Knowledge Discovery Based Identification of Musical Pitches and Instruments in Polyphonic Sounds. [⋆]

Rory A. Lewis, Xin Zhang, Zbigniew W. Raś

University of North Carolina, Comp. Science Dept., 9201 Univ. City Blvd. Charlotte, NC 28223, USA

Abstract

Pitch and timbre detection methods applicable to monophonic digital signals are common. Conversely, successful detection of multiple pitches and timbres in polyphonic time-invariant music signals remains a challenge. A review of these methods, sometimes called "Blind Signal Separation", is presented in this paper. We analyze how musically trained human listeners overcome resonance, noise, and overlapping signals to identify and isolate what instruments are playing and then what pitch each instrument is playing. The part of the instrument and pitch recognition system, presented in this paper, responsible for identifying the dominant instrument from a base signal uses temporal features proposed by Wieczorkowska [1] in addition to the standard 11 MPEG7 features. After retrieving a semantical match for that dominant instrument from the database, it creates a resulting foreign set of features to form a new synthetic base*n* signal which no longer bears the previously extracted dominant sound. The system may repeat this process until all recognizable dominant instruments are accounted for in the segment. The proposed methodology incorporates Knowledge Discovery, MPEG7 segmentation and Inverse Fourier Transforms.

Key words: MPEG-7; Polyphonic; MIR; Fourier Transforms; Pitch Detection; Independent Component Analysis; Instrument Detection; Blind Signal Separation.

1 Introduction

Blind Signal Separation (BSS) and Blind Audio Source Separation (BASS) have recently emerged as the subjects of intense work in the fields of Signal Analysis and Music Information Retrieval. This paper focuses on the separation of harmonic signals of musical instruments from a polyphonic domain for purpose of music information retrieval. First, it recognizes the state of the art in the fields of signal analysis. Particularly, Independent Component Analysis and Sparse Decompositions. Next it reviews music information retrieval systems that *blindly* identify sound signals. Herein we first present a new approach to the separation of harmonic musical signals in a polyphonic time-invariant music domain and then secondly, the construction of new correlating signals which include the inherent remaining noise. These signals represent new objects which when included in the database, with continued growth, improve the accuracy of the classifiers used for automatic indexing.

1.1 Signal Analysis

In 1986, Jutten and Herault proposed the concept of Blind Signal Separation [1] as a novel tool to capture clean individual signals from noisy signals containing unknown, multiple and overlapping signals [9]. The Jutten and Herault model comprised a recursive neural network for finding the clean signals based on the assumption that the noisy source signals were statistically independent. Researchers in the field began to refer to this noise as the *cocktail party* property, as in the undefinable buzz of incoherent sounds present at a large cocktail party. By the mid 1990's researchers in neural computation, finance, brain signal processing, general biomedical signal processing and speech enhancement, to name a few, embraced the algorithm. Two models dominate the field; Independent Component Analysis (ICA) [3] and Sparse Decompositions (SD) [19].

[⋆] This paper was not presented at any IFAC meeting. Corresponding author Zbigniew W. Raś, Tel. (704) 687-8574, Fax (704) 687-3516

Email addresses: rorlewis@uncc.edu (Rory A. Lewis), cynthiaxz@uncc.edu (Xin Zhang), ras@uncc.edu (Zbigniew W. Raś).

[1] See Appendix A.

1.1.1 Independent Component Analysis

ICA originally began as a statistical method that expressed a set of multidimensional observations as a combination of unknown latent variables [9]. The principle idea behind ICA is to reconstruct these latent, sometimes called *dormant*, signals as hypothesized independent sequences where k = the unknown independent mixtures from the unobserved independent source signals:

$$x = f(\Theta, s), \tag{1}$$

where $x = (x_1, x_2, ..., x_m)$ is an observed vector and f is a general unknown function with parameters Θ [2] that operates the variables listed in the vector $s = (s_1, ..., s_n)$

$$s(t) = [s_1(t), ..., s_k(t)]^T. \tag{2}$$

Here a data vector *x(t)* is observed at each time point t, such that given any multivariate data, ICA can decorrelate the original noisy signal and produce a clean linear co-ordinate system using:

$$x(t) = \mathbf{A}s(t), \tag{3}$$

where $\mathbf{A}$ is a $n \times k$ full rank scalar matrix. For instance (Fig. 1), if a microphone receives input from a noisy environment containing a jet fighter, an ambulance, people talking and a speaker-phone, then $x_i(t) = a_{i1} * s_1(t) + a_{i2} * s_2(t) + a_{i3} * s_3(t) + a_{i4} * s_4(t)$. In this case we are using $i = 1 : 4$ ratio. Rewriting it in a vector notation, it becomes $x = \mathbf{A}^* s$. For example, looking at a two-dimensional vector $x = [x_1 x_2]^T$ ICA finds the decomposition:

$$\begin{vmatrix} x_1 \\ x_2 \end{vmatrix} = \begin{vmatrix} a_{11} \\ a_{21} \end{vmatrix} s_1 + \begin{vmatrix} a_{12} \\ a_{22} \end{vmatrix} s_2 \tag{4}$$

$$\mathbf{x} = \mathbf{a}_1 s_1 + \mathbf{a}_2 s_2 \tag{5}$$

where a_1, a_2 are basis vectors and s_1, s_2 are basis coefficients.

