Multimed Tools Appl
DOI 10.1007/s11042-016-3582-0

VOICE2TUBA: transforming singing voice into a musical instrument

José L. Santacruz[1] · Lorenzo J. Tardón[1] · Isabel Barbancho[1] · Ana M. Barbancho[1] · Emilio Molina[1]

Received: 30 July 2015 / Revised: 12 April 2016 / Accepted: 3 May 2016

Abstract In this paper, a scheme to synthesize and convert singing voice into tuba sound is presented. First, our method estimates the fundamental frequency (F_0) and the aperiodicity of a monophonic audio signal, in order to obtain the pitch and volume variations of human voice. Then, the parameters extracted are used to generate a musical excerpt emulating a certain musical instrument (tuba) in such a way that the melody resembles the original sung song. To this end, two different generation approaches are devised. One of them is based on additive signal synthesis from harmonic amplitudes. The other one converts the F_0 curve into a MIDI stream, in order to allow the play back with a virtual tuba.

Keywords Voice transformation · Sound synthesis · Music application

1 Introduction

Nowadays, interactive voice applications have become an interesting field for development and study. Automatic transcription of sung melodies has very different applications in serious games, museums, education and many creative purposes [3, 8, 14]. There is a lot of literature on melody transcription [15, 19] and voice transformation methods [11, 12, 24]. These two concepts can be considered jointly to allow the implementation of a low level

✉ Ana M. Barbancho
abp@ic.uma.es

[1] ATIC Research Group, Andalucía Tech, E.T.S.I. Telecomunicación, Dpto. Ingeniería de Comunicaciones, Universidad de Málaga, Campus Universitario de Teatinos s/n, 29071, Málaga, Spain

voice-to-tuba transformation method, such that a vocal melody will be turned into a tuba melody conserving some voice features.

Thus, in this paper, we propose an interactive method for pitch-based voice-to-tuba transformation that can be used for entertainment and music learning applications (ie. to learn the timbre of a certain instrument with the student's own melodies). The system obtains the fundamental frequency and other specific parameters of an input audio signal using the Yin algorithm [2], then, after some ad-hoc post-processing stages the voice-to-instrument transformation is performed.

This process, voice-to-instrument conversion, is performed by following two different approaches. On the one hand, a new signal is synthesized from the F_0 curve by making use of an instrument harmonic amplitude database. The harmonic relative amplitudes of musical instrument depend on the signal pitch, since different ranges of F_0 mean different spectral harmonic distributions for a certain instrument. Wind instruments, like a tuba or a bassoon, have been considered because of their resemblance with human voice. Specifically, we will focus on tuba.

On the other hand, in the second conversion approach devised, the pitch information obtained is processed by a note segmentation and transcription scheme, followed by a transition correction block. This latter stage is aimed at eliminating undesired pitch transitions to obtain a better musical result.

In order to compare the quality of both methods against a different approach based on a state-of-the-art technique, a third transformation scheme has been implemented. This method is a timbre morphing technique [1] based on mixing properties of two sounds, which in this case, are a singing voice and a tuba note.

The paper is organized according to the diagram shown in Fig. 1. In Section 2, the methodology used in this project is explained. Section 2.1 presents the feature extraction stage. This step is based on pitch detection by making use of the Yin algorithm [2] and the later filtering of both the F_0 curve and the aperiodicity. In Section 2.2, the transformation of the signal by means of the methods presented above is described. Section 2.2.1 describes the conversion method based on the harmonic amplitudes, in Section 2.2.2 the MIDI transcription process is detailed. The results, the evaluation methodology and a comparison between both methods and the technique based on timbre morphing are described in Section 3. Finally, some conclusions are drawn in Section 4.

2 Methodology

In this section, we expose the methodology used for the entire conversion schemes which is accomplished by different steps depending on the transformation method. Some steps are common to both methods, and others are specific to one of them.

Roughly, the procedure to convert singing voices into a musical sample can be divided into two steps: (1) feature extraction and (2) voice to instrument transformation. These steps are illustrated in Fig. 2.

2.1 Feature extraction

In order to perform the desired transformation some parameters must be extracted from the singing voice. These parameters and the procedures used to perform the adaptation of the parameters obtained by the Yin algorithm to our specific task are described in the next two sections.

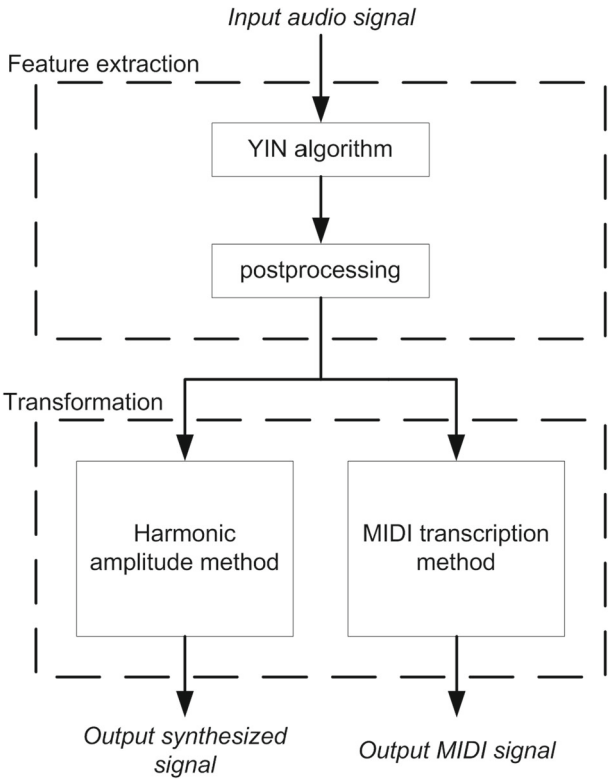

Fig. 1 Scheme of the proposed conversion methods

2.1.1 Parameter extraction: YIN algorithm

To create a musical excerpt that sounds similarly to the sung song, some information of the pitch and the rhythm of the later are required. In this sense, it has to be observed that the F_0 curve actually contains most of the information needed.

So, the first step is the estimation of the F_0 curve. This step is accomplished by using the YIN algorithm [2] which has shown good performance in this task in many music transcription system [17, 26]. This algorithm works on the idea of the autocorrelation method and defines a number of steps for improved performance. Specifically, the authors define the *cumulative mean normalized difference function* $d'_t[\tau]$, which is given by:

$$d'_t[\tau] = \begin{cases} 1 & \tau = 0 \\ \frac{d_t[\tau]}{\frac{1}{\tau}\sum_{j=1}^{\tau} d_t[j]} & \text{otherwise} \end{cases} \tag{1}$$

where τ is an integer lag expressed in samples: $\tau \in [0, W)$, with W the window size in samples and $d_t[\tau]$ the squared difference function:

$$d_t[\tau] = \sum_{j=n}^{n+W} (x[j] - x[j+\tau])^2 \tag{2}$$

where $x[\tau]$ is the amplitude of the input signal x at τ.

