A PEN-BASED MUSICAL SCORE EDITOR

Sébastien Macé Eric Anquetil
IRISA – INSA de Rennes
Campus Universitaire de Beaulieu
35042, Rennes Cedex, France
{sebastien.mace, eric.anquetil}@irisa.fr

Elodie Garrivier Bruno Bossis
Laboratoire MIAC – Département Musique
Université Rennes 2 – Campus Villejean
35043, Rennes Cedex, France
bruno.bossis@uhb.fr

ABSTRACT

We present in this paper a pen-based musical score editor prototype. It allows the user to edit quite the same way as if it was on a sheet of paper, all of his drawing strokes being progressively recognized and replaced by their corresponding symbol by the system, right during the editing. Most of the classical musical symbols are already available, as well as the basic editing possibilities. We propose a new way to recognize the musical symbols thanks to the knowledge of musical structure, graphically represented by boxes. Finding the structural box in which a stroke is drawn reduces the possible symbols. It helps minimizing ambiguities and increases recognition rates. First tests among musicians are satisfactory and encouraging.

1. INTRODUCTION

Today, pen-based human-computer interfaces are subjected to a strong expansion. Thanks to the use of a pen, these kinds of interfaces allow the user to interact with the machine by drawing on a touch screen. This kind of interaction is very intuitive. Indeed, it corresponds to a very usual way of communication, and makes it possible to have the advantages of the pen and the paper (durability, quality) without the drawbacks (spatial limitation, difficult modification of its content). Figure 1 presents an example of such an interface.

Figure 1. Example of a pen-based interface: the user interacts with the system thanks to a pen.

Musical score editing is a typical application that can integrate this new intuitive pen-based interaction. It allows the user to edit a musical score as simply as if it was on a usual sheet of paper. But this time, he profits from the advantages of such an interface: he can modify easily any part of his document, save time by automatically recopying a continuation of notes at another place in his document, and more easily distribute his work on a large scale [6].

The work presented in this paper is the result of the collaboration between the IMADOC team [3] from the IRISA research laboratory of Rennes [2] and the MIAC, a laboratory of Rennes 2 from the research team Arts: Pratiques et Poétiques [7]. The research topics of the IMADOC (IMAge et DOCument) team are the handwriting recognition and the pen-based interaction. The main research topic of the MIAC (Musique et Image: Analyse et Création) laboratory is a musicological approach of the relations between sound and image. This collaboration aims at combining the IMADOC knowledge in term of pattern recognition and pen-based interaction with the experience of professional musicians from the MIAC, in order to realize a pen-based system as intuitive as possible.

In this paper we present the development of a first version of a pen-based musical score editor, which already incorporates enough functionality to understand and appreciate the interest of such an interface. The user draws on the touch screen what we call strokes, and obtains the document containing the symbols corresponding to his strokes, like it is presented in Figure 2. In our system, the transformation of the user strokes occurs directly while the user draws.

Figure 2. The strokes of a user drawn on a touch screen (above figure), and the corresponding recognized symbols (below figure).

We present in the second section of this paper a state of the art in pen-based musical score editing. In the third section, we give a description of the prototype. In the fourth section, we present the recognition process of handwritten strokes. In the last section, we present the first feedbacks of the tests of our system.
2. STATE OF THE ART OF PEN-BASED MUSICAL SCORE EDITING

There are few pen-based musical score input systems presented in the literature, because realizing such an application presents a lot of difficulties. The main one is the recognition of handwritten strokes. Indeed, in a musical score, we can find various kinds of symbols. This vocabulary is constituted of specific musical symbols, figures, letters... Some of them are quite similar, and the fact that every scripter has its own way to realize a musical score and draw his symbols increases the recognition problem. A way to reduce this problem is to impose constraints to the user to limit the number of possibilities to realize a given symbol.

That is the reason why most of the existing systems propose a different set of gestures than the usual one to draw musical symbols. The main advantage is to simplify the recognition of the gestures, because the system is generally unistroke, i.e. one stroke is enough to realize any musical symbol. In these cases, any stroke can be analysed independently of the previous and the next ones. The main drawback of this approach is that the user has to learn a new way of writing music.

Forsberg et al. present The Music Notepad [1], which allows the user to edit music with a mix of pen gestures and menu selections. They propose a very different way of editing music in comparison of the usual one used on a sheet of paper.

Ng et al. [5] present Presto, a system that makes it possible to realize musical scores faster than in the usual way, with a set of gestures designed to be learnt easily and quickly. Their system is also unistroke.

We have found only one system that makes it possible to write the music in quite a same way as if it was on a sheet of paper. Miyao et al. [4] propose a set of gestures that is almost the same as the usual one. As a consequence, their system is not completely unistroke.

3. FUNCTIONALITIES OF THE DEVELOPED SYSTEM

3.1. Main functionalities of the editor

The system allows the user to edit on the staffs bass or treble clefs, whole, half and quarter notes. Thanks to additional lines, these notes can be drawn above or below the staff. Notes can have a durational dot. All the accidentals are available for the notes and the key signature. Dynamics are also present. Rests and half rests are not available yet, but quarter, eighth, sixteenth, thirty second and sixty fourth rests are. Line bars can be drawn on the staff.

The system is able to deal with documents with as many pages as needed. The user can see outlines of several pages as the same time, which makes comparison between two pages easier, just like two sheets of paper.

The system proposes some of the usual editing possibilities, like undoing the last stroke, zooming in or out. The symbols can be selected by drawing a circle around them, and then moved to another place in the document, or deleted.

It is also possible to save the document in a file, and to load a saved document. In the current version, it is impossible to export a file so as it could be used in another, more traditional, musical score editor.

