Constituting a Musical Sign Base through Score Analysis and Annotation

Véronique Sébastien, Didier Sébastien, Noël Conruyt
LIM - Laboratoire d'Informatique et de Mathématiques, EA2525
University of Reunion Island
Saint-Denis, La Réunion (FRANCE)
veronique.sebastien/didier.sebastien/noel.conruyt@univ-reunion.fr

***Abstract* - The recent progress in Information and Communication Technologies has given birth to advanced applications in the field of instrumental e-learning. However, most of these applications only propose a limited number of lessons on predetermined pieces, according to the vision of a single music expert. Thus, this article introduces a web platform to create music lessons dynamically and collaboratively, with the assistance of a semi-automatic score annotation module: @-MUSE. To do so, we first describe a new methodology to design such a platform: Sign Management. Then, we detail its general architecture as an Iterative Sign Base System based on a common practice in musical learning: score annotation. Lastly, we give various algorithms to generate relevant annotations on a score in order to explain it. These algorithms are based on the analysis of musical patterns difficulty. They are implemented within a module of @-MUSE called Score Analyzer. We present here its first results.**

Keywords - e-learning; knowledge management; sign management; multimedia; semantic web; musical score; music information retrieval; decision support

I. INTRODUCTION

Information and Communication Technology for Education (ICTE) expanded rapidly these last years. Indeed more and more teachers resort to platforms such as Moodle or Blackboard to design their own online courses. While this trend is being confirmed to learn academic disciplines such as mathematics and languages [10], it remains rare for know-how transmission and sharing, for instance in the field of music learning. Indeed, know-how transmission requires heavy multimedia usage and interaction to show the “correct gesture” and is thus complex to implement.

In music, some instrumental e-learning solutions exist in the form of offline tools, such as instructional DVDs (see the technical report of E-guitare [25]), or business software (Guitar Pro® [26], GarageBand® [27]). Nevertheless, getting a feedback from the teacher is capital in know-how acquisition: "is my gesture correct?". But few applications try to implement a learner to teacher communication axis through video upload and commentaries on the Web (see the FIGS [28] glosses system). Still, the lessons provided by these platforms remain limited to a fixed list of pieces. Although a student can suggest a new title, the realization of a whole lesson on these platforms requires heavy installations and treatments (multi-angle video recording, 3D motion capture), as well as the intervention of multiple actors other than the teacher himself. While these methods produce high quality teaching material, the realization of a new course remains a complex and expensive process. In parallel, several teachers, for instance retired experts, wish to transmit their know-how in a simple way, without any constraint on the recording location and time, and with minimal efforts for tool appropriation.

We thus introduce in this paper a complementary framework to rapidly create dynamic music lessons on new pieces with multimedia annotations [1]. This framework for music learning is called @-MUSE (Annotation platform for MUSical Education). As described in [18], an online annotation system is chosen because it allows musicians to work with digital scores in a way similar to traditional lessons, where scores are a support for memory and information sharing [24]. In addition, the digital transposition of this common practice enables to enrich it with multimedia incrustation, collaborative working and mobility. As such, its aim is also to constitute a scalable music playing sign base (see part II) to collect and share tips and performances on all possible artistic works referenced on music data warehouse such as MusicBrainz.org [29], and which can evolve according to the learners’ needs, whatever their level may be. Indeed, most of the existing music learning applications target beginners and do not provide an environment adapted to the teaching of advanced instrumental techniques. This sign base is generated with ISBS (Iterative Sign Base System), which aim is to define the structure of pieces with an ontology, describe them with a questionnaire, then capture interpretations with @-MUSE in order to preserve and share experts know-how. For the time being, the chosen field is music, and more precisely piano techniques, as it is a very demanding and historically rich domain. Still, the conceived system and its principles remain largely applicable to other instruments. Besides, as the authors are musicians themselves, it is easier for them to experiment and interact with professionals from this field. Indeed, the proposed methods and experimentations presented in what follows result from a collaboration with teachers and students from the Conservatory of Music of Reunion Island.

In this paper, we first introduce the methodology and principles of Sign Management that supports ISBS. Then, we describe the general architecture of @-MUSE, the ISBS annotation module designed to constitute a Musical Sign Base (MSB). To assist users into feeding and exploiting this base, we describe various methods to generate relevant annotations (i.e. explanations) on a score. These annotations are generated according to descriptive logics used by pianists when they study a new piece. This method is mainly based on extracting main parts of a piece and detecting its inherent difficulties. Therefore, after reviewing practical rules to structure a piano score, we present our algorithms to automatically analyze its playing difficulties. Lastly, we present Score Analyzer, a module of @-MUSE which implements the methods and algorithms we conceived to detect score and performance difficulties, and discuss its first results.

II. METHODOLOGY: SIGN MANAGEMENT

Sign Management deals with the management of know-how rather than knowledge. It manages live knowledge, i.e., subjective objects found in interpretations of real subjects (individuals) on the scene (live performances) rather than objective entities found in publications on the shelf (bookish knowledge). A Sign is a semiotic and dynamic Object issued from a Subject and composed of three parts, Data, Information and Knowledge. All these subjective components communicate together to build a chain of "sign-ifications" that we want to capture.

Sign management is thus more central than Knowledge management for our purpose in instrumental music learning. Indeed, the musical signs to treat are made of emotional content (performances), technical symbols (scores) and tacit knowledge (rational and cultural know-how). Thus, a Sign is the interpretation of an object by a subject at a given time and place, composed of a form (Information), content (Data) and a sense (Knowledge). The sign management process that we have created is made on a Creativity Platform for delivering an instrumental e-learning service [6][7][17]. It is founded on an imitation and explanation process for understanding gestures that produce a right and beautiful sound. The advantage for learners is that we are able to decompose the teacher's movement and understand the instructions that are behind the process of playing a piece of music. In fact, a lovely interpretation is made of a lot of technical and motivated details that the learner has to master, and the way we want to deliver this information is to show examples from experts through multimedia annotations indexed on the score. To do so, we introduce a new module to design dynamic music lessons through multimedia annotations: @-MUSE.

Indeed, as shown on Figure 1, an annotation can be considered as a structure including all three components of a sign: a symbolic form (the "written" document, which can be a score, or a tablature or lyrics in music), a substance (for example a video showing the musical performance) and a meaning (the explanation from the annotator point of view).

Annotations thus become a support to enable fruitful dialogs between users on @-MUSE. In order to let users compose lessons in a dynamic way, we propose the semantic architecture proposed in part III.

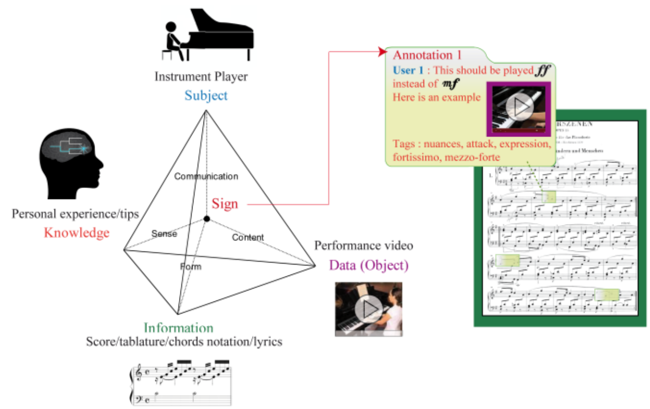

Figure 1. Sign management on @-MUSE

III. @-MUSE GLOBAL ARCHITECTURE

As the aim of @-MUSE is to enable dynamic teaching and learning through annotations, it is capital that its architecture remains flexible. The use of Semantic Web tools is thus an appropriate lead to allow the platform to benefit from a "networking effect". Indeed, a significant amount of scattered musical resources already exist on the Web and can be relevant in the context of music lessons. These resources can be music metadata (MusicBrainz.org), digital scores (images, PDF, MusicXML free or proprietary files available on Werner Icking Archive [30]), multimedia documents (video recordings of performances and lessons on YouTube [31] or eHow [32]) or simple textual comments. They constitute the different sign components listed in part II: data, information and knowledge. As many of these resources benefit from a Creative Commons License [33], they can be used in the context of a music lesson, complementary to high quality resources from a professional multimedia capture set [7]. Figure 2 exposes a comparison between architectures of traditional instrumental e-learning applications and @-MUSE. In the first case, lessons are defined in a static way. Each one corresponds to a musical piece, with its associated resources: video, audio and image files synchronized together to form the lesson. While this system produces complete instructions, it cannot establish relations between two distinct resources or pieces, which is an essential point when learning music as a whole. In the second case, @-MUSE dynamically creates lessons by linking related resources posted by users and presenting them through an adapted interface [18]. If a resource is not available (for instance, a logic representation of a score), the system still works with a temporary replacement (for instance a simple image representing the score) in the frame

