SINGING VOICE RESYNTHESIS USING VOCAL SOUND LIBRARIES

Nuno Fonseca,
CIIC/ESTG,
Polytechnic Institute of Leiria
Leiria, Portugal
nuno.fonseca@ipleiria.pt

Aníbal Ferreira,
FEUP,
Oporto University
Porto, Portugal
ajf@fe.up.pt

ABSTRACT

Although resynthesis may seem a simple analysis/synthesis process, it is a quite complex task, even more when it comes to recreating a singing voice. This paper presents a system whose goal is to start with an original audio stream of someone singing and recreate the same performance (melody, phonetics, dynamics) using an internal vocal sound library (choir or solo voice). By extracting dynamics and pitch information, and looking for phonetic similarities between the original audio frames and the frames of the sound library, a completely new audio stream is created. The obtained audio results, although not perfect (mainly due to the existence of audio artifacts), show that this technological approach may become an extremely powerful audio tool.

1. INTRODUCTION

The concept of resynthesis (recreating something that already exists), and some variations such as transynthesis (recreating an audio stream with a completely different sound) [1] and audio mosaicing (creating an audio stream with existing small audio fragments) [2], has many interesting audio applications.
They can be used as an audio transformation tool, by changing an audio stream into another, allowing a much deeper transformation than traditional audio effects. For instance, transforming one musical instrument into a completely different musical instruments (e.g. violin → trumpet), or by "replacing" the original audio stream with a similar content with better "sonic" quality (e.g. replacing a poor upright piano recording with top-of-a-class grand piano samples).
Also, resynthesis can be seen as a way to improve the usability of today's highly complex virtual instruments. From sampling libraries to virtual synthesizers, the availability of tens or even hundreds of parameters/choices makes the work of musicians much more powerful but also much more challenging and time-intensive. As such, resynthesis can offer the chance to improve the user interface by which the users show how a synthesized line should be played (e.g. a trumpet player that uses his trumpet to show the performance musical characteristics that he/she wants to apply to his saxophone virtual instrument).
When it comes to the singing voice, all the same applications apply, although the associated complexity increases due to a new dimension: text/phonetics.
The transformation of the singing voice can be performed in several ways, such as: "simple" audio effects (EQ, reverb, compressor, etc) [3], vocoders [3], gender/age transformation [4], solo-to-choir transformation [5], etc. Nevertheless, most of these methods are limited to a sub-type of transformation. In a near future, resynthesis could eventually be used as a more general transformation tool, working with many types of transformation at once (e.g. transforming a homemade recording of an amateur female singer into a symphonic male choir at a concert hall). Also, by accessing the information between the analysis and the synthesis stages, one may be given the freedom to change some audio/musical parameters (e.g. correct pitch, etc).
With the appearance of commercial singing voice virtual instruments with text building features, like Vocaloid [6] (based on the work of [7]), "EW/QL Symphonic Choirs" [8] (based on the work of one of the authors [9]), and others, the computer can now be used to create singing lines. Although these tools can create, in many cases, realistic results, it is a very time consuming task with a long learning curve. Although work has been done using voice as a user interface to synthesis/audio mosaicing [10] or even to control singing synthesizers [11], there is still the need to enter manual information (like the phonetics text). Once again, resynthesis could be used to increase not only the productivity of the composer/musician, but also to achieve better and more realistic results, even without experience in this type of tools.
Last, but not the least, resynthesis of the singing voice could also be used to bring back the voice of singers that are deceased or that simple lost their singing abilities with age, by using the existing solo recording archives to create new songs. For instance, using a musical performance of Luciano Pavarotti with the voice of Frank Sinatra, or vice-versa.
This paper presents a singing voice resynthesis method, based on sample libraries and phonetic similarity between audio frames, as a way to transform an original monophonic singing recording into a completely new one, trying to keep the same musical/phonetic performance characteristics, but using internal singing samples with a concatenative approach [12] to create the output audio stream.

2. PROPOSED METHOD

The proposed method for resynthesizing the singing voice consists of four modules and a sound library (figure 1). The work is done off-line (not in real-time), which means that each module works with the entire stream at once, producing information vectors that are passed to the other modules. The system works with frames with the size of 23.21 ms (1024 samples at 44.1 kHz) and a hop size of 11.6 ms (512 samples at 44.1 kHz).

