Info Vis

Article

Information Visualization
1–14

DOI: 10.1177/1473871616655468
ivi.sagepub.com
SAGE

Understanding the structure of musical compositions: Is visualization an effective approach?

Roberto De Prisco, Delfina Malandrino, Donato Pirozzi, Gianluca Zaccagnino and Rocco Zaccagnino

Abstract

Experienced musicians have the ability to understand the structural elements of music compositions. Such an ability is built over time through the study of music theory, the understanding of rules that guide the composition of music, and countless hours of practice. The learning process is hard, especially for classical music, where the rigidity of the music structures and styles requires great effort to understand, assimilate, and then master the learned notions. In particular, we focused our attention on a specific type of music compositions, namely, music in chorale style (four-voice music). Composing such type of music is often perceived as a difficult task because of the rules the composer has to adhere to. In this article, we propose a visualization technique that can help people lacking a strong knowledge of music theory. The technique exploits graphic elements to draw the attention on the possible errors in the composition. We then developed an interactive system, named *VisualMelody*, that employs the proposed visualization technique to facilitate the understanding of the structure of music compositions. The aim is to allow people to make four-voice music composition in a quick and effective way, that is, avoiding errors, as dictated by classical music theory rules. We have involved 40 people in testing VisualMelody in order to analyze its effectiveness, its usability, and the overall user satisfaction. We partitioned the people involved in the evaluation study into two groups evenly splitting the musical expertise. Then, we had one group use VisualMelody without the visualization facilities and the other using the tool enhanced with our visualization. On average, people in the group that used our visualization were 60% faster and produced music with less errors.

Keywords

Melodic analysis, classical music, visualization, user evaluation

Introduction

Music is a fundamental pillar in every human culture. It exists in many forms and gets perceived in a variety of different ways. Everyday, people listen to music for several reasons, for example, to create a pleasant atmosphere evoking memories and suggesting dreams, influence emotions, get through health problems, express inner feelings, or enhance creativity behaviors, or just for the *pleasure* of listening to it. The everyday musical listening is also strongly related with the mood and/or the preferences of each individual. Moreover, music can be listened to at different levels. People can be interested in sensing the nature of music and what it wants to evoke, without any need to understand its underlying structure. However, a different situation

Department of Computer Science, University of Salerno, Fisciano, Italy

Corresponding author:
Rocco Zaccagnino, Department of Computer Science, University of Salerno, Via Giovanni Paolo II, 84084 Fisciano (SA), Italy.
Email: zaccagnino@dia.unisa.it

arises when dealing with music requires a conscious experience and participation, when musicians have to compose music, when students have to learn difficult musical rules, and so on. In such cases, considerable efforts are required to understand the structure of musical compositions. In this perspective, a difficult task is that of mastering classical music composition rules. Untrained people may only be able to feel the sound elements such as pitch, rhythm, volume, and speed. Conversely, the form of music and the *harmonic, melodic*, and *rhythmic* structural aspects are usually perceived only by musicians who have received extensive training in music theory. The learning curve is steep and makes classical music apparently sophisticated and of difficult comprehension.

Learning musical rules is hard, especially for classical music, where the rigidity of the music structures and styles require considerable efforts. The most common way to learn music is through the study of musical scores, which contain an encoding of the music. However, the analysis of musical scores is a demanding task, and beginners have to spend considerable amount of time to learn the rules of music theory, before being able to fully understand all the hidden details of music. It is well known that musicologists study the harmonic structure of pieces and annotate them by hand. This is a time-consuming process, especially when large corpora have to be analyzed.[1] It is worth to note that the knowledge of music theory and composition techniques is essential to understand the melodic and harmonic structures of musical compositions.

Other researchers have already considered the problem of visualizing the structure of music.[2–4] A critical issue related to music visualization is about the creation of an informative and insightful visualization from which users, with a poor classical music background, are able to *see* the sound and, at the same time, *semantically understand* the underlying structure of music. The most important challenges to address when visualizing musical structures lie in both the preprocessing of the input data (i.e. music elements) and the design of a meaningful and intuitive graphical representation of music. In Malandrino et al.,[2] the authors described and evaluated a graphical representation that allows users to understand the harmonic structure of musical compositions. In this work, we deal with a different musical aspect, that is, the melody, and a more sophisticated problem, that is, the *analysis–compositional process*. In particular, we focus our attention on the *melodic* analysis–compositional process, where the goal is to compose the melodic structure of a music composition by defining the notes of each melodic voice and at the same time by analyzing the *relations* among all voices. Classical music rules require that each single voice proceeds in harmony with the others, according to the well-known melodic rules.[5]

We remark that the melodic analysis is a very important aspect for the music composition process. Usually, the melodic analysis and the melodic compositional process are tasks performed at the same time. In fact, during the compositional process, the musician also performs melodic analysis to make sure that the actual result will be pleasant and musically correct.

In this work, we focus on a specific type of music, namely, *chorale* music.[5,6] A chorale consists of four independent voices, called *bass, tenor, alto*, and *soprano*, that follow classical music rules that composers have to adhere to.[5] When considering four-part music, we can think at two dimensions, a horizontal dimension corresponding to the melody and a vertical dimension corresponding to the harmony. The melodic and the harmonic structures of a musical composition are closely related and usually a good harmonic structure requires a good melodic relation among the voices and vice versa. We have to emphasize that although the number of melodic rules is not high, their application is not a trivial task since the violation of melodic rules has to be verified for each pair of voices and at any instant (more precisely, at every music beat). The higher is the compliance with the rules, the higher is the correctness of the musical composition. It is worth to note that classical musicians are trained in music conservatories with classes that deal exactly with composition rules for four-voice music.

The main contributions of our work are as follows:

- Investigate the effectiveness of Information Visualization techniques during the melodic analysis–compositional process;
- Propose an approach to support users in quickly understanding theoretical rules by exploiting graphical elements that convey music concepts in a meaningful, simple, and pleasant way;
- Build a tool, named VisualMelody, that implements the proposed approach;
- Validate the effectiveness of our approach, in terms of time to complete tasks and their corresponding correctness (i.e. reduced number of errors). We also analyzed the system usability and the overall user satisfaction.

From the *musical point of view*, the main novelty of our work is the design of a visualization approach to address the complexities that arise when studying the melodic structure of music compositions. The approach helps in understanding and remembering the strict rules used in classical compositions for four-voice music. The rules that we consider are *direct fifths, direct octaves, direct unisons*, and *voice crossing* between pairs of melodic lines (the rules will be explained in the "Background" section).

From the *visualization point of view*, the main novelty is the introduction of visual elements side to side with the musical score, to convey *augmented* information, when studying melodic compositions.

The rest of this article is organized as follows: In section "Related work," we discuss relevant work. In section "Background," we define the melodic analysis problem, while in section "Visualization approach for the melodic structure of music," we discuss how we employed the visualization for our goal and the characteristics of our approach. In section "VisualMelody," we present the tool that implements our approach, while in section "Evaluation," we discuss the results of a study aiming at assessing the effectiveness of our idea, the system usability, and user satisfaction. Finally, in section "Conclusion and future work," we conclude with some final remarks and future directions.

