Analysis and Editing

Yuxiang Liu[1], Roger B. Dannenberg[2], and Lianhong Cai[1]

[1] Tsinghua National Laboratory for Information Science and Technology,
Tsinghua University, Beijing, China
[2] School of Computer Science, Carnegie Mellon University, Pittsburgh, US
liuyuxiang06@mails.tsinghua.edu.cn, rbd@cs.cmu.edu
clh-dcs@tsinghua.edu.cn

*Published in: *Proceedings of the Seventh International Conference on Intelligent Computing,* Cairo, Egypt, December 2015

Abstract. Digital music editing is a standard process in music production correcting mistakes and enhancing quality, but this is tedious and tir consuming. The Intelligent Music Editor, or IMED, automates routine mu editing tasks using advanced techniques for music transcription (especia score alignment), and signal processing. The IMED starts with multi recorded tracks and a detailed score that specifies all of the notes to be play A transcription algorithm locates notes in the recording and identifies th pitch. A scheduling model tracks instantaneous tempo of the recorded perfor ance and determines adjusted timings for output tracks. A time-domain pi modification/time stretching algorithm performs pitch correction and ti adjustment. An empirical evaluation on a multi-track recording illustrates proposed algorithms achieve an onset detection accuracy of 87% and a detai subjective evaluation shows that the IMED improves pitch and timing accur while retaining the expressive nuance of the original recording.

Keywords: Intelligent Music Editor, Music Transcription, Score-Audio Ali ment, Pitch Estimation, Time Stretching, Pitch Shifting.

1 Introduction

Editing allows recording engineers and producers to make *incremental* c audio rather than discarding a mostly-good recording and starting over. U the biggest limitation of editing is the human time it takes to perform the ed most edits simply adjust notes to achieve better rhythmic and tuning ac seems quite possible to automate a large fraction of the most desirable e could lower recording costs and enable many more creative musicians t recordings with a professional sound.

In this paper, we describe an Intelligent Music Editor (IMED) and disc lems that arise in practice. As a highly automated and easy to use system analysis and editing, IMED is able to analyze music content by linking s symbolic representations of music. The overall strategy is to use symbo

D.-S. Huang et al. (Eds.): ICIC 2010, LNAI 6216, pp. 123–131, 2010.

can automatically manipulate music recordings, according to an automatica ated plan or user instructions, by moving notes, stretching time, correcting mixing tracks. The output of the system is an edited version of the origina justments to timing, tempo, pitch, and loudness.

On one hand, IMED offers great flexibility to musicians and editors by recordings to be automatically refined in term of pitch, timing and dynamic the other hand, content-based music analysis is useful for music understa retrieval. Currently, music structure labeling is performed manually. Hence analysis techniques can effectively reduce the workload of human anno would be an indispensable component of a music information retrieval syste

2 Related Work

An early work describing the need of intelligent audio editor was presented [1]. Although the concept that an audio editor should be able to make use content was promising, the actual system was not very practical due to tech its of that time.

Tzanetakis [2] implemented a working prototype of an intelligent edito music, which provided an interactive experimentation environment for comb testing content-based analysis components in the domain of Music Inform trieval (MIR).

A number of studies on score-audio alignment [3-5], structural analysis analysis-resynthesis of signals [8] have been performed. However, there little work on integrating these techniques into a platform except the former of IMED [9]. This early work was in essence a proof-of-concept and not in for production work. The present study is the first attempt to use large-scale editing techniques on an actual studio recording. This study raises many problems that did not arise in earlier work, which often looked at carefull data.

3 Analysis: Score-Assisted Music Transcription

The major task of the analysis stage is to transcribe performed music int sponding score. In another words, the program identify notes in an audio associate them with notes in the reference score. A large number of autom niques for music transcription have been proposed the majority of whi notes directly from music signals and seldom involve reference scores. In the full transcription is not necessary because we have a reference score a pect the performers to adhere closely to the score. While music transcriptio largely unsolved problem, audio-score alignment is relatively easy. In IME accomplished in the two steps: score-assisted high-resolution onset dete pitch estimation.

$$\operatorname{argmax} P(T \mid R, M)$$

We want to find the best (most likely) correspondence T between a given
cording, *R*, and score, *M*. The recording comes from a real-time performa
will naturally contain deviations from a precise rendering of the score. On
hand, the score is an abstract, discrete representation of a continuous phe
and composers expect and rely upon musicians to interpret the score with e
and feeling, so there will always be deviations.

