what will be heard and is listening for "pop-out" effects, patterns, similarities and anomalies which indicate structural features and interesting relationships in the data.
2. ***Monitoring***. A "listening search" for familiar patterns in a limited and unambiguous set of sounds.

The acceptance of a sonification may be influenced by the quality of the audio output, just as the perceived quality of television set was influenced by the quality by the audio. After having encountered obnoxious sounds, ambiguous meanings, negative connotations, and incomprehensibility in sounds used for background notification in operating systems, Jonathan Cohen strongly suggested that an experienced sound designer should be involved in any such project [Barrass and Kramer 1999]. With the expanding world of sound synthesis algorithms and control schemes, it should not be a major task to provide easy-to-use tools and systems that allow non-experts to make their own sonifications tailored to their particular task. However, there is still a surprising gap when it comes to a practical sonification toolbox. This is seen as a major obstacle for sonification.

The evaluation and validation of auditory stimuli for experimental or application use is an important component to the successful completion of a project utilizing sound. The choice of methods to test the perceptual properties of auditory stimuli depends on the goals of the specific system [Bonebright *et al.* 1998]. Bonebright *et al.* provide a general framework for data collection and analysis techniques appropriate for evaluating the perceptual properties of auditory stimuli. They present guidelines for subject selection, sample size, number of stimuli, pilot testing, number and type of practical trials, duration of data collection sessions, and examples of computer software that can be used to automate data collection procedures. The three main methods that can be used for determining the *perceptual qualities of single auditory events* are discussed; *identification tasks*, *context-based ratings* and *attribute ratings*.

Important for many sonification projects, is to examine the *associations among auditory events*. In [Bonebright *et al.* 1998], they recommend three techniques; *discrimination trials*, *similarity ratings* and *sorting tasks*. For most applications using multiple audio signals, it is important to determine if the auditory stimuli are distinguishable from one another and to measure the extent to which subjects can discriminate among the stimuli. This can be accomplished with a simple discrimination task. The similarity (or dissimilarity) rating method is a common data collection method in perceptual studies and provides a means for examining the perceptual structure of a set of stimuli without imposing experimenter bias. This type of information can be useful in understanding how, and perhaps even why, subjects confuse stimuli. Sorting tasks are another method for collecting similarity data that provides information about perceptual relations among stimuli.

According to Barrass and Kramer the major issues in sonification raised by experienced researchers in the field, which can be summarized as follows [Barrass and Kramer 1999]:

1. ***Veridicality***. The need to ensure that relations in the data can be heard correctly and confidently in the sounds,
2. ***Usefulness***. The effect that a sonification has on a task
3. ***Usability***. The amount of usage required before a sonification becomes useful,
4. ***Acceptance***. How much a sonification is actually used in practice
5. ***Tools***. Support for sonification by people who are not necessarily experts.

4.7 Conclusion

In this chapter the definition of a sonification was presented and discussed, the interdisciplinary nature of the auditory display was highlighted, and a very brief history of the field was presented. Furthermore, an overview of the applications that sonification can be incorporated into was given, a discussion of the main sonification techniques and their categorization. A selection of the issues one must have in mind when designing sonifications and issues that still need to be addressed were also presented.

It was made clear that designing an auditory display is an interdisciplinary affair, which needs to take many considerations into account, especially the knowledge of auditory perception and the specific task at hand (relating to the data structure). Translating the data relations into sound is made more difficult due to the fact of the lack of orthogonality, i.e. the interactions of the perceptual parameters. Experienced researchers in the field of sonification still see the need for conducting empirical research in auditory perception that is closer linked to sonification. Testing the sonifications is a crucial source of information that provides knowledge of the usefulness of the resulting auditory display, and this knowledge, together with a broadly accepted toolbox, is crucial for further expanding the use of auditory displays.

Chapter 5

5 Synthesis Techniques

In this chapter a broad range of synthesis techniques will briefly be presented. Synthesis techniques are very relevant in sonification procedures due to the fact that they, together with mapping functions, translate the data into sound. The information used to write this section is mainly taken from [Roads 1996] from chapters 4 to 7. The synthesis techniques listed below are an extract of the most commonly known techniques:

- Waveform Table-lookup synthesis
- Sampling synthesis
- Additive synthesis
- Multiple Wavetable synthesis
- Wave Terrain synthesis
- Granular synthesis
- Subtractive synthesis
- Modulation synthesis
- Physical modeling, Karplus-Strong, and Formant synthesis
- Sound Spatialization

The above techniques are listed in the order in which they will be presented in current chapter. Furthermore, the description of granular synthesis will be more thorough due to the fact that it will be used later in the thesis. It will also become apparent to the reader why this technique is interesting to use in classification or categorization sonifications. For a more in-depth coverage of these and other sound synthesis and sound transformation techniques please see [Roads 1996].

5.1 Waveform Table-lookup Synthesis

The process of repeatedly scanning a wave-table in memory is called table-lookup synthesis, and is the core operation of a *digital oscillator*, which is a fundamental sound generator in most synthesizers. A digital oscillator can be controlled by many time-varying parameters causing it to undergo temporal and spectral change. An array of such digital oscillators can be combined to produce what is called additive synthesizer, which will be described in section 5.3.

5.2 Sampling Synthesis

Sampling synthesis is different from the classical technique of waveform synthesis briefly described above. Instead of scanning a small fixed wavetable containing one cycle of a waveform, a sampling system scans a large wavetable that contains thousands of individual cycles, i.e. several seconds of prerecorded sound. Since the sampled waveform changes over the attack, sustain, and decay portion of the event, the result is a rich and time-varying sound. All sampling instruments are designed around the basic notion of playing back prerecorded sounds, shifted to the desired pitch. Despite advances in sampling technology, samplers retain a "mechanistic" sound quality that makes them distinguishable from the animated sounds produced by good human performers. Furthermore, theses techniques require large memory capabilities since each sound (recorded instrument in the synthesizer) requires a large wavetable.

5.3 Additive synthesis

Additive synthesis is one of the oldest and most heavily researched synthesis techniques and is a class of sound synthesis techniques based on the summation of elementary waveforms, such as sinusoidal, triangular or rectangular waveforms, to create a more complex waveform. The sound is computed by;

$$s(t) = \sum_i a_i(t) \cdot \sin(2\pi f_i t + \varphi_i) \qquad 5.1$$

Sine waves with frequency f_i and corresponding phase φ_i are multiplied by an amplitude envelope $a_i(t)$ and superimposed and stored into $s(t)$. The spectrum of many periodic functions, which many sounds can approximated to, consists of frequencies with are integer multiples of a fundamental frequency. Using 5.1 these sounds can be computed as;

$$s(t) = a(t) \sum_i a_i \cdot \sin(2\pi i f_0 t + \varphi_i) \qquad 5.2$$

Additive synthesis models can be further expanded for more flexible control over the sound and have been used for timbre modeling [Jensen 2001], [Marentakis and Jensen 2002] to include irregularities, such as discussed in section 3.2.3. However, when mimicking realistic sounds complicated trajectories in parameter space have to be conducted and noisy and transient sounds are especially hard to model using this model [Hermann 2002]. In Figure 29 an illustration of a simple digital additive synthesis with two time-varying parameters: these are frequency (F) and amplitude (A) envelopes.

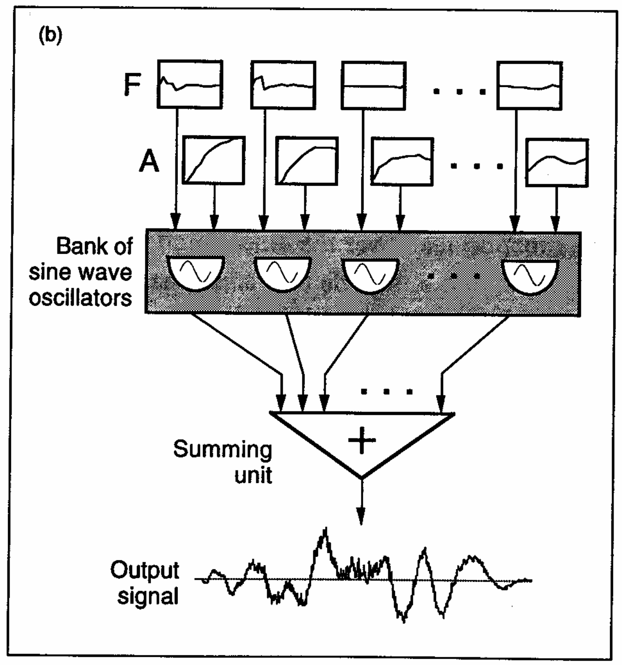


Figure 29. An illustration of time-varying additive synthesis with separate frequency (F) and amplitude (A) envelopes. This figure is taken from [Roads 1996].

5.4 Multiple Wavetable synthesis

By *multiple wavetable synthesis*, one refers to two simple yet sonically effective methods: *wavetable crossfading* and *wavestacking*, though these are not the only synthesis methods that can use multiple wavetables.

Instead of scanning a single wavetable repeatedly, the oscillator crossfades between two or more wavetables over the course of an event, i.e. the event begins with waveform 1, and as 1 begins to fade away, waveform 2 fades in, and so on. The crossfading procedure is shown in Figure 30. Wavetable crossfading is the core of what has been called variously *compound synthesis*, *vector synthesis* (by Sequential Circuits, Korg, and Yamaha), and *L/A* or *Linear Arithmetic* synthesis (Roland).

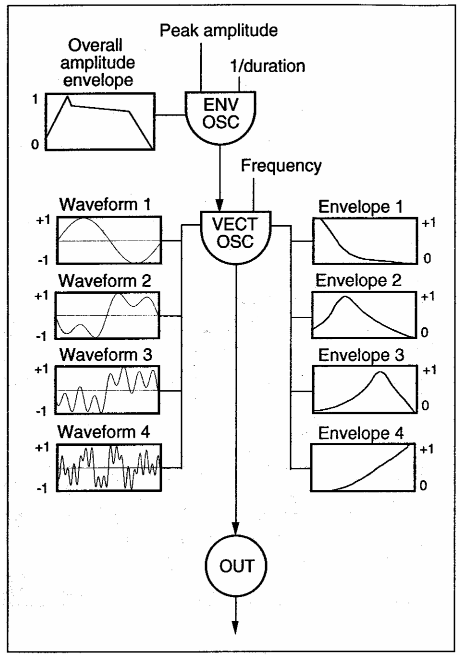


Figure 30. Wavetable crossfading (vector synthesis) instrument using four wavetables. Each envelope on the right applies to a wavetabel on the left. This figure is taken from [Roads 1996].

5.5 Wave Terrain Synthesis

As mentioned, many synthesis techniques start from the fundamental principle of wavetable lookup: a wavetable is scanned by means of an index that is incremented at each sample period. It is possible to extend the principle of wavetable lookup to the scanning of three-dimensional "wave surfaces". Such surfaces are called *wave terrain.* Compared to the traditional wavetable which is a function of only one variable, wave terrains can be plotted as a function of two variables as shown in Figure 31. A scan over the terrain is called an *orbit*, and is naturally a function of x and y, to use the axis of Figure 31.

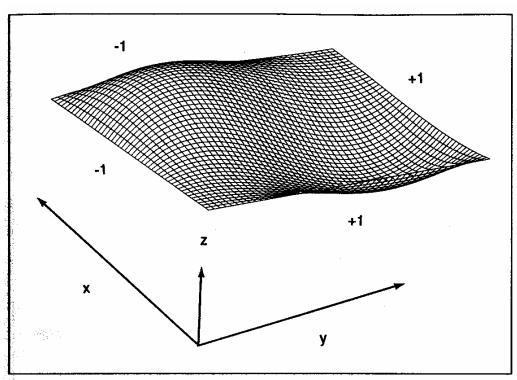


Figure 31. The waveform terrain is a three-dimensional surface. The height (z-axis) of the terrain represents the waveform value. This figure is taken from [Roads 1996].

5.6 Granular synthesis

Granular synthesis (see [Truax 1988], [Roads 1996], [Childs 2002]) is a probabilistic sound generation method, based on drawing many short packets of sound called *grains* or *granules*, from source waveforms. A sound grain lasts a brief moment (typically 1 to 100 ms), which approaches the minimum perceivable event time for duration, frequency, and amplitude discrimination [Roads 1996].

An amplitude envelope shapes each grain. This envelope can vary in different implementations from a Gaussian bell-shaped curve to a simple three-stage line-segment attack/sustain/decay, each of which creating sonically different sounds. The grain duration can be constant, random, or it can vary in a frequency-dependent way. The waveform within the grain can be of two types: *synthetic* or *sampled*. Synthetic waveforms are typically sums of sinusoids scanned at a specified frequency. For sampled grains, one typically reads the waveform from a predetermined location in a stored sound file, with or without pitch-shifting. Several parameters can be varied on a grain-by-grain basis, including the duration, envelope, frequency, location in a sound file (for sampled grains), spatial location, and waveform (a wavetable for synthetic grains, or a file name or input channel for sampled grains). A simple granular synthesis process is shown in Figure 32. The resulting sound signal can be written as;

$$s(t) = \sum_i a_i g(t - t_i, \theta_i) \qquad 5.3$$

where g is the time-domain representation of a grain, whose envelope shape may be a function of parameters θ.

Despite the simplicity of the instrument, to generate even a plain, uncomplicated sound requires a massive amount of control data. These parameters describe each grain: starting time, amplitude, etc. The complexity of the sound generated by granular synthesis derives from the amount of control data fed to it. If n is the number of parameters for each grain, and d is the average grain density per second of sound, it takes $d \times n$ parameter values to specify one second. Since d typically varies between a few dozen and several thousand, it is clear that for the purposes of compositional control, a higher-level unit of organization for the grains is needed. The purpose of such a unit is to let composers specify large quantities of grains using just a few global parameters. Existing granular synthesis methods can be classified into five types, according to the organization of the grains:

- Fourier and wavelet grids
- Pitch-synchronous overlapping streams
- Quasi-synchronous streams
- Asynchronous clouds
- Time-granulated or sampled-sound stream, with overlapped, quasi-synchronous, or asynchronous playback.

As mentioned, the asynchronous granular synthesis has been implemented in this project and therefore is the only organization type that will be described.

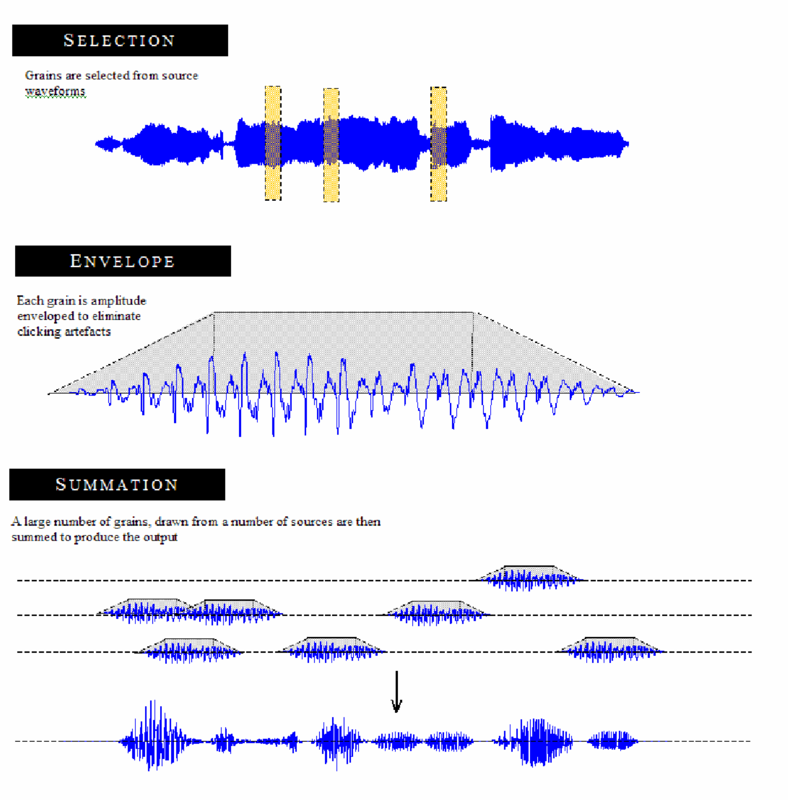


Figure 32. Simple granular synthesis process. A much greater number og grains would be used in real output for a smoother waveform. When a new grain is created, a section of the waveform is copied. The position of the section is determined by the temporal distribution across the waveform. This section is the envelope. All of the currently active grains are summed to produce the final output. This figure is taken from [Williamson and Murray-Smith 2004].

5.6.1 Asynchronous Granular Synthesis

The *asynchronous granular synthesis* (AGS) scatters grains in a statistical manner over a specified duration within regions inscribed on the frequency-time space. These regions are called *clouds* – the units which the composer works with. The composer specifies a cloud in terms of the following parameters:

- Start time and duration of cloud
- Grain duration or grain duration range
- Density of grains per second or by time frame
- Bandwidth of the cloud (only for synthetic waveforms)
- Amplitude envelope of the cloud
- Waveform(s) within the grains (only for synthetic waveforms)
- Spatial dispersion of the grains in the cloud

By varying these seven parameters of AGS one can realize a wide range of effects. For example, short grain durations lead to crackling, explosive sonorities, while longer durations create a much smoother impression. Sparse grain densities create pointillistic textures, while high grain densities create more massive blocks of sounds. The spatial algorithm of a cloud can involve random scattering or panning effects over the duration of the cloud, and enhances granular texture.

An analogy exists between the AGS and those created in the visual domain by *particle synthesis*. Particle synthesis has been used to create fire, water, clouds, fog, and grass-like textures, which are analogous to some of the audio effects possible with AGS (crackling fire, water gurgling, windy gusts, and explosions).

5.6.2 Using Granular Synthesis to Display Probability Densities

Recently in [Williamson and Murray-Smith 2004], it was proposed that sonification via granular synthesis is a particularly suitable method for performing the translation of changing conditional and joint probabilities. Consider each cloud in the AGS as a conditional probability density function of a specific class. The probability of being in each class is given by a probabilistic model of the data. For example, if the grain density is held constant, then the number of grains to be drawn from each class is the probability of being in the class multiplied by the grain density.

A spatial distribution example was made together with the implementation of the AGS in the project process. An illustration in Figure 33 shows how a mixture of three Gaussians could be used to map regions of a two-dimensional state-space to sound. Each Gaussian is associated with a specific sound. As the cursor moves through the space, the timbre and/or the pitch of the sound changes accordingly. Although here the densities are in a simple spatial configuration, the technique is general and is applicable to higher dimensions. A QuickTime animation created of a fix path through the state-space was created and this can be found on the attached CD-Rom in folder "Granular Synthesis Using Probabilities". Figure 34 shows a flowchart of the Matlab implementation of the sonification of conditional probabilities (or joint probabilities) using AGS with sampled sounds. The sampled sounds used in the implementation were made on a software synthesizer from Native Instruments called Absynth. The parameters in the initialization together with the sampled sounds were optimized such that the author found the sound results created as pleasurable as possible. The spatial dispersion of the grains where distributed evenly in the left and right channel thus creating a stereo sound file.

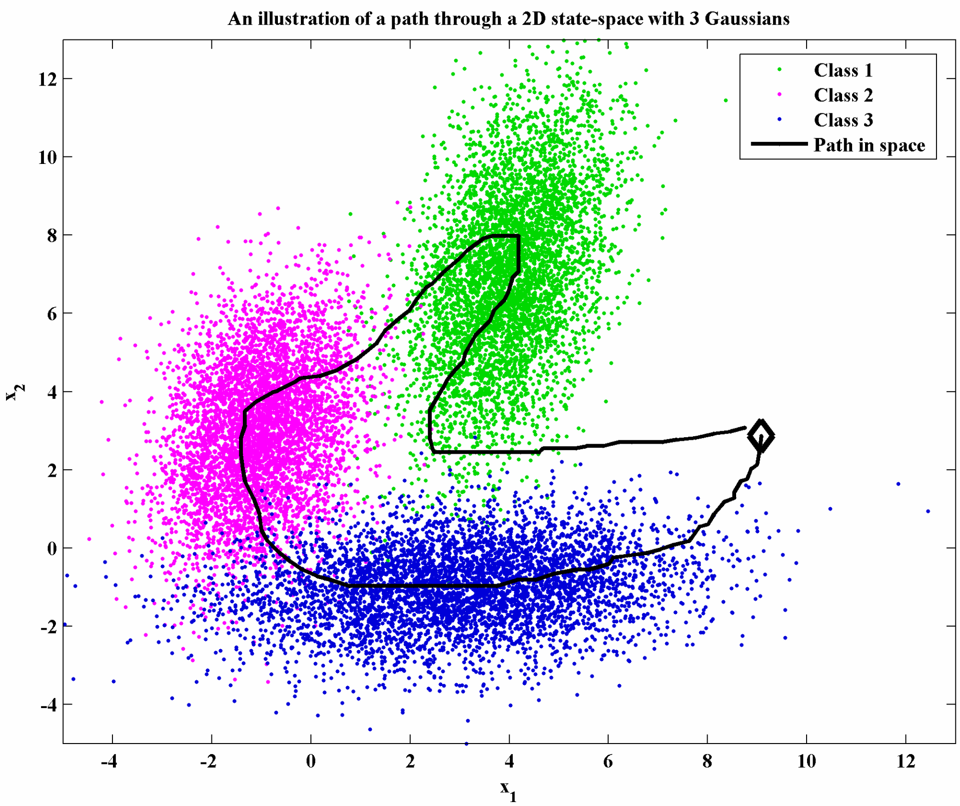


Figure 33. Illustration of a path through a two-dimensional state space comprised of three Gaussians. As the cursor (black diamond) moves along the path the sound at each point is given by the conditional probabilities of being in either of the classes given by the present point of the cursor. Imagine this path as being a tracking of some process through three distinct states represented by sounds.