Related work

The ability to understand the melodic structure of a musical composition is important in every genre of music, from classical to pop music. In particular, the melodic analysis is an important step for many techniques and problems of classical musical composition because it enhances the ability to compose good music. A rich body of work that aims at exploring the melodic analysis rules has emerged. The problem has attracted numerous computer music researchers pushing them to investigate how to automate the melodic analysis and the melodic composition. Today, automated melodic analysis is an important and interesting music research topic, and many frameworks have been developed. For example, in Illescas et al.,[7] the authors describe a system that analyzes a music score from both the melodic and the harmonic points of view. Each note is tagged as harmonic or not harmonic according to the corresponding chord. The system highlights the tonality changes and performs the harmonic analysis of each chord with respect to its tonal function.

For what concerns visualization of the music, different attempts have been made. *Arc Diagrams* represents one of the first examples to visualize repetitions in music compositions.[8,9] The *Isochords* system[3] is a method for visualizing the chord structure, progression, and voicing of musical compositions represented in MIDI format. Taking advantage of a Tonnetz grid, which emphasizes consonant intervals and chords, Isochords shows how harmony changes over time. Other approaches use three-dimensional (3D) views to visualize music components. Smith and Williams[10] discussed the possibility of visualizing MIDI in 3D space using color to mark timbre. The *comp-i* system[11] shows the structure of music as a whole using 3D piano roll visualization. Its main goal is to allow users to perform visual exploration of a given MIDI dataset in an immersive and intuitive manner. In Fonteles et al.,[12] the authors designed and implemented a simplified 3D particle system to generate real-time animated particle emitter fountains choreographed by music, for visual entertainment and music composition. Another MIDI-based system available in the literature is *Music Animation Machine*.[13] It encompasses a number of visualizations including a basic two-dimensional (2D) piano roll notation for visualizing structure. This visualization is additionally expanded with colors based on pitch classes using the well-known circle of fifths (it shows the relationships among the 12 tones of the chromatic scale, their corresponding key signatures, and the associated major and minor keys). Colors based on a circle of fifths for visualization of tonal distributions and understanding consonance and dissonance intervals have also been explored in other work.[14,15] In his work, Sapp[16] visualizes a hierarchy of key regions of a given composition, where x-axis represents time (from the start to the end of the composition) and y-axis represents the duration of key-finding algorithm's sliding window. When the window size increases, more notes are included and may affect the analyzed tonality. These hierarchical key analysis diagrams are useful for comparing the impact of using different time scales and viewing the harmonic structure and relationships between key regions in the composition.

Finally, in Chan et al.,[4] the authors propose a visualization solution to reveal the semantic structure in classical orchestral works. . In addition to reveal the structure of the compositions, their focus was on how to portray the sophistication and beauty of music.

Background

In this section, we briefly recall some music notions for the benefit of the reader not familiar with them. We also define some formal notations that we will use throughout this article.

In music theory, the notes (more specifically, the pitches, which corresponds to the frequency that produces the notes) are selected according to specific criteria. To explain their organization, we will use the piano keyboard as reference: each key of the piano corresponds to a specific pitch (note), and there are 88 notes, divided into *octaves*. The piano keyboard has roughly 7 octaves, usually numbered from 0 to 6, which contain the 88 keys ($12 \times 7 = 84$ keys in 7 octaves plus 4 extra keys); notes outside this range are hardly used because their frequencies are too low or too high to be pleasantly perceived by our ear. In each octave, there are 12 notes, named C, C# (or Db), D, D# (or Eb), E, F, F# (or Gb), G, G# (or Ab), A, A# (or Bb), and B. Given a note in an octave k ($0 \leqslant k < 7$) having frequency F, then the note with the same name in the octave $k+1$ will have frequency $2F$. Usually, the frequency of 440 Hz is chosen as the reference frequency for the note A in the middle octave of the piano keyboard (this is just a convention, and there are musicians who prefer slightly different values, such as 442 Hz). Using 440 Hz as reference frequency, the note A in the seven octaves of the piano has the following frequencies: 27.5, 55, 110, 220, 440, 880, 1760, and 3520 Hz. As explained before, the piano keyboard has 84 keys for the seven octaves, ranging from (C, 0), that is, C in the octave number 0, through (B, 6), that is, B in the octave number 6. In addition, there are three extra keys, A, A#, and B before the first octave, and one extra key C, after the last octave.

The *interval* between two notes is basically the distance, on a logarithmic scale, of the pitches. We say that two notes i and j have the following intervals:

- Unison, if $i = j$;
- Fifth, if $|i - j| \equiv 7 \pmod{12}$;
- Octave, if $i \neq j$ and $|i - j| \equiv 0 \pmod{12}$.

For a detailed explanation of the above definitions, we refer the reader to a well-known music theory book by Piston.[5] The piano covers the entire range of notes. Other instruments, and the human voices, have a smaller extension, that is, they can produce only a subset of all the notes. For example, a trumpet has an extension ranging from (F#, 3) to (C, 6). The human voice has an extension that is subjective: each individual has his or her own, and the frequencies covered by male voices are lower than the female voices. The frequencies covered by human voices are classified into four classes: *bass*, ranging from (E, 3) to (D, 5); *tenor*, ranging from (C, 4) to (G, 5); *alto*, ranging from (G, 4) to (C, 6); and *soprano* ranging from (C, 5) to (A, 6). Sometimes, intermediate classes are considered (baritone, in between bass and tenor, and half-soprano, in between alto and soprano). This classification is important for this article because the type of music that we considered is music written in four parts (chorale style), whereas each part belongs to one of these four voices.

A music composition is composed of a sequence of measures, and each measure consists of a given number of beats. Each beat is associated with a *chord*, that is, a set of notes. Since we focus on four-part music, the chords that we consider are four-note chords, and each note is sung or played by one of the voices. Thus, each chord contains the note played by each instrument at a specific beat. The sequence of notes for each part is called a *melodic line* (or simply a line). Therefore, a four-part composition is made up of four lines, one for each voice. The way these lines move determines the quality and pleasantness of the music.

Classical music theory established strict rules about the movement of these lines. Mastering these rules is at the heart of classical music composition. Although the rules are defined considering four-part music, they are also relevant for all classical music forms since the structure of each classical music form can be described in terms of four-part music. Moreover, the principles established in these rules apply as well to other styles, such as pop music.

We are now ready to introduce the main rules of classical music theory that we consider in this article. These rules concern both the movement of a single line and the relationship between the movements of two different lines. Single lines should move using the notes of the tonality of the composition. Moreover, jumps that determine specific intervals are preferred to others. As an example, jumps greater than one octave are less preferable than to those within an octave. Hence, since the movements of a single line are quite straightforward to check, we focus our attention on the rules about the relationship between the movements of two different lines.

According to what dictated by music theory rules, for any given pair of lines, some specific patterns should be avoided: (1) two lines that move by creating two consecutive unisons, (2) two lines that create two consecutive octaves or fifths, and (3) two melodic lines that intersect. In Figure 1, we show examples of these forbidden patterns.