As every action can be realized with the pen, the other hand could use for example a piano.

Figure 3 presents a screenshot of the pen-based musical score editor.

Figure 3. Screenshot of the musical score editor.

3.2. Music symbols editing

3.2.1. Progressive Recognition

In our pen-based musical score editor, the recognition occurs directly while the user draws, with no need to ask for it: every time a stroke is realized, it is replaced by the corresponding symbol. In the normal mode of the application, only the symbolic representation of the recognized symbol is visible. However, the strokes drawn by the user can be shown if necessary.

3.2.2. Musical writing guided by structural boxes

In order to help the user, small boxes can appear on the interface, indicating the places where particular symbols can be realized. For example, as presented in Figure 4, when the user draws a filled note with its stem, boxes appear all around it: on the left, a box indicates where to draw the accidental; on the right, a box indicates where to draw the durational dot; on the opposite side of the stem, a box indicates where to draw the accent; below, a box indicates where to draw the nuance.

Figure 4. Structural boxes indicating where to draw specific musical symbols.
The realization of a symbol can disable one or more other structural boxes. For example, on the right of a clef without accidental, a flat or a sharp can be drawn; drawing a flat makes it possible to add other ones, but remove the possibility to draw a sharp: the corresponding box disappears. This phenomenon is presented in Figure 5.

![Figure 5. On the right of a clef can be drawn sharps or flats (left figure); once a flat is drawn, it is possible to draw another one but impossible to add a sharp (middle figure); then, more flats can be added (right figure).](image)

In order to lighten the editing window, we have chosen to show only the free boxes. Once the user has got accustomed to the application, he can change from a "novice mode", in which all the free zones are visible, to an "expert mode", in which none of them are, making the editor look just like a traditional sheet of paper.

4. RECOGNITION

4.1. Problem

The most difficult problem of such a pen-based system is the recognition of the user’s strokes, i.e. their transformation into their equivalent symbolic representation. As we have seen in section 2, the recognition of the usual musical gestures is much more difficult as a lot of symbols are quite similar. The definition of a new input method can simplify the recognition process, but the user has to get used to it.

We have chosen a compromise between the existing systems: as long as an elementary symbol can be drawn with a single stroke, we didn’t change the way to realize it (for example notes, stems, clefs, etc.); on the contrary, when an elementary symbol needs at least two strokes to be drawn, we changed it, but chose a gesture as close as possible of the original one (for example, a sharp is usually constituted of four strokes, in our system an horizontal one is enough to draw it).

As a consequence, a given stroke can have different meanings depending on the context. For example, a small circle can be recognized as a whole note, as the character "o" or as the figure "0". We decided to use structural knowledge we have about music to make a decision in those cases.

4.2. Recognition driven by the structure

In order to improve the recognition mechanism, we use the highly structured property of musical scores, which gives us information about the place where a symbol is located relatively to others. Indeed, we know that a symbol very close of a note on its left has a high probability of being an accidental. As it is shown in Figure 6, in our system, the recognition of the graphical gestures is driven by the structure of the document.

![Figure 6. Symbolic representation of the recognition system.](image)

We first present a formalism to emphasize the structure knowledge we use to realize our system. Then, we present how we analyse the structure of the document and use the recognition systems.

4.2.1. Formalization of the structural knowledge

The structural knowledge used in our system is formalized by rules in which we use the following formalism:
- various elements are music symbols;
- "+-" indicates that the symbols on the right can be drawn relatively to the symbol on the left;
- superscript indicates the maximum number of elements of a given nature that can be drawn; "n" means that there can be from 0 to n elements; no indication implicitly means "1";
- "+[position]" are the position operators, which correspond to the relative position of the elements on its right compared to the one on the left of the "+-" operator; no indication implicitly means "[on]".

Here are some of the rules used in our system:

| above_staff | crescendo
| |
| below_staff | decrescendo
| |
| additional_line | note
| |
| clef | note
| |
| note | silence
| bar_line
| |
| clef | [on the right] (flat | sharp)
| |
| note | [on the left] accidental
| [on the right] number
| |
| note | [on the right] durational_dot
| [above|below] accent
| |
| note | [above|below] nuance
| [above|below] stem
| |
| silence | [at the end] (flag | beam)
| |
| silence | [on the right] durational_dot
| |
The obtained boxes are then ordered thanks to their "row" value, in order to get a list of structural boxes, from the most probable to the less probable. We can then apply the classifiers associated to the boxes, from the more probable to the less probable. As soon as the stroke is recognized, the recognition process is over with success, the user stroke is associated to its corresponding recognized musical symbol. On the contrary, no recognition after the analysis of all the boxes leads to a failure of the recognition system; the user stroke is lost and has to be realized again.

5. CONCLUSION

This paper presents a pen-based musical score editing prototype. Even though it still doesn’t incorporate all the musical notations, most of the usual ones are already implemented, making it possible to realize that this kind of interfaces is really adapted to music editing. Various demonstrations have been realized on a tablet PC among musicians, and the feedbacks are really satisfactory. All agree about the innovation of this application and its facility of use.

These positive results encourage us to go further on the development of this prototype. Of course, we will increase the number of musical symbols available in the application to make it as complete as possible. Then, we will think of a saving format for the documents, for them to be for example reusable in other applications. One possibility considered is to export the musical scores in MIDI format, allowing the user to hear the music he has written. We will also think of experimental testing protocols.

Other future works will aim at defining generic formalisms to represent structural knowledge and the interaction with the user. Such tools could be used in the end in other contexts than music.

6. REFERENCES