To form a correspondence between R and M, we first apply an over
alignment to stay on track and then use some more refined methods at the
note level in order to achieve a high temporal accuracy.

Score Alignment

The essence of audio-score alignment is as follows: Audio signals are di
50ms long, half-overlapping frames. Both score and audio frames are conv
sequences of chroma vectors [10]. A distance metric is used to measure dis
between these chroma vectors. Finally, we employ Dynamic Time Warping
find the optimal match between the two time series. Please refer to our prev
[4] for more detail. Each track of the multi-track recording is separately alig
corresponding part (a MIDI track or channel) in a reference score.

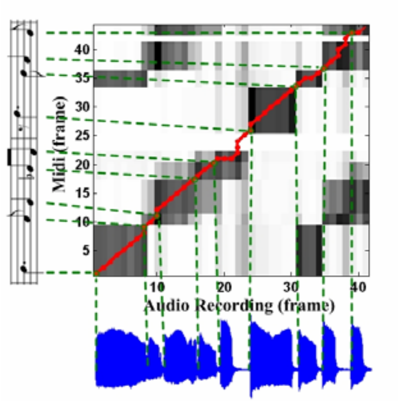

Fig. 1. Audio-Score Alignment

In an ensemble, not all musicians play all the time. Therefore, we need
detection process to determine segments where the musician is actually pla
ally, this can be achieved by tracking energy of sounds and using an empiri
old to distinguish segments with low energy. Unfortunately, in most cases,
for a live concert, musicians cannot be separated perfectly from each oth
interference among instruments is inevitable. Hence even when a musician

same with score alignment, but we replace the chroma vector with Short Ti
(STE) and redefine the distance measure as below:

$$w \times | STE_{mid} - STE_{audio} | + (1 - w) \times | d(STE_{mid}) - d(STE_{audio}$$

where the first term is difference of STE between a midi frame and an au
The second term is difference of the first derivative of STE, which reflect
difference between midi and audio. This definition is motivated by the ph
that for any individual note generated by a musical instrument, there is an a
An attack is extremely salient when it occurs right after a silent segment an
rapid amplitude rise in the waveform. We believe involving dynamic dif
STE is useful for detecting silence boundaries.

A pitch change will definitely result in a sudden spectral change. Inspir
observation, when computing dissimilarity between midi and audio frame
spectral changes between successive frames into account so as to improve
accuracy at note boundaries. In our implementation, the overall distance is d

$$w \times Dist(M_i, R_j) + (1 - w) \times | Dist(M_i, M_{i-1}) - Dist(R_j, R_{j-1}) |$$

where M_i stands for the chroma vector of the i^{th} frame in a Midi, and R_j is t
vector of the j^{th} frame in a audio recording. $Dist(a,b)$ is Euclidean Distanc
vector a and b. The first term corresponds to the spectral difference bet
frame and audio frame, while the second term can be considered as the dif
their first derivative.

Once the distance measure is specified, the rest of the process is essenti
ter of searching for a shortest path through the dissimilarity matrix usin
programming.

Bootstrap Learning for Accurate Onset Detection

Due to the limitation of chrome feature extraction, the analysis windows s
be shorter. Hence the temporal resolution of score alignment is 25ms in
mentation. In our experience, even highly overlapped windows with a sm
size do not improve the temporal resolution of alignment results. Howeve
not precise enough for editing, so additional refinement is needed. To this e
alignment data to train an onset classifier.

The features used for onset classification are energy, fundamental freq
relative strengths and frequency deviations of the first three harmonics, and
crossing rate. We use overlapping analysis windows of size 23.2ms (1024
a sample rate of 44.1 kHz). The hop size is 5.8ms (256 samples), provid
temporal resolution.

