Hindawi Publishing Corporation
Advances in Human-Computer Interaction
Volume 2015, Article ID 850474, 13 pages
http://dx.doi.org/10.1155/2015/850474

Research Article

CaRo 2.0: An Interactive System for Expressive Music Rendering

Sergio Canazza, Giovanni De Poli, and Antonio Rodà

Department of Information Engineering, University of Padova, Via Gradenigo 6/A, 35131 Padova, Italy

Correspondence should be addressed to Sergio Canazza; canazza@dei.unipd.it

Received 6 August 2014; Revised 14 December 2014; Accepted 27 December 2014

Academic Editor: Francesco Bellotti

In several application contexts in multimedia field (educational, extreme gaming), the interaction with the user requests that system is able to render music in expressive way. The expressiveness is the added value of a performance and is part of the reason that music is interesting to listen. Understanding and modeling expressive content communication is important for many engineering applications in information technology (e.g., Music Information Retrieval, as well as several applications in the affective computing field). In this paper, we present an original approach to modify the expressive content of a performance in a gradual way, applying a smooth morphing among performances with different expressive content in order to adapt the audio expressive character to the user's desires. The system won the final stage of Rencon 2011. This performance RENdering CONtest is a research project that organizes contests for computer systems generating expressive musical performances.

1. Introduction

In the last years, several services based on Web 2.0 technologies have been developed, proposing new modalities of social interaction for music creation and fruition [1]. Notwithstanding differences among the systems, all these services tend to divide users in two categories: on the one hand the large group of music listeners, which mainly have the job of evaluating and recommending music; on the other hand, the restricted group of music content creators, which are required to have skills in the field of music composition or music performance. This partition limits the participation and interaction with the music content. Applications like Guitar-Hero try to fill this gap, giving also to non-musically trained users the possibility to experiment with music performance. Unfortunately, up to now game designers did not consider a very important aspect of music, that is, the player's expressiveness: a performance is rated "perfect" only if it is played exactly like the musical score, without any interpretation, any humanity.

Our studies on music expressiveness [2, 3] led us to develop a tool for the auralisation of multimedia objects for Web-based applications [4]. Our audio authoring tool allows a user to manage audio expressive content, applying a smooth morphing among different expressive intentions in music performances and adapting the audio-expressive character to their taste. The audio-authoring tool lets you associate different expressive characters with various multimedia objects.

In this paper we present a system, namely, CaRo 2.0 (CAnazza-ROdà, from the name of the two main authors: besides, *caro* in Italian means "dear"), that allows an active experience of music listening. In the CaRo 2.0 system the Valence-Arousal emotional space and Energy-Kinetics sensory space are generalised into the idea of an abstract control space, where expressive intentions are associated with positions and objects. Each user can invent or select their own expressive concepts and then design the mapping of these concepts to positions and movements on this space. The emotion and sensory control metaphors are implemented as particular and significant cases. When the user accesses a musical content, (s)he can act on the control space by selecting a point or drawing a trajectory, thus changing the character of the music. The system allows the user to engage an active and personalised experience by the interactive control of the expressive nuances of the music (s)he is listening to.

2. Modeling Expressive Music Performance

2.1. Expressive Deviations. The contribution of the performer to expression communication has two aspects: to clarify

the composer's message enlightening the musical structure and to add his personal interpretation of the piece. A mechanical performance of a score is perceived as lacking of musical meaning and is considered dull and inexpressive as a text read without any prosodic inflection. Indeed, human performers never respect tempo, timing, and loudness notations in a mechanical way when they play a score: some deviations are always introduced, even if the performer explicitly wants to play mechanically [5].

Most studies on musical expression aim at understanding the systematic presence of *deviations* from the musical notation as a communication means between musician and listener. Deviations introduced by technical constraints (such as fingering) or by imperfect performer skill are not normally considered part of expression communication and are thus often filtered out as noise.

At a physical information level, the main parameters considered in the models of the musical expression, called *expressive parameters*, are related to timing of musical events and tempo (fast or slow), dynamics (loudness variation), and articulation (the way successive notes are connected). These parameters are particularly relevant for keyboard instruments. Moreover, they are the basic parameters of the MIDI protocol (Musical Instrument Digital Interface: see http://midi.org/aboutmidi/tut_history.php) and thus are easily measurable on digital music instruments and can be used as control signals for rendering a music performance. In some instruments and in the singing voice other acoustic parameters are taken into account such as vibrato and microintonation or pedalling at the piano. In contemporary music, timbre is often an essential expressive parameter; sometimes also virtual space location or movement of the sound source is used as expression feature.

The analysis of these systematic deviations has led to the formulation of several models that try to describe their structure, with the aim to explain where, how, and why a performer modifies, sometimes unconsciously, what is indicated by the notation in the score. It should be noticed that, although deviations are only the external surface of something deeper and often not directly accessible, they are quite easily measurable and thus widely used to develop computational models for performance understanding and generation.

2.2. Expressive Intentions. In general, musical expression refers both to the means used by the performer to convey the composer's message and to his/her own contribution to enrich the musical message. Expressiveness related to the musical structure may depend on the dramatic narrative developed by the performer and on the stylistic expectation based on cultural norm (e.g., jazz versus classic music) and the actual performance situation (e.g., audience engagement). Recently, more interest has been given to the expressive component due to the personal interpretation of the performer [6, 7].

Many studies (see, e.g., [8–13]) demonstrated that it is possible to communicate expressive content at an abstract level, changing the interpretation of a musical piece. In fact, the musician organises acoustical or perceptual changes in sound communicating different feelings, images, or emotions to the listener. The same piece of music can be performed differently by the same musician (trying to convey a specific interpretation of the score or the situation) by adding mutable *expressive intentions*.

Notice that sometimes expressive intentions the performer tries to convey can be in contrast with the character of the musical piece. A slightly broader interpretation of expression as *kansei* (Japanese term indicating sensibility, feeling, and sensitivity) [14, 15] or affective communication [16] is proposed in some Japanese or American studies. We prefer the broader term *expressive intention* that includes emotion, affect, and other sensorial and descriptive words or actions. Furthermore, this term evidences the explicit intent of the performer in communicating expression. Most music performances would involve some intention from the performer's side regarding what the music should express to the listeners. Consequently, interpretation involves assigning some kind of meaning to the music.

When we talk of deviation, it is important to define which is the reference used for computing deviation. Very often the score is taken as *reference*, both for theoretical (the score represents the music structure) and for practical (it is easily available) reasons. However the choice depends on the problem one wants to focus on. When we studied how a performer plays a piece according to different expressive intentions, we found that a clearer interpretation and best results in simulation are obtained by using a *neutral* performance as [17]. By "neutral" we intend a human performance without any specific expressive intention. In other studies the mean performance (i.e., the mathematical mean across different performances by the same or many performers) was the used reference to investigate stylistic choices and preferences (e.g., [18, 19]).

2.3. Categorical versus Dimensional. Most people have an informal understanding of musical expression. While its importance is generally acknowledged, the basic constituents are less clear. Often the simple range expressive-inexpressive is used. Regarding the affective component of music expression, the two theoretical traditions that have most strongly determined past research in this area are *categorical* (also called discrete) and *dimensional* emotion theories.

The assumption of the *categorical* approach is that people experience emotions as categories that are distinct from each other. Theorists in this tradition propose the existence of a small number of basic or fundamental emotions that are characterised by very specific response patterns. From these basic emotions, all other emotional states can be derived. The focus is on characteristics that distinguish emotion from one another. There is a reasonable agreement on four basic emotions: happiness, sadness, anger, and fear. The next common two are disgust and surprise. In this case expression information is represented by labels indicating the expression category and eventually a number or adjective indicating the degree of that expression.

