2017-7

Key Inference from Irish Traditional Music Scores and Recordings

Pierre Beauguitte
Technological University Dublin, pierre.beauguitte@mydit.ie

Bryan Duggan
Technological University Dublin, bryan.duggan@tudublin.ie

John D. Kelleher
Technological University Dublin, john.d.kelleher@tudublin.ie

Recommended Citation

Beauguitte, P., Duggan, B. and Kelleher, J. (2017) Key inference from Irish traditional music scores and recordings. *14th Sound and Music Computing Conference, July 5-8, 2017, Espoo, Finland.*

KEY INFERENCE FROM IRISH TRADITIONAL MUSIC SCORES AND RECORDINGS

Pierre Beauguitte
Dublin Institute of Technology
pierre.beauguitte@mydit.ie

Bryan Duggan
Dublin Institute of Technology
bryan.duggan@dit.ie

John D. Kelleher
Dublin Institute of Technology
john.d.kelleher@dit.ie

ABSTRACT

The aim of this paper is to present techniques and results for identifying the key of Irish traditional music melodies, or *tunes*. Several corpora are used, consisting of both symbolic and audio representations. Monophonic and heterophonic recordings are present in the audio datasets. Some particularities of Irish traditional music are discussed, notably its modal nature. New key-profiles are defined, that are better suited to Irish music.

1. INTRODUCTION

Key detection is a common task in Music Information Retrieval (MIR), and has been a part of the Music Information Retrieval Evaluation eXchange (MIREX) since its start in 2005. Motivations for it include automatic analysis and annotations of large databases. The present study focuses on key identification for Irish traditional music tunes. Key-finding algorithms are tested on two collections of audio recording, of session and solo recordings, representing 636 tunes overall. Symbolic transcriptions have been compiled for all the tunes. A range of methods from the literature are benchmarked on both the audio and symbolic data. Some modifications to these methods, as well as new methods, including a set of parametric models, are presented and tested.

A musical key consists of a tonic note, represented by a pitch class (C, $C\#$, D...), and a mode, or rather mode family, which can be minor or major. Consequently, there are always 24 candidate keys, for the 12 semitones of the octave and the two considered modes. Enharmonic equivalence is used, which means that we do not distinguish between different spellings of the same note in the twelve-tone equal temperament, *e.g.* $D\#$ and Eb. Throughout the paper we will adopt the convention of denoting major keys by upper-case letters, and minor keys as lower-case ones.

The standard approach to identifying keys in a musical piece is to use key-profiles [1]. They can be seen as vectors assigning weights to the twelve semitones, denoted $(p[i])_{i=0,...,11}$. One key-profile per mode is defined for the tonic note C, and transposition to the another tonic note is performed by rotating the elements in the vector. A histogram $(h[i])_{i=0,...,11}$ of cumulative durations of each pitch class in the musical excerpt is generated, and the score is the weighted sum of the histogram with the key-profile.

$$s(p,h) = \sum_{i=0}^{11} p[i] * h[i] \quad (1)$$

The estimated key is then the one corresponding to the highest scoring profile:

$$key(h) = key(\arg\max_{p \in \mathcal{P}} s(p,h)) \quad (2)$$

where $\mathcal{P}$ is the set of 24 key-profiles representing candidate keys.

This method only needs a pitch class histogram from the musical excerpt. From a symbolic representation, obtaining this is straightforward. From an audio representation, an extra step of computing a chromagram is required, which does not represent any significant difficulties. It can however add some noise to the histogram because of the harmonics present in an audio signal. The resulting pitch class histogram is, in the classification proposed in [2], a low-level global descriptor.

Other methods for key-identification are based on higher-level features. For example, in [3] the intervals of a melody are analyzed, which presupposes that an automatic transcription of the signal has been performed beforehand. In [4], an HMM is trained to estimate the key from a sequence of chords.

In preparing this paper we also experimented with some machine learning (ML) models for key-detection (such as multinomical regression models). Generally, ML models perform best on relatively balanced datsets; consequently, in order to train our ML models we introduced transposed tunes into the dataset in order to balance the distribution in the data. However, the performance of these ML models was relatively weak and as a result we do not include a description of them nor the data preparation carried out for them in this paper.

