2008-01-01

A system for automatically annotating traditional Irish music field recordings

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

Brendan O'Shea
Dublin Institute of Technology, brendan.oshea@comp.dit.ie

Padraig Cunningham
University College Dublin

Recommended Citation

Duggn, Brian and O'Shea, Brendan and Cunningham, Padraig: A system for automatically annotating traditional Irish music field recordings. 6th. International Workshop on Content-Based Multimedia Indexing, Queen Mary University of London, England, June, 2008.

A SYSTEM FOR AUTOMATICALLY ANNOTATING TRADITIONAL IRISH MUSIC FIELD RECORDINGS

Bryan Duggan, Brendan O' Shea

DIT School of Computing

Kevin St

Dublin 8

Ireland

{bryan.duggan, brendan.oshea}@comp.dit.ie

Pádraig Cunningham

School of Informatics and Computer Science

UCD

Dublin

Ireland

padraig.cunningham@ucd.ie

ABSTRACT

This paper presents MATT2 *(Machine Annotation of Traditional Tunes)*. MATT2 is a novel system which can automatically annotate field recordings of traditional Irish music with useful metadata such as tune name, key signature, time signature, composer and discography. MATT2 works by using a number of algorithms to automatically transcribe digital audio to be annotated to the ABC music notation language. It then compares these transcriptions against a corpus of 860 human made transcriptions in ABC using a variation of the edit distance algorithm. Results using MATT2 to annotate fifty recordings of flute and fiddle tunes demonstrate a high success rate at annotating recordings made by different musicians. Additionally, several of the recordings successfully annotated in testing MATT2 were recorded in imperfect conditions, with badly degraded audio.

1. INTRODUCTION

Repertoire in Irish traditional music is primarily acquired orally. Musicians playing Irish music learn tunes by hearing the tune played by fellow musicians in sessions, classes, workshops and from commercial recordings [1]. It is common at workshops such as those held as part of the Willie Clancy Summer School [2] for students to use electronic devices to record their classes. Increasingly students use digital audio field recorders such as the M-Audio Micro Track II, which record high quality audio directly to WAV or MP3 format [3]. In this way, over the years musicians can acquire many hours of field recordings in standard audio formats. Similarly, organisations such as Na Píobairí Uilleann, Comhaltas Ceoltóirí Éireann and the Irish Traditional Music Archive have been acquiring field recordings of traditional music for over sixty years and these organisations now possess many thousands of hours of recordings in a variety of formats and on a variety of different media [5,4].

In order for these archives to be usefully searched, they must be annotated with appropriate metadata, such as tune names, time signatures, key signatures and instruments. Additionally for musicological and ethnographic study, archives could be annotated with stylistic metadata. This is the main goal of the MATT2 project and this paper presents a system for automatically annotating field recordings of monophonic traditional dance music. First, using a number of DSP (Digital Signal Processing) algorithms, tunes are transcribed to the ABC music notation language. Once a transcription is made, the system compares it against a corpus of 860 human made tune transcriptions. The ABC language has the advantage of being based on ASCII text and so tunes in ABC can be easily processed and analysed using algorithms for textual information retrieval. Using this approach, a high success rate for both long and short phrases of music is reported. Section 2 of this paper presents a brief discussion of traditional music archiving. Section 3 presents related work on the use of ABC and on MIR (Music Information Retrieval) systems in general. Section 4 describes the algorithms used in implementing MATT2. Section 5 presents experimental results which establish the effectiveness of MATT2 by annotating fifty pieces of input audio recorded in imperfect conditions. Section 6 presents conclusions and future work.

2. BACKGROUND

Current estimates suggest there are at least seven thousand traditional tunes in existence [6,1,7]. It is proposed that in the past, many isolated rural communities in Ireland developed their own repertoire of tunes and that widespread knowledge of a common repertoire did not occur until the publication of catalogues of traditional tunes such as O'Neill's The Music of Ireland in 1903 [8,9]. In his seminal work, O'Neill collected 1850 tunes played by emigrant Irish musicians in Chicago in the latter part of the nineteenth century.

In 1991, the ABC music notation language was introduced by Chris Walshaw [10]. The format was

designed primarily for folk and traditional tunes which can be written on one stave in standard classical notation. ABC files are ASCII text files and so can be edited by any text editor, without the necessity for special software. Each file (known as a *tune book*) can contain multiple tunes. There is an active Internet community engaged in the transcription of tunes in ABC format. For example, the website thesession.org has over seven thousand tunes entered by the traditional music community [7].