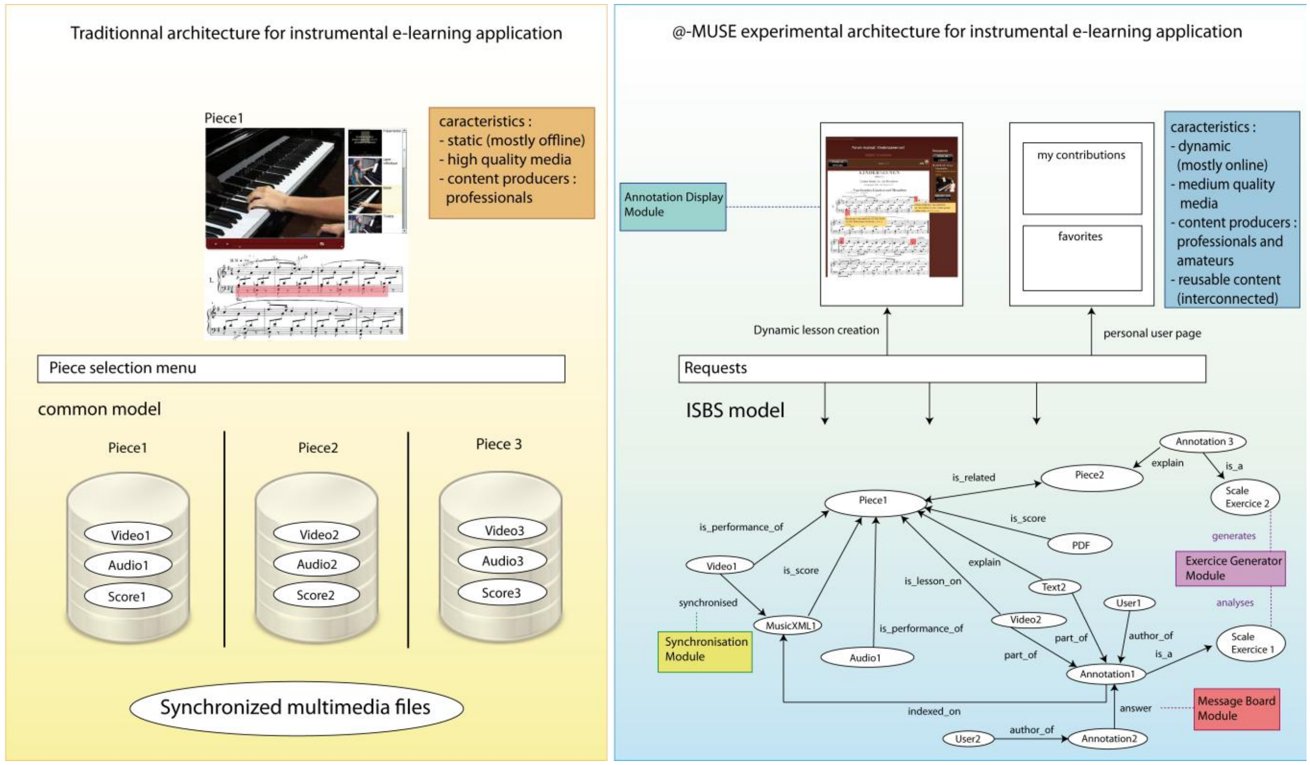

Figure 2. Architecture comparison between traditional instrumental e-learning application and @-MUSE

of a degraded mode. It can then point to any user the need to provide such resource to enable new functionalities on the platform. As more links are created between resources, different representations of the same piece can be proposed to learn how to play it. Some links such as a time synchronization between two representations (e.g. a video performance and a logical description of the score) can be realized by specific independent modules (Figure 2).

We have done previous work in [19] to propose an adapted ontology to link musical resources in an educational context using the Resource Description Framework (RDF [12]). This allows people to tag their annotations with appropriate concepts, such as "Technical exercise", "Harmony", "Fingering", etc. The idea is to generalize existing annotations to include them on other relevant pieces. As such, a learner may start a new piece on his own, and still dispose of basic information, thanks to these semi-automatically generated annotations (see part IV.D).

In the end, the association of these elements will allow the creation of an Iterative Sign Base System for music, in the same vein as IKBS (Iterative Knowledge Base System [5]) for environmental data. The difference here lies in the manipulation of semiotic objects (signs), instead of conceptual ones (knowledge), as described in part II. The following chapter explains how new signs can be generated on this platform through semi-automatic score annotation, and thus participate in the enrichment of the MSB by demanding minimal efforts from the users of @-MUSE.

IV. KNOWLEDGE EXTRACTION ON DIGITAL SCORES

ISBS is a sign base model designed to collect musical signs such as scores (model) and performances (cases), in order to explain and compare them. A score can be considered as a database containing musical information. In this frame, score-mining represents the applications of data-mining methods to one or several digital scores. Thus, our aim is to infer knowledge by analyzing these particular data [23]. To realize such analysis in a semi-automatic way, we need to detect specific patterns within a score. This detection could be made directly on performances [21] but audio signal analysis algorithms are difficult to implement in a Web-application and may be less precise than those based on symbolic representations in an educational context. That is why we rely on XML representations of a score to design our inference system. MusicXML [3] is an XML open source format to describe digital scores staff by staff, measure by measure, and lastly note by note (Figure 3).

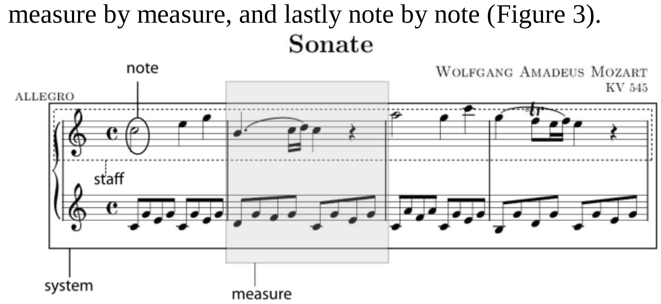

Figure 3. Score logical structure

In what follows, we review and propose different methods to extract various playing information from a piece metadata and structure, for educational purposes. We base these methods on how a pianist would address an unknown piece. As detailed in the descriptive model presented in [19], the musical work is first replaced in its context (composer, period, form metadata). Its global structure is then determined in order to visualize how the piece is shaped. This step may also be helpful to design a work schedule adapted to the piece. Then, its difficulty is evaluated, firstly globally (tempo, length), and then part by part, in order to determine what type of work can be made on this piece and where. The following parts present methods to set up these different steps in the frame of @-MUSE. Then, we propose several tracks to exploit the detected difficult parts to generate relevant annotations on the score. In the last part of this chapter, we present Score Analyzer: a module of @-MUSE designed to automatically determine the difficulty level of piano pieces, and discuss its results.

A. *Musical work context analysis*

The context of a musical work provides several pieces of information on how to play it. Information metadata such as its title or composer, present in the MusicXML file, allow to obtain more information about its genre. Indeed, music Web Services, such as MusicBrainz, Amazon, Last.fm, etc. query a piece by title and composer to identify it. This allows to extract metadata on a piece, for example a portrait and biography of its composer, or an indication about the piece style with a link to the corresponding Wikipedia page if it exists.

Several performances of the piece can also be requested on video sharing websites such as YouTube. Still it is important to evaluate the quality of these resources before posting it on educational platforms. Communities such as MuseScore.com [38] allow its users to choose the appropriate video by themselves and then synchronize it to their scores, thus insuring some reliability between content and form. All these additional multimedia resources contribute to the creation of a complete music lesson environment.

B. *Form and structure analysis*

To play a piece of music correctly, it is very important to grasp its structure. Indeed, the pianist's playing can change drastically from one part of the piece to another (especially on advanced pieces, which may contain several contrasted parts). Moreover, specific behaviors are often expected according to the encountered notes pattern. For example, a pianist is often taught to slow down at the end of the piece, to breathe between two phrases, or to augment the volume at the end of an ascendant arpeggio. For our purpose, structure detection is also important to refine annotations indexation. It allows musicians to annotate directly a phrase or a pattern, without having to indicate it as such beforehand. Moreover, it enables advanced searching on the pieces base (i.e. by musical pattern, by melody, by form [2]) and sensibly refines the automatic difficulty estimation.

