

AUDIO KEY FINDING USING FACEG: FUZZY ANALYSIS WITH THE CEG ALGORITHM

Ching-Hua Chuan

Department of Computer Science
University of Southern California
Integrated Media Systems Center
Los Angeles, CA
chinghuc@usc.edu

Elaine Chew

Epstein Dep of Industrial & Systems Engineering
University of Southern California
Integrated Media Systems Center
Los Angeles, CA
echew@usc.edu

Our key finding system consists of a series of $O(n)$ real-time algorithms for determining key from polyphonic audio. The system comprises of two main parts as shown in Figure 1 [1]. The first part (the upper dashed box) generates pitch class information from audio using the standard FFT and a fuzzy analysis technique. The second component (the lower dashed box) uses the pitch class information to determine the key using Chew's Spiral Array model and Center of Effect Generator (CEG) key finding algorithm [2, 3]. A cleanup procedure uses key information to clarify the input

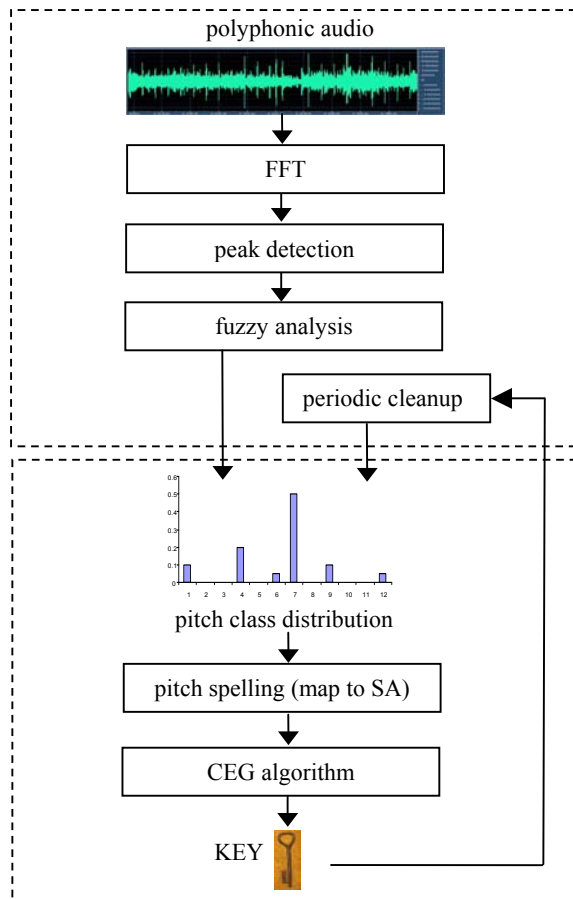


Figure 1. Graph of audio key finding system.
distribution periodically.

1 PITCH CLASS DETERMINATION

The first component of the system consists of three steps for generating pitch class information from audio. We use the Fast Fourier Transform (FFT) to extract frequencies from audio wave signals. Then we employ a peak detection method to map frequencies to pitches. A fuzzy analysis technique is employed to improve the

correctness of pitch class information using information on the overtone series.

We use the standard FFT to generate a frequency spectrum for each audio wave. We first define standard reference frequencies, from C_2 to B_6 , assuming that the pitch A above middle C is 440 Hz. The frequency range for each pitch is then bounded by the midpoints between its adjacent and its own reference frequencies. The peak detection method selects the maximum value within each frequency range using a heuristic based on the assumptions that a peak value is larger than the average values to its left and right, and that only one such maximum value exists. We use the maxima identified in the frequency ranges to construct the pitch class distribution. Details are provided in [4].

The fuzzy analysis technique improves the correctness of the pitch class distribution based on the overtone series. Because pitch frequencies are defined on a logarithmic scale, lower pitches produce more errors in mapping from frequency to pitches than higher ones. The fuzzy analysis technique uses the presence of the first overtones to clarify pitches of frequencies below 261 Hz (the pitch C_4). More details are given in [1].

2 KEY FINDING FROM PITCH CLASS INFORMATION

Key assignment from pitch class information forms the second component of the system. We use Chew's Spiral Array model [2, 5] to represent the pitches in a 3-D space. The model forms the basis for both the key finding and the pitch spelling algorithms [6, 7]. We map the pitch class information from the previous part of the system to the Spiral Array model using the sliding window pitch spelling algorithm described in [7]; the CEG (Center of Effect Generator) key finding algorithm [3] is then used to determine the key.

The Spiral Array is a 3-dimensional model that represents pitches, intervals, chords, and keys in the same space for easy comparison. The pitches are shown as points on a helix, and adjacent pitches are related by intervals of perfect fifths, while vertical neighbors are related by major thirds. Central to the Spiral Array is the idea of the center of effect (CE), the representing of tonal objects as the weighted sum of their lower level components. In audio key finding, the CE is the sum of pitch points weighted by the pitch class distribution.

In the Center of Effect Generator (CEG) algorithm, key is determined by a nearest neighbor search among the major and minor key representations in the Spiral Array space. For short audio samples, the CEG algorithm progressively uses the cumulative pitch class information to determine the key.

