Polyphonic Audio Key Finding Using the Spiral Array CEG Algorithm

Ching-Hua Chuan[†] and Elaine Chew[‡]
University of Southern California Viterbi School of Engineering
[†]Department of Computer Science, [‡]Epstein Department of Industrial and Systems Engineering
Integrated Media Systems Center, Los Angeles, CA
{chinghuc, echew}@usc.edu

Abstract

Key finding is an integral step in content-based music indexing and retrieval. In this paper, we present an $O(n)$ real-time algorithm for determining key from polyphonic audio. We use the standard Fast Fourier Transform with a local maximum detection scheme to extract pitches and pitch strengths from polyphonic audio. Next, we use Chew's Spiral Array Center of Effect Generator (CEG) algorithm to determine the key from pitch strength information. We test the proposed system using Mozart's *Symphonies*. The test data is audio generated from MIDI source. The algorithm achieves a maximum correct key recognition rate of 96% within the first fifteen seconds, and exceeds 90% within the first three seconds. Starting from the extracted pitch strength information, we compare the CEG algorithm's performance to the classic Krumhansl-Schmuckler (K-S) probe tone profile method and Temperley's modified version of the K-S method. Correct key recognition rates for the K-S and modified K-S methods remain under 50% in the first three seconds, with maximum values of 80% and 87% respectively within the first fifteen seconds for the same test set. The CEG method consistently scores higher throughout the fifteen-second selections.

1. Introduction and Background

In tonal music, the relations among pitches that generate the key constitute one of the main features of a melody. Having the key information will be valuable for content-based music indexing and retrieval. In this paper we propose a method for extracting tonal features from polyphonic audio in real time. Using the pitch class and pitch strength information from the standard Fast Fourier Transform (FFT), the Spiral Array Center of Effect Generator (CEG) (see [2][3]) algorithm generates the key and returns the key name in a completely automatic fashion. Our choice of the CEG algorithm is distinct from existing approaches which typically use Krumhansl & Schmuckler's (K-S's) [10] probe tone profile method. We choose the CEG method for several reasons. The Spiral Array is a hierarchical model that represents pitches, intervals, chords, and keys within a single spatial framework, allowing for comparisons and analyses of tonal structures for multiple purposes. The CEG algorithm performs an efficient nearest-neighbor search for the closet key in the 3-dimensional Spiral Array model. The Spiral Array model offers ways to solve the pitch-spelling problem efficiently (see[4][5][6]), which provides us with the capability to determine the key completely automatically. The CEG algorithm has also been shown to perform better in differentiating keys with similar pitch classes than other methods [3]. We extend the CEG algorithm from key finding using symbolic data to key finding from polyphonic audio. With the pitch class and pitch strength information generated by the standard FFT, we show that the CEG algorithm provides a correct rate of over 90% within the first three seconds, and achieves a maximum correct rate of 96%, when tested on 61 audio samples of Mozart's symphonies. For comparison, we also feed the same pitch class and strength information to the K-S method and Temperley's modified version of the K-S method [18]. The K-S and improved K-S methods provide results of less than 50% in the first three seconds and reach maxima of 80% and 87% respectively.

Related work in audio tonal description includes approaches proposed by Gómez & Herrera [9] and Pauws [14]. Both approaches use the standard FFT as the basis for their pitch detection methods and the K-S method for key finding. Gómez & Herrera used three times the traditional resolution of the pitch frequency spectrum of the FFT method for pitch detection. They used a Harmonic Pitch Class Profile as input to the K-S method to find the key. Pauws incorporated rules for avoiding noise and emphasizing pitch loudness, and applied the K-S method to generate the key. Gómez & Herrera reported an 84% correct key detection rate among 878 excerpts of classical music. Pauws used 237 classical piano sonatas as the test set and his method returned a result of 59.1% within 5 seconds, and reached the maximum of 72.2% within 30 seconds. Also a form of tonal description, but stopping short of determining key, Tzanetakis, Ermolinskyi, and Cook [7] generated pitch histograms in audio and symbolic music for information retrieval and genre classification.