However, the granularity scale to structure a musical work is large and complex, as it can go from whole pieces collections to simple notes. To guide users through a precise musical work structuring process, we propose the descriptive model shown in Figure 4 which follows some descriptive logics. We consider a *Musical Work* (or *Collection*) to be composed of several *Pieces* (for example a Sonata including three movements, i.e. three "sub-pieces"), each *Piece* being constituted of different *Parts* (for example, a theme and its developments). Each *Part* may contain several *Subparts* (recursive definition). A piece *Part* is a group of *Notes*, and can be designated as a "*Theme*", a "*Melody*", a "*Voice*", or a "*Phrase*" thanks to an appropriate Musical Form Taxonomy to create specialized part names. A *Part* may contain several repeated and/or remarkable *Patterns*, each one being composed of several *Notes* (or *Rests*). The descriptive model designed in this way can match any musical work, from the simplest to the most complex ones.

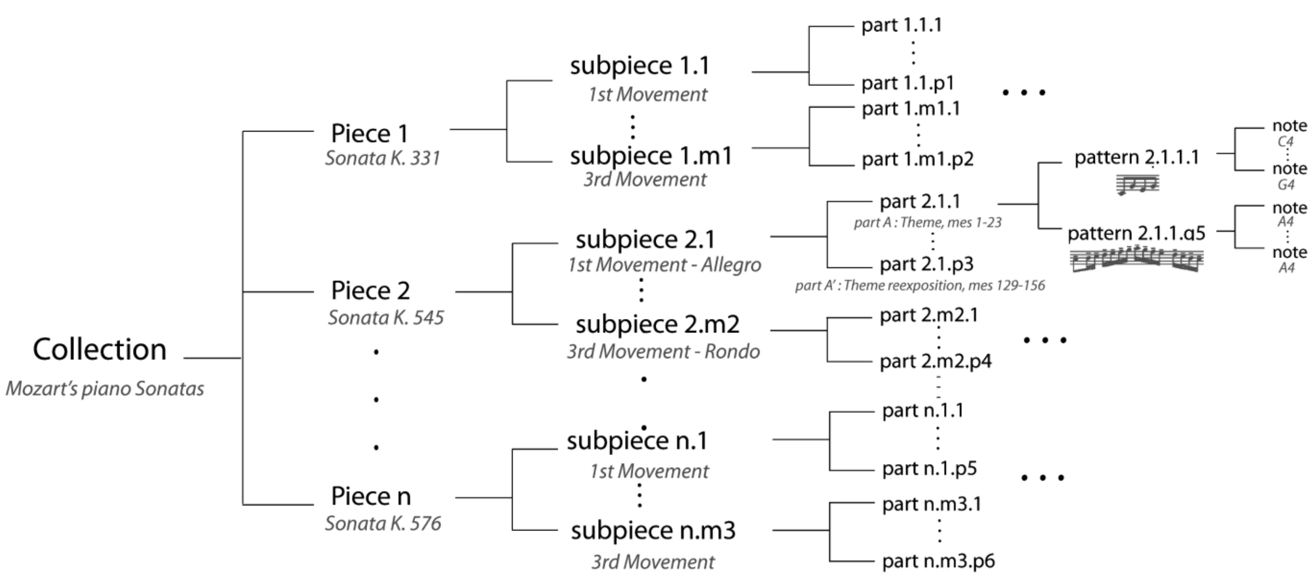

Figure 4. Musical piece structure descriptive model

To define the piece's structure as illustrated previously in a Semantic Web context, we need an appropriate Ontology. As Music Recommendation Systems are more and more popular, several works exist to tag musical content. For instance, the Music Ontology [14] allows to semantically describe pieces metadata. It is particularly fitted for the Music Industry and is largely used on web radios such as Last.fm [35] in order to describe *Artists*, *Albums* and *Tracks*. Concerning music structuring, it allows to tag a song in order to indicate its *Chorus* and *Verse* on a *TimeLine* [15]. This *TimeLine* can be synchronized with a score to obtain a symbolic decomposition of the piece. However, the Music Ontology does not provide a spectrum of structures large enough to be used in our frame (especially on classical music).

Its extension, the Symbolic Music Ontology [15], presents two interesting concepts for us: the *Voice* and the *Motif* (i.e. a pattern) concept, each designing a group of notes. Indeed, indicating voices on a score is an interesting feature in the frame of an educational platform such as @-MUSE, especially for polyphonic instruments. A *Motif* is a small group of notes which occurs repeatedly through the piece. In folk music, a *Motif* can be related to a type of dance by its rhythmic characteristics. For example, the siciliana gave its name to the eponymous rhythm pattern. As such, it constitutes an interesting annotation material.

The Neuma Ontology [16] is more specialized than the Music Ontology. It defines a *Fragment* entity allowing the separation of a piece into different parts, which can then be described thanks to a dedicated taxonomy. A *Fragment* has a *Beginning*, an *End*, both indicated as measure numbers, and several *Voices*. As Neuma is specialized in Gregorian music, it can define *Fragments* as *Phrases*, *Sub-Phrases*, or *Melody*. But to match a larger spectrum of musical pieces, more concepts have to be added, such as *Theme* and *Development* (necessary to describe Bach's Fugues for example), *Introduction*, *Variations*, *Coda*, etc.

To sum up, we dispose of various solutions to fragment a symbolic representation or a performance of a musical piece, but not to identify the created fragments appropriately, and thus uncover the form of the piece. Besides, The Music Ontology is already the base for several extensions related to music, such as the Chord Ontology, the Symbolic Music Ontology, or the Instrument Taxonomy [15]. Therefore, we intend to use the Music Ontology as a base to build a Musical Form Taxonomy in future works.

While users can enter this structuring information manually through score annotation, it is interesting to study how we can assist them on this operation through automated tools. Indeed, on a MusicXML score, several methods can be used to automatically detect and extract some of the parts described previously. The simplest method is based on symbols detection. Indeed, score symbols such as direction texts (e.g. "meno mosso"), tempo and key modifications, double bars generally indicate the beginning of a new part within the piece (Figure 5). This method gives acceptable

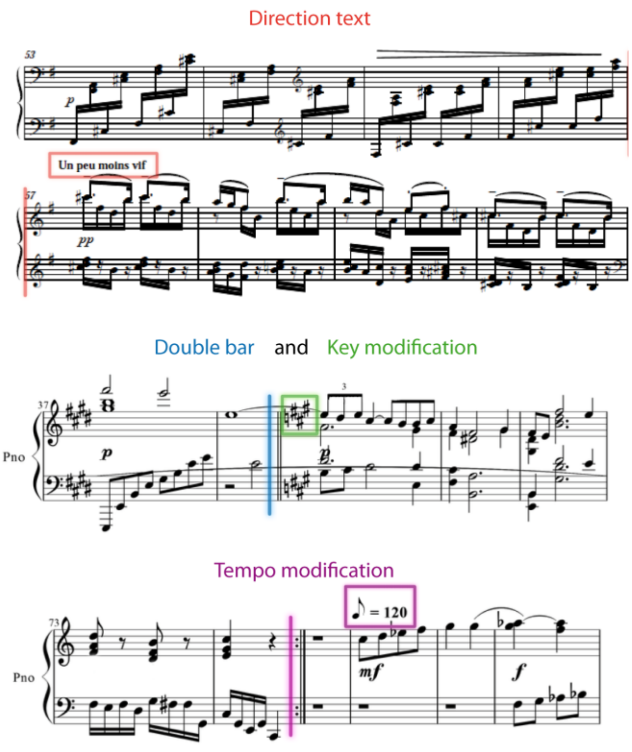

Figure 5. New parts detection on a score

results most of the time. Some exceptions may occur, especially on contemporary pieces, which often present unconventional structures.

The second method, more complex, is based on how musicologists and musicians generally cut a piece, according to melodic and harmonic features. The most representative harmonic feature marking the end of a musical phrase is called a cadence (characteristic chords sequence). Detecting cadences within a MusicXML consists in identifying specific harmonic sequences. Thanks to this method, we can identify more specific phrases.

For most tunes, repetitive patterns may be identified within phrases, sometimes with slight differences. For instance, Beethoven's Fifth Symphony starts off with the repetition of one of the most famous musical pattern (Figure 6). As the harmony evolves through the piece, the rhythm and the intervals of the pattern remain unchanged. Yet, pattern detection in music is much more complex than the given example, as it does not only involves rhythms and pitch features, but also polyphonic ones. Moreover, it does not present a fixed definition of "similarity", in opposition

Figure 6. Musical patterns on a score (extract from the Fifth Symphony by Beethoven)

to text patterns detection. Two fragments can be considered as “similar”, without having the same pitches (for example, the first two patterns in Figure 6). Several works exist on Musical Pattern Discovery [23]. Among them, [11] presents a method based on time windows and define different types of patterns (abstract patterns, prefixes, patterns network).