The main contribution of this paper is a set of new key-profiles, outperforming existing keys-profiles on the corpora considered. In Section 2, we present the key-finding method, and existing key-profiles defined in the literature. In Section 3, new key-profiles are introduced. Section 4 gives a description of the datasets. Section 5 gives the details and results of the experiment using the profiles. Sec-

tion 6 presents a parametric extension of one of the models, the method used to select the parameters, and the performance of this model. Finally Section 7 discusses the study and indicates future ideas for research.

2. RELATED WORK

2.1 Triads

Certainly the most naive way to define a key-profile is to consider only the triad of the tonic chord. For example, in C it is expected that the pitch classes of the tonic C, the third E and the fifth G will be the most frequent, as reflected in the key-profiles:

$$p_{\text{triad}}(C) = [1,0,0,0,1,0,0,1,0,0,0,0]$$
$$p_{\text{triad}}(c) = [1,0,0,1,0,0,0,1,0,0,0,0]$$

2.2 Krumhansl-Kessler

The key-profiles established in [5] were obtained by perceptual experiments, in contrast with the triads presented above which were motivated by musical theory. Subjects were asked to rate how well the different pitch classes fit with a short musical excerpt establishing a key. The Krumhansl-Kessler key-profiles are a well known method for key detection.

$$p_{\text{KK}}(C) = [6.35, 2.23, 3.48, 2.33, 4.38, 4.09, 2.52, 5.19, 2.39, 3.66, 2.29, 2.88]$$
$$p_{\text{KK}}(c) = [6.33, 2.68, 3.52, 5.38, 2.60, 3.53, 2.54, 4.75, 3.98, 2.69, 3.34, 3.17]$$

2.3 Lerdahl's Basic Spaces

The "basic spaces" defined by Lerdahl in [6] are derived from the diatonic scale of each key. Different weights are given to the degrees of the scale: 5 for the tonic (index 0 for C and c), 4 for the fifth (index 7 for C and c), 3 for the third (index 4 for C, 3 for c), 2 to the rest of the diatonic scale, and 1 to the remaining semitones.

$$p_{\text{Lerdahl}}(C) = [5,1,2,1,3,2,1,4,1,2,1,2]$$
$$p_{\text{Lerdahl}}(c) = [5,1,2,3,1,2,1,4,2,1,2,1]$$

It is worth noting that the natural minor scale, or Aeolian scale, is considered here: the natural seventh (index 10) is taken as part of the scale, not the augmented seventh (index 11) as would be the case with the harmonic minor scale.

2.4 Leman's Tone Center Images

In [7], the *simple residue image* (or R-image) of a chord is generated as a weighted combination of the undertone series of the tonic. The *tone center images* are then derived by summing the R-images of the chords present in the common cadences. The three typical cadences selected in [7] are

$$\begin{cases} \text{I} & \text{IV} & \text{V} & \text{I} \\ \text{I} & \text{II} & \text{V} & \text{I} \\ \text{I} & \text{VI} & \text{V} & \text{I} \end{cases}$$

where the type of the chord the depends on the scale considered. In the major case, the classic major scale (Ionian) is used. However in the minor case, the harmonic scale is chosen, where the seventh degree is one semitone higher than in the natural scale.

The Tone Center Images (TCI) are then obtained by summing the R-images of the chords, weighted by how often they occur in the cadences, and normalizing:

$$6 * \text{I} + 3 * \text{V} + \text{II} + \text{IV} + \text{VI}$$

After normalization, the key-profiles obtained are:

$$p_{\text{Leman}}(C) = [0.36, 0.05, 0.21, 0.08, 0.24, 0.21, 0.05, 0.31, 0.07, 0.24, 0.09, 0.10]$$
$$p_{\text{Leman}}(c) = [0.34, 0.11, 0.15, 0.25, 0.11, 0.25, 0.02, 0.31, 0.24, 0.09, 0.12, 0.14]$$

3. NEW PROFILES

In this section two new pairs of key-profile are introduced.