The tune given in Figure 1 is typical of the transcriptions that can be sourced in ABC from publicly available databases. In this transcription the transcriber has included a significant amount of useful metadata with the notation for the tune such as the source of the transcription, the discography and a listing of similar tunes.

```
X:422
T:Come West Along the Road
R:reel
S:Session
H:See also #432, in A. This version is
also played in A.
H:1st part similar to "Over the Moor to
Peggy", #710
D:Arcady: Many Happy Returns
D:Noel Hill & Tony McMahon: \'I gCnoc na
Gra\'i
Z:id:hn-reel-422
M:C|
K:G
d2BG dGBG|~G2Bd efge|d2BG dGBG|1 ABcd
edBc:|2 ABcd edBd||
|:g2bg egdg|(3efg dg edBd|1 g2bg
egdB|ABcd edBd:|2 gabg efge|dega bage||
```

Figure 1: The tune "Come West Along the Road" in the ABC format [11].

The body encoding supports such features as ornaments, bar divisions, sharps, flats, naturals, repeated sections, key changes, guitar chords, lyrics and variations.

Between 1997 and 2000, a group of musicians under the leadership of Dan Beimborn and John Chambers, undertook a grass roots project to transcribe three of O'Neill's books to electronic format using the ABC music notation language. As copyright had expired on O'Neill's original books, they made their work freely available on the Internet [11].

Many of the tunes from O'Neill's books are played differently by musicians today, as is normal with a living tradition. Around the same period (the late 1990's) Henrik Norbeck collected nearly two thousand tunes in ABC format from various sessions and recordings. This collection contains many modern settings of tunes from O'Neill's books [12]. The MATT2 system uses a corpus of reels in ABC format drawn from Henrik Norbeck's transcriptions.

2. RELATED WORK

In related work, ABC transcriptions have been used to encourage creativity in traditional music sessions and to facilitate e-learning in non traditional settings [13-15] by making available thousands of tunes in ABC format on a mobile device. To achieve this, TunePal was developed. TunePal runs on a variety of handheld computing devices running Symbian OS and the Windows Mobile operating system. TunePal supports the retrieval of tunes by text searches and playback of ABC files and is in use by many hundreds of traditional musicians [13-15]. ABC transcriptions have also been used in two computational creativity applications [16,17]. To develop MATT1 (Machine Learning Articulation of Traditional Tunes) detailed transcriptions were made of the playing of flute player and maker Eamonn Cotter. These were used to train a machine learning system to *ornament* traditional music [18,9,19,20] in the style of the musician. In [17] a *computational creativity* algorithm is presented that can compose new reels by analysing the structure and n-gram note sequences present in a corpus of reels. A group of domain experts then evaluated the generated tunes for aesthetic value and correctness.

The application proposed in this paper is an MIR system. These systems convert melodies to sequences of symbols that can be matched against query patterns using methods from general string matching. To overcome transcription errors several applications use a pitch contour representation (up, down, or repeat) to give a representation of the direction of the pitch intervals [23,22,21]. Many systems also use hummed or sung queries as input [22,24,25]. Hu & Dannenberg [24] compare a number of approaches to MIR and conclude that MIR systems must deal with many difficult problems, including robustness in the face of transcription errors, transposition invariance and tempo variance (*rubatto*). Algorithms must also allow searching for substrings within an overall melody. Widmer *et al*. [27] similarly classify the problem of feature extraction from digital signals for MIR systems as error prone and suggest that systems should be developed that model higher level models of music cognition that human listeners apply when listening to music. They describe *GenreCrawler*, an alternative to DSP approaches that uses web mining as a technique for MIR. MIR in traditional Irish music has an additional difficulty in that traditional musicians rarely play tunes as transcribed in books. In fact it is reported that a good traditional musician will almost never play a tune twice, identically [18,9,19,28,29].

Interestingly, Adams *et al.* [26] suggest that coarsely quantised melodic representations such as simplified pitch contour representations do not improve retrieval performance for query-by-humming systems. Also we suggest that humming is an inefficient query mechanism for the problem being addressed by this research. A far better approach is to allow queries to be generated directly from

the playing of traditional instruments. While there are MIR systems that allow users to search for traditional Irish dance tunes using text based musical queries [7,15] and there are MIR systems that allow users to search for melodies using sung or hummed queries [23,25], there are no MIR systems that we are aware of that allow musicians to search for traditional Irish dance tunes using queries played on traditional instruments. Some examples of the above include the website thesession.org [7] which contains an extensive collection of over seven thousand traditional dance tunes in the ABC language. The session.org supports text queries by any of the metadata associated with a tune or melodic queries in the ABC language. Similarly, Melodyhound [23] a publicly accessible MIR system that supports sung queries and contains a large collection of traditional Irish dance tunes does not generate positive results when queries are presented in the form of melodies played on the tin-whistle or wooden flute.

