Pitch Histograms in Audio and Symbolic Music Information Retrieval

George Tzanetakis
(corresponding author)

Computer Science Department
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
USA
Tel: +1 412-268-3974
Fax: +1 412-268-5576
gtzan@cs.cmu.edu

Andrey Ermolinskyi, Perry Cook

Computer Science Department
Princeton University
35 Olden Street
Princeton NJ 08544
USA
Tel:+1 609-258-5030
Fax: +1 609-258-1771
andreye@princeton.ed
prc@cs.princeton.edu

ABSTRACT

In order to represent musical content, pitch and timing information is utilized in the majority of existing work in Symbolic Music Information Retrieval (MIR). Symbolic representations such as MIDI allow the easy calculation of such information and its manipulation. In contrast, most of the existing work in Audio MIR uses timbral and beat information, which can be calculated using automatic computer audition techniques.

In this paper, Pitch Histograms are defined and proposed as a way to represent the pitch content of music signals both in symbolic and audio form. This representation is evaluated in the context of automatic musical genre classification. A multiple-pitch detection algorithm for polyphonic signals is used to calculate Pitch Histograms for audio signals. In order to evaluate the extent and significance of errors resulting from the automatic multiple-pitch detection, automatic musical genre classification results from symbolic and audio data are compared. The comparison indicates that Pitch Histograms provide valuable information for musical genre classification. The results obtained for both symbolic and audio cases indicate that although pitch errors degrade classification performance for the audio case, Pitch Histograms can be effectively used for classification in both cases.

1. INTRODUCTION

Traditionally, music information retrieval (MIR) has been separated in symbolic MIR where structured signals such as MIDI files are used, and audio MIR where arbitrary unstructured audio signals are used. For symbolic MIR, melodic information is typically utilized while for audio MIR typically timbral and rhythmic information is used. In this paper, the main focus is the representation of global pitch content statistical information about musical signals both in symbolic and audio form. More specifically, Pitch Histograms are defined and proposed as a way to represent pitch content information and are evaluated in the context of automatic musical genre classification.

Given the rapidly increasing importance of digital music distribution, as well as the fact that large web-based music collections are continuing to grow in size exponentially, it is obvious that the ability to effectively navigate within these collections is a desirable quality. Hierarchies of musical genres are used to structure on-line music stores, radio stations as well as private collections of computer users.

Up to now, genre classification for digitally stored music has been performed manually and therefore automatic classification mechanisms would constitute a valuable addition to existing music information retrieval systems. One could, for instance, envision an Internet music search engine that searches for a set of specific musical features (genre being one of them), as specified by the user, within a space of feature-annotated audio files. Musical content features that are good for genre classification can be used in other type of analysis such as similarity retrieval or summarization. Therefore genre classification provides a way to evaluate automatically extracted features that describe musical content. Although the division of music into genres is somewhat subjective and arbitrary, there exist perceptual criteria related to the timbral, rhythmic and pitch content of music that can be used to characterize a particular musical genre. In this paper, we focus on pitch content information and propose Pitch Histograms as way to represent such information.

Symbolic representations of music such as MIDI files are essentially similar to musical scores and typically describe the start, duration, volume, and instrument type of every note of a musical piece. Therefore, in the case of symbolic representation the extraction of statistical information related to the distribution of pitches, namely the Pitch Histogram, is trivial. On the other hand, extracting pitch information from audio signals is not easy. Extracting a symbolic representation from an arbitrary audio signal, called “polyphonic transcription”, is still an open research problem solved only for simple and synthetic “toy” examples. Although the complete pitch information of an audio signal can not be extracted reliably, automatic multiple pitch detection algorithms can still provide enough accurate information to calculate overall statistical information about the distribution of pitches in the form of a Pitch Histogram. In this paper, Pitch Histograms are evaluated in the context of musical genre classification. The effect of pitch detection errors for the audio case is investigated by comparing genre classification results for MIDI and audio-from-MIDI signals. For the remainder of the paper it is important to define the following terms: symbolic, audio-from-MIDI and audio. Symbolic refers to MIDI files, audio-from-MIDI refers to audio signals generated using a synthesizer playing a MIDI file and audio refers to general audio signals such as mp3 files found on the web.

This work can be viewed as a bridge connecting audio and symbolic MIR through the use of pitch information for retrieval and genre classification. Another valuable idea described in this paper is the use of MIDI data as the ground truth for evaluating audio analysis algorithms applied to audio-from-MIDI data.

The remainder of this paper is structured as follows: A review of related work is provided in Section 2. Section 3 introduces Pitch Histograms and describes their calculation for symbolic and audio data. The evaluation of Pitch Histograms features in the context of musical genre classification is described in Section 4. Section 5 describes the implementation of the system and Section 6 contains conclusions and directions for future work.

2. RELATED WORK

Music Information Retrieval (MIR) refers to the process of indexing and searching music collections. MIR systems can be classified according to various aspects such as the type of queries allowable, the similarity algorithm, and the representation used to store the collection. Most of the work in MIR has traditionally concentrated on symbolic representations such as MIDI files. This is due to several factors such as the relative ease of extracting structured information from symbolic representations as well as their modest performance requirements, at least compared to MIR performed on audio signals. More recently a variety of MIR techniques for audio signals have been proposed. This development is spurred by increases in hardware performance and development of new Signal Processing and Machine Learning algorithms.

Symbolic MIR has its roots in dictionaries of musical themes such as Barlow and DeRoure (1948). Because of its symbolic nature, it is often influenced by ideas from the field of text information retrieval (Baeza-Yates and Ribeiro-Neto, 1999). Some examples of modeling symbolic music information as text for retrieval purposes are described in Downie (1999) and Pickens (2000). In most cases the query to the system consists of a melody or a melodic contour. These queries can either be entered manually or transcribed from a monophonic audio recording of the user humming or singing the desired melody. The second approach is called Query-by-humming and some early examples are Kageyama, Mochizuki and Takashima (1993) and Ghias, Logan, Chamberlin and Smith (1995). A variety of different methods for calculating melodic similarity are described in Hewlett and Selfridge-Field (1998). In addition to melodic information, other types of information extracted from symbolic signals can also be utilized for music retrieval. As an example the production of figured bass and its use for tonality recognition is described in Barthelemy and Bonardi (2001) and the recognition of Jazz chord sequences is treated in Pachet (2000). Unlike symbolic MIR which typically focuses on pitch information, audio MIR has traditionally used features that describe the timbral characteristics of musical textures as well as beat information. Representative examples of techniques for retrieving music based on audio signals include: performances of the same orchestral piece based on its long-term energy profile (Foote, 2000), discrimination of music and speech (Logan, 2000) (Scheirer & Slaney, 1997), classification, segmentation and similarity retrieval of musical audio signals (Tzanetakis & Cook, 2000), and automatic beat detection algorithms (Scheirer, 1998) (Laroche, 2001).