3.1 Basic spaces for modal scales

The key-profiles introduced in 2.3 are based on the natural scales, or Ionian mode for major and Aeolian mode for minor. In Irish music, two other modes are commonly used: the Mixolydian mode (major with a minor seventh) and the Dorian mode (minor with a major sixth). The two basic spaces are modified so that they are suited to both major modes (Ionian and Mixolydian) and minor modes (Aeolian and Dorian). This is done by setting $p[10] = p[11]$ in the major case, and $p[9] = p[8]$ in the minor case:

$$p_{\text{Lerdahl}^\star}(C) = [5,1,2,1,3,2,1,4,1,2,2,2]$$
$$p_{\text{Lerdahl}^\star}(c) = [5,1,2,3,1,2,1,4,2,2,2,1]$$

This idea of considering the minor and major seventh as equivalent has already been used for the task of Irish traditional music tune transcription and recognition in [8].

3.2 Cadences

These key-profiles are inspired by Leman's tone center images. As mentioned in 2.4, cadences play an important role in establishing a tonal center. In Irish traditional music, the most commmon cadence is I - IV - V - I [9]. In the case of minor tunes, we also consider the chord sequence VII - VII - I - I, often used in accompaniments. Consequently the formulae to obtain the key-profiles are

$$\begin{cases} \text{Major:} & 2 * \text{I} + \text{IV} + \text{V} \\ \text{Minor:} & 4 * \text{I} + 2 * \text{VII} + \text{IV} + \text{V} \end{cases}$$

Instead of considering R-images, chords are simply represented by their triad, as introduced in 2.1. The resulting key-profiles are:

$$p_{\text{Cadences}}(C) = [3,0,1,0,2,1,0,3,0,1,0,1]$$
$$p_{\text{Cadences}}(c) = [5,0,3,4,0,3,0,5,1,0,3,0]$$

Finally, these profiles are also modified to account for the Mixolydian and Dorian modes:

$$p_{\text{Cadences}^\star}(C) = [3, 0, 1, 0, 2, 1, 0, 3, 0, 1, 1, 1]$$
$$p_{\text{Cadences}^\star}(c) = [5, 0, 3, 4, 0, 3, 0, 5, 1, 1, 3, 0]$$

4. DATASETS

This section introduces the datasets used for this study.

4.1 Audio datasets

Two sets of recordings are used in this study, representing overall 636 audio items. In both cases, each tune was annotated with key information by the first author.

4.1.1 Foinn Seisiún

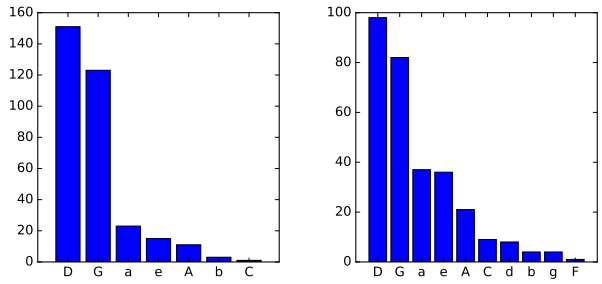
This collection consists of session recordings accompanying the Foinn Seisiún books published by the Comhaltas Ceoltóirí Éireann organisation. Instruments in the recordings include flute, tin whistle, uillean pipes (Irish bagpipes), accordion, concertina, banjo, piano, guitar, bodhran (drum). They offer good quality, homogeneous examples of the heterophony inherent to an Irish traditional music session. The whole collection consists of 3 CDs, representing 327 tunes. The first 2 CDs (273 tunes) are available under a Creative Commons Licence, while the third is commercially available. In five instances, two recordings of a same tune are present. In four cases, we decide to keep both as different items in our dataset, since the set of instruments recorded is different. Only in one case is the exact same recording present, in which case we discard one of the recordings. In the end, this dataset contains 326 distinct recordings, and is denoted FS_{audio} in the rest of the article.

