# MUSICAL OFFSET DETECTION OF PITCHED INSTRUMENTS: THE CASE OF VIOLIN

**Che-Yuan Liang, Li Su, Yi-Hsuan Yang**
Academia Sinica
{mister2dot4,lisu,yang}@citi.sinica.edu.tw

**Hsin-Ming Lin**
University of California, San Diego
hsl040@ucsd.edu

## ABSTRACT

Musical offset detection is an integral part of a music signal processing system that requires complete characterization of note events. However, unlike onset detection, offset detection has seldom been the subject of an in-depth study in the music information retrieval community, possibly because of the ambiguity involved in the determination of offset times in music. This paper presents a preliminary study aiming at discussing ways to annotate and to evaluate offset times for pitched non-percussive instruments. Moreover, we conduct a case study of offset detection in violin recordings by evaluating a number of energy, spectral flux, and pitch based methods using a new dataset covering 6 different violin playing techniques. The new dataset, which is going to be shared with the research community, consists of 63 violin recordings that are thoroughly annotated based on perceptual loudness and note transition. The offset detection methods, which are adapted from well-known methods for onset detection, are evaluated using an onset-aware method we propose for this task. Result shows that the accuracy of offset detection is highly dependent on the playing techniques involved. Moreover, pitch-based methods can better get rid of the soft-decaying behavior of offsets and achieve the best result among others.

## 1. INTRODUCTION

In the literature, offset detection has been frequently mentioned in the context of performance analysis [14], automatic music transcription (AMT) [4, 13, 21, 24, 29], note segmentation [10, 15, 18, 26], and computational auditory scene analysis (CASA) [19]. In these systems, offset detection is required for complete measurements of duration, intonation, vibrato, dynamics, and other kinds of note-based properties of music [14]. However, to date, offset detection is mostly treated as a component in a large system. Few studies, if any, are dedicated to offset detection.

The challenges of offset detection can be illustrated by the attack-decay-sustain-release (ADSR) model of music signals. First, consider the ADSR envelope of a plucked string signal in Figure 1(a). The envelope of such signals usually consists of a short attack, unobservable sustain, and a gradual decay right before the release. Due to the difference in hearing threshold among human listeners, the possible region of perceptual offset time (*i.e.* medium gray region) can be fairly wide due to the gentle slope of the release. Because of this, offset detection may slip into the game of comparing the subjective listening thresholds. In contrast, there is little ambiguity associated with the onset time (*i.e.* dark gray region) due to the short attack.

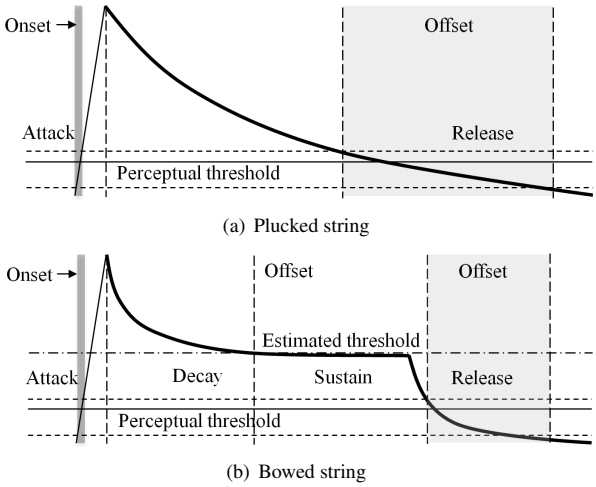



**Figure 1**. The ADSR envelops of a plucked string (upper) and a bowed string signal (lower). The gray blocks show the ambiguity of onset (dark) and offset (light) due to the variation of hearing threshold. The bold-line segments of the envelopes are the possible regions to detect an offset.

Figure 1(b), on the other hand, shows the possible ADSR envelope of a bowed string signal, which contains four discernible parts. Because the release time is shorter, the temporal uncertainty of the perceptual threshold of such signals should be less than that of plucked string signals. In practice, however, computationally estimating the perceptual threshold in bowed string signals may not be easy, due to the similar shapes of the decay and the release parts. Things are more complicated in real-world signals that contain rich variation in the employed instruments and playing techniques, which would shape the ADSR envelope in totally different ways. Indeed, the challenges of offset detection can be attributed to the gentle slope of the release part and the rich variation in timbre in music signals.

This paper presents a preliminary attempt focusing on musical offset detection. Specifically, this paper discusses various aspects of offset detection research, from building a dataset, designing an algorithm informed by the aforementioned challenges, to evaluating the performance of offset detection. We restrict our discussion on the violin, and investigate the offset detection of its six different playing techniques. This way, we exclude musical signals with very long releases, such as the pedaled piano.

