Adaptation of String Matching Algorithms for Identification of Near-Duplicate Music Documents

Matthias Robine, Pierre Hanna, Pascal Ferraro, Julien Allali

Adaptation of String Matching Algorithms for Identification of Near-Duplicate Music Documents

Matthias Robine, Pierre Hanna, Pascal Ferraro and Julien Allali
LaBRI - Universite de Bordeaux 1
F-33405 Talence cedex, France
firstname.name@labri.fr

ABSTRACT

The number of copyright registrations for music documents is increasing each year. Computer-based systems may help to detect near-duplicate music documents and plagiarisms. The main part of the existing systems for the comparison of symbolic music are based on string matching algorithms and represent music as sequences of notes. Nevertheless, adaptation to the musical context raises specific problems and a direct adaptation does not lead to an accurate detection algorithm: indeed, very different sequences can represent very similar musical pieces. We are developing an improved system which mainly considers melody but takes also into account elements of music theory in order to detect musically important differences between sequences. In this paper, we present the improvements proposed by our system in the context of the near-duplicate music document detection. Several experiments with famous music copyright infringement cases are proposed. In both monophonic and polyphonic context, the system allows the detection of plagiarisms.

1. INTRODUCTION

The number of music documents available on the World Wide Web is highly increasing. Each year, over 10000 new albums of recorded music are released and over 100000 new musical pieces are registered for copyright [19]. For example, the total number of musical pieces registered in France by the French professional association SACEM, protecting artist rights, reached 250000 pieces [7] in 2004. One of the role of this organization is to help justice to take decision about plagiarism complaints. Plagiarism is the act of copying or including another author idea without proper acknowledgment. It is important to note that a plagiarism can only be decided by justice. Some famous proceedings about plagiarism happen in the last few years: Madonna and Salvatore Acquaviva in Belgium, Georges Harrison and The Chiffons in UK, *Les feuilles mortes* and *La Maritza* in France, etc. In 2004, SACEM had only verified 18000 (out of 250000) musical pieces in order to determine their originality. A complete musical analysis is performed by experts only if a complaint is lodged. Considering the important number of new music documents registered every year, it is difficult to check for possible plagiarism. For example, a SACEM member recently registered a piece that was the perfect copy of a Ravel's piece. However, it is impossible to listen and manually compare all the music document registered.

Some studies in the context of the Music Information Retrieval research area deal with computer-based techniques that may help listeners to retrieve near-duplicate music documents and may help justice to determine plagiarisms. These investigations mainly concern the open problem of the estimation of the music similarity. The notion of similarity is very difficult to define precisely and the music similarity remains one of the most complex problem in the field of the music information retrieval. This notion may strongly depend on the musical culture, on personal opinion, on mood, etc.

From a computational point of view, evaluating the similarities consists of computing a similarity measure between a pair of musical segments. Several algorithms have been proposed for achieving such a task between audio signals. But the main of these approaches are based on timbre similarity, mainly evaluated with statistics on low-level audio features. For example, Music Browser (Sony CSL, Paris) computes a similarity measure according to Gaussian models of cepstrum coefficients [13]. However, since this information about timbre is not relevant for the copyright protection of music documents, SACEM considers musical elements such as melody, harmony or rhythm. Therefore, computer-based systems should be able to study these musical elements. Then, two problematics are raising: the extraction of musical elements from audio signals in order to define symbolic data, and comparing these data.

In this paper, we present new techniques based on edit alignment algorithms. In Section 2, we present some of the existing string matching algorithms that have been adapted to the musical context. Then in Section 3, we describe some improvements dedicated to music documents. In Section 4, we introduce different options for estimating music similarity. We present finally in Section 5 some perspectives and remaining problems in the context of the detection of near-duplicate music documents or plagiarisms.

SIGIR '07 Amsterdam. Workshop on Plagiarism Analysis, Authorship Identification, and Near-Duplicate Detection

2. MEASURING SIMILARITY BETWEEN SEQUENCES

Musical pieces can be described as sequences of elements (notes) [12]. Measuring similarity between sequences is a well-known problem in computer science which has applications in many fields such as text processing, data compression, bio-informatics [9, 15]. In this section, we treat the string matching algorithms that can be adapted to the musical context.