Some pitch detection method must first be applied to extract pitches from polyphonic audio in order to find the key of any acoustic music excerpt, including a live performance, a CD or a tape recording. Most algorithms for pitch detection are designed to extract precisely every pitch present and none others. We hypothesize that such exact pitch detection may not be necessary for key finding using the CEG algorithm, and that pitch information from the harmonics may actually assist in determining key more quickly and accurately. Firstly, octave imprecision is not a relevant issue in key finding. Secondly, pitches that belong to the same key are close to each

0-7803-9332-5/05/$20.00 ©2005 IEEE

other in the Spiral Array model. These pitches also tend to form the strongest harmonics of a tone. Since the CEG algorithm generates a center of effect (CE) from pitch information and determines the key by a nearest neighbor search, the key recognition procedure is not affected by octave displacement, nor is it strongly affected by the addition of harmonics.

In this paper, we propose a heuristic peak selection algorithm to determine pitch class and pitch strength information from a standard FFT. Approaches that use sophisticated signal processing methods to precisely identify each pitch in an excerpt have been proposed in the literature [12][13][14][15]. To reduce the computational complexity of exact pitch detection, Tolonen and Karjalainen built a two-channel model [19]. Instead of using signal processing methods alone to improve the correctness of pitch estimation, Szczerba & Czyzewski [17] added a neural network module for pitch prediction. Another approach based on neural network proposed by Dziubinski & Kostek [8] reduced the octave and harmonic errors to build an octave immune pitch detection system. The approach which has the highest correct rate reported so far was proposed by Abdallah and Plumbley [1]. They used a probabilistic model to transcribe a live recording of Bach's Fugue in G-minor No. 16, with only one note error for the first nine and a half bars. Research for extracting every single pitch in audio improves the correct pitch detection rate by eliminating the harmonic and octave errors. However, these information may benefit key determination either by emphasizing the root pitch or by constructing the stronger harmonics of a tone.

We use the CEG algorithm proposed by Chew [2][3] to determine key from pitch class and strength information. Other researchers have proposed solutions to the key finding problem. In 1971, Longuet-Higgins & Steedman [11] proposed an algorithm that used the Harmonic Network, a two-dimensional array representing salient pitch relations, to differentiate between major and minor keys by shape mapping. In 1986, Krumhansl & Schmuckler developed a widely accepted model called the probe tone profile method (henceforth referred to as the K-S method), which constructs pitch class profiles for major and minor keys by using user ratings from probe tone experiments. By calculating the correlations among the pitch information and the template key profiles, the key is determined as the one with the highest correlation value. In 1999, Temperley [18] improved upon the K-S method by modifying the key profiles to emphasize the differences between diatonic and chromatic scales. Temperley also adjusted the weights of the forth and seventh pitches so as to differentiate the keys which have highly similar pitch class signatures. These approaches have been used in numerous key-findings projects.

The rest of the paper is organized as follows. Section 2 explains the pitch detection method and the CEG algorithm using the Spiral Array model. Section 3 describes the evaluation experiments and presents the key finding results for the CEG algorithm, K-S method, and Temperley's improved K-S method. Discussions and conclusions follow in Section 4.

2. The CEG Algorithm for Audio Key Finding

In this section, we present an *O(n)* algorithm to determine the key from polyphonic audio based on the CEG algorithm. The sequence of actions is depicted graphically in Figure 1. The method consists of two stages – pitch detection and key finding. The pitch detection part is responsible for recognizing and transforming polyphonic audio into pitch class and pitch strength information. The key finding part uses the Spiral Array CEG algorithm to generate the key. Both stages are described in Sections 2.1 and 2.2 respectively.