Discovering predefined patterns is easier if we know what features we are seeking for (mainly regular intervals). Table I gives some examples of *patterns*. They are not specific to a particular piece, as they can appear in any pieces through different sorts. For instance in jazz and classical music, *scales* are so common that they provide specific exercises collection [8]. In MusicXML files, using a memory window of successive intervals may identify these patterns. Identified sequences correspond to one of the pattern listed in Table I. For instance, arpeggios correspond to a sequence of thirds, scales to a sequence of ascendant or descendant seconds, and trills to a sequence of alternated ascendant and descendant seconds. As shown in part D, the identification of these patterns plays an important role in guiding a learner through annotations generation.

TABLE I. MUSICAL PATTERNS EXAMPLES

Pattern name	Example
Scale	
Arpeggio	
Trill	
Real sequence	

Melodies identification within polyphonic music can be linked to the problem of *Voices* detection. Sometimes, *Voices* are indicated in MusicXML files. Humans create them with a software (i.e. Finale® [36] or MuseScore® [37]). But most of the time, this indication is not given, and the *melody* should be extracted by identifying its characteristic features (form, length, occurrences within the piece, etc). Work is in progress concerning this issue.

C. *Difficulty analysis*

Once the general structure of the piece has been identified, we then analyze its technical difficulty, first globally, and then part by part. In Table II, we propose seven criteria affecting the level of a piece for piano and detail how they can be estimated from a MusicXML file. Globally, a piece difficulty depends on its tempo, its fingering, its required hand displacements, as well as its harmonic, rhythmic and polyphonic specificities. Of course, these various criteria affect each other in a complex manner. For example, hand displacement is strongly affected by fingering, as noted in Table II.

Indeed, among these seven criteria, fingering plays an important role. Several works present methods to automatically deduce fingering on a given musical extract for piano ([4][13][9]). Most of them are based on dynamic programming. All possible fingers combinations are generated and evaluated, thanks to cost functions. The latter are determined by kinematic considerations. Some functions, even consider the player’s hand size to adjust its results, such as in [9]. Then, expensive (in term of effort) combinations are suppressed until only one remains, which will be displayed as the resulting fingering. While the result often differs from a fingering determined by a human expert, it remains largely playable and exploitable in the frame of an educational usage. However, few algorithms can process polyphonic extracts [9], and many other cases are ignored (i.e., left hand, finger substitutions, black and white keys alternation).

Even if more work is needed on this issue, the use of cost functions remain relevant as it is close from the process humans implicitly apply while working on a musical piece. That is why we extend this idea and create complementary criteria to design a piece difficulty analyzer for piano learning. For each criterion described in Table II, a score is calculated in percentage.

The speed of playing *P* is determined as a percentage of the speed of the fastest possible piece. This speed has been fixed to a tempo of 176 beats per minute for a quarter note, multiplied by a shortest note value of sixteen (sixteenth note value). This value *s* was estimated after a selection of piano pieces renowned for their fast tempi. To avoid insignificant short values (i.e. trills), only values occurring on more than 15% of the total number of notes are taken into account. As shown in Part E, this calculation gives results close to a pianist evaluation of a piece playing speed. To determine the proportion of difficult displacements, we first search all positions changing within the MusicXML file. Indeed, two successive note elements (<note>) in the file do not necessarily imply a new hand position as these notes can belong to the same chord (and thus be played at the same time), be tied, or be a rest (<note></rest>). Once we are sure that two successive <note> elements correspond to a hand movement, we estimate its realization cost. First, we calculate the length of the gap in semitones, then the time imposed to realize the movement and lastly, the required fingering. The fine tuning of these three parameters allow to get a precise estimation of the displacement degree of difficulty. Chords, alterations and irregular rhythms are detected through XML parsing (see Table II).

The piece difficulty level is thus the average rate of each criterion. Furthermore, some weighting coefficients can be affected to each criterion to reflect the particularities of the player. For instance, pianists who are really at ease with polyrhythm would not consider it a relevant factor, thus affecting it a 10% weight. However, we insist that the resulting difficulty rate should be interpreted with care and remains a simple approximation. As stated in [22], a nice performance is not a mere addition of criteria since it contains an important subjective part such as morphological or physical facilities, psychological attention or concentration, etc. Still, it proposes an interesting approximation of a piece level, especially for large scores databases such as Free-scores.com [39].

Although the presented criteria were modeled after piano players experience, they can be adapted to others instruments. For instance, the fingering criterion can be transposed to the guitar by switching cost functions, and

TABLE II. PLAYING DIFFICULTY CRITERIA IN PIANO PRACTICE

Performance difficulty criterion	Musicological definitions	Cost function definition	Examples	MusicXML implementation
Playing speed	Tempo: speed or pace of a musical piece. May be indicated by a word (ex: allegro) or by a value in BPM (Beats Per Minute) Pulsation: reference value indicated in the tempo: 𝅝 = 1, 𝅗𝅥 = 2, ♩ = 4, ♪ = 8, 𝅘𝅥𝅯 = 16, etc.	$\text{Playing speed} = \frac{\text{tempo} \times \text{shortest value}}{176 * 16}$ With all tempi using a quarter note ♩ as a reference Unit: percentage of quickest playable piece (fixed at 176 𝅘𝅥𝅯)	*P1*: tempo = 120 ♪ = 60 ♩ Shortest value = 𝅘𝅥𝅯 $P1\ playing\ speed = \frac{60 \times 16}{176 \times 16} = 34\ \%$ *P2*: tempo = 120 ♩ Shortest value = 𝅘𝅥𝅯 $P2\ playing\ speed = \frac{120 \times 16}{176 \times 16} = 68\ \%$	<note><type> elements Tempo attribute in <sound> element
Fingering	Fingering: choice of finger and hand position on various instruments. Different notations exist according to the instrument. (ex: in piano: 1 = thumb, 2 = index finger, 3 = middle finger, etc.) See [4][9][13][20] for more detail.	If *m1*, *m2*, ..., *mn* represent the measures of a given piece *P*, $\text{Fingering\_difficulty(P)} = \sum_{i=1}^{i=n} \text{Fingering\_cost}(m_i) > 50$	*P* =  Fingering_cost(m_1) = 10 Fingering_cost(m_2) = 0 Fingering_cost(m_3) = 70 Fingering_difficulty(*P*) = 70	<measure> and <note> elements
Hand Displacement	Interval: pitch distance between two notes, in semitones. A hand displacement is considered difficult when two successive positions are spaced by more than 12 semitones (7 if played by close fingers on the same hand) within a short time interval. The displacement cost of an interval increases with its gap length	If *P* is a piece containing *n* displacements, and among them *s* difficult displacements (*s*<*n*), $\text{Displacement\_difficulty(P)} = \frac{s}{n}$	*P* =  Displacement_difficulty(*P*) = 7 / 17 = 41 %	Combined <note> elements where <pitch> gap > 12. Associated fingering file.
Polyphony	Chord: aggregate of musical pitches sounded simultaneously.	Proportion of chords and chords sequences in the piece	*P* =  Chords_proportion(*P*) = 6/16 = 38%	<chord> element
Harmony	Tonality: system of music in which specific hierarchical pitch relationships are based on a key "center", or tonic. Various tonalities impose various sharps and flats as a key signature. The most basic ones (no alteration) are A minor and C major.	Proportion of altered notes	*P* =  Altered_notes_proportion(*P*) = 3/25 = 12%	<alter> and <accidental> elements
Irregular Rhythm	Polyrhythm: simultaneous sounding of two or more independent rhythms. Example: synchronizing a triplets over duplets	Proportion of remarkable polyrhythm patterns (Time reference = pulsation)	 *P* = Polyrhythm_proportion(*P*) = 4/4 = 100%	<time-modification> element
Length	The length of the piece in beats. NB: the number of pages cannot really reflect the length of a piece because of page setting parameters	Number of measures × number of beats per measure.	*P* =  Length(*P*) = 3*3 = 9	<beats> element of <time> element and <measure> elements

hand displacements by adapting the gap threshold according to a representative set of guitarists.

In the following part, we study how the criteria of Table II can be used to generate relevant annotations on difficult parts of a new piece added on our @-MUSE platform.