Specifically, we discuss possible approaches to manually annotating offset times in music and then propose a new one (see Section 3). The proposed approach is adopted to construct a new offset detection dataset, which we have made available to the research community online. [1] With the new dataset, we present and evaluate a number of offset detection algorithms based on the spectral flux, energy and pitch attributes of music (see Section 4). To investigate the effect of playing techniques, an in-depth technique-by-technique discussion is also presented (see Section 5). As another contribution of this paper, a new evaluation measure for offset detection is also proposed and discussed.

## 2. RELATED WORK

Most of the offset detection algorithms are implemented in two main directions: thresholding on energy salience, and thresholding on pitch salience. The energy salience can be the physical, perceptual or pitch-wise sound levels [15, 16, 20]. The thresholding on pitch salience is often seen in the context of AMT [13, 24], where the offset can be regarded as the falling position of a pitch salience function for a specific pitch. In Non-negative matrix factorization (NMF)-based AMT, the offset is usually determined by a threshold on the activation matrix [29]. Other approaches, such as novel features like correntropy [10], data-driven models such as the hidden Markov models (HMM) [4, 14], support vector machines (SVM) [18], have also been applied to offset detection. Spectral flux-based approaches (*i.e.* using temporal difference of spectrum-based representations) [3, 6, 7, 28], despite being a conventional method in onset detection, are rarely used in offset detection except for some studies [19, 21, 26]. Post-processing with known onset information is sometimes used [10, 15].

## 3. DATASET CONSTRUCTION

### 3.1 Annotating the Offset

There are several possible ways to annotate the offset of musical notes and build a dataset, depending on the data format of the music content. For example, one can take the timestamps of note-off message in MIDI as the ground truth for offset. Although audio data for experiments can be generated by MIDI efficiently, this method cannot accurately indicate the perceptual offset time in many cases. For example, a control message "sustain pedal" makes the synthesizer prolong the amplitude envelope even after the note-off message. In this case, the perceptual offset can fall far behind the note-off message. Alternatively, one can also construct a music dataset from video recordings. Video can plausibly provide visual clues to a performer's movement which are sometimes helpful to estimate the offset time. Inaccuracy, however, may result from audio-visual asynchrony and low frame rates.

Another useful way to specify the offset times is to annotate on the spectrogram of waveform with the aid of audio visualization and musical signal analysis tools such as Sonic Visualiser. This method, however, may not be reliable due to the mismatch between the physical and perceptual offset. For example, human has varied audibility threshold in different pitch frequency ranges. Perceptual limitations, such as simultaneous masking and temporal masking, may also affect. Therefore, a more practical way is to incorporate visualization software and the hearing perception of musicians, despite the cost may be higher. Since their is no procedure for such a perception-based offset annotation, we propose a new one below.

### 3.2 Proposed Offset Annotation Procedure

Considering the perceptual aspects of pitched instruments, the validity of our annotation is based on two assumptions: First, if a note onset and its fundamental frequency (F0) are both retrieved, its offset time is the first moment when the sound intensity level is below the auditory threshold for a certain period of time. Second, for continuous notes, the sound intensity level may always be above the threshold. Therefore, the offset time of preceding note should be exactly or very close to the onset time of the subsequent note unless there are polyphnic notes.

With the aid of a visualization tool such as the Audacity, we propose the following steps for annotating offset times.

1. Remove DC offset (bias) and normalize maximum amplitude to -1.0 dB (software default value). This is done by the "normalize" function in Audacity.
2. Transcribe all identifiable pitches, excluding unstable overtones and unidentifiable sound resulting from playing faults or specific playing techniques (e.g. *flageolet* or *sul ponticello*).
3. Carefully and repeatedly listen to a short part of sound sample as well as zoom in the display of waveform in order to catch the onset position.
4. Identify the position within a pitch where we find the start of "attack" of amplitude envelope in the waveform. The timestamp corresponds to the note onset.
5. Catch the first perceived disappearance (*i.e.* below the audibility threshold) of that given pitch. The corresponding timestamp is the note offset time.
6. For continuous notes, we simply find consequent note onset time and use it as the preceding note offset time. However, in case of a clear note overlapping, we annotate the onset and the offset independently.
7. If the time is still not assured, we play the sound at slower speeds and repeats steps 3–6. This is helpful in estimating note onset or offset precisely.