To eliminate excessive non-tonal noise that may have accumulated in the pitch class distribution, we introduce a periodic cleanup procedure. The cleanup process runs periodically, and uses the key results to zero-out the weights on pitch classes that are not in the key, and those that comprise the four smallest values among the 12 pitch classes.

The key finding system generates a key answer every 0.37 seconds with non-overlapping sliding windows. The pitch classes are re-spelt whenever an answer is generated by the accumulative information of pitch classes from previous 5 seconds. The cleanup procedure adjusts pitch class distribution every 2.5 seconds.

To generate one answer as the global key, we decide a stopping point from the optimized results of the training data. Instead of taking the whole piece into account, we only test the beginning part of the piece (starting from the non-silent signal to the stopping point). At the stopping point, the results of the training data have the highest correct percentage. The stopping point we obtained from the training data is 13.32 seconds. The global key is determined by the majority of the key answers of the segments from the beginning to 13.32 seconds.

3 EVALUATION RESULTS

The evaluation was conducted using 1,252 audio files synthesized from two synthesizing software: Winamp and Timidity. The points for each key answer were assigned according to its relation to the ground truth (listed in Table 1). The composite score is calculated by averaging the Winamp and Timidity scores. Table 2 records the evaluation results for our system.

Table 1. Point assignments for key answers

	Correct	Perfect 5 th	Relative	Parallel	Others
Point	1	0.5	0.3	0.2	0

Table 2. Summary of the evaluation results

System and algorithms	Fuzzy analysis CEG algorithm with Spiral Array model	
Rank	7	
Composite percentage score	79.10%	
Synthesizer	Winamp	Timidity
Total score	1002.3	977.3
Percentage score	80.1%	78.1%
Correct keys	937	905
Perfect 5 th errors	83	95
Relative major/minor errors	66	68
Parallel major/minor errors	20	22
Other errors	146	162
Runtime (s)	3299	3468
Machine	OS:XP, Processor: Intel P4 3.0GHz, RAM:3GB	

4 SYSTEM COMPARISONS

Six groups participated in the audio key finding contest for MIREX 2005, including Chuan & Chew, Gómez, İzmirl, Pauws, Purwins & Blankertz, and Zhu (listed alphabetically). In this section, we compare the known

algorithms between proposed systems based on the short abstracts made available to all participants (first parts of [8, 9, 10]). Details for Purwins & Blankertz and Zhu's systems were not available at the time of writing.

Table3. Summary of system comparisons

	Chuan & Chew	Gómez	İzmirlı	Pauws
Sample window size	370ms	93ms	unknown	100ms
Pitch extraction	FFT	FFT	FFT	FFT
Resolution	Semitone	1/3 semitone	Semitone	Semitone
Audio characteristic analysis	Fuzzy analysis with periodic update	Modified K-S profile based on audio characteristics	Combination K-S/T, with weighted spectra profile	Chroma spectrum
Key templates	Key representations in Spiral Array			Profiles from training data
Query	CE in Spiral Array	HPCP	Chroma templates	Subharmonic summation
Key finding method	CEG	K-S (correlation)	K-S (correlation)	Some statistical measure
Selection criteria	Nearest key (Euclidean distance) at stopping point	Template with highest correlation coefficient at start or for entire file	Highest votes based on sum of correlation coefficients weighted by confidence values	unknown

We list the each system's sample window size, pitch extraction technique, resolution, audio characteristic analysis method, key templates, query method, key finding method, and key selection criteria in Table 3.

From Table 3, we can see that the major differences between the systems are: audio characteristics analysis, key templates, and final selection criteria. The first two are sometimes combined in the design of the system. In Chuan & Chew's system, we use the key spirals as the pre-computed key templates. We used a fuzzy analysis technique to clarify the audio signals in order to generate more accurate pitch classes. In Gómez's system, the key templates are also pre-computed; starting from the Krumhansl-Schmuckler pitch class profiles [11], she alters the profiles to take into account the chord and harmonics characteristic of audio signals. In İzmirlı's and Pauws' systems, the key templates are constructed with audio characteristics. İzmirlı creates the key templates from monotonic instrument sounds, weighted by a combination of the K-S and Temperley's modified profiles [12]. The key templates in Pauws' system are completely data-driven, which are learned from training data.

It is one matter to determine key, a feature that varies over time, but another to decide on which key to select as the "solution" when one must do so in a competition of this kind. We decided to use only a segment at the beginning of a test file. The best selection was determined to be the first 13.32 seconds of each piece, a

parameter that yielded the most number of correct answers on the test set provided by the organizers. The reason for choosing a sample from only the beginning of the piece is that that is when the key is most likely to be established before modulating to other related keys. This assumption is validated by the results of Gómez's two system submissions. In the two versions of the system, the one using only the "start" portion of a piece performs better than the one employing the entire piece (labeled "global"). It is important to note that this assumption would not hold true for every piece.