2.1 Musical Sequences

Several techniques for evaluating symbolic music similarities have been introduced during the last few years. Geometric algorithms consider geometric representations of melodies and compute the distance between objects.Some of these systems [20] are closely linked to the well-known piano-roll representation. Other ones represent notes by weighted points [17].

We propose here to investigate adaptations of string matching algorithms, since experiments show their accuracy and their flexibility in the musical context [8]. Such adaptation requires a representation of musical pieces as sequence. In the case of monophonic music (no more than one note is sounded at any given time), a musical piece can be associated to a sequence of integers, representing pitches of successive notes.

2.2 String Matching Algorithms

In [11], Levenshtein defines the notion of edit distance between two strings. This distance is defined as the minimum cost of all possible sequences of elementary operations (edit operations) that transform one string into the other. This distance can be computed in quadratic time $O(|S_1| \cdot |S_2|)$ and linear space using a dynamic programming algorithm [21]. A dual problem of edit distance is to compute alignment of two strings. The alignment of two strings consists in computing a mapping between the symbols of the strings. Symbols not involved in the mapping are designed as gap. The main difference between alignment and edit distance is that alignment computes a score of similarity: the highest is this score the highest is the similarity.

In many applications, two strings may not be highly similar in their entirety but may contain regions that are highly similar. In this case, the problem is to find and extract a pair of regions, one from each of the two given strings, that exhibits high similarity. This is called *local alignment* or *local similarity problem* [16]. The computation of a local similarity allows us to detect local conserved areas between both sequences. Experiments show that considering local alignment improves the quality of symbolic melodic similarity systems [8].

3. ALGORITHMIC IMPROVEMENTS FOR MUSIC DOCUMENTS

Experiments during the the first Music Information Retrieval Evaluation eXchange (MIREX 2005) [6] clearly show that the accuracy of direct application of the existing string matching algorithms is limited. That is the reason why several improvements have been recently proposed which are presented in this section.

3.1 Representations of Music

Musical pieces are associated to sequences of notes. The representation of notes is therefore an important problem. Symbolic music analysis systems generally consider the information about pitch and duration [12] which are assumed to be the two main characteristics of musical notes. Several alphabets of characters and set of numbers have thus been proposed to represent these parameters [18]. The vocabulary chosen highly depends on the application. For applications like near-duplicate music document detection, some music retrieval properties are expected. For instance, since a musical piece can be transposed and played faster or slower without degrading the melody, such systems have to be transposition invariant and tempo invariant. In the monophonic context, only a few representations enables systems to be transposition and tempo invariant: representing pitches by the difference between successive pitches (*interval*) or in the case of tonal music, by the difference between the pitch and the key of the musical piece for example.

Experiments have been performed in [8] which confirm that the *interval* parameter leads to the most precise symbolic melodic similarity system. Moreover, other experiments show that taking into account the duration of notes significantly improves such systems.

3.2 Edit Operations specific to Music

Substitution is the main edit operation and mainly determines the accuracy of the music similarity algorithm. For some applications, the substitution score is assumed as constant. However, in the musical context, this assumption must be discussed [18]. It is obvious that substituting one pitch with another one has not always the same influence on the general melody. For example, substituting a C note with a G note (fifth) slightly modifies a melody in comparison with substituting a C note with a D note. As introduced by [12] the substitution score may be correlated to the consonance interval. It has to be determined according to consonance: the fifth (7 semitones) and the third major or minor (3 or 4 semitones) are the most consonant intervals in western music. Experiments show that this choice significantly improves algorithms [8].

Other improvements have been experimentally shown. For example, considering the note duration for the calculation of the insertion/deletion scores improves the quality of the similarity systems. Indeed, the insertion of a half note may disturb more significantly a melody than the insertion of a sixteenth note.