[1] http://mac.citi.sinica.edu.tw/offset_detection/

| Technique | # of clips | # of offsets |
|---|---|---|
| *Pizzicato* | 13 | 144 |
| *Spiccato* | 5 | 168 |
| *Sordino* | 10 | 539 |
| *Flageolet* | 8 | 48 |
| *Sul tasto* | 12 | 140 |
| *Sul ponticello* | 15 | 187 |
| Total | 63 | 1,226 |

**Table 1**. Detailed information of the proposed dataset.

### 3.3 Proposed Dataset

The dataset contains 63 violin solo excerpts with a total of 1,226 notes derived from the YouTube video clips in [27] and several sound clips from the website "CompositionToday.com" [1]. This dataset, however, does not include information about music score, fingering, dynamics, vibrato, recording environment acoustics, etc. The excerpts covers 6 playing techniques, namely *flageolet* (harmonic), *pizzicato* (pluck the string), *sordino* (mute), *spiccato* (bounce the bow), *sul ponticello* (bow nearing the bridge) and *sul tasto* (bow nearing the fingerboard), all of which are widely used in orchestration [2]. These techniques produce various patterns of temporal envelopes, thereby providing a practical reference set for evaluating offset detection algorithms. Detailed information about the number of clips and notes for each playing techniques is listed in Table 1. We consider these techniques because the dataset is intended to be used as an extension of our previous work [27]. For more comprehensive experiments, people need to include more playing techniques such as legato and detache.

We hired a professional musician to annotate the dataset. The musician has profession-level training in music school and has more than 20 years of experience in playing musical instruments. He also has long experience in composing string quartet and orchestral work, and in sound mixing and recording technology. From the musician's feedback, finishing a precise note annotation and double-check costs 1 to 2 minutes through the above process.

## 4. METHOD

In our study, features are extracted from three different aspects of music, including fundamental frequency, energy envelope and magnitude spectrum. We evaluate the three aspects separately to investigate their feasibility for offset detection. Revising a few previous approaches for onset detection based on these aspects, we discuss five possible offset detection algorithms in the following subsections.

### 4.1 Fundamental frequency

In what follows, we denote $f_{0n}$ as the fundamental frequency at the frame index $n$. The corresponding MIDI number $m_n$ can be obtained by the relation $m_n = \lfloor 12 \cdot \log_2(f_{0n}/440) \rceil + 69$.

We adopt the spectral-domain YIN algorithm [9] to estimate the fundamental frequency. The algorithm reduces the computation complexity of the original, time-domain YIN algorithm [12], and can produce efficient and robust estimate of fundamental frequency. It estimates the fundamental frequency by finding the minimum of the tapered square difference function $d_n(\tau)$ below a certain threshold. The function $d_n(\tau)$ is formulated as:

$$d_n(\tau) = \frac{2}{N} \sum_{k=0}^{N/2+1} |(1 - e^{2j\pi k\tau/N})\mathbf{X}_n(k)|^2 , \quad (1)$$

where $\tau$ is the time lag, and $\mathbf{X}_n(k)$ is the short-time Fourier transform (STFT) spectrum at frame index $n$. The window size of STFT is set to $N = 2048$ in our implementation.

The minimum of Eq.(1) indicates the periodicity. The smaller the $d_n(\tau)$ is, the higher the confidence that the input signal has a fundamental frequency at $1/\tau$. Conversely, if $d_n(\tau)$ is too high then the input signal is considered non-pitched. The fundamental frequency $f_0$ is represented as:

$$f_{0n} = \left( \arg\min_{\tau} d_n(\tau) \right)^{-1} \text{s.t.} \quad 1 - d_n(\tau) > \delta_c . \quad (2)$$

We consider the term $c_n = 1 - \min d_n(\tau)$ as the *pitch confidence*; as it measures whether an input is periodic and therefore can determine whether it is a pitch signal [9, 25]. In our implementation, we set the pitch to zero (*i.e.* $m_n = 0$) if the confidence is below a threshold $\delta_c$. We set $\delta_c = 0.7$ empirically.

#### *4.1.1 Pitch change*

Pitch change has been known as a useful onset detector for pitched non-percussive instruments like bowed strings, where the input signal is usually excited constantly and exhibits no obvious amplitude or phase variation [11, 17]. Pitch change is a clear indicator of a note transition, which typically contains an offset of the previous note and the onset of the latter note. When the pitch contour changes from one pitch to another, we expect that there should be one note ending and another note starting. We note that the limitation of this idea is that it cannot deal with the case of repeating notes.

Based on the above observation, we propose the following offset detection method using pitch change information. We consider there is an offset event at frame $n$, if the following two rules are satisfied:

$$\text{mod}_{12}(m_n - m_{n-1}) \geq 1 \quad \wedge \quad c_n - c_{n-1} < 0 . \quad (3)$$

Similar to the onset detector proposed in [17], the modulo operator in the first rule is applied to prevent octave errors, although it also hinders the detection of transitions of octave(s). Because $m_n = 0$ when $c_n \leq \delta_c$, the first rule also captures voice/unvoice transition. We also observe that the falling moment of confidence function can indicate the chance of a stable pitch fading that enables us to distinguish offset from onset.