4.1.2 Grey Larsen's 300 gems

Grey Larsen's recordings, accompanying the book *300 Gems of Irish Music for All Instruments*, is a set of MP3 files commercially available. They consist of studio quality recordings of tunes played on Irish flute, tin and low whistles, and anglo concertina. None of the 300 audio recordings are of the same tune. This dataset is denoted GL_{audio} in the rest of the article.

4.2 Symbolic datasets

For each tune in both audio corpora, a symbolic transcription was collected in ABC format. The majority of the transcriptions were found online, mostly on the collaborative website `www.thesession.org`. A small number of tunes were not available, in whose cases the audio recordings were manually transcribed to ABC by the first author. It is important to note that the symbolic transcriptions do not correspond exactly to the music played in the audio recording. Indeed Irish music is always interpreted with ornaments and small variations. The score is rather seen as an outline of the melody to be played. The difference between recordings and scores is even more clear for the session recordings: the audio signal is then heterophonic, as the different musicians are not playing exactly the same melody.



(a) Keys in *FS* (b) Keys in *GL*

Figure 1. Distributions of keys in *FS* and *GL*

This time, all redundant copies of duplicate tunes present in the Foinn Seisiún collection are discarded, as in such cases the score remains the same even though the recording differs. Hence FS_{symb} contains 322 items, and GL_{symb} 300.

4.3 Distribution of keys

Both datasets are unbalanced in terms of key distribution, as can be seen on Figure 1. These distributions are actually quite representative of the reality of how Irish music is played. The keys of *D* and *G* are indeed the most common in sessions, in part due to the fact that some instruments are limited to these scales (*e.g.* keyless flute, whistle, uileann pipes, ...).

5. EXPERIMENT 1

5.1 Pitch class histograms extraction

The first step of the algorithm is to extract a pitch class histogram from the musical piece. In the case of symbolic representation, this poses no difficulty. The software `abc2midi` [1] was used to parse the ABC files.

For each audio recording, a chromagram was first generated, then the chroma vectors were summed over time. Chromagrams were obtained using the `madmom` [2] library. Several methods of computing the chromas were tested: standard pitch class profile, harmonic pitch class profile [2], and Deep Chroma extractor [10]. This last method, using a deep neural network trained to extract chromas from a spectrogram, consistently outperformed the others. Consequently all the results reported below are obtained with the Deep Chroma method.

5.2 Key search

For each of the key-profile sets, the first step is to normalize the profile:

$$\overline{p}[i] = \frac{p[i]}{\sum_{i=0}^{11} p[i]}$$

This is important for cases where the major and minor profiles do not have the same sum, *e.g.* Krumhansl-Kessler profiles, or the ones based on cadences.

[1] `http://abc.sourceforge.net/abcMIDI/`
[2] `http://madmom.readthedocs.io`

	FS_{audio}	FS_{symb}	GL_{audio}	GL_{symb}
Triad	0.873	0.795	0.671	0.709
KK	0.854	0.869	0.665	0.696
Lerdahl	0.887	**0.890**	0.689	0.777
Leman	0.848	0.829	0.646	0.666
Lerdahl*	0.893	**0.890**	0.711	**0.798**
Cadences	0.883	0.874	0.677	0.770
Cadences*	**0.902**	0.818	**0.713**	0.750

Table 1: MIREX scores for the 4 corpora using different key-profiles

For a musical piece having the pitch class histogram h, the estimated key is then obtained by:

- for each of the 24 normalized candidate key-profiles $\overline{p}$, compute the score $s(\overline{p}, h)$ (Equation (1))
- choose the key whose profile yields the highest score (Equation (2))

The key finding algorithm is unbiased, in the sense that the 24 key-profiles are evaluated in the same way, with no preference for the common keys. Hence the fact that the keys are unbalanced in the datasets as shown in Figure 1 is not an issue for this study.