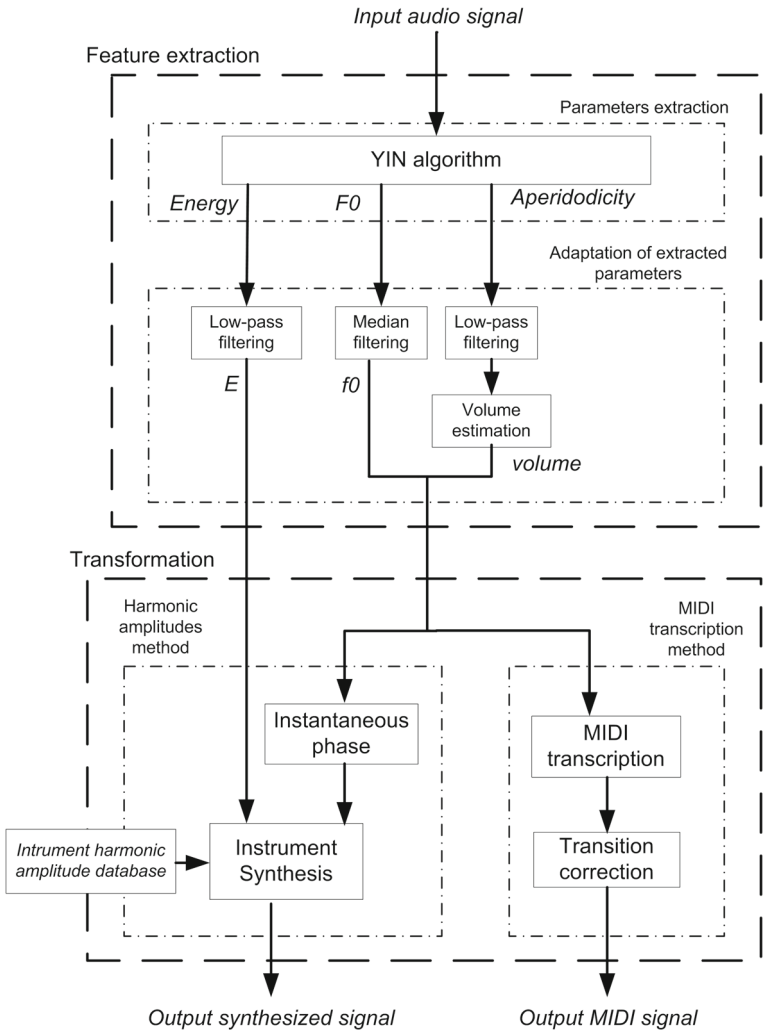

Fig. 2 Detailed scheme of the proposed conversion methods

The Yin algorithm finds the local minimum in $d'_t[\tau]$ with the smallest lag (τ') and then performs a parabolic interpolation in order to find more accurately the location (τ_p) of that minimum that leads to the fundamental frequency using: $f_0 = f_s/\tau_p$, where f_s is the sampling frequency. The minimum $d'_t[\tau']$ must be under a predefined threshold which, as the authors of the algorithm recommend, we set to 0.1. The threshold can be interpreted as the maximum proportion of aperiodic power in a periodic signal tolerated. In order to understand this idea, consider the following identity (see [2] for details):

$$2(x^2[n] + x^2[n+T]) = (x[n] + x[n+T])^2 + (x[n] - x[n+T])^2 \tag{3}$$

Taking the average over a window W and dividing by 4, the following expression appears:

$$\frac{1}{2W}\sum_{j=n+1}^{n+W}(x^2[j]+x^2[j+T]) = \frac{1}{4W}\sum_{j=n+1}^{n+W}(x[j]+x[j+T])^2+\frac{1}{4W}\sum_{j=n+1}^{n+W}(x[j]-x[j+T])^2 \quad (4)$$

The left-hand side of (4) approximates the power of the signal, and the two right-hand side terms are a partition of this power. Observe that the second one is zero if the signal is periodic with period T, which suggests it interpretation as the "aperiodic power" component of the signal. Now, note that it $\tau = T$, the numerator of (1) ($d_t[\tau]$) is proportional to the aperiodic power, and its denominator (average of $d[\tau]$) is approximately twice the signal power [2]. Thus, the aperiodicity measure $ap = d'_t[\tau_p]$, is approximately proportional to the aperiodic/total power ratio.

Regions with harmonic structure show low aperiodicity (see [2]). Conversely, unvoiced frames usually present high aperiodicity. This fact is illustrated in Fig. 3, which shows the curves obtained from the waveform of a woman singing the Happy Birthday song.

Figure 3 shows the audio waveform (a) of the four phrases in the song and the aperiodicity (b). Voiced frames usually present low aperiodicity and stable F_0. As it can be observed, aperiodicity has a significant value at frames between the phrases of the song, when the singer is silent or is just breathing.

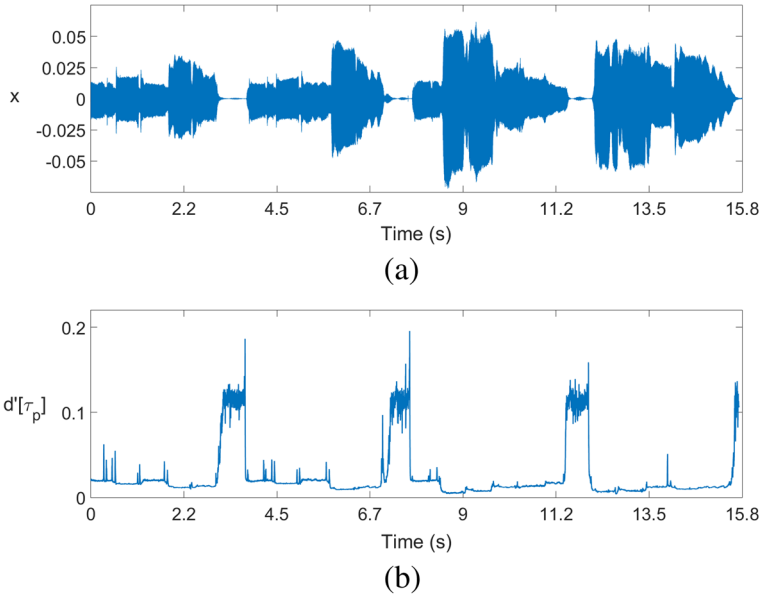

Fig. 3 Signal amplitude and aperiodicity curve in the Happy Birthday song **a** Amplitude of the original signal x. **b** Aperiodicity of the same signal, which shows significant values at frames between the phrases of the song, when the voice is hardly audible

Table 1 Yin algorithm parameters and values employed

Parameter	Value
Frame length	1470 samples
Hop size	735 samples
Integration window length	1470 samples
F0 min	80 Hz
F0 max	900 Hz
Resolution	16 bits
Sampling rate	44100 Hz

Thus, this algorithm can be used for both extracting the F_0 curve and identifying voiced/unvoiced frames. Also, the aperiodicity parameter will be used later for shaping the volume of the generated audio signal. For this same purpose, the *energy* parameter is calculated at this step as well. In the case of the harmonic amplitude method, the aperiodicity alone is not enough to achieve a natural release of singing phrases, so we also used the instantaneous *energy* of the signal estimated by using a moving window to improve the performance. In the window, the energy (E) is obtained as:

$$E = \sum_{n=0}^{N} |x[n]|^2 \tag{5}$$

with $N = 1470$ samples.