3.3 Weighting by Taking into Account Music Theory

We think that a preliminary music analysis may highlight the properties that help listeners to perceptually discriminate two musical patterns. This analysis may therefore lead to the modification of edit operations specific to music. For example, the notes located on the stronger beats in a bar can be considered as more important than the other ones

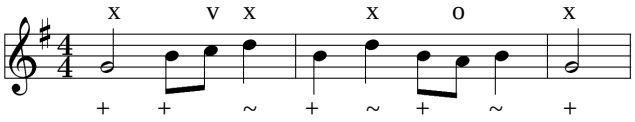

Figure 1: Analysis of a musical piece allows to identify the different functions of the notes and their placement inside the bar. Above the notes, "x" tags the importance of the note regarding the tonality limited to the tonic and the dominant tones (respectively G and D for a G Major tonality here). "v" is used to identify the passing note and "o" for a note on the weak part of the beat (which is not a passing note). Under the staff, "+" stands for the strong beats and "~" for the weak ones.

and can be weighted more than the notes placed on weak beats.

In [14], we proposed to use some notions of music theory to improve the edit-based systems. A few musical elements are analyzed and taken into account during the calculation of the edit score.

Tonality: One of the most important characteristics of the traditional western music is the tonality. The tonic is the pitch upon which all the other pitches of a piece are hierarchically centered. The scale associated to a tonality begins by the tonic. In western tonal music, the tonic and the dominant are very important. They are often used and their succession composes for example the perfect cadence that commonly ends a musical piece. In the G major or in the G minor key, tonic is the note G and dominant is the note D, like in the example of the Fig. 1. Therefore, the alignment algorithm proposed takes into account the tonic and the dominant: if the difference in semi-tones (modulo 12) between each note of the melody and the tonic equals 0 (the tonic note) or 7 (dominant), the note is assumed to be important and is therefore marked. The musical sequence alignment favours matches between these marked notes.

Passing Notes: The algorithm proposed in [14] detects the passing notes in a musical piece. A passing note is assumed as a note between two others in a constant movement (ascending or descending) which is diatonic or chromatic. There is one occurrence of a passing note in Fig. 1. The edit scores are computed according to the information about the passing notes so that the insertion or the deletion of passing notes is less penalized by the similarity system.

Strong and Weak Beats: The bar is a segment of time in a musical piece defined as a given number of beats of a given duration. In function of their position in the bar, the beats can be strong or weak with parts that are also strong or weak. We have proposed to mark the notes placed on the beats. A weight is associated to each of these notes, depending of the strength of the beat. In 4/4 time, the strong beats are the first (a weight 4 is given), and the third (weight 2) of the bars. Other beats are weighted with 1, and the other notes, which are not on the beats, are not weighted. An example of the different strengths is illustrated in Fig. 1. Our algorithm takes into account these weighted notes by favouring matches between notes on strong beats, and by not penalizing insertion or deletion of notes on the weak part of the beat.

3.4 Adaptation to Polyphony

To take into account the polyphonic nature of musical sequences, we propose to use a quotiented sequence representation. Formally, a quotiented sequence is a sequence graph with an equivalence relation defined on the set of vertices, such that the resulting quotient graph is also a sequence. A quotiented sequence can be considered as a self-similar structure represented by sequences on two different scales. A quotiented sequence can also be modelled by a tree of depth 2 where the leaves represent the support sequence and the interior nodes represent the quotient sequence. In the context of polyphonic music, notes that occur at the same time are grouped to form a quotiented sequence $Q = (S, W, \pi)$ where S is a suite of notes, W the suite of chords and π the application that maps a set of notes to each chord. Each vertex of the quotiented sequence is labelled by the pitch and the duration of each note. [10] has proposed two distances between quotiented sequences based on the computation of an optimal suite of edit operations that preserves equivalence relations on sequence vertices.

Furthermore, as previously explained, since a near-duplicate musical piece can be transposed (one or several times) without degrading the melody, algorithms for detecting near-duplicate music have to be transposition invariant. Thus, [2] proposes an original dynamic programming algorithm that allows edit based algorithms to take into account successive local transpositions and to deal with transposed polyphonic music.

3.5 System for Detecting Near-Duplicate Music Documents

According to the improvements presented in this section, we developed an edit-distance based algorithm for estimating similarity between symbolic melodic fragments. It allows us to consider a musical piece (or a fragment) and compare it to a symbolic music database. The system presented computes an edit score by comparing the musical piece tested and all the pieces of the database. The more important the score is, the more similar the pieces compared are. This system have already been evaluated in the last few years. It obtains the very accurate results with MIREX 2005 training database [8]. It also participated to the MIREX 2006 contest and obtained the best results in the monophonic context. Differences with other edit-distance based algorithms show that the optimizations proposed, specific to the musical context, permit to significantly improve such algorithms.