5.3 Evaluation metrics

The MIREX evaluation metrics is defined as follows: let k be the ground truth annotation, and $\hat{k}$ the estimated key, then the accuracy score for this item is:

$$acc = \begin{cases} 1 & \text{if } k = \hat{k} \\ 0.5 & \text{if } \hat{k} \text{ is the perfect fifth of } k \\ 0.3 & \text{if } \hat{k} \text{ is the relative of } k \\ 0.2 & \text{if } \hat{k} \text{ is the parallel of } k \\ 0 & \text{otherwise} \end{cases}$$

These scores are then averaged across the whole dataset.

5.4 Results

Results are given for the seven pairs of key-profiles considered. All the MIREX accuracy scores are reported in Table 1. The * superscript indicates the modal versions of the key-profiles presented in Section 3.

Two observations can be made from this table. First, comparing the MIREX scores on the two symbolic datasets shows that inferring the key of the tunes in *GL* is harder than in *FS*. Second, on the *FS* collection, most key-profiles yield better results on the audio data than on the symbolic data. The opposite is true for *GL*. Hence it appears that inferring keys from heterophonic or polyphonic audio is easier than on monophonic recordings. An explanation for this is that the harmonic content is richer in heterophonic and polyphonic signals.

The new key-profiles introduced in Section 3 (Lerdahl*, Cadences and Cadences*) outperform the existing key-profiles on all four datasets. Inspired by Lerdahl's original key-profiles which assign different weights to degrees of the scale (see Section 2.3) we believe that the performance of the Cadences key-profiles can also be improved by introducing weights. Specifically by assigning different weights to the tonic, third and fifth degrees in the triads uses to build the Cadence profiles presented in Section 3.2. To test this hypothesis we ran a second experiment which is described in Section 6.

6. EXPERIMENT 2

6.1 Weighted cadences

The model proposed here is a parameterized version of the previously introduced Cadences profiles. The parameters considered are the three weights given to the three notes of the triads, denoted $W = (w_1, w_3, w_5)$ for the tonic, third and fifth respectively. Then, the following profiles can be derived from the cadences chosen in Section 3.2:

$$\begin{aligned} p_{\text{Cadences}(W)}(C) = [&2w_1 + w_5, 0, w_5, 0, 2w_3, w_1, 0, \\ &w_1 + 2w_5, 0, w_3, 0, w_3] \\ p_{\text{Cadences}(W)}(c) = [&4w_1 + w_5, 0, 2w_3 + w_5, 4w_3, \\ &0, w_1 + 2w_5, 0, w_1 + 4w_5, \\ &w_3, 0, 2w_1 + w_3, 0] \end{aligned}$$

The modal versions of these profiles, Cadences*(W) are obtained in the same manner as in Section 3.

6.2 Experimental Methodology

Generally, the goal of a model evaluation experiment on a dataset is to estimate the performance of the model on unseen data (i.e, data that was not used to train the model). To achieve this it is traditional in machine learning to first separate a dataset into a training set and test set. The training set is then used to train and compare models in order to choose a single best model and the test set is solely used to evaluate the best model as judged based on the relative performance of models on the training set. The reason for this is that if the performance of models on the test set is used to select the best model then the test set is actually part of training the model. In other words the same data cannot be used to select the best model and to evaluate its performance. In machine learning, using the performance of a model on the test set to select the best model is known as "peeking" at the test set. It is equivalent to allowing the model to look at the test set prior to running the test which is problematic because it can result in optimistic performance scores.

Returning to the concerns of the current paper each set of weights applied to the weighted cadences key-profiles defines a separate key-detection model. Using a grid-search process we can iterate across a grid of model parameters with each point on the grid defining a separate set of weights (and hence a distinct model). The grid-search process provides a mechanism where we can iterate through a set of models and test each model in turn. The problem with this methodology, however, is that if we select the best set of weights (or model) by iterating across the grid and simply

selecting the set of weights with the highest performance on the datasets this model selection methodology suffers from the problem of "peeking" introduced in the preceding paragraph. In other words we will find a single best-set of weights on the dataset but the performance of this model on the dataset will not be a realistic measure of the model on unseen data. We will refer to this methodology as the *Best-Weights* method.