Although accurate multiple pitch detection on arbitrary audio signals (polyphonic transcription) is an unsolved problem, it is possible to extract statistical information regarding the overall pitch content of musical signals. Pitch Histograms are such a representation of pitch content that has been used together with timbral and rhythmic features for automatic musical genre classification in Tzanetakis and Cook (2002). The idea of Pitch Histograms is similar to the Pitch Profiles proposed in (Krumhansl, 1990) for the analysis of tonal music in symbolic form. The original version of this paper first appeared in Tzanetakis, Ermolinskyi and Cook (2002). Pitch Histograms are further explored and their performance is compared both for symbolic and audio signals in this paper. The goal of the paper is not to demonstrate that features based on Pitch Histograms are better or more useful in any sense compared to other existing features but rather to show their value as an additional alternative source of musical content information. As already mentioned, symbolic MIR and audio MIR traditionally have used different algorithms and types of information. This work can be viewed as an attempt to bridge these two distinct approaches.

3. PITCH HISTOGRAMS

Pitch Histograms are global statistical representations of the pitch content of a musical piece. Features calculated from them can be used for genre classification, similarity retrieval as well as any type of analysis where some representation of the musical content is required. In the following subsections, Pitch Histograms are defined and used to extract features for genre classification.

3.1 Pitch Histogram Definition

A Pitch Histogram is, basically, an array of 128 integer values (bins) indexed by MIDI note numbers and showing the frequency of occurrence of each note in a musical piece. Intuitively, Pitch Histograms should capture at least some amount of information regarding harmonic features of different musical genres and pieces. One expects, for instance, that genres with more complex tonal structure (such as Classical music or Jazz) will exhibit a higher degree of tonal change and therefore have more pronounced peaks in their histograms than genres such as Rock, Hip-Hop or Electronica music that typically contain simple chord progressions.

Two versions of the histogram are considered: an unfolded (as defined above) and a folded version. In the folded version, all notes are transposed into a single octave (array of size 12) and mapped to a circle of fifths, so that adjacent histogram bins are spaced a fifth apart, rather than a semitone. More specifically if we denote n the MIDI note number (C4 is 60) then the following conversion can be used to get the folded version index c: c = (n mod 12). For mapping to the circle of fifths the following conversion can be used c’ = (7 x c) mod 12.

Folding is perform in order to represent pitch class information independently of octave and the mapping to the circle of fifths is done in order to make the histogram better suited for expressing tonal music relations and it was found empirically that the extracted features result in better classification accuracy. As an example a piece in C major will have strong peaks at C and G (tonic and dominant) and will be more closely related to a piece in G major (G and D peaks) than a piece in C# major. The mapping to the circle of fifths makes the Pitch Histograms of two harmonically related pieces more similar in shape that when the chromatic histogram is used. It can therefore be said that the folded version of the histogram contains information regarding the pitch content of the music (or a crude approximation of harmonic information), whereas the unfolded version is useful for determining the pitch range of the piece. As an example consider two pieces both mostly in C major, one of which is two octave higher on average than the other. These two pieces will have very similar folded histograms however their unfolded histograms will be different as the higher piece will have more energy at the higher pitch bins of the unfolded Pitch Histogram.

3.2 Pitch Histogram Features

In order to perform automatic musical genre classification, after the Pitch Histogram has been computed, it is transformed into a four-dimensional feature vector. This feature vector is used as a characterization of the pitch content of a particular musical piece. For classification, a supervised learning approach is followed, where labeled collections of such feature vectors are used to train and evaluate automatic musical genre classifiers.

The following four features based on the Pitch Histogram are proposed for classifying musical genres:

- PITCH-Fold: Bin number of the maximum peak of the folded histogram. This typically corresponds to the most common pitch class of the musical piece (in tonal music usually the dominant or the tonic).
- AMPL-Fold: Amplitude of the maximum peak of the folded histogram. This corresponds to the frequency of occurrence of the main pitch class of the song. This peak is typically higher for pieces that do not contain many harmonic changes.
- PITCH-Unfold: Bin number of the maximum peak of the unfolded histogram. This corresponds to the octave range of the musical pitch of the song. For example, a flute piece will have a higher value of this feature than a bass piece even if they are in the same tonal key.
- DIST-Fold: Interval (in bins) between the two highest peaks of the folded histogram. For pieces with simple harmonic structure, this feature will have value 1 or –1 corresponding to a music interval of a fifth or a fourth.

These features were chosen based on experimentation and subsequent evaluation in the task of musical genre classification. As an example Jazz music tends to have more chord changes and therefore has lower values of AMPL-Fold on average. Rather than trying to find thresholds empirically, a disciplined machine learning approach was used were these informal observations as well as other non-obvious patterns in the data are learned and evaluated for classification. This is done by training a statistical classifier using labeled feature vectors as examples for each class of interest. The choice of the particular feature set is an important one, as it is desirable to filter out the irrelevant statistical properties of the histogram while retaining information identifying the pitch content. Although this choice is not necessarily optimal, it will empirically be shown to be effective for musical genre classification.

3.3 Pitch Histogram Calculation

For MIDI files, the histogram is constructed using a simple linear traversal over all MIDI events in the file. For each encountered Note-On event (excluding the ones played on the MIDI drum channel), the algorithm increments the corresponding note's frequency counter. The value in each histogram bin is normalized in the last stage of the calculation by dividing it by the total number of notes of the whole piece. This is done in order to account the variability in the average number of notes per unit time between different pieces of music. This is normalization doesn't affect the relative frequencies of occurrence of each pitch class. Example unfolded Pitch Histograms belonging to two genres (Jazz and Irish Folk music) are shown in Figure 1. By visual inspection of this figure, it can be seen that the Pitch Histograms corresponding to Irish Folk music have few and sparse peaks indicating a smaller amount of harmonic change than exhibited by Jazz music. This type of information is what the proposed features attempt to capture and use for automatic musical genre classification.

For calculating Pitch Histograms from audio data, the multiple pitch detection algorithm proposed in (Tolonen & Karjalainen, 2000) is used. The following subsection provides a description of this algorithm.

3.4 Multiple Pitch Detection Algorithm

The multiple pitch detection used for Pitch Histogram calculation is based on the two channel pitch analysis model described in Tolonen & Karjalainen (2000). A block diagram of this model is shown in Figure 2. The signal is separated into two channels, below and above 1kHz. The channel separation is done with filters that have 12 dB/octave attenuation at the stop band. The lowpass block also includes a highpass rolloff with 12dB/octave below 70 Hz. The high-channel is half-wave rectified and lowpass filtered with a similar filter (including the highpass characteristic at 70Hz) to that used for separating the low channel.

The periodicity detection is based on "generalized autocorrelation" i.e. the computation consists of a discrete Fourier transform (DFT), magnitude compression of the spectral representation, and an inverse transform (IDFT). The signal x2 of Figure 2 is obtained as follows:

$$\begin{aligned} X2 &= \mathrm{IDFT}(|\mathrm{DFT}(x_{low})|^k) + \mathrm{IDFT}(|\mathrm{DFT}(x_{high})|^k) \\ &= \mathrm{IDFT}(|\mathrm{DFT}(x_{low})|^k + |\mathrm{DFT}(x_{high})|^k) \end{aligned}$$

where x_{low} and x_{high} are the low and the high channel signals before the periodicity detection blocks in Figure 2. The parameter k determines the frequency-domain compression (for normal autocorrelation k=2). The Fast Fourier Transform (FFT) and its inverse (IFFT) are used to speed the computation of the transforms.