4. MUSIC SIMILARITY

In this section, we propose to illustrate with examples the different ways for automatically evaluating the musical similarity between musical pieces. We consider some famous examples of plagiarisms in order to show that a computer-based method is able to automatically detect near-duplicate

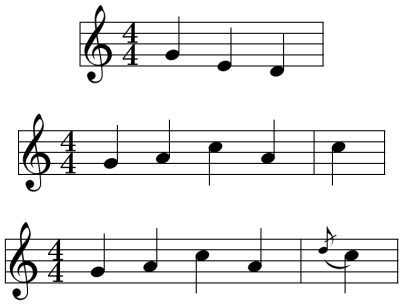

Figure 2: Short musical motifs composing the structure of the two songs *My Sweet Lord* (G. Harrison) and *He's So Fine* (R. Mack): motif A (top), motif B (middle) and motif C (bottom).

music documents. Two different approaches are investigated with systems considering melody and harmony.

4.1 Melodic Similarity

Two of the main characteristics of western music are rhythm and melody. Symbolic musical pieces are here represented by sequences of notes (see Section 2). The presented tests concern music copyright infringement cases in the United States in the last few years [5].

One of the most famous proceedings about music plagiarism concern George Harrison and his song *My Sweet Lord* that was released in 1970 on the album *All Things Must Pass* [1]. He was suspected for plagiarism of the song *He's So Fine* composed in 1963 by Ronald Mack and performed by The Chiffons. Although Harrison explained that he did not knowingly appropriate the melody of this song, the court concluded in 1976 that he had – maybe unconsciously – copied the melody of *He's So Fine*.

In order to take its decision, the court looked at the structure of the two songs. Fig. 3 shows two fragments of each of these songs. *He's So Fine* is composed of four variations of a short musical motif (motif A, Fig. 2), followed by four variations of motif B (Fig. 2). The second use of the motif B series includes a unique grace note, illustrated in motif C (Fig. 2). *My Sweet Lord* has a very similar structure in that it is composed of four variations of motif A, followed by three variations of motif B. The fourth variation of motif B includes the grace note illustrated in motif C.

The first experiments consider these two songs. Fig. 3 shows two excerpts of them. We note that even if the two melodies sound very similar, the excerpts of the melody are really different. The query of the system is defined as a part of the melody of the plagiarism *My Sweet Lord*. The database of musical pieces considered is the database proposed during MIREX 2006, *i.e.* the UK subset of the RISM A/II collection (about 15,000 incipits). The RISM A/II (International inventory of musical sources) collection is composed of one half-million notated real world compositions. The incipits are symbolically encoded music. They are monophonic and contain between 10 and 40 notes. The database also contains the monophonic melodies of *My Sweet Lord* and *He's So Fine*.

The first query corresponds to the structure considered by the court, *i.e.* repetitions of motifs illustrated by Fig 2. The second query is the excerpt of *My Sweet Lord* associated to three repetitions of motif A. The third query is the excerpt associated to the three repetitions of motif B then one motif C. The fourth query is the excerpt associated to motif A followed by motif B. Finally, the two last queries correspond to long excerpts of the monophonic melody of *My Sweet Lord* and *He's So Fine*. Tab. 1 shows the name of the most similar pieces found in the database with these different queries and their corresponding score. The scores associated to the three estimated most similar pieces are presented. The results obtained are the ones expected at the exception of the second query. In this case, the melody of *He's So Fine* is ranked far from the top 3 (the score obtained is 25.5). The little size of the motif A certainly justifies this error. For all the other queries, the most similar piece detected is the melody of *My Sweet Lord* (or *He's So Fine* for the last query), which only shows that the detection system is perfectly able to retrieve a piece from an exact excerpt. More interestingly, the second piece estimated as the most similar is the melody of *He's So Fine* (*My Sweet Lord* for the last query). Although the two sequences representing the two melodies are very different (see Fig. 3), the system proposed is able to detect their musical similarity. The two melodies seem to be also different from the structure composed of the motifs considered by the court (first query). Nevertheless, here again, the system succeeds in retrieving the two melodies. It is also important to note the difference between the scores of rank 2 and 3. As expected it becomes very significant (83 instead of 52 or 45) when the whole melody is considered, since the sequence of notes is longer.