1.1.2 Sparse Decomposition

Sparse decomposition was first introduced in the field of image analysis by Field and Olshausen[18]. Nowadays, the most general SD algorithm is probably Zibulevsky's where his resulting optimization is made on two factors based on the output vector's entropy and sparseness. Similar to ICA, in SD, the resulting signal *x(t)* is the sum of the unknown $n \times k$ matrix $\mathbf{A}$ and noise $\xi(t)$, where n represents the sensors and k represents the unknown scalar source signals.:

$$x(t) = As(t) + \xi(t). \tag{6}$$

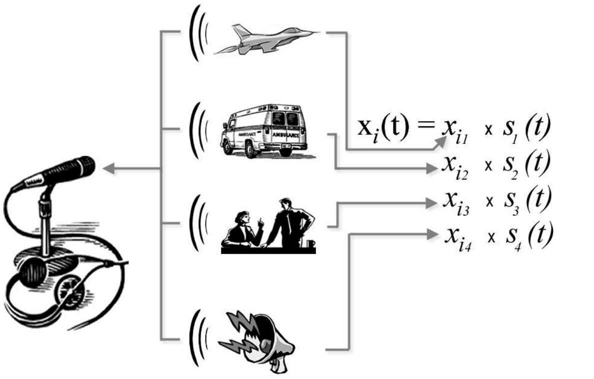

Fig. 1. A noisy cocktail party

The signals are "sparsely" represented in a signal dictionary [25]:

$$s_i(t) = \sum_{k=1}^{k} C_{ik\varphi k}(t), \tag{7}$$

where the ik and φk represent the atoms of the dictionary.

1.2 Music Information Retrieval

In the field of Music Information Retrieval systems, algorithms that analyze polyphonic time-invariant music signals systems operate in either the time domain [7], the frequency domain [21] or both the time and frequency domains simultaneously [13]. Kostek takes a different approach and instead divides BSS algorithms into either those operating on multichannel or single channel sources. Multichannel sources detect signals of various sensors whereas single channel sources are typically harmonic [6]. For clarity, let it be said that experiments provided herein switch between the time and frequency domain, but more importantly, per Kostek's approach, our experiments fall into the multichannel category because, at this point of experimentation two harmonic signals are presented for BSS. In the future, a polyphonic signal containing a harmonic and a percussive may be presented.

1.2.1 BSS in MIR, A Brief Review

In 2000, Fujinaga and MacMillan created a real time system for recognizing orchestral instruments using an exemplar-based learning system that incorporated a k nearest neighbor classifier (k-NNC) [8] using a genetic algorithm to recognize monophonic tones in a database of 39 timbres taken from 23 instruments. Also, in 2000, Eronen and Klapuri created a musical instrument recognition system that modeled the temporal and spectral characteristics of sound signals [11]. The classification system used thirty-two spectral and temporal features and a signal processing algorithms that measured the features of the acoustic signals. The Eronen system was a step forward in BSS because the system was pitch independent and it successfully isolated tones of musical instruments using the full pitch range of 30 orchestral instruments played with different articulations. Also, both hierarchic and direct forms of classification were evaluated using 1498 test tones obtained from the McGill University Masters Samples (MUMs) CDs including "home made" recordings from amateur musicians.

In 2001 Zhang constructed a multi-stage system that segmented the music into it individual notes, estimated the harmonic partial estimation from a polyphonic source and then normalized the features for loudness, length and pitch [24]. The features included the 1) temporal features accounting for rising speed, degree of sustaining, degree of vibration, and releasing speed, 2) spectral features accounting for the spectral energy distribution between low, middle and high frequency sub-bands and the partial harmonic such as brightness, inharmonicity, tristimulus, odd partial ratio, irregularity and dormant tones. Zhang's system successfully identified instruments playing in a polyphonic music pieces. In one the polyphonic source contained 12 instruments including, cello, viola, violin, guitar, flute, horn, trumpet, piano, organ, erhu, zheng, and sarod. The significance of Zhang's system was in the manner it used artificial neural networks to find the dominant instrument: First it segmented each piece into notes and then categorized the music based on the what instrument played the most notes. It then weighted this number by the likelihood value of each note when it is classified to this instrument. For example, if all the notes in the music piece were grouped into K subsets: $I_1; I_2; ...I_K$, with I_i corresponding to the ith instrument, then a score for each instrument was computed as:

$$s_{I_i} = \sum_{x \in I_i} O_i(x), \quad i = 1 \sim k \tag{8}$$

where x denotes a note in the music piece, and $O_i(x)$ is the likelihood that x will be classified to ith instrument. Next, Zhang normalized the score to satisfy the following condition:

$$s_{I_i} = \sum_{i=1}^{k} s(I_i) = 1 \tag{9}$$

It is interesting to note the similarity between this and Zibulevsky's Eq.07 *infra.* Zhang used 287 music monophonic and polyphonic pieces and he reached an accuracy of 80 % success in identifying the dominant instrument and 90 % if intra-family confusions were able to be dismissed. Classification of the Zhang's system incorporated a Kohonen self-organizing map to select the optimal structure of each feature vector.

In 2002, Wieczorkowska, collaborated with Slezak, Wróblewski and Synak [1] and used MPEG-7 based features to create a testing database for training classifiers used to identify musical instrument sounds. She used seventeen MPEG-7 temporal and spectral descriptors observing the trends in evolution of the descriptors over the duration of a musical tone, their combinations and other features. Wieczorkowska compared the classification performance of the kNNC and rough set classifiers using various combinations of features. Her results showed that the kNNC classifier outperformed, by far, the rough set classifiers.