The peaks of the summary autocorrelation function (SACF) (signal x2 of Figure 2) are relatively good indicators of potential pitch periods in the signal analyzed. In order to filter out integer multiple of the fundamental period, a peak pruning technique is used. The original SACF curve is first clipped to positive values and then time-scaled by a factor of two and subtracted from the original clipped SACF function, and again the result is clipped to have positive values only. That way, repetitive peaks with double the time lag of the basic peak are removed. The resulting function is called the enhanced summary autocorrelation (ESACF) and its prominent peaks are accumulated in the Pitch Histogram calculation. More details about the calculation steps of this multiple pitch detection model, as well as its evaluation and justification can be found in Tolonen & Karjalainen (2000).

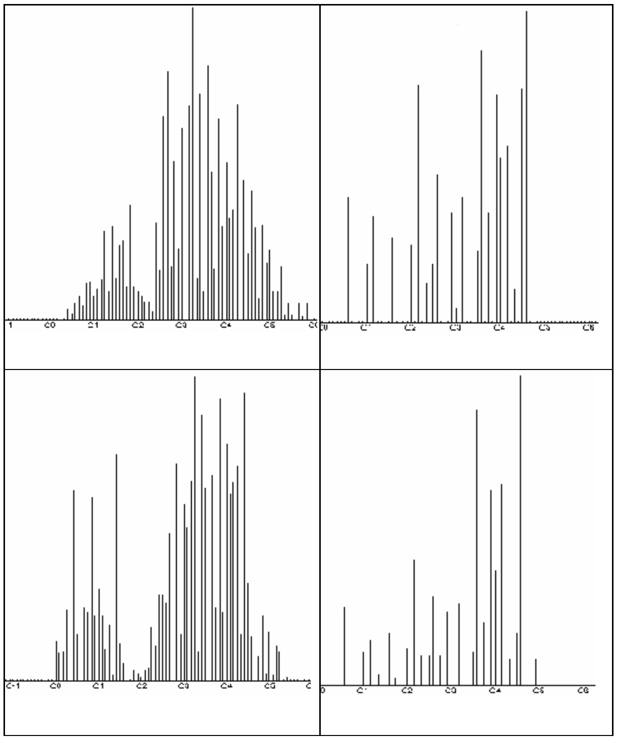

Figure 1. Unfolded Pitch Histograms of two Jazz pieces (left) and two Irish folk songs (right). The sparseness of the left side histograms results from the few chord changes of Irish folk music.

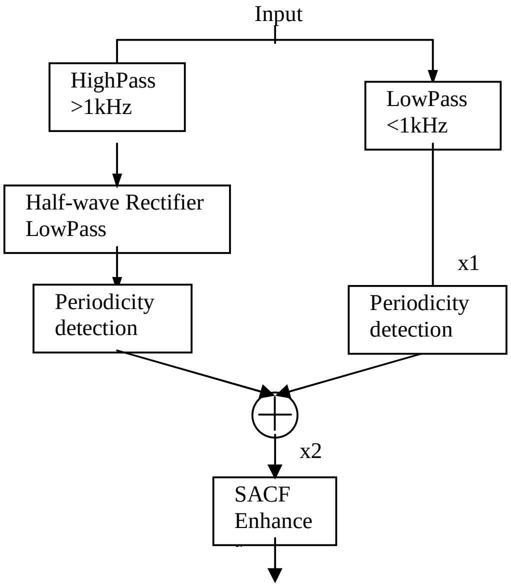

Figure 2. Multiple Pitch Detection

4. GENRE CLASSIFICATION USING PITCH HISTOGRAMS

One way of evaluating musical content features is through automatic musical genre classification. In this section, the proposed Pitch Histogram features are computed from MIDI and audio-from-MIDI representations, evaluated and the results for each case are compared.

4.1 Overview of Pattern Classification

In order to evaluate the performance of the proposed feature set, a supervised learning approach was used. Statistical pattern recognition (SPR) classifiers were trained and evaluated using a musical data set collected from various sources. The basic idea behind SPR is to estimate the probability density function (pdf) of the feature vectors for each class. In supervised learning, a labeled training set is used to estimate this pdf and this estimation is used to classify unknown data. In the described experiments, each class corresponds to a particular musical genre and the k-nearest-neighbor (KNN) classifier is used.

In the KNN classifier, an unknown feature vector is classified according to the majority of its nearest labeled feature vectors from the training set. The main purpose of the described experiments is comparing the classification performance of Pitch Histogram features in audio and symbolic form rather than obtaining the best classification performance. The KNN classifier is a good choice for this purpose because its performance is not as sensitive to the form of the underlying class pdf as that of the other classifiers. Moreover, it can also be shown that the error rate of the KNN classifier will be at most twice the error rate of the best possible (Bayes) classifier as the size of the training set goes to infinity. A proof of this statement, as well as a detailed description of the KNN classifier and pattern classification in general can be found in (Duda et. al, 2000).

4.2 Details

The five genres used in our experiments are the following ones: Electronica, Classical, Jazz, Irish Folk and Rock. While by no means exhaustive or even fully representative of all existing musical classes, this list of genres is diverse enough to provide a good indication of the amount of genre-specific information embedded into the proposed feature vectors. The choice of genres was mainly dictated by the ease of obtaining examples for each particular genre from the web. A set of 100 musical pieces in MIDI format is used to represent and train classifiers for each genre. An additional 5*100 audio pieces were generated using the Timidity software audio synthesizer to convert the MIDI files. Moreover, 5*100 general audio pieces (not corresponding to the MIDI files but belonging to the same genres) were also used for comparison and evaluation. Each file is represented as a single feature vector and 150 seconds of the file are used in the histogram calculation in all these cases.

For classification, the KNN(3) classifier is used (basically the majority label of the three nearest neighbors in the training set is used to label the unknown feature vector). For evaluation, a 10-fold cross-validation paradigm is followed. In this paradigm, the training set is randomly divided into k (=10 in our case) disjoint sets of equal size n/k, where n is the total number of labeled examples. The classifier is trained i times, each time with a different set held out as a validation set in order to ensure that the evaluation results are not affected by the particular choice of training and testing sets. The estimated performance is the mean and standard deviation of the i iterations of the cross-validation. In the described experiments, 100 iterations are used.

4.3 MIDI Representation

The classification results for the MIDI representation are shown in Figure 3, plotted against the probability of random classification (guessing). It can be seen that the results are significantly better than random, which indicates that the proposed pitch content feature set does contain a non-negligible amount of genre-specific information. The full 5-genre classifier performs with 50% accuracy, which is more than twice better than chance (20%).