#### *4.1.2 Pitch confidence*

Another perspective is to directly use the pitch confidence function as an offset detector. In this case, errors of pitch

detection would not influence the performance. The basic idea is that the time instant when the pitch confidence changes from pitched to non-pitched is considered as the offset time. Therefore, this method searches for the moment that the pitch confidence falls below the threshold. In other words, there is an offset event at $n$, if the following conditions meet:

$$c_{n-1} > \delta_c \quad \wedge \quad c_n < \delta_c \,. \tag{4}$$

Please note that this method is conceptually similar to the way many NMF-based automatic transcription algorithms detect offsets: they usually detect offsets by thresholding on the activation matrix [29]. While our method uses $c_n$ to measure pitch confidence, NMF-based methods use the value of activation to measure pitch confidence.

### 4.2 Energy envelope

The energy envelope as used by the human auditory system [6] has been proven to be a robust feature in many onset detection tasks [7, 8, 22]. Here we compute the energy-like temporal envelope based on this feature. The pre-processing step starts from raw STFT spectra with frame size 2048, then map into 141 sub-bands by a set of triangular filter bank equally spaced in log-scale ranging from 30 Hz to 17000 Hz. Then, the feature is scaled by the logarithm $x \mapsto \log(1+x)$. Finally, the energy-like envelope is formulated as: $\mathrm{E_n} = \sum_k |\bar{\mathbf{X}}_\mathrm{n}(k)|^2$ , where $\bar{\mathbf{X}}_n(k)$ is the pre-processed spectra magnitude of bin $k$. Since the perceptual offset is a subjective threshold lies between the decaying phase of energy envelope, and in most cases note offset is interrupted by succeeding onset, that make the setting an absolute thresholding infeasible. Therefore, we employ the relative threshold peak-picking algorithm [5] to find the valley of energy envelope as offset.

### 4.3 Spectral flux

Spectral flux is one of the most common, easy-to-implement yet powerful methods for onset detection [3, 7]. It can be formulated as: $\mathrm{SF_n} = \sum_k H(|\bar{\mathbf{X}}_n(k)| - |\bar{\mathbf{X}}_{n-1}(k)|)$, where $H(x) = \frac{|x|+x}{2}$ is the rectifier function, and the pre-processed spectral bins $\bar{\mathbf{X}}_n(k)$ are the ones that are described in Section 4.2.

We are interested in whether the idea of spectral flux can be adopted for offset detection. Two *reversed* variants of spectral flux are considered:

- **Reverse rectification** ($\mathrm{SF_{rr}}$)**:** The rectifier function $H$ in onset detection selects only the positive flux while suppresses the negative flux. Conversely, for offset detection, $H$ is replaced by $H' = \frac{|x|-x}{2}$, which suppress all the positive flux.
- **Reverse coding** ($\mathrm{SF_{rc}}$)**:** The other setting is to compute spectral flux in the opposite direction, *i.e.*, from the future to the past: $\sum_k H(|\bar{\mathbf{X}}_n(k)| - |\bar{\mathbf{X}}_{n+3}(k)|)$, to reverse the raw audio signal, and apply the normal spectral flux method to the reversed signal as looking for onset in the opposite direction.

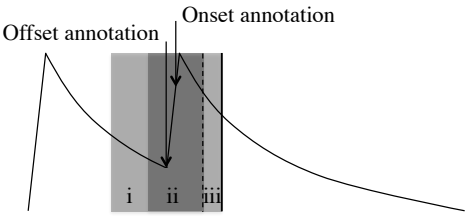



**Figure 2**. A case when an offset and its succeeding onset are very close, the right margin of the tolerance window for offset (the solid line) might fall behind the right margin of the tolerance window for onset (the dash line) of onset. Regions i–iii are all within the tolerance window for offset, while region ii is within the tolerance window for onset.

## 5. EVALUATION

### 5.1 Onset-aware evaluation metric

We employ the standard measures for evaluation: precision, recall and F-score. In evaluation, the offset estimate that falls within a tolerance window of length $2\delta_{\text{tolerance}}$ of the groundtruth offset time is considered to be a true positive. Moreover, the estimate and the groundtruth can only be matched at most once, based on maximum cardinality bipartite matching [23]. The remaining estimates are considered false positives. The tolerance window (centered at the groundtruth annotation) can be written as $\Delta W = [-\delta_{\text{tolerance}}, +\delta_{\text{tolerance}}]$. This is referred to as the *conventional* tolerance window.

A typical problem of this evaluation method is depicted in Fig. 2. As mentioned in Section 1, the tolerance window for offset detection is often set to be wider than that for onset detection in most previous work. [2] In Fig. 2, the right margin of the tolerance window for offset of the current note (*i.e.* the solid line in Fig. 2) falls behind the right margin of the tolerance window for onset of the succeeding note (*i.e.* the dash line in Fig. 2). Such a situation occurs for more than 80% of notes in our dataset, when $\delta_{\text{tolerance}}$ is set to 100ms. If the offset is annotated given the transition offset annotation rule that we suggest, region iii should not be considered as a possible true positive area.