Despite the difficulties outlined, in experiments, MATT2 demonstrates robust transcription accuracy. It also matches audio from substrings in the corpus and so can match any phrase from a tune, not just the incipit. To our knowledge, MATT2 represents the first attempt to adapt MIR to the specific characteristics of traditional Irish dance music and to support queries played on traditional instruments.

MATT2 makes use of the edit distance string matching algorithm. Edit distance also known as *Levenshtein distance* or *evolutionary distance* [30,31] is a concept from information retrieval and it describes the number of edits (insertions, deletions and substitutions) that have to be made in order to change one string to another. It is the most common measure to expose the similarity between 2 strings. The edit distance $ed(x, y)$ between strings $x = x_1 \ldots x_m$ and $y = y_1 \ldots y_n$, where $x, y \in \Sigma*$ is the minimum cost of a sequence of editing steps required to convert x into y. Σ is the alphabet of possible characters and $\Sigma*$ is the set of all possible sequences of $ch \in \Sigma$. Edit distance can be calculated using dynamic programming [32]. Variations on the edit distance algorithm have been applied in domains such as DNA analysis and automated spell checking and are commonly used in MIR systems [24,25,33].

3. SYSTEM DESIGN

MATT2 works on mono, digital audio files in the WAV format recorded at 44KHz. A high level diagram of the sub systems that make up MATT2 are presented in Figure 2. The subsystems present in MATT2 will now be described.

3.1. Onset Detection
The audio file to be annotated is first segmented into candidate note onsets and offsets using an onset detection function adapted from Gainza [34,35]. The onset detection function ODCF is based on time domain FIR comb filters. ODCF discovers harmonic characteristics of the input signal and is therefore tolerant to energy changes in an input signal

not caused by note onsets and is also better at detecting onsets in *legato* playing typical of windblown traditional instruments such as the flute and the tin whistle.

The input signal is first segmented into overlapping frames of 2048 samples (approximately 46 milliseconds). Each frame overlaps with the previous frame by 75%. Each frame is then passed through a bank of twelve FIR comb filters. A FIR comb filter works by summing the input signal with a delayed version of the same input signal. The delay of the filter is calculated as being the length in time of a single period of a waveform at the frequency.
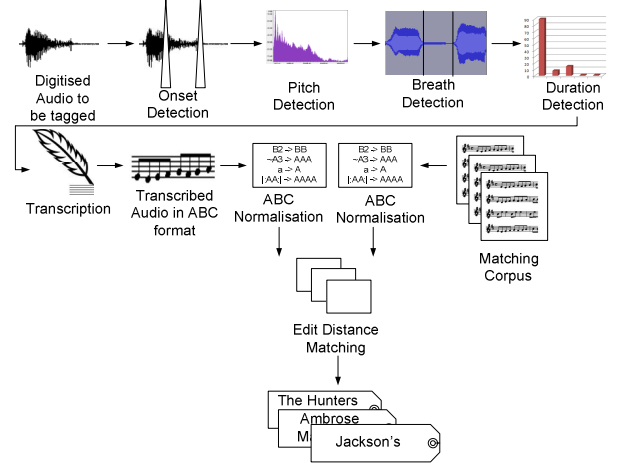

Figure 2: High level diagram of the MATT2 tune annotation system

This has the effect of amplifying the frequency or a harmonic thereof in the input signal that matches the frequency being filtered. Thus, the energy of the input signal is doubled only if the peaks of the signal coincide with the peaks of the FIR comb filter. This will only occur for a given delay and its integer multiples [36]. Twelve filters $E(m, D)$ with different delays are used corresponding to the twelve semitones in the key of D3 as per (1).

$$E(m,D) = \frac{\sum_{n=D+1}^{N} [x(n) + x(n-D)]^2}{4 * x^2(n)} \qquad (1) \text{ [34]}$$

For each frame of audio examined, the outputs of the audio passed through each of the twelve filters are calculated. A value for the ODF $dE(m)$ is then calculated as being the sum of the difference between the outputs of each of the twelve filters in successive frames squared, as described in (2).