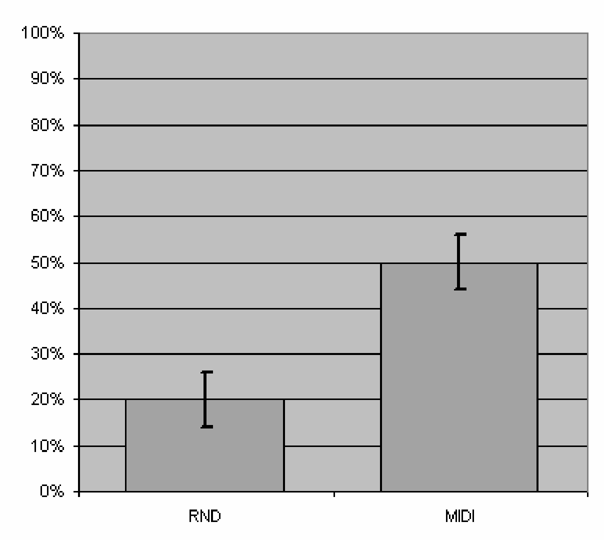

Figure 3. Classification accuracy comparison of random and Audio-from-MIDI

Table 1. MIDI genre confusion matrix (percentage values)

	Electr.	**Class.**	**Jazz**	**Irish**	**Rock**
Electr.	**32**	2	3	1	21
Class.	8	**33**	24	9	15
Jazz	9	42	**55**	2	21
Irish	12	19	8	**83**	12
Rock	39	4	10	5	**31**

The classification results are also summarized in Table 1 in the form of a so-called confusion matrix. Its columns correspond to the actual genre and the rows to the genre predicted by the classifier. For example, the cell of row 5, column 3 contains value 10, meaning that 10% of jazz (row 5) was incorrectly classified as rock music (column 3). The percentages of correct classifications lie on the main diagonal of the confusion matrix. It can be seen that 39% of rock was incorrectly classified as Electronica and the confusion between Electronica and other genres is a source of several other significant miscalculations. All of this indicates that the harmonic content analysis is not well suited for Electronica music because of its extremely broad nature. Some of its melodic components can be mistaken for rock, jazz or even classical music, whereas Electronica's main distinguishing feature, namely the extremely repetitive structure of its percussive and melodic elements is not reflected in any way on the Pitch Histogram. It is clear from inspecting the Table that certain genres are much better classified based on their pitch content than other something which is expected. However even in the cases of confusion, the results are significantly better than random and therefore would provide useful information especially if combined with other features.

In addition to these results, some representative pair-wise genre classification accuracy results are shown in Figure 4. A 2-genre classifier succeeds in correctly identifying the genre with 80% accuracy on average (1.6 times better than chance). The classifier correctly distinguishes between Irish Folk music and Jazz with 94% accuracy, which is the best classification result. The worst pair is Rock and Electronica, as can be expected, since both of these genres often employ simple and repetitive tonal combinations.

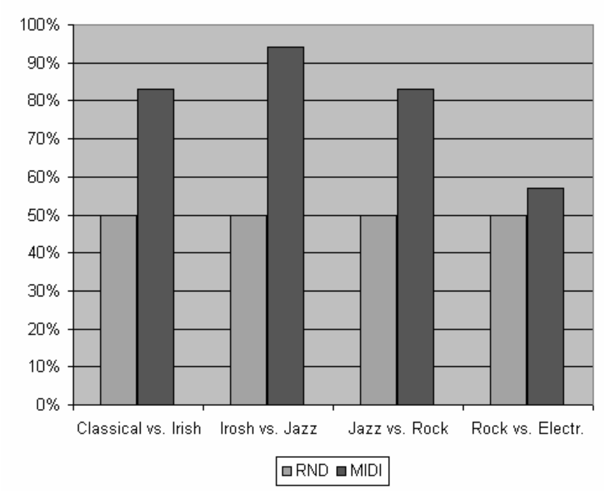

Figure 4. Pair-wise evaluation in MIDI

It will be shown below that other feature-evaluating techniques, such as the analysis of rhythmic features or the examination of timbral texture can provide additional information for musical genre classification and be more effective in distinguishing Electronica from other musical genres. This is expected because Electronica is more characterized by its rhythmic and timbral characters rather than its pitch content.

An attempt was made to investigate the dynamic properties of the proposed classification technique by studying the dependence of the algorithm's accuracy on the time-domain length of the supplied input data. Instead of letting the algorithm process MIDI files for the full length of 150 seconds, the histogram-constructing routine was modified to only process the first n-second chunk of the file, where n is a variable quantity. The average classification accuracy across one hundred files is plotted as a function of n in Figure 5.

The observed dependence of classification accuracy to the input data length is characterized by two pivotal points on the graph. The first point occurs at around 0.9 seconds, which is when the accuracy improves to approximately 35% from the random 20%. Hence, approximately one second of musical data is needed by our classifier to start identifying genre-related harmonic properties of the data. The second point occurs at approximately 80 seconds into the MIDI file, which is when the accuracy curve starts flattening off. The function reaches its absolute peak at around 240 seconds (4 minutes).

4.4 Audio generated from MIDI representation

The genre classification results for the audio-from-MIDI representation are shown in Figure 6. Although the results are not as good as the ones obtained from MIDI data, they are still significantly better than random classification. More details are provided in Table 2 in the form of a confusion matrix. From Table 2, it can be seen that Electronica is much harder to classify correctly in this case, probably due to noise in the feature vectors caused by pitch errors of the multiple-pitch detection algorithm. A comparison of these results with the ones obtained using the MIDI representation and general audio is provided in the next subsection. We have no reason to believe that the outcome of the comparison was in any way influenced by the specifics of the MIDI-to-Audio conversion procedure. Experiments with different software synthesizers for audio-from-MIDI conversion showed no significant change in the results. The main reason for the decrease in performance is due to the complexity of multiple pitch detection in audio signals even if they are generated from MIDI. Of course, no information from the original MIDI signal is used for the computation of the Pitch Histogram in audio-from-MIDI case.

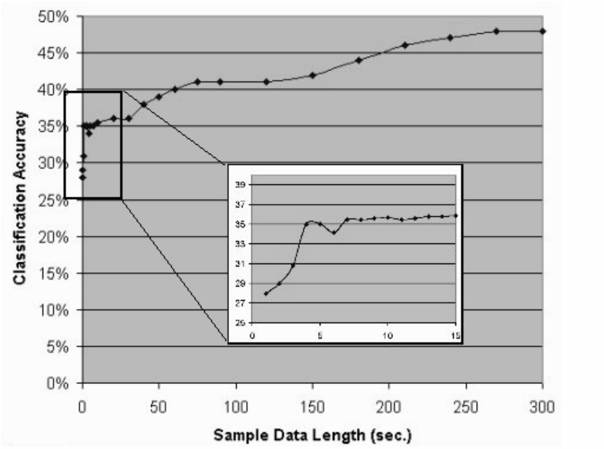

Figure 5. Average classification accuracy as a function of the length of input MIDI data (in seconds)

