Audio Engineering Society
Convention Paper

Presented at the 121st Convention
2006 October 5–8 San Francisco, CA, USA

The Singing Tutor: Expression Categorization and Segmentation of the Singing Voice

Oscar Mayor[1], Jordi Bonada[1], and Alex Loscos[1]

1 Music Technology Group, Institut Universitari de l'Audiovisual, Universitat Pompeu Fabra.
Ocata 1, 08003 Barcelona, Spain
http://www.iua.upf.edu/mtg

{omayor,jbonada,aloscos}@iua.upf.edu

ABSTRACT

Computer evaluation of singing interpretation has traditionally been based exclusively on tuning and tempo. This article presents a tool for the automatic evaluation of singing voice performances that regards on tuning and tempo but also on the expression of the voice. For such purpose, the system performs analysis at note and intra-note levels. Note level analysis outputs traditional note pitch, note onset and note duration information while Intra-note level analysis is in charge of the location and the expression categorization of note's attacks, sustains, transitions, releases and vibratos. Segmentation is done using an algorithm based on untrained HMMs with probabilistic models built out of a set of heuristic rules. A graphical tool for the evaluation and fine-tuning of the system will be presented. The interface gives feedback about analysis descriptors and rule probabilities.

1. INTRODUCTION

Singing voice is considered to be the most expressive musical instrument. Singing and expressing emotions are strongly coupled making clearly distinguishable when a singer performs sad, happy, tender, or aggressive [1,2].

Singing voice automatic scoring has become quite popular in the past few years being *Singstar* [3], *Ultrastar* [4] or *Karaoke Revolution* [5] an evidence of such popularity. However, the algorithms applied in these videogame applications are rude and far too distant from current voice analysis research.

Many people have been working in the field of performance analysis of the singing voice including solo or polyphonic singing voice transcription [6,7], score alignment [8,9] and expressivity [10] but there is a lack in references about automatic expressive detection or

transcription and expression categorization of singing performances as it is focused in this article.

We present here a tool for the automatic evaluation of a singing voice performance with precise note segmentation and expression detection. The application includes a friendly graphical interface for visualization of pitch, vibratos, portamentos, scoops, attacks, sustains, releases…

2. SINGING TUTOR OVERVIEW

Before analyzing emotions in the singing voice first we have to go to a lower level and analyze and categorize different expressive aspects or expressive executions that the singing implies. In most cases this expression can be categorized and defined in the same way for all musical instruments but there are some cases where the singing voice must be separated from the rest because of its timbre characteristics and also for the enormous control that a singer has over the pitch and dynamics of the voice [10].

In order to analyze and classify the expression characteristics in singing performances we need to label different expressive resources and extract the features that better describe a performance and that better distinguish one performance from another. Firstly we decide which features are more relevant in the singing voice and then we will try to derive a set of heuristic rules, based on the analysis descriptors (pitch, energy, spectral coefficients, mel coefficients and its derivatives) that can uniquely identify each expression. With this set of rules we have developed an hypothetic probabilistic model and using hidden markov models based on this model, not in a training process [8], we automatically segment the performance into notes and expression regions.

The segmentation process includes first a note segmentation which consists in aligning the singing performance to a reference midi and then an expression segmentation which is basically an expression transcription of the performance, segmenting each note in sub-regions (attack, release, sustain, vibrato or transition) and assigning an expressive label to each region. In figure 1 you can see an overview diagram of the analysis, note segmentation and expression transcription processes.

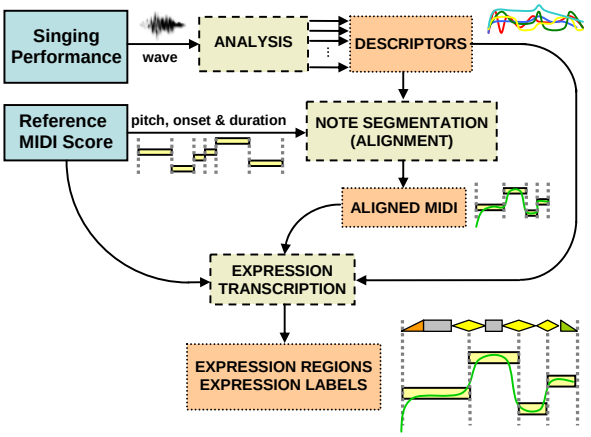

Figure 1 Overview of the analysis and note segmentation/expression transcription process.