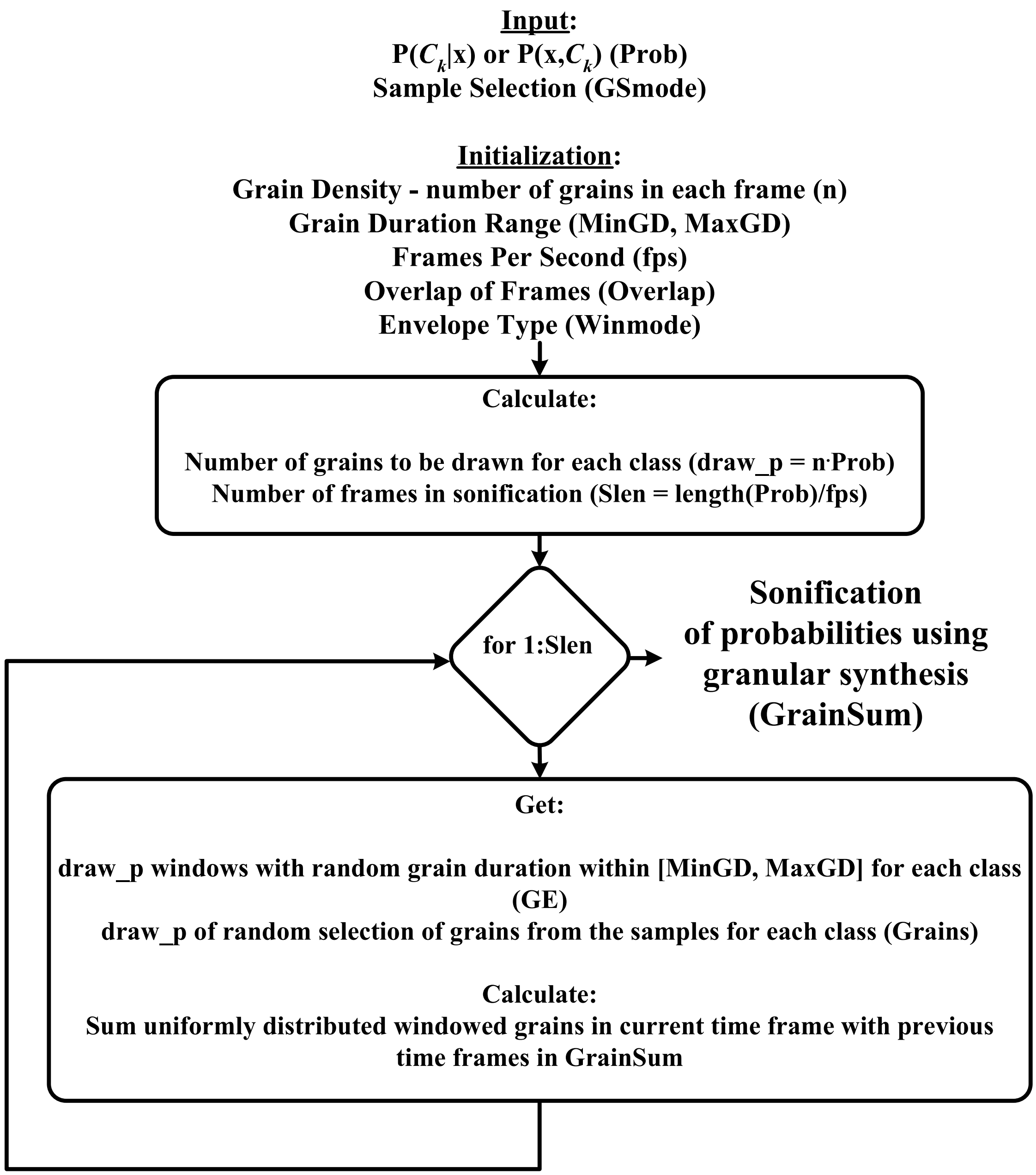


Figure 34 shows an overview of the sonification process using probabilities to control an AGS, with the help of a flowchart. The number of grains drawn from each cloud for each time frame and the length of the sonification is controlled via the input data.

5.7 Subtractive synthesis

Subtractive synthesis implies the use of time-variant filters to shape the spectrum of a usually broad band input sound, known as the excitation source. As the source signal passes through a filter, the filter boosts or attenuates selected regions of the frequency spectrum. If the original source is spectrally rich and the filter is flexible, subtractive synthesis can sculpt close approximations of many natural sounds. In Figure 35 the two main parts of the subtractive synthesis are shown.

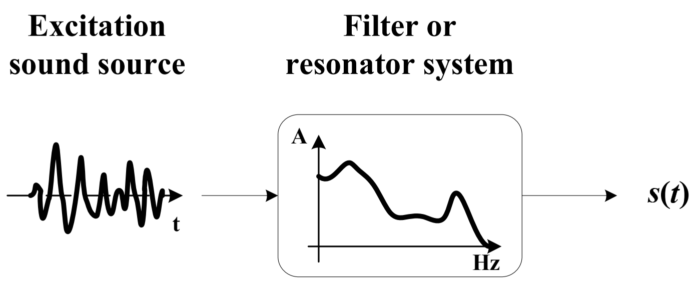


Figure 35. The two main parts of the subtractive synthesis are the complex excitation sound source and the filter or resonator system.

5.8 Modulation synthesis

Modulation in electronic and computer music means that some aspects of one signal (the *carrier*) varies according to an aspect of a second signal (the *modulator*). The familiar effects of *tremolo* (slow amplitude variation) and *vibrato* (slow frequency variation) in traditional instruments and voices exemplify acoustic modulation. In these cases the carrier is a pitched tone, and the modulator is a relatively slow-varying function (less than 20 Hz). When the frequency of modulation rises into the audible range, modulation products or sidebands begin to appear. These are new frequencies added to the spectrum of the carrier. Thus, making modulation synthesis nonlinear, though more efficient in terms of parameter data, memory requirements, and computation time than additive synthesis and subtractive synthesis are. Listed below are some examples of modulation synthesis

- Ring modulation
- Amplitude modulation
- Frequency modulation (FM)
- Multiple-Carrier FM
- Multiple-Modulator FM

The basic FM synthesis technique, referred to as *simple* FM or *Chowning* FM [Roads 1996], means computing a sound signal by

$$s(t) = a(t)\sin\left(2\pi f_c t + I(t)\sin\left(2\pi f_m t\right)\right) \qquad 5.4$$

where f_m is the modulation frequency, f_c is the carrier frequency and the $I(t)$ is the modulation index, which controls the amplitude of the modulator and thus the amount of modulation. As mentioned above, a vibrato effect can be realized for $f_m << f_c$. The frequency of the modulator is usually taken as a fixed multiple $f_m = \gamma f_c$, where for $\gamma = 1$ results in sounds containing frequencies that are integer multiples of the carrier frequency. Sounds with such a spectrum occur in many musical instruments where higher harmonics or overtones arise from the vibration properties of solids [Hermann 2002]. Specific modulation functions $I(t)$ allow to mimic several instruments, e.g. an electric piano (linear decaying function) or brass instruments (rising function). Several timbre classes can be realized using other modulation ratios γ = 2, 3, 1.414, etc. The latter produces bell-like sounds, with $\gamma = 2$ the sound of organ pipes can be synthesized [Roads 1996].

5.9 Physical Modeling, Karplus-Strong and Formant Synthesis

Physical modeling synthesis models the acoustics of traditional instruments, such as a jet of air through a mouthpiece into resonating pipes or a guitar string being plucked at a certain position on the string. Implementing such “high-level” controls, such as plucking position, in other synthesis models would demand extensive parameter modifications, whereas in a physical model the plucking position can be controlled directly by the position of excitation [Hermann 2002]. The limited computation power makes it still necessary to reduce the complexity of the model for simulation. A simplified, though more computational efficient, variant of physical modeling is *Spring-Mesh models*, *Modal synthesis*, *Karplus-Strong synthesis* and *Digital Waveguides*. *Karplus-Strong synthesis* simulates the sound of plucked-string instruments such as guitars, mandolins, and harpsichords; drum like sounds can also be generated. *Formant synthesis* circumscribes a body of techniques that can simulate the resonances of the human vocal tract, as well as those of traditional and synthetic instruments.

5.10 Sound Spatialization

Sound spatialization is the projection of sound in three-dimensional space and is not a synthesis technique though can be regarded as a transformation of sound. The most popular spatial illusions are horizontal *panning* – lateral sound movement from speaker to speaker – and reverberating – adding a dense and diffuse pattern of echoes to a sound to situate it in a larger space.

The sound signal arriving at the ears differ in timing, level and spectral profile as mentioned in chapter 3. The auditory system makes use of these cues to infer the location of the sound source. Interaural time differences (ITD) as well as interaural level differences (ILD) are easily computed by considering the traveling time and attenuation on propagation from the source to the ears. Given a stereo loudspeaker setup as shown in Figure 36, a sound can be localized by intensity panning. The virtual source $\boldsymbol{s}$ can be located within the triangle formed by the listener and the loudspeakers. Assuming an angle θ_{max} between the line of sight and a speaker, a direction (d, θ) is reached by using the gains;

$$\mathrm{A}_{amp} = \frac{r}{d}\frac{\theta}{\theta_{max}} \tag{5.5}$$

$$\mathrm{B}_{amp} = \frac{r}{d}\left(1-\left(\theta-\theta_{max}\right)\right) \tag{5.6}$$

This method is called linear panning and is the simplest form of intensity panning.

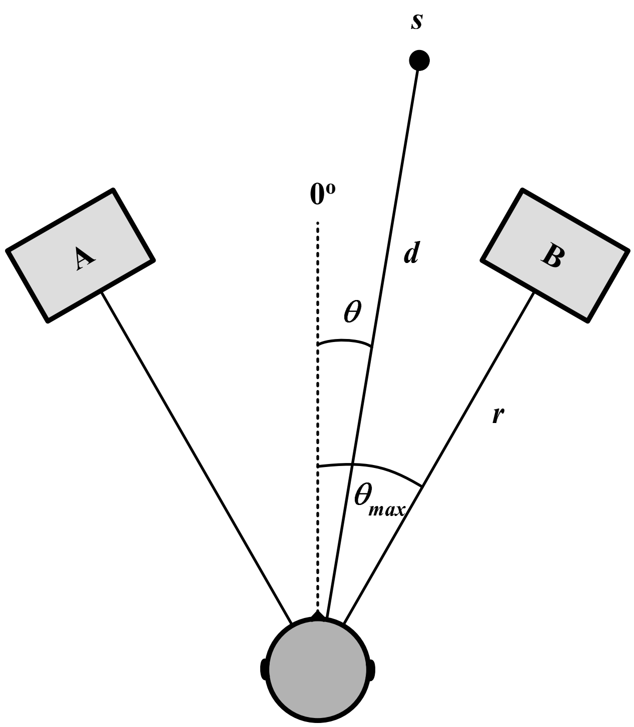


Figure 36. To position a sound source at a point s between two loudspeakers A and B, ascertain the angle θ of the source measured from the middle point between A and B. In the middle θ equals 0 degrees. The angle θ_{max} is the maximum angle, r is the distance of the loudspeakers to the listener, and d is the distance of the sound source s.

As mentioned, spatialization is the projection of sound in three-dimensional space so that the source is characterized by (d, θ, φ) at distance d from the center of the head, at an azimuth θ and elevation φ. The only two effective ways of truly spatializing sound can be achieved by having multi-channel loudspeaker systems, or binaural sound in headphones, which convolves the sound source with angle-dependent filters called head-related-transfer-functions (HRTFs). The latter is a convenient way of presenting 3D sound, though an in-depth presentation of this is out of the scope of this thesis.

5.11 Conclusion

A broad range of sound synthesis techniques were briefly presented in this chapter, where emphasis was laid on the granular synthesis technique. Through [Williamson and Murray-Smith 2004], it was shown that the AGS is an appropriate technique for presenting probabilities, and thus suitable for monitoring known states in a system. An example of a path through a mixture of three Gaussians in a two-dimensional space was implemented to illustrate the above technique. Furthermore, a short section on sound localization and spatialization was presented.

It is clear that different techniques enable various degrees of sound control and results, with trade-offs present in each technique. For sonification it is crucial that the quality of sound produced by the chosen technique be of a certain standard, so as not to be annoying through extended use to the listener. Furthermore, the choice of technique should allow the designer to have access to sound parameters that can convey the information of the data in a predictable and suitable way. Extending sonifications into stereo or three-dimensional sound adds extra parameters, where one can convey information through position or the motion of a sound source.

As mentioned in the introduction to this chapter the translation of data into sound is achieved by the synthesis techniques, and having knowledge and access to various synthesis techniques while designing sonifications is extremely important. The author encourages future researches in this field to commence a project by investigating possible ways of achieving access to various synthesis techniques, for example controlling software synthesizers through MIDI (Musical Instrument Digital Interface).

Chapter 6

6 EEG Sonification

This chapter focuses on sonification of electroencephalographic (EEG) data and presents a concept for an auditory browser of EEG time courses. Initially, a brief introduction to EEG will be given, followed by a section regarding the artifacts found in EEG signals together with a subsection on the analysis of EEG data using ICA. Section 6.3 will focus on other researcher's attempts on EEG sonification and, finally, a section on designing an auditory browser of EEG time courses, which could facilitate in the decontamination of the EEG signal when using ICA, is presented.

6.1 Brief Introduction to EEG

The EEG was first measured in humans by Hans Berger in 1929. The electrical activity, measured via electrodes at the scalp, which arises from neuronal activity in the brain, is known as electroencephalography. The principal source of the EEG is believed to mostly originate from pyramidal cells of layer 3 and 5 in the neocortex, and the amplitude changes in the EEG is a result of changes in the number of synchronously active neurons. The EEG provide information on activities that occur spontaneously or in response to sensory stimuli, where the latter is commonly known as evoked response potentials (EP), whereas sensory stimuli timed to an event is known as event related potentials (ERP). A classical measure of interest is the EEG coherence, which provides a measure of functional correlations between EEG sources.

The EEG has been decomposed into series of fixed broad spectral bands, though these are based on historical discoveries rather than theoretical framework. These bands are described in Figure 37. Generally, it can be said that the frequency of brain oscillations is negatively correlated with their amplitude [Mørup 2005].

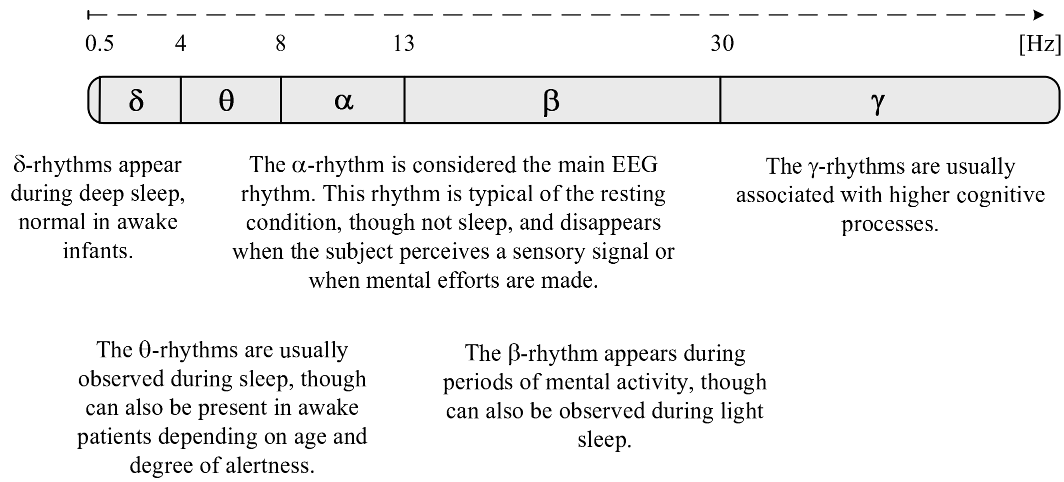


Figure 37. An illustration of the different EEG bands or rhythms on a frequency scale, and short descriptions of them. This figure is redrawn and modified from [Mørup 2005].

6.2 Artifacts in EEG

Contamination of EEG data can occur at many points during the recording process. Artifacts can either be biologically generated or technologically generated – by sources external to the brain. Technologically generated artifacts are, for example, 50 Hz electrical interference (in Europe), poor electrode contact, and other line noises. Biologically generated artifacts stem from eye blinks, eye movement, muscle activity, heart beat (pulse), and head movement. Figure 38 shows waveforms of some of the most common EEG artifacts.

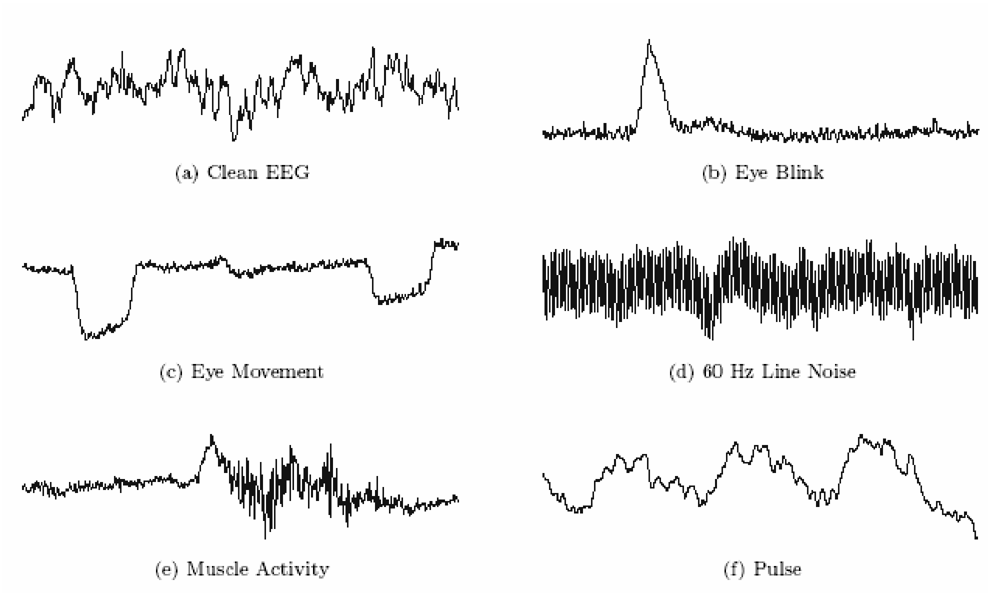


Figure 38. Examples of artifact waveforms. This figure is taken from [Knight 2003].

Artifacts in EEG are commonly handled by discarding the affected segments of EEG. The simplest approach is to discard a fixed length segment from the time an artifact is detected. Often, EEG segments with artifacts larger than an arbitrarily preset value are rejected. However, when limited data are available, or blinks and muscle movements occur too frequently as with some patient groups (e.g. children), the amount of data lost to artifact rejection may be unacceptable. Another common artifact removal strategy is to average trials time-locked to all similar experimental events and discard or ignore averages of data from frontal and temporal electrodes [Jung *et al.* 2001]. An additional dilemma in the artifact removal in EEG signals is the fact that, the artifacts can happen simultaneously, e.g. 50 Hz line noise, eye blink, and heart beat can all be present together. This makes the discrimination between individual artifacts, and artifacts and the non-artifacts even more difficult.

The first attempts at removing artifacts focused on the ocular artifacts. Regression using the electrooculargram (EOG) channel has been attempted in time and the frequency domain [Woestenburg *et al.* 1983]. These methods all rely on a clean measure of the artifact signal to be subtracted out. Since the EOG is contaminated with EEG signals, the regression of the ocular artifacts has the undesired effect of removing EEG signals from the observations. More recently, multivariate statistical techniques, such as PCA, ICA and Parafac models have been proposed to separate and remove noise signals from EEG signals. This approach assumes that EEG observations are generated by linear mixing of a number of source signals, where each method of signal separation applies its own assumptions. Results show that ICA can effectively detect, separate and remove activity in EEG records from a wide variety of artifactual sources, with results comparing favorably to those obtained using regression or PCA based methods [Jung *et al.* 2000], [Jung *et al.* 2001]. The Parafac analysis is still rather new, but seems to be very promising.

6.2.1 ICA and EEG

The ICA algorithm derives independent sources from highly correlated EEG signals statistically and without regard to the physical location or configuration of the source generators. The ICA method is based on assumptions that the time series recorded on the scalp:

- are spatially stable mixtures of activities of temporally independent cerebral and artifactual sources;
- the summation or mixture of potentials arising from different parts of the brain, scalp, and body is linear at the electrodes;
- propagation delays from the sources to the electrodes are negligible.

The two latter assumptions are reasonable for EEG data, and given enough data the first assumption is reasonable as well [Makeig *et al.* 1996]. The following will try to show how ICA (equations 2.40 and 2.41 in section 2.8) is applied to EEG data.

In EEG analysis, the rows of the input matrix, $\mathbf{X}$, are EEG signals recorded at different electrodes and the columns are measurements recorded at different time points. ICA finds an *unmixing* matrix, $\mathbf{W}$, which decomposes or linearly unmixes the multi-channel scalp data into a sum of temporally independent and spatially fixed components. The rows of the output data matrix, **Y**, are called *time courses* or activations of the ICA components. The columns of the estimated mixing matrix or the inverse of **W** give the relative projection strengths of the respective components at each of the scalp sensors. These scalp weights give the *scalp topography* or *scalp maps* of each component, and provide evidence for the components physiological origins.

Once the independent time courses of different brain and artifact sources are extracted from the data, artifact-corrected EEG signals can be derived by eliminating the contributions of the artifactual sources. This is done by removing the non-artifactual time courses from **Y** (i.e. setting them to zero), resulting in **Y´** and then performing the follwing mixing

$$\mathbf{X}' = \mathbf{W}^{-1}\mathbf{Y}' \qquad 6.1$$

where the rows of **X´** are the artifact-corrected EEG signals from the different electrodes. This method has become a standard method of EEG analysis and decontamination, and can easily be done with the help of EEGLAB, which is an open source Matlab toolbox for EEG analysis. The process explained above is illustrated in Figure 39. For further information on ICA and EEG please see [Jung *et al.* 2000], [Jung *et al.* 2001].

Identifying the artifacts in the time courses can be a time consuming affair if many electrodes are used in the EEG measurement and the analysis of these could be accelerated by an auditory browser that aids the user to identify potentially contaminated and non-contaminated time courses, though the basic idea can also be used to detect other relevant information such as the amount of power in the different EEG bands. An application of this type will be constructed in section 6.4. First a section on the previous EEG sonification methods will be presented.

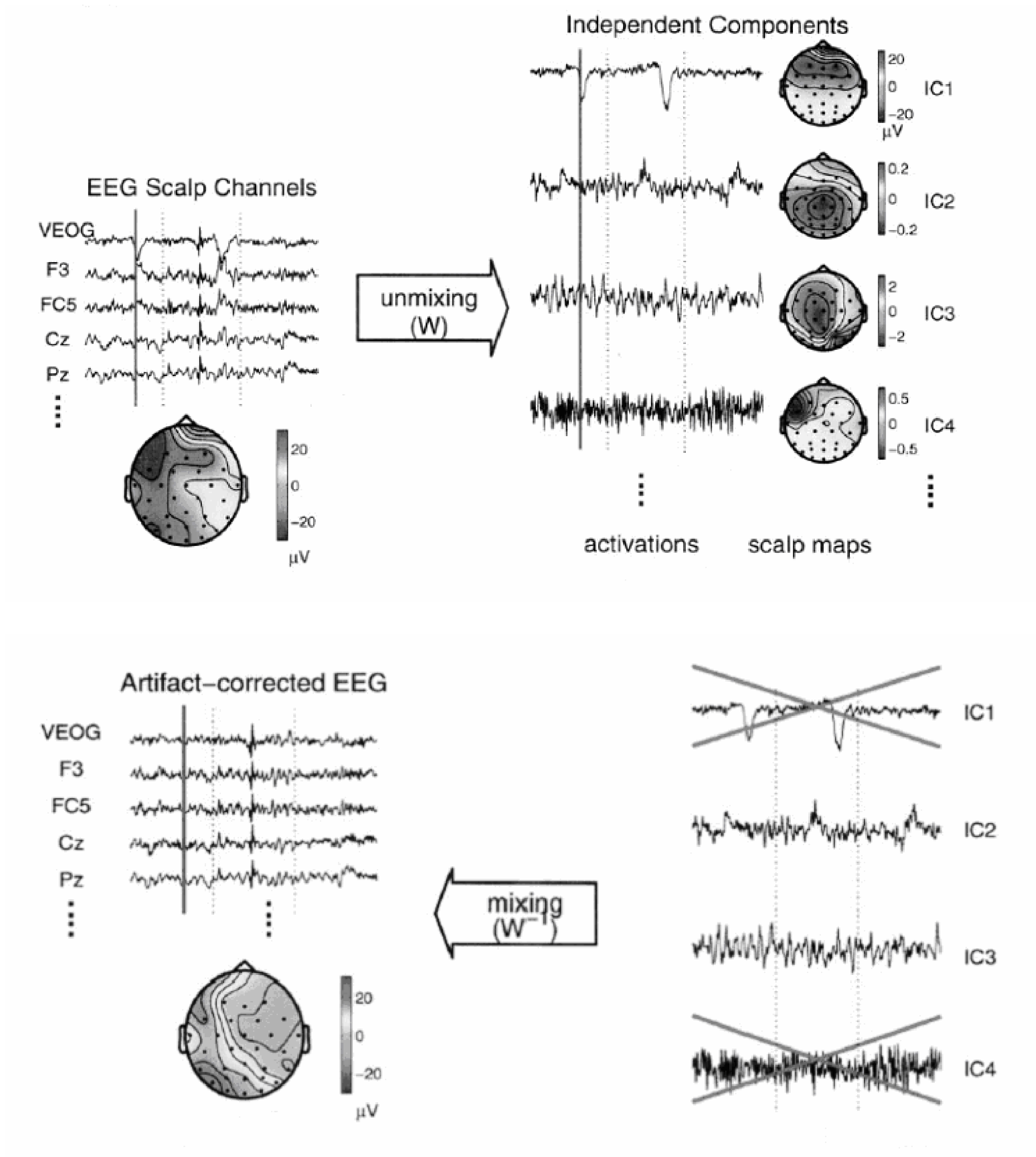


Figure 39. Schematic overview of ICA applied to EEG data as explained in the text above. This figure is modified from [Jung *et al.* 2000].