Query	**rank** 1 **score** 1	**rank** 2 **score** 2	**rank** 3 **score** 3
Motif AAABBBC	Sweet Lord 79.6	So Fine 65.4	X 52.6
AAA from My Sweet Lord	Sweet Lord 44.2	X 30.9	X 29.5
BBBC from My Sweet Lord	Sweet Lord 113.3	So Fine 56.6	X 52.9
AB from My Sweet Lord	Sweet Lord 44.7	So Fine 33.3	X 29.8
Sweet Lord melody	Sweet Lord 178.9	So Fine 83.0	X 52.2
So Fine melody	So Fine 199.7	Sweet Lord 83.0	X 45.5

Table 1: Results of experiments about the detection of the near-duplicate monophonic musical pieces *My Sweet Lord* and *He's So Fine* (X indicates a piece that does not sound similar to the query).

In order to confirm the results of these first experiments, we propose to consider another monophonic database, which is composed of long musical pieces. This database groups more than 1650 various MIDI files collected on the internet. All these files are monophonic. Four other music copyright infringement cases are now considered [5]. For each of the five cases, the monophonic melody is proposed as query, and the system computes all the scores for all the pieces of the database (which contains these melodies). Tab. 2 shows the

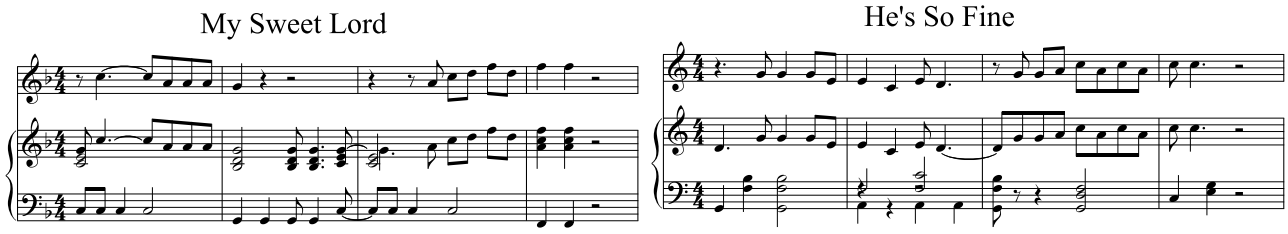

Figure 3: Manual transcriptions of excerpts (corresponding to motif A and motif B) of the two songs *My Sweet Lord* (G. Harrison) and *He's So Fine* (R. Mack).

results obtained by our system (top 3 with associated similarity scores). As expected, the first musical piece of the database estimated as the most similar is the query. The score of the rank 1 thus corresponds to the maximum score. Here, the most important result is the ranked 2 piece. Ideally, it has to correspond to the melody associated to the plagiarism established by he court. Tab. 2 shows that it is always the case, at the exception of the case *Fantasy vs Fogerty*. This error shows the limitations of the current system (see Section 5 for discussion). For all the other cases, the detection system gives the results expected. For cases like *Selle vs Gibb* or *Heim vs Universal* for example, the similarity is evaluated as important. However, the limitations of the system are also shown by the little difference between ranked 2 and ranked 3 scores for the case *Repp vs Webber*. The low score for the rank 2, corresponding to the near-duplicate piece, induces low differences between this score and the other ones obtained with the other pieces of the database. That's why more musical elements have certainly to be considered in order to reduce these differences and to make the system more robust.