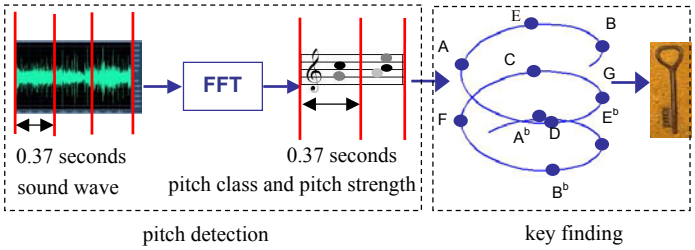


Figure 1. The system contains two main parts: Pitch Detection and Spiral Array Model with CEG key finding algorithm.

2.1 Pitch Detection

We use the standard FFT to generate the frequency spectrum that gives pitch class and strength information (see Figure 2). Relations between pitches are reflected on the logarithmic scale in frequency, and a range of frequencies maps to each pitch. The midpoints between adjacent reference frequencies [21] act as the boundaries for the frequency bands of each pitch. We limit the frequency range to be from 32 Hz (C1) to 1975 Hz (B6). We select 0.37 seconds as our sampling interval because harmonics produce negligible effects at this sampling size. The information we obtained after performing an FFT is a collection of frequency peaks as shown in the example in Figure 2(b). Due to the limited resolution of FFT, numerous local maxima may be found within a pitch frequency band. We employ a heuristic peak selection algorithm to determine the pitch classes present and their relative strength.

Our Local Maximum Selection method for peak selection is based on the following assumptions: (1) a peak value is defined as one that is larger than the average value to its left in the frequency band and that to its right; and (2) within each pitch frequency band, at most one peak value (the highest one) can exist.

We map the peak values into 12 pitch classes with the distance of one semitone between adjacent pitches. This

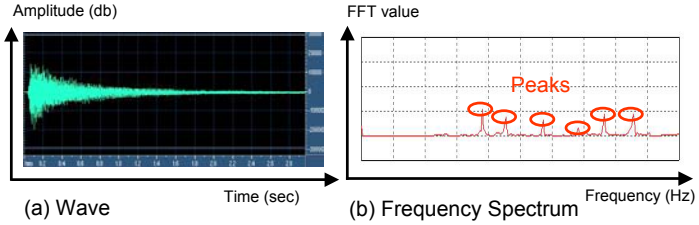


Figure 2. A polyphonic audio sample (a chord with 6 pitches played simultaneously) is shown in wave format (a) in the time domain and (b) in the frequency domain after an FFT.

mapping procedure results in pitch strength quantities for each pitch class. The mapping is designed for key finding purposes and octave relationships are ignored as they do not affect the key. Hence, we sum the peak values for all frequency bands related by octaves to obtain the pitch strength value for a pitch class. The pitch strength values are normalized by first dividing by the largest value, then by the sum of all values. A visual representation of pitch class and strength is shown in Figure 3(a). The pitch with larger strength is shown in darker color. Details of the procedure are outlined in Table 1.

Table 1. Algorithm for extracting Pitch Class and Strength

For each frequency spectrum obtained from a 0.37 second segment:

1. For each frequency band:
   (1) scan from low-to-high frequencies to find the peak value s.t. this value is larger than the average value to its left and the average value to its right; and,
   (2) if more than one peak is found, choose the highest peak value.
2. For each pitch class, k, its strength at time j, F_{jk}, is the sum of all peak values for that frequency band and others related by octaves.
3. Normalize step 1: divide all pitch strength values by the largest one

$$Fjk = \frac{Fjk}{\max_j \{ Fjk \}}, \quad \text{where } k = 0,...,11.$$

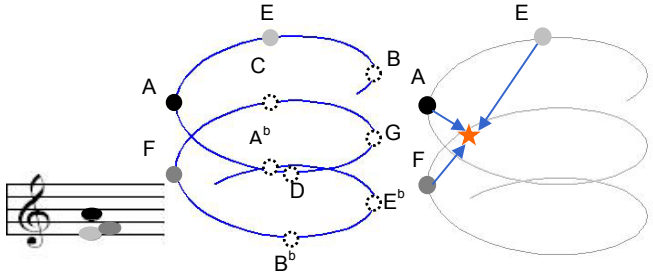
4. Normalize step 2: divide all pitch strength values by their sum