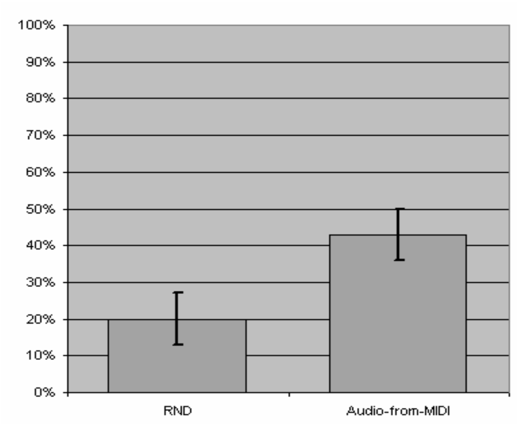

Figure 6. Classification accuracy comparison of random and Audio-from-MIDI

Table 2. Audio-from-MIDI genre confusion matrix

	Electr.	**Class.**	**Jazz**	**Irish**	**Rock**
Electr.	**9**	8	10	3	19
Class.	26	**25**	20	6	25
Jazz	30	39	**51**	6	25
Irish	19	20	9	**83**	10
Rock	16	8	10	2	**21**

4.5 Comparison

One of the objectives of the described experiments was to estimate the amount of classification error introduced by the multi-pitch detection algorithm used for the construction of Pitch Histograms from audio signals. Knowing that MIDI pitch information (and therefore pitch content feature vectors extracted from MIDI) is fully accurate by definition it is possible to estimate this amount by comparing the MIDI classification results with those obtained from the audio-from-MIDI representation. A large discrepancy would indicate that the errors introduced by multiple-pitch detection algorithm significantly affect the extracted feature vectors.

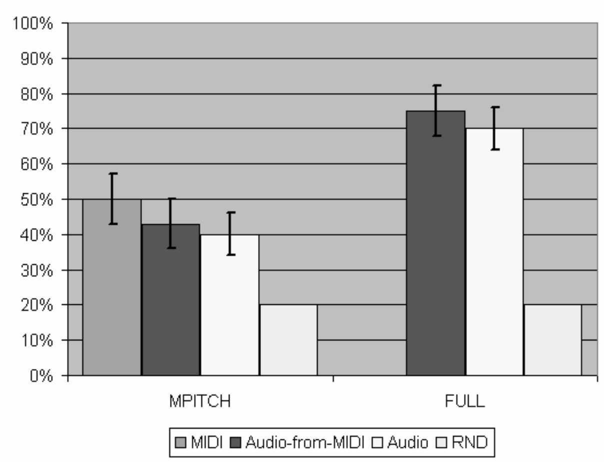

Figure 7. Classification accuracy comparison

Table 3. Comparison of classification results

	Multi-pitch Features	**Full Feature Set**	**RND**
MIDI	50 ±7%	N/A	20%
Audio-from-MIDI	43 ±7%	75 ±6%	20%
Audio	40 ±6%	70 ±6%	20%

The results of the comparison are shown in Figure 7. The same data is also provided in Table 3. It can be observed that there is a decrease in performance between the MIDI and audio-from-MIDI representations. However, despite the errors, the features computed from audio-from-MIDI still provide significant information for genre classification. A further smaller decrease in classification accuracy is observed between the audio-from-MIDI and audio representations. This is probably due to the fact that cleaner multiple pitch detection results can be obtained from the audio-from-MIDI examples because of the artificial nature of the synthesized signals. The comparison of the audio-from-MIDI and audio case is only indicative as the correspondence is only at the genre level. Basically it shows that similar classification results can be obtained for general audio signals as with audio-from-MIDI and therefore Pitch Histograms are not only applicable to audio-from-MIDI data. The detailed results of the audio classification (confusion matrix) are not included as no direct comparison can be performed with the results of the audio-from-MIDI data.

In addition to information regarding pitch or harmonic content, other types of information, such as timbral texture and rhythmic structure can be utilized to characterize musical genres. The full feature set results shown in Figure 7 and Table 3 refer to the feature set described and used for genre classification in Tzanetakis & Cook (2002). In addition to the described pitch content features, this feature set contains timbral texture features (Short-Time Fourier Transform (STFT) based, Mel-Frequency Cepstral Coefficients (MFCC)), as well as features about the rhythmic structure derived from Beat Histograms calculated using the Discrete Wavelet Transform.

It is interesting to compare this result with the performance of humans in classifying musical genre, which has been investigated in Perrot & Gjerdingen (1999). It was determined that humans are able to correctly distinguish between ten genres with 53% accuracy after listening to only 250 milliseconds audio samples. Listening to three seconds of music yielded 70% accuracy (against 10% chance). Ten genres were used for this study. Although direct comparison of these results with the described results is not possible due to different number of genres, it is clear that the automatic performance is not far away from the human performance. These results also indicate the fuzzy nature of musical genre boundaries.

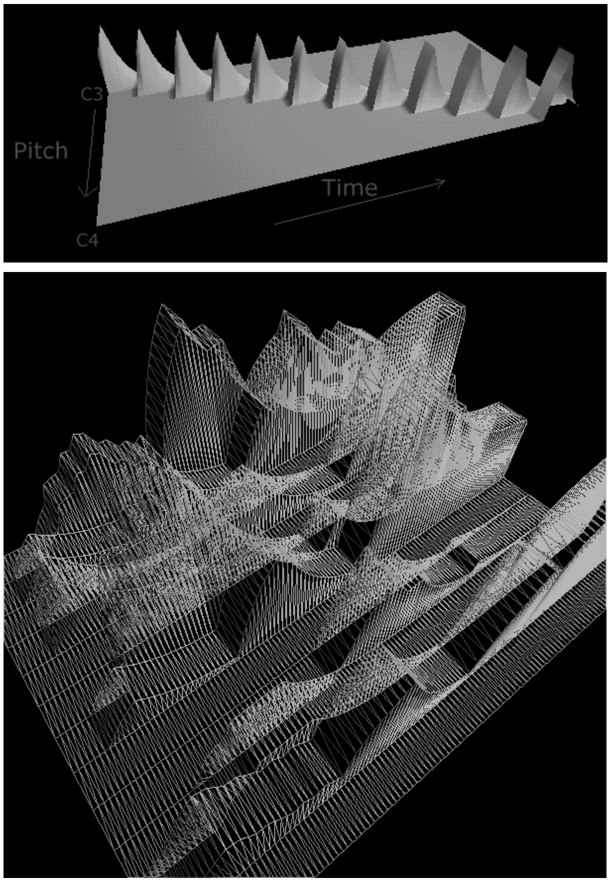

Figure 8. Three-dimensional time-pitch surface (X axis = time, Y axis = pitch, Z axis = bin amp)

5. IMPLEMENTATION

The software used for the audio Pitch Histogram calculation, as well as for the classification and evaluation, is available as a part of MARSYAS (Tzanetakis & Cook, 2000), a free software framework for rapid development and evaluation of computer audition applications. The software for the MIDI Pitch Histogram calculation is available as separate C++ code and will be integrated into MARSYAS in the future. The framework follows a client-server architecture. The server contains all the pattern recognition, signal processing and numerical computations and runs on any platform that provides C++ compilation facilities. A client graphical user interface written in Java controls the server. MARSYAS is available under the Gnu Public License (GPL) at:

http://www.cs.princeton.edu/~gtzan/marsyas.html

In order to experimentally investigate the results and performance of the Pitch Histograms, a set of visualization interfaces for displaying the time evolution of pitch content information was developed. It is our hope that these interfaces will provide new insights for the design and development of new features based on the time evolution of Pitch Histograms.