The focus of the *dimensional* approach is on the identification of emotions based on their placement on a continuous

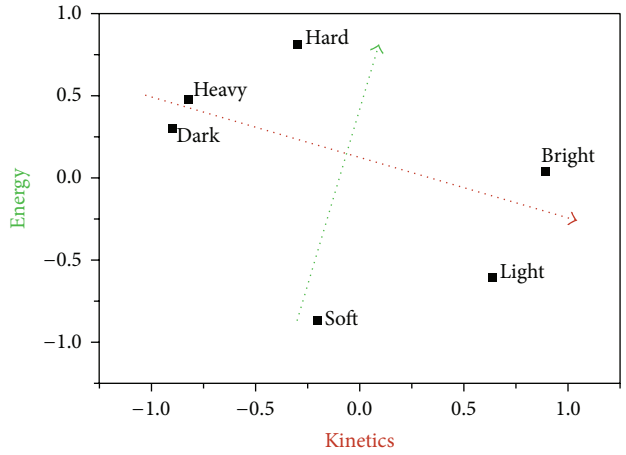

Figure 1: Kinetics-Energy space, as mid-level representation of expressive intentions.

space with a small number of dimensions. This space is derived from similarity judgements, analysed using factor analysis or multidimensional scaling. The dimensional approach provides a way of describing expression which is more tractable than using words, but which can be translated into and out of verbal descriptions. Translation is possible because emotion-related words can be understood, at least to a first approximation, as referring to positions in the dimensional space [20]. Moreover this representation is useful for capturing the continuous change in expression during a piece of music [21].

The most used representation in music research is the two-dimensional Valence-Arousal (V-A) space, even if other dimensions were explored in several studies (e.g., see [22]). The V-A space organises emotions in term of affect appraisal (pleasant-unpleasant) and physiological reaction (high-low arousal). For example, happiness is described as a feeling of excitement (high arousal) combined with positive affect (high valence). A problem with this approach is that specifying the quality of a feeling only in terms of valence and arousal does not allow a very high degree of differentiation, especially in research on music, where one may expect a somewhat reduced range of both the unpleasantness and the arousal of the states produced by music [23].

Unlike the studies on music and emotions, the authors focused on expressive intentions described by sensorial adjectives [2], which are frequently used in music performance, that is, light, heavy, soft, hard, bright, and dark and others. Each of these descriptors has an opposite (soft versus hard) and provokes contrasting performances by the musician. Some performances, played according to the different expressive intentions, were evaluated in listening experiments. Factor analysis, using the performances as variables, found a two-dimensional space (Figure 1). The first factor is characterised by bright/light versus heavy performances, the second one by soft versus hard performances. Acoustical analysis of the performances (Table 1) showed that first factor is closely correlated with Tempo and can be interpreted as the *Kinetics* (or Velocity) factor, while the second one is related

Table 1: Correlation between coordinate axes and acoustic parameters.

	Tempo	Legato	Intensity
Dim. 1 (*kinetics*)	**0.65**	−0.28	−0.25
Dim. 2 (*energy*)	0.33	**−0.72**	**0.73**

to legato/staccato and Intensity and can be interpreted as the *Energy* factor. By legato/staccato, we refer to how much the notes are detached and distinctly separated, as the ratio between the duration of a given note (i.e., the measure of time between note-onset and note-offset) and the interonset interval (i.e., the measure of time between two consecutive note onsets) which occurs between its subsequent notes.

We can use this interpretation of Kinetics-Energy space as an indication of how listeners organised the performances in their own minds, when focusing on sensory aspects. The robustness of this representation was confirmed by synthesising different and varying expressive intentions in a musical performance. We can notice that this representation is at an abstraction level which is between the semantic one (such emotion) and physical one (such as timing deviations) and can thus be more effective in representing the listener's evaluation criteria [24].

3. Computer Systems for Expressive Music Performance

While the models described in the previous section were mainly developed for analysis and understanding purpose, they are also often used for synthesis purposes. Starting from models of musical expression, several software systems for rendering expressive musical performances were developed (see [25] for a review).

The systems for automatic expressive performances can be grouped into three general categories: (i) autonomous, (ii) feedforward, and (iii) feedback systems. Examples for each category are presented.

Given a score, the purpose of all the systems is to calculate the so-called *expressive deviations* to be applied to each note, in order to change the duration and the intensity of the musical events notated in the traditional music score. A performance without *expressive deviations* is indicated by the term "nominal performance."

(1) Autonomous Systems. Autonomous System (AS) is a collection of rules obtained automatically, in which the musical score is processed without user participation and then played in automatic way.

As an example, *YQX* system [26] uses a rule set obtained by means of machine learning algorithms developed in the research field of Artificial Intelligence. Widmer [27] developed an algorithm for automatic extraction of rules specifically designed to analyse musical performances. By applying the algorithm to a collection of performances of Mozart's piano sonatas—played using a grand piano Bösendorfer 290 SE—a rule set has been extracted that suggests some deviations in the expressive rhythmic and in the melodic

structures. For example, the *staccato* rule requires that, if two successive notes have the same pitch but the second has a longer temporal duration, then the first note should be performed *staccato*, and the *delay-next* rule states that if two notes of the same duration are followed by a longer one, then the last note should be played with a slight delay. As mentioned before, these rules were obtained from the training sets provided: in this sense, the rules can change if the system is trained on a different piano repertoire or with music performed by different pianists.

(2) Feedforward Systems. The systems characterised by a feedforward behaviour allow the user to annotate the score and to set the specific parameters of the system depending on the music score to be played, but they do not allow a real-time control during the performance.

A historical example is *Director Musices*, a rule based system for automatic performance developed by the Kungliga Tekniska Högskolan (KTH) in Stockholm, born in the 1980s [28] with the aim of improving the synthesis quality of computer systems for the singing voice. Starting from the work of musical structures analysis by Ipolyi [29] and in collaboration with the violinist Frydén Lars, Johan Sundberg formalised the relation set between the musical structures and the expressive deviations. The method used to formalise most rules is called "analysis by synthesis" [30]. As a first step, a model based on experience and intuition is developed, and it is subsequently applied to different musical examples. Performances generated by this model are evaluated by listening to identify additional changes to the model. The process is reiterated several times, until a stable definition of the model is obtained. At this point, perceptual tests on a larger population are carried out, in order to assess the rule system. The *Director Musices* input is a score in MIDI format, applying all the rules (or a subset of them). Then it computes the expressive deviations for every single note, saving the results to a new MIDI file. The user can choose which rules to apply and how to weight them through numerical coefficients. The deviations calculated by each rule are added together, giving rise to complex profiles, as shown in Figure 2.