Visualization approach for the melodic structure of music

In this section, we describe the visualization technique we designed to facilitate the understanding of the

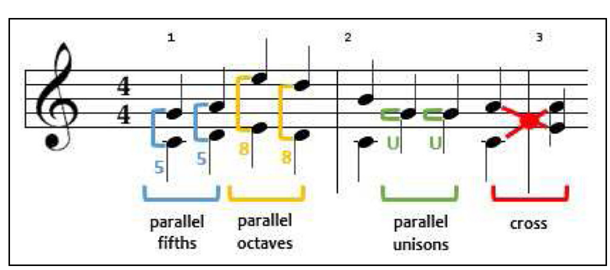

Figure 1. Example of errors on movements of two voices.

melodic structure of music compositions. This technique offers visual clues that simplify the process of melodic analysis during the composition of music, thus facilitating the overall music composition process. We first describe the requirements that we took into account to design the approach and then we explain in detail the mapping between melodic constructs (i.e. intervals and melodic lines) and their graphical representations.

Design process

The design study was defined by following the process discussed in Sedlmair et al.[17] Specifically, in the "Core phase," we collected the requirements from teachers of conservatory classes and from domain experts. The collected requirements were then translated into the definition of our visualization and its implementation and deployment into VisualMelody. In the "Analysis phase," we tested VisualMelody in order to analyze the effectiveness of the idea, that is, whether visualization can allow people (musicians and people generally interested in classical music) to better understand the structure of music composition and avoid the well-known violations of the classical music rules, simply glancing at the visualization, during the composition process. We also tested the system usability and the user satisfaction when interacting with the tool. Finally, easiness, usefulness, and future use of the tool enhanced with visualization were also investigated.

As defined in Sedlmair et al.,[17] just talking to users is necessary but typically not sufficient. Thus, the standard practice is a combination of methods including interviews and observations. We visited three Italian conservatories ("San Pietro a Majella" in Napoli, "Gesualdo da Venosa" in Potenza, and "Martucci" in Salerno), and in each one, we targeted the course of "Composition." For each of these courses, we chose two classes: "Harmony" and "Music Theory." Thus, we involved a total of six classes and interviewed six teachers (one for each class). We noted the following *situations*:

- Slowing down during the composition process, given the necessity of continuously check for possible errors;
- A considerable amount of time spent in recovering from errors and to restart the composition process.

Therefore, we identified, as important requirements, the need of a general method to simplify the process of understanding the structure of a composition and compose music in a quick and efficient way (with a reduced number of errors), as well as in a pleasant way. We then asked teachers and domain experts to express their opinions about the correctness of our observations and the usefulness of an instrument to assist students during the compositional process in order to reduce errors and consequently the time required to compose music. Positive evaluations lead us to proceed with the next phases, that is, translation of the requirements in system's functionalities, and evaluation of effectiveness, usability, and overall user satisfaction.

Graphical elements and concepts

In order to define a visual representation for the melodic structure of a four-part music, we need to choose (1) a visual representation for the melodic lines and (2) a visual representation for the intervals. We decided to visualize the graphical elements side by side and not on the music score. Specifically, they are positioned at the top of the score, in correspondence of the musical constructs that they represent. In this way, we do not alter the layout of the music score, and more importantly, we avoid the confusion (and consequently the annoyance felt by composers) that could be generated if music constructs and graphical elements are contemporaneously represented on the music score.

Melodic line representation. In four-part music, there are four melodic lines, one for each voice. It is quite natural to represent them as lines that connect points, each one corresponding to a note in the sequence. We will refer to this (physical) line as the *trajectory* of the (melodic) line, to avoid confusion between the two types of lines. Each point of the trajectory will be positioned on the vertical axis according to its pitch (frequency). In this way, a trajectory will give information about the movement of the corresponding melodic line. The trajectory is drawn using a color that indicates the corresponding voice. Specifically, we use the BLACK color for the *bass* line, the BLUE color for the *tenor*, the YELLOW color for the *alto*, and the GREEN color for the *soprano*. We apply a semitransparent transformation for trajectories with no violation of the rules. Figure 2 shows the trajectories that represent the melodic lines in the last two measures of Bach's chorale BWV 606.

Two trajectories that cross each other highlight a voice intersection error that should be avoided (see Table 2). Hence, we visualize each crossing point using a RED dot. Also, we mark the intersected trajectories with the RED color. Figure 3 shows the same example of Figure 2, where we introduced a melodic error caused by the intersection of the *alto* and the *soprano* lines. Both the intersection points and the

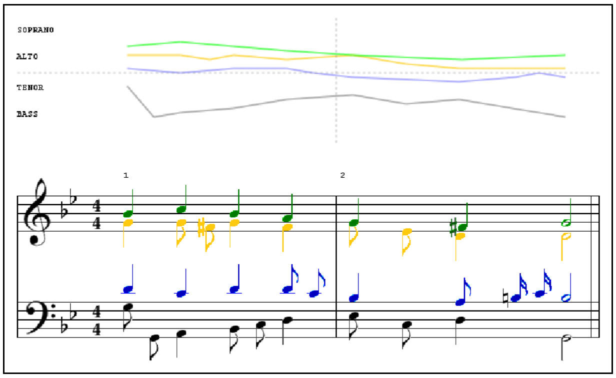

Figure 2. Visualization of the trajectories in the last two measures of BWV 606.

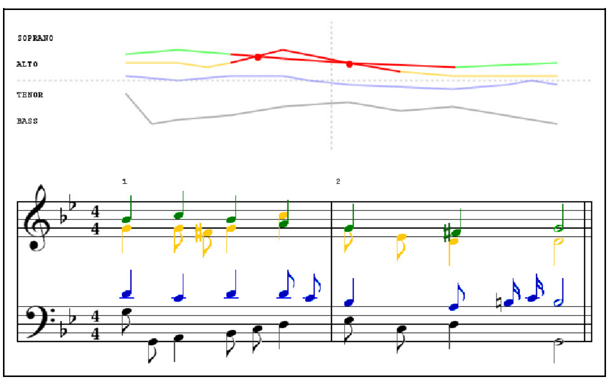

Figure 3. Visualization of the trajectories in the last two measures of BWV 606 with an added intersection, highlighted in RED.

segments of the trajectories involved in this intersection are marked with a RED color.

Interval representation. An interval is basically the "distance" between two notes. Hence, a graphical representation of an interval must highlight such a distance (see Figure 4). The intervals that we want to highlight and that represent a violation of the composition rules are the unisons, the fifths, and the octaves. Each of these intervals will be represented as an ellipse whose two focal points are separated by a distance that roughly represents the "length" of the interval. For the unisons, the ellipse will be a small circle. For fifth and octave intervals, the distance between the two focal points of the corresponding ellipse will be stretched in order to represent the interval itself. To help distinguishing the different types of intervals, we use three colors for drawing the ellipses: GREEN for the unisons, BLUE for the fifths, and YELLOW for the octaves. We apply a semitransparent transformation

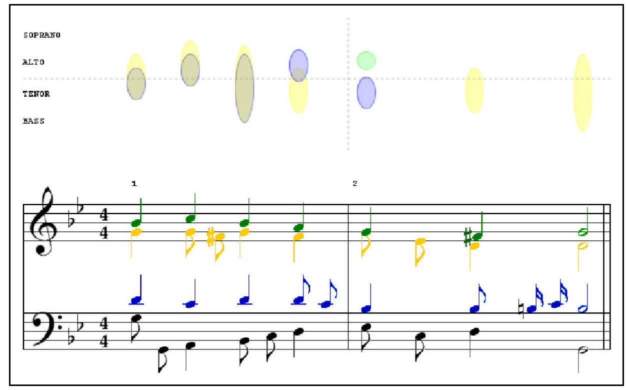

Figure 4. Visualization of the intervals in the last two measures of Bach's chorale BWV 606.