$$dE(m) = \sum_{i=1}^{12} \left[E'(m, D_i) - E'(m-1, D_i) \right]^2 \qquad (2) \text{ [34]}$$

In the case where the pitch of the input signal changes from one note to another, this will result in a peak in the ODF graph. Using statistical techniques, a threshold is

calculated for each 25ms of audio as being the overall average ODF plus the standard deviation of the ODF in that frame [34]. Peaks above the threshold are recognised by the system as candidate note onsets. A *peak* in the ODF is defined as a value preceded by four ascending values and followed by four descending values (though MATT2 supports a configurable value for this). Onsets and offsets are considered by the system to be concurrent as the wooden flute is typically played legato and so a candidate note is considered to be a segment bounded by two adjacent onsets.

Figure 3 shows the signal for the first bar of the tune "The Boyne Hunt" with the detected candidate note onsets marked. The second plot in this figure shows the ODF for the signal, with the dynamic threshold and the candidate onsets marked. In this plot, it is significant that the first note contains a dynamic energy change approximately half way through the note which the ODF has correctly ignored.
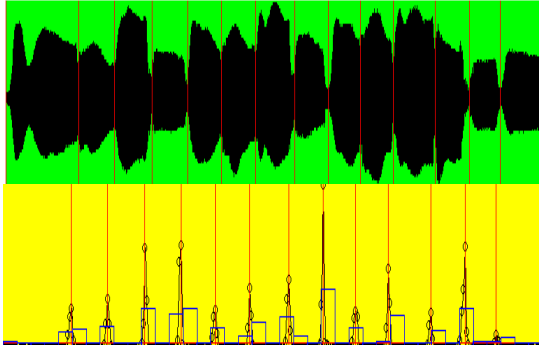

Figure 3: Signal and ODF plots of the first bar of the tune "The Boyne Hunt"

3.2. Pitch detection
To establish the perceived pitch of each note, the *fundamental frequency (F0)* of the note must be derived. The pitch detector subsystem first calculates the highest, nearest power of two $nP(m)$ for the length in samples $dS(m)$ of each segment m of audio bordered by a candidate onset as given in (3).

$$\max\left(nP(m) \mid 2^{nP(m)} < dS(m)\right) \qquad (3)$$

It then performs a short term FFT (Fast Fourier Transform) on the segment. To determine the perceived pitch the system first calculates peaks in the FFT. In this case, a peak is a value bordered by two ascending/descending values. The algorithm then calculates the interval between the first and second peak. This approach works well for the harmonics of the wooden flute.

5.1. Breath detection
The flute is a windblown instrument and hence a musician must periodically take breaths as a piece of music is being played [18,19]. MATT2 incorporates a breath detector sub system to transcribe a breath in the signal. The breath detector first calculates the average absolute amplitude of the entire signal $x(m)$. It then calculates the average absolute amplitude for each candidate note frame $x(n)$.

A breath is marked if either the perceived pitch detected by the pitch detector is less than 100Hz or the average amplitude of a candidate note n is less than a 10% threshold t of the average amplitude over the entire signal m as in (4). Again, this threshold is configurable. Breaths detected before the transcription of the first pitched note are ignored by the system.

$$\left(\frac{\sum_{n=s}^{N}[abs(x(n))]}{N-s}\right) < \left(\frac{\sum_{m=1}^{M}[abs(x(m))]}{M} \times t\right) \qquad (4)$$

3.3. Pitch spelling
A wooden flute used to play traditional music has a range of two octaves, though this can be extended by cross fingering techniques [18-20]. To tag each candidate note with a pitch spelling $pS(n)$, each calculated note frequency is compared with the frequencies of the notes in the key of D3 Major and D4 Major k_1... k_{16} the two octaves playable on a wooden flute. The nearest match for the frequency $f(n)$ is the assigned pitch spelling $pS(n)$ (5).

$$pS(n) = \min\left(abs(f(n) - k_1) \dots abs(f(n) - k_{16})\right) \qquad (5)$$

3.4. Note quantisation
At this stage, the system has a set of candidate notes, with corresponding durations and perceived pitches with spellings. To quantise note durations, the quantisation subsystem generates a *fuzzy histogram* of the intervals between consecutive onsets. As the test matching corpus contains only reels (tunes in 4/4 time) the histogram bin with the highest count is considered to be the length of a quaver note. The fuzzy histogram counts notes within 30% +/- of the the bin width. This 30% is the *fuzz* referred to in Figure 5. The algorithm also updates the bin width each time a candidate is added to the bin, so that the bin widths contain the cumulative average lengths of notes counted.