6.3 Previous EEG Sonifications

The reasons for sonifying EEG data have been linked to the multi-dimensional structure of the EEG data and our auditory systems sensitivity to discern rhythmical and spectral patterns in data of this kind [Baier and Hermann 2004], [Meinicke *et al.* 2004]. Furthermore, it has been stated that, besides the classical analysis techniques such as ERP and coherence studies, sonification of EEG data could be considered as a means of assisting and accelerating data inspection, pattern classification and exploratory data analysis.

The earliest work found on sonification of EEG was done in [Mayer-Kress 1994]. In this article, activation from four electrodes were mapped directly to musical pitches of musical instruments (piano, flute, violin and glocken), which allowed only to present short EEG signal parts (ca. 100 ms) in a reasonable time. The method used could be described as a mixture of audification and parameter mapping techniques, although at the time this was called the "Orchestral paradigm" [Kramer 1994]. The objective of Mayer-Kress was to detect short-time synchronizations during cognitive events in the region of the β- and γ-band. In [Jovanov *et al.* 1999] the same approach was used, even though the objective was different. The main problem with audification is that the resulting sound is very dissonant and independent control over playback speed and pitch is difficult, as mentioned in section 4.5.1, though as described in [Hermann *et al.* 2002] some features, e.g. outliers, are more easily detected from these sonifications than from the extremely noisy time series plots and even in the spectrogram plots.

In [Hermann *et al.* 2002], a new approach in sonification of EEG data was presented. Three sonifications are presented in this paper: *spectral mapping sonification*, allowing the user to follow spectral activations within the brain of each electrode, *distance matrix sonification*, allowing the user to inspect the range and strengths of couplings between different electrodes, and *differential sonification*, allowing the comparison of data recorded for one subject under different testing/stimuli conditions. The spectral sonification uses short time Fourier transforms (STFT) of the time series as a starting point. Given the EEG measurements $s_i(n)$, where $i = 1,\ldots, I$ determines the channel and n is the sample number, the STFT is computed for each channel i by

$$\tilde{s}_i(m,k) = \sum_{n=0}^{N-1} s_i(Cm+n)w(n)\exp\left(-j\frac{2\pi nk}{N}\right) \qquad 6.2$$

where m is the frame number, C the offset between succeeding frames, $k = 0,\ldots,N/2$ the frequency index and N the window (w) width in samples. This gives I spectrograms and for each electrode i a set of N_{osc} time-variant oscillators whose frequency is f_n for $n = 0,\ldots, N_{osc} - 1$ is given by,

$$f_n = \ln(2)\cdot\exp\left(p_{\min}^0 + \frac{n}{N_{osc}-1}\cdot\left(p_{\max}^0 - p_{\min}^0\right)\right) \qquad 6.3$$

where $\left(p_{\max}^{0}, p_{\min}^{0}\right)$ denoted the desired output pitch range in octaves. Let $\tilde{s}_i^k(t)$ denote the time-variant function from interpolating the sequence $\tilde{s}_i(0,k)$, …, $\tilde{s}_i(M,k)$ such that $\tilde{s}_i^k(0) = \tilde{s}_i(0,k)$ and $\tilde{s}_i^k(T) = \tilde{s}_i(M,k)$. Then amplitude of the *i*th oscillator is given by

$$a_i(t) = a \cdot g_\delta\left(\frac{\tilde{s}_i^k(t)}{\max\limits_{t'} \tilde{s}_i^k(t')}\right) \qquad 6.4$$

where $g_\delta(\cdot)$ is a nonlinear function which suppresses all amplitudes less than a given threshold δ and a is a constant. The fraction inside equation 6.4 is a form of normalization of the time-variant function between the values 0 and 1. The parameters of this sonification technique are; the frame size, the overlap size *C*, the pitch range $\left(p_{\max}^{0}, p_{\min}^{0}\right)$, the EEG frequency range, and the threshold δ. As mentioned this allows the user to follow the spectral activation within the brain.

The time-dependent distance matrix sonification is given by;

$$D_{ij}(m) = \left\|\hat{\mathbf{s}}_i(m) - \hat{\mathbf{s}}_j(m)\right\| \qquad 6.5$$

which contain the Euclidian distance between the normalized spectral vectors of channel *i* and *j* in the *m*th window. Thus small numerical values in the distance matrix **D** indicate similar activity in these channels. High similarity is usually expected for electrodes with a small topological distance on the scalp, as a result, the topological distance between electrodes is used to drive the pitch of auditory grains which are superimposed into the sound vector at the appropriate onset. The similarity

$$\exp\left(-D_{ij}(n)\right) \qquad 6.6$$

is used to drive the levels of these grains. Thus loud and high pitched contributions indicate interesting couplings. Sound lateralization, as explained in section 5.10, was also used to give an indication of which electrodes the coupling take place: if both electrodes are located on one side of the scalp the sound is played on the respective channel, couplings between different hemispheres are represented by tones played from the center. As mentioned, this allows the user to inspect the location, range and strengths of couplings between different electrodes.

In contrast to the previous sonifications, the differential sonification technique has a time axis used to distinguish the location of the electrodes, scanning the brain from the frontal side to occipital electrodes. For the comparison, for each condition α and β, each channel *i* and each frequency band *k*, the time sequence of Fourier coefficients $\left|\tilde{s}_\alpha^i[j,k]\right|$, $j = 1,\ldots, N_{i,\alpha}$ is used. The mean

$$\mu_{\alpha,k}^{i} = \frac{1}{N_{i,\alpha}} \sum_j \left|\tilde{s}_\alpha^i[j,k]\right| \qquad 6.7$$

and the standard deviation

$$\sigma^i_{\alpha,k} = \sqrt{\frac{1}{N_{i,\alpha}-1}\sum_j \left(\left|\tilde{s}^i_\alpha[j,k]\right| - \mu^i_{\alpha,k}\right)^2} \qquad 6.8$$

is computed. Assuming that both sequences are independent samples from the same distribution, the random variable

$$t' = \frac{1}{\sigma^i_{\beta,k}}\left(\mu^i_{\alpha,k} - \mu^i_{\beta,k}\right) \qquad 6.9$$

with

$$\sigma^i_{\alpha,\beta,k} = \sqrt{K\left((N_{i,\alpha}-1)(\sigma^i_{\alpha,k})^2 + (N_{i,\beta}-1)(\sigma^i_{\beta,k})^2\right)} \qquad 6.10$$

$$K = \frac{1}{\upsilon}\left(\frac{1}{N_{i,\alpha}} + \frac{1}{N_{i,\beta}}\right) \qquad 6.11$$

is the student-t distributed with $\upsilon = N_{i,\alpha} + N_{i,\beta} - 2$ degrees of freedom. With increasing values of t', it gets more significant that the means for the condition α and β differ. Thus it is used within the sonification to decide, if a sonic marker for frequency band k and channel i contributes to the sonification. The level of the played events increases with t'. As mentioned this allows the comparison of data recorded for one subject under different testing/stimuli conditions. The above is not a thorough description of this sonification technique and for further details please see [Hermann *et al.* 2002].

The above methods can all be categorized as parameter mapping sonifications. In the article they conclude that due to high temporal resolution of the auditory system the EEG data recordings can be analyzed in a very condensed way. The methods were not tested for there usability or usefulness and were only tried for experimental data. In [Hooper 2004] a parameter mapping sonification using generalized mutual information was undertaken to revel functional coupling between cortical regions for all possible paired combinations. Again these methods were not tested for usability or usefulness, though real EEG data was used. In [Meinicke *et al.* 2004], which also was presented in section 2.10 as a source of inspiration for testing the augmented data set, is co-written by our good friend Hermann. Their particular interest was to identify features in EEG data which discriminate between different conditions according to the stimuli presented in the experiments and thereby draw conclusions on the cognitive processes associated with the chosen conditions. The analysis of their results from the data analysis was of course aided by sonifications. The features used in the extended ICA-FX algorithm were obtained by band-pass filtering the EEG signals from 0.3 Hz to 35 Hz and applying Short Time Fourier Transforms (STFTs) with half overlapping windows of 1 s duration to each EEG channel. For each window spectral amplitudes were averaged over the ranges of six frequency bands, shown in Table 1.

Frequency band:	δ	θ	α_1	α_2	β_1	β_2
Range [Hz]:	1–3	4–7	8–10	11–12	13–18	19–30

Table 1. Classification of EEG bands used for analysis.

This table is taken from [Meinicke *et al.* 2004].

Then for each window position they derived one data vector with dimensions according to the number of channels times the number of frequency bands – including the additional label dimension, as described in section 2.10. In the paper the spectral mapping sonification was used. Practically, this was done by mapping the amplitude to the energy within a spectral band resulting in louder sound contributions if the band shows higher activation. Pitch was used to separate the adjacent bands by musical interval, e.g. a fifth. Pitch change is also used to represent variations in the activations. The left/right audio channel represents the left/right hemispheres.

In [Baier and Hermann 2004] an interesting model-based sonification was attempted. In the paper, they develop a method that can be used to estimate both frequency and phase of *rhythmic events* in signals with broad-band spectra. This was done, in short, by using an ensemble of differently, but sharply tuned differential equations (physical model of oscillators) to analyze a range of frequencies in parallel. The information found from this analysis was a temporal ordering of the detected rhythmic events for each differential equations eigenfrequency, giving the spectral information of the rhythmic events. Thus, the output of the model is converted to a sonification which communicates temporal and spectral information about the EEG signals rhythmical events. In the summary, they mention the need to asses the usability of their sonification and the construction of an improved human-computer interface that allows fast and easy navigation and exploration of EEG data.

The objective in [Hinterberger *et al.* 2004] is rather different than the previous mentioned article, though the sonification bears some resemblance to [Mayer-Kress 1994]. They provide audio feedback of brain signals which operate a verbal spelling interface. Thus, interactive sonification is used for the training of self-regulation of the slow cortical potentials (SCPs) used as features to operate the interface. This is done, in short, by; band-pass filtering the EEG signal into the different frequency bands, detecting the temporal extrema in the frequency range below 12 Hz and the band power of the higher frequency bands, and converting the information of the different bands to distinct MIDI instruments changing in pitch and velocity. The change in pitch is governed by the time distance between temporal extrema, whereas the sizes of the maxima serve as values for the touch of a MIDI instrument. They conclude that physiological regulation of SCPs can be learned with auditory and combined auditory and visual feedback, although the performance in the latter case is significantly worse than visual feedback alone. Recently, in [Hinterberger and Baier 2005] a real-time version, called POSER, of the above system is implemented. POSER stands for Parametric Orchestration Sonification of EEG in Real Time. In appendix XX the signal flow chart of their system is illustrated together with an illustration of the extraction of parameters.

6.4 Using Sonification for Identification of Artifactual Features in Time Courses

As stated, the main interest of this thesis is auditory browsing through time courses in the interest of decontamination of the EEG signals by excluding artifactual contaminated time courses. Due to limited time the focus will be on eye blinks, though can be extended to the detection of multiple artifacts or, for example, displaying the amount of activity in each EEG band. The flowchart of sonifying a time course is illustrated in Figure 40, and this is also the general order of topics that are presented in the following sections. More specifically the topic of the following sections consists of the identification of eye blinks due to the lack of available data, examining simple features, creating a training and validation set, optimal window size test, leave-one-out method and cross-patient validation of the different classifiers, and finally using the output of the model as parameters for the granular synthesizer to produce a sonification of the degree of artifact, i.e. eye blinks, contamination of the time course.

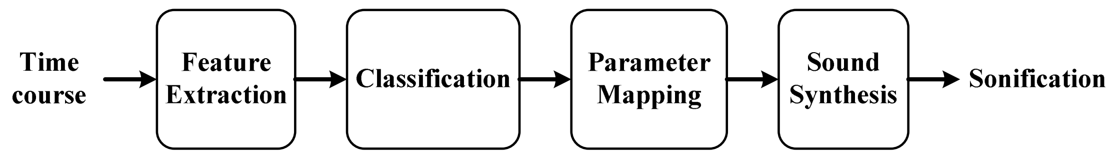


Figure 40. Flowchart of the process of sonifying a time course

6.4.1 Extracting Relevant features for Eye Blinks

The EEG data used in the following was recorded by Sidse Arnfred (senior researcher at the Cognitive Research Unit at the Psychiatric Department at Hvidovre Hospital) and the experiment conducted was equivalent to the visual-binding experiment presented in [Herrmann *et al.* 2004]. In the experiment, Arnfred also recorded her patients sitting in a relaxed state with their eyes open and this data was used to create a training and test set. Due to lack of time, eye blinks from only two patients were located. The initial analysis and preprocessing of the data was performed in EEGLAB. The initial steps included importing data into EEGLAB, referencing the data, assigning channel locations, high-pass filtering (@ 0.1Hz) to remove DC drift, performing ICA and saving the resulting EEG-struct (generated by EEGLAB) for both patients for later use in Matlab.

The identification of the components that capture the most of the eye blinks is a relatively easy affair for as strong as a signal as the one created by the blinking of the eyes. The identification process included comparing the signals recorded in the frontal electrodes, e.g. at the Fpz electrode, and the time courses and the corresponding scalp maps. Scalp maps with main energy in the frontal region, as shown in Figure 41 and corresponding time courses with signals resembling the form in Figure 38b, were found and these were used for the data for this experiment.

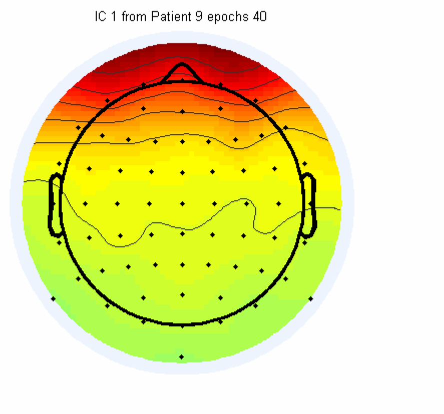


Figure 41. Scalp map of the component capturing the eye blinks

After having isolated the eye blink data, the next step in the process was choosing features that describe the eye blinks. This was relatively easy due to the fact that eye blinks distinguish themselves from other artifacts and greatly from other non-artifactual occurrences in the EEG signal. The two first features tested were variance and zero-crossings. These features were chosen due to the fact that when eye blinks occur, as seen in Figure 38b or Figure 42, a change resembling a strong spike is observed. This large change from the remaining signal will be registered by the variance, whereas low number of zero-crossings should capture signals that do not oscillate around the zero, such as the eye blink. This prior information is used to make relevant features of the data, as suggested in section 2.1. Artifacts such as eye movement and perhaps the commencement of muscle activity, as shown in Figure 38c and e, could also be registered by these features due to their resembles of eye blinks.

In Figure 42 the two features and the corresponding time course is shown. From the figure it is apparent that large values of variance and small values of zero crossings correlate with the presence of eye blinks. From this analysis it was decided that the features variance and zero crossings were good features for identifying eye blinks, though a further validation of the features is needed to confirm to what extent the features describe eye blinks. This was done by manually extracting the eye blinks from either patient to be used for generating eye blink features. The features extracted were then used as training and validation sets. Furthermore, the features from the eye blink sets were extracted for varying window sizes, due to the fact that an optimal window size was imagined.

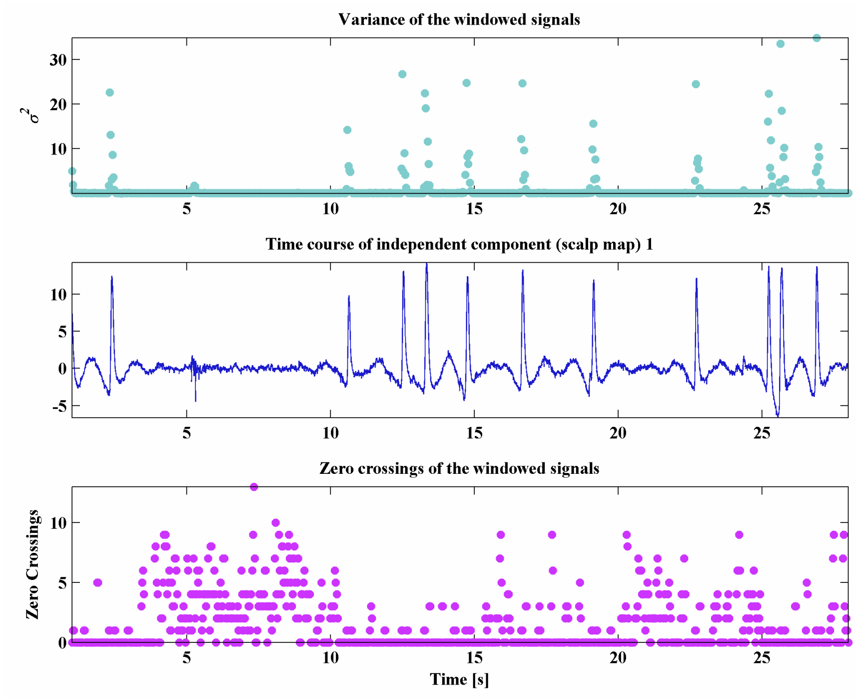


Figure 42. The tested features compared to the time course signal capturing a large part of the eye blinks. A correlation is seen between eye blink occurring, and high variance values and a low number of zero crossings.

The variance of a signal depends on the strength of the signal and taking into account that the ICA components can have arbitrary strength some sort of normalization of this feature is needed. This was solved by normalizing with the variance of the window with the variance of the whole time course signal, as follows

$$\sigma^2_{Normalized} = \frac{\sigma^2_{window}}{\sigma^2_{timecourse}} \qquad 6.12$$

The number of zero crossings was determined by calculating the number of sign changes occurring in a window.

Due to time constraints and the choice of good targets, there were only extracted 37 eye blinks from patient 9, and 14 eye blinks from patient 6. The sparseness of the data encouraged the leave-one-out (LOO) method, described in section 2.5.1, for varying window sizes. The following section presents the results of the validation of the features for the varying window sizes and the different classifiers.

6.4.2 Validation of the Features and the Classifiers

This section presents the inference and evaluation procedure for several classifiers on the eye blink and non-eye blink data. As mentioned the LOO method was used as the validation technique due to sparse target data. This validation was done for all window sizes for the two patients. The measure used to compare the classification performance of the different classifiers is the area under the ROC curve. The reason for this is due to the fact that the probability of an eye blink occurring was estimated to 1% and 3% of the total length of the data for patient 6 and 9, respectively. Thus, the accuracy measure is not adequate due to the fact that the number of negative cases is much larger than the number of positive cases. The probability of an eye blink occurring was estimated by

$$P(\text{eye blinks}) = \frac{\text{Total no. of eye blinks} \cdot \text{Mean length of eye blinks}}{\text{Total length of EEG signal}} \qquad 6.13$$

It should be mentioned that the non-eye blink feature sets were created by selecting 1 – *P*(eye blinks) randomly from the remaining time courses for each training and validation instance. The results for the two patients are presented in Figure 43.

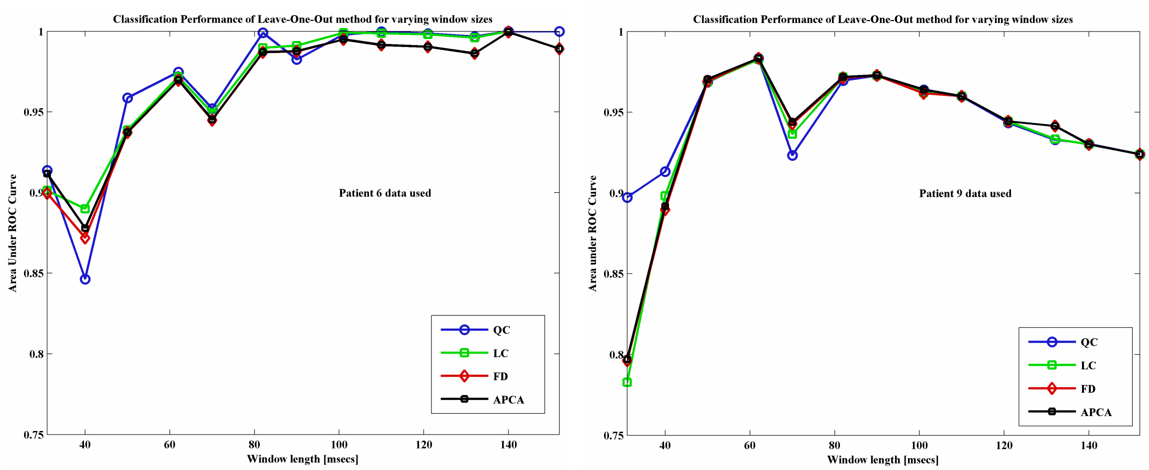


Figure 43. Results of leave-one-out method for patient 6 (left) and 9 (right) with varying window sizes. In connection with chapter 2, the APCA also seems to work comparable to the FD even for non-experimental data.

For every window size the LOO method was performed 14 and 37 times for patient 6 and 9, respectively, and for every window size the mean area under the ROC curve is calculated and presented. The results show very good performance for all the classifiers.

A test usually preformed when data is taken from different patients is Cross-Patient (CP) validation method, which tests the generality of the model across patients. Thus, the training set stems from one patient and the validation from another. In Figure 44, the CP validation method is preformed for both patients and for various window sizes.