An alternative methodology is to use a process called 10-fold cross validation, following the methodology of [11]. The focus of a 10-fold cross-validation process is to estimate the average performance of the models generated by a machine learning algorithm and hyper-parameter set [3] on unseen data. In a 10-fold cross-validation a dataset is split into 10 equally size subsets, or *folds*. Then 10 experiments are run (one per fold). In each experiment 9/10s of the data is used to train a model and the remaining 1/10 (the fold) is used to evaluate the model performance. The overall performance of algorithm and hyper-parameters is then calculated by aggregating the confusion matrices generated by each of the 10 experiments and calculating a performance metric on the resulting matrix. The advantage of a cross-validation methodology is that in each of the 10 experiments distinct data is used to train and test the model. So the final overall accuracy score is representative of the likely performance of a model trained with the tested algorithm on unseen data.

In our context training a model involves selecting the set of weights that perform best on a dataset. So, to utilize a cross-validation methodology to evaluate the weighted cadences approach we need to run a grid-search process in each of the 10 experiments [4] and select the best set of weights for that experiment based on the performance on the 9/10s training portion of the data and then evaluate the performance of these best weights on the (unseen) remaining 1/10 of the data. The advantage of the cross validation methodology is that it provides us with an estimate of the likely performance of a weighted cadences key-profiles on unseen data. This is because in each of the 10 weight tuning and model evaluation experiments (one experiment per fold) distinct sets of tunes are used for weight tuning and evaluation. Consequently, the accuracy scores returned from this process reflect the accuracies of models on unseen tunes (*i.e.* tunes that were not seen during weight tuning). The drawback of this approach, however, is that the weight tuning process in each of these experiments may return different sets of weights. So, although the overall accuracy scores provides an indication of likely performance of a set of weighted cadences key-profiles on unseen data it does not provide an accuracy measure for cadences key-profiles with a specific (fixed) set of weights.

As a result, we decided to use apply both methodologies to the weighted cadences key-profiles. First we applied a cross-validation process to estimate the performance of weighted cadences key-profiles on unseen data. This first step also serves to determine the best grid size to use. We then applied the *Best-Weights* method introduced above, with the chosen grid size, to find the best single set of weights on our dataset.

[3] Hyper-parameters are parameters on a machine learning algorithms as distinct to parameters on a model.

[4] As opposed to the single grid search process that would be used in the *Best-Weights* method.

	FS_{audio}	FS_{symb}	GL_{audio}	GL_{symb}
Cadences(W)	0.891	0.879	0.702	0.769
Cadences$^\star(W)$	**0.908**	0.840	**0.720**	0.745

Table 2: MIREX scores computed from the aggregate matrices after cross-validation on the 4 corpora

6.3 Results

The only hyper-parameter in this experiment is g, the width of the grid. A wide range allows a better fit on the training data, but poses a risk of overfitting it, resulting in poor performance on the test sets. The experiment was performed for g ranging from 2 to 10. The grid size $g = 3$, allowing the weights w_i to take values in $[1, 2, 3]$, gave the best results, and is used for the following results.

Scores calculated from the aggregate matrices after the cross validation on each of the four datasets are presented in Table 2. The models Cadences$^\star(W)$ outperform all other methods on the two audio corpora. However, the Lerdahl* key-profiles evaluated in Experiment 1 remain the best performing ones on the symbolic data. Consequently the rest of this section focuses on Cadences$^\star(W)$ on the audio datasets.