Using the histogram, the duration calculator concludes the length of a quaver qL as being the bin with the highest value. Pseudocode for this algorithm is given in Figure 5.

Figure 5 shows the duration histogram for a twenty eight second phrase of music from the tune "The Hunters Purse". Notes whose durations are close to zero have their durations merged with subsequent notes. This has the effect of eliminating consecutive onsets (false positives caused by noisy onsets) and also eliminating ornamentation notes such as those found in *rolls*, *cuts taps* and *crans* typical of traditional Irish music [18,9,19,20,28].

MATT2 then calculates the maximum bin value in a second fuzzy histogram of the new note durations after consecutive onset and ornamentation elimination. The

system uses this value to be the new length of a quaver qL'. The duration calculator then evaluates the nearest multiple qQ of the quaver length qL for each candidate note n as per (6).
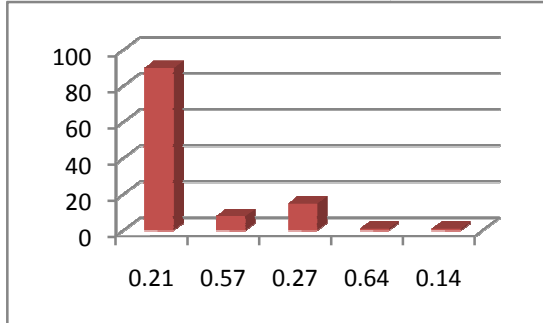

Figure 4: *Fuzzy histogram* **of candidate note lengths from a 28 second phrase from the tune "The Hunters Purse"**

$$qQ(n) = round\left(\frac{nD(n)}{qL}\right) \quad (6)$$

In this way notes are quantised as being quavers, crochets, dotted crochets and minims and ornamentation notes are eliminated from the transcription.

```
foreach (note in transcribed_notes)
begin
    found ← false
    foreach(bin in histogram)
    begin
        bin_start ← bin.width - fuzz
        bin_end ← bin.width + fuzz

        if (note.duration >= bin_start
      and
            note.duration <= bin_end)
        begin
          found ← true
          bin.count ++
          bin.width ← (bin.width  +
            note.duration) / 2
          break
        end
    if not found
    begin
       newNote.count ← 1
       newNote.width ← note.duration
       histogram.add(newNote)
    end
end
quaver_length ← max(histogram)
```

Figure 5: Pseudocode for the fuzzy histogram quaver length calculator

For many of the test recordings used to evaluate MATT2 recorded in imperfect conditions, this approach results in remarkably few transcription errors.

3.5. ABC Normalisation

Before edit distance matching against the corpus is carried out, both the transcribed string and strings from the corpus are normalised. This step is necessary as the ABC format supports features such as repeated sections, which need to be expanded so that they can be correctly matched against transcribed phrases. Normalisation of musical strings has the added advantage of minimising the effect of transcription errors on the calculation of the edit distance. Normalisation involves four stages.

Firstly, all whitespace, ornamentation markers and text comments are removed. When ornamentation markers (~{}) are removed from ABC transcriptions, this has the effect of quantising the duration of the majority of notes in corpus strings to multiples of the duration of a quaver.

Original:
d2BG dGBG|~G2Bd efge|d2BG dGBG|1 ABcd edBc:|2 ABcd edBd||

After Ornamentation removal:
d2BGdGBG|G2Bdefge|d2BGdGBG|1ABcd edBc:|2ABcdedBd||

After note expansion:
ddBGdGBG|GGBdefge|ddBGdGBG|1ABcd edBc:|2ABcdedBd||

After section expansion:
ddBGdGBGGGBdefgeddBGdGBGABcdedBc ddBGdGBGGGBdefgeddBGdGBGABcdedBd

After register normalisation:
DDBGDGBGGGBDEFGEDDBGDGBGABCDEDBC DDBGDGBGGGBDEFGEDDBGDGBGABCDEDBD

Figure 6: Normalisation stages for the A part of the tune "Come West Along the Road"

Secondly, all notes of duration greater than that of a quaver are expanded to be multiple instances of a quaver. This minimises the effect of false negatives in the ODF as *long* notes (false negatives in the ODF) become multiple short notes in both the transcribed phrase and in tunes from the corpus [34]. This also introduces a certain amount of tolerance in the matching subsystem to stylistic variations in the playing of tunes.

Thirdly repeated sections are expanded and bar divisions are removed. ABC supports several notations for different types of repeated phrases [37]. This means for example, that if the transcribed tune was the A part of a tune played twice,

this would be correctly matched against the expanded A part of a tune from the corpus.