The most effective selection criterion was proposed by İzmirli. His system tracks the confidence value for each key answer (a number based on the correlation coefficient between the query and key template), and the global key was selected as the one having the highest sum of confidence values over the length of the piece. The evaluation results demonstrate the importance of the selection criteria.

Apart from its selection criterion, we posit that İzmirli's system's winning performance in the audio key finding competition is due in part to his use of a combination of knowledge-driven (Temperley profiles), perceptually-supported (K-S profiles) and data-driven templates. Gómez's modified K-S profile, based on hypotheses of chord and harmonic effects, also works well, but it appears that theoretical propositions may not be as inclusive of audio-specific features as information learned from audio sounds. We used the Spiral Array's key representations (our key templates) without altering them to fit audio characteristics. The parameters for the original parameters for the key spiral were selected based on constraints derived from symbolic pitch and key relations.

In contrast, Pauws' method depends strongly on the representative ability of the training data. Both İzmirli and Pauws' queries are pitch class distributions constructed in the frequency domain, rather than duration profiles, which may be a more appropriate measure when comparing to the FFT results. In both Gómez's and our approach, we compared some mapping of the duration profile or pitch positions weighted by durations to the key templates.

Distinct from other systems, we reconstruct the pitch class information from audio using a fuzzy analysis technique, before applying the key finding algorithm. The reconstruction of pitch information explored the effect of audio properties such as the harmonics and frequency resolution of audio signals.

5 CONCLUSIONS

An advantage and disadvantage of our system is its lack of dependence on training data. Only the selection criterion (the stopping point) was determined from the sample test set. All other system parameters are pre-determined, and unaffected by training data.

The key finding contest has allowed us to closely inspect and compare several key finding systems in close proximity. Our analysis of the systems and their performances reveal several avenues for improving ours. Improvements that can be made include:

- using a smaller audio sample window size, and separation of the analyses for low and high frequency audio signals;

- the calibrating of the Spiral Array models' parameters to take into account audio frequency features in the positioning of the key representations (templates), a technique similar to Gómez's and İzmirli's treatment of the K-S profiles; and,

- using distance of query from key representations to quantify the confidence of each key choice in order to obtain a more accurate selection of the most likely key.

6 ACKNOWLEDGEMENTS

We thank the MIREX team led by Stephen Downie, and in particular, Andreas Ehmann, Emmanuel Vincent, and Kris West for amassing the database and running the key finding evaluations, and Arpi Mardirossian for her music truncation software.

This research has been funded in part by, and made use of the shared facilities at, the Integrated Media Systems Center, an NSF-ERC, Cooperative Agreement No. EEC-9529152. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors, and do not necessarily reflect those of NSF.

Keywords: polyphonic audio key finding, pitch class profile, evaluation.

REFERENCES

- [1] Chuan, C. H. and Chew, E., "Fuzzy Analysis in Pitch Class Determination for Polyphonic Audio Key Finding," Proceedings of the 6th International Conference on Music Information Retrieval, London, 2005.
- [2] Chew, E. "Towards a Mathematical Model of Tonality", Doctoral dissertation, Department of Operations Research, Massachusetts Institute of Technology, Cambridge, MA, 2000.
- [3] Chew, E., "Modeling Tonality: Applications to Music Cognition", Proceedings of the 23rd Annual Conference of the Cognitive Science Society, Edinburgh, Scotland, 2001.
- [4] Chuan, C. H. and Chew, E., "Polyphonic Audio Key Finding Using the Spiral Array CEG Algorithm", Proceedings of the IEEE International Conference on Multimedia and Expo, Amsterdam, The Netherlands, 2005.
- [5] Chew, E. "Slicing it all ways: mathematical models for tonal induction, approximation and segmentation using the Spiral Array", *INFORMS Journal on Computing*, to appear.
- [6] Chew, E. and Chen, Y. C., "Mapping MIDI to the Spiral Array: Disambiguating Pitch Spellings", H. K. Bhargava and Nong Ye (Eds.), *Computational Modeling and Problem Solving in the Networked World*, Kluwer, pp.259-275. Proceedings of the 8th *INFORMS Computer Society Conference*, ICS2003, Chandler, AZ, Jan 8-10, 2003.
- [7] Chew, E. and Chen, Y. C., "Real-Time Pitch Spelling Using the Spiral Array", *Computer Music Journal*. 29:2, Summer 2005.
- [8] Gómez, E., "Key Estimation from Polyphonic Audio," Abstract of the Music Information Retrieval Evaluation Exchange, 2005.

- [9] İzmirlı, O., "Algorithm for Key Finding from Audio," Abstract of the Music Information Retrieval Evaluation Exchange, 2005.
- [10] Pauws, S., "KEYEX: Audio Key Extraction," Abstract of the Music Information Retrieval Evaluation Exchange, 2005.
- [11] Krumhansl, C., "Cognitive Foundations of Musical Pitch," Oxford University Press, New York, NY, USA, 1990.
- [12] Temperley, D., "What's Key for Key: The Krumhansl-Schmuckler Key-Finding Algorithm Re-considered," *Music Perception*, 17(1), 65-100, 1999.