D. Semi-automatic annotation generation

The previous form, structure and difficulty analysis can be merged to serve as a basis for the next chain of knowledge extraction on the given piece. Indeed, the criteria proposed previously can also be used on fractions of the piece.As such, for a given part, if one of the rate is abnormally high, we can infer that it presents a characteristic difficulty, and thus recommend an appropriate technical exercise to the player (Figure 7).

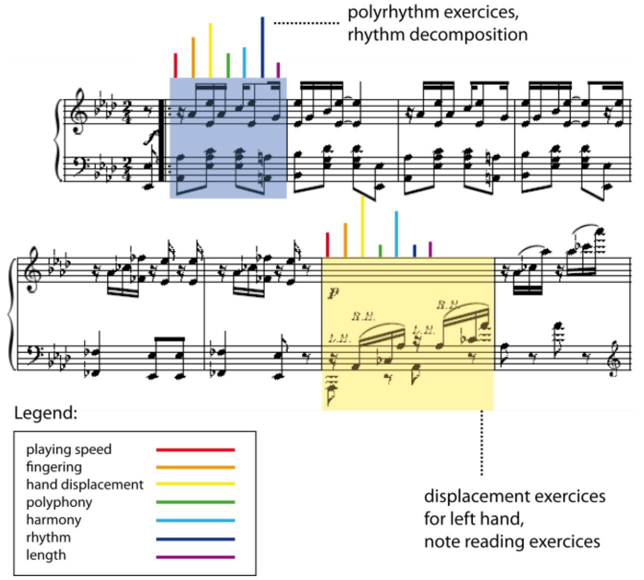

Figure 7. Difficulty analysis and recommendations on a digital score

Identified patterns are associated to corresponding exercises to guide the learner. In this case, most musical exercises consists in decomposition and repetition of subparts of the pattern. For instance, in the case of an arpeggio, the latter will be extended to the whole keyboard and repeated part by part, by adding a new note every ten repetitions. This process can easily be computed as suggested by Figure 9. These exercises can be adapted to the occurrence of the pattern by observing its tonal and rhythmic features.

But at anytime, the annotation's owner and teachers can modify it in order to improve the given explanation with textual and video commentaries, symbols and tags. Users can also invalidate the generated annotation if they consider it as being inappropriate. In this case, the motive for the suppression should be specified. This data will be later used to determine the reasoning error in order to improve the next generated annotations. All in all, automatically generated annotations should be clearly stated as such to users, for example with different colors, to avoid any confusion between public and private knowledge, validated or not by experts, annotations from users with unidentified level, computed or personalized knowledge. That is why, at each step of investigation (structure, difficulty, similar annotations retrieval), a degree of certainty is calculated in order to indicate to the user the reliability of the resulting annotation. Users can then choose to only display annotations which exceed 90% of certainty, or which have been approved by a teacher.

Indeed, filters play an important role on @-MUSE to insure an appropriate personalization of the annotation platform according to each registered user. Figure 8 illustrates this filter system. Thus, users can constitute their own libraries of personalized scores, which are instances of the original score containing all created annotations.

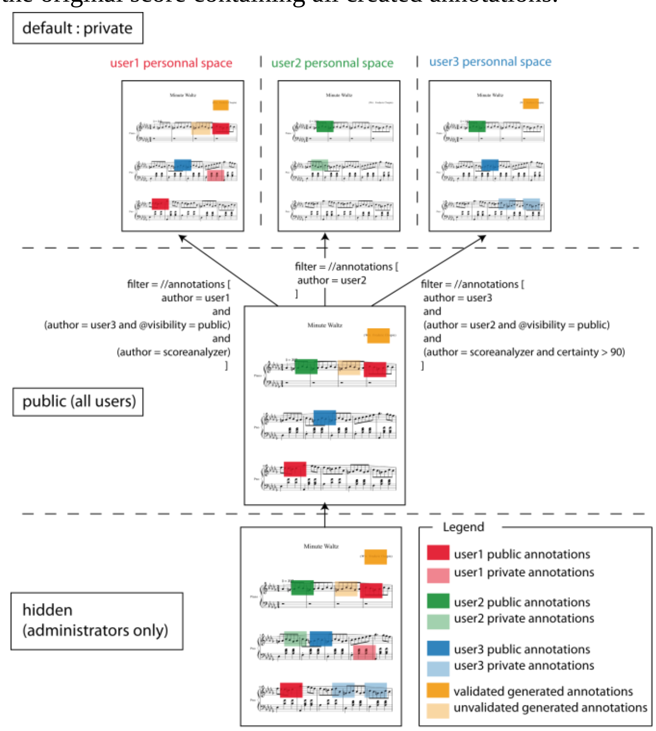

Figure 8. @-MUSE annotations filter system

Inferred playing knowledge can also be used to suggest new pieces to a musician, by analyzing the pieces which are present in his library. Identified criteria for piece learning recommendation are:

- The taste of the learner: if a user profile presents a tendency to play mostly one specific genre (e.g. classical period), two propositions can be made: either play another piece of the same style (not recommended by teachers, as in an academic curriculum, addressing all genres is necessary), or either suggest to discover a modern style one.

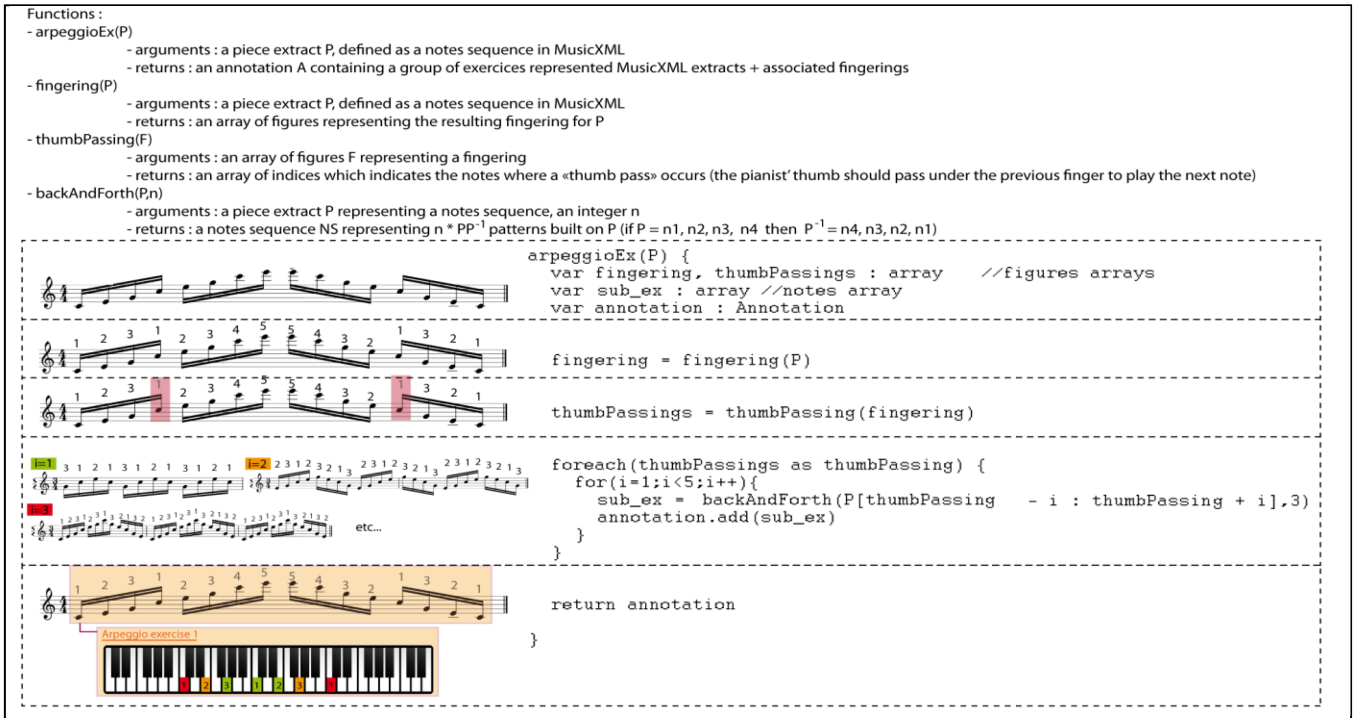

Figure 9. Pseudocode algorithm generating progressive arpeggio exercises

- The difficulties encountered by the learner: studies (e.g. Chopin's Études) constitute recognized works to overcome characteristic difficulties. Once these works have been correctly tagged, they can serve as suggestions to allow students to progress on the identified points. To identify difficulties encountered by a student, we analyze annotations created by this student, or which concern his recordings. To go further, systems which directly analyze performances in real time, such as Apple™ GarageBand® music learning module, need to be studied on advanced pieces.