In light of this observation, we further define a new tolerance window by $\Delta W' = [-\delta_{\text{tolerance}}, +\delta_{\text{post\_tolerance}}]$, making $\delta_{\text{post\_tolerance}}$ dependent on the succeeding onset. In this paper, we set $\delta_{\text{post\_tolerance}} = \min(\delta_\mathrm{t}+50\mathrm{ms}, \delta_{\text{tolerance}})$, where $\delta_\mathrm{t}$ denotes the timestamp of the next onset, and 50ms is a commonly adopted value for $\delta_{\text{tolerance}}$ for onset.

To give a deep insight of the onset-aware tolerance window, let's first consider this: if the offset and succeeding onset are located far apart, the post tolerance would be the same as the conventional tolerance, so the evaluation result will be the same as the result of the conventional metric. But, as the distance becomes closer, post tolerance will shrink to the tolerance of succeeding onset when they are fully overlapped, resulting in a shortened tolerance win-

[2] http://www.music-ir.org/mirex/wiki/2014:Multiple_Fundamental_Frequency_Estimation_\%26_Tracking

| Playing technique | Performance measure | Pitch confidence | | Pitch change | | Energy | | $SF_{rc}$ | | $SF_{rr}$ | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | $M_A$ | $M_B$ | $M_A$ | $M_B$ | $M_A$ | $M_B$ | $M_A$ | $M_B$ | $M_A$ | $M_B$ |
| *Pizzicato* | F-score | 0.689 | 0.671 | 0.578 | 0.557 | 0.695 | 0.695 | 0.576 | 0.556 | 0.587 | 0.567 |
| *Spiccato* | | 0.778 | 0.724 | 0.740 | 0.687 | 0.759 | 0.759 | **0.610** | **0.308** | **0.584** | **0.271** |
| *Sordino* | | 0.727 | 0.718 | 0.701 | 0.686 | 0.555 | 0.512 | 0.650 | 0.598 | 0.652 | 0.596 |
| *Flageolet* | | 0.381 | 0.381 | 0.321 | 0.301 | 0.414 | 0.402 | 0.292 | 0.262 | 0.290 | 0.254 |
| *Sul tasto* | | 0.531 | 0.522 | 0.544 | 0.524 | 0.463 | 0.433 | 0.448 | 0.433 | 0.440 | 0.424 |
| *Sul ponticello* | | 0.522 | 0.518 | 0.44 | 0.434 | 0.338 | 0.309 | 0.314 | 0.302 | 0.310 | 0.299 |
| Overall | Precision | 0.688 | 0.673 | 0.514 | 0.498 | 0.422 | 0.398 | 0.364 | 0.326 | 0.361 | 0.320 |
| | Recall | 0.623 | 0.609 | 0.669 | 0.648 | 0.639 | 0.604 | 0.758 | 0.677 | 0.759 | 0.674 |
| | F-score | **0.654** | **0.640** | 0.582 | 0.563 | 0.508 | 0.480 | 0.492 | 0.440 | 0.489 | 0.434 |

**Table 2**. Comparison of evaluation metrics to offset detection methods. $M_A$: the conventional evaluation metric. $M_B$: the proposed onset-aware evaluation metric.

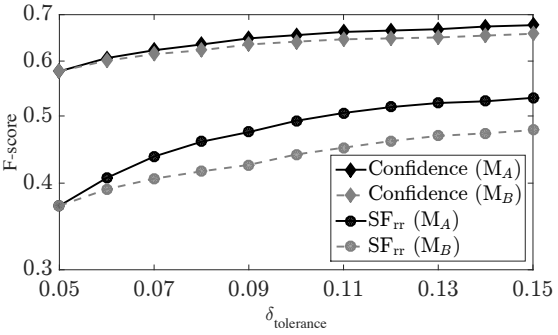



**Figure 3**. Comparison of evaluation using conventional metric (solid line) and proposed onset-aware metric (dash line) on two offset detection methods. Horizontal axis shows the tolerance $\delta_{\text{tolerance}}$ (ranging from 50ms to 150ms), and vertical axis shows average F-score.

dow without region iii. In other words, the right margin of the tolerable window for offset would not exceed the right margin of the tolerable window for the succeeding onset.

### 5.2 Experiment result

Table 2 shows the evaluation of detection algorithms performed by both metrics. First of all, we see that pitch-based methods significantly outperform the others in the overall result according to both metrics. This is perhaps not surprising, given that pitch-based methods have been shown effective for onset detection for notes with slow attack phase. For example, Holzapfel *et al.* [17] have shown that pitch-based methods work much better than SF-based methods for onset detection for bow-string instrument and wind instrument. The decaying phase exhibits similar signal characteristics as soft onsets when "looking reversely" from the end of the signal. This may explain why pitch-based methods also work better than SF-based methods for offset detection.