$$Fjk = Fjk \Big/ \sum_{j=0}^{11} Fjk, \quad \text{where } k = 0,...,11.$$

2.2 The Spiral Array Model and Pitch Strength Weighted CEG Key Finding Algorithm

The Spiral Array Model, proposed by Chew in 2000, is a 3-dimensional model that represents pitches, intervals, chords and keys in the same three-dimensional space for easy comparison. On the Spiral Array, pitches are represented as points on a helix, and adjacent pitches are related by intervals of perfect fifths, while vertical neighbors are related by major thirds as shown in Figure 3(b). 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. The details for constructing the nested spirals in the model are given in [2] and [3].

In the Center of Effect Generator (CEG) algorithm, key selection is performed by a nearest neighbor search in the Spiral Array space as shown in Figure 4. Instead of using the relative pitch durations as the CE weights, we use the normalized pitch strengths to generate the CE. For example, Figure 3(c) shows the CE of three pitches {F, A, and E},



(a) Pitch Strength (b) Spiral Array Model (c) Center of Effect

Figure 3. (a) and (b) illustrate the procedure of mapping pitch strengths onto Spiral Array model and (c) calculates the Center of Effect (CE) by pitch strength weighted formula in (1).

weighted by their pitch strengths. In order to map numeric pitch classes to their appropriate pitch names in the Spiral Array, we use the pitch spelling method described in [4] and [5]. The key finding process is described in Table 2.

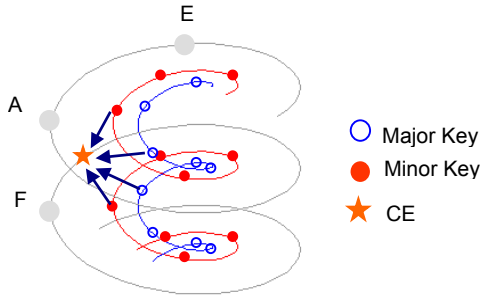


Figure 4. The Center of Effect Generator (CEG) algorithm performs a nearest-neighbor search in the Spiral Array model.

Table 2. Method for Determining Key

For pitch class and strength information from each 0.37 second segment:

1. Assign pitch names to pitch classes (pitch spelling):
   (1) generate CE for previous 5 seconds (or part thereof); and,
   (2) assign pitch names to current pitch classes by nearest neighbor search in Spiral Array space .
2. Determine key:
   (1) generate the cumulative CE from beginning to current point

$$\text{CE}(1,i) = \sum_{j=1}^{i} \sum_{k=1}^{12} \frac{Fjk * Pk}{i},$$

   where Pk is the position of pitch class k; then,
   (2) perform nearest-neighbor search to find closest key.

CEG algorithm has been implemented successfully in a real-time application, MuSA.RT [6], which presents opportunities for an alternate input modality (audio rather than MIDI) and pre-processing method (approximate pitch detection).

2.3 Comparisons with Other Methods

We also implement the K-S method and Temperley's modified K-S method to examine the performance of CEG algorithm. In the K-S and modified K-S methods, each key is represented by a unique numerical key profile. Key finding is done by correlating the duration profile of the pitch classes with the key templates. The template with the highest correlation is chosen as the key. The key profiles for K-S method and Temperley's (T) method are shown in Figure 5, where the x-axis shows the number of half steps from the tonal center and the y-axis shows the average ratings.

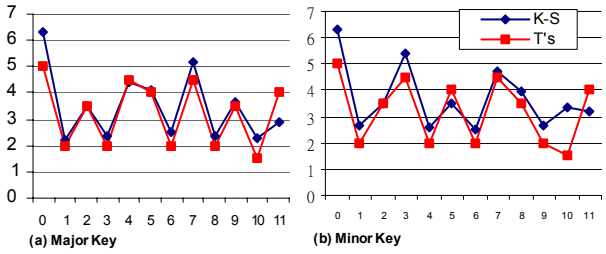


Figure 5. Key profiles for K-S and Temperley's methods.