A software tool has also been developed which can be used for manually segmenting and labeling notes and regions inside notes (attack, sustain, vibrato, release, transition). These inter-regions will not only be labeled with their type but also with an expressive label such as normal, scoop up/down, fall-down, portamento up/down and other expressive labels. This tool gives feedback showing the probability of each heuristic rule in the segmentation given by the user, so the user can change the segmentation and see how the rules and the global probability improve or not. With this feedback, we can change the set of heuristic rules and its associated probabilities and automatically segment the performance again to improve the segmentation results.

3. NOTE SEGMENTATION (NOTE ALIGNMENT)

We are performing a note segmentation taking into account that we have the midi melody of what the user is singing, so we are aligning the midi notes to the notes in the singing performance. As a result of the segmentation we will have the same notes of the midi reference but the onset and duration will be adjusted or aligned to the performance of the user and silences between the notes will be maintain, so if the performance adds or drops notes that are not present in the midi reference, the alignment will try to be as many similar as it can be to the reference but without adding or dropping notes.

3.1. NOTE SEGMENTATION CRITERIA AND RULES

When we segment a singer's performance, we have to take into account the pitch, energy and other analysis descriptors of the voice but also the phonetic changes to determine the note changes and onsets. The set of heuristic rules used for the note segmentation are classified as follows:

Timing/Tempo: Begin time and duration should be close to the MIDI note onset and duration respectively. Each note must have a minimum duration.

Pitch: The beginning and the end of the note must have pitch. The average pitch of the note should match or be close to the reference pitch (after octave correction) the same happens with intervals between two notes.

Energy: Notes should have a minimum energy at the beginning and at the end and a minimum average energy. Note shouldn't start with a negative delta energy, beginning of notes will match with an energy raise.

Vibrato: Note should not end in the middle of a vibrato.

Delta timbre: Note should start and end close to a delta timbre maximum. Often a high delta timbre implies a change in phonetics which matches with a syllable change.

Others: Note should have small zero crossing rates (high probability of being voiced) and positive delta excitation slope at the beginning.

3.2. NOTE SEGMENTATION ALGORITHM (ALIGNER)

The algorithm that aligns the singer's performance to the reference MIDI is based on Hidden Markov Models though it does not use a probabilistic model built out of a training process but it uses a hypothetic probabilistic model (common practice in voice to MIDI alignment) [8] based on a set of heuristic rules.

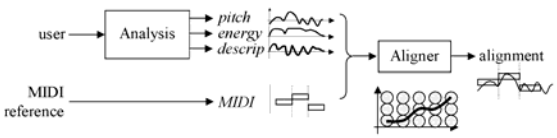

Figure 2 Note segmentation algorithm diagram.

The MIDI reference data consists on silence and note events which are modeled as single state note models. Therefore there is a straightforward mapping between the MIDI data and the note model sequence; each silence links with a silence model and each note with a note model.

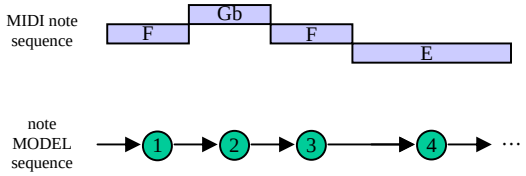

Figure 3 MIDI note and note model sequence.

3.3. NOTE SEGMENTATION VITERBI

The vertical axis of the Viterbi Matrix (figure 4) corresponds to the sequence of note and silence states which represents the melody. Each column of dots in the horizontal axis corresponds to one frame time. The Viterbi Matrix is used to find the most probable path from all possible paths.

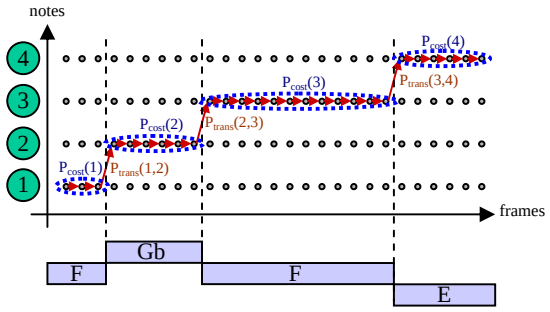

Figure 4 Viterbi Matrix for note segmentation.