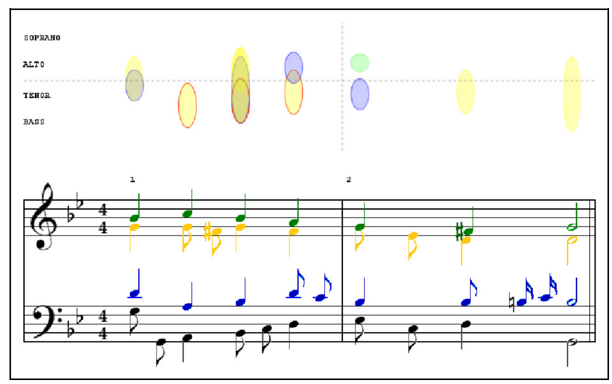

Figure 5. Visualization of the unison, fifth, and octave intervals in the last two measures of Bach's chorale BWV 606 modified with parallel octaves.

for ellipses with no violation of the rules. Every violation of the rules is marked with a RED border on the corresponding ellipses.

Figure 5 shows the same example of Figure 4, where we introduced parallel octave errors. Finally, in Figure 6, we show a complete visualization of an excerpt of four measures from Bach's chorale, as it is shown by VisualMelody tool.

It is worth to note that in some cases, an overlapping of ellipses can occur. As an example, if at a given moment an error occurs between the *bass* and the *soprano* lines and between the *tenor* and the *alto* lines, then we will have a superposition of two ellipses. When an overlap among ellipses occurs, the intersected area will assume the color that is the combination of the overlapped ellipses' colors.

VisualMelody

In this section, we describe the tool, named *VisualMelody*, that implements the visualization

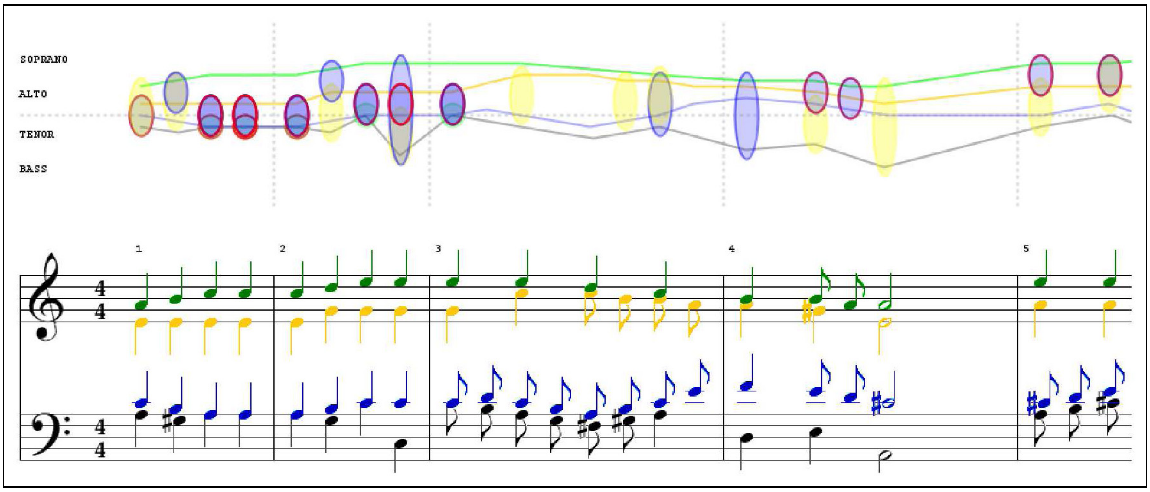

Figure 6. Visualization of an excerpt of four measures from Bach's chorale.

described in the previous section. We first describe the tool; afterward, we provide a simple user scenario.

Functionalities and use case

VisualMelody has been designed and developed to provide individuals with augmented information about music and its background structure, in order to complement the users' experience and facilitate the study of classical compositions and melodic analysis.

The tool is also able to assist people during the composition of four-part music and, in particular, during the definition and the analysis of the melodic structure of the composition itself. VisualMelody can also be used for the melodic analysis of chorales already composed.

While making compositions, users can on-the-fly check for errors deriving from violation of the rules dictated by the classical music theory. As described in section "Introduction," the main difficulty does not lie in the number of rules but on the overlapping of voices at any instant of time (for each score's position), whereas for each pair of voices, we have to check for a possible error. Often, the action of correcting an error may cause another error on the remaining voices. Our visualization, showing the trend lines, could be a useful instrument to recover the error. The tool, as shown in Figure 7, has been implemented in Java, using the Swing API for the development of the graphical user interface and the JFugue library[18] for music programming.

The functionalities provided by VisualMelody can be summarized as follows:

- *Music editing*. VisualMelody provides a music editor for the composition of four-voice music. The tool is designed to edit the most common musical duration, from semiquavers (16th) to semibreves (whole);
- *Melodic visualization*. VisualMelody allows to display a visual representation of the melodic structure of the music. The provided representation follows the rules defined by the approach described in section "Visualization approach for the melodic structure of music";
- *Music playing*. VisualMelody allows to play the composed music. During the configuration phase, it is possible to assign a specific instrument to be used to play each voice;
- *Save and load*. Users can save and load both the score and its corresponding melodic visualization (in a specific.vis format).

In the following, we describe a possible real scenario where VisualMelody can be used in a music class:

- The teacher provides students with explanations about the theoretical concepts needed to understand the melodic structure of musical compositions (standard learning) and classical melodic rules;
- The teacher next proceeds with the explanation about the foundation rules of our visualization approach (innovative learning);
- Students absorb the theoretical concepts as well as the idea behind the visualization approach;
- The teacher provides students with a set of chorales for training purposes (that can be loaded through the tool);
- Students start with the learning process. They are encouraged to write new four-part music with the aid of the representation provided by VisualMelody.