The result of the cross-validation method suggests that the models Cadences$^\star(W)$ generalize well to unseen audio data. In order to obtain one single model (as opposed to the multiple ones resulting from the 10 folds), the *Best-Weights* method was then performed on the combined dataset $(FS + GL)_{audio}$. Grouping the two collections of audio recordings means that the profiles should perform well on both heterophonic and monophonic recordings. The weights obtained are $(3, 1, 2)$, corresponding to the intuition that the tonic and fifth are more important than the third, as in Section 2.3. The resulting key-profiles are:

$$p_{\text{Cadences}^\star(3,1,2)}(C) = [8, 0, 2, 0, 2, 3, 0, 7, 0, 1, 1, 1]$$
$$p_{\text{Cadences}^\star(3,1,2)}(c) = [14, 0, 4, 4, 0, 7, 0, 11, 1, 1, 7, 0]$$

With these profiles, the MIREX scores are 0.901 on FS_{audio} and 0.730 on GL_{audio}, to be compared to the scores in Table 1. The lower score on *FS* is not unexpected: the grid search maximizes the overall score accross the combined audio collection, regardless of the scores on the individual collections. The overall MIREX score on the combined collection is 0.819, compared to 0.811 with the non-parametric Cadences* profiles.

The confusion matrices for these new key-profiles on the audio collections, and for the Lerdahl* ones on the symbolic datasets (on which they are still the highest scoring ones), are given in Tables 3 to 6. Rows indicate the actual keys in the ground truth annotations, while columns indicate estimated keys. Keys that never occur in either the ground truth or the estimations are omitted.

	D	G	a	e	A	b	C	d
D	149	1	0	0	0	0	0	0
G	4	119	0	0	0	0	0	0
a	5	9	7	0	0	0	1	1
e	8	3	1	3	0	0	0	0
A	2	0	0	0	9	0	0	0
b	2	0	0	0	0	1	0	0
C	0	0	0	0	0	0	1	0

Correct	289
Fifth	6
Relative	6
Parallel	0
Neighbour	17
Other	8

Table 3: Confusion matrix for FS_{audio} with key-profiles Cadences*(3, 1, 2)

	D	G	a	e	A	b	C	f#	B
D	131	3	1	0	0	14	0	1	0
G	0	113	1	4	0	1	0	0	0
a	0	7	11	1	4	0	0	0	0
e	1	1	0	11	0	2	0	0	0
A	0	0	0	0	11	0	0	0	0
b	0	0	0	0	0	2	0	0	1
C	0	0	0	0	0	0	1	0	0

Correct	280
Fifth	0
Relative	19
Parallel	5
Neighbour	9
Other	9

Table 4: Confusion matrix for FS_{symb} with key-profiles Lerdahl*

	D	G	a	e	A	C	d	b	g	F
D	82	4	6	1	1	0	4	0	0	0
G	6	71	4	1	0	0	0	0	0	0
a	5	9	17	0	1	3	2	0	0	0
e	10	6	5	13	2	0	0	0	0	0
A	7	0	2	0	11	0	1	0	0	0
C	0	1	2	0	0	5	1	0	0	0
d	1	0	2	0	0	2	3	0	0	0
b	3	0	0	1	0	0	0	0	0	0
g	0	0	0	0	0	2	0	0	2	0
F	0	0	0	0	0	0	0	0	0	1

Correct	205
Fifth	16
Relative	15
Parallel	8
Neighbour	27
Other	29

Table 5: Confusion matrix for GL_{audio} with key-profiles Cadences*(3, 1, 2)

	D	G	a	e	A	C	d	b	g	F	f#
D	85	1	5	1	0	0	0	6	0	0	0
G	2	71	1	7	0	0	0	1	0	0	0
a	1	13	18	3	1	1	0	0	0	0	0
e	3	1	0	26	1	0	0	5	0	0	0
A	4	0	1	0	14	0	0	0	0	0	2
C	0	1	4	0	0	4	0	0	0	0	0
d	0	0	1	0	0	1	6	0	0	0	0
b	0	0	0	0	0	0	0	4	0	0	0
g	0	0	0	0	0	0	0	0	3	1	0
F	0	0	0	0	0	0	1	0	0	0	0

Correct	231
Fifth	3
Relative	22
Parallel	2
Neighbour	20
Other	22

Table 6: Confusion matrix for GL_{symb} with key-profiles Lerdahl*

The three types of errors taken into account in the MIREX evaluation metrics are highlighted in different shades of blue. Another error occurs frequently in this experiment, named "neighbour". The "neighbour" relationship is defined as follows:

> two keys are neighbours if one is major, the other minor, and the minor one has a tonic one tone above the tonic of the major one.