In 2003, Eronen and Agostini both tested, in separate tests, the viability of using decision tree classifiers in Music Information retrieval. They both found that decision tree classifiers ruined the classification results: Eronen's system recognized groups of musical instruments from isolated notes using Hidden Markov Models [4]. Eronen classified the instruments into groups such as strings or woodwinds, not as individual instruments. Agostini's system [16] tested a monophonic base of 27 instruments using eighteen temporal and spectral features with a number of classification procedures to determine which procedure worked most effectively. The experimentation used a number of classical methods including canonical discriminant analysis, quadratic discriminant analysis and support vector machines. Agostini's Support Vector tests yielded a 70 % accuracy on individual instruments. Groups of instruments yielded 81% accuracy. As in this paper's experiments, Agostini's classifiers were MPEG-7 based. The experiments used 18 descriptors for each tone to compute mean and standard deviation of 9 features over the length of each tone. Agostini's system used a 46 ms window for the zero-crossing rate to procure measurements directly from the waveform as the number of sign inversions. To obtain a useable number of harmonics a pitch tracking algorithm controlled each signal by first analyzing it at a low-frequency and repeating it at smaller resolutions until a sufficient number of harmonics was estimated. Interestingly, they used a variable window size to obtain a frequency resolution of at least 1/24 of octaves. The team evaluated the harmonic structure of their signals with FFT's using half-overlapping windows.

In 2004, Kostek developed a 3-stage classification system that successfully identified up to twelve instruments played under a diverse range of articulations [12]. The manner in which Kostek designed her stages of signal preprocessing, feature extraction and classification may prove to be the standard in BSS MIR. In the preprocessing stage Kostek incorporates 1) the average magnitude difference function and 2) Schroeder's histogram for purposes of pitch detection. Her feature extraction stage extracts three distinct sets of features: Fourteen FF1' based features, MPEG-7 standard feature parameters and wavelet analysis. In the final stage, for classification, Kostek incorporates a multi layer ANN classifier.

Importantly, Kostek concluded that she retrieved the strongest results when employing a combination of both MPEG- 7 and wavelet features. Also the performance deteriorated as the number of instruments increased.

2 Experiments

Stepping back and reviewing Kostek, Zhang and Agostini, it became apparent to the authors that BSS works diametrically in opposition to the manner in which trained human listeners segment polyphonic sources of music. When presented with a polyphonic source signal, trained humans overcome resonance, noise and the complexity of instruments playing simultaneously to identify and isolate what instruments are playing and then also identify what pitch each instrument is playing. The basis for the BSS system presented in this paper began by the authors thinking very carefully on how humans, versus classical MIR systems, identify sounds in polyphonic sources. Here a small, anecdotal test formed the seed for the system presented herein:

2.0.2 Trained Human Being's and BSS, a mini experiment

In the Spring of 2006, in order to get a sense of how humans listen to music, one of the authors, Lewis, took an original piece of music he composed and performed with his band, changed it slightly and tested the band members as follows accordingly. Lewis knew these results would be anecdotal and non scientific but he was intrigued by what the outcome would be. Lewis knew that each band member was very familiar with the song and with the instrumentation of the song because they were present when Lewis composed the song, they recorded it over the course of weeks in a studio and they performed the song live in front of audiences many hundreds of times. Essentially, each member knew the song intimately. Lewis made four new versions: Version 1 omitted the kick drum and symbol on drum tracks. Version 2 changed bass notes and omitted some bass notes. Version 3 swapped horn sections around and changed the pitch of the horn at six sections. Finally in Version 4, Lewis extracted the guitar piece and inserted three never before played chords into the song. Lewis asked each member to listen to the three versions of the song - except for the version in which Lewis changed the instrument in which the listener played. For example, Version 3 contained changes to the horn section, here the horn player listened to Version1,2, and 4, not version 3 where he would immediately here his horn solo's were swapped. As the horn player listened to versions 1,2 and 4 he began to get bored. Upon being asked to listen carefully to see what was changed, he could not here the missing drum tracks on version 1, the missing and changed bass guitar on version 2 or the changed guitar tracks on version 4. In fact each member of the band could not hear any changes to other instruments even when asked specifically to listen to them - except for one instance, the bass player identified one of the 14 changes in the guitar track and asked if it was an "earlier" version where Lewis played the guitar track differently.

The authors concluded that trained musicians practically block out instruments they are not interested in. The bass player was interested in one particular guitar sections because he cued one of his solos off of the timing of the missing note. At this moment, he would tune into the guitar and then block it out as he played his solo. The issue became: How do musicians block out sound? How do New Yorker's block out the constant horn honking, ambulance and police sirens so they can fall asleep, or, conversely, how do farmers block out animal sounds so they can fall asleep? The answer, for purposes of this paper is, we do not know how humans block out sound but clearly - they do. More so, even with an in depth study of Kostek, Zhang an Agostini, the system developed transmute the signal into frequency domains and manipulate it but focusing on the dominant timbres, pitches, cepstrums, tristimuluses and frequencies, to name a few. The common factor in all of the above is that only the original sound source is used. In other words, non of the above insert into the equation a foreign entity - as humans probably do. Also, in non of the above approaches we train the classifiers using artificial samples of music objects produced by MIR system.