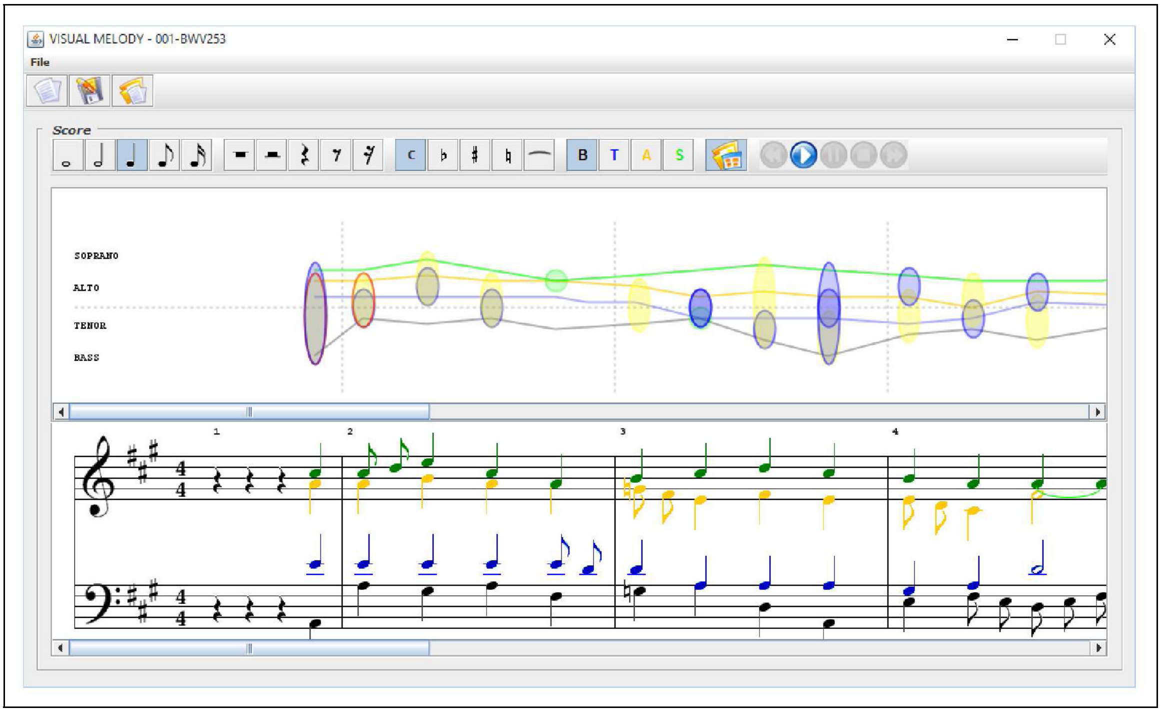

Figure 7. VisualMelody tool. Given a music composition (in the bottom area), its melodic structure is automatically visualized in the top area. The example is in A Major key, but this option is customizable since users can choose any type of key.

Table 1. Participant's demographics.

	Number	Percentage
Total participants	40	100
Gender		
Male	35	87.5
Female	5	12.5
Age (years)		
25–30	15	37.5
30–35	11	27.5
35 +	14	35.0
Education level attained		
Bachelors	3	7.5
Master's	13	32.5
Conservatory	22	55.0
Time spent playing music per day (h)		
< 1	5	12.5
1–2	11	27.5
2 +	24	60.0

The teacher can assist students during the training phase and only observe them during the composition phase. At this stage, in fact, students can leverage VisualMelody functionalities. Specifically, the tool is able to show, for each musical composition, the corresponding melodic visualization. If errors were made, students will immediately be aware of them because the tool will provide clear visual clues about them. Students can immediately correct the errors, while the tool provides real-time analysis of the new score drawing the new corresponding visualization. The software, a user guide with a detailed description of the tool functionalities, and some examples of J.S. Bach's chorales, which could be loaded through the tool, are available anonymously online (https://goo.gl/j4JZ2b).

Evaluation

In this section, we first describe the employed methodology and then the results obtained when 40 people were involved in our evaluation study aiming at assessing the effectiveness of our idea, the usability of the tool we have developed, and the user satisfaction when interacting with it. Finally, easiness, usefulness, and future use of the tool enhanced with visualization were also investigated. In our evaluation study, we followed the standard human–computer interaction (HCI) methodology,[19] commonly applied in different contexts,[20,21] as well as visualization.[22–24]

Method

As shown in Table 1, recruits comprised 40 participants among music students (45%) and music experts (with a conservatory degree, 55%). The sample was mostly male students (87% male and 13% female students) with a mean age of 33 years (standard deviation (SD) = 9 years). Students, with basic knowledge of music, were recruited at the Dipartimento di Informatica, through flyers and word-of-mouth advertising, while music experts were recruited among students and teachers, through flyers and email invitation, at the conservatories of Salerno, Napoli, and Potenza. All participants were volunteers and with normal vision (not colorblind). According to the between-group design,[19] we divided the participants into two groups: 20 participants performed tasks using a *Standard Approach*, that is, using VisualMelody to compose music without any graphical representation, while the other 20 participants performed tasks by composing music with the aid of visualization clues (i.e. *Visual Approach*). Additionally, to ensure equal allocation of subgroups of participants to each experimental condition, we stratified the sample on both gender (even if the number was very low) and music background.

The aims of this study were as follows:

- Analyze whether visualization can facilitate the study of the rules used in classical music (N = 40) and compose music in a quick and effective way (with the smallest number of errors or violations);
- Collect users opinions about the usefulness of the proposed visualization (for participants testing the *Visual Approach*, N = 20);
- Derive the general audience's opinion and reaction to the developed software and its long-term usage (N = 40).

The study envisioned four different phases in which we carried out a *Preliminary Survey*, a *Training Phase*, a *Testing Phase*, and finally a *Summary Survey*, respectively (see Figure 8), as defined in other contexts.[25–27]

At the beginning of each session, participants were briefly introduced to the purpose of the study and the upcoming tasks. In the first phase, we asked participants to fill in a preliminary questionnaire asking for demographic information and information about music expertise and background.

In the Training Phase, users were given a 5-min training period to become familiar with the tool. Users were free to explore and test any of the provided functionalities. We provided them with only basic information on how to use it. Then, they were asked to complete three tasks (Testing Phase). Specifically, for each task, we established the bass line, and we asked participants to compose the *tenor, alto*, and *soprano* lines. Tasks were administered in an increasing difficult level. Participants did not have a time limit for each task. Performance data were collected by

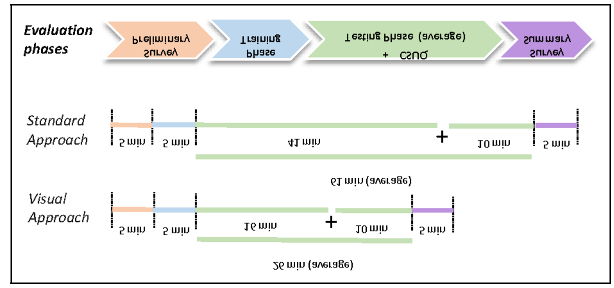

Figure 8. Time line of a single-user evaluation test. On average, the Testing Phase lasted 16 min for the Visual Approach against 41 min for the Standard Approach.

VisualMelody, and in particular, we recorded for each performed task, the completion time as well as the number of inserted or deleted notes. As said before, half of the participants used VisualMelody without any graphical representation, and the other half used the tool enhanced with visual elements.

Tasks were performed on a computer with i7 2.60-GHz CPU, 4.00 GB of main memory, and running Windows 10 OS. Users were not directly monitored, so that they could feel free to test and explore the tool, but they could call for assistance if they did not understand any of the instructions posed. The test was performed in an isolated environment in order to avoid distractions due to the presence of other people. At the end of the Testing Phase, we asked users to spend other 10 min to fill a modified version of the Computer System Usability Questionnaire (CSUQ).[28] The aim was to provide additional information about system usability and user satisfaction when using VisualMelody. To avoid respondents to make arbitrary choices (since puzzled about which one to select) and result in a greater spread of the data,[29] we decided to use here a 5-point Likert scale, with a "1" indicating "Strongly disagree" and a "5" indicating "Strongly agree."