We only performed a few experiments with polyphonic musical pieces. The polyphonic database considered is the *MIDI karaoke* database used during MIREX 2006, which is composed of 1000 pieces collected on the internet. The only experiment performed considers the monophonic melody of *My Sweet Lord*. The detection system compares this monophonic melody to all the polyphonic pieces contained in the MIDI karaoke database. Tab. 3 shows that *He's So Fine* has been still detected as the musical piece of the database the most similar to *My Sweet Lord*. However, in the polyphonic context, the limitations of our system are highlighted. The probability of detecting a high similarity with long polyphonic pieces is more important than with monophonic pieces, because all the notes are taken into account by our system. If the similarity score between two corresponding pieces is low in the monophonic context, the system may not correctly evaluate their similarity in the polyphonic context. For example, with *He's So Fine* as query, the system does not succeed in retrieving the corresponding polyphonic piece (*My Sweet Lord* obtains a score equals to 107.6 whereas the ranked 2 score is 141.3). At the contrary, if the similarity is more important in the monophonic context (for example *My Sweet Lord*), the system succeeds in detecting the near-duplicate polyphonic piece. Here again, the main conclusions are that the system succeeds greatly for some cases, but needs improvements. Considering other musical elements may certainly improve the system in both monophonic and polyphonic contexts.

Query	**rank** 1 **score** 1	**rank** 2 **score** 2	**rank** 3 **score** 3
R. Mack vs G. Harrison (1976)			
Sweet Lord	Sweet Lord 178.9	So Fine 83.0	X 77.5
So Fine	So Fine 199.7	Sweet Lord 83.0	X 75.3
Fantasy vs Fogerty (1994)			
Road	Road 168.9	X 87.6	Jungle 75.9
Jungle	Jungle 146.3	Road 75.9	X 75.5
Heim vs Universal (1946)			
Vagyok	Vagyok 248.6	Perhaps 123.5	X 92.8
Perhaps	Perhaps 215.5	Vagyok 123.5	X 76.8
Repp vs Webber (1997)			
Till You	Till You 135.5	Phantom 50.8	X 50.4
Phantom	Phantom 145.8	Till You 50.8	X 49.7
Selle vs Gibb (1984)			
Let It End	Let It End 192.4	How Deep 118.1	X 68.9
How Deep	How Deep 202.8	Let It End 118.1	X 83.8

Table 2: Results of experiments about the detection of the near-duplicate monophonic musical pieces for a few music copyright infringement cases.

4.2 Harmonic Similarity

Taking only the melody into account may not be sufficient to identify near-duplicate music documents. Let us take an example: a famous french case of plagiarism concerns the musical pieces *Les feuilles mortes* (internationally known as *Autumn leaves*) and *La Maritza*. As we can see on Fig. 4, even if the two pieces are perceptively very similar, a lot of notes are inserted in *La Maritza* regarding to *Les feuilles mortes*. The composer of *La Maritza* has been recognized guilty of plagiarism offense by a french court. Algorithms presented in the previous sections could strongly identify this kind of plagiarism. It is a human music expert that influenced this judgment by exposing the similarities between the two different music scores. His conviction was based on a music analysis of the scores and a look for some duplicated

Query	rank 1 score 1	rank 2 score 2	rank 3 score 3
Sweet Lord	Sweet Lord 160.3	So Fine 96.1	X 89.2
So Fine	So Fine 178.7	X 141.3	X 137.8

Table 3: Results of experiments about the detection of the near-duplicate polyphonic musical pieces *My Sweet Lord* and *He's So Fine* from monophonic melody.

motifs. In fact, he highlighted few similar sequences of notes with the same intervals used. He considered that the chord progression is the same for the refrains of the two musical pieces and that all the notes inserted in *La Maritza* could be considered as ornaments (musical flourishes that are not necessary to the overall melodic or harmonic line). Thus, even if few notes are common to the two musical piece, they are important regarding the harmony.

Therefore, we think that one possibility of improvement would be to base the comparison of two musical pieces first on their harmony. It would consist in finding the different chords that compose each piece before to perform a string matching on the sequences of these chords (on their name, as illustrated by the chord sequence on Fig. 4). All the ornaments and non-chord tone which can be added in a copied document from the original would not be considered (we can call it *melodic noise* in this context). The first step consists of extracting the sequence of the chords for a musical piece. In [3] a model for the tonality of a musical piece is proposed, and some methods to analyse the chord progression from the MIDI format are presented. Extracted chord sequences could then be compared with algorithms of string matching presented in Section 2. As these methods had been successfully evaluated in a musical context for the melody, we expect to obtain again some good results. As previously, we could improve the system by taking into account some musical considerations : the sequence may be invariant considering the tonality for example (a chord sequence C D E is similar to F G A) and the notion of consonance interval could be used as presented for the melody in Section 3. On the same way, a different level of matching may concern the key sequence of a musical piece. When the key of a piece is not constant, there are some modulations, and the musical piece can be segmented in different parts regarding the key (each part is composed with several chords). It could be done with methods proposed in [3, 4] to segment a musical piece in key sequences from a MIDI file.