2.1 Trained Human Being's and new instruments

A human that has never heard a South African Zulu Penny Whistle, cannot - *not hear it* until he or she has heard it a few times. Typically Lewis' band members, like most experienced musicians in bands, can hear a song, listen to the counter instruments playing in the song and play it almost immediately. Except when the humans have not heard an instrument that they normally would block out. This became evident when Lewis brought back to the USA, recordings of songs he purchased in Johannesburg. The band members were not able to focus on anything, let alone their own instrument parts, because of the new instrument, the Zulu Penny Whistle could not be blocked out. Why?

The authors believe the answer lies in the fact that because the band member's had never heard a Zulu Penny Whistle, they had no past data of Zulu Penny Whistle Sounds that would be used to block them out and enable them to focus on their counterpart in the song. Again this lead the authors to believe that humans use a set of sounds in their heads to block out noise in a song so they can focus on exactly the portion of the song they want to listen to. The seminal question the authors asked is the same question that lead them to develop the system presented in this paper which is a system that uses a foreign entities to block out signals in polyphonic signals.

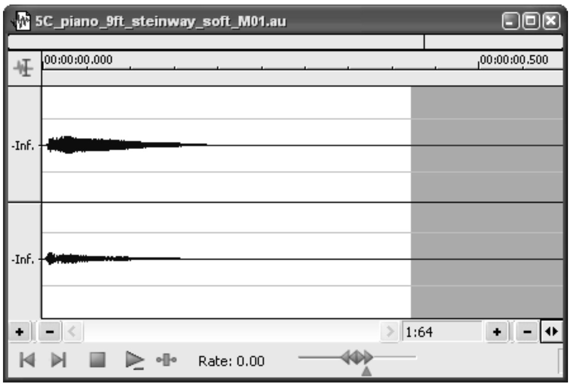

Fig. 2. 5C Piano @ 44,100Hz, 16 bit, stereo

2.2 Overview of the system

In short, when the system reads a polyphonic source, it identifies a dominant aspect of the polyphonic source, finds its match in the database and inserts this foreign entity into the polyphonic source, to do what humans do, i.e., block the portion of the original sound not interested in.

To perform the experiments, the system analyzes 4 separate versions of a polyphonic source (*see samples in figures 3 to 6 below*) containing two harmonic continuous signals obtained from the McGill University Masters Samples (MUMs) CDs. These samples contain a mix of samples one and two, with various levels of noise. Specifically, the first sample contains a C at octave 5 played on a nine foot Steinway, recorded at 44,100HZ, in 16-bit stereo. (Fig. 2) The second sample contains an A at octave 3 played on a Bb Clarinet, recorded at 44,100HZ, in 16-bit stereo. (Fig. 3) The third sample contains a mix of the first and second samples with no noise added, using Sony's Sound Forge 8.0 and containing a pure mix recorded at 44,100HZ, in 16-bit stereo. (Fig. 4) Similarly, the fourth sample contains a mix of the first and second samples with noise added at -17.8 dB (-12.88 %)(Fig. 5). The fifth sample contains a mix of the first and second samples with noise added at -36.05 dB (-1.58 %)(Fig. 6). Finally, the sixth sample contains a mix of the first and second samples with noise added at -8.5 dB (-37.58 %)(Fig. 7).

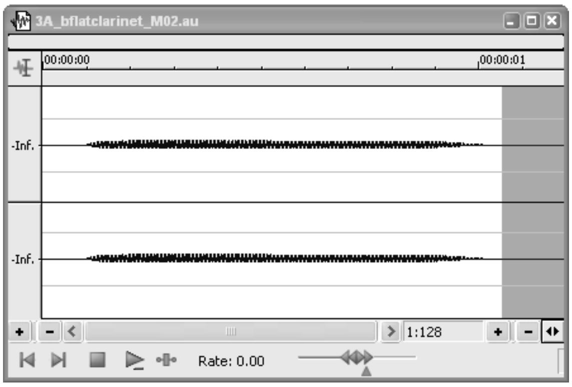

Fig. 3. 3A Bb Clarinet @ 44,100Hz, 16 bit, stereo

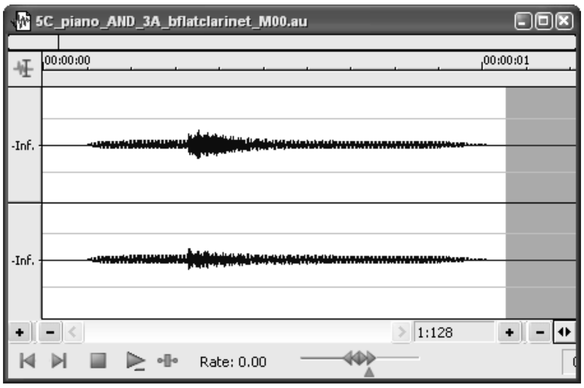

Fig. 4. Piano and Clarinet @ 44,100Hz, 16 bit, stereo - No Noise

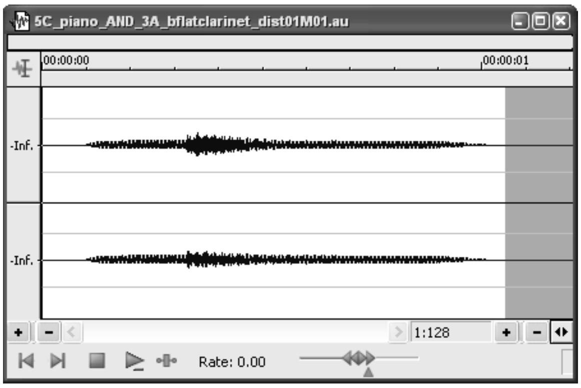

Fig. 5. Piano and Clarinet @ 44,100Hz, 16 bit, stereo - 01 Noise at -17.8 dB (-12.88 %)