These tools provide three distinct modes of visualization:

1) Standard Pitch Histogram plots (Figure 1) where the x-axis corresponds to the histogram bin and the y-axis corresponds to the amplitude. These plots don't show the time evolution of the histogram and just display the final result.

2) Three-dimensional pitch-time surfaces (Figure 8) where the evolution of Pitch Histograms is depicted by appending histograms in time. The axes are: discrete time, discrete pitch (fold or unfolded) and the height is the amplitude of the particular histogram bin at that time and pitch.

3) Projection of the pitch-time surfaces onto a two-dimensional bitmap, with height represented as the grayscale color value (Figure 9).

These visualization tools are written in C++ and use OpenGL for the 3D graphics rendering.

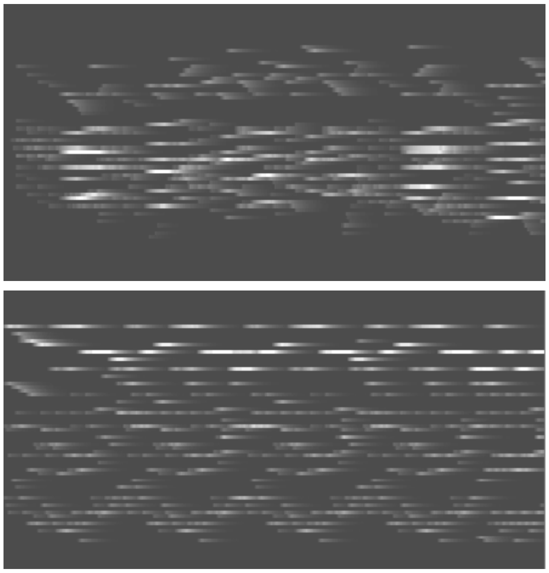

Figure 9. Examples of grayscale pitch-time surfaces: Jazz (top) and Irish Folk music (bottom), X axis = time, Y axis=pitch.

The upper part of Figure 8 shows an ascending chromatic scale of equal-length non-overlapping notes. A snapshot of the time-pitch surface of an actual music piece is shown in the lower part of Figure 8. Although more difficult to interpret visually than the simple scale example, one can observe thick slice that in most cases correspond to chords. By visual inspection of Figure 9, various types of interesting information can be observed. Some examples are: the higher pitch range of the particular Irish piece (lower part) compared to the Jazz piece (upper part), as well as its different periodic structure and melodic movement. These observations seem to generalize to the particular genres and potentially be used for the extraction of more powerful pitch content features.

6. CONCLUSIONS AND FUTURE WORK

In this paper, the notion of Pitch Histograms was introduced and its applicability in the context of musical genre classification was evaluated. A feature set for representing the harmonic content of music was derived from Pitch Histograms and proposed as a basis for genre classification. Statistical pattern recognition classifiers were trained to recognize this feature set and an attempt was made to evaluate their performance on a sample collection of musical signals both in symbolic and audio form. It was established that the proposed classification technique produces results that are significantly better than random classification, which allowed us to conclude that Pitch Histograms do carry a certain amount of genre-identifying information and therefore they are a useful tool in the context of automatic musical genre classification. To the best of our knowledge there has been no previous work that uses features that represent pitch content rather than timbral information for the purposes of MIR for audio signals. As there are no standardized collections for MIR it is still difficult to perform comparative evaluations. We are looking forward to the availability of the RWC Music Database (Goto, et al., 2002) which contains both symbolic and audio data for conducting such experiments.

Another conclusion is that, despite being a highly subjective and ill-defined procedure, musical genre classification can be performed automatically by deterministic means with performance comparable to human genre classification and pitch content information has a significant part in this process both for symbolic and audio musical signals.

A multiple-pitch detection algorithm was used to estimate musical pitches from audio signals, while the direct availability of pitch information in MIDI format made the construction of MIDI Pitch Histograms an easier process. Although the multiple-pitch detection algorithm is not perfect and subsequently causes classification accuracy degradation for the audio case, it still provides significant information for musical genre classification.

It is our belief that the methodology of using MIDI data and audio-from-MIDI data to compare and evaluate audio analysis algorithms applied in this paper can also be applied to other types of audio analysis algorithms, such as similarity retrieval, classification, summarization, instrument tracking, and polyphonic transcription. Another important contribution is the idea that an audio analysis technique does not have to give perfect results in order to be useful especially when machine learning methods are used to collect statistical information.

An interesting direction for further research is a more extensive exploration of the statistical properties of Pitch Histograms and the expansion of the pitch content feature set. For example, we are planning to investigate a real-time running version of the Pitch Histogram, in which time-domain variations of the pitch content are taken into account (see Figures 8, 9). A running Pitch Histogram contains information about the temporal evolution of pitch content that can potentially can be utilized for better classification performance. Another interesting idea is the use of the running Pitch Histogram to conduct more detailed harmonic analysis such as figured bass extraction, tonality recognition, and chord detection. The visualization interfaces described in this paper will be used for exploring the extraction of more detailed pitch content information from music signals in symbolic and audio form.

Although mainly designed for genre classification it is possible that features derived from Pitch Histograms might also be applicable to the problem of content-based audio identification or audio fingerprinting (for an example of such a system see (Allamanche et al., 2001). We are planning to explore this possibility in the future.

Alternative feature sets, as well as different multiple pitch detection algorithms also need to be explored and evaluated in the context of this work. Pitch content features also enable the specification of new types of queries and constraints, such as key or amount of harmonic change that go beyond the traditional query-by-humming (for symbolic) and query-by-example (for audio) paradigms for music information retrieval. Finally, we are planning to use the proposed feature set as a part of a query-based retrieval mechanism for audio music signals.

7. REFERENCES

[1] Allamanche, E. et al. (2001) Content-based identification of audio material using MPEG-7 Low Level Description. In Proc. Int. Symposium on Music Information Retrieval (ISMIR), Bloomington, Indiana.

[2] Baeza-Yates, R., & Ribeiro-Neto, B. (1999) Modern Information Retrieval. Harlow: Addison-Wesley.

[3] Barlow, H., & DeRoure, D. (1948). A Dictionary of Musical Themes. New York: Crown.

[4] Barthelemy, J., & Bonardi, A. (2001) Figured Bass and Tonality Recognition. In Proc. Int. Symposium on Music Information Retrieval (ISMIR), Bloomington, Indiana.

[5] Downie, J. S. (1999) Evaluating a Simple Approach to Music Information Retrieval: Conceiving Melodic N-grams as Text. Ph.D thesis, University of Western Ontario.

[6] Duda, R., Hart, P., & Stork, D. (2000) Pattern Classification. New York: John Wiley & Sons.