In order to avoid ambigüities in later descriptions the parameters of the Yin algorithm as used in this work are shown in Table 1. For human voice, it is reasonable to set the upper and lower limits for the pitch (F_0) to 900 Hz and 80 Hz, respectively. In the frames in which the aperiodicity is over the threshold or the detected pitch is out of the bounds, the F_0 assigned is 0 Hz.

After the parameters are obtained, the data curves have to be adapted to be usable in our task. Specifically, F_0 and aperiodicity should present good stability to avoid undesired unnatural tuba pitch changes and interruptions of the sound after the music transformation or generation is performed. To this end, a median filter that will be described in the next section will be used to smooth the F_0 curve to avoid undesired pitch changes. This step is specially important in the case of the posterior utilization of the MIDI transcription method. Regarding the aperiodicity, a low-pass filter will be used to achieve the desired volume control goal; the energy will be filtered likewise.

2.1.2 Adaptation of extracted parameters

The F_0 curve of a human voice has unavoidable fluctuations due to micro-tonal pitch changes. In order to minimize the effect of this natural phenomenon in the musical excerpt generated, regardless the transformation method, the system devised applies a median filter to this curve. Median filtering has been successfully used previously for singing transcription [5, 9] and speech processing schemes [21]. Since a typical F_0 curve presents noticeable discontinuities a linear approach, such as low-pass filtering, does not work properly for our purpose.

Figure 4 shows an example of the application of median filtering to a pitch curve obtained by the Yin algorithm with the parameters indicated in Section 2.1.1. The F_0 curve obtained directly from the utilization of the Yin algorithm (grey) shows clear oscillations due to a

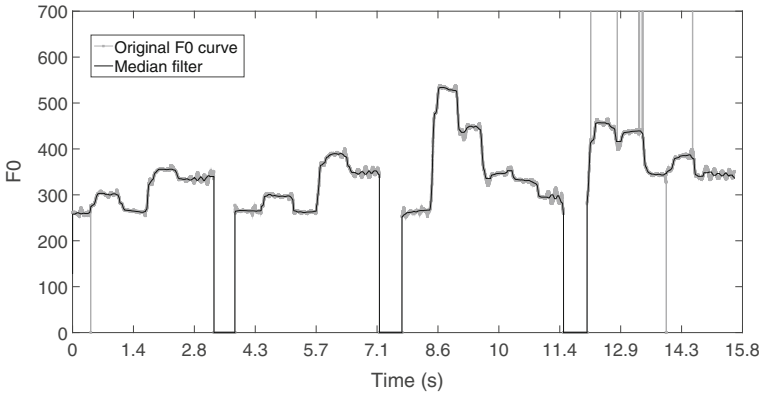

Fig. 4 Example of the application of median filtering to a pitch curve obtained by the Yin algorithm. The filtering process smooths oscillations and removes spurious gaps

natural vibrato of the voice. Some frames of the last phrase sung (between 11.4 and 15.8 sec.) present several spurious values because of errors of the detection of the Yin algorithm. In the filtered F_0 curve (black), the gaps have been removed completely and, additionally, the oscillations have been smoothed. After some tests, we decided to apply a 300-point median filter to get the best subjective results.

The usage of aperiodicity is aimed at helping to modulate the volume of the output signal. So, we define a *volume* parameter as a normalized function of the aperiodicity:

$$volume(t) = \begin{cases} 0 & ap(t) > 0.1 \\ 1 - 10 \cdot ap(t) & \text{otherwise} \end{cases} \tag{6}$$

where $ap(t)$ is the instantaneous aperiodicity of the signal calculated in a frame-by-frame basis, with $ap(t)$ between 0 and 0.1. Recall that $ap(t) > 0.1$ means that the aperiodicity in that frame is too large, so its F_0 detected is set to zero and consequently no transformed signal will be produced for that frame.

Note that due to the intended usage of $volume(t)$, it is not considered necessary to apply a high order filter control its variations of the $volume$. Small variations of the volume will give naturalness to the output sound. Thus, in order to filter out some undesired noise in the estimated aperiodicity curve (shown in Fig. 3) we apply a linear smoother implemented as a low-pass filter. In this case, a 4-order Butterworth filter attains good performance.

In Fig. 5, we show the F_0 and aperiodicity curves after filtering. These data, as shown in this figure, will be used in next blocks.

As previously indicated, the volume alone is not enough to provide with a good modulation of the signal if the harmonic amplitude transformation method is used. The $energy$ parameter is employed to overcome this issue, which is also filtered by a 4th order Butterworth low-pass filter.

Figure 6 shows the relationship between the *energy* and the *volume* parameter of a signal, which are very different from each other. The main reason for using the energy curve is to achieve the natural release of singing phrases, thank to its smoothness at the end and at the beginning of the phrases. The volume curve modulates the intensity of the notes. The

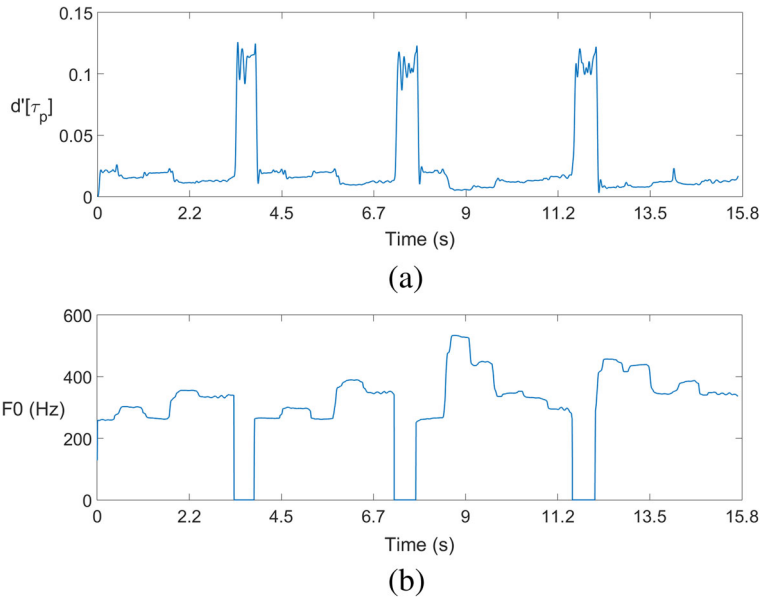

Fig. 5 **a** Aperiodicity $ap = d'_t[\tau_p]$ and **b** filtered F_0 curve of Happy Birthday song performed by a female singer. Observe that in the filtered F_0 curve, gap have been removed and oscillations have been greatly smoothed

simultaneous usage of both *volume* and *energy* parameters allows to achieve natural variations of the intensity and to remove abrupt sound ending in the harmonic amplitude method.