In the final phase, we asked participants to fill in a summary questionnaire, in order to infer the perception about the easiness of the tool and opinions about the intention to use the tool in the future. The entire study lasted between 35 and 45 min for users testing the Visual Approach and between 45 and 70 min for users testing the Standard Approach (see Figure 8). The full test schedule took 2 weeks, according to participants' availability. All questionnaires, translated into Italian for our study, are available anonymously online (https://goo.gl/j4JZ2b).

Finally, the normality assumption (data tested for both normal and log-normal distributions), assessed with the Shapiro–Wilk goodness-of-fit test,[30] was violated. For this reason, nonparametric (Mann–Whitney

U) tests were applied to study differences between the two approaches (i.e. *Standard Approach* and *Visual Approach*). The internal consistency of multi-item scales was checked using Cronbach's[31] alpha. Finally, questionnaire responses were analyzed using SPSS version 20 (http://www-01.ibm.com/software/analytics/spss/).

Melodic evaluation

To evaluate the melodic cost of a chorale C, we compute the melodic value of C by performing an "exception analysis" to identify stylistic anomalies and errors.

Each type of exception corresponds to a *melodic weight* and a *melodic coefficient*. About the melodic weights, we use an approach similar to that proposed in Taube.[32] Each exception has an associated severity level that indicates its relative importance: "warning" and "error." A warning exception is intended to highlight a feature that might be stylistically unusual; an error exception indicates a problem that should be corrected. We assign the melodic weight to an exception on the basis of its severity level: 2 for errors and 1 for warnings. We consider two exception classes: motion exceptions and voicing exceptions.

About the melodic coefficients, we derived them by performing a statistical analysis over a large corpus of chorales composed by J.S. Bach, ranging from chorale BWV 253 through chorale BWV 306 and from chorale BWV 314 to chorale BWV 438. More in detail, we have written a custom script that analyses the chorales and that extracts information from the used harmonization; in particular, we looked for adjacent chords and we computed the percentages of eventual exceptions.

Formally, given a chorale C, its melodic cost is given by

$$f_m(C) = \sum_i a_i w_i$$

where the index i runs over all errors and the values of coefficients a_i and weights w_i are reported in Table 2.

In general, when composing a chorale C, the objective is obviously to minimize $f_m(C)$.

Results

Results of the Preliminary Survey (see Figure 9) allowed us to outline the profile of our participants. The results show that participants have high experience with music (40% were Expert or More Than Expert) and a slightly less experience with melodic analysis (35% Expert or More Than Expert). Additionally, 43% spend >2 h/day playing music. Moreover, as we can see in Figure 10, our participants had high familiarity with music notation software (15% declared to have never used a software to compose music).

Results of the second part of the study are shown in Figures 11 and 12. Specifically, in Figure 11, we show the time required to complete each task (i.e. completion time) for both involved groups, while in Figure 12, we show the cost (value) of each composition.

Participants who tested the *Visual Approach* were able to compose chorales in less time when compared with users testing the tool without visualization. The correctness of the composition was evaluated according to the rules defined in section "Melodic evaluation." We recall that the lower the cost, the better the solution.

Table 2. Single-voice and two-voice errors: weights and coefficients.

Exception class	*w*	*a*	Meaning
Motion exceptions			
Parallel octaves	2	.075	Consecutive octaves
Parallel fifths	2	.11	Consecutive fifths
Parallel unisons	2	.04	Consecutive unisons
Voice exceptions			
Voice intersection	1	.23	Voice above or below

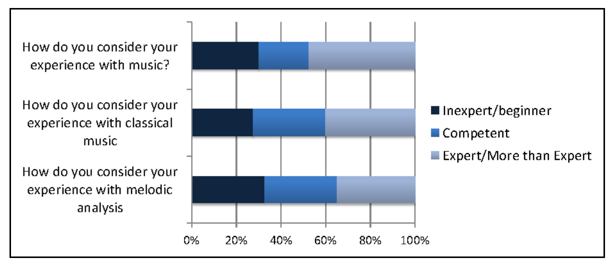

Figure 9. Experience with music and melodic analysis.

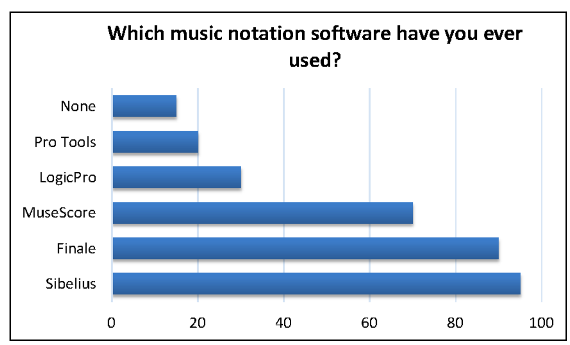

Figure 10. Most used music notation software (%).

Table 3. Task completion times and cost metrics.

	Approach	Tasks			U, p-value
		Task 1	Task 2	Task 3	
Time	Visual	318	330	329	-5.41, $p < .001$
	Standard	818	840	831	
Cost	Visual	0.32	0.31	0.27	-5.41, $p < .001$
	Standard	3.96	3.39	3.56	

Reported test statistics (U for Mann–Whitney). Differences for all tasks are significant for $p < .001$.

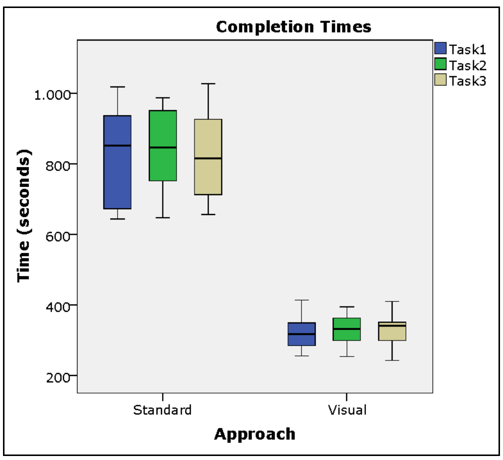

Figure 11. Comparison of completion times for each task.

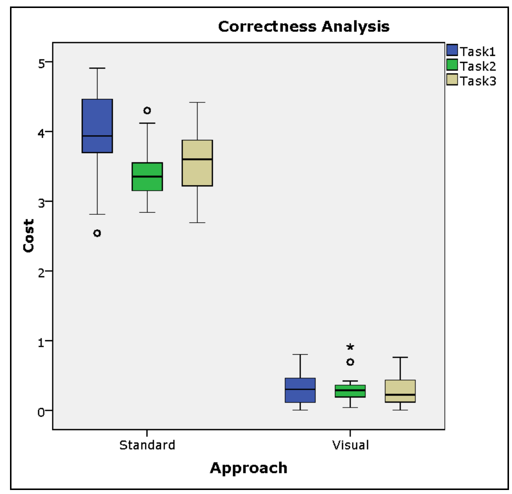

Figure 12. Comparison of the costs for each task.

Composing music with the aid of visualization elements allowed to save on average 60% of the time required to perform the same task without visualization. Additionally, the group testing the *Visual Approach* was able to reduce the number of melodic errors by at least 85%. This is a pretty important result because it means that people not using the visual tool made many mistakes. Table 3 reports a summary of the comparison.

