SYMBOLIC AND AUDIO KEY DETECTION BASED ON A HIDDEN MARKOV MODEL

Néstor Nápoles López
McGill University, CIRMMT
nestor.napoleslopez@mail.mcgill.ca

Claire Arthur
McGill University, CIRMMT
claire.arthur@mcgill.ca

Ichiro Fujinaga
McGill University, CIRMMT
ichiro.fujinaga@mcgill.ca

ABSTRACT

We introduce a model for the task of key detection based on a Hidden Markov Model (HMM). We assume a total number of 24 major and minor keys which are represented as a hidden state in the HMM. Additionally, 12 pitch-classes represent the possible observation symbols generated by a hidden state.

At first, the model is initialized with the same probability of starting in any of the 24 keys. In the following transitions, the probability distribution is computed from a geometric model of key-distance that outputs 9 groups of keys with respect to the current key, each group is less-likely to be the next key in the transition. In the case of the observation symbols, the probability distributions are obtained from well-known *key-profiles* used for key detection.

We have evaluated this model using a total of 260 pieces of music, these pieces consist mostly of classical music pieces but examples of popular music have also been included. Using the best combination of major and minor *key-profiles*, the model has shown an accuracy of 84.8% guessing the key of these pieces, replicating the same metrics used by the audio key detection task of the Music Information Retrieval Evaluation eXchange (MIREX).

1. INTRODUCTION

The task of key detection has interested the Music Information Retrieval (MIR) community for several years. At the same time, in its audio version, the key detection task has continuously attracted different participants of the Music Information Retrieval Evaluation eXchange (MIREX), since its first edition. There have been many attempts to solve the problem of detecting the key or tonality of a piece of music, however, some methodologies have prevailed in many of the proposed solutions to the problem. One of these methodologies is the design and use of *key-profiles*, which have been present in many, if not all, the proposed solutions to the problem. We present an algorithm that makes uses of *key-profiles* as the emission probability distributions used in a Hidden Markov Model. Additionally, a geometric model of key distance is used to design the transition probabilities of the model.

The original algorithm was designed to work with symbolic music files (e.g., midi, MusicXML, MEI, **kern). We present, however, an extension of the algorithm that can work with audio files by converting chromagrams into a sequence of *pitch-classes*.

2. MODEL

The model we present assumes a total of 24 major and minor keys, each of which corresponds to a hidden state. In the same manner, the model assumes a total of 12 *pitch classes*, each of which represents a possible observation emitted by one of the hidden states.

As we mentioned, the model has been originally designed to work with symbolic music files as input, however, an extension has been implemented allowing the model to work with audio. The difference between the two models lies mainly in the observation sequence serving as input and how it is obtained.

2.1 Symbolic Key Detection

The symbolic version of the algorithm receives a sequence of notes as input, in chronological order. When several notes sound simultaneously (e.g., during a chord), the order of the notes from lower to upper in the staff is preferred. The pitch height of each note in the sequence, together with its pitch-spelling (e.g., whether the note was f# or gb) is ignored by the model, keeping only its *pitch-class*.

2.2 Audio Key Detection

In the audio version, the sequences of notes have been replaced by a sequence of chroma vectors, computed for audio frames of the same size. These chroma vectors are discretized and their most prominent bins are turned into elements of a *pitch-class* sequence for that particular audio frame. Once this process has been completed for all the audio frames, the input of both algorithms—symbolic and audio—are transparent to the rest of the model, as they have a similar format, namely, a sequence of *pitch-classes*.

3. PARAMETERS OF THE MODEL

3.1 Initial probabilities

The initial probabilities of the model are the same for each key:

```
initial_probability(state) = 1.0 / 24.0
```

3.2 State transitions

The probability distributions for state transitions that happen after the initialization have been taken from a geometric model of key distance. This model considers *dominant* and *subdominant* keys (i.e., the *circle of fifths*) in the vertical axis and *parallel/relative* keys in the horizontal axis.

Using this model, all 24 keys can be divided into 9 groups, ordered by distance to a given key. The probability of a transition to another key in the next group decreases exponentially. For example, given a key of C Major, the 9 groups of keys look like this:

```
Group   Keys
0       C
1       G F a c
2       d e f g
3       D Eb A Bb
4       E Ab bb b
5       Db B
6       eb f#
7       c# ab
8       F#
```

The given key always conforms group 0 (i.e., a key is always most likely to transition to itself in the next observation than to any other key).

This structure of nine groups of keys repeats for all major and minor keys and can be implemented as a single vector of 24 elements that rotates depending on the current key.[1] Figure 1 shows a plot of the probability distributions when the current key is C Major.

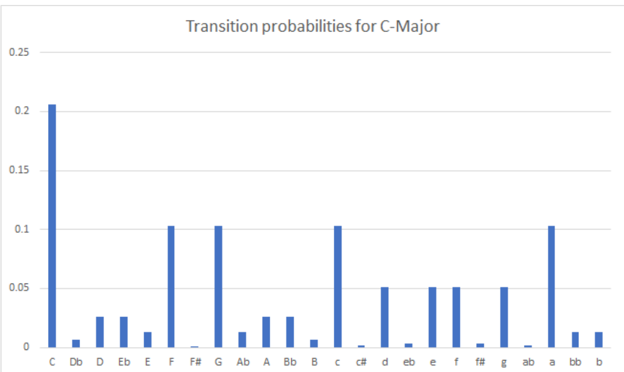

Figure 1. Probability distribution for the next state if the current state is C Major

[1] We make our implementation available here: https://github.com/napulen/justkeydding

3.3 Emission probabilities

The emission probability distributions are obtained from common key-profiles used by other key detection algorithms. Particularly, we considered the same five pairs of key-profiles used in the comparison by Albrecht, which are also listed in the documentation page of the *keycor* program developed by Craig Sapp.

4. DATASET

In order to test this algorithm, we have used a dataset that contains a total of 260 pieces. 164 classical music pieces as synthesized audio from midi with varied instrumentation (string quartets, wind quintets, symphonies, solo piano, sonatas for various instruments); In addition, there were 45 classical (solo cello, solo piano, string quartet) and 51 popular pieces (taken from McGill Billboard corpus) in the form of *real* audio recordings.

5. RESULTS

We compared the output of all the different combinations of major and minor *key-profiles* using the same evaluation metrics established by the MIREX audio key detection task. The best result was obtained by using the *simpleweights key-profile* from Craig Sapp, with a result of 84.8%.

6. *

References