3. Evaluation Results

To evaluate the key finding system, we choose Mozart's symphonies. The test set consists of a total of 61 files

representing renditions of 28 symphonies by Mozart. Details of the symphonies represented are listed in Table 3. The symphonies were obtained from *www.classicalarchives.com*. The key of each symphony is stated explicitly in its title. Furthermore, the symphony is a composition for orchestra, containing sounds from a wide variety of acoustic instruments. A typical symphony contains multiple movements in different tempi and keys. The first movement is often cast in sonata form, in which the music departs from and returns to the main key. The second and third movements are frequently in related keys while the last movement would return to the home key. For these reasons, we use only the first fifteen seconds of the first movements so that the test samples are likely to remain in the stated key for the entire duration of the sample.

Table 3. The Mozart Symphonies

* Uppercase letter names represent major keys and lowercase minor keys.

Symphony #	1	4	5	6	7	9	11	14	16	17	20	22	25	27
Key	Eb	D	Bb	F	D	C	D	A	C	G	D	C	g	G
Versions	1	1	1	1	1	1	1	1	1	1	1	2	3	1

Symphony #	28	29	30	31	32	33	34	35	36	37	38	39	40	41
Key	C	A	D	D	G	Bb	C	D	C	G	D	Eb	g	C
Versions	1	3	3	2	1	1	2	4	3	1	5	3	7	8

First, we transform the MIDI files into wave format and run the pitch detection method every 0.37 seconds to obtain the cumulative pitch class and strength information. We use this same pitch information as the input to the CEG algorithm, the K-S method, and Temperley's improved K-S method. The results at each 0.37 second time chunk are shown in Figure 6. An answer is considered to be wrong if the algorithm's choice of key is different from the stated one.

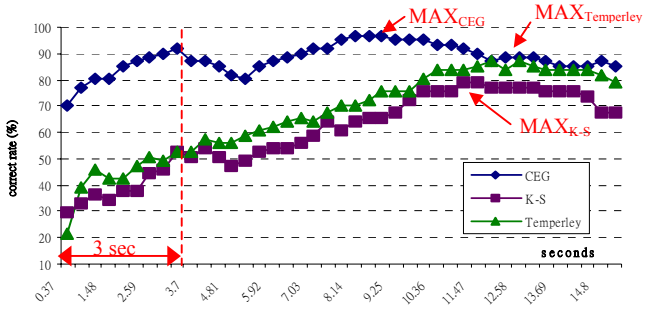


Figure 6. The comparison results of CEG algorithm, K-S method, and Temperley's improved K-S methods with Mozart symphonies.

Compared to the K-S and modified K-S methods, the pitch strength weighted CEG algorithm consistently achieves the best results for the Mozart test set throughout the fifteen second segments. As shown in Figure 6, the correct rate exceeds 90% within the first three seconds and maintains a rate of over 90% from 7.5 to 12 seconds. The best overall result, 96% correct, is achieved 8 seconds into the pieces. In contrast, the correct rates for the K-S model and Temperley's improved K-S method are under 50% within the first three seconds; the results improve over time to reach optima of 80% and 87% at 11 seconds and 14 seconds respectively.

4. Discussion and Conclusions

We have presented an $O(n)$ real-time algorithm for determining key from polyphonic audio. The algorithm uses a local maxima selection method to determine pitch class and strength from FFT results coupled with key finding using the Spiral Array CEG method. The FFT and local maxima selection method produced pitch class information reinforced by natural harmonics of the tones present. The system shows promising results when tested on Mozart's Symphonies, with the CEG algorithm performing better than the K-S and modified K-S methods. Future work includes testing the system on a larger and more varied corpus of music, conducting further experiments to determine the effect of pitch class information reinforced by harmonics, and exploring ways to bias the pitch detection process to benefit key finding.