2.2.1 Formal Procedure

In explaining the system procedures reference will be made to the two foreign samples housed in the database (Fig. 2) (Fig. 3) containing the piano 5c and clarinet 3a. The polyphonic input to the system will consist of the four variations of the mix of piano 5c and clarinet 3a. For the purpose of this discussion it is also assumed that the clarinet 3a is the dominant feature of all four variations of the mix. The system reads the input and uses an FFT to transform it into the frequency domain. In the frequency domain it determines that the fundamental frequency of 3a with a woodwind-like timbre is dominant Fig. 8). The system searches the database and first extracts all 3a pitches of each instrument. Next it separates all woodwind-like sounds in the 3a temporary cache. At this point it uses the MPEG-7 descriptors based classifier to find 3a clarinet as close to the one identified. Here it extracts the wave of 3a clarinet and performs a FFT on this, a foreign sound entity. It subtracts the resultant of the foreign entities FFT from the input entities FFT leaving an FFT, that when subjected to an IFFT pro-

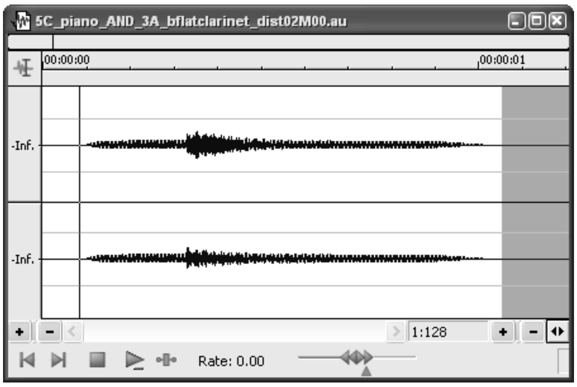

Fig. 6. Piano and Clarinet @ 44,100Hz, 16 bit, stereo - 02 Noise at -36.05 dB (-1.58 %)

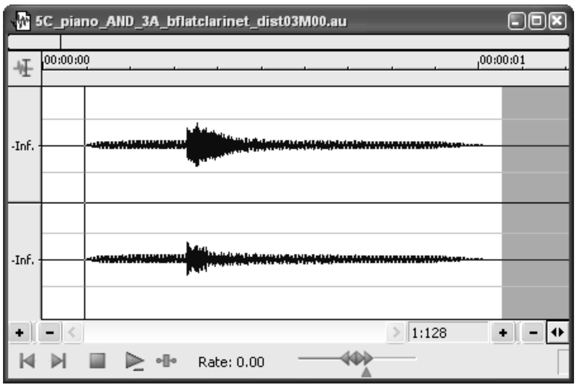

Fig. 7. Piano and Clarinet @ 44,100Hz, 16 bit, stereo - 03 Noise at -8.5 dB (-37.58 %)

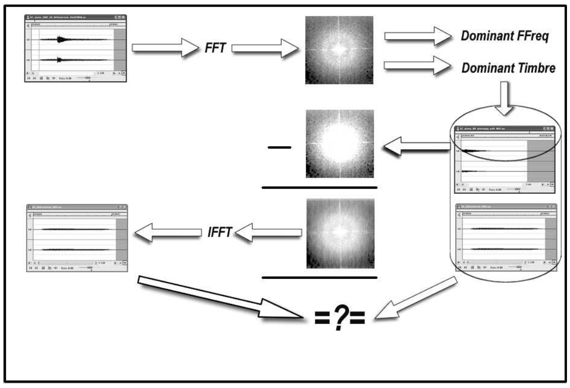

Fig. 8. Theoretical Procedure: Subtracting a foreign, extracted signal's FFT from the source FFT

duces a wave that contains only the piano 5c, resonance, harmonics and other negligible noise.

2.2.2 MPEG-7 features

In considering the use of MPEG-7, the authors recognized that a sound segment containing musical instruments may have three states: transient, quasi-steady and decay. Identifying the boundary of the transient state enables accurate timber recognition. Wieczorkowska presented a timbre detection system in [5] where she split each sound segment into 7 equal intervals. Because different instruments require different lengths, we use a new approach to look at the time it takes for the transient duration to reach the quasi-steady state of the fundamental frequency [23]. It is estimated by computing the local cross-correlation function of the sound object and the mean time to reach the maximum within each frame. Our system developed herein is based on the following MPEG-7 Descriptors:

2.2.3 AudioSpectrumCentroid

The AudioSpectrumCentroid is a description of the center of gravity of the log-frequency power spectrum. Spectrum centroid is an economical description of the shape of the power spectrum. It indicates whether the power spectrum is dominated by low or high frequencies and, additionally, it is correlated with a major perceptual dimension of timbre; i.e.sharpness. To extract the spectrum centroid: 1. Calculate the power spectrum coefficients; 2. Power spectrum coefficients below 62.5 Hz are replaced by a single coefficient, with power equal to their sum and a nominal frequency of 31.25 Hz; 3. Frequencies of all coefficients are scaled to an octave scale anchored at 1 kHz.