We expect that the result of onset-aware evaluation (using $\Delta W'$) would be equal or less than conventional metric (using $\Delta W$). The interesting finding is that, while most methods we considered have similar result for the two evaluation metrics, the result of SF-based methods degrades a lot when the onset-aware metric is adopted. For the overall result, the result of the two SF methods decreases by 11% and 13%, respectively. The most severe degradation is seen in *spiccato*. This result indicates that SF-based methods may be prone to produce many estimations within region iii of Fig. 2.

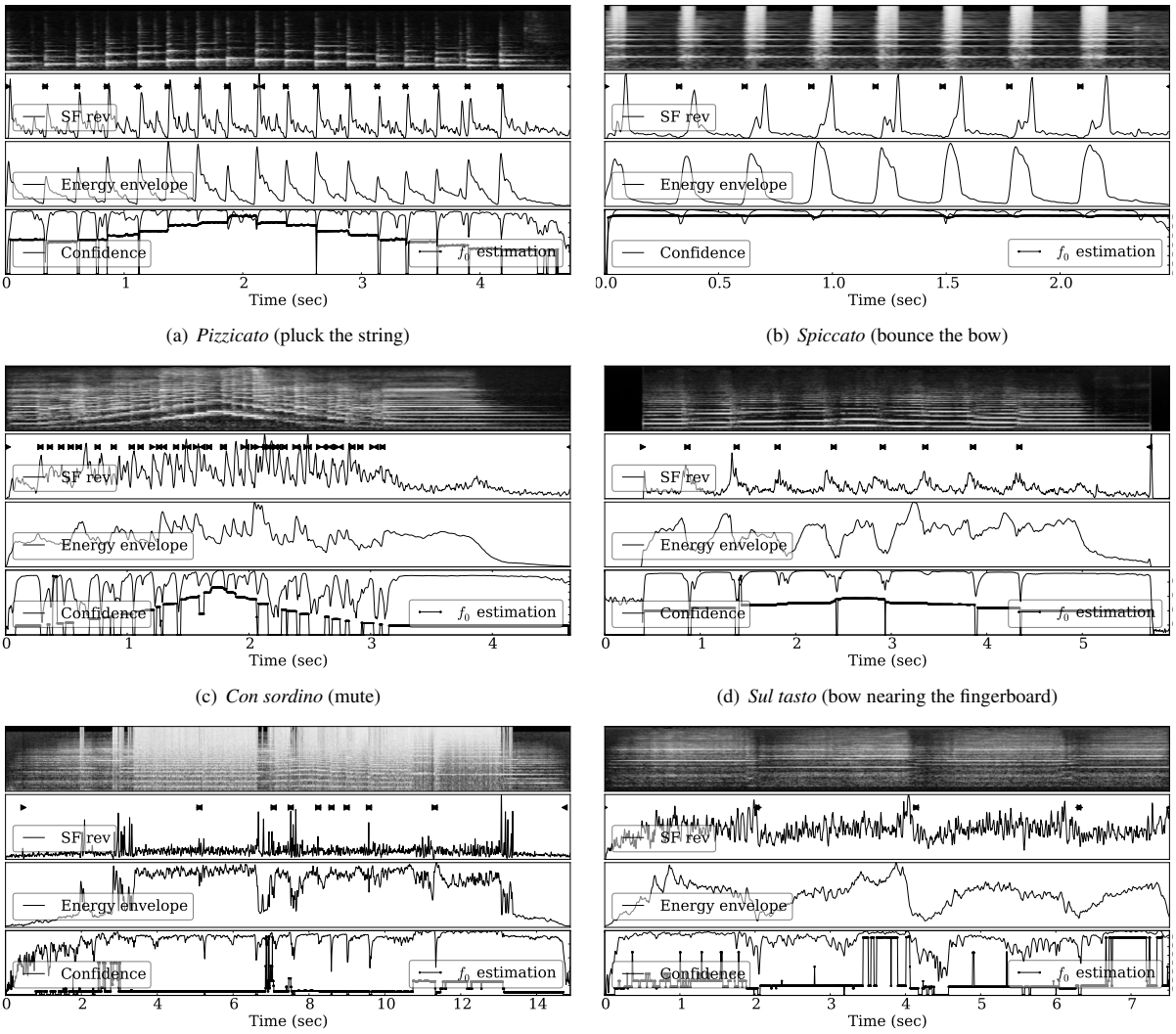
Fig. 3 compares the result of the pitch confidence method and the $SF_{rr}$ methods using the two metrics. As it will be shown later in Section 5.3, spectral flux exhibits temporal alignment issues while the pitch confidence method does not. It can be seen that the pitch confidence method does not suffer from the penalty of proposed metric while $SF_{rr}$ does. We note that the bipartite matching mechanism we adopted may have also avoided some of the estimation inside region iii of Fig. 2. But, by using the proposed metric, we can ensure region iii is fully eliminated. This is important because the conventional metric may give us over-optimistic result.