The *Shunji* [31] is an algorithm for automatic performance that uses the case-based reasoning paradigm: the well-known artificial intelligence technique that learns from examples, alternative to those used by the software *YQX*. Unlike the latter, however, *Shunji* is a supervised system, not providing a solution to the problem proposed in an autonomous way, but interacting with the user proposing possible solutions and receiving controls to improve the result. *Shunji* is based on a hierarchical segmentation algorithm [32], inspired by the *Generative Theory of Tonal Music* by Lerdahl and Jackendoff [33]. The performances provided as an example to the algorithm are segmented using the hierarchical model and the expressive deviations of each segment are stored in a database. When the system input is a score that had not been processed before, the software segments the score and then, for each segment, searches the database for the most similar segment, in order to extract the performance characteristics. Then it applies the expressive deviations thus obtained to

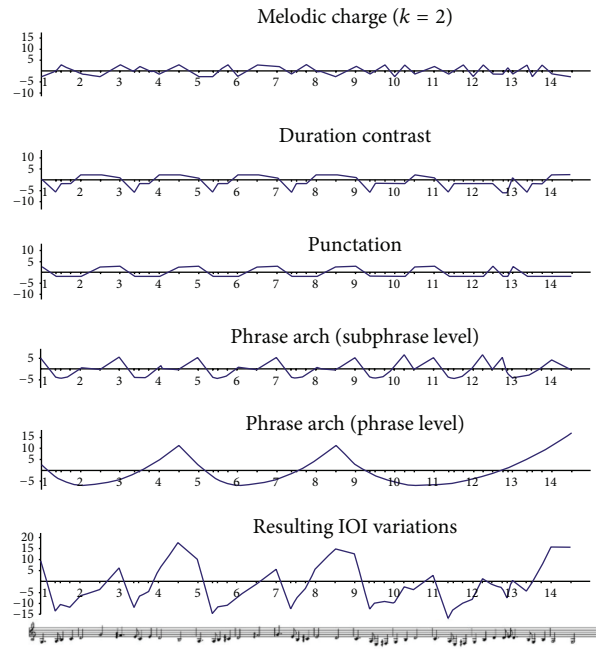

FIGURE 2: Example of the rules system by KTH application (adapted from [30]). In particular, the figure shows the resulting interonset interval (IOI, i.e., the time from one tone's onset to that of the next) deviations by applying of four rules (phrase arch, duration contrast, melodic charge, and punctuation) to the Swedish nursery tune *Ekorrn satt i granen*.

the new score. The user can interact with the software by controlling the segmentation of the score, by providing new examples, by changing the parameters used to define the similarity between two musical phrases, and by choosing the examples to extract the performance characteristics among those that the system has considered to be more similar. This approach also works using a reduced number of examples, if they are well chosen.

(3) Feedback Systems. The systems described above work independently or require user intervention to define some performance details before replay. On the contrary, in the feedback systems, the user can control the performance in real time during replay. As the orchestral conductor controls the musicians' performance, the user can modify the *expressive deviations* in an interactive way.

In the VirtualPhilharmony [34], the user is given a score. Before starting the performance and with the help of some tools, (s)he has to choose an expressive model, that is, a set of expressive deviations in relation to the articulation and agogic aspects of each single note. During the performance, a sensor measures the user's arm movements who mimics the gestures of an orchestral conductor with a wand in his/her hand. The system then modifies the temporal and dynamic characteristics of the performance in a consistent way, following the user's intentions, taking into account the particular aspects of the interaction conductor-musician, including the latency time between the conductor's gesture and the musician execution and the prediction of the next

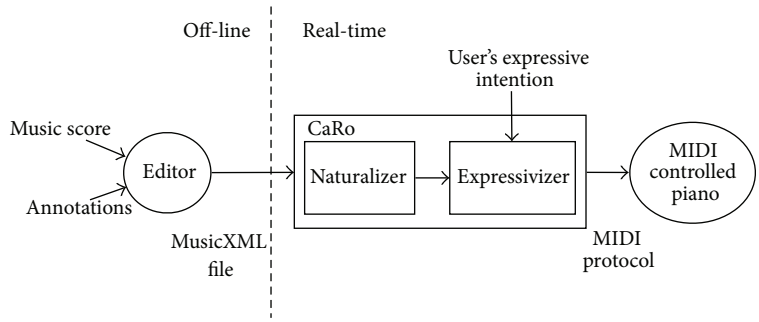

Figure 3: System architecture: CaRo 2.0 receives as input an annotated score in MusicXML format and generates in real-time messages to play a MIDI controlled piano.

beat, which allows the musician to maintain the correct tempo between a conductor's gesture and the next gesture.

The differences between the systems described above include both the algorithms for computing the expressive deviations and the aspects related to the user's interaction. In the autonomous systems, the user can interact with the system only in the selection of the training set and, as a consequence, the performance style that the system will learn. The feedforward systems allow a deeper interaction with the model parameters: the user can set the parameters, listen to the results, and then fine-tune the parameters again until the results are satisfying. The feedback systems allow a real-time control of the parameters: the user can change the parameters of the performance while (s)he is listening to it, in a similar way as a human musician does. Since the models for music performance usually have numerous parameters, a crucial aspect of the feedback systems is how to allow the user to simultaneously control all these parameters in real-time. VirtualPhilharmony allows the user to control in real-time only two parameters (intensity and tempo); the other ones are defined offline by mean of a so-called performance template.

Also the system developed by the authors, and described in the next sections, is a feedback system. As explained later, the real-time control of the parameters is allowed by means of a control space based on a semantic description of different expressive intentions. Starting from the trajectories drawn by the user on this control space, the system maps concepts such as emotions or sensations in the low level parameters of the model.

4. The CaRo 2.0 System

4.1. Architecture. CaRo 2.0 simulates the tasks of a pianist who reads a musical score, decodes its symbols, plans the expressive choices, and finally executes the actions in order to actually play the instrument. Usually, pianists analyse the score very carefully before the performance and they add annotations and cues to the musical sheet in order to emphasise rhythmic-melodic structures, section subdivisions, and other interpretative indications. As shown in Figure 3, CaRo 2.0 receives a musical score as input, compliant with the MusicXML standard [35]. Any music editor able to export in MusicXML format (e.g., Finale (http://www.finalemusic.com/) or MuseScore (http://musescore.org/)) can be used to write the score and add annotations. CaRo 2.0 is able to decode and process a set of expressive cues, as specified in the next section. The slurs can be hierarchically structured to individuate sections, motifs, and short musical cells. Moreover, textual labels can be added to specify the interpretative choices.

The specificity of CaRo 2.0, in comparison to the systems presented in Section 3, is that it is based on a model that explicitly distinguishes the expressive deviations related to the musical structure from those related to other musical aspects, such as the performer's expressive intentions. The computation is done in two steps. The first module, called *Naturalizer*, computes the deviations starting from the expressive cues annotated in the score. As the cues usually indicate particular elements of the score, such as a musical climax or the end of a section, the deviations have shapes that reflect the structure of the score and the succession of motif and musical cells (see, e.g., Figure 12). The deviations computed by the Normalizer correspond to a performance that is musically correct, but without a particular expressive emphasis. The second module, called *Expressivizer*, modifies the deviations computed in the previous step, in order to characterize the performance according to the user's expressive intentions.

4.2. Rendering. The graphical interface of CaRo 2.0 (see Figure 11) consists in a main window hosting the Kinetics-Energy space (see Section 2.3), used for the interactive control of the expressive intentions, and in a main menu through which the user can load an annotated musical score in MusicXML format and launch the execution of the score. When the user loads the MusicXML score, the file is parsed and three data structures are generated: note events, expressive cues, and hierarchical structure. At this level, the musical score is as a collection of separate events EV[ID], with the event index ID $= 1, \ldots, N$. For instance, a score can be described as a collection of notes and rests. Each event is characterised by a time reference and a list of attributes (such as pitch and duration).

The first structure is a list of note events, each one described by the following fields: ID number, onset (ticks), duration (ticks), bar position (ticks), bar length (beats),

Table 2: Correspondence among graphical expressive cues, XML representation, and rendering parameters.