2.2.4 AudioSpectrumSpread

The AudioSpectrumSpread is a description of the spread of the log-frequency power spectrum. Spectrum spread is an economical descriptor of the shape of the power spectrum that indicates whether it is concentrated in the vicinity of its centroid, or else spread out over the spectrum. It allows differentiating between tone-like and noise-like sounds. To extract the spectrum Spread, we Calculate the spectrum spread as the RMS deviation with respect to the centroid, on an octave scale.

2.2.5 HarmonicSpectralCentroid

The HarmonicSpectralCentroid is computed as the average over the sound segment duration of the instantaneous HarmonicSpectralCentroid within a running window. The instantatneous HarmonicSpectralCentroid is computed as the amplitude (linear scale) weighted mean of the harmonic peaks of the spectrum. To extract the Harmonic Spectral Centroid, 1. Estimate the harmonic peaks over the sound segment. 2. Calculate the instantaneous HarmonicSpectralCentroid. 3. Calculate the average HarmonicSpectralCentroid for the sound segment.

2.2.6 HarmonicSpectralDeviation

HarmonicSpectralDeviation is computed as the average over the sound segment duration of the instantaneous

HarmonicSpectralDeviation within a running window. The instantaneous HarmonicSpectralDeviation is computed as the spectral deviation of log-amplitude components from a global spectral envelope. The Harmonic Spectral Deviation is extracted using the following algorithm 1. Estimate the harmonic peaks over the sound segment. 2. Estimate the spectral envelope. 3. Calculate the instantaneous HarmonicSpectralDeviation. 4. Calculate the average HarmonicSpectralDeviation for the sound segment.

2.2.7 HarmonicSpectralSpread

The HarmonicSpectralSpread is computed as the average over the sound segment duration of the instantaneous HarmonicSpectralSpread within a running window. The instantaneous HarmonicSpectralSpread is computed as the amplitude weighted standard deviation of the harmonic peaks of the spectrum, normalized by the instantaneous HarmonicSpectralCentroid. It is extracted using the following algorithm 1. Estimate the harmonic peaks over the sound segment. 2. Estimate the instantaneous HarmonicSpectralCentroid. 3. Calculate the instantaneous HarmonicSpectralSpread for each frame. 4. Calculate the average HarmonicSpectralSpread for each sound segment.

2.2.8 HarmonicSpectralVariation

The HarmonicSpectralVariation, is the mean over the sound segment duration of the instantaneous HarmonicSpectralVariation. The instantaneous HarmonicSpectralVariation is defined as the normalized correlation between the amplitude of the harmonic peaks of two adjacent frames. It is extracted using the following algorithm. 1. Estimate the harmonic peaks over the sound segment. 2. Calculate the instantaneous HarmonicSpectralVariation each frame. 3. Calculate the HarmonicSpectralVariation for the sound segment.

2.3 Classifiers

The classifiers, applied in the investigations on musical instrument recognition, represent practically all known methods. In our research, so far we have used four classifiers (Bayesian Networks, Logistic Regression Model, Decision Tree J-48 and Locally weighted learning) upon numerous music sound objects to explore the effectiveness of our descriptors. Bayesian Networks is a widely used statistical approach, which represent the dependence structure between multiple variables by a specific type of graphical model, where probabilities and conditional-independence statements are strictly defined. It has been successfully applied to speech recognition [26], [10]. Logistic regression model is a popular statistical approach of analyzing multinomial response variables, since it does not assume normally distributed conditional attributes which can be continuous, discrete, dichotomous or a mix of any of these; it can handle nonlinear relationships between the decision attribute and the conditional attributes. It has been widely used to correctly predict the category of outcome for new instances by maximum likelihood estimation using the most economical model. For details, see [14]. Locally weighted regression is a well-known lazy learning algorithm for pattern recognition. It votes on the prediction based on a set of nearest neighbors (instances) of the new instance, where relevance is measured by a distance function. The local model consists of a structural and a parametric identification, which involve parameter optimization and selection. For details see [17]. Decision Tree-J48 is a supervised classification algorithm, which has been extensively used for machine learning and pattern recognition [20], [22]. A Tree-J48 is normally constructed top-down, where parent nodes represent conditional attributes and leaf nodes represent decision outcomes. It first chooses a most informative attribute that can best differentiate the dataset; it then creates branches for each interval of the attribute where instances are divided into groups; it repeats creating subbranches until instances are clearly separated in terms of the decision attribute; finally it tests the tree by new instances in a test dataset.

3 Results

The authors conducted the experiments bearing four issues in mind: Firstly, what is the confidence of the system in recognizing correctly either one of the two instruments. Secondly, if new sound objects had to be build and used for training classifiers in order to increase their accuracy. Thirdly, when the system subtracted the first instrument from the second instrument, out of curiosity, the authors performed an inverse FFT to hear if indeed the first instrument was missing, here the authors judged how well the first instrument was subtracted. In all experiments, presented below, we tested the sufficiency of our features and sound objects for building successful classifiers.