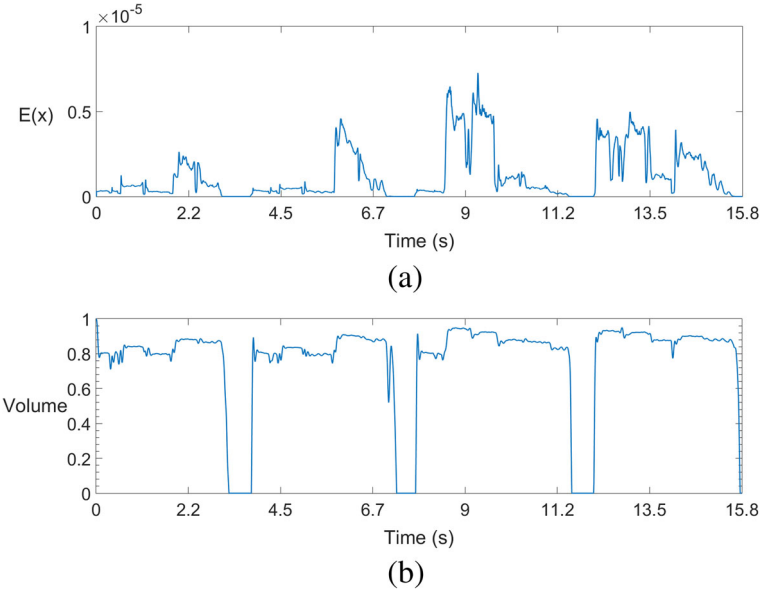

Fig. 6 Energy and volume curves in the Happy Birthday song performed by a female singer. **a** Energy curve smooths the beginning and the end of the phrases. **b** Volume curve modulates the intensity of the notes

2.2 Transformation methods

The goal of this step is to generate a new signal that resembles the original sung melody as closely as possible but sounds as played by a certain instrument by making use of the parameters described in the previous section.

As shown in Fig. 1, we have considered two different approaches to perform this transformation: the harmonic amplitude method and the MIDI transcription scheme (see also Fig. 2).

2.2.1 Harmonic amplitude method

The additive synthesis is a well known sound synthesis technique [16], that consists on the superposition of sinusoidal components whose frequency and amplitude varying with time producing a dynamic spectrum. Specifically, the synthesis of a given musical instrument sound is performed by using a set of sound oscillators with amplitude and frequency controlled by the information obtained by a previous analysis of a real instrument sound.

This approach constitutes a simple additive synthesis model based on [7] for generating a dynamic spectrum. This synthesis model is not suitable to reproduce many special features of wind instruments, but it does generate a wide range of timbres and allows tracking the temporal evolution of the amplitude and frequency of harmonics.

It must be observed that the spectrum of a sound changes noticeably with the fundamental frequency [18]. The amplitude and frequency of each harmonic play an essential role in timbre perception, and therefore they must be properly considered in order to get a synthesized signal as realistic as possible.

The synthesis model selected is publicly available as a toolbox library in *Matlab/GNU* based on *Csound* code in [6]. The toolbox contains a very complete implementation, but we only make use of the database containing information of the relative amplitudes of the harmonics of the selected wind instruments. Note that this spectral harmonic distribution depends on both the specific instrument and the pitch.

The concrete distribution and ranges of F_0 depend on the instrument. There are ten wind instruments available: horn, clarinet, oboe, bassoon, flute, piccolo, sax, trumpet, tuba and trombone. Synthesis is performed by using a deterministic model [23] based on additive synthesis.

This model is ultimately defined by the following expression, where the output signal $y(t)$ is:

$$y(t) = \sum_{k=1}^{K} r_k \cos(2\pi k F_0 t + \phi_k) \tag{7}$$

where K is the number of harmonics, r_k is the amplitude of harmonic k, F_0 is the fundamental frequency and ϕ_k is the phase offset of harmonic k.

2.2.2 MIDI transcription method

At the present stage, melody transcription is performed at low-level (extracting voice parameters directly) and higher structural levels [10] (related to note segmentation or rhythm and harmony organization).

Thanks to the filtering stages performed earlier, the F_0 curve presents high stability within each note, which makes it appropriate to obtain its MIDI representation. However, portamentos are still a problem for a good MIDI conversion. The process to overcome this

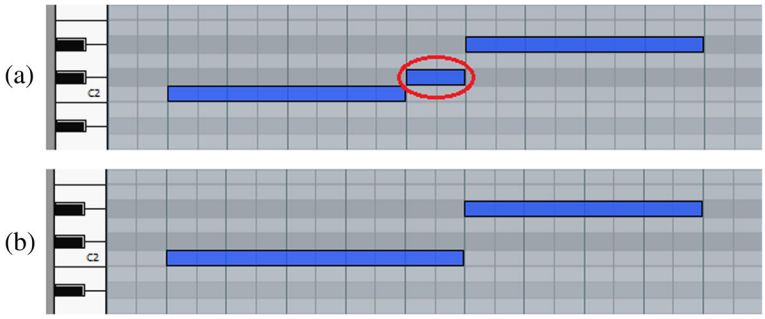

Fig. 7 **a** The duration of the second note is under 130 ms. **b** After the transition correction process, the duration of the note is added to the previous one

issue consists on the removal of the notes with duration under 130 ms, adding the duration of these notes to the previous note. Thus, only long portamentos (>130 ms) will be interpreted as actual notes and the rhythmic structure obtained after the feature extraction stage will be mostly maintained. This process is illustrated in Fig. 7, which shows a conventional MIDI representation. Each bar is a note, with length and height representing the length and the pitch of the note, respectively. The length of the second note of the first scheme (a) is under 130 ms, thereby its duration is added to the previous note, giving the second scheme (b).

Notes must properly defined to generate a correct MIDI stream. In our approach, each note is assigned the values of *pitch, onset/offset time* and *velocity*.

Pitch assignment is done by simply rounding the filtered F_0 curve to the closest MIDI note, assuming A4 - 440 Hz standard tuning. Pitch transients and portamentos under 130 ms are not considered actual notes as described before. Onset and offset times are placed according to note changes and the detected voiced/unvoiced regions. The velocity of a note represents its loudness in MIDI notation being a number between 0 and 127. In our case, the previously estimated parameter $volume$ (Section 2.1.1) is used to set the velocity. Since our parameter $volume(t)$ ranges between 0 and 1, the $velocity$, is defined as follows.

$$velocity(t) = \lfloor volume(t) \cdot 127 \rfloor \tag{8}$$

where $\lfloor\rfloor$ means round towards the nearest smaller integer.