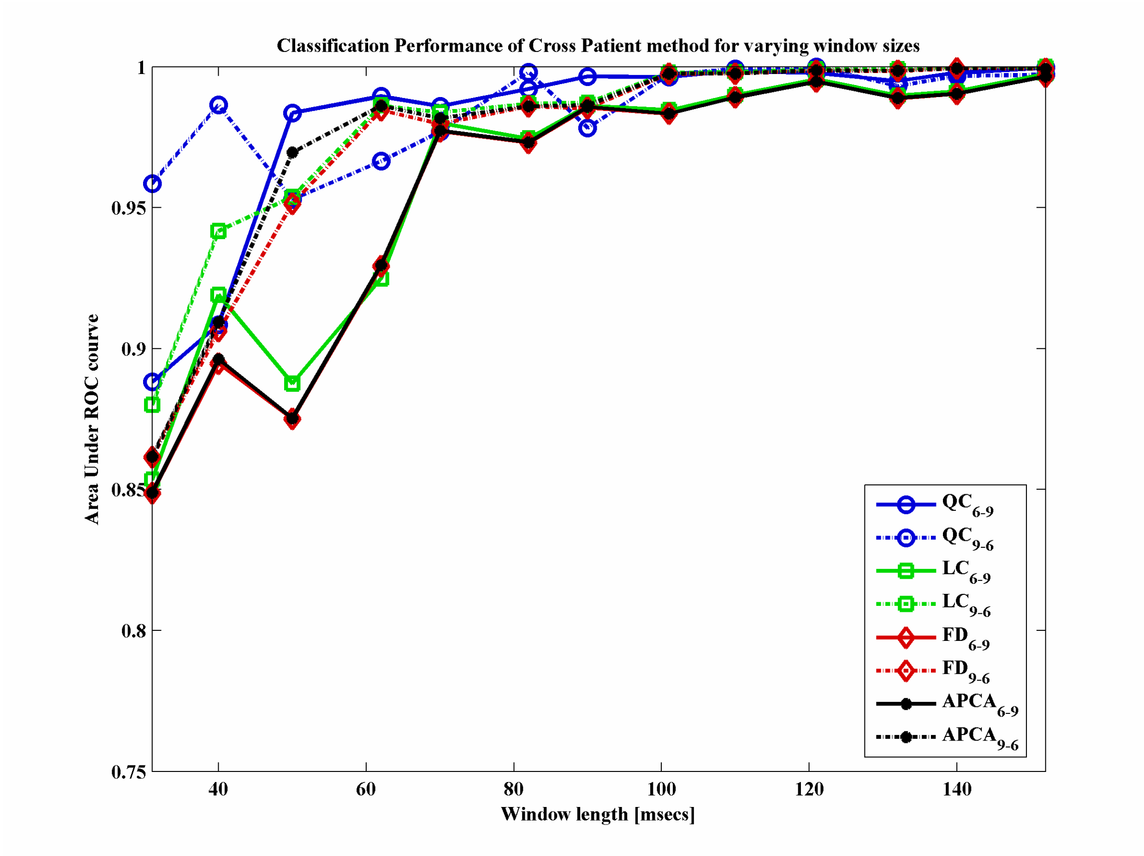


Figure 44. Results of Cross patient method for two patients with varying window sizes. The solid lines show the results of training with patient 6 data and validating with patient 9 data (6-9), and the dashed lines show the results of training with patient 9 data and validating with patient 6 (9-6). The best results are generally obtained for larger window sizes, and when training with data from patient 9 and validating with data from patient 6, though both give surprisingly good results.

Due to the surprisingly good results for the CP test, the data from both patients were combined and the LOO method was performed on the 37 + 14 = 52 targets. The results of which are shown in Figure 45. From this it was decided to use the linear classifier as the discriminant function using a window size of 121 msecs, which is a trade-off between discrimination and the resulting length of the feature vector. The mean pooled covariance matrix and the mean means for both classes were estimated from the 52 trials. These where then tested using the same LOO method though no training was preformed, since the parameters of the classifier already where estimated. The results from this were nearly indistinguishable from the results obtained in the former test, and thus these parameters were saved to be used in the decision making procedure (classifier) before the parameter mapping and sonification components.

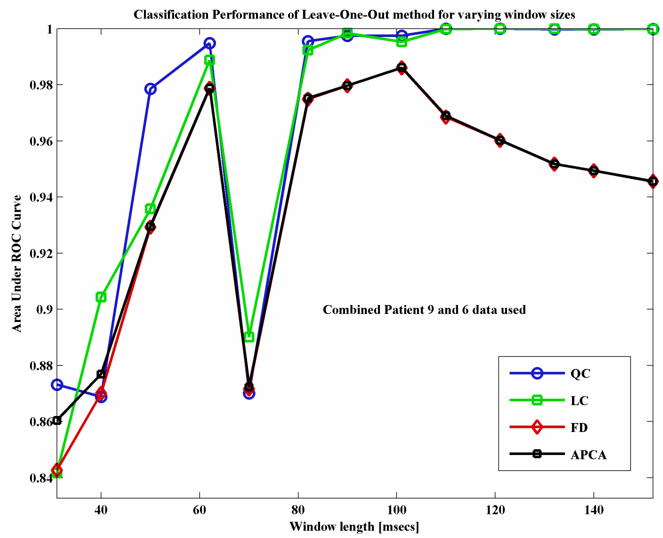


Figure 45. Results of LOO method on the combined data from patients 9 and 6.

6.4.3 Mapping of the Classifier Output to the Granular Synthesis Component

The idea behind the browser is to allow the user to quickly sift through time course data and identify the ones that are contaminated with artifacts, i.e. in this case eye blinks. Since the time courses for EEG experiments are rather long (in this case just under 10 minutes) some sort of further compression of the data is needed to generate sonifications of lengths that are not tiring to the user. A length of about 4-5 seconds was found (by the author) to be an acceptable length for the sonifications. Thus the sonifications give the user a rough estimate of where and how contaminated a time course is. This information is conveyed by using two sounds that are uniquely different depending on the classification of the linear discriminant, as illustrated in Figure 46.

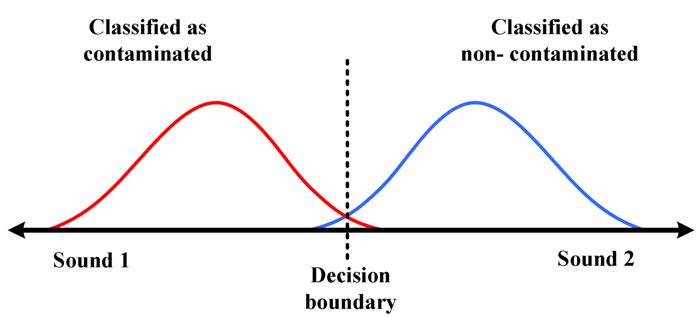


Figure 46. A schematic illustration of the information being conveyed by the sonifications.

To insure that the user is able to differentiate between contaminated and non-contaminated time courses the sounds are to be as different as possible, thus one could have sounds that are different in pitch and timbre. As described in chapter 5, the strengths of these sounds are controlled by the probability of being in either class given the value of the features and the grain density specified in the granular synthesis, though, as shall be explained, this is done a bit differently due to the compressing of the signal.

The output from the linear classifier component is a vector **v** with values either -1 or 1 indicating a detection of an eye blink or a non-eye blink, respectively. The time compression procedure chunks **v** it windows v_{win} and produces one value w, either -1 or 1, for each window. The value of w is determined by

$$w = \sum_i v_{win}^i > \lambda(1 - \delta) \qquad 6.14$$

where λ is the window length and δ is a threshold value. The values in v_{win} are summed producing a value between the [$-\lambda$, λ] for each instance. Thus, if the threshold δ is, for example, set to 1, then half of the values in v_{win} need to be classified as eye blinks before w is given the value -1. The output from the time compression component is a vector **w**, which is a compressed form of **v**.

The values in **w** now have to be conditioned to the granular synthesis component, which takes a matrix **P** with values between 0 and 1. This is done by what is christened the convert to pseudo probabilities component, and is preformed by

$$\mathbf{P} = \tanh\left[c \cdot \left[(\mathbf{w}+1)/2 \quad 1 - (\mathbf{w}+1)/2\right]\right] \qquad 6.15$$

where c is a constant that determines the balance of the two sounds when either is detected. Equation 6.15 produces an $n \times 2$ matrix, where n is the length of **w**. A flowchart of the process from the time course to the matrix **P** is shown in Figure 47.

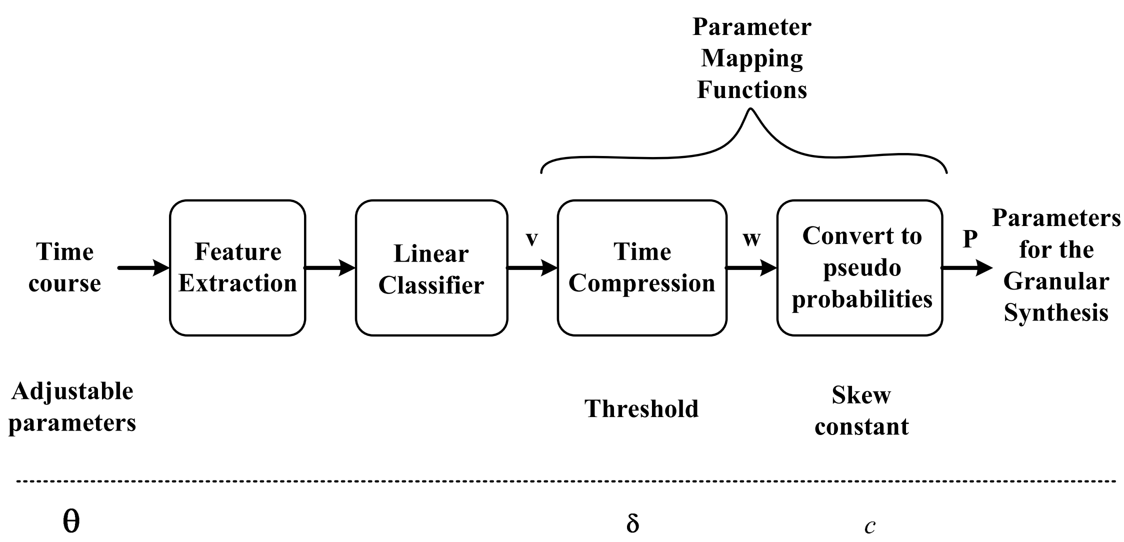


Figure 47. A flow chart of the process from the time course to the creation of the P matrix.

The threshold parameter δ was adjusted so that for time courses with few eye blinks (<5%) where registered the percentage of non eye blinks classified in **v** would be the same as in **w**. For time courses with many eye blinks registered in **v** (>40%), thus where eye blinks occur regularly, the resulting **w** gave close to 100% eye blink classification. Thus, when listening to the sonification of time courses with many eye blinks the main sound one hears is the sound connected to the contaminated sound, whereas the time courses with no eye blinks or little eye blink-ish waveform shapes, i.e. spikes, one mainly hears the sound connected with the non-contaminated sound, though, where there are spikes in the waveform one hears the sound connected to the eye blink sound.

A graphical user interface (GUI) in Matlab was constructed to connect all the components and make an interface where it is easy to sonify different time course from an EEG-struct produced by EEGLAB.

6.4.4 GUI of the Sonification procedure

This section gives a brief overview of the GUI designed in this project, depicted in Figure 48.

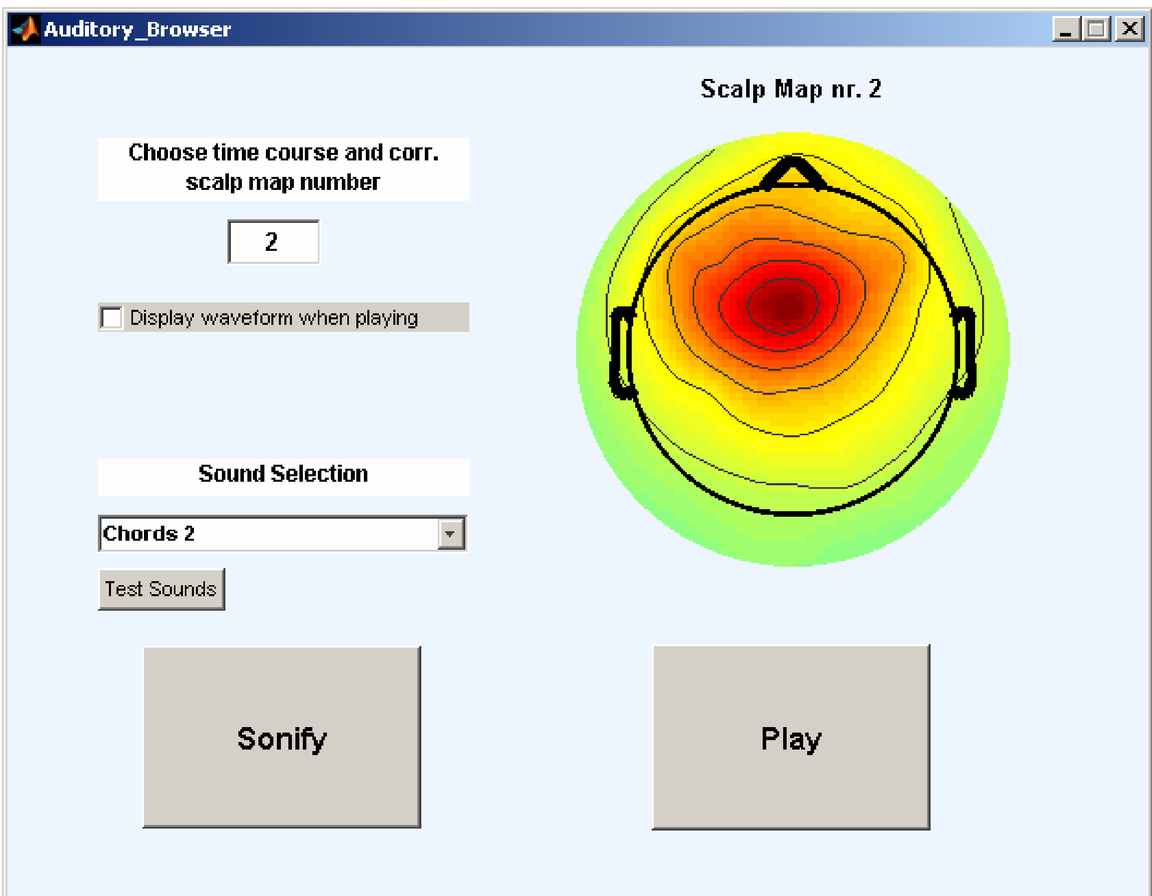


Figure 48. GUI of the time course auditory browser.

The GUI was kept very simple with only the essential parameters assessable to the user. The main parameters are;

- ***Choice of independent component***. Corresponding to the time course sonified and scalp map displayed (top left hand side)
- ***Choice of sounds***. One can choose between several sounds (middle left hand side)
- ***Perform sonification***. (bottom left hand side)
- ***Play sonification***. (bottom right hand side)

For the auditory browser to work one must have an EEG-struct from EEGLAB with the ICA components of the EEG signal in memory. An example of a typical way a user would employ the auditory browser could be: the user enters a component to be sonified and the scalp map of the selected component appears, if the sound type has been selected, one can create the sonification simply by pressing the Sonify-button, wait until the process is done and then press the Play-button to listen to the sonification. A further description of how to use the GUI and two small examples can be found in the enclosed CD-Rom, please see under folder "GUI and Instructions".

6.5 Conclusion

Initially, a short introduction to EEG was given followed by a section on the artifacts present in EEG signals. Furthermore, a brief section on applying ICA on EEG for the purpose of artifact removal was presented, which led to a section on previous methods for EEG sonifications. Finally, a suggestion of how to use sonifications in auditory browsing of time courses with the intent to decontaminate the EEG signal when using the ICA method was presented. In this section descriptions of the features and of the validation procedure of the classifiers were included, together with a section were the author attempted to formalize the heuristic procedure of mapping the outputs of the classifier to an asynchronous granular synthesis. In connection with this a GUI of the browser was created to allow and encourage the reader to try the auditory browser.

Chapter 7

7 Discussion

The author believes that some statements that are repeated by many researchers paint a glorified picture of sonification as being a solution to the “limited” dimensionality of the visual system and thus proving useful for analyzing complex multivariate data. Is it clear how much information we can absorb and understand at one time? Does the auditory system allow comprehensible analysis of information in higher dimensionality than the visual system? These are important questions which the author is unable to answer, though feels are very important questions that need answering before statements such as the one presented above can be used as an incentive to use sonifications.

Sonification, however, does allow data to be experienced in a new way, and can in some instances supply information that would not have been accessible otherwise. Can it then be specified in which instances using sonification will give a deeper understanding of the data? Like every method, sonification will not be the best choice for all problems, though, as mentioned, it does allow a new way of viewing data. A hint to this question could have been given by the examples of the Voyager 2 and the Quantum Whistle, where the information that was being sought was embedded in time series signals of high noise levels and was therefore unable to be perceived via the visual system.

Using auditory displays leaves the eyes free for other important tasks that require the eyes fixed, for example, in many medical and critical monitoring situations. Here the concept of backgrounding is used, thus only drawing attention to specific information when unexpected states in the sonification, i.e. the data, are perceived, usually when large changes occur.

Sonification may be a good aid for rapid screening of data since an auditory stream can be consumed with comparable little effort. Though usability tests proving where and why sonifications are superior to its visual counterpart are an invaluable source of information, and should be encouraged and possibly conducted with psychologists or psychophysicists. As mentioned earlier, the high temporal resolution, i.e. short detection times, of the auditory system allows the possibility of presenting data in a compressed form. Sound is also a good form of displaying time series data sets [Flowers 2005] due to the fact that one can retain the time structure of the data. As in the auditory browser presented in this thesis, one can via sound not only detect *if* eye blink or eye blink-ish events occur, but also *when* in time they occur. This is an advantage over static visual displays such as pie chart or numerical presentations of how many blinks were detected.

When using sound one should refrain from using it in situations where it can interfere with speech communication or in open workspaces, where it can rapidly become annoying. Using bad sound quality or tiring sounds can also create fatigue and annoyance and will ultimately result in other solutions being found to convey the information. The limited ranges of loudness are a drawback of using sound. The ranges have to be in a range of pleasantness, i.e. presenting something too loud creates fatigue and annoyance, and presenting a sonification too soft makes the sonification sensitive to masking effects created by other sounds in the environment resulting in strenuous listening. The lack of orthogonality of auditory dimensions make sonifications become increasingly unpredictable the higher dimensions are used to convey the information. When working in this high dimensionality [Hermann 2002] best recommends the designer of a sonification to experiment until a solution is found that subjectively allows distinguishing the parameters of the sound.

The cultural bias and the user's ability to understand and interpret sounds play an important part in presenting data via sound. Musicians, for example, are better at discerning changes in sound than non-musicians. This leads to the learnability of auditory displays: inexperienced users might need to be trained in using auditory displays before they can yield the full benefit from these types of displays. Furthermore, experienced users may also require other sonifications and other interaction possibilities than inexperienced users [Flowers 2005]. A reason for this is that auditory displays are not common in conveying information and as with everything that is met for the first time some degree of familiarization is needed.

The auditory browser presented in this thesis was intended to aid in the search of isolated artifacts in time courses for researchers that use ICA to form "decontaminated" versions EEG signals. Since the author used a lot of time on studying sonification literature and perhaps sidetracked and used too large a part on investigating the interesting possibilities of augmenting data sets for extracting relevant features in binary classification problems, the time remaining was limited. Due to this the only artifact taken into account was eye blinks, which is the easiest artifact to detect. This could of course be extended to include a search for more artifacts, such as muscle activity and line noise. A further extension could have been a search through ERP time courses for stimulus-locked, response-locked and non-phase locked activities as preformed manually in [Jung *et al.* 2001]. This could aid in finding "decontaminated" versions of ERP and the components that constitute the ERP. The auditory browser can be categorized as lying in between parameter mapping and parameterized auditory icons on the analogic/symbolic continuum. The reason for this is that the auditory browser does not represent analogically what is happening in the data, though does give a sense of the categorical time structure of the data.

Instead of creating a model of eye blinks one could also have chosen to sonify the features of the eye blink. Sonifying the two features in this thesis is not a difficult affair; one could, for example, map one parameter to pitch and the other to a prominent dimension of timbre, such as brightness. However, if more features are needed in specifying other artifacts this kind of "feature mapping" becomes more difficult and less intuitive. Therefore a model that classifies the multidimensional feature space and outputs values corresponding to which state it is in is a more general solution. One could argue that one removes oneself further from the actual data, due to the fact that the model only can identify the learned states from the current features, thus making it a less flexible approach if other searches for non-learned states are wanted, i.e. exploratory data analysis. However, exploratory data analysis and this form of monitoring are two separate tasks as suggested in [Kramer 1994 p. 15], and one could argue that new features can be extracted and the classifier updated only requiring the need to find a new way of representing the new state, making this approach a very flexible approach. Extending the feature space could have the result that other models types are needed, such as neural networks, relevance vector machines or Hidden Markov Models [Williamson and Murray-Smith 2003], though this does not influence the parameter mapping when using the probability outputs of the classifier.

A feature that the author wanted to have added to the GUI was a "zoom" function, which is inspired by zooming in the visual domain. This would have allowed the user to listen to a short presentation of the time course, as it is now possible, and then be able to zoom in to listen to a more detailed representation of an interesting part of the time course. Thus, allowing the user the possibility to listen to a small area of the time course prolonged in time. This would perhaps have been interesting for ERP analysis, as suggested in [Mayer-Kress 1994]. Another, possibly farfetched, idea the author had was to use the scalp maps, i.e. the columns of the mixing matrix, to control where the sound in a three dimensional space originated from. Thus, if, for example, listening to the eye blink time course the sound generated would be perceived as coming in front of the listener. This would perhaps eliminate the need to present the scalp map in the GUI and for ERP analysis this could give an idea of where in the brain the process stem from and how they evolve in time. The author had imagined this best could be realized by using HRTF. Recently, in [Lokki 2005], HRTFs were used to create a three dimensional sound space to inform researchers in room acoustics in slow motion about the spatial distribution of early reflections and spectrum of each reflection. This was intended as a more intuitive investigation tool for room acoustic designers.

As mentioned, the hypothesis of this form of browsing being more efficient and informative than the present form of analysis of time courses has to be tested, though was out of the scope of this thesis. Empirical research about learnability, reliability and usability are an important requirement for a more quantitative assessment of the performance of different sonification strategies and should be encouraged until a clearer picture of why sonifications work or do not work. This is important due to the fact that most of the mappings from data to sound are done in a heuristic fashion, requiring quantitative assessment of the sonification. It is suggested that a study must be done on the audience demographics [Walker and Kramer 1996], e.g. researchers in EEG, and thereafter the obtained knowledge should influence the choice of the testing technique(s) and the sonification, also including the GUI.

For researchers that in the future are interested in the field of sonification one needs to have easy access to multiple sound synthesis techniques and sound transformation procedures. Having access and in-depth knowledge about these could make the mapping procedure easier and more intuitive.

Finally, the author reconfirms that, the frequently stated need for a flexible toolbox for sonification research and data exploration, still is in great demand.

Chapter 8

8 Conclusion

This project concerns auditory browsing based on monitoring the states (i.e. contaminated and non-contaminated) in EEG time courses. Features were extracted from the EEG time courses and a classification of these was performed. The granular synthesis technique was used to translate the probabilities of being in a state at a given time to auditory information. To ensure that relevant changes in the time course data were perceived when listening to the sonification; the classification component was an important part of the translation or mapping process. As a part of the classification study, the concept of augmented data sets for binary classification problems was investigated.

The concept of augmented data sets was presented and a heuristically investigated in chapter 2. Here it was seen that the discriminatory value of the PCA increased to the level of linear discriminant functions. The results seemed to show that when using APCA the $d+1$ eigenvector for small class labels in most cases gave the best results. The APCA procedure was tested on experimental data (chapter 2) in many dimensions and on real data (chapter 6) in two dimensions where it, in both cases, gave consistent results. Compared to other linear discriminant techniques, however, the APCA method is very computationally inefficient. Furthermore, a preliminary investigation of augmenting ICA (infomax) was also given. For data types of non-zero mean the AICA showed a general increase in discriminatory value when compared to ICA. For data types of zero mean, which are super Gaussian distributed, the directions of the column vectors in the mixing matrix seemed to correlate better with the directions of the data set than those obtained when ICA was run on the same data. Both the APCA and AICA are limited to binary classification problems and it is clear that further investigations of these methods are necessary to give a lucid and precise explanation of what is going on.

A brief introduction to auditory perception, and sound synthesis was presented as part of the understanding of the sonification procedure. Auditory perception and findings in this field are crucial to have in mind when designing sonifications and to understand the limitations of sonifications. Sound synthesis is a basic part in converting data to sound and in-depth knowledge of these techniques will aid the designer of auditory displays to perhaps more intuitively perform these mappings. A mapping of probabilities using asynchronous granular synthesis was presented and implemented, which was later utilized for an auditory browser. The chapter on sonification gave an overview of the field of sonification and presented a section on the issues in designing sonifications where observations in auditory perception were related to sonification design.