3.1 Experiment 1: Classification of original sounds from MUMs (10 folds).

Our system's classification example was based on the original sounds from MUMs using Steinway 9' piano and the alto and bass flutes. Varying degrees of noise constituted the progressively noisy mixture samples. Both training/testing for piano vs. alto flute and bass flute (10 folds) was performed. Here, the authors created 55 samples of piano, 30 alto flutes and 31 bass flutes. Note "LGR" means Logistic Regression Model. "accuracy" specifically denotes the Classification Accuracy. The results are presented in Tab. 1. *ClassAc* is the abbreviation for *Classification Accuracy.*

Table 1
Experiment 1

$MUMs$	$TreeJ48$	LRM	$BayesianNet.$
$ClassAc$	99.0991%	100%	99.0991%

Table 2
Experiment 2 with noise

$MUMs$	$TreeJ48$	LRM	$BayesianNet.$
$ClassAc$	50%	50%	50%

Table 3
Experiment 3 with noise

$PianoWN$	$TreeJ48$	LRM	$BayesianNet.$
$ClassAc$	100%	100%	100%

3.2 Experiment 2: Classification of echoed sounds.

For training, we have used the same data set as in the Experiment 1. Testing was done for echoed sounds which are foreign to the classifier. This experiment involved substantial noise in the signal domain. The authors performed testing for piano with noise vs. alto flute and bass flute without noise. For the results see Tab. 2. All three classifiers recognized only about half of the submitted objects which means either additional features or new objects for the training face are still needed.

3.3 Experiment 3: Classification of subtracted piano.

In training for piano vs. none piano (alto flute and bass flute), the authors used 55 samples of piano from MUMs, 25 samples of piano with noise in the signal domain, 25 samples of piano obtained by subtracting flute from mixed samples of piano, flute, and additional noise, 30 alto flutes and 31 bass flutes. In all cases we used Steinway 9' piano. Testing was done for echoed sounds which are foreign to the classifier. This experiment involved substantial noise in the signal domain. The authors performed testing for piano with noise vs. alto flute and bass flute without noise. For the results see Tab. 3. All sound objects submitted and accepted by the classifier have been recognized correctly. All three classifiers recognized about half of the submitted objects which means additional features for the training face are still needed. *PianoWN* is the abbreviation for *piano with noise.*

Table 4
Experiment 4 with noise

$ClarinetWN$	$TreeJ48$	LRM	$BayesianNet.$
$ClassAc$	100%	100%	100%

3.4 Experiment 4: Classification of subtracted Clarinet (10 folds).

In training/testing for Clarinet vs. none clarinet (alto flute and bass flute), the authors used 13 samples of clarinet obtained by subtracting flute from mixed MUMs samples of clarinet, flute, and additional noise, 30 alto flutes and 31 bass flutes also from MUMs. This experiment also involved substantial noise in the signal domain. For the results see Tab. 4. All sound objects submitted and accepted by the classifier have been recognized correctly. Similarly to Experiment 3, all three classifiers recognized about half of the submitted objects which means additional features for the training face are needed. *ClarinetWN* is the abbreviation for *clarinet with noise.*

4 Conclusion

This paper presents initial research concerning automatic indexing of audio by musical instruments of definite pitch, used in contemporary orchestras. Our ultimate goal is to perform automatic classification of musical instrument sound from real recordings for broad range of sounds, independently on the fundamental frequency of the sound. Full range of musical scale for each instrument will be investigated.

Acknowledgements

This work is supported by the National Science Foundation under grant IIS-0414815.

References

[1] D. Slezak P. Synak A.Wieczorkowska and J. Wrblewski. Kdd-based approach to musical instrument sound recognition. *M.-S. Hacid, Z.W. Ra?, D.A. Zighed, Y. Kodratoff (eds.): Foundations of Intelligent Systems.*, Proc. of 13th Symposium ISMIS 2002, Lyon, Franc 4519 Berlin, Heidelberg:28– 36, 2002.

[2] Ella Bingham. Advances in independent component analysis with applications to data mining. *Helsinki University of Technology: Dissertation for the degree of Doctor of Science in Technology*, 2003.

[3] J.F. Cardose. Blind source separation : Statistical principles. *IEEE Proc.*, 9035:2009–2026, 1998.

[4] A. Eronen. Musical instrument recognition using ica-based transform of features and discriminatively trained hmms. *Proceedings of the Seventh International Symposium on Signal Processing and its Applications, ISSPA 2003, Paris, France, 1-4 July*, pages 133–136, 2003.

[5] A. Wieczorkowska et al. Application of temporal descriptors to musical instrument sound. *Journal of Intelligent Information Systems, Integrating Artificial Intelligence and Database Technologies*, 21, no. 1, July, 2003.

[6] B. Kostek et al. Estimation of musical sound separation algorithm effectiveness employing neural networks. *Journal of Intelligent Information Systems*, 24 number 2/3:133–135, May 2005.

[7] S. Amari et al. Multichannel blind deconvolution and equalization using the natural gradient. *Proc. IEEE Workshop. Signal Processing Advances in Wireless Comm.,*, pages 101–104, April 1997.

[8] I. Fujinaga. and K. MacMillan. Realtime recognition of orchestral instruments. *Proceedings of the International Computer Music Conference - Best Presentation Award*, pages 141–143, 2000.

[9] J. Herault and C. Jutten. Space or time adaptive signal processing by neural network models. In J. S. Denker, editor, *Neural Networks for Computing*, volume 151, pages 206–211. American Institute of Physics, New York, 3rd edition, 1986.

[10] J. Glass K. Livescu and J. Bilmes. Hidden feature models for speech recognition using dynamic bayesian network - geneva, switzerland. *Proc. Eurospeech*, pages 2529–2532, September 2003.

[11] A. Eronen A. Klapuri. Musical instrument recognition using cepstral coefficients and temporal features. *In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pages 753–756, 2000.

[12] B. Kostek. Musical instrument classification and duet analysis employing music information retrieval techniques. *Proc. of the IEEE.*, 92(4), 2004), pages=712–729,.