Finally, MIDI transcription was performed with the MIDI tool kit for Matlab developed by Ken Schutte [22]. This tool kit allows reading and writing MIDI files by using Matlab matrices. In this tool, each note corresponding to a MIDI *note message* includes onset/offset times, MIDI note number and velocity.

Finally, in order to play the MIDI melody with the tuba sound we used a free orchestral sample library from Sonatina Symphonic Orchestra created by University of Iowa Musical Instrument Samples (MIS) [25].

3 Results and evaluation

We performed the voice-to-tuba transformation process to 38 melodies available in our dataset with the two methods described above. The system performance has been subjectively evaluated by means of a survey open to volunteers in our website. Thus, the participants in the survey are both trained and untrained musicians. We presented 5 out of the

38 original melodies and their transformations to each participant in the survey (we selected a small subset of samples in order to maintain the attention of the participants during the evaluation) and we asked them about different subjective features such as similarity to the real instruments and resemblance to the original melodies. Additionally, we performed and objective evaluation of the transcription method used in our transformation schemes on the database used in these experiments with the standard MIREX note-tracking measures [4].

Recall that our dataset contains 38 melodies sung by adult and child untrained singers, these audio excerpts have been recorded with a sample rate of 44100 Hz and a resolution of 16 bits [13]. The recordings are not clean and some background noise is present. The duration of the excerpts ranges from 15 to 86 s summing a total duration of 1154 s. This music collection can be broken down into three categories, according to the type of singer:

- Children (our own recordings): 14 melodies of traditional children songs (557 s) sung by 8 different children (5–11 years old).
- Adult male: 13 pop melodies (315 s) sung by 8 different adult male untrained singers. These recordings were randomly chosen from the public MTG-QBH dataset [20].
- Adult female: 11 pop melodies (281 s) sung by 5 different adult female untrained singers.These recordings were also randomly chosen from the public MTG-QBH dataset.

First, we will present the results of the objective evaluation of the performance of the F_0 curve extracted and then, the results extracted from the subjective evaluation performed by means of the open survey previously described.

3.1 Objective evaluation of the F_0 curve used as input to the transformation scheme

The evaluation of melody transcription systems commonly consists on the comparison between the transcription and human annotations. In our case, we have used an evaluation framework [13] specifically designed for this purpose, with evaluation measures based on standard MIREX note-tracking measures [4], plus some extra information about the error type.

In our evaluation framework, the definition of correctly transcribed notes is based on the combination of a number of conditions: correct onset, correct offset and correct pitch. Specifically, we have considered three different definitions of correct note as defined in MIREX, which have been described extensively in [13].

- Correct onset, pitch and offset (COnPOff).
- Correct onset and pitch (COnP).
- Correct pitch (COP).

These three parameters are analyzed using three different measures: precision, recall and F-measure [13].

We have included some evaluation measures to identify incorrect notes with one single error. These are:

- Only-Bad-Onset (OBOn).
- Only-Bad-Pitch (OBP).
- Only-Bad-Offset (OBOff).

On the other hand, segmentation errors have been described by using two different definitions:

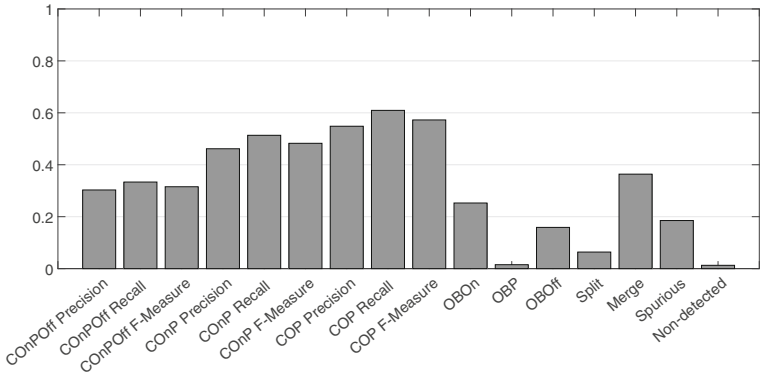

Fig. 8 Results of the evaluation of the measures computed for the system. Range is from 0 to 1. The parameters have been analyzed using three different measures: precision, recall and F-measure [13]

– Split: a note that is incorrectly segmented into different notes.
– Merged: a set of consecutive notes are incorrectly considered to be merged into the same note.

Finally, incorrect notes with voicing errors are described by two other definitions:

– Spurious: an unvoiced sound produced a false transcribed note.
– Non-detected: a sung note is not transcribed at all.

In Fig. 8, we show the results obtained for each evaluation measure computed for our system, in a 0 to 1 range. We consider the pitch detection performance, represented by the bars labeled COP pretty good, with a value close to 0.6. It is followed by the performance of offset and onset times, labelled COn and COff, with values close to 0.5 in the case of correct onset, and around 0.3 in the case of correct onset and offset. However, the merge error (close to 0.4) is the main issue of the method, as often happens in other MIDI transcription systems.

It is important to note that the final perceptual evaluation relies on the performance of the F_0 detection scheme and the two parameters subjectively scored depend on the quality of the estimation of the pitch and rhythm of the sung melody and the filtering stages described before.

3.2 Perceptual evaluation

In this section, the results of the subjective evaluation of the audio samples created by the proposed schemes will be drawn. First, we present the particularities of this evaluation. Then, Sections 3.2.1 and 3.2.2 detail the results obtained on harmonic amplitude method and MIDI transcription method, respectively. Finally, in Section 3.2.3 the comparison between both methods is exposed.

Among the ten available wind instruments in the musical instrument data set [6], we chose the tuba at the sight of its observed sound quality. Figure 9 shows the comparison between the harmonics of a real tuba and the synthesized tuba of our system, playing the same note, a C#3. Since its frequency, 141 Hz, is between 121 and 161 Hz, the synthesized tube is generated with 8 harmonics, according to the corresponding column of Table 2.