In chapter 6 previous EEG sonification techniques were presented together with the design procedure of the auditory browser implemented in the course of the project. Furthermore, a GUI was designed to combine the components in an easy to use interface.

References

Baier, G. and Hermann, T. The Sonification of Rhythms in Human Electroencephalogram. Proc. of the ICAD 2004, Sydney, Australia, July, 2004

Barras, S. and Kramer, G. Using Sonification. Multimedia Systems, Springer-Verlag, pp. 23-31, 1999.

Benford, S. and Greenhalgh, C. Introducing Third Party Objects into the Spatial Model of Interaction. Fifth European Conference on Computer Supported Cooperative Work, Lancaster, UK, 1997.

Bishop, C.M. Neural Networks for Pattern Recognition. Oxford University Press Inc., New York, USA, 1995.

Blattner, M. M., Sumikawa, D. A., and Greenberg, R. M. Earcons and icons: Their structure and common design principles. Human Computer Interaction, 4, 1, pp. 11-44, 1989.

Bonebright, T.L., Miner, N.E., Goldsmith, T.E. and Caudell, T.P. Data Collection and Analysis Techniques for Evaluating the Perceptual Qualities of Auditory Stimuli. Proc. of ICAD 1998, University of Glasgow, UK, Nov. 1998.

Bregmann, A. Auditory Scene Analysis. Cambridge, MA: MIT Press, 1990.

Childs, E. Achorripsis: A sonification of probability distributions. Proc. of the ICAD 2002, 2002.

Ersbøll, B.K. and Conradsen, K. An introduction to statistics: volume 2. 6. edition, DTU-tryk, Kgs. Lyngby, 2003.

Fawcett, T. ROC Graphs: Notes and Practical Considerations for Data Mining Researchers. Intelligent Enterprise Technologies Laboratory, HP Laboratories Palo Alto, HPL-2003-4, California, Jan. 2003.

Fernström, J.M. and McNamara, C. After Direct Manipulation – Direct Sonification. ICAD 1998, Glasgow, Scotland, 1998.

Flowers, J.H. Thirteen Years of Reflection on the Auditory Graphing: Promises, Pitfalls, and Potential New Directions. Proc. of the ICAD 2005, Limerick, Ireland, July 2005.

Gaver, W. W. Using and Creating Auditory Icons. In G. Kramer (Ed.), Auditory Display: Sonification, Audification, and Auditory Interfaces, Addison-Wesley, pp. 417-46, 1994.

Hartmann, W.M. Signals, Sound, and Sensations. Springer, New York, 1998.

Hermann, T. Sonification for Exploratory Data Analysis. Ph.D. Dissertation at Bielefeld University, Germany ,Aug. 2002.

Hermann, T. and Ritter, H. Listen to your Data: Model-Based Sonification for Data Analysis. Proc. of the ISIMADE '99, Baden-Baden, Germany, 1999.

Hermann, T., Meinicke, P., Bekel, H., Müller, H.M., Weiss, S. and Ritter, H. Sonifications for EEG Data Analysis. Proc. of the ICAD 2002, Kyoto, Japan, July 2002.

Herrmann, C.S., Lenz, D., Junge, S., Busch, N.A. and Maess, B. Memory-matches Evoke Human Gamma-responses. BMC Neuroscience 5:13, 2004.

Hinterberger, T. and Baier, G. Parametric Orchestral Sonification of EEG in Real Time. Proc. of the International Workshop on Interactive Sonification, Bielefeld, Germany, Jan. 2004.

Hinterberger, T., Baier, G., Mellinger, J., and Birbaumer, N. Auditory Feedback of Human EEG for Direct Brain-Computer Communication. Proc. of the ICAD 2004, Sydney, Australia, July 2004.

Hooper, G. EEG Sonification. Proc. of the ICAD 2004, Sydney, Australia, July 2004.

Jensen, K. The Timbre Model. Workshop on current research directions in computer music, Barcelona, Spain, 2001.

Jovanov, E., Starcevic, D., Marsh, D., Obrenovic, Z., Radivojevic, V. and Samardzic A.Multi modal presentation in virtual telemedical environments, High-Performance Computing and Networking, Proceedings Lecture Notes In Computer Science, 1593: 964-972 1999.

Jung, T., Makeig, S., Westerfield, M., Townsend, J., Courchesne, E., and Sejnowski, T.J. Removal of eye activity artifacts from visual event-related potentials in normal and clinical subjects. Clinical Neurophysiology 111, pp. 1745-1758, 2000.

Jung, T., Makeig, S., Westerfield, M., Townsend, J., Courchesne, E., and Sejnowski, T.J. Analysis and Visualization of Single-Trial Event-Related Potentials. Wiley-Liss Inc., Human Brain Mapping 14:166-185, 2001.

Kennedy, R.L., Lee, Y, Van Roy, B, Reed, C.D. and Lippmann, R.P. Solving Data Mining Problems through Pattern Recognition. Prentice Hall, PTR, New Jersey, 1995.

Knight, J.N. Signal Fraction Analysis and Artifact Removal in EEG. Masters Thesis, Colorado State University, 2004

Kolenda, T. Independent Component Analysis, Master Thesis, Department of Mathematical Modeling, Technical University of Denmark, 1998.

Kramer, G. Auditory Display - Sonification, Audification, and Auditory Interfaces, Proceedings Volume, Addison Wesley, 1994.

Kramer, G., Walker, B., Bonebright, T., Cook, P. Flowers, J., Miner, N. and Neuhoff, J. Sonification Report: Status of the field and research agenda. Technical Report, ICAD, 1999.

Kronland-Martinet, R., Guillemain, P. and Ystad, S. The Timbre Model – Discrimination and Expression. Mosart Deliverabel D22, Evaluation report of Timbre modeling, 2002.

Kwak, N., and Choi, C.H. Feature extraction based on ICA for Binary Classifications Problems. IEEE Trans. On Knowledge and Data Engineering, 2003.

Lokki, T. Aurilization of Simulated Impulse Responses in Slow Motion. AES 118th Convention Paper 6500, Barcelona, Spain, May 2005.

Makeig, S., Jung, T.P., Ghahremani, D., Bell, A.J. and Sejnowski, T.J. Blind separation of event-related brain responses into independent components. Proc. Natl. Acad. Sci. USA, 94:10979-10984, 1997.

Marentakis, G. and K. Jensen, Sinusoidal Synthesis Optimization, Proc. of the ICMC, Göteborg, Sweden 2002.

Mayer-Kress, G. Sonification of Multiple Electrode Human Scalp Electroencephalogram. Proc. of the ICAD, 1994.

Meinicke, P., Hermann, T., Bekel, H., Müller, H.M., Weiss, S. and Ritter, H. Identification of Discriminative features in the EEG. Intelligent Data Analysis 8, IOS Press, pp. 97-107, 2004.

Moore, B.C.J. An Introduction to the Psychology of Hearing. 4. edition Academic Press, London, 1997.

Mørup, M. Analysis of Brain Data - Using Multi-Way Array Models on the EEG, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2005

Nuehoff, J.G., McBeath, M.K. and Wanzie, W. C. Dynamic Frequency Change Influences Loudness Perception: A Central, Analytical Process. Journal of Experimental Psychology: Human Perception and Performance, Vol 25(4), pp 1050-1059, 1999.

Patterson, R. Guidelines for Auditory Warning Systems on Civil Aircraft. Civil Aviation Authority Paper 82017, 1982.

Poulsen, T. Ear, Hearing and Speech: A short introduction. Version 1.2, Ørsted-DTU, Lyngby, 2001.

Roads, C. The Computer Music Tutorial. The MIT Press. Massachusetts, USA, 1996.

Truax, B. Real-Time Granular Synthesis with a Digital Signal Processor. Computer Music Journal, vol. 12, no. 2, pp. 14-26, 1988.

Walker, B. and Ehrenstein . Congruency Effects with Dynamic Auditory Stimuli: Design Implications. Proc. of ICAD, Palo Alto, California, Nov. 1997.

Walker, B. and Kramer, G. Human Factors and the Acoustic Ecology: Considerations for Multimedia Audio Design. Proc. of the ICAD, Palo Alto, California, Nov. 1996.

Williamson, J. and Murray-Smith, R. Granular Synthesis for Display of Time-Varying Probability Densities. Proc. of the International Workshop on Interactive Sonification, Bielefeld, Jan. 2004.

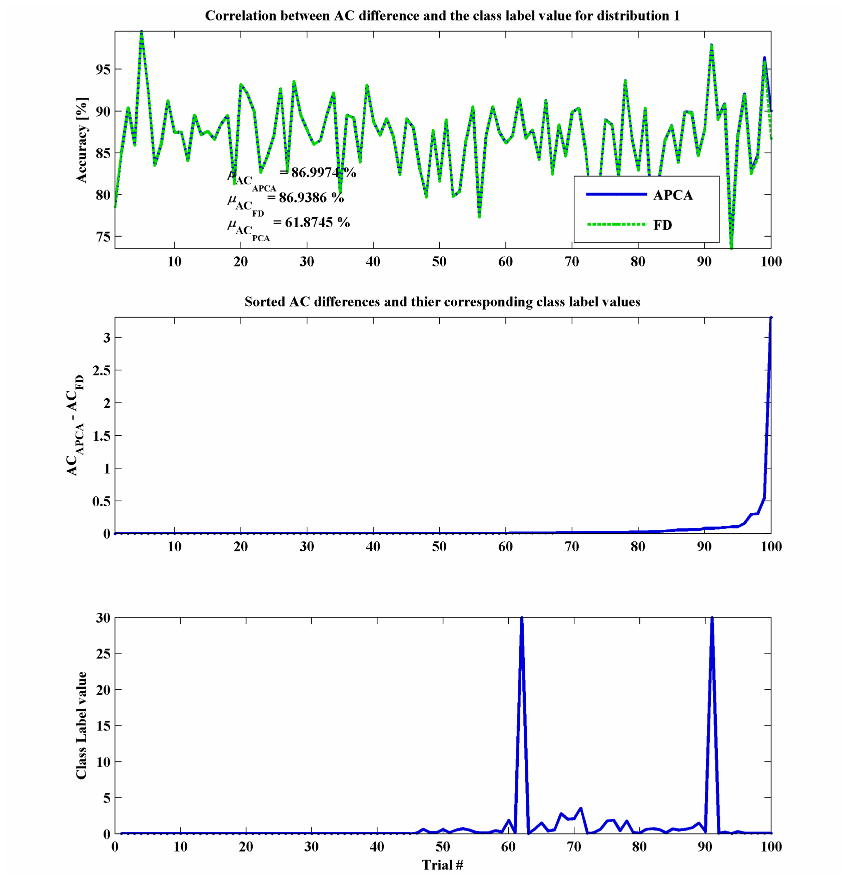
Williamson, J. and Murray-Smith, R. Audio Feedback with Gesture Recognition. Technical Report TR-2002-127, Department of Computer Science, University of Glasgow, UK, Dec. 2002.

Woestenburg, J.C., Verbaten, M.N. and Slangen, J.L. The Removal of the Eye-Movement Artifact From the EEG by Regression Analysis in the Frequency Domain. Biol Psychol 16:127-147, 1983

Zwicker, E. and Fastl. H. Psychoacoustics: Facts and Models. 2. edition, Springer, Berlin, 1999.

A note to the interested reader: ICAD papers are not published in a collected form but as individual papers and therefore no page numbers are given.

Appendix A: Hotelling's T^2 test performed accuracy differences between APCA and PCA for 100 trails of distribution 1 and 2.

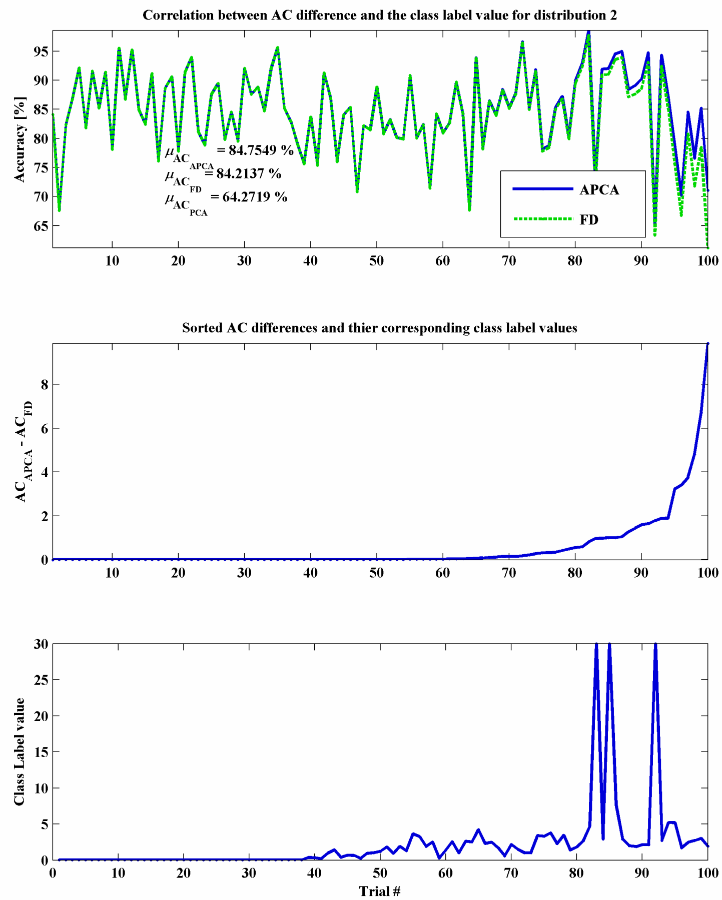


Results for distribution 1 presented as sorted by there accuracy differences

A test for the mean value of the CCP differences to equal zero was preformed. Hotelling's T2 was used, where the test hypothesis, about the mean vector $\boldsymbol{\mu}$ being equal to a given vector $\boldsymbol{\mu}_0$, is

$$H_0 : \mu = \mu_0 = 0$$

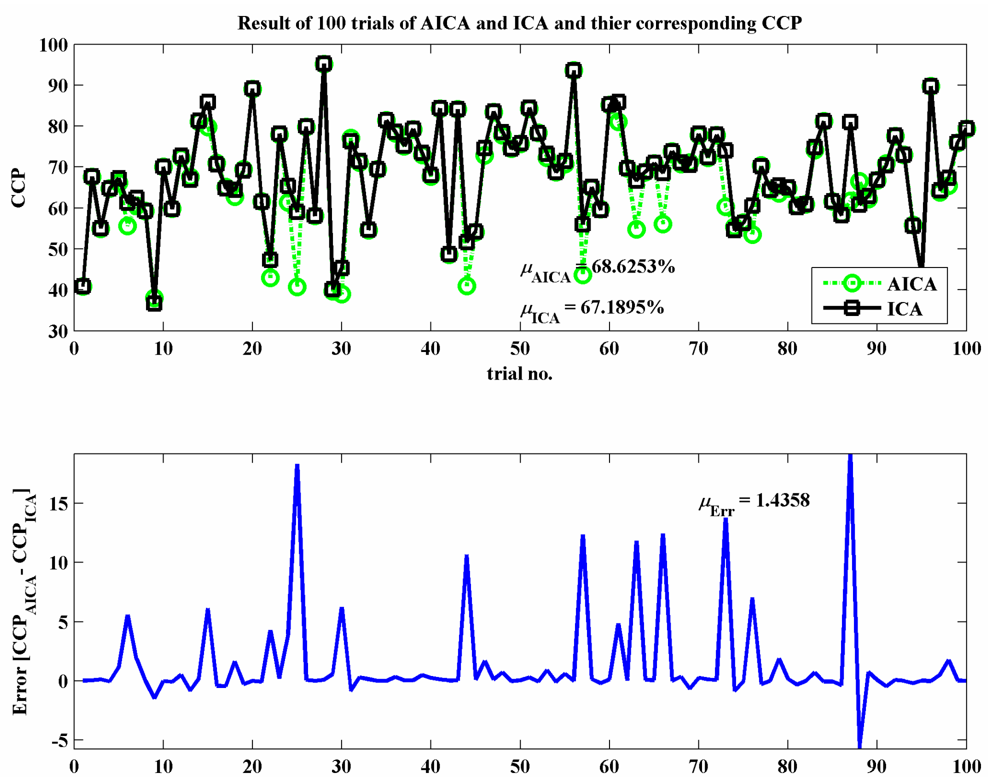
The results were $3.061 > F(1,99)_{0.999} = 6.9$, thus the null hypothesis is not rejected for levels larger than 0.1%.



Results for distribution 2 presented as sorted by there accuracy differnces

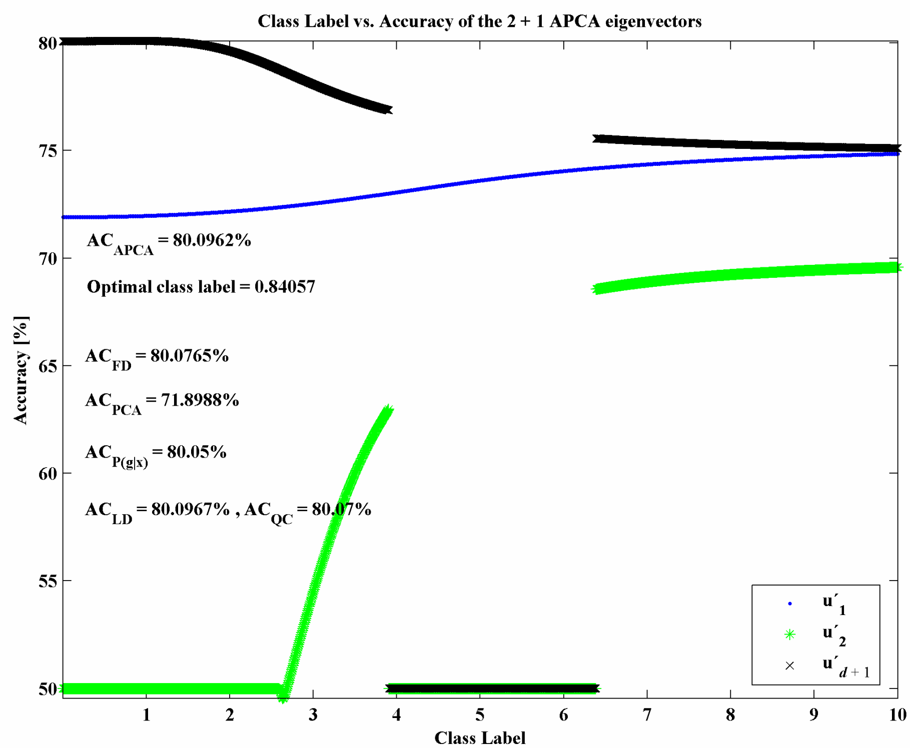
Here the results were $14.5559 > F(1,99)_{0.999} = 6.9$, thus the null hypothesis is rejected for levels larger than 0.1%. This shows that APCA might have some advantage over FD for distributions with non alike covariance matrices.

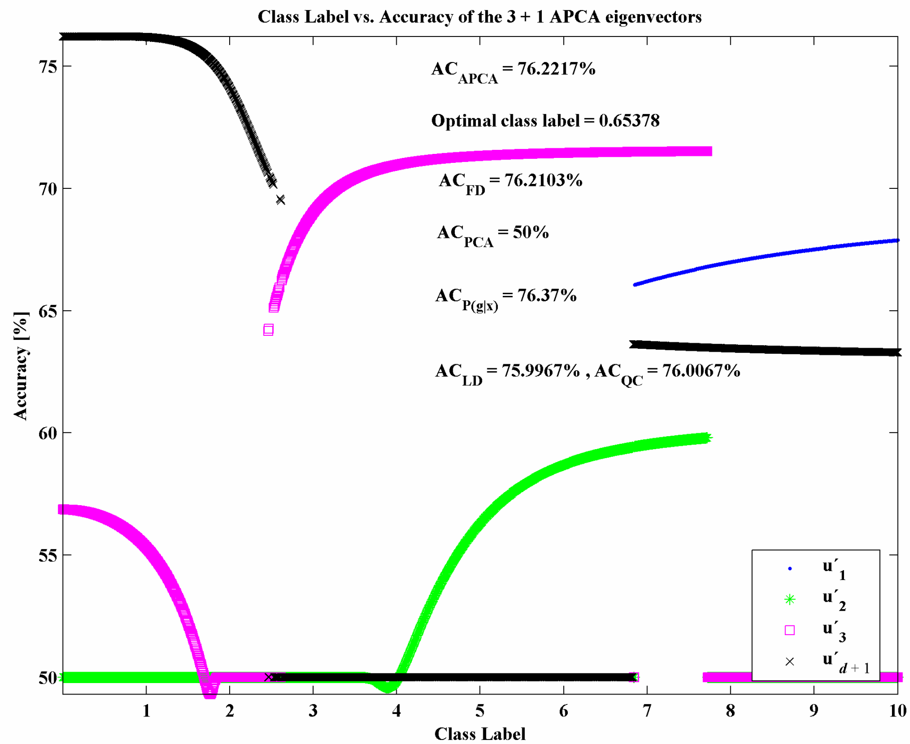
Appendix B: Hotelling's T^2 test performed accuracy differences between AICA and ICA for 100 trails of data type 1 and 2.

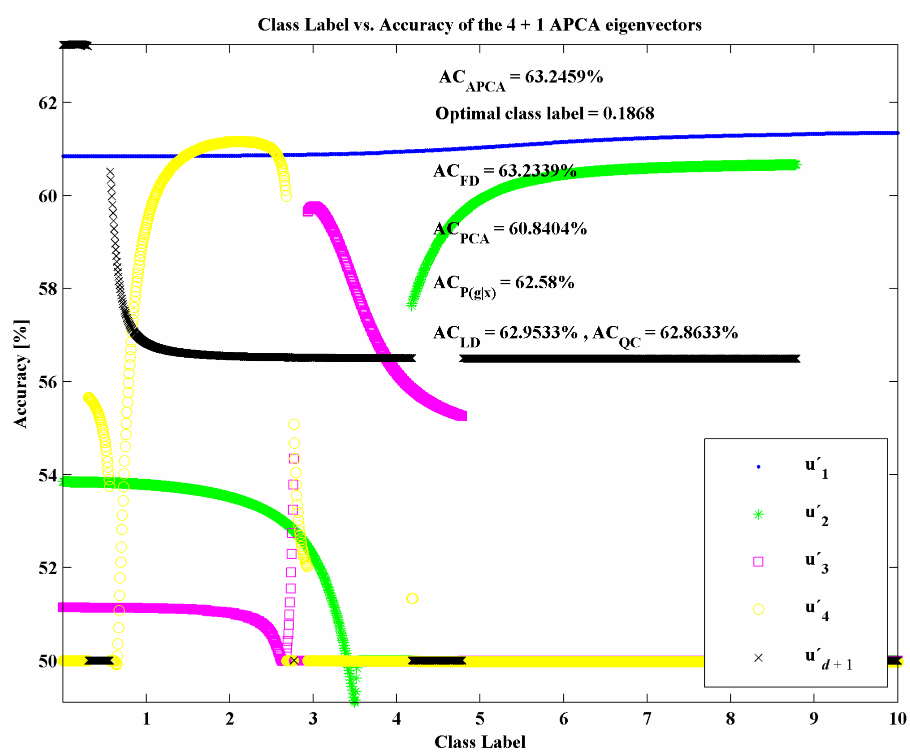


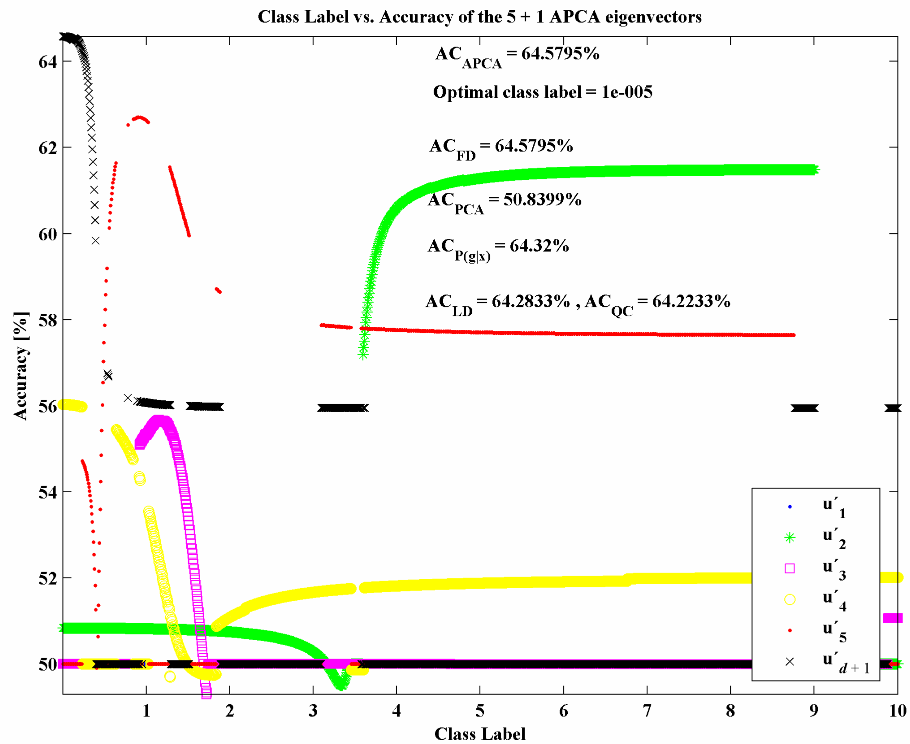
Here the results were 15.2651 > $F(1,99)_{0.999}$ = 6.9, thus the null hypothesis is rejected for levels larger than 0.1%. This shows that AICA might have some advantage over ICA for distribution 2 – data with non-alike covariance matrices.