[13] R.H. Lambert and A.J. Bell. Blind separation of multiple speakers in a multipath environment. *Proc. ICASSP*, April:423–426, 1997.

[14] S. le Cessie and J.C. van Houwelingen. Ridge estimators in logistic regression. *Applied Statistics*, 41, no. 1:191–201, 1992.

[15] Te-Won Lee. http://inc2.ucsd.edu/ tewon/. *Institute for Neural Computation (INC), at UCSD*, 2006.

[16] G Agostini M Longar and Emanuele Pollastri. Musical instrument timbres classification with spectral features. *EURASIP Journal on Applied Signal Processing*, Issue 1:5–14, 2003.

[17] C. G. Atkeson A. W. Moore and S. Schaal. Locally weighted learning for control. *Artificial Intelligence Review*, 11 no. 1-5:11–73, Feb. 1997.

[18] B.A. Olshausen and D.J. Field. Sparse coding of natural images produces localized, oriented, bandpass receptive fields. *Technical Report CCN-100-95, Dept. of Psychology, Cornell University*, pages 607–609, 1996.

[19] M. Zibulevsky B. A. Pearlmutter and P. Kisilev. Blind source separation by sparse decomposition. *Independent Component Analysis: Principles and Practice*, 2001.

[20] J. R. Quinlan. C4.5: Programs for machine learning. *Morgan Kaufman*, San Mateo CA, 1993.

[21] P. Smaragdis. Blind separation of convolved mixtures in the frequency domain. *IEEE Procedures Neurocomputing*, 22:21–34, 1998.

[22] A. Wieczorkowska. Classification of musical instrument sounds using decision trees. *8th International Symposium on Sound Engineering and Mastering*, ISSEM'99:225– 230, 1999.

[23] Z. Ras X. Zhang. Differentiated harmonic feature analysis on music information retrieval for instrument recognition. *IEEE International Conference on Granular Computing*, Proc. of IEEE GrC 2006, Atlanta, Georgia, May 2006.

[24] T. Zhang. Instrument classification in polyphonic music based on timbre analysis. *SPIE's Conference on Internet Multimedia Management Systems II (part of ITCom'01, Denver, Aug.*, 4519:136– 147, 2001.

[25] M. Zibulevsky and B. A. Pearlmutter. Blind source separation by sparse decomposition in a signal dictionary. *Letter Communicated by Michael Lewicki*, 2000.

[26] G. Zweig. Speech recognition with dynamic bayesian networks. *Ph.D. dissertation, Univ. of California, Berkeley, California*, 1998.

A ICA Timeline

This paper addresses the state of the art in BSS and ICA from 2000 to 2006. Herein is a brief history of ICA work achievements prior to 2000.[15]

1986 – Herault and Jutten - introduce Concept
1991 – Jutten and Herault.
1994 – Karhunen and Joutsensalo.
1994 – Cichocki Unbehauen and Rummert.
1994 – Comon - cost functions between the sensors.

Unsupervised learning rules
1961 – Barlow.
1992 – Linsker.
1992 – Atick.
1994 – Nadal and Parga - low-noise neurals.
1995 – Bell and Sejnowsk - forecasting
1996 – Roth and Baram.
1995 – Bell and Sejnowski - information-theoret.
1996 – Cardoso and Laheld.

Other algorithms for performing ICA.
1990 – Gaeta and Lacoume - maximum likelihood.
1995 – Bell and Sejnowski.
1992 – Pham.
1996 – Pearlmutter and Parra.
1996 – MacKay.
1997 – Cardoso.
1997 – Girolami and Fyfe.
1991 – Cover and Thomas - negentropy.
1997 – Girolami and Fyfe - multiple output.

Nonlinear PCA algorithms for ICA.
1994 – Karhunen and Joutsensalo.
1997 – Xu
1993 – Oja
1994 – Comon
1997– Girolami and Fyfe.
1995 – Bell and Sejnowski – infomax algorithm.
1998 – Lee et al - infomax principle.

The original infomax learning rules.
1995 – Bell and Sejnowski - super-Gaussian sources.
1997 – Girolami and Fyfe - negentropy.
1997 – Lee Girolami and Sejnowski - infomax.
1997 – Amari - natural gradients.
1996 – Cardoso and Laheld - relative gradients.
1997 – Lee Girolami and Sejnowski - physiological data.

Demonstrating the power of the learning algorithm.
1996 – Makeig et al - EEG and ERP data.
1997 – Jung et al - EEG - line noise.
1997 – McKeown et al - human brain.
1997 – Bell and Sejnowski – edge filterss.
1997 – TBartlett and Sejnowskihe - sparse distribution
1997 – Gray Movellan and Sejnowski - face recognition .

Multichannel blind source separation problem.
1994 – Yellin and Weinstein .
1995 – Ngyuen and Jutten.
1996 – Torkkola - convolved sources.

1997 – Lee Bell and Lambert - feedforward system.
1996 – Lambert - polynomial filter matrix.
1997 – Lee Bell and Orglmeister – speech recognition system.

Tackle limitations of ICA - nonlinear mixing model.
1996 – In Hermann and Yang.
1997 – Lin and Cowan.
1997 – Pajunen - nonlinear self-organizing.
1992 – Burel Lee.
1997 – Koehler and Orglmeister.
1997 – Taleb and Jutten.
1997 – Yang Amari and Cichocki - flexible mixing model.
1998 – Hochreiter and Schmidhuber –low complexity coding.