To take a concrete example, D and e are neighbour keys. In terms of modes, the scales of D Mixolydian and of e Aeolian contain the exact same pitch classes. It is not rare in Irish music that a tune labelled as e minor changes its "tonic center" for a few bars to the neighbour key D, *e.g.* the well known Cooley's reel. On both audio datasets, this type of error is the most common. As such, and although the MIREX evaluation metrics does not take these errors into account, reporting them is relevant.

Relative keys are the next most common errors, on both audio and symbolic datasets. The scales of two relative keys contain, as is the case with neighbour keys, the same pitch classes, if one considers the Aeolian mode. Changes of tonic center in a tune between its key and the relative key are also quite common in Irish music. The high frequencies of these two types of errors can be explained by the specific characteristics of Irish traditional music.

Table 7 gives the percentages of correctly inferred keys per mode. A clear difference in performance appears between the major and minor keys, suggesting that minor keys are harder to detect than major keys.

	FS_{audio}	FS_{symb}	GL_{audio}	GL_{symb}
Major	97.5%	91.1%	80.6%	82.5%
Minor	26.8%	58.5%	39.3%	64.0%

Table 7: Proportions of correct inference per mode

7. CONCLUSION AND FUTURE WORK

In this paper, a range of existing key detection algorithms was tested on datasets of audio and symbolic Irish music. Modifications of these models, and a new set of key-profiles, were introduced, which improved the performance on both types of representation. Error analysis showed that the most common confusions were between relative and neighbour keys, which reflects some specific characteristics of Irish music.

As stated in [9], it is sometimes difficult to pinpoint the key of an Irish tune. The key annotations on the datasets were made by the first author, and it is possible that other annotators could annotate some of the tunes differently. A way to quantify these ambiguities will be to gather annotations from other experienced musicians in order to obtain a Cohen's kappa coefficient. It is expected that most errors made by the key-matching algorithms will be on tunes where annotators disagree.

8. REFERENCES

[1] D. Temperley, *The cognition of basic musical structures*. Cambridge, Mass: MIT Press, 2001.

[2] E. Gómez, "Tonal Description of Polyphonic Audio for Music Content Processing," *INFORMS J. on Computing*, vol. 18, no. 3, pp. 294–304, Jan. 2006.

[3] S. T. Madsen, G. Widmer, and J. Kepler, "Key-Finding with Interval Profiles." in *ICMC*, 2007.

[4] K. Noland and M. B. Sandler, "Key Estimation Using a Hidden Markov Model." in *Proceedings of the 7th International Society for Music Information Retrieval Conference*, 2006, pp. 121–126.

[5] C. L. Krumhansl, *Cognitive Foundations of Musical Pitch*, ser. Oxford Psychology Series. Oxford, New York: Oxford University Press, 1990.

[6] F. Lerdahl, "Tonal Pitch Space," *Music Perception: An Interdisciplinary Journal*, vol. 5, no. 3, pp. 315–349, Apr. 1988.

[7] M. Leman, *Music and Schema Theory*, ser. Springer Series in Information Sciences, T. S. Huang, T. Kohonen, M. R. Schroeder, and H. K. V. Lotsch, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1995, vol. 31, dOI: 10.1007/978-3-642-85213-8.

[8] B. Duggan, "Machine annotation of traditional Irish dance music," Ph.D. dissertation, Dublin Institute of Technology, 2009.

[9] F. Vallely, Ed., *The Companion to Irish Traditional Music*, second edition ed. Cork: Cork University Press, 2011.

[10] F. Korzeniowski and G. Widmer, "Feature learning for chord recognition: the deep chroma extractor," in *Proceedings of the 17th International Society for Music Information Retrieval Conference*, New York, USA, 2016.

[11] J. D. Kelleher, B. Mac Namee, and A. D'Arcy, *Fundamentals of machine learning for predictive data analytics: algorithms, worked examples, and case studies*. Cambridge, Massachusetts: The MIT Press, 2015.