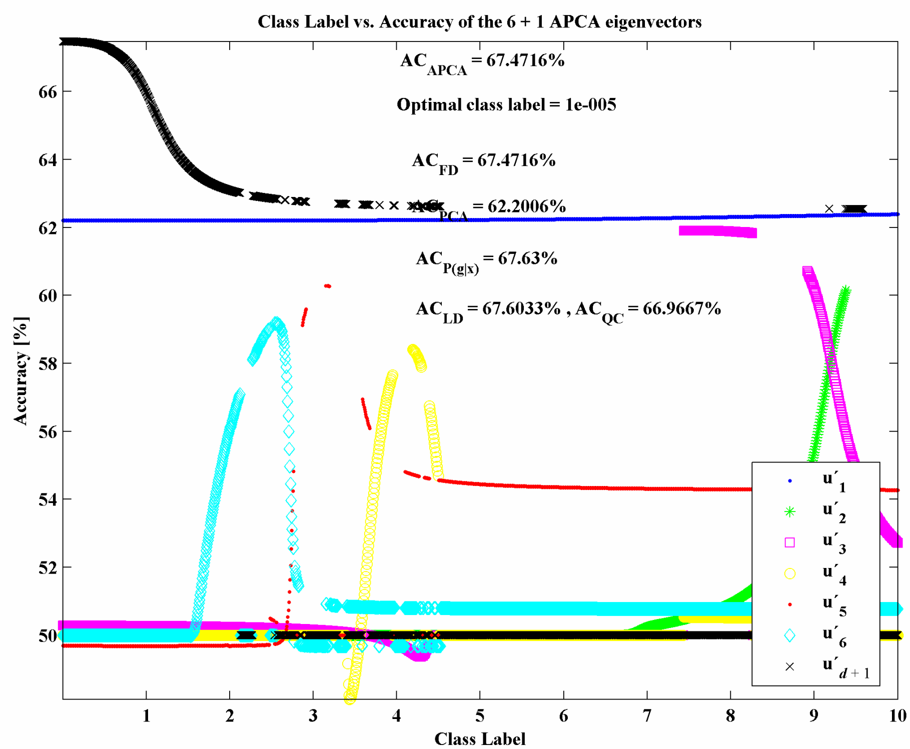
Appendix C1: Accuracy of APCA eigenvectors for higher dimensions as a function of class label.

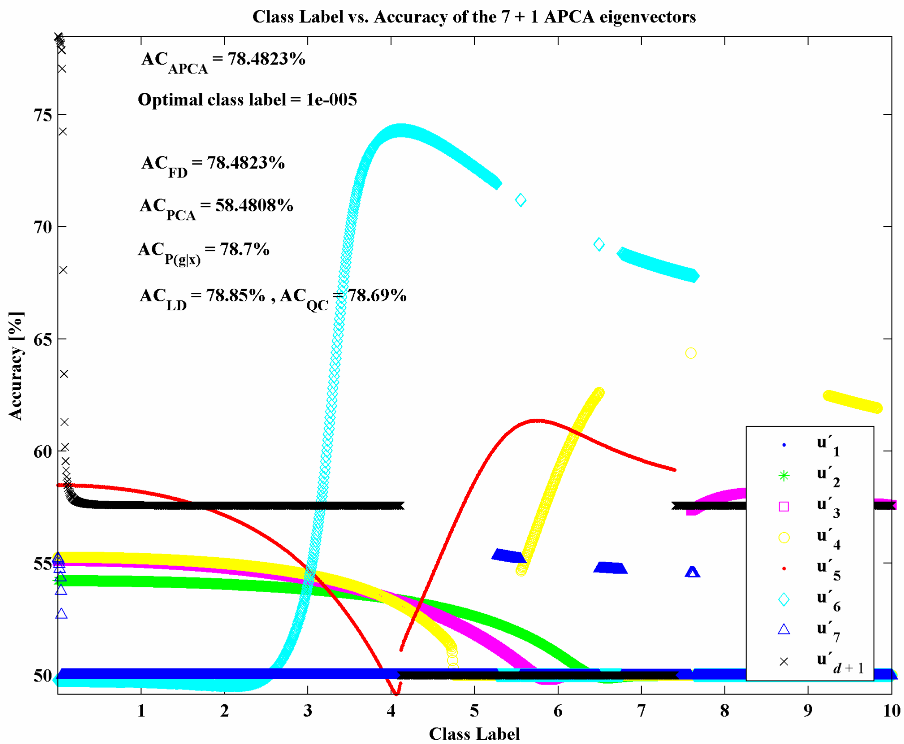



Class Label vs. Accuracy of the 3 + 1 APCA eigenvectors
AC_{APCA} = 76.2217%
Optimal class label = 0.65378
AC_{FD} = 76.2103%
AC_{PCA} = 50%
$AC_{P(g|x)}$ = 76.37%
AC_{LD} = 75.9967% , AC_{QC} = 76.0067%
Accuracy [%]
Class Label
u'_1
u'_2
u'_3
u'_{d+1}

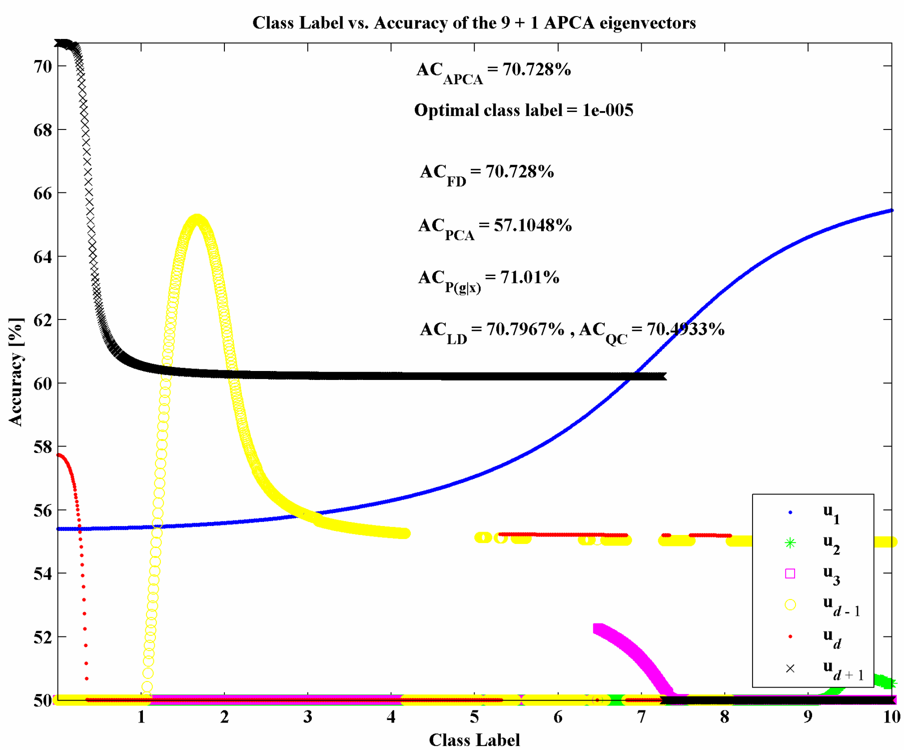

Class Label vs. Accuracy of the 4 + 1 APCA eigenvectors
AC_{APCA} = 63.2459%
Optimal class label = 0.1868
AC_{FD} = 63.2339%
AC_{PCA} = 60.8404%
$AC_{P(g|x)}$ = 62.58%
AC_{LD} = 62.9533% , AC_{QC} = 62.8633%
Accuracy [%]
Class Label
u'_1
u'_2
u'_3
u'_4
u'_{d+1}


Class Label vs. Accuracy of the 5 + 1 APCA eigenvectors
AC_{APCA} = 64.5795%
Optimal class label = 1e-005
AC_{FD} = 64.5795%
AC_{PCA} = 50.8399%
$AC_{P(g|x)}$ = 64.32%
AC_{LD} = 64.2833% , AC_{QC} = 64.2233%
Accuracy [%]
Class Label
u'_1
u'_2
u'_3
u'_4
u'_5
u'_{d+1}

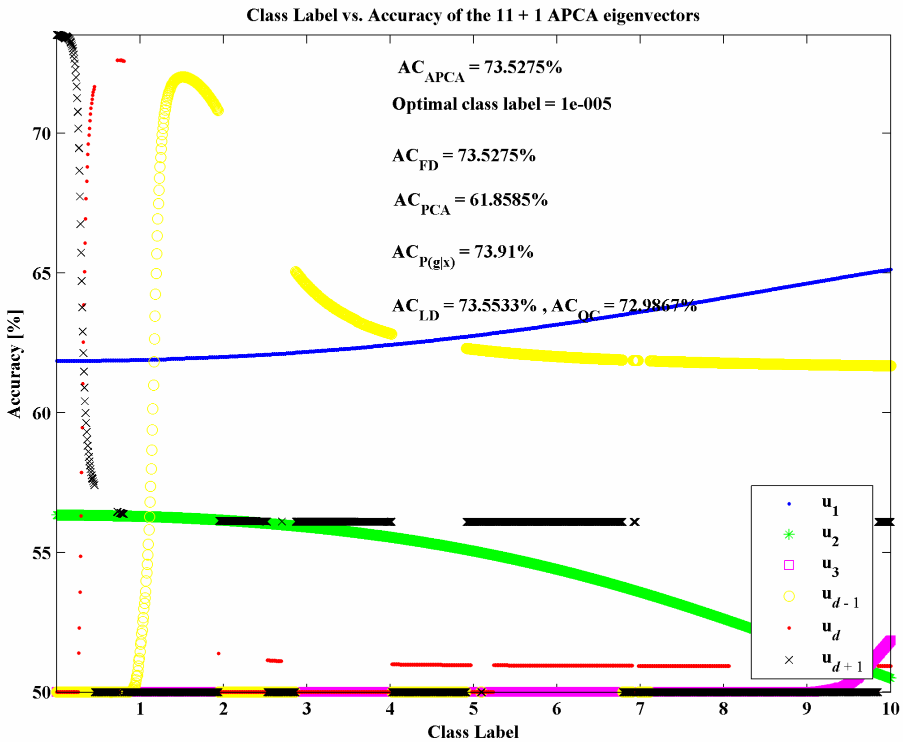

Class Label vs. Accuracy of the 6 + 1 APCA eigenvectors
AC_{APCA} = 67.4716%
Optimal class label = 1e-005
AC_{FD} = 67.4716%
AC_{PCA} = 62.2006%
$AC_{P(g|x)}$ = 67.63%
AC_{LD} = 67.6033% , AC_{QC} = 66.9667%
Accuracy [%]
Class Label
u'_1
u'_2
u'_3
u'_4
u'_5
u'_6
u'_{d+1}



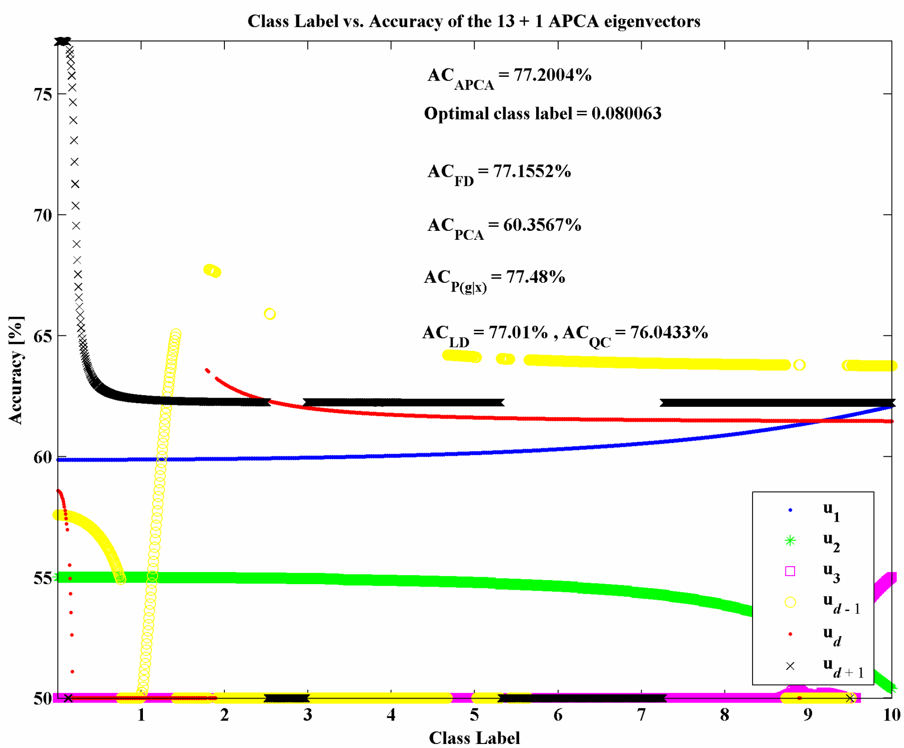
Class Label vs. Accuracy of the 7 + 1 APCA eigenvectors
AC_{APCA} = 78.4823%
Optimal class label = 1e-005
AC_{FD} = 78.4823%
AC_{PCA} = 58.4808%
$AC_{P(g|x)}$ = 78.7%
AC_{LD} = 78.85% , AC_{QC} = 78.69%
Accuracy [%]
Class Label
u'_1
u'_2
u'_3
u'_4
u'_5
u'_6
u'_7
u'_{d+1}



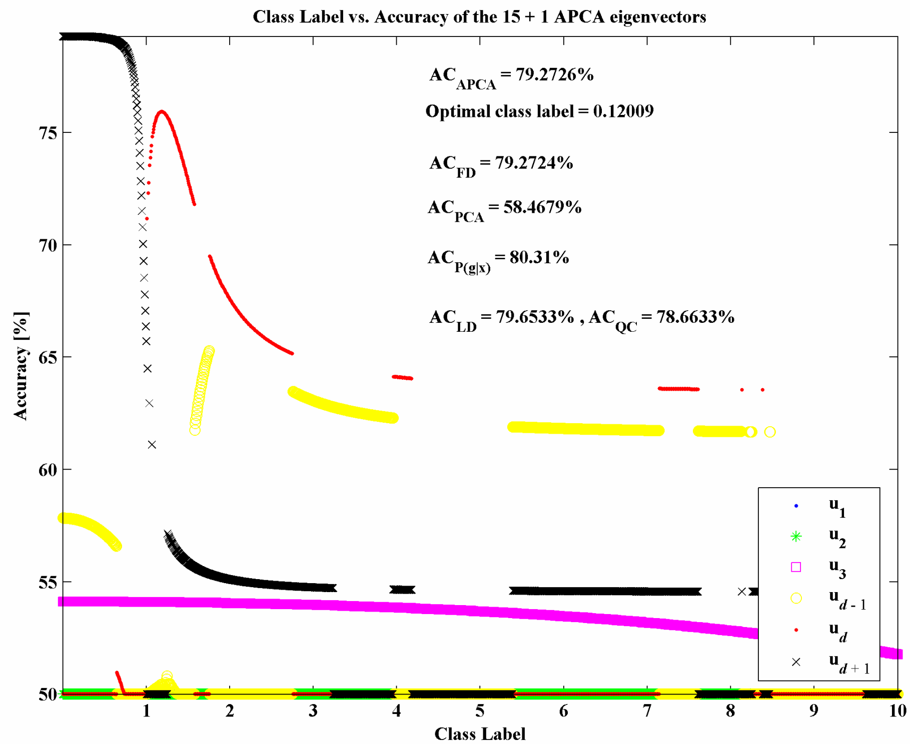
Class Label vs. Accuracy of the 9 + 1 APCA eigenvectors
AC_{APCA} = 70.728%
Optimal class label = 1e-005
AC_{FD} = 70.728%
AC_{PCA} = 57.1048%
$AC_{P(g|x)}$ = 71.01%
AC_{LD} = 70.7967% , AC_{QC} = 70.4933%
Accuracy [%]
Class Label
u_1
u_2
u_3
u_{d-1}
u_d
u_{d+1}

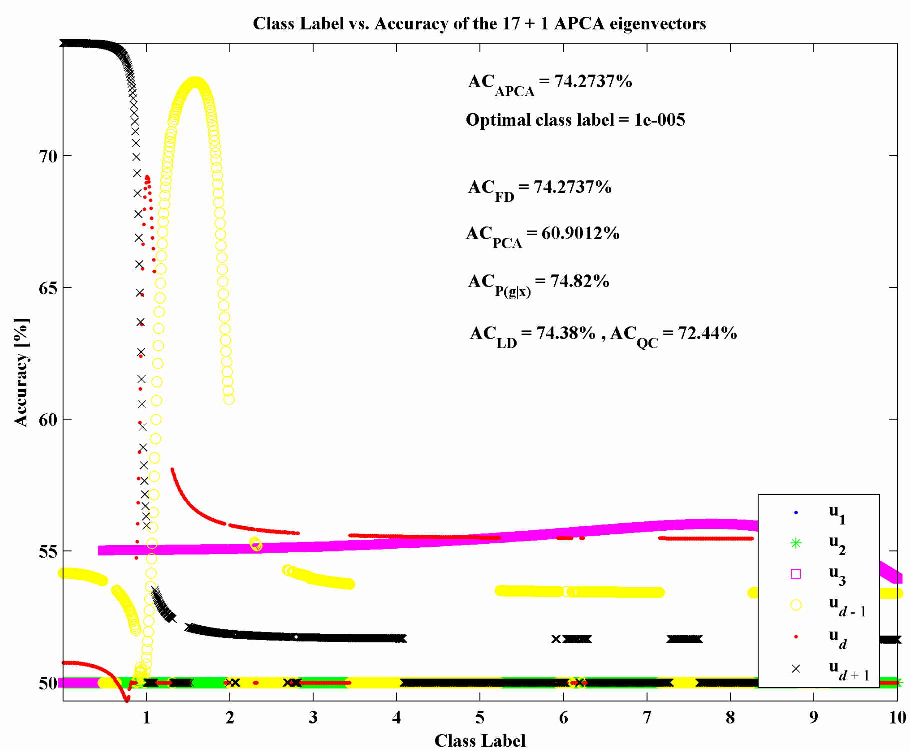


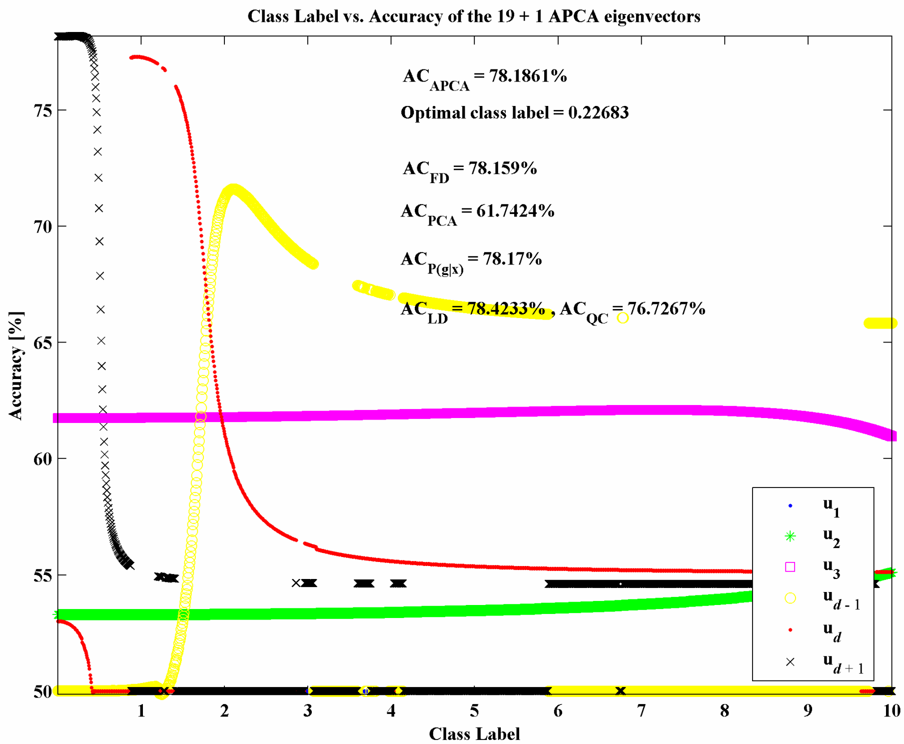
Class Label vs. Accuracy of the 11 + 1 APCA eigenvectors
AC_{APCA} = 73.5275%
Optimal class label = 1e-005
AC_{FD} = 73.5275%
AC_{PCA} = 61.8585%
$AC_{P(g|x)}$ = 73.91%
AC_{LD} = 73.5533% , AC_{QC} = 72.9867%
u_1
u_2
u_3
u_{d-1}
u_d
u_{d+1}
Accuracy [%]
Class Label