Graphical symbol	MusicXML code	Event value
•	<articulations ><staccato default-x="3" default-y="13" placement="above"/></articulations >	DR[n]
>	<articulations><accent default-x="–1" default-y="13" placement="above"/></articulations>	KV[n]
–	<articulations><tenuto default-x="1" default-y="14" placement="above"/></articulations>	DR[n]
𝄾	<articulations><breath-mark default-x="16" default-y="18" placement="above"/></articulations>	O[n]
⌒	<notations ><slur number="1" placement="above" type="start"/></notations>	O[n], DR[n], KV[n]
Ped.	<direction-type><pedal default-y="–91" line="no" type="start"/></direction-type>	KV[n]

voice number, grace (boolean), and pitch, where ticks and beats are common unit measures for representing musical duration (for the complete MIDI detailed specification, see http://midi.org/techspecs/midispec.php). The last field is a pointer to a list of pitches, each one represented by a string of three characters, following the schema XYZ, where X is the name (from A to G) of the note, Y is the octave (from 0 to 8), and Z is the alteration (-1 stands for flat ♭ and $+1$ for sharp ♯). A chord (more tones played simultaneously) is represented by a list of pitches with more than one entry. An empty list indicates a music rest.

The second structure is a list of expressive cues, as specified in the score. Each entry is defined by type, ID of the linked event, and voice number. Among many different ways to represent expressive cues in a score, the system is currently able to recognise and render the expressive cues listed in Table 2.

The third data structure describes the hierarchical structure of the piece, that is, its subdivision, top to bottom, in periods, phrases, subphrases, and motifs. Each section is specified by an entry composed by begin (ticks), end (ticks), and hierarchical level number.

For clarity reasons, it is useful to express the duration of the musical event in seconds instead of metric units such as ticks and beats. This conversion is possible taking the tempo marking written in the score, if any, or normalising the total score length to the performance duration. This representation is called *nominal time* t_{nom}, the event onset is called nominal onset time $O_{\text{nom}}[n]$, the event length is called nominal duration $\text{DR}_{\text{nom}}[n]$, and distance between two adjacent events is called nominal inter onset interval:

$$\text{IOI}_{\text{nom}}[n] = O_{\text{nom}}[n+1] - O_{\text{nom}}[n]. \quad (1)$$

When the user selects the command to start execution of the score, a timer is instantiated to have a temporal reference for the processing of the musical events. Let t_p be the time past from the beginning of the performance. Since the system is interactive and the results depend on the user's input at the time t_p, the expressive deviations (and therefore the values of each note event) have to be computed just in time before they are sent to the MIDI interface. The deviations are calculated by the `PlayExpressive()` function, which is periodically called by the timer with a time interval T. This interval is a critical parameter of the system and its value depends on a trade-off between two conditions. First, the time interval must be large enough to allow the processing and playing of the note events; that is, $T < t_C + t_M$ where t_C is the time for calculating the expressive deviations, generally negligible, and t_M is the time to transmit the note events by means of the MIDI protocol, a serial protocol with a transfer rate of 31.250 bps. Since a note event is generally coded with three bytes (and each byte, including the start and stop bits, is composed by 10 bits), the time to transfer a note event is about 1 ms. Therefore, assuming that a piano score can have at most 10 simultaneous notes, T should be greater than 10 ms. Another condition is that T should be small enough to give the user the feeling of real-time control of the expressive intentions; that is, $T < t_E$, where t_E is the time the user needs to recognise that the expressive intention has changed. Estimating t_E is not a trivial task, because many aspects have to be considered, such as the cognitive processes and the cultural and musical context. However, experimental evidence suggests that certain affective nuances can be recognised also in musical excerpts with a length of 250 ms [36]. A preliminary test showed that a period $T = 100$ ms guarantees the on-time processing of the musical events and has been judged sufficient to satisfy the real-time requirement in this context. Figure 4 shows how the `PlayExpressive()` function works.

(1) Naturalizer. The Naturalizer computes the deviations that are applied to the nominal parameters to obtain a natural performance. Each time the `PlayExpressive()` function is called by the timer, the note events list—which contains all the notes to be played—is searched for the events with an onset time $O_{\text{nom}}[n]$ such that $t_p < O_{\text{nom}}[n] \leq t_p + T$. If no event is found the function ends; otherwise the expressive cues list is searched for cues linked to the notes to be played. For each expressive cue that is found, a *modifier* is instantiated. The modifier is a data structure that contains the deviations to be applied to the expressive parameters, due to the expressive cue written in the score. The parameters influenced by the different expressive cues are listed in Table 2. Depending on the type of cue, the modifier can work on a single event only, identified by its ID number, or on a sequence of events, identified by a temporal interval (the modifier's scope) and a voice number. The expressive deviations can be additive or proportional, in function of the type of cue and the expressive parameters. The deviations are

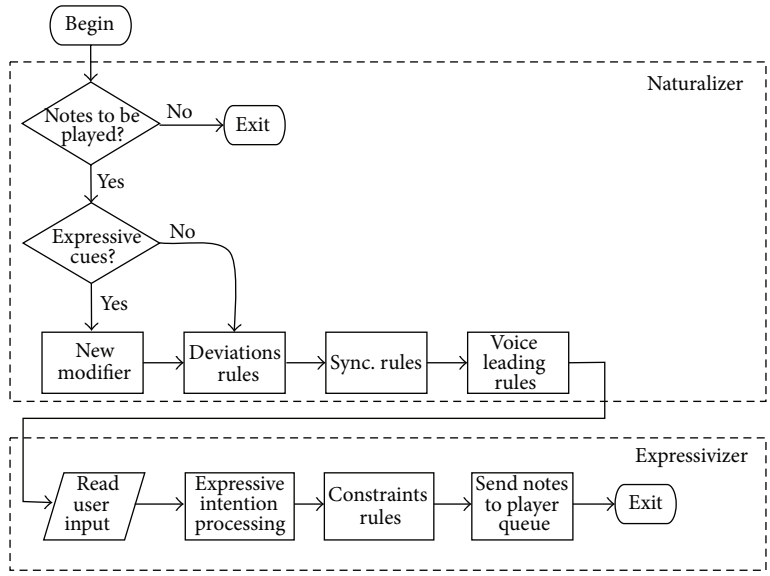

Figure 4: Schema of the `PlayExpressive()` function.

applied by the successive module, named *deviation rules*. This module combines the deviations defined by all the modifiers within a temporal scope between t_p and $t_p + T$, following a set of rules. The output is, for each note, a set of expressive parameters, which define how the note is played. Although CaRo 2.0 can control several different musical instruments (e.g., violin, flute, and clarinet), with a variable number of parameters, this paper is focused on piano performances, and the parameters computed by the rules system are $\mathrm{IOI}_{\mathrm{nat}}$, $\mathrm{DR}_{\mathrm{nat}}$, and $\mathrm{KV}_{\mathrm{nat}}$, where the subscript nat identifies the parameters of the natural performance and KV is the Key-Velocity, a parameter of the MIDI protocol related to the sound intensity.

(2) Expressivizer. The next step of the rendering algorithm is called *Expressivizer* and its aim is to modify the expressive deviations computed by the *Naturalizer*, in order to render the user's different expressive intentions. It is based on the multilayer model shown in Figure 5: the user specifies a point or a trajectory in an abstract control space; the coordinates of the control space are mapped on a set of expressive features; finally, the features are used to compute the deviations that must be applied to the score for conveying a particular expressive intention. As an example, Figure 7 shows a mapping surface between points of the control space and the feature Intensity. Based on the movements of the pointer on the x-y plane, the values of the Intensity feature are thus computed. The user can select a single expressive intention, and the system will play the entire score with that nuance. Trajectories can also be drawn in the space, to experience the dynamic aspects of the expressive content.