2.1. Dynamics

The "dynamics" module is responsible for extracting the dynamics (loudness) information from the original audio stream. For each frame, a Hann window is applied and its energy is obtained by summing the square of sample values.

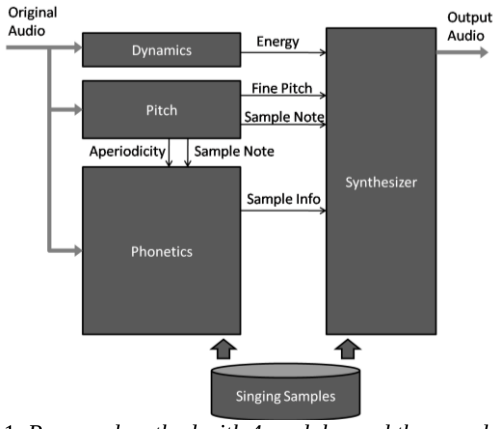

Figure 1. *Proposed method with 4 modules and the sound library*

2.2. Pitch

The "pitch" module (figure 2) is responsible for extracting the pitch information from the original audio stream, not only their course value (musical notes) but also other fine pitch information (like small pitch variations over time) that is responsible for some of the musical performance (e.g. vibrato, portamento).
The module is based on the YIN method [13], which extracts not only the pitch but also an "aperiodicity" parameter (with values between 0 and 1) that can be used later as a voicing measure by the "phonetics" module.

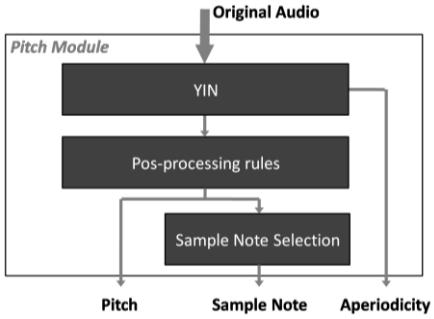

Figure 2. *Pitch Module*

After the application of the YIN method, a small set of post-processing heuristics (rules) and median smoothing are applied to overcome the pitch errors in unvoiced areas.
Although the internal sound library includes samples for each note (semitone interval), by using only the samples that correspond to the obtained pitch will lead to many transitions between samples (e.g., a vibrato note that crosses a note frontier). The goal of the "sample note selection" is to decrease the number of sample transitions (due to pitch changes), by choosing the best sample note to be used on each frame, considering that the synthesis module can change its frequency later within ± 1.5 semitones (a smaller value would increase the number of sample transitions due to pitch changes and a larger value would create significant time/frequency artifacts like formant shifts, etc.).

2.3. Phonetics

The "phonetics" module (figure 3) is responsible for deciding which frames/samples, from the internal sound library, will be used during synthesis.

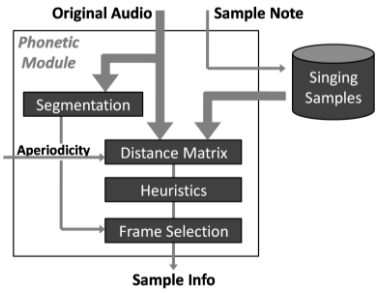

Figure 3. *Phonetic Module*

Instead of using a phonetic recognizer approach, the system uses a frame similarity method that tries to find the frames that are "phonetically similar" (or close) to the ones in the original audio frames. To accomplish that, a distance matrix is calculated with the "phonetic" distance between the original audio frames and the sound library frames.
The "phonetic" distance (eq. 1) between the original frame i and the sample library frame j is calculated based on the Euclidian distance in 4 dimensions/domains: MFCC, LPC frequency response, LPC distance and voicing.

$$D_{i,j}=\sqrt{D_{MFCC}(i,j)^2+D_{LPC\ resp}(i,j)^2+D_{LPC\ dist}(i,j)^2+D_{Ap}(i,j)^2} \quad (1)$$