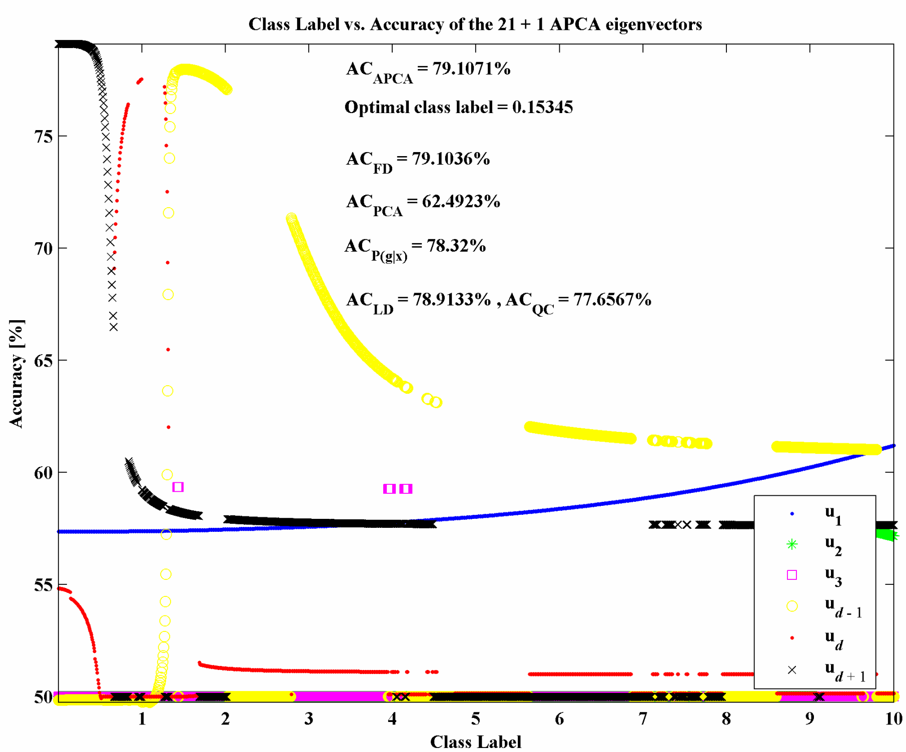
Class Label vs. Accuracy of the 13 + 1 APCA eigenvectors
AC_{APCA} = 77.2004%
Optimal class label = 0.080063
AC_{FD} = 77.1552%
AC_{PCA} = 60.3567%
$AC_{P(g|x)}$ = 77.48%
AC_{LD} = 77.01% , AC_{QC} = 76.0433%
u_1
u_2
u_3
u_{d-1}
u_d
u_{d+1}
Accuracy [%]
Class Label


Class Label vs. Accuracy of the 15 + 1 APCA eigenvectors
AC_APCA = 79.2726%
Optimal class label = 0.12009
AC_FD = 79.2724%
AC_PCA = 58.4679%
AC_P(g|x) = 80.31%
AC_LD = 79.6533% , AC_QC = 78.6633%
u_1
u_2
u_3
u_d-1
u_d
u_d+1
Accuracy [%]
Class Label

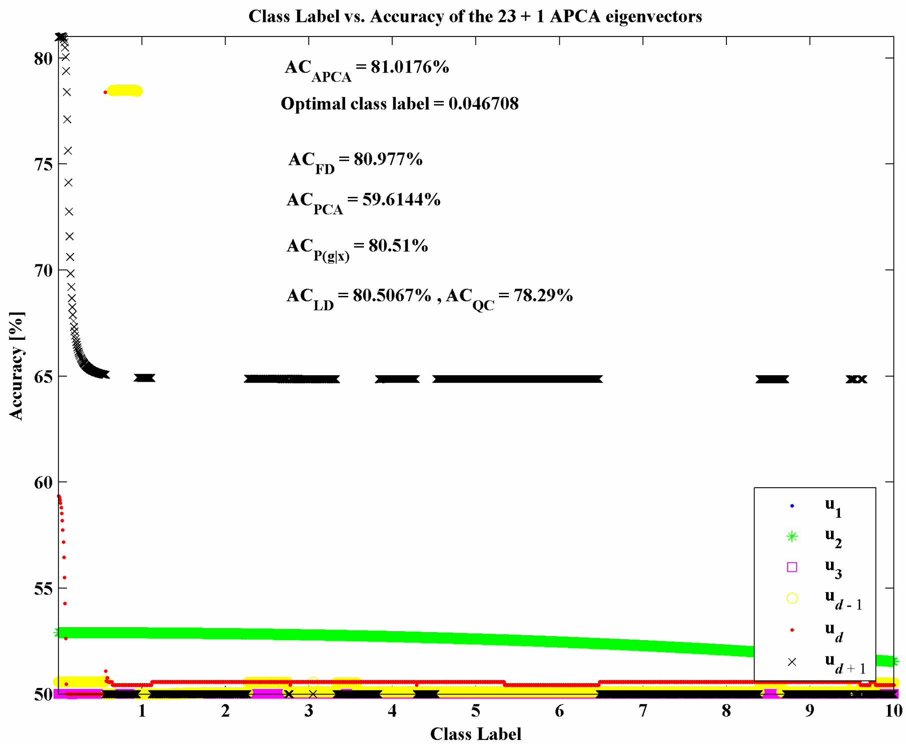

Class Label vs. Accuracy of the 17 + 1 APCA eigenvectors
AC_APCA = 74.2737%
Optimal class label = 1e-005
AC_FD = 74.2737%
AC_PCA = 60.9012%
AC_P(g|x) = 74.82%
AC_LD = 74.38% , AC_QC = 72.44%
u_1
u_2
u_3
u_d-1
u_d
u_d+1
Accuracy [%]
Class Label

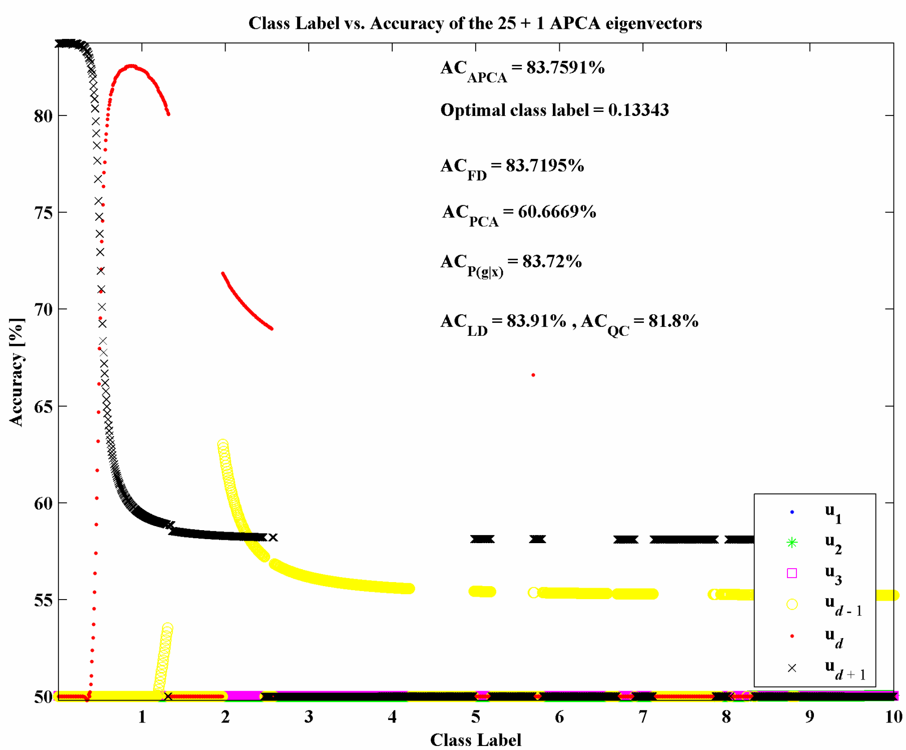


Class Label vs. Accuracy of the 19 + 1 APCA eigenvectors
AC_APCA = 78.1861%
Optimal class label = 0.22683
AC_FD = 78.159%
AC_PCA = 61.7424%
AC_P(g|x) = 78.17%
AC_LD = 78.4233% , AC_QC = 76.7267%
u1
u2
u3
ud-1
ud
ud+1
Accuracy [%]
Class Label

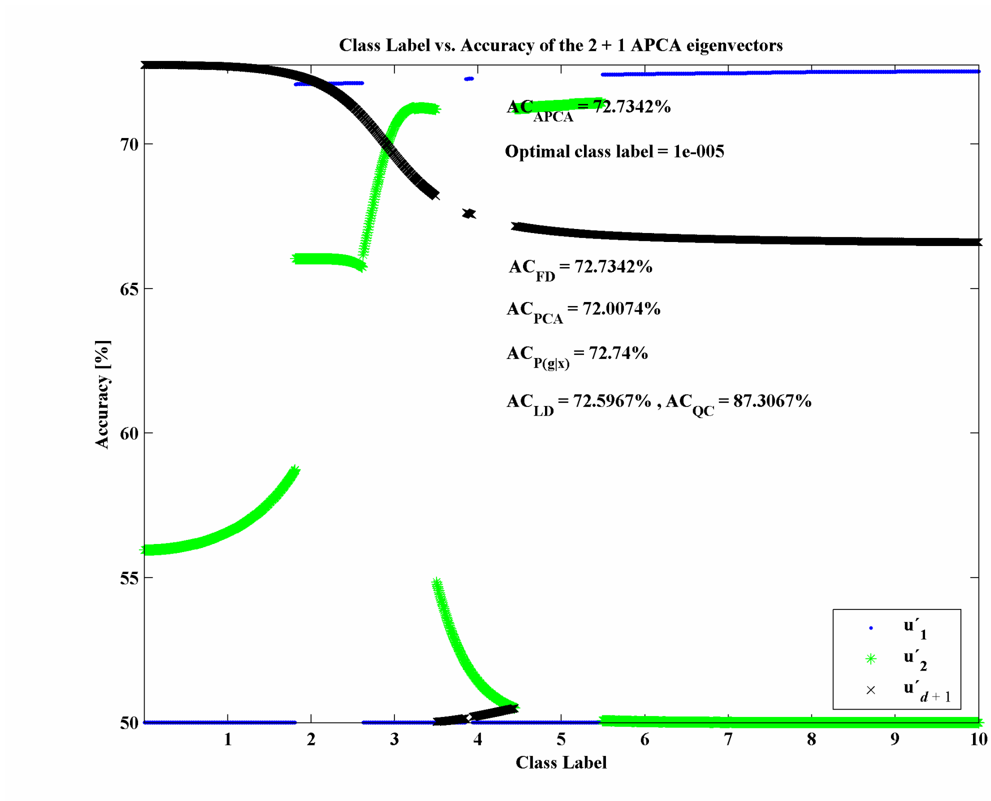


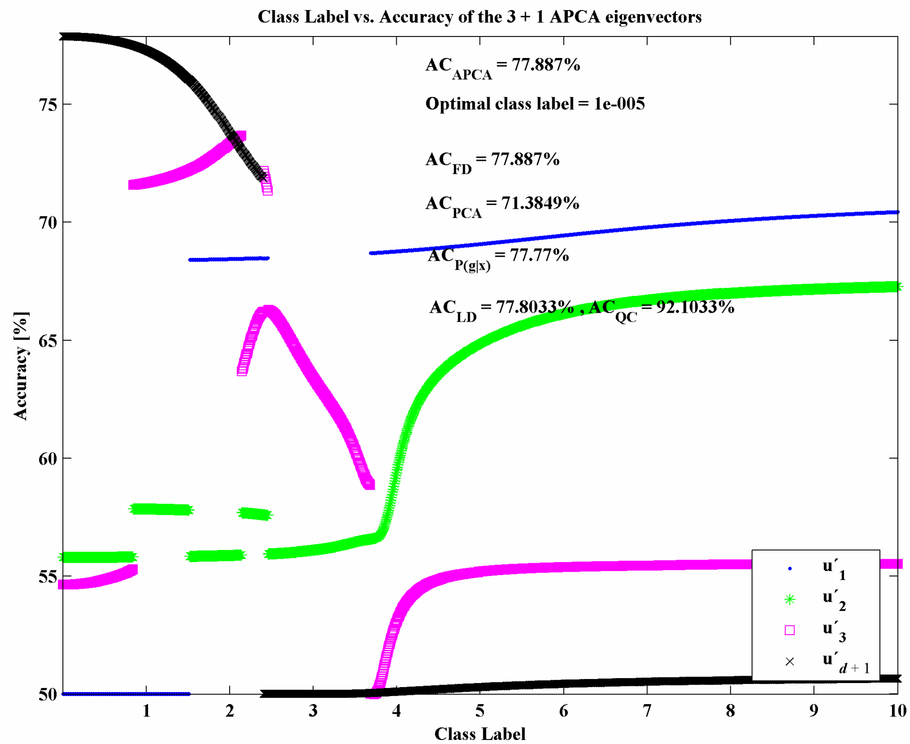
Class Label vs. Accuracy of the 21 + 1 APCA eigenvectors
AC_APCA = 79.1071%
Optimal class label = 0.15345
AC_FD = 79.1036%
AC_PCA = 62.4923%
AC_P(g|x) = 78.32%
AC_LD = 78.9133% , AC_QC = 77.6567%
u1
u2
u3
ud-1
ud
ud+1
Accuracy [%]
Class Label

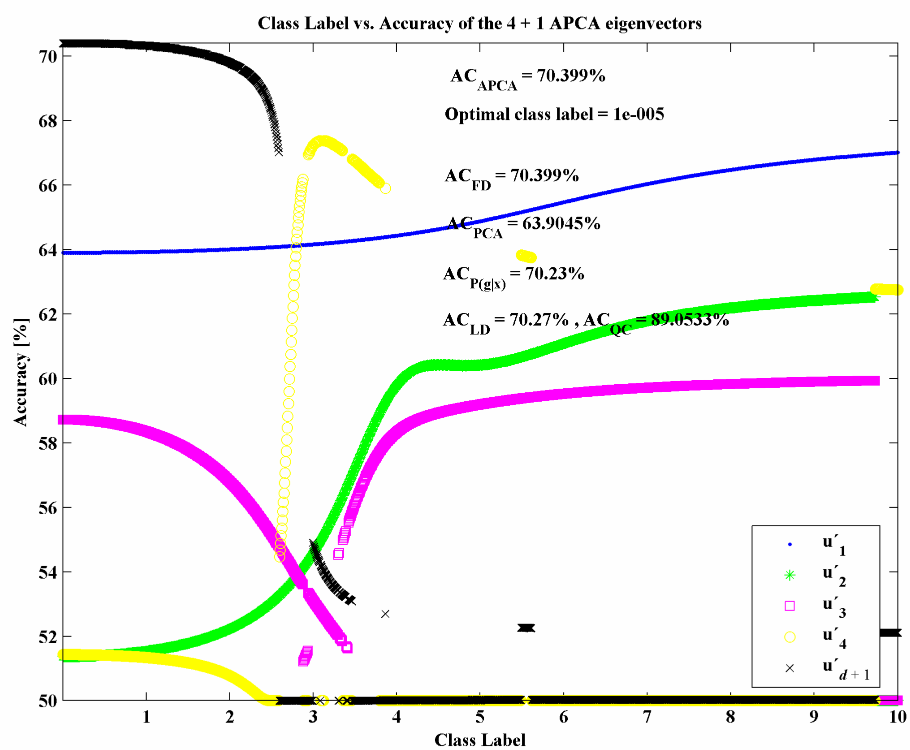

Class Label vs. Accuracy of the 23 + 1 APCA eigenvectors
AC_{APCA} = 81.0176%
Optimal class label = 0.046708
AC_{FD} = 80.977%
AC_{PCA} = 59.6144%
$AC_{P(g|x)}$ = 80.51%
AC_{LD} = 80.5067% , AC_{QC} = 78.29%
Accuracy [%]
Class Label
u_1
u_2
u_3
u_{d-1}
u_d
u_{d+1}

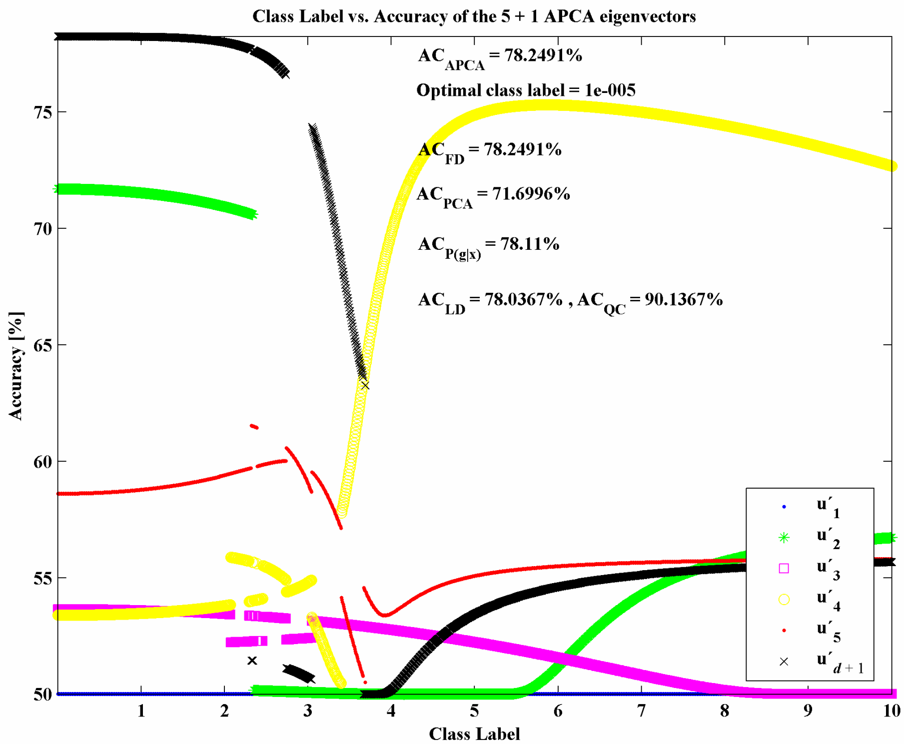

Class Label vs. Accuracy of the 25 + 1 APCA eigenvectors
AC_{APCA} = 83.7591%
Optimal class label = 0.13343
AC_{FD} = 83.7195%
AC_{PCA} = 60.6669%
$AC_{P(g|x)}$ = 83.72%
AC_{LD} = 83.91% , AC_{QC} = 81.8%
Accuracy [%]
Class Label
u_1
u_2
u_3
u_{d-1}
u_d
u_{d+1}

Appendix C2: Accuracy of APCA eigenvectors for higher dimensions as a function of class label.

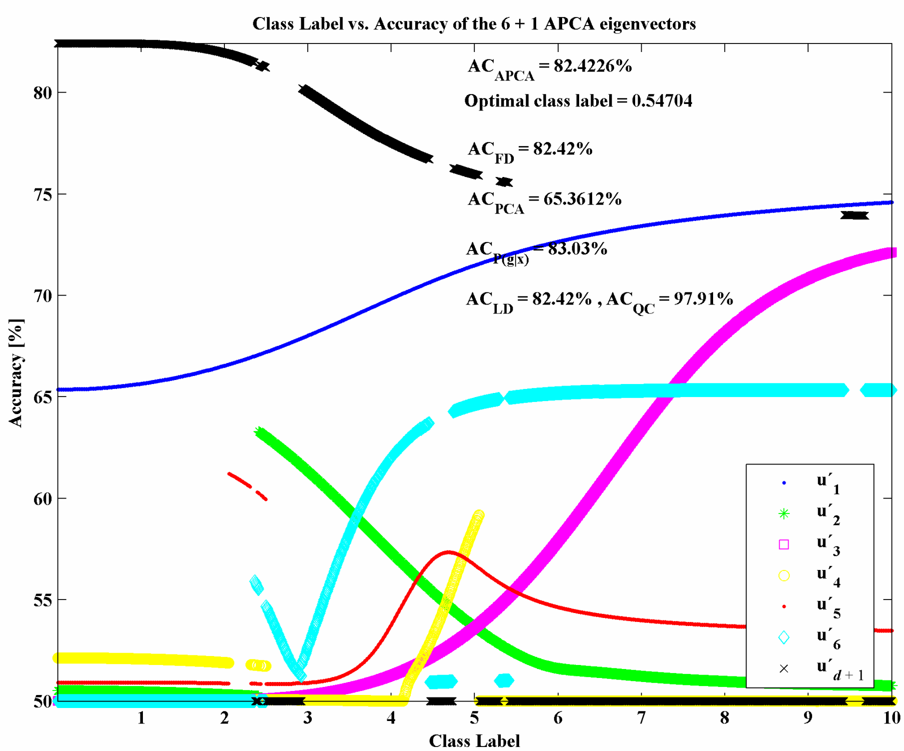



Class Label vs. Accuracy of the 3 + 1 APCA eigenvectors
AC_APCA = 77.887%
Optimal class label = 1e-005
AC_FD = 77.887%
AC_PCA = 71.3849%
AC_P(g|x) = 77.77%
AC_LD = 77.8033% , AC_QC = 92.1033%
Accuracy [%]
Class Label
u´1
u´2
u´3
u´d + 1

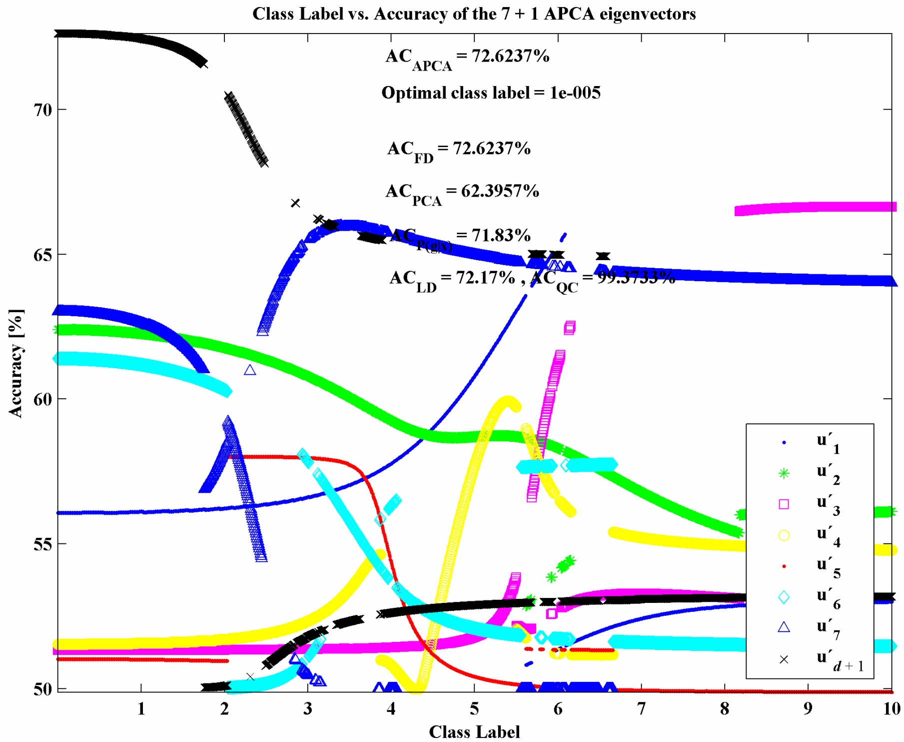

Class Label vs. Accuracy of the 4 + 1 APCA eigenvectors
AC_APCA = 70.399%
Optimal class label = 1e-005
AC_FD = 70.399%
AC_PCA = 63.9045%
AC_P(g|x) = 70.23%
AC_LD = 70.27% , AC_QC = 89.0533%
Accuracy [%]
Class Label
u´1
u´2
u´3
u´4
u´d + 1