The probability associated to each node computes the best path to reach that node, starting from the beginning of the song. We distinguish two types of probability: transition probability (Ptrans) and cost probability (Pcost). The transition probability in our case is always 1, therefore the path probability only depends on the cost probability. In the example shown in figure 4 the probability for the drawn path would be computed as (1):

$$P = P_{cost}(1)\cdot P_{trans}(1,2)\cdot P_{cost}(2)\cdot P_{trans}(2,3)\cdot P_{cost}(3)\cdot P_{trans}(3,4)\cdot P_{cost}(4)$$
$$= P_{cost}(1)\cdot P_{cost}(2)\cdot P_{cost}(3)\cdot P_{cost}(4) \quad (1)$$

The cost probability is computed using heuristic rules which observe the voice descriptors. Given an observation window (a beginning and ending frame) and a given note model, a set of rules are applied to the voice descriptors and each rule computes a rule probability. The cost probability is then the multiplication of all the rule probabilities.

In figure 5 we can see a screenshot of the result obtained after the application of the segmentation/aligner algorithm, on the top we have the wave of the phrase performed by an amateur singer, in the middle window we represent the pitch curve and the reference midi notes one octave below the original pitch and the aligned performance notes drawn over the pitch. In the lower window we draw the energy of the performance. The resulting score is the same as the reference midi but with the notes shortened or lengthened and the onsets shifted to fulfil better the rules applied in the segmentation/alignment algorithm.

Figure 5 Result of the note segmentation screenshot.

4. EXPRESSION CATEGORIZATION (EXPRESSION TRANSCRIPTION)

We have made an study of different executions by professional singers in order to distinguish and classify several common expressive resources used by them. First of all we have decided to divide a note in several regions (Attack/Transition, Sustain, Vibrato and Release/Transition).

4.1. EXPRESSION CATEGORIZATION CRITERIA AND RULES

Not all the expression regions have to be present in a note, for instance we only will have an attack when a silence precedes a note or a release when the note precedes a silence, in other cases there is no attack or release and a transition substitutes them, this transition connects the previous note with the current one or the current note with the next one. Sustain and vibrato regions will only appear when the note has a sustained part (longer notes or when the singer performs a vibrato). Both regions could appear in the same note for instance in a long sustained note where the performer applies vibrato at the end of it.

The minimum number of regions in a note is one, this happens in a fast ascending legato scale of short notes, where all the notes are labelled with just a transition region, with no sustain regions and only the first and last note will have an attack and a release respectively.

We have classified the expression regions as following with several expression labels for each region, each region has some heuristic rules associated used in the expression transcription module to find the best transcription of the performance:

ATTACK RULES

- Will have a minimum duration.
- Last frame must have pitch.
- Should begin with low energy and big delta energy increment and end with small delta energy increment (but not negative).
- Should start with big delta timbre.
- Don't allow vibrato inside an attack region.

ATTACK EXPRESSION LABELS

- Normal: Follows the basic attack rules.
- Up: Begins with higher pitch.
- Scoop-up: Begins some semitones below the pitch and at the end reaches the note pitch.
- Scoop-fry: Begins an octave below the pitch and performs a glissando to reach the pitch.

SUSTAIN RULES

- Will have a minimum duration and allow big durations.
- Should begin and end with pitch.
- Should have a minimum energy and should be stable along the region.
- Allow high delta pitch at the beginning.
- Delta timbre should be low not to allow big phonetic changes which imply a transition.
- Don't allow vibrato inside a sustain region.

SUSTAIN EXPRESSION LABELS

- Normal: Follows the basic sustain rules.
- Fall-down: The pitch should always go down.

VIBRATO RULES

- Will have a minimum duration.
- Should begin and end with pitch.
- Should have a minimum energy.
- Delta timbre should be low not to allow big phonetic changes which imply a transition.
- Should have a minimum margin between highest and lower delta pitch.
- Should have high vibrato descriptor along the region.

VIBRATO EXPRESSION LABELS

- Normal: Follows the basic vibrato rules.

RELEASE RULES

- Will have a minimum duration.
- First frame must have pitch.
- Should begin with small delta energy increment and at the end have low energy and a big delta energy decrement.
- Should start with low delta timbre and end with big delta timbre.
- Don't allow long vibrato, the pitch should be stable.

RELEASE EXPRESSION LABELS

- Normal: Follows the basic release rules.
- Up: The pitch should go up at the end.
- Fall-down: The pitch should go always down.

TRANSITION RULES

- Will have a minimum and maximum duration.
- Last and first frames must have pitch.
- Will have a minimum energy at begin and end.
- Begin and end of the transition should be stable and we don't allow vibrato but penalize when long stable region.
- Should match reference note pitch interval.

