A Change Discrimination Onset Detector with Peak Scoring Peak Picker and Time Domain Correction

Nick Collins
Centre for Music and Science
Faculty of Music
University of Cambridge
11 West Road
Cambridge
CB3 9DP
nc272@cam.ac.uk http://www.cus.cam.ac.uk/ nc272/

ABSTRACT

An onset detector is described based on the most successful onset detector for non-pitched percussive audio events from an earlier comparative study (Collins, 2005a). The detection function is an adaptation of the log intensity difference change discrimination originally introduced by Anssi Klapuri (Klapuri, 1999). A novel peak picking method is used based on scoring the most salient peaks with respect to the local function terrain. Discovered onset positions are corrected using parallel finer resolution time domain methods. The implementation is much faster than realtime and causal, being a conversion of code released for a real-time computer music system. In the MIREX evaluation, across both percussive and non-percussive onsets, the algorithm performed well on the former but was not successful on the latter. It was the most efficient and had the best overall time localisation of onsets of the algorithms in the contest.

1 Algorithm Overview

A real-time causal onset detector had been developed for live computer music use, using the most successful onset detector for non-pitched percussive (NPP) samples from a comparative study (Collins, 2005a). This algorithm was adapted as an entry for the MIREX 2005 Audio Onset Detection contest. Figure 1 gives an overview of the processing steps in the algorithm.

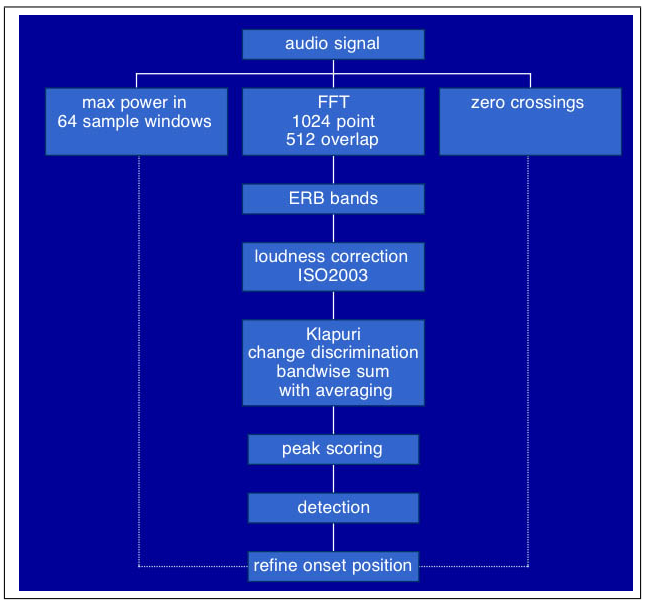

Figure 1: Overview of the algorithm

2 Detection Function

The detection function is an adaptation of the log intensity difference change discrimination originally introduced by Anssi Klapuri (Klapuri, 1999). Processing is FFT based and involves an intensity to decibel transform in ERB scale bands with equal loudness contour correction and a general averaged difference over the last three frames (Collins, 2005a). The onset detector calculates a 1024 point FFT with hop size of 512, assuming target 44100Hz audio.

3 Peak Picker

This peak picking algorithm was inspired by the global visual peak picking possible by a human operator in an audio editor. Whilst I have kept the function local in basis, commensurate with fast causal onset detection, extensions can be envisaged to widen the scope, and perhaps tradeoff local with global trends in the detection function (the essential problem of peak picking being recognising a local variation as a significant change or just as noise).

Figure 3 gives pseudo code for a peak picking algorithm which scores local peaks over a seven frame window. The input detection function $df(i)$ has been normalised to the range 0 to 1 (from a prediction of typical intensities). Evidence that a given point is below any other in this window leads to a large penalty, and the amount of excess over all other local points is the factor of concern. A threshold is then set for detections; a value of 0.34 was empirically determined in previous evaluation tests as the best performing across the NPP test set used in Collins (2005a).

Peaks are required to have a minimum separation of 3 FFT frames.

4 Time Domain Correction

To improve time resolution, maximum intensities are taken in the time domain in 64 sample blocks, in paral-

Figure 2: Pseudocode for peak picking

```
For all frames i=1 to N
score=0
For j=i-3 to i+3
        temp=df(i)-df(j)
        if (temp<0.0) temp=temp*10
        score= score+temp
        if ((score<threshold) AND (time since last event > minimum event separation)) onset detected
```

lel to the FFT. A discovered onset position is corrected to a local minima of this function within 16 blocks prior to the discovered onset (ie, within those samples collected for the current triggering FFT frame). This sample position is further corrected to a nearby zero crossing (or intensity minima) for smooth segmentation, within the previous 441 samples.