In the next part, we present an implementation of the algorithms we proposed, in the form of an @-MUSE module called Score Analyzer.

E. Score Analyzer

The criteria presented in the previous sections have been implemented in a Web application called Score Analyzer [40]. This module is integrated to @-MUSE as a Web service in order to automatically evaluate a piece level and identify its difficult parts and advice apprentice musicians on their technical points.

Score Analyzer's engine takes any well-formed MusicXML file as input and parses it to extract knowledge exploitable from a performer point of view. For the time being, it targets pianists, but its main frame and several of its algorithms can directly be applied to other instruments. Following the scheme we detailed previously, the context of the piece is briefly analyzed (title, composer) and a few statistics are displayed (Figure 10). Then, main parts of the piece are identified, and lastly, difficulty estimations are given for each criteria identified in part C. At the time this paper is written, work is still in progress to display the results directly on the score under the form of annotations, in order to enhance the user experience.

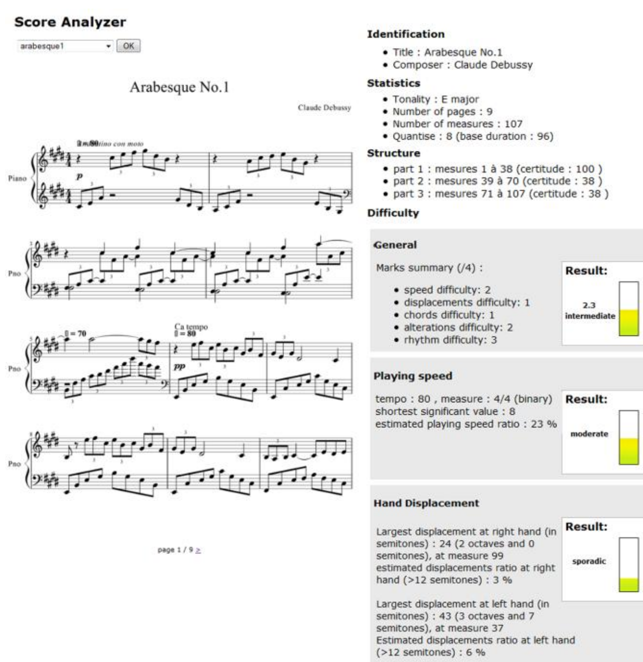

Figure 10. Score Analyzer's interface

To make the results more readable from a user point of view, percentages output from the formula given in Table II were replaced by marks, from 1 (beginner) to 4 (virtuoso), expressing the estimated levels of difficulty. Figure 11 details the defined slots for each criterion. These curves were calibrated on a sample of piano pieces commonly used in music schools and representative of various classical genres and levels. For each criterion, we dispose of at least one piece known to maximize its result. As such, the larger the spectrum of calibration pieces is, the more reliable the results are. Indeed, most of the criteria do not have a linear distribution (Figure 11), which constitutes a pianistic reality. For example, chords presence is considered as high starting from 60% occurrences on a given piece, as pieces constituted of only chords remain seldom cases. Therefore, most pieces concentrate around the center of Figure 11 graph. Easy ones occupy its left down corner, and difficult ones its top right one.

The synchronization between both hands is also taken into account. For instance, if each hand obtains a mark of 2 for the displacements criterion, then the global difficulty mark for this criterion will be 3, as playing with both hands will create an additional difficulty.

The protocol to evaluate the accuracy of our system simply consists in comparing results from Score Analyzer and pianists estimations. To do so, we use two distinct sources. The first one consists in difficulty estimations from the Free-scores online music community [39]. These estimations result from user comments and thus correspond to the experience of a large population of musicians. The second one consists in a precise evaluation of each piece (under the form of a questionnaire) by two experimented piano teachers from Reunion Island Conservatory of Music. This second source favors a qualitative approach. On Figure

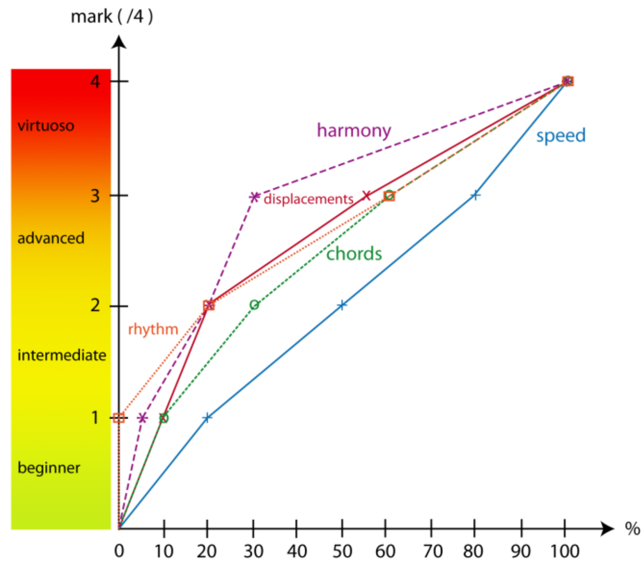

Figure 11. Evolution of marks in function of percentages for each criterion

12, we give the results of this experimentation on a corpus of ten representative piano pieces. This corpus was elaborated to cover a large range of genres and levels, and is regularly studied in music schools.

The main result of this first evaluation is that the estimations provided by Score Analyzer globally correspond to those of the majority of musicians. As such, its usability within an annotation platform containing a large collection of scores such as @-MUSE is relevant. Punctually, some gaps may be noticed between Score Analyzer results and human evaluations, as seen on Figure 12 for Bach's Invention n°1. In this case, this is due to the absence of counterpoint evaluation, which is one of Bach's

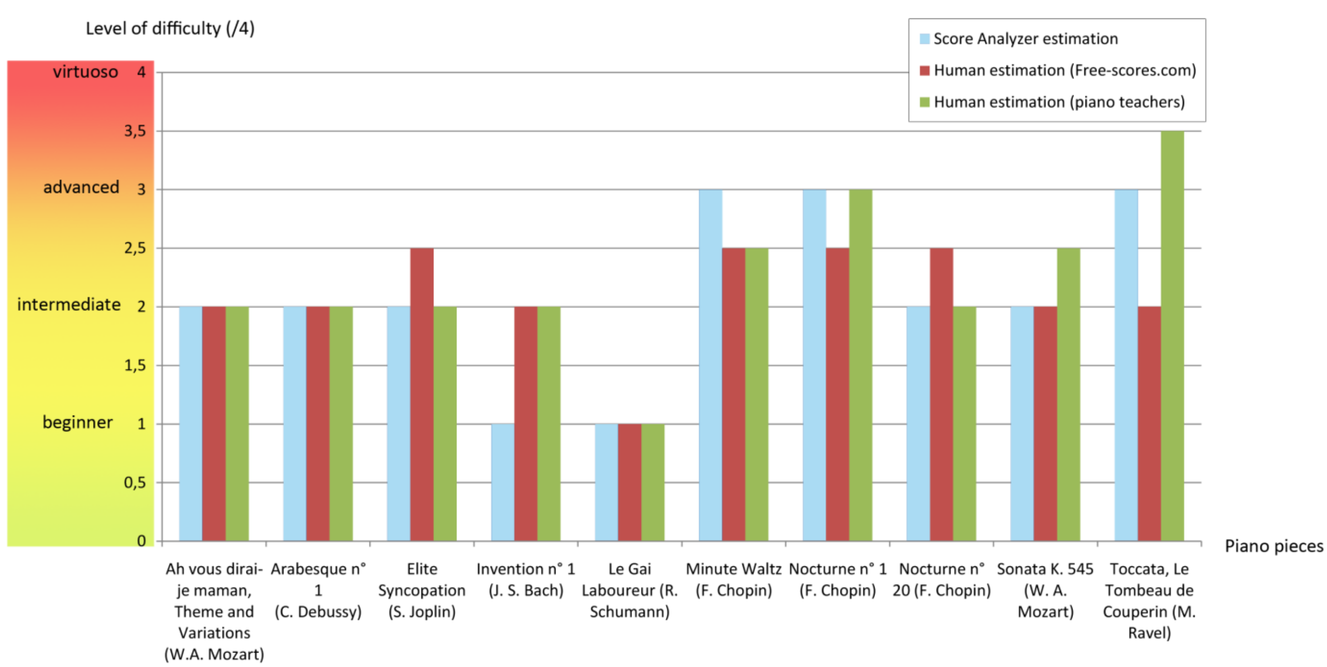

Figure 12. Comparison of difficulty estimations on ten piano pieces

characteristic and which can be particularly tricky to carry out for beginners. To implement this feature, more work need to be done on voices detection (see Part B). We also notice that this protocol induces a few biases. The first one is the lack of estimations for some pieces, thus reducing the objectiveness of the difficulty assessment. Typically, the Toccata from Ravel is clearly an advanced/virtuoso piece, but it was inappropriately marked by the only user who commented it. Hence the gap we can note on Figure 12. This type of cases points up Score Analyzer's interest on new untreated pieces. The second one is the users level. The population of Free-scores community is very heterogeneous, and as such, some users comments are only valid for their level, which is not always appropriate to approach the chosen piece (i.e. a beginner comments an intermediate piece as advanced for his level, and vice versa).