Finally all notes are transformed to be in the same register. This is achieved by transforming lower case characters in the ABC of tunes to upper case. Figure 6 shows examples of each stage in the ABC normalisation process.

3.6. Edit distance matching

One final transformation is carried out on strings from the corpus before they are compared with transcribed strings. Occasionally, strings from the corpus are shorter than transcribed strings. For example, the transcribed string might be from a double reel, while the string from the corpus could be from a single reel (a tune half the length). In order to gain the maximal impact from the transcription, corpus strings shorter then transcribed strings are duplicated until their length is greater than the length of the transcribed string. This approximates what a real musician would do in order to extend the duration of a tune [38,37,17].

The minimum edit distance $eF(c)$ for each string c from the corpus Z then calculated using a cost of one for insertions, deletions and substitutions, for each pair consisting of the transcribed string s in substrings of c. A variation of the classic edit distance algorithm described in [31] is used to search for the minimum edit distance for a search string in substrings of a target string. This way any phrase from a tune can be matched not just complete tunes and not just incipits. Edit distances are normalised by dividing by the length of the transcribed string to produce $eF(c)$. Two methods for establishing the lowest edit distance were implemented. Firstly, MATT2 returns tunes whose edit distance is less than a configurable threshold. The system also returns the top ten matching tunes in order of lowest edit distance.

4. INTERFACE

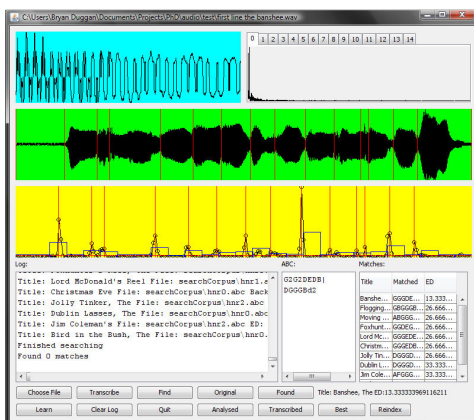MATT2 was developed in Java. A screenshot of the system is presented in Figure 7.

Figure 7: Screenshot of MATT2

The interface to MATT2 displays several useful plots of the outputs of each stage in the transcription and the matching algorithm such as the current frame being analysed, the onset detection function and the FFT of each detected note. Additionally, the interface displays the transcription in ABC format and the title of the current closest tune match. MATT2 can also play the original WAV file being analysed, the transcribed pitches and durations, the quantised transcription in ABC, the closest match and any of the matched tunes. When the matching algorithm terminates, MATT2 displays and can play any of the top ten closest matching tunes, with their corresponding edit distances. It can also operate in batch mode where it will attempt to annotate all the WAV files in a folder.

4. EXPERIMENT AND RESULTS

To evaluate MATT2, nine subjects recorded the A and B parts to a number of double and single reels. These recordings were made in imperfect conditions (a kitchen in a house, a school room and a pub) and contain ambient noise such as chairs moving, doors opening and foot taps. The recordings were edited so that the audio being tested contained a mix of complete tunes and segments of tunes taken from the start, middle and end of tunes. Some deliberately challenging audio such as archive recordings, flute duets, flute and fiddle duets, and fiddle solos was also included.

For forty three of input audio files, the algorithm correctly identified the tune. A further five tunes were correctly annotated within the top ten closest matches. Just two of the test audio files were incorrectly annotated. Experiments have shown that MATT2 demonstrates a remarkable robustness to tempo variations, musical style, ornamentation, musician variations, instruments and recording environments. Amazingly, one of the tunes tested was recorded over thirty years ago by the flute player Packie Duignan on an analogue tape recorder and the audio had badly degraded. We did not attempt to pre-process this audio before annotating it. For this test, the algorithm did not correctly identify the tune; however the correct tune was the second closest match. The correct tune could have been returned with the addition of a simple heuristic. Further fiddle solos, flute duets and flute and fiddle duet was also included in test audio. In all cases the algorithm correctly identified the tune, despite the fact that the onset detection function used is reported to be inaccurate at detecting onsets in fiddle recordings [35].