[7] Foote, J. (2000) ARTHUR: Retrieving Orchestral Music by Long-Term Structure. In Proc. Int. Symposium on Music Information Retrieval (ISMIR), Plymouth, MA.

[8] Ghias, A., Logan, J., Chamberlin, D., & Smith, B.C. (1995) Query by humming: Musical information retrieval in an audio database. In Proc.of ACM Multimedia, 231-236.

[9] Goto, M. et al. (2002) RWC Music Database: Popular, Classical and Jazz Music Databases. In Proc. Int. Symposium on Music Information Retrieval (ISMIR), Paris, France.

[10] Hewlett, W.B., and Selfridge-Field, Eleanor (Eds) (1998) Melodic Similarity: Concepts, Procedures and Applications. Computing in Musicology, 11.

[11] Kageyama, T., Mochizuki, K., & Takashima, Y. (1993) Melody Retrieval with Humming. In Proc. Int. Computer Music Conference (ICMC), 349-351.

[12] Krumhansl, C.L. (1990) Cognitive Foundations of Music Pitch. New York: Oxford University Press.

[13] Laroche, J. (2001) Estimating Tempo, Swing and Beat Locations in Audio Recordings. In Proc. IEEE Int. Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), 135-139, Mohonk, NY.

[14] Logan, B. (2000) Mel Frequency Cepstral Coefficients for Music Modeling. In Proc. Int. Symposium on Music Information Retrieval (ISMIR), Plymouth, MA.

[15] Pachet, F. (2000) Computer Analysis of Jazz Chord Sequences: Is Solar a Blues. Readings in Music and Artificial Intelligence, Miranda, E. Ed, Harwood Academic Publishers.

[16] Perrot, D., & Gjerdigen, R. (1999) Scanning the dial: An exploration of factors in the identification of musical style. In Proc. of the Society for Music Perception and Cognition pp.88, (abstract).

[17] Pickens, J. (2000) A Comparison of Language Modeling and Probabilistic Text Information Retrieval Approaches to Monophonic Music Retrieval. In Proc. Int. Symposium on Music Information Retrieval (ISMIR), Plymouth, MA.

[18] Scheirer, E., & Slaney, M. (1997) Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator. In Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP), Munich, Germany.

[19] Scheirer, E. (1998) Tempo and Beat Analysis of Acoustic Musical Signals. Journal of the Acoustical Society of America, 103(1):588,601.

[20] Tolonen, T., & Karjalainen, M. (2000) A Computationally Efficient Multipitch Analysis Model. IEEE Trans. On Speech and Audio Processing, 8(6), 708-716.

[21] Tzanetakis, G., & Cook, P. (2000) Audio Information Retrieval (AIR) Tools. In Proc. Int. Symposium on Music Information Retrieval (ISMIR), Plymouth, MA.

[22] Tzanetakis, G., & Cook, P., (2002) Musical Genre Classification of Audio Signals. IEEE Transactions on Speech and Audio Processing, 10(5), 293-302 .

[23] Tzanetakis, G., Ermolinskyi, A. and Cook, P., (2002) Pitch Histograms in Audio and Symbolic Music Information Retrieval, In Proc. Int. Conference on Music Information Retrieval (ISMIR), Paris, France, 31-38.

[24] Tzanetakis, G. & Cook, P. (2000) Marsyas: A framework for audio analysis. Organised Sound. 4(3), 2000.

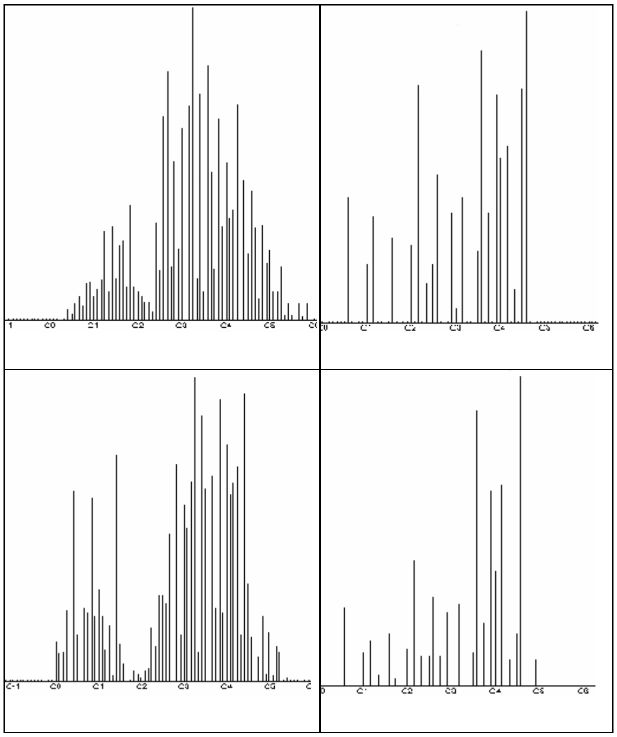

Figure 1. Unfolded Pitch Histograms of two Jazz pieces (left) and two Irish folk songs (right). The sparseness of the left side histograms results from the few chord changes of Irish folk music.