Another important finding is that the pitch confidence method consistently outperforms the pitch change method, when the onset-aware metric is adopted. Results show that the pitch change method has higher recall but much lower precision, possibly due to the fluctuation of confidence above and below threshold causes some false alarms. It is possible to mitigate the issue by proper post-processing, such as by padding the continuous note or using median filter, but if the pitch confidence method is employed we do not have to deal with such an issue.

### 5.3 Illustration

The upper part of Fig. 4 shows the spectrogram and the offset detection functions of *pizzicato* and *spiccato*. [3] Though both techniques produce sound by pulse-like excitation, we can see the envelope of *spiccato* is much smoother than spiccato in terms of attack and decay phase possibly, because of the elasticity of bow cause the striking contacts the string slightly longer (*i.e.* leading to longer sustain) than the plucking string. SF based methods typically take the the beginning of decay as the offset position, as shown in Fig. 2, while the estimates of other methods appear to be closer to the ground truth. SF-based methods are prone to produce temporal detection errors largely in *pizzicato* and *spiccato*, making the conventional evaluation metric for onset less appropriate for evaluate the result for offset. However, some estimations of *pizzicato* is a lot earlier than

[3] We only put one of the spectral flux based methods due to their high similarity of detection curve.




(a) *Pizzicato* (pluck the string)

(b) *Spiccato* (bounce the bow)

(c) *Con sordino* (mute)

(d) *Sul tasto* (bow nearing the fingerboard)

(e) *Sul ponticello* (bow nearing the bridge)

(f) *Flageolet* (harmonic)

**Figure 4**. Comparison of the signal characteristic of six playing techniques. From top to bottom are spectrogram, spectral flux, energy envelope, and pitch-based offset detection curves. The right-pointing triangle denotes the onset annotation and the left-pointing triangle denotes the offset annotation.

*spiccato* that it is even located within the onset tolerance (*i.e.* region ii). In that extreme situation, we may have to shorten the onset tolerance by a few milliseconds in our evaluation metric.

The low part of Fig. 4 shows the other four bowing techniques. For the lower two techniques, all of the detection functions exhibit the fluctuating curve due to the noise-like overtones, leading to inferior result for *sul ponticello* and *flageolet*. In such case, energy relatively remains in the the same level of performance. On the other hand, from the middle part of Fig. 4, we can see the pitch confidence is still a good indicator of offset for *con sordino* and *sul tasto*.

## 6. CONCLUSION

In this paper, we have discussed the challenges of offset detection, the methodology of constructing an offset detection dataset, some detection algorithms, and a few considerations in evaluation. Based on the newly constructed violin dataset, we have firstly investigated the behaviors of musical offsets in the signals generated by various kinds of mechanism. We find that, in general, the pitch confidence based offset detection function outperforms algorithms based on energy and spectral flux. For the playing techniques having sharp envelopes such as *pizzicato* and *spiccato*, energy-based method can be competitive. We have also proposed an onset-aware evaluation metric that is more reliable than the conventional ones in avoiding overestimation of true positives. We hope that these findings can contribute to the advance of research on automatic music transcription and melody tracking.

## 7. ACKNOWLEDGMENT

This work was supported by the Ministry of Science and Technology of Taiwan under the contracts MOST 102-2221-E-001-004-MY3, MOST 104-2221-E-001-029-MY3, and the Academia Sinica Career Development Program.

## 8. REFERENCES


[1] Compositiontoday.com - sound bank – violin. http://www.compositiontoday.com/sound_bank/violin/.

[2] S. Adler. *The Study of Orchestration–3rd Edition*. WW Norton, 2002.

[3] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler. A tutorial on onset detection in music signals. *IEEE Trans. Speech Audio Proc.*, 13(5):1035–1047, 2005.

[4] E. Benetos and S. Dixon. Polyphonic music transcription using note onset and offset detection. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Proc.*, pages 37–40. IEEE, 2011.

[5] S. Böck, A. Arzt, F. Krebs, and M. Schedl. Online real-time onset detection with recurrent neural networks. In *Proc. Int. Conf. Digital Audio Effects*, pages 1–4, 2012.

[6] S. Böck, F. Krebs, and M. Schedl. Evaluating the online capabilities of onset detection methods. In *Proc. Int. Soc. Music Information Retrieval Conf.*, pages 49–54, 2012.

[7] S. Böck and G. Widmer. Maximum filter vibrato suppression for onset detection. In *Proc. Int. Conf. Digital Audio Effects*, 2013.

[8] Sebastian Böck and Florian Krebs. MIREX onset detection task. In *Music Information Retrieval Evaluation eXchange*, 2012. [Online] http://www.music-ir.org/mirex/abstracts/2012/BK2.pdf.