Typically, onset detection systems are intended to work with a wide va
puts, and for detectors based on machine learning, one would expect to requ
set of hand-labeled training examples. In our system, however, it is an adva
the detector to a particular instrument or even a particular performer. Furthe
can use alignment data rather than hand-labeled data as training data. In ou

rate set of onset labels as a side-effect of training the classifier. Due to spa
tions, we refer the reader to a previous publication [11] for details.

3.2 Pitch Estimation

Once musical signals are segmented into notes, the YIN algorithm [12]
estimate a pitch for each note. In YIN, an average magnitude difference
(AMDF) between a segment of the signal and the same segment lagged
period is calculated. (Shown in Fig. 2(a)) The algorithm searches for a
(Point P2 in Fig. 2(a)) throughout the AMDF curve, which varies as a functi

Although this approach works well, like most pitch estimation algorithm
fers from too low/high errors, where a longer period (P3 in Fig. 3a) or a shor
(P1 in Fig. 2(a)) valley is chosen incorrectly. These valleys often occur
higher or lower than actual pitch.

In a music performance, it is rare that a performer plays a note that is
away from a reference note in a score. This encourages us to restrict the sea
in a small neighborhood around the reference pitch (Fig. 2(b)). Our experim
that this simple method works extremely well for eliminating octave errors.

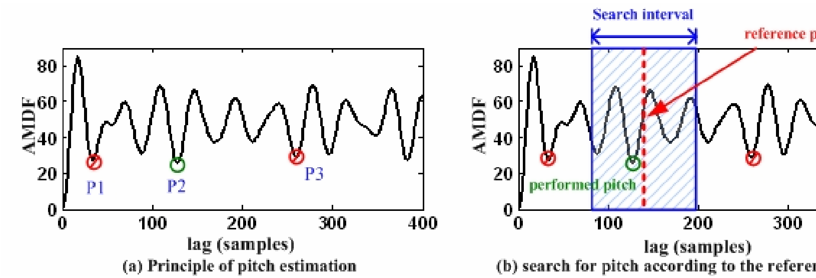

Fig. 2. Pitch Estimation Algorithm

4 Automatic Editing

Usually, performed tracks differ from both the reference scores and eac
terms of note timing, durations, and tuning. This is due to differences in m
pretation, limitations to human accuracy, and simply performance errors. I
make tracks sound more natural and coherent, the IMED first constructs pla
the labeled notes in terms of starting time, duration, and pitch.

4.1 Reschedule Timing and Determine Pitch

At first, it might appear obvious how to modify note timing – simply adjus
to match the time in the score. This simple approach might be useful, but in
will take out the "life" of the music, making it sound mechanical or "robot

IMED must therefore track the overall tempo of the whole ensemble an just each individual instrument track to match the group. We assume tha relatively stable and drastic tempo changes do not happen. Accordingly, w mate instantaneous tempo at a reference score position by linear regressio tual performed onset times nearby.

To calculate the instantaneous tempo at time x in a midi track, we pick whose onset (based on midi time) is in the sliding window [x-T/2, x+T/2] tracks. The sliding window size T is set to 20 seconds empirically, which linear regression procedure to span about forty beats. A linear regression pr then applied to find the least squares fit of a linear function that maps beat to predicted onset time for beat x is simply the value of the fitted function evalu

Compared to timing, pitch determination is rather simple. A n performance practices will be used by musicians for expression, suc glissando and portamento, during a performance. A good editor is require these expressive effects. Therefore, instead flattening performed pitch to corresponding to the MIDI number in the score, we shift the whole pitch c interval between the weighted average of the estimated pitch curve and derived from MIDI key number.

4.2 Time Adjustment and Pitch Shifting

Time stretching and pitch shifting are carried out simultaneously by a hi timescale-pitch modification algorithm based on Pitch Synchronous Overla (PSOLA) [13]. Our implementation relies on the élastique SOLOIST 2.0 zplane.development [14].

In order to avoid clicks at splice points, the whole track is edited in a c manner. Thus, if there is a phrase with several notes, we do not separate transform them, and splice them back together. Instead, we transform all o using time-varying stretch factors and pitch shift amounts, allowing the PS rithm to handle the details.