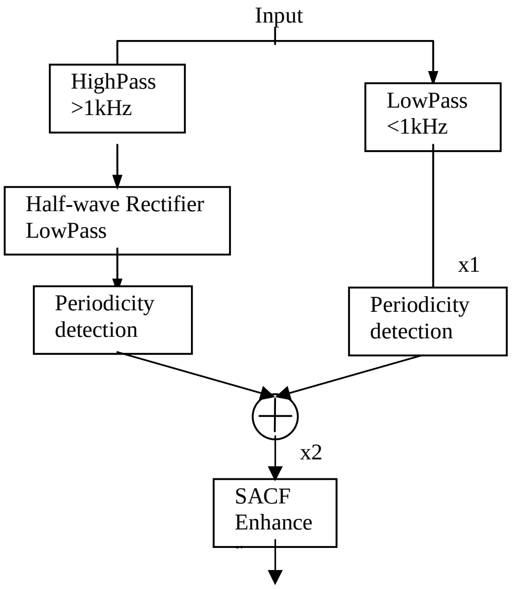

Figure 2. Multiple Pitch Detection

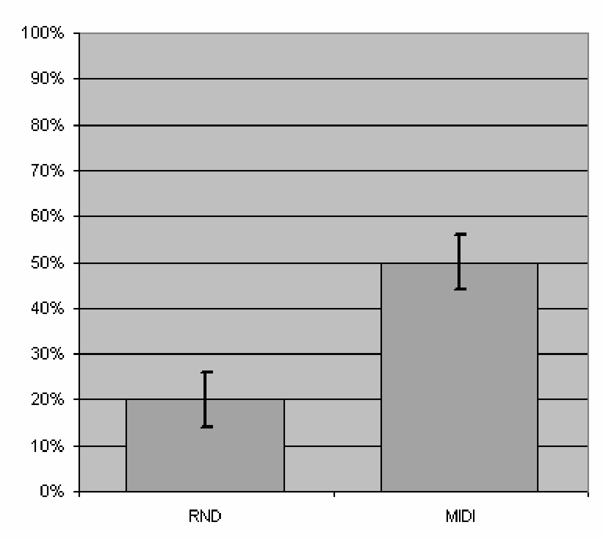

Figure 3. . Classification accuracy comparison of random and Audio-from-MIDI

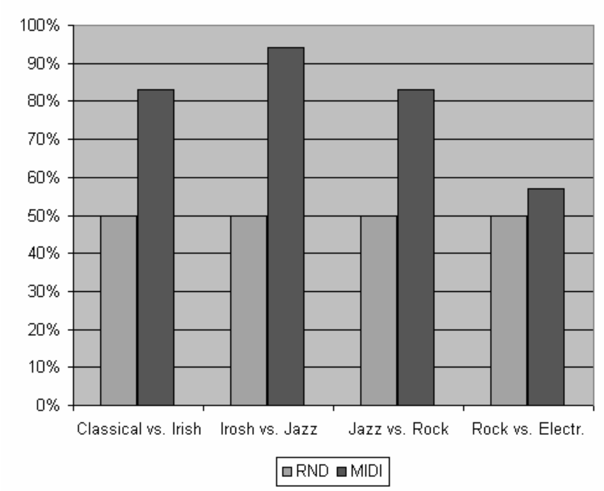

Figure 4. Pair-wise evaluation in MIDI

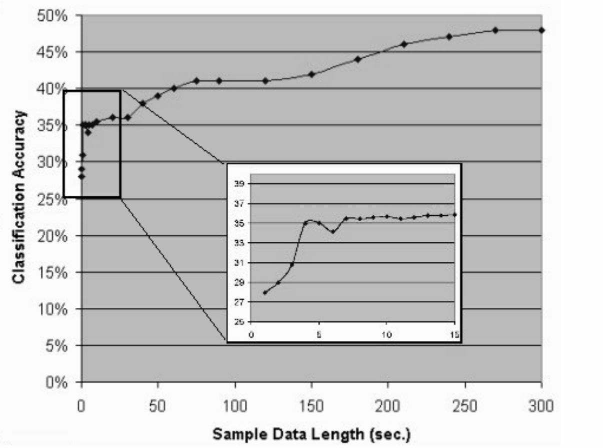

Figure 5. Average classification accuracy as a function of the length of input MIDI data (in seconds)

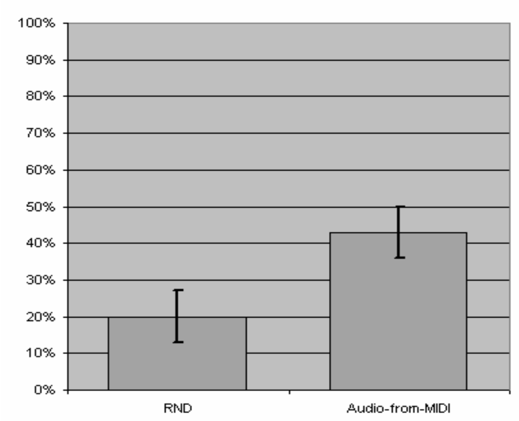

Figure 6. Classification accuracy comparison of random and Audio-from-MIDI

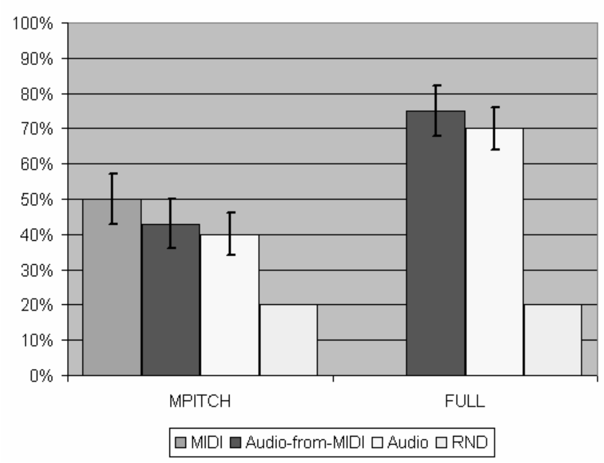

Figure 7. Classificatin accuracy comparison

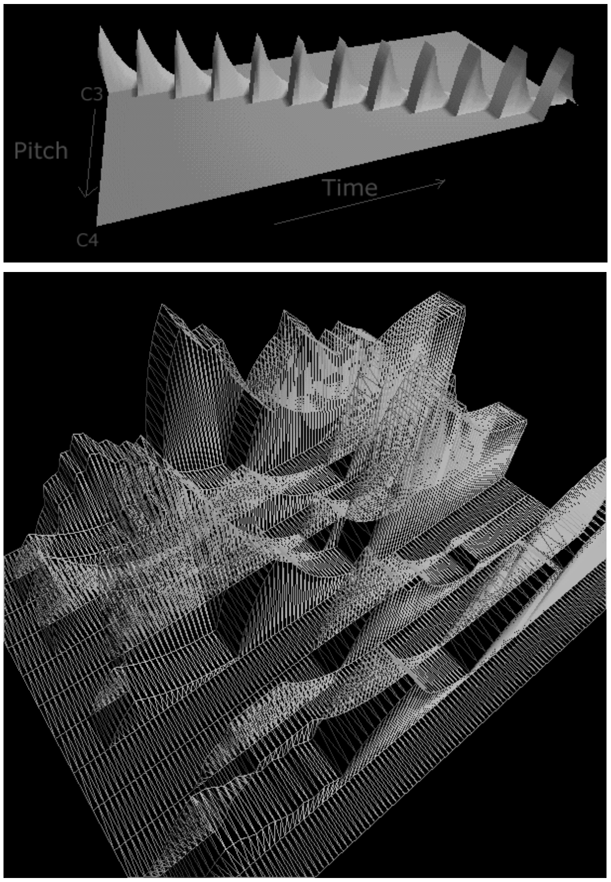

Figure 8. Three-dimensional time-pitch surface

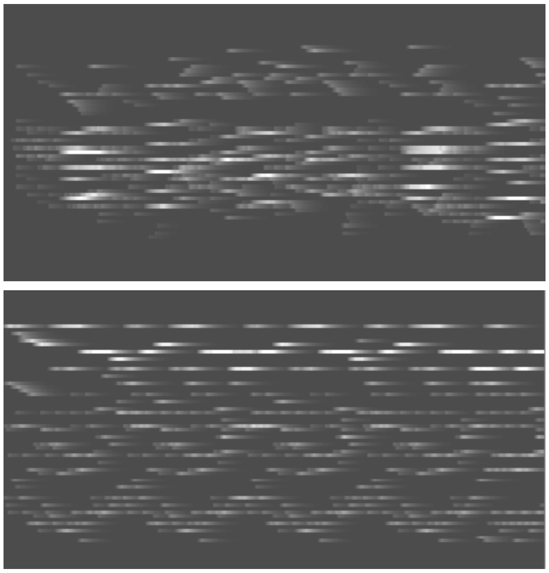

Figure 9. Examples of grayscale pitch-time surfaces: Jazz (top) and Irish Folk music (bottom), X axis = time, Y axis=pitch.