TRANSITION EXPRESSION LABELS

- Normal: Follows the basic transition rules between two notes with the same pitch.
- Normal-up: Follows the basic transition rules in an ascending interval between two notes.
- Normal-down: Follows the basic transition rules in a descending interval between two notes.
- Scoop-up: Longer duration, stable at begin and not stable at the end. Reach target pitch after the note onset. Ascending interval.
- Scoop-down: Same as in the scoop-up but descending interval.
- Portamento-up: Same as scoop-up but reach target pitch before note onset.
- Portamento-down: Same as scoop-down but reach target pitch before note onset.

4.2. EXPRESSION TRANSCRIPTION ALGORITHM

In figure 6 all the possible expression paths are shown. These paths are considered by the expression recognition module and the path with highest

probability among all is the one chosen. Once the system knows the MIDI reference notes, then it can build the expression model sequence.

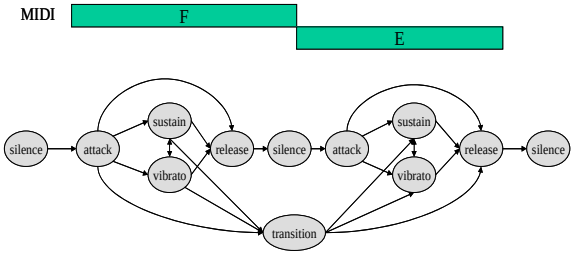

Figure 6 Expressive transcription expression model.

4.3. EXPRESSION TRANSCRIPTION VITERBI

The vertical axis of the Viterbi Matrix (figure 7) corresponds to the sequence of expression models. Each column in the horizontal axis corresponds to one frame time. The Viterbi Matrix is used to find the most probable path from all possible paths.

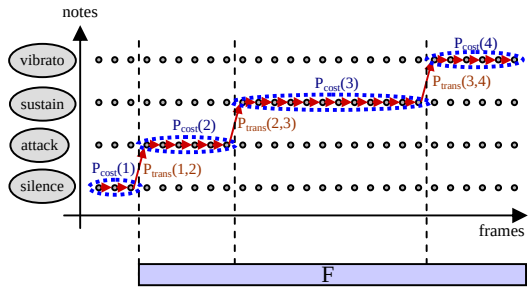

Figure 7 Viterbi matrix for expression transcription.

The probability associated to each node computes the best path to reach that node, starting from the beginning of the song. We distinguish two types of probability: transition probability (Ptrans) and cost probability (Pcost). The transition probability in our case is always 1, therefore the path probability only depends on the cost probability. In the following example the probability for the drawn path would be computes as (2):

$$P = P_{cost}(1)\cdot P_{trans}(1,2)\cdot P_{cost}(2)\cdot P_{trans}(2,3)\cdot P_{cost}(3)\cdot P_{trans}(3,4)\cdot P_{cost}(4) \quad (2)$$
$$= P_{cost}(1)\cdot P_{cost}(2)\cdot P_{cost}(3)\cdot P_{cost}(4)$$

The cost probability is computed using heuristic rules which observe the voice descriptors. Given an observation window (a beginning and ending frame) and a given note model, a set of rules are applied to the voice descriptors and each rule computes a rule probability. The cost probability is then the multiplication of all the rule probabilities.

At the end of the song the system can perform a backtracking on the viterbi matrix and compute the best path among all. We can see in the following figure the best expression path: silence – attack – transition – vibrato – release – silence.

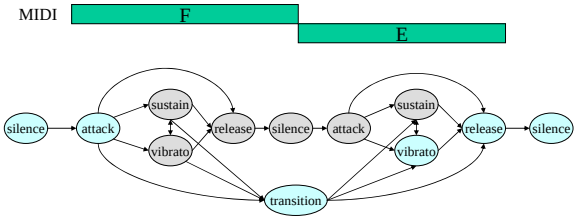

Figure 8 Best expression path.

Once the best expression type path has been chosen, then the most probable label for each expression type has to be estimated. We can see the final results in figure 9. The final segmentation is shown on the top, accurately indicating begin and end time of each expression type. The attack has been labeled as “scoop up”, the transition as “normal”, the vibrato as “regular” and the relase as “fall down”.

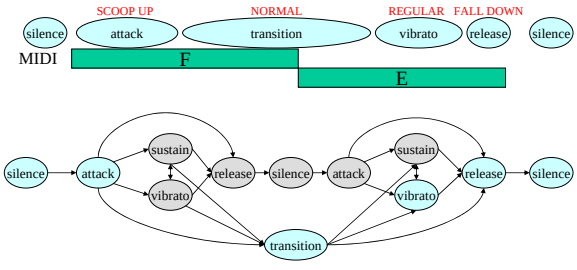

Figure 9 Best expression path with labels.