The average normalised edit distance was 39% for the closest matches, while the average of the nearest second match was 53%. The difference between these (14%) can be considered as a confidence level. As an example of the results the system generates, Figure 8 shows the edit distances returned by MATT2 for the tune, "The Golden Keyboard", played by flute player Eamonn Cotter. This

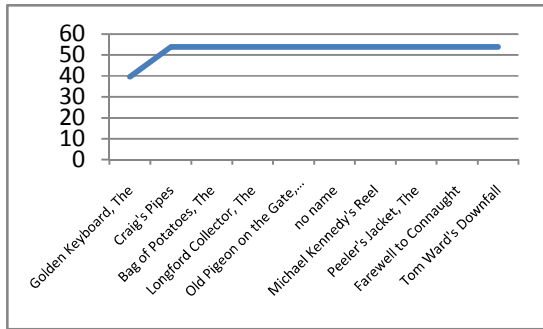recording was made on a portable MP3 recorder and subsequently transcoded to WAV format.

Figure 8: Top ten edit distances for a recording of the tune "The Golden Keyboard"

The edit distances shown in the graph are normalised and there is a 14% difference between the correct tune and the nearest closest match. The second and subsequent closest matches have only small variances. These results are summarised in Table 1.

Average ED (closest):	39%
Average ED (next):	53%
Difference:	14%
Average ED (closest, correct):	36%
Average ED (next, correct)	52%
Difference:	16%
Average ED (closest, incorrect):	56%
Average ED (next, incorrect):	57%
Difference:	1%

Table 1: Average edit distances for the closest match and the next closest match for tunes correctly and incorrectly annotated

When incorrect matches were considered, it was discovered that in all cases incorrect matches were as a result of transcription errors. Further, the transcription errors were caused by either unusually prominent foot taps or the musician started slow and then speeded up as the tune was played, which affected note duration quantisation.

6. CONCLUSIONS

In this paper MATT2 a system for annotating field recordings of traditional Irish music with metadata was described. MATT2 combines a novel transcription system that makes use of ODCF to detect onsets and fuzzy histograms to quantise note durations. MATT2 also makes use of publicly available transcriptions in ABC made by the traditional music community to match transcriptions against. A novel string normalisation technique that takes advantage of the ABC language to eliminate the effect of transcription errors and stylistic variations in input audio was also presented. Further MATT2 improves on pitch contour representations of music strings in MIR systems by using accurate pitch and duration information from input audio in the melody matching subsystem. To our knowledge, MATT2 represents the first attempt to adapt MIR to the specific characteristics of traditional Irish dance music.

Experiments demonstrate that the approach outlined in this paper is robust to variations in musician, style and instrument. In successfully testing MATT2 on badly degraded archive audio, we conclude that the approach presented can be further developed for use on the many thousands of hours of archived recordings of traditional music that currently exist and that are being collected. Our system currently works on segments from tunes or complete tunes. One interesting complication we have not addressed is that tunes are usually played in sets of two or more tunes and each tune is usually played twice. We therefore have the problem that a single audio input file may contain several tunes.

In the cases where the algorithm did not correctly identify the correct tune, we conclude that the transcription subsystem was not able to accurately transcribe the tune. We therefore feel that improving transcription accuracy will lead to more accurate matching and will focus on this problem in future work. Widmer *et al*. [27] state that transcription algorithms need the kind of higher level musical knowledge that humans poses and we hope to try and develop this approach, possibly using a *preference rule* style approach similar to the work of Temperly [39]. Support for the transcription of tunes in different time signatures is planned as is experimentation with corpora of tunes in other popular time signatures such as jigs and hornpipes.

REFERENCES

[1] G. Wallis and S. Wilson, *The Rough Guide to Irish Music*, London: Rough Guides, 2001.

[2] Tony Kearns and Barry Taylor, *A Touchstone for the Tradition - The Willie Clancy Summer School*.

[3] "M-AUDIO - MicroTrack II - Professional 2-Channel Mobile Digital Recorder"; http://www.m-audio.com/products/en_us/MicroTrackII-main.html.

[4] "Irish Traditional Music Archive / Taisce Cheol Dúchais Éireann"; http://www.itma.ie/.

[5] "Comhaltas: The Music"; http://comhaltas.ie/music/.

[1] S Driscoll, "A Trio of Internet Stars: ABC's," *Fiddler Magazine*, vol. 11, Summer. 2004.

[7] "The Session"; http://www.thesession.org/.

[8] M. Krassen, *O' Neil's Music of Ireland*, Waltons, 1975.

[9] N. Keegan, *The Words of Traditional Flute Style*, MPhil Thesis, University College Cork, Music Department, 1992.

[10] C. Walshaw, "The ABC home page," 2007; http://www.walshaw.plus.com/abc/.