Of course, these biases may correspond to real experiences from users, as each musician approaches a piece differently, with his own skill, culture, feelings and motivation. In this frame, the final purpose of Score Analyzer is to provide an objective advice by informing a user if he chose a piece too difficult for him (a common case in musical education, but also a motive to progress), suggesting appropriate pieces to progress, and guiding him through the sight-reading of new pieces, by indicating difficult parts. On scores databases, Score Analyzer's results could be pointed up when the difficulty level of a piece has not been entered or do not present enough estimations to be relevant.

Thus, to ensure a more reliable human evaluation, we also questioned piano teachers (Figure 12). Most of their assessments correspond to Free-scores and Score Analyzer ones. However, we notice that Score Analyzer results correspond more to teachers' estimations rather than to Free-scores community ones, thus confirming the relevance of our criteria. As such, we dispose of a quantitative validation (lot of answers, less reliability), as well as a qualitative one (few answers, high reliability) on the evaluation of the global difficulty of a piano piece by Score Analyzer.

To go further into details, we also evaluate the quality of the calculation for each criterion. Indeed, our purpose is not only to indicate the level of difficulty of the piece, but also to find in what it is difficult (or not). To do so, we once again compare Score Analyzer results to pianists' assessments. Figure 13 presents an example of such an evaluation on three pieces of different levels (easy, intermediate and advanced).

Despite a few special cases, the estimations globally match (gap $\leq$ 0.5). This experimentation underlines the slight underrating of the Fingering and Rhythm criteria by Score Analyzer. Indeed, teachers evaluate these parameters with more factors than Score Analyzer does for the time being. For instance, rhythms difficulties do not include only the occurrence of many awkward rhythmic patterns in the piece, but also the required stability (for example, the multiple notes repetitions in the Toccata from Ravel) and strictness. Some of these parameters cannot be computed for

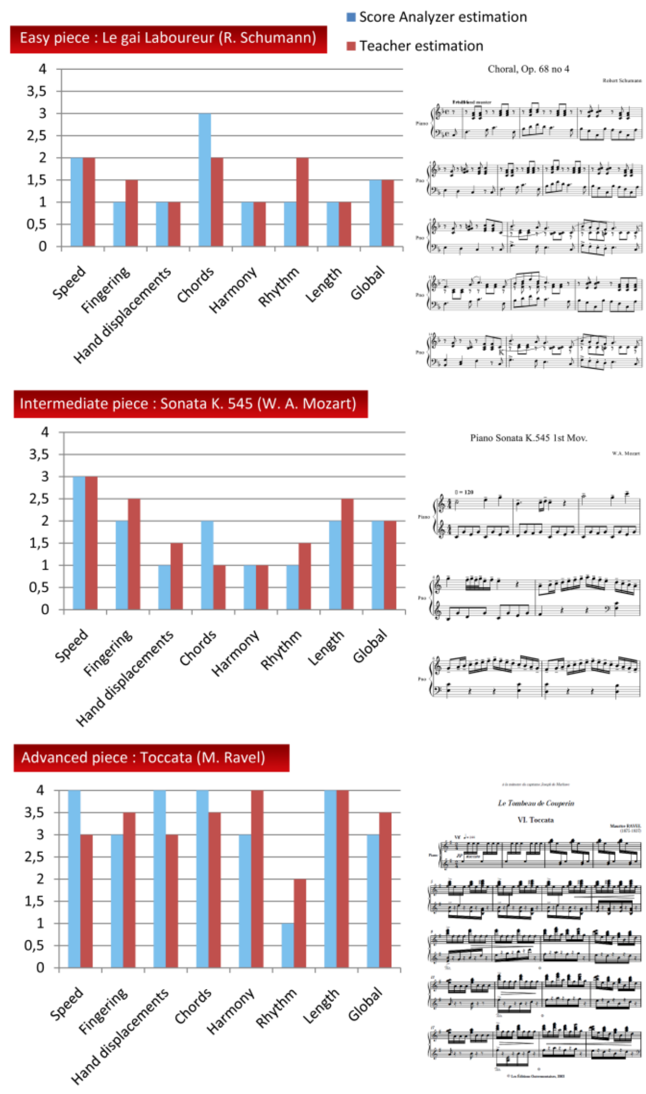

Figure 13. Comparison of difficulty estimations per criteria on three piano pieces

the moment, either because of their nature (expression depiction difficulty), or their complexity (specific patterns dependence). Thus, these first results also give us leads to enhance our calculations.

Working with musicians enabled us to confirm Score Analyzer's first result, but also to raise its main limit concerning high-level works: musicality consideration. Indeed, any Music Information Retrieval (MIR) system is limited as it can only consider processable data (audio/video signal, notes, tempo, text). As for expression and feelings, this remains an issue only noticeable, in multiple ways, to humans. Still, some leads about how a piece should "sound" could be suggested to beginners by analyzing styles, composers, direction texts and nuances, as well as previous annotations on similar pieces. This would constitute an interesting and challenging perspective for our platform.

V. Conclusion and perspectives

In this paper, we have proposed a methodology (Sign Management), a model (Iterative Sign Base System) and some inference methods (score-mining) to build an instrumental e-learning platform called @-MUSE. This platform allows teachers and learners to create music lessons dynamically with the assistance of a semi-automatic pieces annotator. These lessons can evolve according to the users' needs by submitting contextual exercises to them, in the form of multimedia annotations. These exercises are generated from the original score based on the identification of remarkable parts and their playability. Users can then give their point of view on the generated annotations but also add new ones, thanks to a dedicated symbols library as well as a multimedia capture module. The more knowledge is created on the platform, the more detailed the lessons will be, thanks to the emerging network effect resulting from the semantic linking of the various resources.

To generate relevant annotations, we have particularly insisted on the importance of finding difficulties within a score. To do so, we have presented Score Analyzer, a module of @-MUSE enabling automatic evaluation of piano pieces difficulty. Score Analyzer's first results have been presented and validated by confronting them to pianists' assessments.

Different perspectives are considered for this work. Concerning Score Analyzer, the presented experimentations suggests several leads to enhance the difficulty estimation, the main one being a further analysis of the genre and composer of the piece to better study the adapted playing style. Once again, this requires a close collaboration with professional music teachers and musicologists. Of course, a larger MusicXML pieces base would also allow to improve our criteria. We also intend to study in detail their applicability to other instruments and types of performances (chamber music, orchestration, etc). But what really constitutes the next challenge in this project is to distinguish what type of expressive knowledge can be automatically explicitated on a score. Indeed, extracting purely expressive features (emotion, intensity, rubato) from a score remains a tough task, as it rarely includes the basic information to do so. Moreover, imposing "rules" in musicality is a delicate task, as it can lead to conformism. The method we recommend is thus to analyze high level pre-annotated scores and research implicit rules based on the genre of the considered work (for instance, in classical music, it is conventional to soften the end of a phrase). Elements of fuzzy logic would then allow us to balance the relevance of an "expressive" annotation according to the context of the piece.

As for @-MUSE development, our ongoing work is to deliver an interface adapted to tablet devices (Figure 14), which would allow to use our platform directly in front of the instrument, guaranteeing an experience close to a traditional music lesson. Once these modules are merged, the @-MUSE project will give birth to a real e-community dedicated to music practice, and not only to music consumption. As such, the collaborative aspects of such a platform need to be studied to approach music learning under an entertaining angle, for instance by proposing specific group performances (Global Sessions [34]) and game features. Finally, as implied by our platform's name, learning music should first and foremost be a pleasure.