Information about completion times and errors was logged by the tool in a CSV file. We also recorded the number of notes added and deleted. Users testing the *Visual Approach* were able to significantly ($p < .001$) reduce the number of insertions and deletions (see Figure 13), highlighting how our approach, by visualizing the possible violations, allows users to avoid errors and save time.

Users' perceptions and general satisfaction about the experimented tool were analyzed through our modified version of the CSUQ. Items in CSUQ relate to effectiveness, ease of use, pleasantness of the system interface, and overall satisfaction. Specifically, we computed five factor scores: Effectiveness and Efficacy, Usefulness, Satisfaction, Easy of Use and Learning, and finally Clearness. The reliability value for these scores was .96 (Cronbach's alpha), above the recommended value of .70 given in the literature.[33]

As we can see from Figure 14, all questions were positively rated. Users testing the *Visual Approach* provided more positive results, with a significant difference mainly with regard to Usefulness (e.g. question Q15: "VisualMelody has all the functions and capabilities I expect it to have," $U = 60$, $p < .001$) and Satisfaction factors (e.g. question Q16: "Overall, I am satisfied with VisualMelody" $U = 58$, $p < .001$). An important result is that the most rated score was the Efficacy, confirming that users perceived the tool very useful to improve performance (see questions Q3, Q4, and Q8 in section "CSUQ" of the questionnaire available online).

Finally, the third part of the study allowed us to report on the easiness of interacting with the tool and users' intention to use the tool in the future. Specifically, the results of the Summary Questionnaire are shown in Table 4.

The most interesting result is about the positive attitude of the whole sample toward the long-term usage of the tool (mean of 4.9 and 4.2 for the *Visual Approach* and *Standard Approach*, respectively).

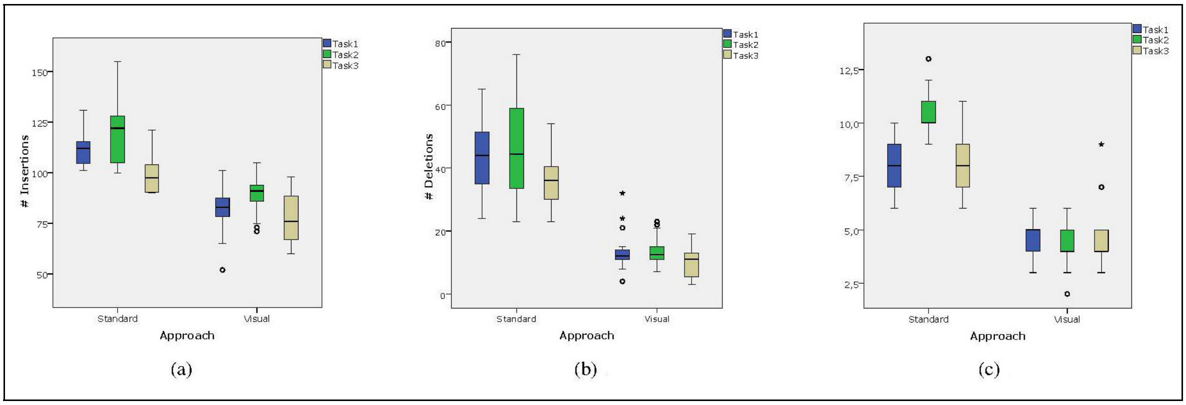

Figure 13. Comparison of the analyzed approaches in terms of note interactions with VisualMelody. Statistical differences exist between the two approaches for all the analyzed operations ($p < .001$): (a) number of note insertions—U values of 0.5, 7, and 26.5 for task 1, task 2, and task 3, respectively; (b) number of note deletions—U values of 3.5, 0.5, and 0.0 for task 1, task 2, and task 3, respectively; and (c) average time for each operation—U values of 3, 0.0, and 20 for task 1, task 2, and task 3, respectively.

Table 4. Summary Survey results.

Question	Visual	Standard	U, p-value
It was simple to use VisualMelody	4.4	3.9	126, **.046**
It was simple to add notes	4.2	3.8	177, .552
It was simple to delete notes	4.1	3.6	148, .163
It was simple to modify the height of a note	4.6	3.7	78, **.001**
It was simple to modify the duration of a note	3.6	3.2	126, **.046**
Do you will continue to use the system in the future?	4.8	4.2	131, .063

Rating on a 5-point Likert scale. U test statistics for Mann–Whitney. Statistically significant results are marked in bold.

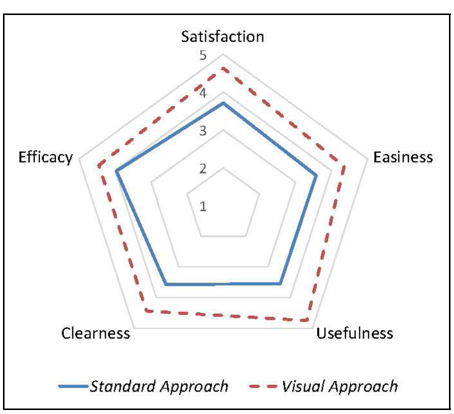

Figure 14. Results across all subscales. Rating on a 5-point Likert scale. Cronbach's $\alpha = .96$.

Finally, as we can see in Table 5, participants who tested the *Standard Approach* replied positively when interviewed about the usefulness of a means able to visualize the melodic lines during the composition process. However, participants who tested the *Visual Approach* stated that they found out very useful the idea of having melodic lines visualized simultaneously during the composition process. They found out this approach very useful, in terms of both productivity and quality of the final result (as also underlined in informal discussions at the end of the study).

Conclusion and future work

In this article, we defined a visualization to display the structure of four-voice music composition with the aim of making quick and effective the study of rules for classical music composition. Musicians and general audience can be provided with a mechanism able to clarify complex relationships in music using visual clues, visualized on the fly, during the music composition process. We developed a tool, named VisualMelody, to provide an implementation of this approach, by considering the most important subset of melodic rules (i.e. parallel errors).

An evaluation study, that involved 40 participants, allowed us to assess the effectiveness of our idea in

Table 5. Summary Survey results.

	Question	Mean (standard deviation)
Q_S	Could you find helpful a tool that shows a graphical representation of the melodic lines during a music composition process?	4.7 (0.7)
Q_V	Do you think it was useful the idea of graphically show melodic lines during a music composition process?	4.9 (0.4)

Question Q_S submitted to the Standard Group, Q_V to the Visual Group. Rating on a 5-point Likert scale.

terms of time to complete tasks and their corresponding correctness. Users who tested VisualMelody with our visualization were 60% faster than the counterpart that tested the tool without any visualization while the number of melodic errors was highly reduced.

VisualMelody was tested in order to analyze system usability and user satisfaction. Participants who tested the *Visual Approach* provided us with positive feedback about the usefulness of the tool (mean 4.8, 5-point Likert scale) and the overall satisfaction (mean 4.6, 5-point Likert scale). While the group that tested the *Visual Approach* expressed high satisfaction about the use of the tool enhanced with visualization, all participants expressed their intention to continue to use the tool in the future.