The MFCC distance (eq. 2) considers a vector difference ranging from coefficient C1 till coefficient C12, disregarding the C0/energy coefficient. All dimensions are normalized within [0,1] (by dividing with a norm_x value, which corresponds to the maximum difference value within each domain).

$$D_{MFCC}\ (i,j) = \left[\frac{\sum_1^{12}\left(c_n^i - c_n^j\right)^2}{\text{norm}_1}\right] \quad (2)$$

The LPC frequency response distance (eq. 3) is based on the difference of the logarithmic frequency response of LPC (resampled at 10kHz with 12 coefficients,) using 128 bins.

$$D_{LPC\ resp}\ (i,j) = \left[\frac{\sum_1^{128}\left(X_n^i - X_n^j\right)^2}{\text{norm}_2}\right] \quad (3)$$

The LPC distance (eq. 4) is based on a symmetrical version the Itakura-Saito distance between LPC coefficients.

$$D_{LPC\ dist}\ (i,j) = \left[\frac{D_{IS}(LPC(i),LPC(j))+D_{IS}(LPC(j),LPC(i))}{2*\text{norm}_3}\right] \quad (4)$$

The voicing distance uses a binary approach (eq. 5), returning 0 unless Ap_i is below 0.2 (voicing area) and if the voicing relation (based on the YIN aperiodicity value) between frames are equal or above 2 regarding each other.

$$D_{Ap}(i,j) = \begin{cases} 1,\ Ap_i < 0.2 \wedge \dfrac{\max(Ap_i, Ap_j)}{\min(Ap_i, Ap_j)} \geq 2 \\ 0,\ otherwise \end{cases} \quad (5)$$

Although we could simply choose, for each original audio frame, the frame from the internal sound library with the lowest distance, that would lead to too much sample transitions during synthesis, so it is fundamental to create mechanisms to reduce the

number of transitions. As such, three features were implemented: segmentation, heuristics and concatenation cost.
Segmentation is responsible for grouping original audio frames into segments. A segment with n frames must be replaced with n consecutive frames from an internal audio sample file, i.e., no sample transitions should occur within the segment. Our segmentation method uses the difference between LPC coefficients of consecutive frames. The obtained local peaks will represent segmentation points – frames whose LPC coefficients don't change significantly probably represent the same phoneme. Although this segmentation method produces many more segmentation points than the number of phonemes present on the original audio, it is better to have false transition points than having undetected transition points.
The consecutive n sample frames that will be used on each segment are the ones that present the lower accumulated distance square errors.
The heuristics phase also helps reducing the number of transitions, by applying some rules with additional searches. For instance, if two consecutive segments use the same sample file, then the segments are merged, and a new search for the new segment is made, looking for the best match.
Although finding the best match for each segment is important (considering the "phonetic" similarity between the original segment and the one to be created), that may create very abrupt transitions between segments, originating audio artifacts. To help prevent this kind of issues, the concept of concatenation cost is added. This concept, that is used on concatenative speech synthesis [14], specifies that the best match for a segment is the one that considers the difference between the original (target) segment and the created one – target cost – and also the difference between the transition points of the segment – concatenation cost.
In this last phase, although there is already a potential frame sequence result that corresponds to the one with the lowest target cost (from the segments point of view), additional heuristics will try to decrease the overall cost by decreasing the concatenation cost. So, searches are performed taking these new cost estimation parameters, by replacing some segment sequences with others or by merging segments.

2.4. Synthesis

With the information about the dynamics, pitch and which frames of the internal sound library should be used on each output frame, the synthesis module (figure 4) is responsible to merge all things together, generating a new audio stream, totally created with the internal vocal samples, and trying to maintain the same phonetic and musical performance of the original voice.
To create the output audio stream, each selected frame may have its pitch changed, based on the difference between the frame pitch and the chosen sample note.
The next step is a phase alignment process, trying to improve the phase alignment between the previous output frame and the current one (that overlaps in 512 samples). A shift between 0 and 512 samples on the new frame is tested (applying a Hann window to each situation) and the value that produces the best correlation on the overlap area is applied as an offset to the new frame.
The final step is to calculate the frame energy and change its gain to match the energy value of the original audio frame, keeping the dynamics behavior of the original audio.