The abstract control space S is represented as a set of N triples $S_i = (l_i, p_i, e_i)$ with $i = 1, \ldots, N$ where l_i is a verbal label that semantically identifies an expressive intention (e.g., bright, light, and soft), $p_i \in \mathbb{R}^2$ represents the coordinates in the control space of the expressive intention l_i, and $e_i \in \mathbb{R}^k$ is a vector of features which characterise the expressive intention l_i.

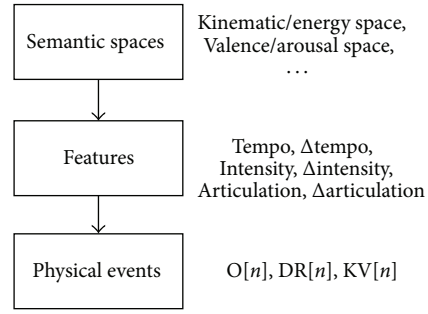

Figure 5: Multilayer model.

S defines the values of the expressive features for the N points of coordinates p_i. To have a continuos control space, in which each point of the control space is associated with a feature vector, a mapping function $f : \mathbb{R}^2 \rightarrow \mathbb{R}^k$ is defined as

$$f(x) = \begin{cases} e_i, & \text{if } x = p_i \\ \sum_{i=1}^{n} a_i \cdot e_i, & \text{elsewhere} \end{cases} \quad (2)$$

with

$$\frac{1}{a_i} = \left\| x - p_i \right\|^2 \cdot \sum_{j=1}^{N} \frac{1}{\left\| x - p_j \right\|^2}. \quad (3)$$

Let x be a point in the control space; the mapping function calculates the corresponding features vector $y = f(x)$ as a linear combination of N features vectors, one for each expressive intention defined in S. The features vectors are weighted inversely proportional to the square of

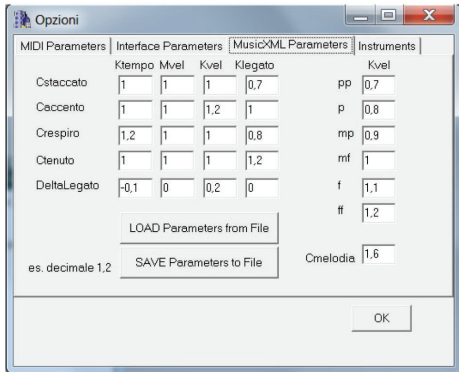

Figure 6: The options window (designed for musicians) that allows the fine tuning of the model. For general public, it is possible to load preset configurations (see the button "load parameters from file").

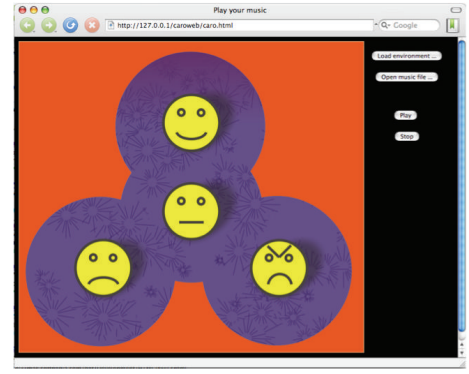

Figure 8: A control space designed for an educational context. The position of the emoticons follows the results of [37].

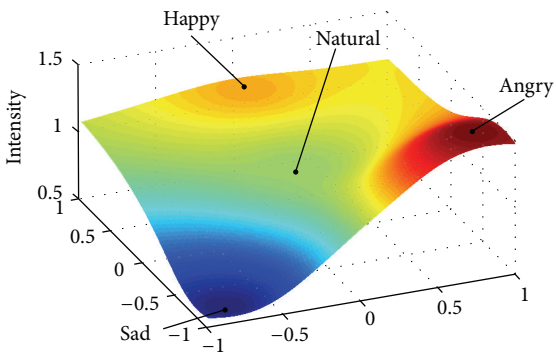

Figure 7: Map between the x-y coordinates in the control space and the Intensity feature.

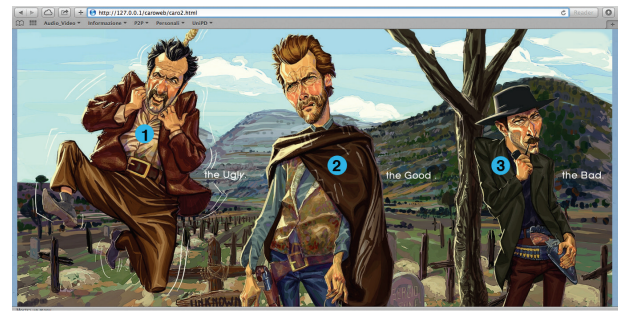

Figure 9: The control space is used to change interactively the sound comment of the multimedia product (a cartoon from the movie "The Good, The Bad and The Ugly" by Sergio Leone, in this case). The author can associate different expressive intentions with the characters (here, see the blue labels 1, 2, and 3).

the euclidean distance between their corresponding expressive intention and x. Finally, the resulting features vector can be used to compute the expressive deviations, represented by the physical layer in Figure 5. For example, the Tempo feature is used to compute the parameters of the nth physical event from the output of the *Naturalizer* module, according the following equations:

$$O_{\text{exp}}[n] = O_{\text{nat}}[n-1] + \left(O_{\text{nat}}[n] - O_{\text{nat}}[n-1]\right) \cdot \text{Tempo}$$
$$\text{DR}_{\text{exp}}[n] = \text{DR}_{\text{nat}}[n] \cdot \text{Tempo}, \tag{4}$$

where O_{exp} and O_{nat} are the onset times of the expressive and natural performance, respectively, and DR_{exp} and DR_{nat} are the event durations.

4.3. User-Defined Control Space for Interaction. The abstract control space constitutes the interface between the user concepts of expression and their internal representation into the system. The user, following his/her own preferences, can create the semantic space, which controls the expressive intention of the performance. Each user can select his/her own expressive ideas and then design the mapping of these concepts to positions and movements on this space. A preferences window (see Figure 6) allows the user to define the set of performance styles, by specifying for the ith style a verbal semantic label l_i, the coordinates in the control space p_i, and the features vector e_i.

As a particular and significant case, the emotional and sensory control metaphors can be used. For example, a musician may find it more convenient to use the Kinetics-Energy space (see Figure 1) because it is based on terms such as bright, dark, or soft that are commonly used to describe musical nuances. On the contrary, a nonmusician or a child may prefer the Valence-Arousal space, with emotions being a very common experience for every listener. The sensory space can be defined by placing the semantic labels (with their associated features) Hard, Soft, Dark, and Bright in four orthogonal points of the space. Similarly the Valence-Arousal space can be obtained by placing the emotional labels Happy, Calm, Sad, and Angry at the four corners of the control space. These spaces are implemented as presets of the system, which can be selected by the user (see, e.g., Figure 8).