We may therefore match a music document at least on three levels : one for comparing the melodic sequences, one for the chord sequences and the third for the key sequences. Let us imagine what could be the main interest of using all these levels for detecting similarities and near-duplicate documents. All the musical pieces registered in the world, the music inserted in movies, video games or websites constitute a huge music database in which a high level matching could allow to look for similarities as a filter. Only the pieces that would be similar on high levels, with a same chord progression for example, could be compared at the melodic level. It also gives a way to deal easily with polyphonic sounds reduced to a monophonic sequence of chords. Although the harmony of two similar musical pieces is generally very similar, it is not always true and this approach may complement the comparison at the melodic level. On another way, some pieces have the same chord progression without plagiarism. The matching of the chord sequences could therefore be used for looking for musical variations for example.

5. PERSPECTIVES FOR NEAR-DUPLICATE DETECTION

Existing algorithms that can be applied to detect near-duplicate music documents rely on string matching or geometric algorithms. Results obtained with such algorithms are quite good if the musical sequences are nearly the same. When studying a few music copyright infringement cases, it appears that musical sequences composed of very different notes can be musically very similar. Therefore, we have proposed some improvements specific to the musical context. Elements of musical theory have to be taken into account in order to improve the existing systems. The first experiments proposed in the previous section show that, when considering these improvements, edit-based systems are able to detect plagiarisms. Nevertheless, some limitations have been shown with some examples. Therefore, we propose some new perspectives by considering both melody, rhythm and harmony.

We have exposed several representations of a musical piece with the aim of finding similar pieces in a database. Concerning the representation of a melody in a monophonic or polyphonic context, we expect to test the impact of each factor of similarity – intervals, rhythm, harmonic function of the notes – and to evaluate how these parameters are independent and could be combined. The combination which is used for the moment is only a first step. We can also imagine to match sequences for each of these parameters independently. The system could give normalized results as score of similarity which could be used in different ways. One possibility would be to obtain a probability of plagiarism offense which can be finally confirmed by a human. A second possibility would be to test the similarity regarding a special parameter only if the precedent score regarding another parameter was over a threshold of similarity.

Furthermore, other musical rules than in [14] are needed to be implemented for considering and detecting the ornaments and non-chord tones that are less important in a musical piece to detect a near-duplicate document. We also aim at improving and evaluating our methods in the polyphonic context.

We expect to implement the hierarchical model we have presented in Section 4.2 to compare efficiently a great number of music documents using three different levels : melodic, chord and key level. We aim at finding the best method to use this model in the plagiarism domain with using the upper levels as filters in a big music database of polyphonic documents for example.

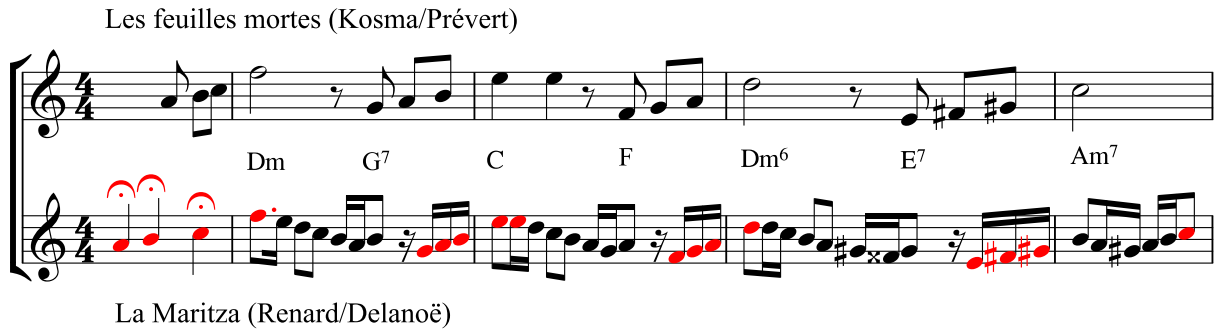

Figure 4: Manual transcriptions of excerpts of the two songs *Les feuilles mortes* and *La Maritza*. All the notes of the melody from *Les feuilles mortes* are also present in the Maritza's melody (red notes). The inserted black notes in *La Maritza* can be considered as ornaments.