Table 2 shows the spectral harmonic distribution and the relative amplitudes for 6 different pitch ranges for the tuba. The relation between the relative amplitudes remains constant

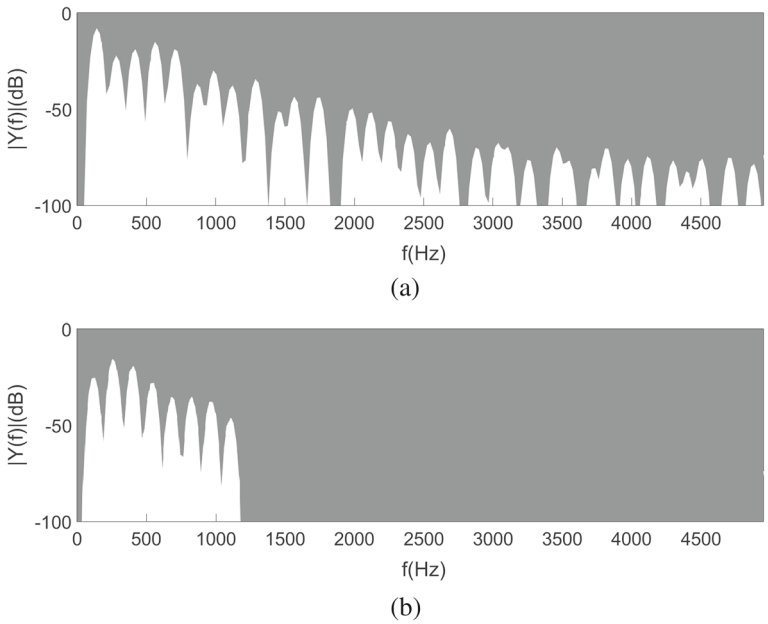

Fig. 9 Real tuba (**a**) and synthesized tuba (**b**) harmonic spectra playing a C#3 note. The real instrument produces a large number of harmonics. On the other hand, according to Table 2, and taking into account that the frequency of the C#3 is between 121 and 161 Hz, the synthesized tuba sound is created with 8 harmonics with their corresponding relative amplitudes

within a pitch range, but they are also modulated by the signal volume. The table reflects the well known fact according to which lower notes are richer in harmonics, being this the reason why the number of harmonics shown in the table decreases with the growing pitch.

A comparison between the harmonics of a real tuba and the MIDI tuba selected in our system is qualitatively shown in Fig. 10. Since the sound of the MIDI instrument is extracted from a sample library generated from many recordings of real instruments [25], the number of harmonics is very similar to the real case. The difference between the relative harmonic amplitudes may be due to the instruments manufacturing and the recording studio.

The data of the subjective evaluation are gathered by means of an open survey[1] in which each participant in the survey had to evaluate two different aspects of the results of the transformation of 5 distinct sung melodies: the similarity of the sound with a real tuba and their resemblance to the original melody.

A total of 21 persons participated in the survey, 19.05 % of them had no musical background while 80.95 % had.

3.2.1 Synthesized tuba

Table 3 shows the results for the survey of the synthesized tuba. The selected melodies were performed by 3 adult female, 1 adult male and a child singer. The participants were

[1] http://www.atic.uma.es/voice2tuba/Voice2Tuba_survey2

Table 2 Relative harmonic amplitudes for the synthesized tuba extracted from the *Csound* toolbox [6]

F0 range (Hz)	0 ∼ 68	68 ∼ 90	90 ∼ 121	121 ∼ 161	161 ∼ 216	>216
Relative harmonics	2.63	3.18	0.68	0.32	1.07	0.44
	1.17	3.82	0.74	0.93	0.53	0.18
	3.12	2.99	0.79	0.59	0.35	0.08
	1.52	2.24	0.39	0.24	0.18	0.02
	2.39	1.99	0.32	0.10	0.05	
	1.54	1.05	0.17	0.09	0.02	
	1.34	0.74	0.10	0.08		
	1.33	0.50	0.10	0.02		
	0.76	0.37	0.07			
	0.73	0.36	0.06			
	0.45	0.28	0.03			
	0.38	0.28				
	0.38	0.21				
	0.39	0.13				
	0.29	0.10				
	0.23	0.06				
	0.17	0.02				
	0.15					
	0.07					
	0.06					
	0.09					
	0.03					

The number of harmonics and their relative amplitudes depend on the pitch

asked to score between 1 and 5 (where 1 means poor and 5 excellent) the quality of the transformation and its resemblance with the original melody.

The majority of the evaluators considered that the resemblance to the original melody was better than the sound quality. Note that the resemblance to the original melody, with this sound generation method, heavily depends on the pitch, onset and offsets obtained after the pitch extraction scheme and filtering stage. On the other hand, we think that the similarity to a tuba sound strongly depends on the model selected for the additive synthesis of the sound.

The result regarding the resemblance to the original melody is very satisfactory, however the result on the similarity to a real tuba melody attained a lower score.

3.2.2 MIDI tuba

In this section, the results found after the same survey described in the previous section with the audio excerpts created using the MIDI tuba approach are shown. According to Table 4, it is clear that the audio melody sounds like played by a real tuba, according to the participants' responses and also, the score about the resemblance the original melody is noticeably good, although the mean is lower than with the previous approach.

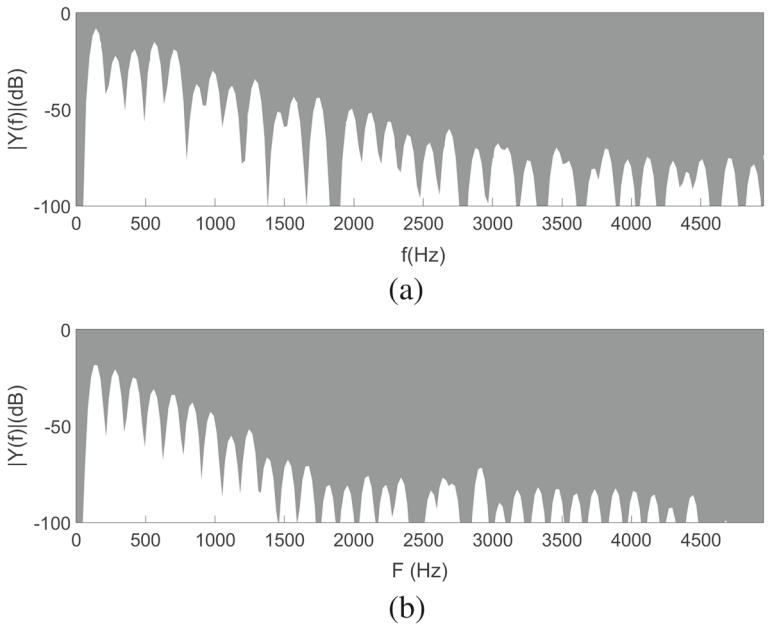

Fig. 10 Real tuba (**a**) and MIDI tuba (**b**) harmonic spectra playing a C#3 note. The real instrument produces a large number of harmonics. Regarding the MIDI tuba, its sound is created from a real instrument of Sonatina Symphonic Orchestra [25], showing a similarly large number of harmonics

3.2.3 Comparison between the transformation approaches

In this section, we perform a comparative analysis of the subjective results attained by the transformation methods proposed.

But before that, we present an additional third transformation method which has been considered and included in the survey in order to compare the quality of our approach with a method derived from a different state-of-the-art scheme. This scheme is timbre morphing technique based on the harmonic plus stochastic model [1], which have proved to attain remarkable performance for voice synthesis. The timbre morphing is a transformation that generates new audio excerpts with hybrid properties extracted from two others elements. The technique relies on the interpolation of the harmonic and the residual components [23] of two sounds, in our particular case, a singing voice and a tuba note.