Specific and new expressive intentions can be associated with objects/avatars, which are placed in the screen, and a movement of the pointer, from an object to another one, changes continuously the resulting music expression. For example, a different type of music can be associated with the characters represented in the image. Also the object, with its

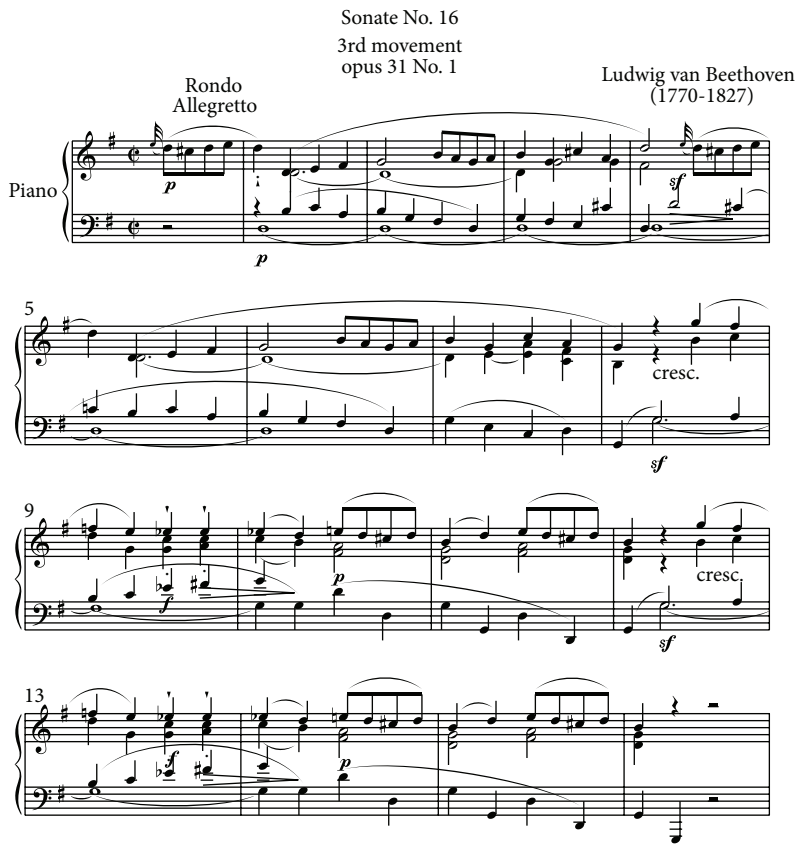

Figure 10: The music excerpt used to evaluate the CaRo 2.0 software. The score has been transcribed in MusicXML format, including all the expressive indications (slurs, articulation marks, and dynamic cues).

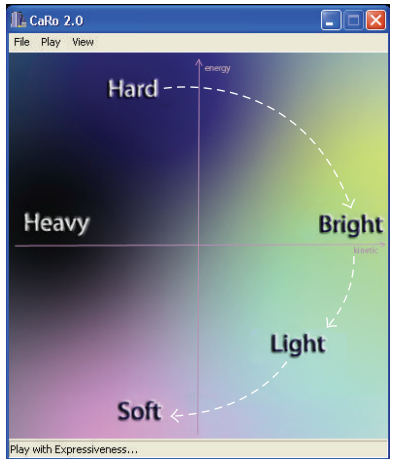

Figure 11: The graphical user interface of CaRo 2.0. The dotted line shows the trajectory drawn for rendering the *hbls* performance.

expressive label and parameters, can be moved, and then the control space can dynamically vary (see, e.g., Figure 9).

More generally, the system allows a creative design of abstract spaces, according to artist's fantasies and needs, by inventing new expressions and their spatial relations.

5. Results

This section reports the results of some expressive music renderings obtained with CaRo 2.0, with the aim of showing how the system works. The beginning of the 3rd Movement of the Piano Sonata number 16 by L. van Beethoven (see Figure 10) was chosen as a case study, because it is characterised by a large number of expressive nuances and, at the same time, it is written with a quite simple musical texture (four voices at most), which makes it easier to visualise and understand the results. Seven performances of the same Beethoven's Sonata were rendered: five performances are characterised by only one expressive intention (bright, hard, heavy, light, and soft) and were obtained keeping the mouse fixed on the corresponding adjective (see the GUI of Figure 11); one performance, named *hbls*, has been obtained drawing a trajectory on the two-dimensional control space going from hard to bright, light, and soft (see the dotted line of Figure 11); finally, a nominal performance was rendered applying no expressive rule or transformation to the music score.

The system rendered the different performances, correctly playing the sequences of notes written in the score. Moreover, expressive deviations were computed depending on the user's preferences specified by means of the control space. Figure 12 shows the piano roll representation of the nominal performance, compared to the performances

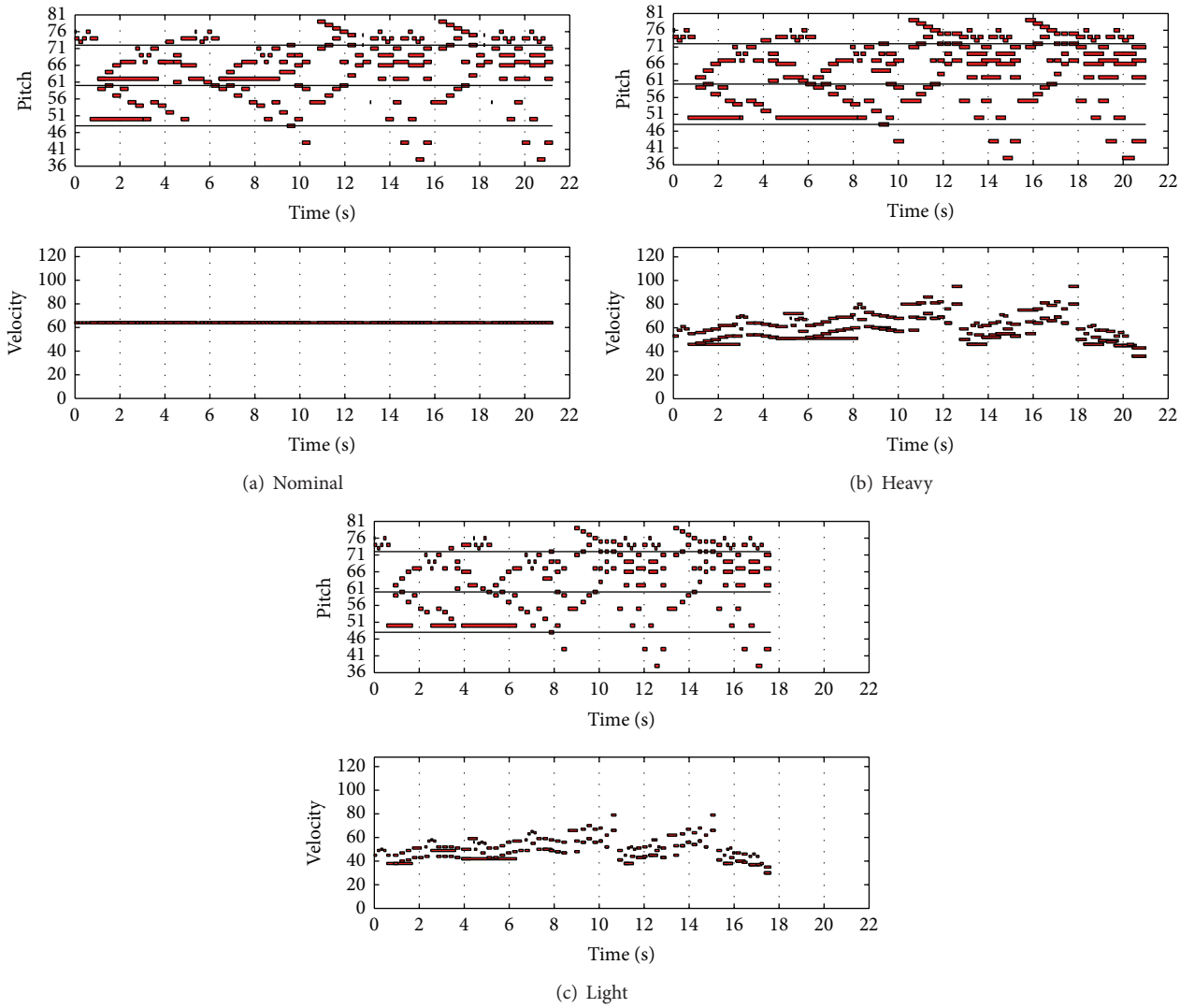

FIGURE 12: Piano roll representation of the performances characterized by heavy (b) and light (c) expressive intention. The nominal performance (a) has been added as a reference.

characterised by the expressive intentions heavy and light. It can be noted that the onset, duration, and Key-Velocity of the notes change in function of the expressive intention. (A piano-roll representation is a method of representing a musical stimulus or score for later computational analyses. It can be conceived as a two-dimensional graph: the vertical axis is a digital representation of different notes; the horizontal axis is a continuous representation of time. When a note is played, a horizontal line is drawn on the piano-roll representation. The height of this line represents which note was being played, the beginning of the line represents the note's onset, the length of the line represents the note's duration, and the end of the line represents the notes offset [38]). The analysis of the main expressive parameters (see Figures 12 and 13) shows that these values match with the correspondent parameters of real performances analyzed in [39]. Moreover, Figure 14 shows how the expressive parameters change gradually following the user's movements between different expressive intentions.