6. REFERENCES

[1] *Copyright Website*. `http://www.benedict.com/Audio/Harrison/Harrison.aspx`.

[2] J. Allali, P. Hanna, P. Ferraro, and C. Iliopoulos. Local Transpositions in Alignment of Polyphonic Musical Sequences. 2007. Submitted.

[3] E. Chew. *Towards a Mathematical Model of Tonality*. PhD thesis, MIT Cambridge, MA, 2000.

[4] E. Chew. Regards on Two Regards by Messiaen: Automatic Segmentation using the Spiral Array. In *Proceedings of the Sound and Music Computing Conference (SMC)*, Paris, France, 2004.

[5] Columbia Center for New Media Teaching and Learning. *Music Plagiarism Project*. `http://ccnmtl.columbia.edu/projects/law/library/`.

[6] J. S. Downie, K. West, A. F. Ehmann, and E. Vincent. The 2005 Music Information retrieval Evaluation Exchange (MIREX 2005): Preliminary Overview. In *ISMIR*, pages 320–323, 2005.

[7] P. Emberger. Dossier SACEM : Le Livre Blanc. In *Keyboards Magazine*, volume 200, Sep 2005.

[8] P. Ferraro and P. Hanna. Optimizations of Local Edition for Evaluating Similarity Between Monophonic Musical Sequences. In *Proceedings of the 8th International Conference on Information Retrieval - RIAO 2007, Pittsburgh, PA, USA*, May 2007.

[9] D. Gusfield. *Algorithms on Strings, Trees and Sequences - Computer Science and Computational Biology*. Cambridge University Press, 1997.

[10] P. Hanna and P. Ferraro. Polyphonic Music Retrieval by Local Edition of Quotiented Sequences. In *Proceedings of the 5th International Workshop on Content-Based Multimedia Indexing (CBMI'07)*, Bordeaux, France, June 2007. To appear.

[11] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Sov. Phys. Dokl.*, 6:707–710, 1966.

[12] M. Mongeau and D. Sankoff. Comparison of Musical Sequences. *Computers and the Humanities*, 24(3):161–175, 1990.

[13] F. Pachet, J.-J. Aucouturier, A. La Burthe, A. Zils, and A. Beurive. The cuidado music browser : an end-to-end electronic music distribution system. *Multimedia Tools and Applications*, 2006. Special Issue on the CBMI03 Conference.

[14] M. Robine, P. Hanna, and P. Ferraro. Music similarity: Improvements of edit-based algorithms by considering music theory. *Internal report RR-1433-07, LaBRI, University of Bordeaux 1*, 2007.

[15] D. Sankoff and J. B. Kruskal, editors. *Time Wraps, Strings Edits, and Macromolecules: the Theory and Practice of Sequence Comparison*. Addison-Wesley Publishing Company Inc, University of Montreal, Quebec, Canada, 1983.

[16] T. Smith and M. Waterman. Identification of Common Molecular Subsequences. *Journal of Molecular Biology*, 147:195–197, 1981.

[17] R. Typke, R. C. Veltkamp, and F. Wiering. Searching Notated Polyphonic Music Using Transportation Distances. In *Proceedings of the ACM Multimedia Conference*, pages 128–135, New-York, USA, 2004.

[18] A. L. Uitdenbogerd. *Music Information Retrieval Technology*. PhD thesis, RMIT University, Melbourne, Victoria, Australia, July 2002.

[19] A. L. Uitdenbogerd and J. Zobel. Matching Techniques for Large Music Database. In *Proceedings of the ACM International Conference on Multimedia*, pages 56–66, Orlando, Florida, USA, 1999.

[20] E. Ukkonen, K. Lemström, and V. Mäkinen. Geometric Algorithms for Transposition Invariant Content-Based Music Retrieval. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR'03)*, pages 193–199, Baltimore, USA, October 2003.

[21] R. A. Wagner and M. J. Fisher. The String-to-String Correction Problem. *Journal of the association for computing machinery*, 21:168–173, 1974.