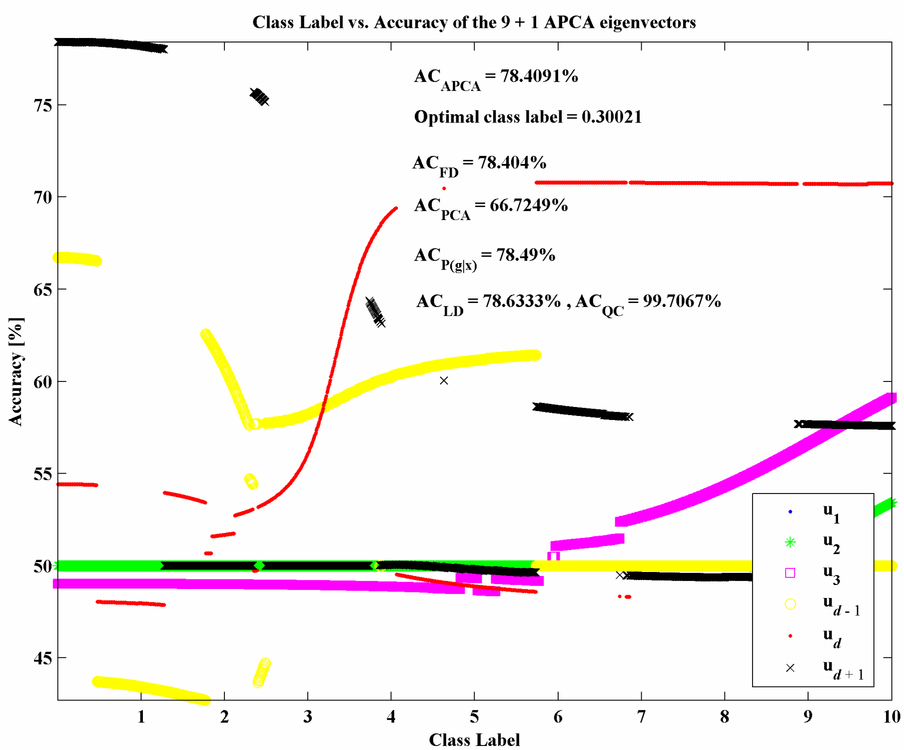
Class Label vs. Accuracy of the 5 + 1 APCA eigenvectors
AC_APCA = 78.2491%
Optimal class label = 1e-005
AC_FD = 78.2491%
AC_PCA = 71.6996%
AC_P(g|x) = 78.11%
AC_LD = 78.0367% , AC_QC = 90.1367%
Accuracy [%]
Class Label
u′1
u′2
u′3
u′4
u′5
u′d+1



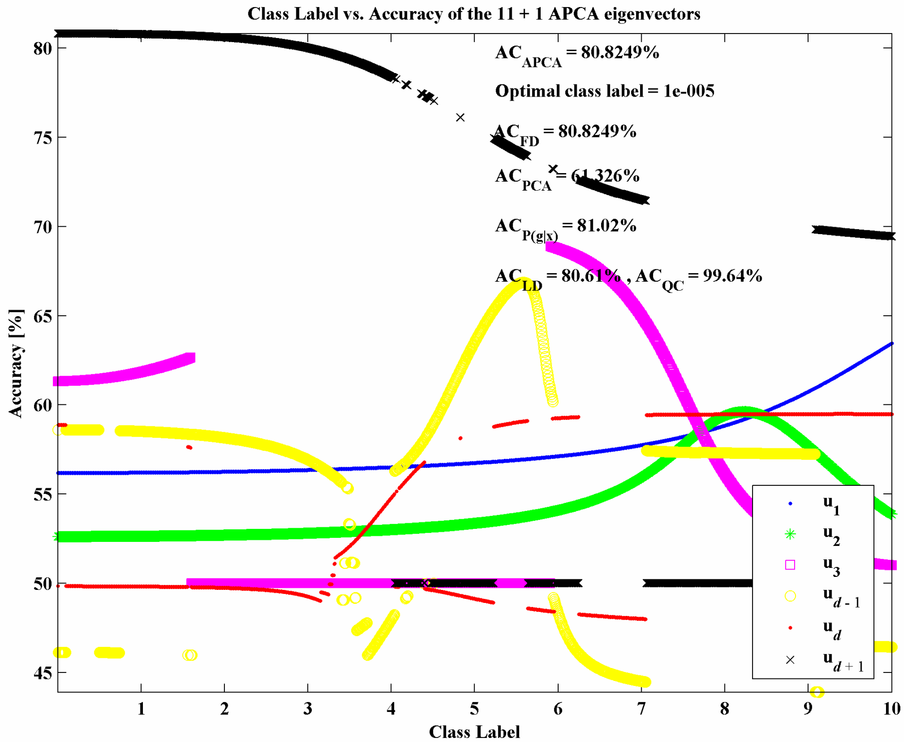
Class Label vs. Accuracy of the 6 + 1 APCA eigenvectors
AC_APCA = 82.4226%
Optimal class label = 0.54704
AC_FD = 82.42%
AC_PCA = 65.3612%
AC_P(g|x) = 83.03%
AC_LD = 82.42% , AC_QC = 97.91%
Accuracy [%]
Class Label
u′1
u′2
u′3
u′4
u′5
u′6
u′d+1

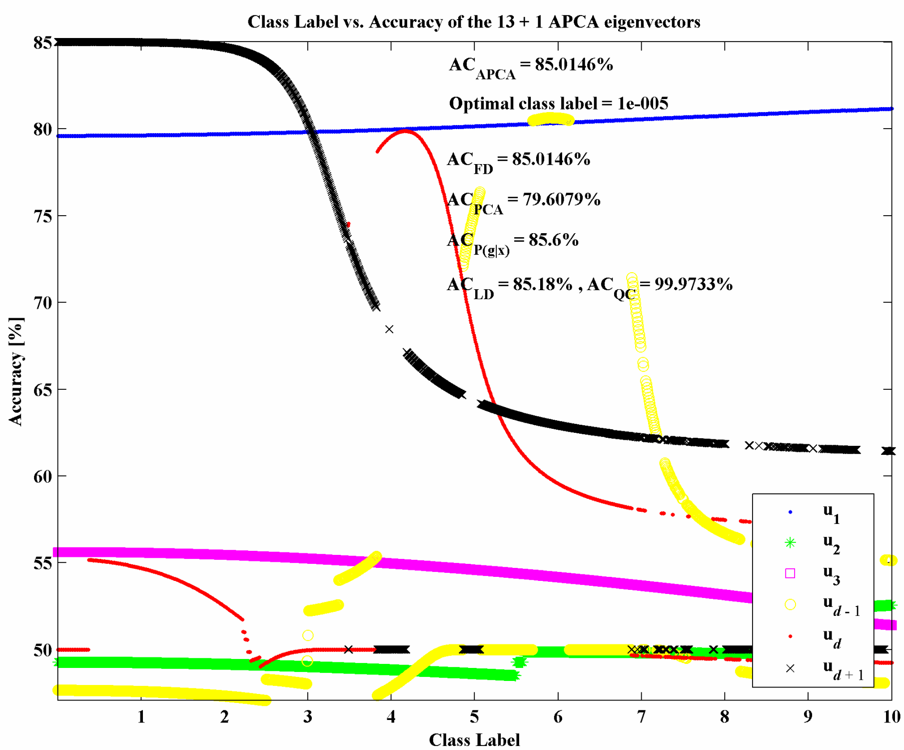


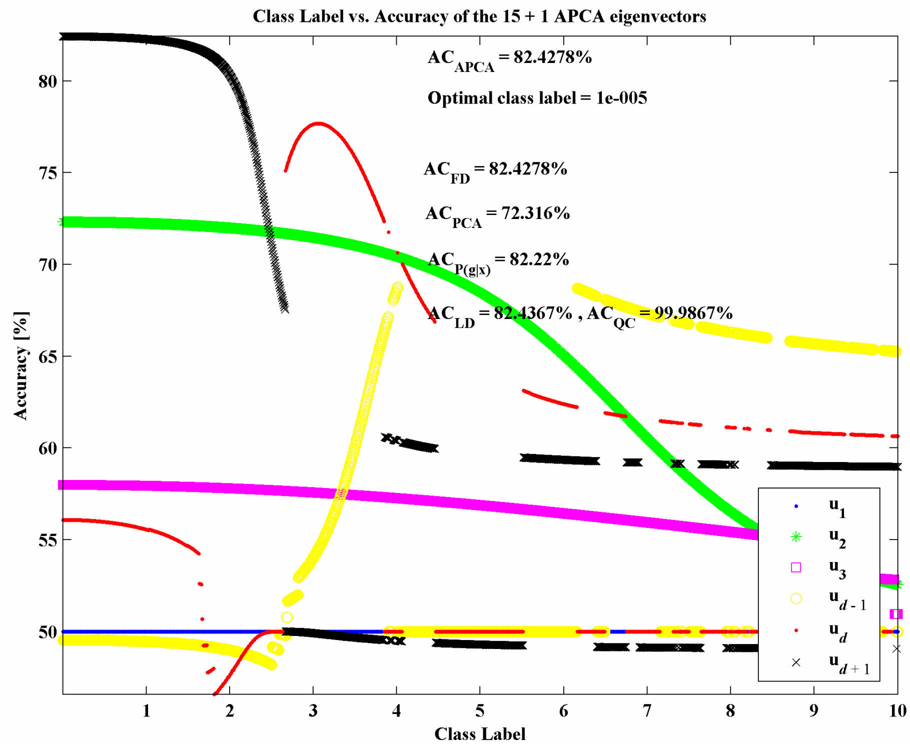
Class Label vs. Accuracy of the 7 + 1 APCA eigenvectors
AC_APCA = 72.6237%
Optimal class label = 1e-005
AC_FD = 72.6237%
AC_PCA = 62.3957%
AC_P(g|x) = 71.83%
AC_LD = 72.17% , AC_QC = 99.3733%
u′1
u′2
u′3
u′4
u′5
u′6
u′7
u′d + 1
Accuracy [%]
Class Label

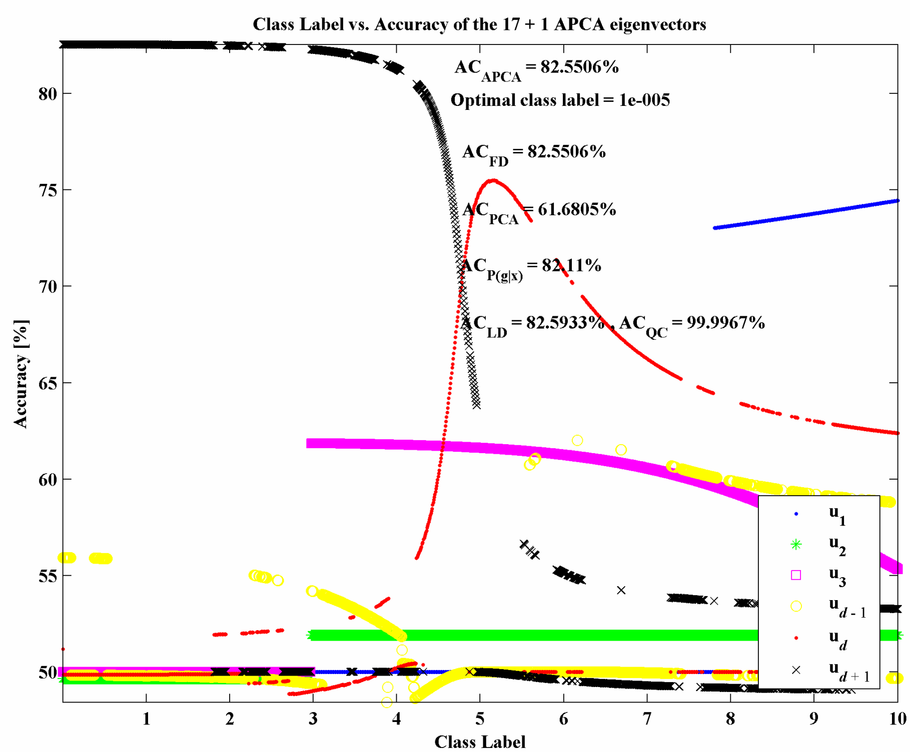


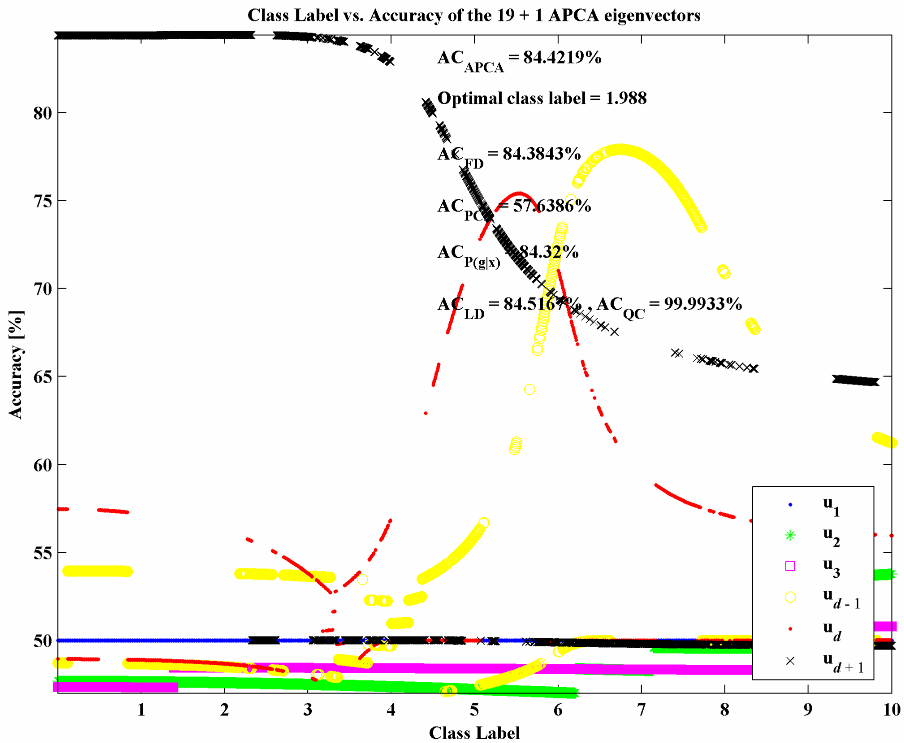
Class Label vs. Accuracy of the 9 + 1 APCA eigenvectors
AC_APCA = 78.4091%
Optimal class label = 0.30021
AC_FD = 78.404%
AC_PCA = 66.7249%
AC_P(g|x) = 78.49%
AC_LD = 78.6333% , AC_QC = 99.7067%
u1
u2
u3
ud - 1
ud
ud + 1
Accuracy [%]
Class Label


Class Label vs. Accuracy of the 11 + 1 APCA eigenvectors
AC_{APCA} = 80.8249%
Optimal class label = 1e-005
AC_{FD} = 80.8249%
AC_{PCA} = 61.326%
$AC_{P(g|x)}$ = 81.02%
AC_{LD} = 80.61% , AC_{QC} = 99.64%
Accuracy [%]
Class Label
u_1
u_2
u_3
u_{d-1}
u_d
u_{d+1}

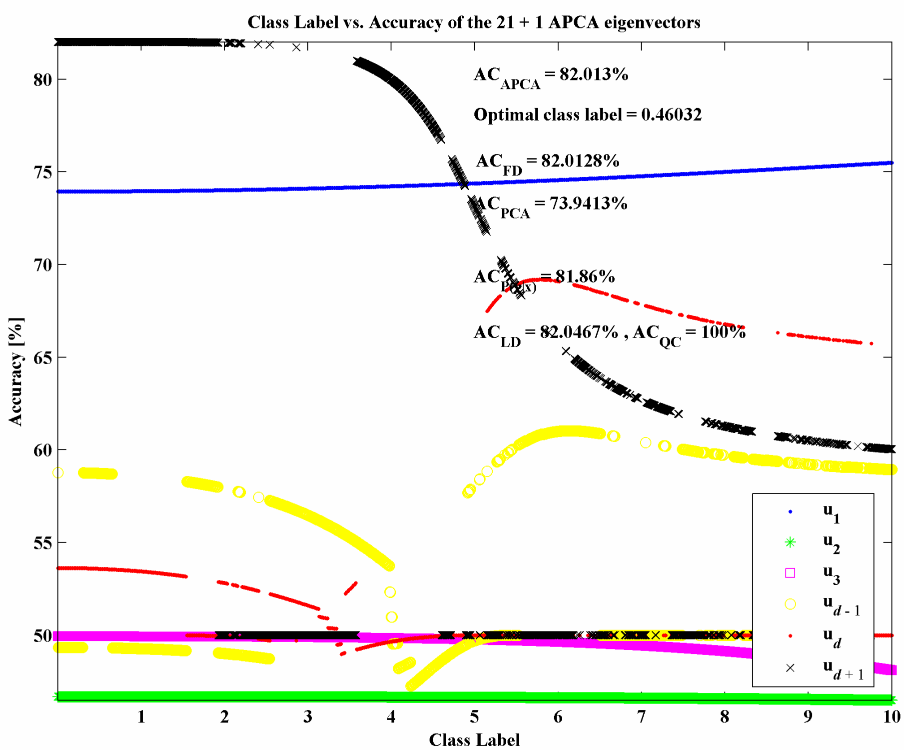

Class Label vs. Accuracy of the 13 + 1 APCA eigenvectors
AC_{APCA} = 85.0146%
Optimal class label = 1e-005
AC_{FD} = 85.0146%
AC_{PCA} = 79.6079%
$AC_{P(g|x)}$ = 85.6%
AC_{LD} = 85.18% , AC_{QC} = 99.9733%
Accuracy [%]
Class Label
u_1
u_2
u_3
u_{d-1}
u_d
u_{d+1}


Class Label vs. Accuracy of the 15 + 1 APCA eigenvectors
AC_{APCA} = 82.4278%
Optimal class label = 1e-005
AC_{FD} = 82.4278%
AC_{PCA} = 72.316%
$AC_{P(g|x)}$ = 82.22%
AC_{LD} = 82.4367% , AC_{QC} = 99.9867%
Accuracy [%]
Class Label
u_1
u_2
u_3
u_{d-1}
u_d
u_{d+1}

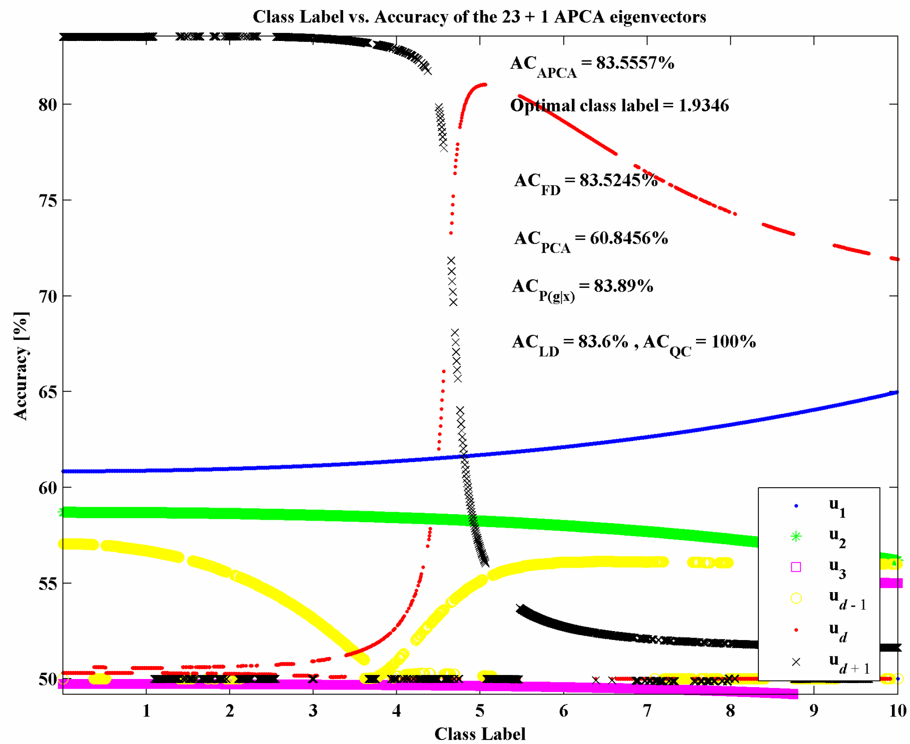

Class Label vs. Accuracy of the 17 + 1 APCA eigenvectors
AC_{APCA} = 82.5506%
Optimal class label = 1e-005
AC_{FD} = 82.5506%
AC_{PCA} = 61.6805%
$AC_{P(g|x)}$ = 82.11%
AC_{LD} = 82.5933% , AC_{QC} = 99.9967%
Accuracy [%]
Class Label
u_1
u_2
u_3
u_{d-1}
u_d
u_{d+1}

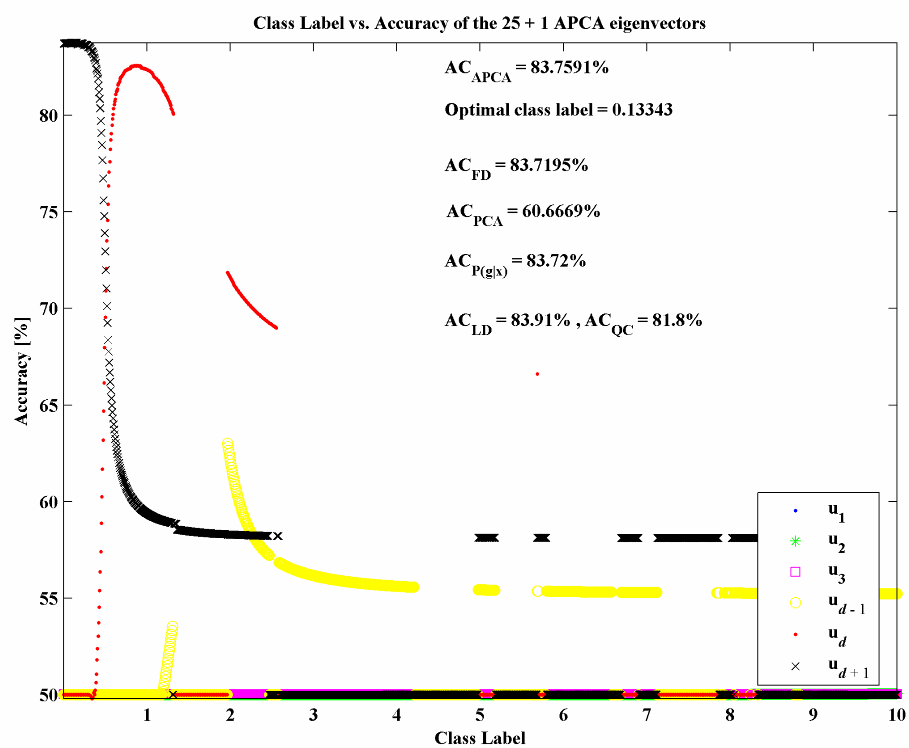


Class Label vs. Accuracy of the 19 + 1 APCA eigenvectors
AC_{APCA} = 84.4219%
Optimal class label = 1.988
AC_{FD} = 84.3843%
AC_{PCA} = 57.6386%
AC_{P(g|x)} = 84.32%
AC_{LD} = 84.5167% , AC_{QC} = 99.9933%
Accuracy [%]
Class Label
u1
u2
u3
ud - 1
ud
ud + 1



Class Label vs. Accuracy of the 21 + 1 APCA eigenvectors
AC APCA = 82.013%
Optimal class label = 0.46032
AC FD = 82.0128%
AC PCA = 73.9413%
AC P(g|x) = 81.86%
AC LD = 82.0467% , AC QC = 100%
Accuracy [%]
Class Label
u1
u2
u3
ud - 1
ud
ud + 1


Class Label vs. Accuracy of the 23 + 1 APCA eigenvectors
AC_APCA = 83.5557%
Optimal class label = 1.9346
AC_FD = 83.5245%
AC_PCA = 60.8456%
AC_P(g|x) = 83.89%
AC_LD = 83.6% , AC_QC = 100%
Accuracy [%]
Class Label
u_1
u_2
u_3
u_d-1
u_d
u_d+1


Class Label vs. Accuracy of the 25 + 1 APCA eigenvectors
AC_APCA = 83.7591%
Optimal class label = 0.13343
AC_FD = 83.7195%
AC_PCA = 60.6669%
AC_P(g|x) = 83.72%
AC_LD = 83.91% , AC_QC = 81.8%
Accuracy [%]
Class Label
u_1
u_2
u_3
u_d-1
u_d
u_d+1