[11] J. Chambers, "O' Neills Books," 2007; http://trillian.mit.edu/~jc/music/book/oneills/.

[12] H. Norbeck, "ABC Tunes," 2007; http://www.norbeck.nu/abc/index.html,

[13] B. Duggan, "A Portable Tune Teaching Tool for Traditional Musicians," *DIT Annual Showcase of Learning & Teaching Activities*, Jan. 2007.

[14] Bryan Duggan, "Enabling Access to Irish Traditional Music Archives on a PDA," *Eight Annual Irish Educational Technology Users Conference, DIT Bolton St, Ireland*, May. 2007.

[15] Bryan Duggan, "Learning Traditional irish Music using a PDA," *IADIS Mobile Learning Conference, Trinity College Dublin*, Jul. 2006.

[16] B. Duggan, Z. Cui, and P. Cunningham, "MATT - A System for Modelling Creativity in Traditional Irish Flute Playing," *Third ECAI Workshop on Computational Creativity*, Riva Del Garda, Italy: 2006.

[17] Nan Zheng and Bryan Duggan, "A Combinational Creativity Approach to Composing Traditional Irish Reels," *18th Irish Conference on Artificial Intelligence and Cognitive Science, Dublin Institute of Technology, Ireland*, Aug. 2007.

[18] G. Larson, *The Essential Guide to Irish Flute and Tin Whistle*, Mel Bay Publications, Inc., 2003.

[19] C. Hamilton, *The Irish Flute Players Handbook*, Cork: Breac Publications, 1990.

[20] F. Vallely, *Timbre: The Wooden Flute Tutor*.

[21] R. Dannenberg et al., "The MUSART Testbed for Query-by-Humming Evaluation," *Computer Music Journal*, vol. 28, 2004, pp. 34-48.

[22] A. Ghias et al., "Query by humming - musical information retrieval in an audio database," *In ACM Multimedia 95 Proceedings, San Francisco*, 1995.

[23] "Melodyhound: Query by Humming"; http://www.melodyhound.com/query_by_humming.0.html.

[24] N. Hu and R. Dannenberg, "A comparison of melodic database retrieval techniques using sung queries," *Proceedings of the second ACM/IEEE-CS joint conference on Digital libraries*, 2002, pp. 301-307.

[25] W. Birmingham et al., "Musart: Music Retrieval Via Aural Queries," *Ann Arbor*, vol. 1001, pp. 48109-2110.

[26] N. Adams, M. Bartsch, and G. Wakefield, "Coding of sung queries for music information retrieval," *Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.*, 2003, pp. 139-142.

[27] G. Widmer et al., "From Sound to `Sense' via Feature Extraction and Machine Learning: Deriving High-Level Descriptors for Characterising Music," *Sound to Sense:Sense to Sound: A State-of-the-Art*, 2005.

[28] S. Tansey, *The Bardic Apostles of Innisfree*, Tanbar Publications, 1999.

[29] F. Vallely, *Flute Routes to 21st Century Ireland*, PhD Thesis, National University of Ireland, 2004.

[30] V. Levenshtein, "Binary Codes Capable of Correcting Deletions, Insertions and Reversals," *Soviet Physics Doklady*, vol. 10, 1966, p. 707.

[31] G. Navarro and M. Raffinot, *Flexible Pattern Matching in Strings: Practical On-Line Search Algorithms for Texts and Biological Sequences*, Cambridge University Press, 2002.

[32] Maarten Grachten, Josep Llu´ıs Arcos, Ramon L´opez de M´antaras,, "TempoExpress, a CBR Approach to Musical Tempo Transformations, Advances in Case-Based Reasoning," Madrid, Spain: 2004.

[33] K. Lemstrom and S. Perttu, "SEMEX-An Efficient Music Retrieval Prototype," *First International Symposium on Music Information Retrieval (ISMIR)*, 2000.

[34] M. Gainza, *Music Transcription within Irish Traditional Music*, PhD Thesis, Dublin Institute of Technology, Faculty of Engineering, 2006.

[35] M. Gainza and E. Coyle, "Automating Ornamentation Transcription," *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 1, 2007.

[36] Stephen W Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*, California Technical Pub, 1997.

[37] S. Mansfield, "How to Interpret ABC Notation," 2007; http://www.lesession.co.uk/abc/abc_notation.htm.

[38] F. Vallely, *The Companion to Irish Traditional Music*, New York University Press, 1999.

[39] D. Temperley, *The Cognition Of Basic Musical Structures*, MIT Press, 2004.