As future work, we are planning a further experiment, involving classes from many conservatories, in order to validate our approach *directly* in the classroom, the first place where rules have to be learned and assimilated in order to effectively compose classical music. Moreover, we are also planning an extensive and representative experimental study involving a large sample of people, including older age groups, with more diversified music skills. The aim is to study how the general audience, for example, users who are not familiar (or have low familiarity) with the considered domain, would react to the proposed approach. In this work, we want to mainly assess the usefulness and the effectiveness of Information Visualization in assisting people during the melodic composition process. As further work, it could be interesting to analyze the effectiveness of colors to convey information as well the aesthetic choices in terms of colors, shapes, and types of lines (dotted, continuous, and dashed), given their role in improving efficiency, effectiveness, and attractiveness.[34,35] Another interesting future direction could be to widen the set of melodic rules for which visualization can help in reducing errors.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

References

1. De Haas W, Magalhães J, Wiering F, et al. Automatic functional harmonic analysis. *Comput Music J* 2013; 37(4): 37–53.
2. Malandrino D, Pirozzi D, Zaccagnino G, et al. A color-based visualization approach to understand harmonic structures of musical compositions. In: *Proceedings of the 19th international conference on information visualisation*, Barcelona, 22–24 July 2015, pp. 56–61. New York: IEEE.
3. Bergstrom T, Karahalios K and Hart JC. Isochords: visualizing structure in music. In: *Proceedings of graphics interface*, Montréal, QC, Canada, 28–30 May 2007, pp. 297–304. New York: ACM.
4. Chan W, Qu H and Mak W. Visualizing the semantic structure in classical music works. *IEEE T Vis Comput Gr* 2010; 16(1): 161–173.
5. Piston W. *Harmony* (ed M DeVoto). 5th ed. New York, NY: W. W. Norton & Company, Inc., 1987.
6. Czarnecki C. *J.S. Bach 413 chorales: analyzed—a study of the harmony of J.S. Bach*. Solvay, NY: Seezar Publications, 2013.
7. Illescas PR, Rizo D and Iesta JM. Harmonic, melodic, and functional automatic analysis. In: *Proceedings of the 2007 international computer music conference*, Copenhagen, Denmark, 27–31 August 2007, vol. I, pp. 165–168. Michigan publishing.
8. Wattenberg M. The shape of song, http://turbulence.org/project/the-shape-of-song/
9. Wattenberg M.Arc diagrams: visualizing structure in strings. In: *Proceedings of the IEEE symposium on information visualization, INFOVIS*, Boston, MA, 28–29 October 2002, pp. 110–116. New York: IEEE.
10. Smith S and Williams G. A visualization of music. In: *VIS97 IEEE Visualization '97 Conference*, Phoenix, AZ, USA 18–24 October 1997, pp. 499–503. Los Alamitos, CA, USA: IEEE Computer Society Press.
11. Miyazaki R, Fujishiro I and Hiraga R. Exploring MIDI datasets. *In: ACM SIGGRAPH 2003 Sketches and Applications, SIGGRAPH '03*, San Diego, California, 2003, pp 1–1. New York, NY, USA: ACM.
12. Fonteles JH, Rodrigues MAF and Basso VED. Creating and evaluating a particle system for music visualization. *J Visual Lang comput* 2013; 24(6): 472–482.
13. Malinowki S. The music animation machine "music worth watching," http://www.musanim.com/

14. Ciuha P, Klemenc B and Solina F. Visualization of concurrent tones in music with colours. In: *Proceedings of the international conference on multimedia*, Firenze, 25–29 October 2010, pp. 1677–1680. New York: ACM.
15. Mardirossian A and Chew E. Visualizing music: tonal progressions and distributions. In: *Proceedings of the 8th international conference on music information retrieval*, Vienna, 23–27 September 2007, pp. 189–194. Austrian Computer Society (OCG).
16. Sapp CS. Visual hierarchical key analysis. *Comput Entertain* 2005; 3(4): 1–19.
17. Sedlmair M, Meyer MD and Munzner T. Design study methodology: reflections from the trenches and the stacks. *IEEE T Vis Comput Gr* 2012; 18(12): 2431–2440.
18. Koelle D. Music programming for Java and JVM languages, http://www.jfugue.org/
19. Lazar J, Feng JH and Hochheiser H. *Research methods in human-computer interaction*. Chichester: John Wiley & Sons, 2010.
20. Leon P, Ur B, Shay R, et al. Why Johnny can't opt out: a usability evaluation of tools to limit online behavioral advertising. In: *Proceedings of the SIGCHI conference on human factors in computing systems*, Austin, TX, 5–10 May 2012, pp. 589–598. New York: ACM.
21. Malandrino D, Manno I, Palmieri G, et al. Tailorable infrastructure to enhance mobile seamless learning. *IEEE Trans Learn Technol* 2015; 8(1): 18–30.
22. Riche NH and Dwyer T. Untangling Euler diagrams. *IEEE T Vis Comput Gr* 2010; 16(6): 1090–1099.
23. Blake A, Stapleton G, Rodgers P, et al. How should we use colour in Euler diagrams? In: *Proceedings of the 7th international symposium on visual information communication and interaction*, Sydney, NSW, Australia, 5–8 August 2014, pp. 149–158. New York: ACM.
24. Al-Musawi M, Ledesma A, Nieminen H, et al. Implementation and user testing of a system for visualizing continuous health data and events. In: *Proceedings of the 2016 IEEEEMBS international conference on biomedical and health informatics (BHI)*, Las Vegas, NV, 24–27 February 2016, pp. 156–159. New York: IEEE.
25. Malandrino D, Scarano V and Spinelli R. How increased awareness can impact attitudes and behaviors toward online privacy protection. In: *Proceedings of the international conference on social computing, SocialCom 2013*, Alexandria, VA, 8–14 September 2013, pp. 57–62. New York: IEEE.
26. Fish A, Gargiulo C, Malandrino D, et al. Visual exploration system in an industrial context. *IEEE T Ind Inform* 2016; 12(2): 567–575.
27. Conati C, Carenini G, Hoque E, et al. Evaluating the impact of user characteristics and different layouts on an interactive visualization for decision making. *Comput Graph Forum* 2014; 33(3): 371–380.
28. Lewis JR. IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. *Int J Hum Comput Int* 1995; 7(1): 57–78.
29. Krosnick J and Presser S. Chapter 9. Question and questionnaire design. In:Wright J and Marsden P (eds) *Handbook of survey research*. 2nd ed. Bingley: Emerald, 2009, pp. 263–313.
30. Shapiro SS and Wilk MB. An analysis of variance test for normality (Complete samples). *Biometrika* 1965; 52(3–4): 591–611.
31. Cronbach LJ. Coefficient alpha and the internal structure of tests. *Psychometrika* 1951; 16(3): 297–334.
32. Taube H. Automatic tonal analysis: toward the implementation of a music theory workbench. *Comput Music J* 1999; 23(4): 18–32.
33. Nunnally JC and Bernstein IH. *Psychometric theory*. New York: McGraw-Hill, 1994.
34. Stasko J, Catrambone R, Guzdial M, et al. An evaluation of space-filling information visualizations for depicting hierarchical structures. *Int J Hum Comput St* 2000; 53(5): 663–694.
35. Cawthon N and Moere AV. The effect of aesthetic on the usability of data visualization. In: *Proceedings of the 11th international conference on information visualization*, Zurich, 4–6 July 2007, pp. 637–648. New York: IEEE.