6. Assessment

The most important public forum worldwide in which computer systems for expressive music performance are assessed is the Performance RENdering CONtest (Rencon) which was initiated in 2002 by Haruhiro Katayose and colleagues as a competition among different systems (http://renconmusic.org). During the years the Rencon contest evolved toward a more structured format. It is an annual international competition in which entrants present computer systems they have developed for generating expressive musical performances, which audience members and organisers judge. One of the goals of Rencon is to solve the researchers' common problem, difficulty in evaluating/grading performances generated by his/her computer system within their individual endeavours, by entrants meeting together at the same site [40]. An assessment (from the musical performance expressiveness point of view) of

Table 3: Results of the final stage of the Rencon 2011 contest. Each system was required to render two different performances of the same music piece. Each performance has been evaluated both by the audience attending the contest live and by people connected online through a streaming service.

System	Performance A			Performance B			Total score
	Live	Online	Total	Live	Online	Total	
CaRo 2.0	464	56	520	484	61	545	1065
YQX	484	64	548	432	65	497	1045
VirtualPhilharmony	452	46	498	428	32	460	958
Director Musices	339	33	372	371	13	384	756
Shunji System	277	24	301	277	20	297	598

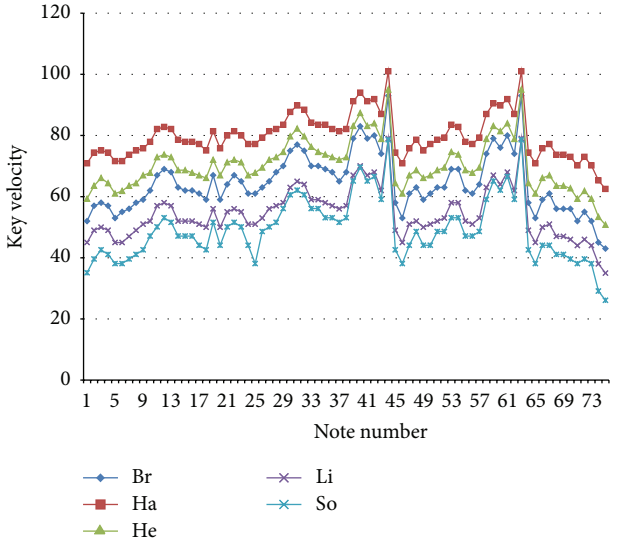

Figure 13: The Key-Velocity values of the performances characterised by the different expressive intentions (bright, hard, heavy, light, and soft). Only the melodic line is reported in order to have a more readable graph.

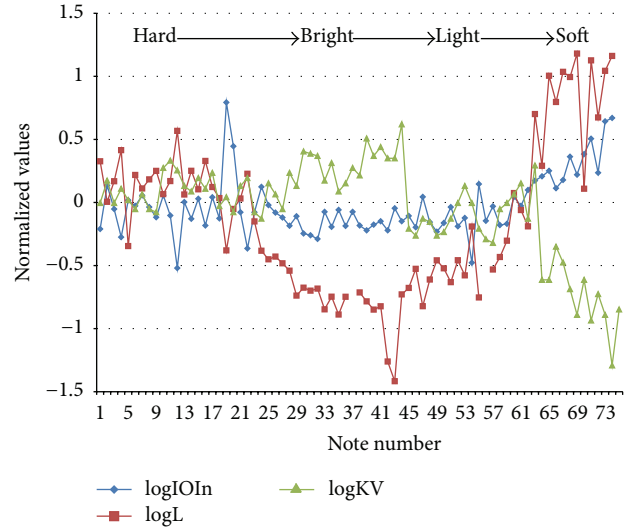

Figure 14: The values of the three main expressive parameters of the performance obtained drawing the trajectory of Figure 14.

CaRo 2.0 was carried out participating in Rencon 2011. Our system was classified for the final stage, where the participating systems were presented during a dedicated workshop. The performances played were openly evaluated by a large (77 participants) and qualified audience of musicians and scientists under the same conditions in the manner of a competition. Each of the five entrant systems was required to generate two different performances of the same music piece, in order to evaluate the adaptability of the system. After listening to each performance, the listeners were asked to rate it from the viewpoint of how much do you give applause to the performance, on a 10-point scale (1 = nothing, 10 = ovation). The CaRo 2.0 system won that final stage (see Table 3). The results as well as the comments by the experts are analysed in [41]. The performances played during Rencon 2011 can be listened to in http://smc.dei.unipd.it/advances_hci/.

7. Conclusion

The CaRo 2.0 could be integrated in the browser engines of the music community services. In this way, (i) the databases keep track of the expressiveness of the songs played; (ii) the users create/browse playlists based not only on the song title or the artist name, but also on the music expressiveness. This allows that, (i) in the rhythms games, the performance will be rated on the basis of the expressive intentions of the user, with advantages for the educational skill and the users involvement of the game; (ii) in the music community, the user will be able to search happy or sad music, accordingly to the user affective state or preferences.

In the rhythm games, despite their big commercial success (often bigger than the original music albums), the gameplay is—almost—entirely oriented around the player's interactions with a musical score or individual songs by means of pressing specific buttons, or activating controls on a specialized game controller. In these games the reaction of the virtual audience (i.e., the score rated) and the result of the battle/cooperative mode are based on the performance of the player judged by the PC. Up to now the game designers did not consider the player's expressiveness. As future work, we intend to insert the CaRo system in these games, so that the music performance is rated "perfect" not if it is only played exactly like the score, but considering also its

interpretation, its expressiveness. In this way, these games can be used profitably also in the educational field.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] N. Bernardini and G. De Poli, "The sound and music computing field: present and future," *Journal of New Music Research*, vol. 36, no. 3, pp. 143–148, 2007.

[2] S. Canazza, G. De Poli, A. Rodà, and A. Vidolin, "An abstract control space for communication of sensory expressive intentions in music performance," *Journal of New Music Research*, vol. 32, no. 3, pp. 281–294, 2003.

[3] S. Canazza, G. De Poli, A. Rodà, and A. Vidolin, "Expressiveness in music performance: analysis, models, mapping, encoding," in *Structuring Music through Markup Language: Designs and Architectures*, J. Steyn, Ed., pp. 156–186, IGI Global, 2012.

[4] S. Canazza, G. De Poli, C. Drioli, A. Rodà, and A. Vidolin, "Audio morphing different expressive intentions for multimedia systems," *IEEE Multimedia*, vol. 7, no. 3, pp. 79–83, 2000.