Consider, however, that PSOLA is by definition pitch synchronous, so o periods can be inserted or deleted to change the duration of a note or segm dio. This means the length of output signals is not guaranteed to satisfy the stretch ratio exactly. Although differences are rather small respectively (than one period each), accumulated errors could still affect quality of result t

To avoid accumulated quantization error, we update the stretch ratio for iteratively, treating the PSOLA algorithm as a “black box” whose next in from the i^{th} sample in the source track and whose next output will be writte sample of the destination track. Now, suppose when the program begins to p k^{th} note that the next note (at k+1) has onset times corresponding to sampl in the source and destination tracks, respectively. The stretch ratio for th should then be $\frac{j'-j}{i'-i}$, which will place the $k+1^{th}$ note as accurately as independent of any previous quantization error.

Although we have been working with many pieces of music, a detailed eval
been carried out on a recording of "Embraceable You". The music lasts 3 m
45 seconds. The instrumentation consists of five "horn" tracks: alto saxoph
saxophone, baritone saxophone, trumpet and trombone. The total number c
all five tracks is 1767. The performance was recorded in a studio, and all
played at the same time. There was a close microphone in front of each
There is obvious "cross-talk" between channels, but the multi-track sourc
was edited extensively by hand for a resulting compact disc.

To measure the performance of the onset detection algorithms which i
lence alignment, note alignment, and bootstrapped onset detection, we run
tor on the acoustic recordings and then manually correct mistakes by movi
onset times to appropriate positions in the recording. Because the detector is
score alignment, extra or missing note errors occur only when a perform
incorrectly. Other errors include inaccurate onset timing and note shifti
onsets are correctly identified but assigned to the wrong note in the score
revising all the note onsets in recordings is too time consuming, we only co
ous mistakes whose deviation is larger than 25ms.

Table 1. Accuracy of onset detection

	Trumpet	Alto Sax	**Tenor Sax**	**Baritone sax**	Trombone
Correct Onset	316	355	357	238	277
Total Onset	326	371	373	326	361
Accuracy	96.93%	95.69%	95.71%	73.00%	76.73%

Tab. 1 illustrates the overall detection accuracy. As shown, the overall a
87%. For trumpet, alto sax and tenor sax, the accuracy is much better, with
95%, showing the feasibility of annotating the music in a fully automati
However, the performance is not so satisfactory when IMED deals with ba
and trombone tracks. The poorer performance could be due to bass charact
these two instruments. Take the baritone sax track as an example. Most
around pitch C3 (a fundament frequency of 123Hz). Such low pitches p
program from extracting spectrum and F0 of audio signals accurately. A
longer window size for spectral analysis may help to gain a better spectral r
tion, it will result in a low temporal resolution. Balancing the trade off betw
tral and temporal accuracy and improving onset detection in the bass reg
interesting challenge for future investigation.

To compare the edited sound with the original recording, we conducted a
evaluation. We have three versions of the recording: an original version w
editing, an automatic edited version without any manual intervention, and
version based on handmade corrected onset labels. For each version, all in
are mixed into one track. A subject first listens to the original version pickin
lems where notes are not synchronized in the recording or where a note is pla
tune. Then he listens to the two edited versions and finds out whether errors
At the same time, he should pay attention to whether the edited sounds a

problems, 15 intonation (tuning) problems, and 1 note held too long. IMED
33 of 42 timing problems, 3 remained the same, and 6 became worse. In ad
new timing problems were introduced. Of the intonation problems, 8 were
5 were rated the same, and 2 became worse. The single long note problem
worse after editing. Finally, there were 3 objectionable editing artifacts detec

We can say that IMED reduced the original 58 problems, to only 28 prob
ting the manual editing task by more than half. With hand-corrected timing
change is that the number of timing problems that grew worse was reduced
3. These numbers are encouraging, but they do not tell the whole story. O
edited recording suffers from a loss of "crispness," probably due to the fa
main pitch- and time-shifted signals are added to the "bleed through" signal
ing from the same instrument but arriving through other tracks, which are
differently. This creates a chorus-like effect that generally sounds muddy an
creates noticeable interference sounds. There are several solutions to this,
better isolation of instruments in the recording process, recording instrumen
a-time, automatic attenuation of "bleed through" when the main instrume
track is not playing, and noise removal techniques.