5. THE GRAPHICAL SEGMENTATION TOOL

A graphical tool have been created inside the smstools2 software package that helps the user to change the boundaries of the notes and its inside regions, it allows to add, delete or change label regions.

Figure 10 Segmentation tool to supervise results.

The GUI tool also visualizes the probability for any region or note rule applied to be the one that the user has manually edited (figure 10), so when the user changes the boundaries, the probabilities of each rule are updated and the aim is to obtain the probabilities that minimize the global sum of all rule probabilities.

Figure 11 Smstools2 software.

The GUI tool has views to display the descriptors calculated in the analysis process so the user can view values of these descriptors at any time to change the heuristic rules and improve the automatic note segmentation and expression transcription results. In figure 11 we can view in the bottom window the display of values of some analysis descriptors against time simultaneously, each descriptor with a different color. Above this window we can see the pitch curve of the performance, and the results of the note-segmentation and expression transcription as well as the notes of the midi score.

6. EVALUATION

Three commercial pop songs have been used to evaluate the system and some amateur singers have been asked to sing the songs. The recorded performances have been analyzed and note segmentation and expression transcription have been performed, from these analysis results, more than 1000 notes have been evaluated achieving more than 95% accuracy in the note segmentation, using as reference manual segmentation by a musician and allowing a tolerance of 30 milliseconds, so boundaries automatically segmented within this margin are considered as correct. For the expression transcription evaluation, there is no a simple way to evaluate the results, as sometimes there are many ways to correctly transcript the same performance, for instance putting a very-short sustain or not when the stable part of a note is about 50ms, or putting a long sustain fall-down and a short normal release instead of a short sustain and longer release fall-down. For this reason, the expression transcription evaluation has been done manually observing the results by an expert and making annotations about the segmentation errors. More than 2000 expression sub-regions have been checked and more than 95% of the correctly note-segmented ones are correctly transcribed.

7. ACKNOWLEDGEMENTS

This work was supported by the Music Technology Group in the Pompeu Fabra University and Yamaha Corp.

8. REFERENCES

[1] Sundberg, J, Frydén, L & Friberg, A (1995) . Expressive aspects of instrumental and vocal performance. (invited paper). In R Steinberg, ed.: Music and the Mind Machine. The Psychophysiology and Psychopathology of the Sense of Music, Berlin: Springer. 49-62.

[2] Sundberg, J., Iwarsson, J., & Hagegård, H. (1995) . A singer's expression of emotions in sung performance. (Invited paper). In O Fujimura and M Hirano, ed:s. Vocal Fold Physiology: Voice Quality Control, San Diego: Singular Press 1995, 217-229.

[3] SingStar: karaoke game for Sony's Playstation 2. It allows you to evaluate how good you are when you sing by analyzing your voice pitch. . http://www.singstargame.com/

[4] Ultrastar: open source PC conversion of famous karaoke game Singstar. . http://sourceforge.net/projects/ultrastar/

[5] Karaoke Revolution. Karaoke game for Sony's Playstation 2. The scoring system here is based on pitch and rhythm. . http://www.karaokerevolution.net/

[6] Viitaniemi, T., Klapuri, A. & Eronen, A., (2003). . A probabilistic model for the transcription of single-voice melodies. In: Huttunen, H., Gotchev, A. & Vasilache, A. (eds.), Proceedings of the 2003 Finnish Signal Processing Symposium, FINSIG'03, Tampere, Finland, 19 May 2003, pp. 59-63.

[7] Ryynänen, M. P. & Klapuri, A. P., (2004) . Modelling of note events for singing transcription. In Proceedings of ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Processing, SAPA 2004, Jeju, Korea, 3 October 2004, pp. 6 p.

[8] Cano, P., Loscos, A. & Bonada, J., (1999) . Score-Performance matching using HMMs. In Proceedings of the International Computer Music Conference, Beijing, China, October 22-28, 1999. Pages 441-444.

[9] Dannenberg, Roger B. & Ning Hu. (2003). . Polyphonic Audio Matching for Score Following and Intelligent Audio Editors. In Proceedings of the 2003 International Computer Music Conference San Francisco: International Computer Music Association, pp. 27-33.

[10] Sundberg, J., (2001) . Expression in music: Comparing vocal and instrumental performance. In I Karevold, H Jörgensen, IM Hanken, E Nesheim, eds, Flerstemmige innspill 2000, NMH-publikasjoner 2000:1, Norges Musikkhögskole, Oslo, 5-19.