[5] D. Fabian, R. Timmers, and E. Schubert, Eds., *Expressiveness in Music Performance: Empirical Approaches Across Styles and Cultures*, Oxford University Press, Oxford, UK, 2014.

[6] G. De Poli, "Methodologies for expressiveness modelling of and for music performance," *Journal of New Music Research*, vol. 33, no. 3, pp. 189–202, 2004.

[7] G. Widmer and P. Zanon, "Automatic recognition of famous artists by machine," in *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI '04)*, pp. 1109–1110, Valencia, Spain, 2004.

[8] A. Gabrielsson, "Expressive intention and performance," in *Music and the Mind Machine*, R. Steinberg, Ed., pp. 35–47, Springer, Berlin, Germany, 1995.

[9] P. Juslin, "Emotional communication in music performance: a functionalist perspective and some data," *Music Perception*, vol. 14, no. 4, pp. 383–418, 1997.

[10] S. Canazza, G. De Poli, and A. Vidolin, "Perceptual analysis of the musical expressive intention in a clarinet performance," in *Music, Gestalt and Computing*, M. Leman, Ed., pp. 441–450, Springer, Berlin, Germany, 1997.

[11] G. De Poli, A. Rodà, and A. Vidolin, "Note–by–note analysis of the influence of expressive intentions and musical structure in violin performance," *Journal of New Music Research*, vol. 27, no. 3, pp. 293–321, 1998.

[12] R. Bresin and A. Friberg, "Emotional coloring of computer-controlled music performances," *Computer Music Journal*, vol. 24, no. 4, pp. 44–63, 2000.

[13] F. B. Baraldi, G. De Poli, and A. Rodà, "Communicating expressive intentions with a single piano note," *Journal of New Music Research*, vol. 35, no. 3, pp. 197–210, 2006.

[14] S. Hashimoto, "KANSEI as the third target of information processing and related topics in Japan," in *Proceedings of the International Workshop on Kansei-Technology of Emotion*, pp. 101–104, 1997.

[15] S. Hashimoto, "Humanoid robots for kansei communication: computer must have body," in *Machine Intelligence: Quo Vadis*, P. Sinčák, J. Vaščák, and K. Hirota, Eds., pp. 357–370, World Scientific, 2004.

[16] R. Picard, *Affective Computing*, MIT Press, Cambridge, Mass, USA, 1997.

[17] S. Canazza, G. De Poli, C. Drioli, A. Rodà, and A. Vidolin, "Modeling and control of expressiveness in music performance," *Proceedings of the IEEE*, vol. 92, no. 4, pp. 686–701, 2004.

[18] B. H. Repp, "Diversity and commonality in music performance: an analysis of timing microstructure in Schumann's 'Traumerei'," *The Journal of the Acoustical Society of America*, vol. 92, no. 5, pp. 2546–2568, 1992.

[19] B. H. Repp, "The aesthetic quality of a quantitatively average music performance: two preliminary experiments," *Music Perception*, vol. 14, no. 4, pp. 419–444, 1997.

[20] L. Mion, G. De Poli, and E. Rapanà, "Perceptual organization of affective and sensorial expressive intentions in music performance," *ACM Transactions on Applied Perception*, vol. 7, no. 2, article 14, 2010.

[21] S. Dixon and W. Goebl, "The 'air worm': an interface for real-time manipulation of expressive music performance," in *Proceedings of the International Computer Music Conference (ICMC '05)*, Barcelona, Spain, 2005.

[22] A. Roda, S. Canazza, and G. De Poli, "Clustering affective qualities of classical music: beyond the valence-arousal plane," *IEEE Transactions on Affective Computing*, vol. 5, no. 4, pp. 364–376, 2014.

[23] P. N. Juslin and J. A. Sloboda, *Music and Emotion: Theory and Research*, Oxford University Press, 2001.

[24] A. Camurri, G. De Poli, M. Leman, and G. Volpe, "Communicating expressiveness and affect in multimodal interactive systems," *IEEE Multimedia*, vol. 12, no. 1, pp. 43–53, 2005.

[25] A. Kirke and E. R. Miranda, "A survey of computer systems for expressive music performance," *ACM Computing Surveys*, vol. 42, no. 1, article 3, pp. 1–41, 2009.

[26] G. Widmer, S. Flossmann, and M. Grachten, "YQX plays Chopin," *AI Magazine*, vol. 30, no. 3, pp. 35–48, 2009.

[27] G. Widmer, "Discovering simple rules in complex data: a meta-learning algorithm and some surprising musical discoveries," *Artificial Intelligence*, vol. 146, no. 2, pp. 129–148, 2003.

[28] J. Sundberg, A. Askenfelt, and L. Fryden, "Musical performance: a synthesis-by-rule approach," *Computer Music Journal*, vol. 7, no. 1, pp. 37–43, 1983.

[29] I. Ipolyi, "Innforing i musiksprakets opprinnelse och struktur," *TMHQPSR*, vol. 48, no. 1, pp. 35–43, 1952.

[30] A. Friberg, R. Bresin, and J. Sundberg, "Overview of the KTH rule system for musical performance," *Advances in Cognitive Psychology, Special Issue on Music Performance*, vol. 2, no. 2-3, pp. 145–161, 2006.

[31] S. Tanaka, M. Hashida, and H. Katayose, "Shunji: a case-based performance rendering system attached importance to phrase expression," in *Proceedings of Sound and Music Computing Conference (SMC '11)*, F. Avanzini, Ed., pp. 1–2, 2011.

[32] M. Hamanaka, K. Hirata, and S. Tojo, "Implementing 'a generative theory of tonal music'," *Journal of New Music Research*, vol. 35, no. 4, pp. 249–277, 2006.

[33] F. Lerdahl and R. Jackendoff, *A Generative Theory of Tonal Music*, MIT Press, Cambridge, Mass, USA, 1983.

[34] T. Baba, M. Hashida, and H. Katayose, "A conducting system with heuristics of the conductor 'virtualphilharmony'," in *Proceedings of the New Interfaces for Musical Expression Conference (NIME '10)*, pp. 263–270, Sydney, Australia, June 2010.

[35] M. Good, "MusicXML for notation and analysis," in *The Virtual Score: Representation, Retrieval, Restoration*, W. B. Hewlett and E. SelfridgeField, Eds., pp. 113–124, MIT Press, Cambridge, Mass, USA, 2001.

[36] I. Peretz, L. Gagnon, and B. Bouchard, "Music and emotion: perceptual determinants, immediacy, and isolation after brain damage," *Cognition*, vol. 68, no. 2, pp. 111–141, 1998.

[37] R. Plutchik, *Emotion: A Psychoevolutionary Synthesis*, Harper & Row, New York, NY, USA, 1980.

[38] D. Temperley, *The Cognition of Basic Musical Structures*, MIT Press, Cambridge, Mass, USA, 2001.

[39] S. Canazza, G. de Poli, S. Rinaldin, and A. Vidolin, "Sonological analysis of clarinet expressivity," in *Music, Gestalt, and Computing. Studies in Cognitive and Systematic Musicology*, vol. 1317 of *Lecture Notes in Computer Science*, pp. 431–440, Springer, Berlin, Germany, 1997.

[40] H. Katayose, M. Hashida, G. de Poli, and K. Hirata, "On evaluating systems for generating expressive music performance: the rencon experience," *Journal of New Music Research*, vol. 41, no. 4, pp. 299–310, 2012.

[41] G. de Poli, S. Canazza, A. Rodà, and E. Schubert, "The role of individual difference in judging expressiveness of computer assisted music performances by experts," *ACM Transactions on Applied Perception*, vol. 11, no. 4, article 22, 2014.