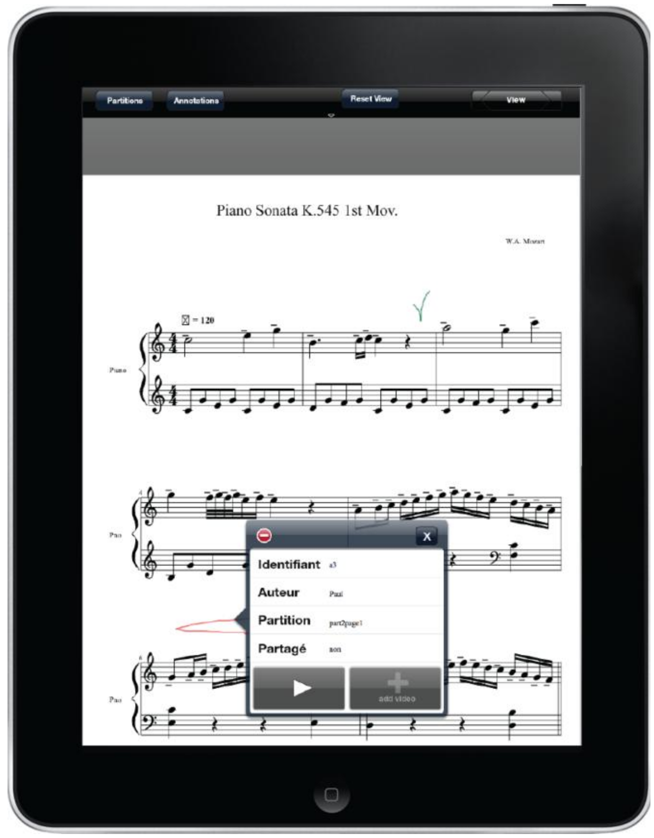

Figure 14. @-MUSE tactile interface

Acknowledgment

The authors thank Marie-Claude Equoy and Maïté Cazaubon (piano teachers at Reunion Island Conservatory of Music) for their expertise on piano education issues, as well as Paul Sébastien (trainee at the University of Reunion Island) for the development and test of @-MUSE tactile interface.

References

[1] V. Sébastien, D. Sébastien, N. Conruyt, "Dynamic Music Lessons on a Collaborative Score Annotation Platform", The Sixth International Conference on Internet and Web Applications and Services , ICIW 2011, St. Maarten, Netherlands Antilles, 2011, pp. 178-183.

[2] L. Abrouk, H. Audeon, N. Cullot, C. Davy-Rigaux, Z. Faget, D. Gross-Amblard, H. Lee, P. Rigaux, A. Tacaille, E. Gavignet, and V. Thion-Goasdoue. The Neuma Project: Towards On-line Music Score Libraries. In Intl. Workshop on Exploring Musical Information Spaces, WEMIS'09, 2009.

[3] G. Castan, M. Good, and P. Roland, "Extensible Markup Language (XML) for Music Applications: An Introduction", The Virtual Score: Representation, Retrieval, Restoration, MIT Press, Cambridge, MA, pp. 95-102, 2001.

[4] C.-C. Lin, "An Intelligent Virtual Piano Tutor", National Chung Cheng University 2006.

[5] N. Conruyt and D. Grosser, "Knowledge management in environmental sciences with IKBS: application to Systematics of Corals of the Mascarene Archipelago", Selected Contributions in Data Analysis and Classification, Series: Studies in Classification, Data Analysis, and Knowledge Organization, Springer, ISBN: 978-3-540-73558-8, 2007, pp. 333-344.

[6] N. Conruyt, O. Sébastien, V. Sébastien, D. Sébastien, D. Grosser, S. Caldéroni, D. Hoarau, and P. Sida, "From Knowledge to Sign Management on a Creativity Platform, Application to Instrumental E-learning", 4th IEEE International Conference on Digital Ecosystems and Technologies , DEST 2010, IEEE Press, 2010, pp. 367-374.

[7] N. Conruyt, O. Sébastien, and V. Sébastien, "Living Lab in practice: the case of Reunion Creativity Platform for Instrumental e-Learning", 13th International Conference on Interactive Computer Aided Learning, ICL 2010, September 15-17, Hasselt, Belgium, 2010.

[8] C. L. Hanon, "The Virtuoso Pianist in Sixty Exercises", 1873.

[9] A. Al Kasimi, E. Nichols, and C. Raphael, "A simple algorithm for automatic generation of polyphonic piano fingerings", 8th International Conference on Music Information Retrieval, September 23rd-27th, Vienna, Austria, 2007.

[10] K.J. Kim and C.J. Bonk, "The Future of Online Teaching and Learning in Higher Education", Educause Quarterly, vol. 29, 2006, pp. 22-30.

[11] O. Lartillot, "Une analyse musicale automatique suivant une heuristique perceptive", 3ème Conférence Internationale Francophone sur l'Extraction et la Gestion des Connaissances, EGC 03, Lyon, France, 2003.

[12] O. Lassila and R. R. Swick, "Resource Description Framework (RDF) Model and Syntax", W3C specification, 1998.

[13] R. Parncutt, J. A. Sloboda, M. Raekallio, E. F. Clarke, and P. Desain. "An Ergonomic Model of Keyboard Fingering for Melodic Fragments", Music Perception: An Interdisciplinary Journal Vol. 14, No. 4, 1997, pp. 341-382.

[14] Y. Raimond, S. Abdallah, M. Sandler, and F. Giasson, "The Music Ontology", Proceedings of the International Conference on Music Information Retrieval, ISMIR, 2007.

[15] Y. Raimond, "A distributed Music Information System", Ph.D. Thesis, Department of Electronic Engineering, Queen Mary, University of London, November 2008.

[16] P. Rigaux, "Neuma Ontology Specification", Project Neuma Report, Lamsade-CNRS, ANR-08, 2008.

[17] O. Sébastien, N. Conruyt, and D. Grosser, "Defining e-services using a co-design platform: Example in the domain of instrumental e-learning", Journal of Interactive Technology and Smart Education, Vol. 5, issue 3, ISSN 1741-5659, Emerald Group Publishing Limited, 2008, pp. 144-156.

[18] V. Sébastien, D. Sébastien, and N. Conruyt, "A collaborative platform model for digital scores annotation", 3rd Annual Forum on e-Learning Excellence in the Middle East, Dubaï, 2010.

[19] V. Sébastien, D. Sébastien, and N. Conruyt, "An Ontology for Musical Performances Analysis. Application to a Collaborative Platform dedicated to Instrumental Practice", The Fifth International Conference on Internet and Web Applications and Services, ICIW 2010, Barcelona, 2010, pp. 538-543.

[20] J. A. Sloboda, E. F. Clarkeb, R. Parncutt, and M. Raekallio, "Determinants of Finger Choice in Piano Sight-Reading", Journal of Experimental Psychology: Human Perception and Performance, Volume 24, Issue 1, 1998, pp. 185-203.

[21] D.R. Stammen and B. Pennycook, "Real-time Recognition of Melodic Fragments using the Dynamic Timewarp Algorithm". ICMC Proceedings, 1993, pp. 232-235.

[22] M. Stanley, R. Brooker, and R. Gilbert, "Examiner Perceptions of Using Criteria in Music Performance Assessment". Research Studies in Music Education, June 2002, vol. 18, issue 1, pp. 46-56.

[23] T. Li, M. Ogihara and G. Tzanetakis, "Music Data Mining", CRC Press, July 2011.

[24] M. A. Winget, "Annotations on musical scores by performing musicians: Collaborative models, interactive methods, and music digital library tool development", Journal of the American Society for Information Science and Technology, 2008

[25] http://e-guitare.univ-reunion.fr, visited on the 23/01/2012.

[26] http://www.guitar-pro.com, visited on the 23/01/2012.

[27] http://www.apple.com/ilife/garageband/#basic-lessons, visited on the 23/01/2012.

[28] Flash Interactive Guitar Saloon: http://e-guitare.univ-reunion.fr/figs, visited on the 23/01/2012.

[29] http://musicbrainz.org, visited on the 23/01/2012.

[30] http://icking-music-archive.org, visited on the 23/01/2012.

[31] http://youtube.com, visited on the 23/01/2012.

[32] http://www.ehow.com/, visited on the 23/01/2012.

[33] http://creativecommons.org/, visited on the 23/01/2012.

[34] http://www.youtube.com/watch?v=ZTOmYLTitGg, visited on the 23/01/2012.

[35] http://www.last.fm, visited on the 23/01/2012.

[36] http://www.finalemusic.com, visited on the 23/01/2012.

[37] http://musescore.org, visited on the 23/01/2012.

[38] http://musescore.com/sheetmusic, visited on the 23/01/2012.

[39] http://www.free-scores.com/, visited on the 23/01/2012.

[40] http://e-piano.univ-reunion.fr/tests/ScoreAnalyser/readScore.php, visited on the 23/01/2012.