Table 3 Results of synthesized tuba survey (mean values)

	It sounds like a real tuba	It resembles the original melody
Melody 1	2.52	4.28
Melody 2	2.28	3.57
Melody 3	2.33	3.52
Melody 4	2.47	4.19
Melody 5	2.52	3.9
Mean	2.42	3.89

The mean value of the resemblance is significant higher than the mean sound quality

Table 4 Results of the MIDI tuba survey (mean values)

	It sounds like a real tuba	It resembles the original melody
Melody 1	4.38	4.28
Melody 2	3.85	3
Melody 3	4.23	2.76
Melody 4	4.33	3.85
Melody 5	4.33	4.19
Mean	4.22	3.61

The sound quality is very high since it is extracted from a sample library. However, the resemblance is slightly lower than is the case for the synthesized tuba

With this additional transformation method included in the survey, we asked the participants to re-evaluate the system performance in terms of quality and resemblance with the original melodies. A total of 18 persons participated in the survey, all of them with some musical background. The results are presented in Table 5.

Observing the results of the three methods (Tables 3, 4 and 5), it can be concluded that regarding the similarity of the melodies created to a real tuba melody, the mean score attained by the MIDI approach is significantly higher than the one obtained by the additive synthesis approach, which reflects the difference in goodness between the models used in the two schemes. However, in spite of this fact, the score of the resemblance to the original melody is higher for the case of the melody created by the additive synthesis scheme. This observation seems to suggest that an additive synthesis scheme, with an improved musical instrument model, would be preferred over the MIDI approach for a singing voice transformation application. Both methods obtained better results than the timbre morphing approach, selected for the comparative evaluation, which, on the other hand, obtained reasonable results in terms of resemblance with the original melody, which is probably related to the fact that this is also a synthesis method.

Additionally, we asked the participants in the survey to classify the three methods in order of preference. As shown in Table 6, the order of preference in all cases from best to worst was: the MIDI transformation, the synthesized tuba and the tuba morphing. With a mean value of 87.77 %, the MIDI transformation method was chosen the best approach for all samples. In second place, the synthesized tuba was chosen with a mean value of 68.88 %. And finally, the worst transformation method was the tuba morphing for the 77.77 % of the participants. Detailed data regarding these choices are shown in Table 6.

Table 5 Results of timbre morphing survey (mean values)

	It sounds like a real tuba	It resembles the original melody
Melody 1	1.27	3.72
Melody 2	1.27	2.88
Melody 3	1.5	3.05
Melody 4	1.61	3.72
Melody 5	1.55	3.94
Mean	1.44	3.46

Both parameters (quality and resemblance) are lower than the values obtained using our two proposed methods

Table 6 Classification of preferences between the three methods

	Synthesized tuba	MIDI tuba	Tuba morphing
(a)			
Best case			
Melody 1	11.11 %	88.88 %	0 %
Melody 2	22.22 %	77.77 %	0 %
Melody 3	11.11 %	88.88 %	0 %
Melody 4	5.55 %	88.88 %	5.55 %
Melody 5	0 %	94.44 %	5.55 %
(b)			
Intermediate case			
Melody 1	83.33 %	11.11 %	5.55 %
Melody 2	66.66 %	16.66 %	16.66 %
Melody 3	55.55 %	11.11 %	33.33 %
Melody 4	55.55 %	11.11 %	33.33 %
Melody 5	83.33 %	5.55 %	11.11 %
(c)			
Worst case			
Melody 1	5.55 %	0 %	94.44 %
Melody 2	11.11 %	5.55 %	83.33 %
Melody 3	33.33 %	0 %	66.66 %
Melody 4	38.88 %	0 %	61.11 %
Melody 5	16.66 %	0 %	83.33 %

(a) Best method. With a mean value of 87.77 %, the best method chosen in all cases was the MIDI tuba. (b) Intermediate method. The second best method chosen by the 68.88 % of the participants was the synthesized tuba. (c) Worst method. The 77.77 % of the participants chose the tuba morphing as the worst method

4 Conclusions

In this paper we have presented a voice-to-tuba transformation system with applications in museums, serious games and creative purposes. The system presented makes use of two distinct music transformation methods based on additive synthesis and MIDI transcription.

A subjective evaluation of the system performance has been carried out by means of an open survey. The results show that the MIDI approach led to melodies that sound like actual tubas while the melodies created by the synthesis approach did not obtain such high score regarding this aspect of the evaluation. However, the usage of the synthesis scheme led to melodies with higher resemblance to the original sung melodies than using the MIDI approach, which seems to suggest that the usage of an advanced synthesis model could become the best choice for our task.

Furthermore, a comparison between the two implemented methods and an additional timbre morphing-based scheme has also been done. A similar subjective evaluation has been carried out through a second open survey, which highlighted the better quality and resemblance attained by our two proposed approaches.

Finally, it must not be forgotten that although the system described has proved to be successful to transform into a tuba, it can still be improved by enhancing the sung melody analysis scheme.

Acknowledgments This work has been funded by the Ministerio de Economía y Competitividad of the Spanish Government under Project No. TIN2013-47276-C6-2-R. This work has been done at Universidad de Málaga, Campus de Excelencia Internacional (CEI) Andalucía TECH.