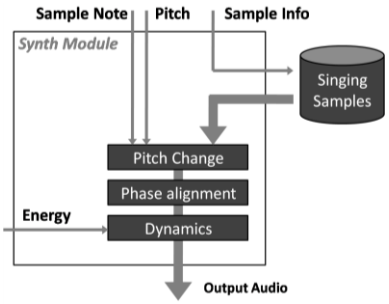

Figure 4. *Synth Module*

3. IMPLEMENTATION AND TESTS

3.1. Implementation

The system was implemented on MATLAB with the exception of the phonetic selection module that was implemented on C++ due to its memory/computational requirements.

3.2. Sound Libraries

The system uses two different sets for the singing samples. The first set corresponds to choir samples, from [8], with 5 different ranges/voice types (basses, tenors, altos, sopranos, boys), each file with a single phoneme, up to 1 second of duration. The phonemes were 8 vowels (uh, ee, oo, ih, eh, oh, eu, ah), 14 pitched consonants (b, d, g, j, l, m, n, r, rr, th, v, w, y, z) and 11 non-pitched consonants (ch, f, h, k, p, q, s, sh, t, th, x).
The second set uses a completely different approach. It includes female solo recordings from [15], where each file consists of a word, with durations up to 9 seconds. 46 words were used: Bene, Breathe, Close, Dark, Death, Domine, Dream, Drown, Im, Fall, Fire, Fly, Gaia, Grass, Hasan, Hate, How, In, Len, Love, Luxet, Ly, Mei, Ness, Of, Ooze, Pray, Priest, Row, Ruins, Run, San, Sing, So, Soft, This, True, Uram, Ventius, Ver, Vosh, Fortuna, From, Gravis, Is, Rain, The.

3.3. Test Set

The system was tested with small fragments from the following solo recordings:

- Tom's Diner – Suzanne Vega
- Amazing Grace – LeAnn Rimes
- Frozen – Madonna
- The Rhythm of the Night – Corona
- Bohemian Rhapsody – Fugees
- Relax (take it easy) – Mika

The obtained audio files can be downloaded from http://www.estg.ipleiria.pt/~nuno.fonseca/papers/dafx2010/.

3.4. Analysis of the results

Table 1 shows the behavior of the obtained results considering some subjective evaluation done by the authors. The first column (Lib) indicates the sound library that was used: Solo (female solo voice, words recordings, from [15]) or Choir (Altos choir, phoneme recordings, from [8]). The second column indicates the song abbreviation from the ones in section 3.3. The next 3 columns present the dynamics/pitch/phonetic similarity between the original fragment and the output (scoring from "0" to "+ + +").

The final column presents the undesired presence of audio artifacts, such as noises, clicks, phase artifacts, etc. (scoring from "0" to "- - - ")

Table 1: *Subjective analysis of the results*.

Lib.	Song	Dyn.	Pitch	Phonetic	Artifacts
Solo	Toms	+ + +	+ +	+ +	–
	Grace	+ + +	+ + +	+ +	– – –
	Frozen	+ + +	+ +	+ +	– –
	Night	+ +	+ +	+	– – –
	Bohem	+ + +	+ + +	+	– – –
	Relax	+ + +	+ +	+	– – –
Choir	Toms	+ + +	+ + +	+ +	–
	Grace	+ + +	+ +	+	– – –
	Frozen	+ + +	+ +	+ +	– –
	Night	+ + +	+ + +	+	– – –
	Bohem	+ + +	+ + +	++	– –
	Relax	+ +	+ +	+	– – –

As can be seen/heard, the dynamics and pitch similarity are positive, although with minor errors in some parts. From the phonetic point of view, the text is understandable most of the time. But, by listening to the obtained results, it is clear that the major problem is the existence of severe artifacts: abrupt transitions, clicks, noises, etc.

4. APPLICATION OF THE PROPOSED METHOD

This approach presents important features like:

- It works with solo or choir samples;
- The sound library doesn't need to be annotated or be on a form of a virtual instrument/synthesizer/formal sample library;
- The sound library can be simply a large set of someone solo recordings (e.g. Elvis solo voice recordings);
- It tries to preserve the musical characteristics of the original performance (fine pitch, dynamics) and also the text/phonetic performance;
- It's language free. Since it works on a phonetic level and without phonetic/language training, it is able to work with any language, being limited only by existing sound libraries phonemes (but still being able to find the closest one, if some phoneme doesn't exist).