References

[1] Abdallah, S. A. and Plumbley, M.D. (2004) *Polyphonic Music Transcription by Non-Negative Sparse Coding of Power Spectra*. ISMIR 2004 – 5th International Conference on Music Information Retrieval.

[2] Chew, E. (2000). *Towards a Mathematical Model of Tonality*. Doctoral dissertation, Department of Operations Research, Massachusetts Institute of Technology, Cambridge, MA.

[3] Chew, E. (2001). *Modeling Tonality: Applications to Music Cognition*. Proceedings of the 23rd Annual Conference of the Cognitive Science Society, Edinburgh, Scotland.

[4] Chew, E. and Chen, Y. C. (2003). *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.

[5] Chew, E. and Chen, Y. C. (2005). *Real-Time Pitch Spelling Using the Spiral Array*. *Computer Music Journal*. 29:2, Summer 2005.

[6] Chew, E. and François, A. R. J. (2003). *MuSA.RT - Music on the Spiral Array . Real-Time* . In Proceedings of the ACM Multimedia '03 Conference, MM'03 , Berkeley, CA, Nov 2-8, 2003.

[7] Cook, Perry R. (2001). *Music, Cognition, and Computerized Sound: An Introduction to Psychoacoustics*. MIT Press.

[8] Dziubinski M. and Kostek B. (2004). *High Accuracy and Octave Error Immune Pitch Detection Algorithms*. *Archives of Acoustics*.

[9] Gómez, E. Herrera, P. (2004). *Estimating The Tonality Of Polyphonic Audio Files: Cognitive Versus Machine Learning Modelling Strategies*. ISMIR 2004 – 5th International Conference on Music Information Retrieval.

[10] Krumhansl, C.L. (1990). *Quantifying Tonal Hierarchies and Key Distances. Cognitive Foundations of Musical Pitch*, chapter 2, 16-49.

[11] Longuet-Higgines, H.C., Steedman, M.J. (1971). *On Interpreting Bach*. *Machine Intelligence*, vol. 6, 221-241.

[12] Mitra, Sanjit K. (2001). *Digital Signal Processing: A Computer Based Approach*, 2nd Edition. McGraw-Hill

[13] Noll, A.M. and Schroeder, M.R. (1964) *Short-time Cepstrum Pitch Detection. Journal of the Acoustical Society of America*, vol 36, pp 1030.

[14] Pauws, S. (2004) *Musical Key Extraction from Audio*. ISMIR 2004 – 5th International Conference on Music Information Retrieval.

[15] Rabiner and Gold (1975). *Theory and Applications of Digital Signal Processing. IEEE Transactions on Acoustics, Speech, and Signal Processing AU-20; 322-337.*

[16] Schroeder, M.R. (1968). *Period Histogram and Product Spectrum: New Methods for Fundamental-Frequency Measurement. Journal of the Acoustical Society of America*, vol. 43, pp. 829-834.

[17] Szczerba, M. and Czyzewski, A. (2002). *Pitch Estimation Enhancement Employing Neural Network-Based Music Prediction*. Proc. IASTED Intern. Conference, Artificial Intelligence and Soft Computing, 413 - 418, 17.7.2002-19.7.2002, Banff, Canada.

[18] Temperley, D. (1999). *What's Key for Key? The Krumhansl-Schmuckler Key-Finding Algorithm Reconsidered*. *Music Perception*, 17(1), 65-100.

[19] Tolonen, T. and Karjalainen, M. (2000), *A Computationally Efficient Multipitch Analysis Model. IEEE Trans. On Speech and Audio Processing*, 8(6): 708-716.

[20] Tzanetakis, G. Ermolinskyi, A. and Cook, P. (2003). *Pitch Histograms in Audio and Symbolic Music Information Retrieval. Journal of New Music Research*, 32(2), 143-152.

[21] International Organization for Standardization, *Acoustics – Standard Turing Frequency (Standard Musical Pitch)*, ISO 16:1975.