6 Conclusions and Future Work

In this paper, an intelligent music editor, which transcribes music recor
makes adjustments to note pitch and timing in an automatic fashion, is pres
believe this represents the first attempt to edit a multi-track, studio recor
matically. By combining score-audio alignment with a bootstrap learning
the proposed transcription model yields an overall onset detection accurac
which shows the feasibility of a fully automatic music transcription syste
domain algorithm based on PSOLA is proven effective at pitch shifting
stretching to achieve a natural and musical result. A subjective evaluatio
strated that the system automatically corrects pitch errors and adjusts timin
destroying the musicality of the recording. In fact, the process improves th
quality of the recording as if it were edited by hand.

Nevertheless, there is plenty of room for improvement in the editor. B
with real studio recordings, our work has revealed a number of practical i
may guide future researchers who are interested in this new problem.

As for score alignment, our current model based on chroma features su
pitches are low. A series of time domain features should be considered,
effectiveness will be explored in our future research. In addition, it is te
align all tracks simultaneity, which helps to use timing information of
notes in other track when the detector fails to identify an onset in a bass t
rently, IMED assumes performances are correct except for small timing
errors, so it does not detect outright mistakes such as a missing or extra note
be possible to detect mistakes automatically and even use similar notes
cording to synthesize a performance. For example, string matching algorith

Acknowledgments

A major part of work was done while the first author was at Carnegie Mello
sity. The first author was supported by the National Natural Science Four
China (No. 90820304). This material is also based on work by the seco
supported by the (U.S.) National Science Foundation Grant No. 0855958.
like to thank Tim Flohrer, Alexander Lerch, and zplane.development
élastique SOLOIST 2.0 SDK library.

References

1. Chafe, C., Mont-Reynaud, B., Rush, L.: Toward an Intelligent Editor for Dig Recognition of Musical Constructs. Computer Music Journal 6 (1982)
2. Tzanetakis, G., Hu, N., Dannenberg, R.: Toward an Intelligent Editor for Jazz IEEE Workshop on Image Analysis for Multimedia Interactive Systems, L (2002)
3. Orio, N., Schwarz, D.: Alignment of Monophonic and Polyphonic Music to a The International Computer Music Conference, pp. 155–158. International Co sic Association, San Francisco (2001)
4. Dannenberg, R., Hu, N.: Polyphonic Audio Matching for Score Following and Audio Editors. In: The International Computer Music Conference. International Music Association, San Francisco (2003)
5. Ewert, S., Muller, M., Grosche, P.: High resolution audio synchronization us onset features. In: The IEEE International Conference on Acoustics, Speech, Processing, Taipei, Taiwan (2009)
6. Tzanetakis, G., Cook, P.: Multifeature Audio Segmentation for Browsing and In: Workshop on Applications of Signal Processing to Audio and Acoustics, USA (1999)
7. Aucouturier, J., Sandler, M.: Segmentation of Musical Signals using Hidd Models. In: Audio Engineering Society (AES) Convention, Amsterdam, (2001)
8. Woodruff, J., Pardo, B., Dannenberg, R.: Remixing Stereo Music with Sco Source Separation. In: 7th International Conference on Music Information Re ceedings, Victoria, Canada, pp. 314–319 (2006)
9. Dannenberg, R.: An Intelligent Multi-Track Audio Editor. In: The Internationa Music Conference. The International Computer Music Association, San Francis
10. Fujishima, T.: Realtime chord recognition of musical sound: a system using c music. In: International Computer Music Conference, Beijing, China (1999)
11. Hu, N., Dannenberg, R.: Bootstrap learning for accurate onset detection. Mac ing 65, 457–471 (2006)
12. de Cheveigné, A., Kawahara, H.: YIN, a fundamental frequency estimator for music. The Journal of the Acoustical Society of America 111, 1917–1930 (200
13. Moulines, E., Charpentier, F.: Pitch-synchronous waveform processing techniq to-speech synthesis using diphones. Speech Communication 9 (1990)
14. Flohrer, T.: SDK Manual (for élastique SOLOIST 2.0 SDK). Zplane.Develop (2007)