As such, it can be used as a tool for different applications such as:

- Control of singing voice virtual instruments, without the need to enter music or text/phonetic information.
- Recreation of deceased artists performances (e.g. Elvis Presley, Kurt Cobain), based on existing solo recording material
- Voice effect/transformation, transforming a vocal performance by applying different solo/choir sample libraries (e.g. symphonic choir, opera singer, pop singer).

5. CONCLUSIONS

A resynthesis approach to the singing voice was proposed. The method makes use of vocal sound libraries and concatenative sound synthesis to recreate a new audio stream with the same musical/phonetic performance.

The obtained audio results, although being able to somehow keep the musical/phonetic performance, still present severe audio artifacts (noise, clicks, etc), due to same lack of continuity on the output audio stream, which needs to be addressed in the future. Nevertheless, the proof is made of a concept whose potential may evolve to a powerful tool in the near future.

6. ACKNOWLEDGMENTS

The authors would like to acknowledge the Portuguese Science and Technology Foundation for partial financial support (Grant SFRH/BD/30300/2006 and Project ID PTDC/SAU-BEB/104995/2008).

7. REFERENCES

[1] W. Chang, Y. Siao, A. Su, "Analysis and Transynthesis of Solo Erhu Recordings using Addtivie/Subtractive Synthesis", in *Proc. of 120th AES Convention*; Paris, France, May 2006.

[2] A. Zils, F. Pachet, "Musical Mosaicing", COST-G6 Workshop on Digital Audio Effects, (DAFx-01), Limerick, 2001.

[3] U. Zolzer, *DAFX – Digital Audio Effects*, John Wiley & Sons, Ltd, 2002.

[4] O. Mayor, J. Bonada, J. Janer, "KaleiVoiceCope: Voice Transformation from Interactive Installations to Video-Games", in *Proc. of AES 35th International Conference: Audio for Games*, 2009.

[5] J. Bonada, "Voice Solo to Unison Choir Transformation", in *Proc. of 118th AES Convention*; Barcelona, Spain, May 2005.

[6] Yamaha, "Vocaloid", information available at http://www.vocaloid.com/index.en.html, Accessed in February 2010.

[7] J. Bonada, X. Serra, "Synthesis of the Singing Voice by Performance Sampling and Spectral Models", IEEE Signal Processing Magazine, 24, pp. 67-79, 2007.

[8] EASTWEST, "EASTWEST/Quantum Leap Symphonic Choirs", information available at http://www.soundsonline.com/Symphonic-Choirs-Virtual-Instrument-PLAY-Edition-pr-EW-182.html, Accessed in February 2010.

[9] N. Fonseca, "VOTA Utility: Making the computer sing", in *Proc. of 114th AES Convention*; Amsterdam, Netherlands, March 2003.

[10] J. Janer, M. de Boer, "Extending voice-driven synthesis to audio mosaicing", in *Proc. of 5th Sound and Music Computing Conference*, 2008.

[11] T. Nakano, M. Goto, "VocaListener: a singing-to-singing synthesis system based on iterative parameter estimation", SMC 2009, Porto, Portugal, July 2009.

[12] D. Schwarz, "Current Research In Concatenative Sound Synthesis", International Computer Music Conference (ICMC 2005), Barcelona, Spain, September 5-9, 2005.

[13] A. Cheveigné, H. Kawahara, "YIN, a fundamental frequency estimator for speech and music", The Journal of the Acoustical Society of America, Vol. 111, No. 4. (2002), pp. 1917-1930.

[14] A. J. Hunt, A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database", ICASSP96, volume 1, 7-10 May 1996, pp. 373 – 376.

[15] EASTWEST, "EASTWEST/Quantum Leap Voices of Passion", information available at http://www.soundsonline.com/Quantum-Leap-Voices-Of-Passion-Virtual-Instrument-pr-EW-174.html, Accessed in Feb. 2010.