[9] P. M. Brossier. *Automatic annotation of musical audio for interactive applications*. PhD thesis, Queen Mary, University of London, 2006.

[10] S. Chang and K. Lee. A pairwise approach to simultaneous onset/offset detection for singing voice using correntropy. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Proc.*, pages 629–633. IEEE, 2014.

[11] N. Collins. Using a pitch detector for onset detection. In *Proc. Int. Soc. Music Information Retrieval Conf.*, pages 100–106, 2005.

[12] A. De Cheveigné and H. Kawahara. Yin, a fundamental frequency estimator for speech and music. *The Journal of the Acoustical Society of America*, 111(4):1917–1930, 2002.

[13] A. Degani, R. Leonardi, P. Migliorati, and G. Peeters. A pitch salience function derived from harmonic frequency deviations for polyphonic music analysis. In *Proc. Int. Conf. Digital Audio Effects*, 2014.

[14] J. Devaney, M. I. Mandel, and I. Fujinaga. A study of intonation in three-part singing using the automatic music performance analysis and comparison toolkit (AMPACT). In *Proc. Int. Soc. Music Information Retrieval Conf.*, pages 511–516, 2012.

[15] A. Friberg, E. Schoonderwaldt, and P. N. Juslin. Cuex: An algorithm for automatic extraction of expressive tone parameters in music performance from acoustic signals. *Acta acustica united with acustica*, 93(3):411–420, 2007.

[16] J. Glover, V. Lazzarini, and J. Timoney. Real-time segmentation of the temporal evolution of musical sounds. In *Proc. Meetings on Acoustics*, volume 15. Acoustical Society of America, 2014.

[17] A. Holzapfel, Y. Stylianou, A. C. Gedik, and B. Bozkurt. Three dimensions of pitched instrument onset detection. *IEEE Trans. Audio, Speech, Language Proc.*, 18(6):1517–1527, 2010.

[18] L.-C. Hsu, Y.-L. Wang, Y.-J. Lin, C. D. Metcalf, and A. W.-Y. Su. Detection of motor changes in violin playing by emg signals. In *Proc. Int. Soc. Music Information Retrieval Conf.*, 2014.

[19] G. Hu and D. Wang. Auditory segmentation based on onset and offset analysis. *IEEE Trans. Audio, Speech, Lang. Proc.*, 15(2):396–405, 2007.

[20] C. Kehling, J. Abeßer, C. Dittmar, and G. Schuller. Automatic tablature transcription of electric guitar recordings by estimation of score- and instrument-related parameters. In *Proc. Int. Conf. Digital Audio Effects*, 2014.

[21] A. Kobzantsev, D. Chazan, and Y. Zeevi. Automatic transcription of piano polyphonic music. In *Proc. Int. Symp. Image and Signal Processing and Analysis*, pages 414–418, 2005.

[22] E. Marchi, G. Ferroni, F. Eyben, L. Gabrielli, S. Squartini, and B. Schuller. Multi-resolution linear prediction based features for audio onset detection with bidirectional lstm neural networks. In *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, pages 2164–2168, 2014.

[23] C. Raffel, B. McFee, E. J. Humphrey, J. Salamon, O. Nieto, D. Liang, and D. P. W. Ellis. mir_eval: A transparent implementation of common MIR metrics. In *Proc. Int. Soc. for Music Information Retrieval Conf.*, 2014.

[24] J. Salamon and E. Gómez. Melody extraction from polyphonic music signals using pitch contour characteristics. *IEEE Trans. Audio, Speech, Lang. Proc.*, 20(6):1759–1770, 2012.

[25] J. Serra, G. K. Koduri, M. Miron, and X. Serra. Assessing the tuning of sung indian classical music. In *Proc. Int. Soc. Music Information Retrieval Conf.*, pages 157–162, 2011.

[26] R. Sridhar and T. V. Geetha. Raga identification of carnatic music for music information retrieval. *International Journal of recent trends in Engineering*, 1(1):571–574, 2009.

[27] L. Su, H.-M. Lin, and Y.-H. Yang. Sparse modeling of magnitude and phase-derived spectra for playing technique classification. *IEEE/ACM Trans. Audio, Speech and Language Proc.*, 22(12):2122–2132, 2014. [Online] http://mac.citi.sinica.edu.tw/violin-playing-technique/.

[28] L. Su and Y.-H. Yang. Power-scaled spectral flux and peak-valley group-delay methods for robust musical onset detection. In *Proc. Sound and Music Computing Conf.*, 2014.

[29] E. Vincent, N. Bertin, and R. Badeau. Enforcing harmonicity and smoothness in bayesian non-negative matrix factorization applied to polyphonic music transcription. *IEEE Trans. Audio, Speech, Lang. Proc.*, 18(3):528–537, 2010.