References

1. Bonada J, Serra X, Amatriain X, Loscos A (2011) Spectral processing. DAFX: digital audio effects, 2nd edn, pp 393–445
2. de Cheveigné A, Kawahara H (2002) YIN, a fundamental frequency estimator for speech and music. J Acoust Soc Am 111(4):1917–1930
3. Dittmar C, Großmann H, Cano E, Grollmisch S, Lukashevich HM, Abeßer J (2010) Songs2see and globalmusic2one: two applied research projects in music information retrieval at fraunhofer idmt. In: Ystad S, Aramaki M, Kronland-Martinet R, Jensen K (eds) CMMR, Lecture notes in computer science, vol 6684. Springer, Berlin, pp 259–272
4. Downie JS (2013) Mirex contest website. http://www.music-ir.org/mirex/
5. Haus G, Pollastri E (2011) An audio front end for query-by-humming systems. In: 2nd annual International Society for Music Information Retrieval conference (ISMIR2001), pp 65–72
6. Horner A (2002) Cooking with Csound. Part 1, Woodwind and brass recipes. A-R Editions, Middleton
7. Horner A, Ayers L (1998) Audio in the new millennium. J Audio Eng Soc 46(10):868–879
8. Howard DM, Welch G, Brereton J, Himonides E, Decosta M, Williams J, Howard A (2004) WinSingad: a real-time display for the singing studio. Logoped Phoniatr Vocol 29(3):135–144. doi:10.1080/14015430410000728
9. Krige W, Herbst T, Niesler T (2008) Explicit transition modelling for automatic singing transcription. J New Music Res 37(4):311–324
10. Lesaffre M, Leman M, De Baets B, Martens J (2004) Methodological considerations concerning manual annotation of musical audio in function of algorithm development. In: Proceedings of the International Society for Music Information Retrieval conference (ISMIR04), pp 64–71
11. Mayor O, Bonada J, Janer J (2009) Kaleivoicecope: voice transformation from interactive installations to video games. In: Proceedings of 35st AES conference: audio for games, pp 1–8
12. Mayor O, Bonada J, Janer J (2011) Audio transformation technologies applied to video games. In: Proceedings of 41st AES conference: audio for games, pp 1–8
13. Molina E, Barbancho I, Barbancho AM, Tardón LJ (2014) Evaluation framework for automatic singing transcription. In: 15th International Society for Music Information Retrieval conference (ISMIR14), pp 567–572
14. Molina E, Tardón LJ, Barbancho I, Barbancho AM (2014) The importance of f0 tracking in query-by-singing-humming. In: 15th International Society for Music Information Retrieval conference (ISMIR14), pp 277–282
15. Molina E, Tardón L, Barbancho A, Barbancho I (2015) Sipth: singing transcription based on hysteresis defined on the pitch-time curve. IEEE/ACM Trans Audio Speech Lang Process 23(2):252–263
16. Moorer J (1977) Signal processing aspects of computer music: a survey. Proc IEEE 65(8):1108–1137
17. Poliner GE, Ellis D, Ehmann A, Gomez E, Streich S, Beesuan O (2007) Melody transcription from music audio: approaches and evaluation. IEEE Trans Audio Speech Lang Process 15(4):1247–1256
18. Risset JC, Wessel D (1999) Exploration of timbre by analysis and synthesis. In: Deutsch D (ed) The psychology of music. Academic, New York, pp 113–169
19. Ryynänen M (2006) Singing transcription. In: Klapuri A, Davy M (eds) Signal processing methods for music transcription. Springer Science + Business Media LLC, Berlin, pp 361–390
20. Salamon J, Serrà J, Gómez E (2013) Tonal representations for music retrieval: from version identification to query-by-humming. Int J Multimed Inf Retr, special issue on Hybrid Music Information Retrieval 2:45–58
21. Schafer RW, Rabiner LRs (1990) Digital representations of speech signals. Kaufmann, San Mateo, pp 49–64
22. Schutte K (2012) Midi toolkit for matlab. http://www.kenschutte.com/midi/
23. Serra X (1997) Musical sound modeling with sinusoids plus noise. Musical signal processing, pp 1–25

24. Stylianou Y (2009) Voice transformation: a survey. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2009), pp 3585–3588
25. University of Iowa: Musical Instrument Samples (MIS): Sonatina Symphonic Orchestra (2014). http://sso.mattiaswestlund.net/
26. Viitaniemi T, Klapuri A, Eronen A (2003) A probabilistic model for the transcription of single-voice melodies. In: Proceedings of Finnish Signal Processing Symposium, pp 5963–5957

José L. Santacruz received his M.Sc. degree in Telecommunication Engineering at the Universidad de Málaga (UMA), Málaga, Spain in 2014. He is a researcher of the Application of Information and Communications Technologies (ATIC) Research Group at the Department of Ingeniería de Comunicaciones of the Universidad de Málaga since 2015. His research is focused on signal processing of audio signals, specifically, singing voice and its applications.

Lorenzo J. Tardón received his degree in Telecommunications Engineering from University of Valladolid, Valladolid, Spain, in 1995 and his Ph.D. degree from Polytechnic University of Madrid, Madrid, Spain, in 1999. In 1999 he worked for ISDEFE on air traffic control systems at Madrid-Barajas Airport and for Lucent Microelectronics on systems management. Since November 1999, he has been with the Department of Communications Engineering, University of Málaga, Málaga, Spain. Lorenzo J. Tardón is currently the head of the Application of Information and Communications Technologies (ATIC) Research Group. He has worked as main researcher of different projects on audio and music analysis. He is a member of several international journal committees on communications and signal processing. In 2011, he has been awarded the 'Premio Málaga de Investigación' by the Academies 'Bellas Artes de San Telmo' and 'Malagueña de Ciencias'. His research interests include serious games, audio signal processing, digital image processing and pattern analysis and recognition.

Isabel Barbancho received her degree in Telecommunications Engineering and her Ph.D. degree from the University of Málaga (UMA), Málaga, Spain, in 1993 and 1998, respectively, and her degree in Piano Teaching from the Málaga Conservatoire of Music in 1994. Since 1994, she has been with the Department of Communications Engineering, UMA, as an Assistant and then Associate Professor. During 2013, she has been a Visiting Scholar at University of Victoria, Victoria, BC, Canada. She has been the main researcher in several research projects on polyphonic transcription, optical music recognition, music information retrieval, and intelligent content management. Her research interests include musical acoustics, signal processing, multimedia applications, audio content analysis, and serious games. Dr. Barbancho received the Severo Ochoa Award in Science and Technology, Ateneo de Málaga-UMA in 2009 and the 'Premio Málaga de Investigación 2011' Award from the Academies 'Bellas Artes de San Telmo' and 'Malagueña de Ciencias'.

Ana M. Barbancho received her degree in Telecommunications Engineering and her Ph.D. degree from University of Málaga, Málaga, Spain, in 2000 and 2006, respectively. In 2001, she also received her degree in Solfeo Teaching from the Málaga Conservatoire of Music. Since 2000, she has been with the Department of Communications Engineering, University of Málaga, as an Assistant and then Associate Professor. Her research interests include musical acoustics, digital signal processing, new educational methods, and mobile communications. Dr. Barbancho was awarded the 'Second National University Prize to the Best Scholar 1999/2000' by the Spanish Ministry of Education in 2000, the 'Extraordinary Ph.D. Thesis Prize' by ETSI Telecomunicación of University of Málaga in 2007 and the 'Premio Málaga de Investigación' by the Academies 'Bellas Artes de San Telmo' and 'Malagueña de Ciencias' in 2010.

Emilio Molina received his degree in Telecommunications Engineering from the University of Málaga (UMA), Málaga, Spain, in 2011. In 2012, he obtained the Professional Degree of Classic Piano from the Conservatori del Liceu, Barcelona, Spain, and his M.Sc. in Sound and Music Computing from the Universitat Pompeu Fabra, Barcelona, Spain, in 2013. He was awarded the Best Final Year Project award from University of Málaga in 2007 and he was nominated as finalist for the Best Final Year Project Award by the Official National Telecommunications Engineering Board in 2013. Currently, he is a Ph.D. candidate at the Application of Information and Communications Technologies (ATIC) Research Group. His main research topic is the automatic analysis and processing of audio signals and applications.