5 Implementation

Whilst this implementation is not an 'as fast as possible' reacting onset detector (due to the three frame averaging and seven frame peak picking method), it is causal, and useful for real-time event analysis. The algorithm has been publicly released in the bbcut2 library for SuperCollider 3 (available from the author's web site) as part of an on-the-fly event analysis system. The command line C code executable (Mac OS X Altivec only with libsndfile) submitted for the evaluation task is a direct conversion of the SuperCollider UGen built to enable on-the-fly event analysis, removing the offset detection, and runs causally, much faster than realtime with a latency of four FFT frames[1]

6 Evaluation Prediction

When submitting the algorithm to the contest I made the following formal predictions:

Whilst the detection function utilised scored highly on the NPP task, it fared much worse on the pitched non-percussive (PNP) test case (Collins, 2005a). This is more thoroughly critiqued in a further under review study (Collins, 2005b). It is expected in the context of the MIREX evaluation that the procedure be relatively effective at NPP and perhaps transient heavy polyphonic audio, but that it fail on PNP cases like the singing voice, confounded by AM associated with vibrato to produce many false positives.

A more general procedure might assess the target for stability of pitch percept, probably based in instrument recognition work (Collins, 2005b). Only for percussive transients would the change discrimination process above be the segmentor. However, I am curious to see the performance of the algorithm proposed herein, and submit it to the competition in the knowledge that its performance on some instrumental cases like strings will be substantially worse.

[1]This is still too long for an as fast as possible onset detector, being perceptibly late by 46mS. In fact, because of perceptual attack time properties, even a 5mS latency onset detector used as a trigger may be perceptibly late with respect to a triggering event and it is perhaps unreasonable to seek such a reactive solution; a human would anticipate to achieve synchronisation

7 Evaluation

The MIREX results (`http://www.music-ir.org/evaluation/mirex-results/audio-onset/index.html`) bore out these predictions somewhat. Table 1 summarises the overall results, though the reader is referred to the web site for a more extensive breakdown by classes of target sound and the results for another 6 algorithms. The evaluation test set consisted of 85 files across 9 classes, totaling 14.8 minutes of audio.

In overall terms my algorithm came mid-table, chasing a pack of similar F-measure achieving algorithms. It was fastest overall (running at 74 times faster than real-time) by at least a factor of four, though this is not to say that other implementations, which may for instance have been written in MATLAB rather than C, could not be made more efficient[2]. It had been specifically optimised for real-time performance use and used the Altivec routines to speed up the FFT calculations. It was also the most accurate in overall time resolution of onset positions, though only a few milliseconds more accurate than some rivals. On average, it detected onsets 1 ms earlier than the annotated onset positions.

Table 2 gives a breakdown of results across classes. As predicted, the algorithm performed well on percussive onsets (and most of the algorithms scored highly here). As also predicted, performance was substantially degraded on slow strings and singing voice (and the best results for these two cases across algorithms gave F-measure scores of 57.92% and 45.33% respectively). Other cases were intermediate.

One curiosity is that the algorithm's performance on the sustained strings showed many false negatives rather than false positives, against prediction. This is perhaps most likely traceable to annotations at perceptual attack times well after the physical onset of the sound (The log difference detection function tends to fire nearer the latter), and/or the threshold setting of the algorithm, which could have risked more false positives to remove some false negatives.

The doubled onsets score was caused by my failure to set a high enough number of frames required between successive detections, and had already been fixed in the

[2]The different machines used for assessments may also have some bearing on these results.

algorithm	F-measure	precision	recall	total correct	total FP	total FN	total merged	total doubled	mean abs distance	speed (s)
1. Lacoste & Eck	80.07%	79.27%	83.70%	7974	1776	1525	210	53	0.0115	4713
6. Collins	72.10%	87.96%	68.26%	6174	629	3325	168	35	0.0069	12
9. West	48.77%	48.50%	56.29%	5424	7119	4075	146	0	0.0138	179

Table 1: Overall results for the algorithm compared to top and bottom of the table (summary)

class	num files	ranking (of 9)	F-measure	precision	recall	total correct	total FP	total FN	total merged	total doubled
Solo Bars and Bells	4	1	99.28%	98.91%	99.67%	321	3	3	0	0
Solo Drum	30	1	92.31%	95.92%	90.28%	2668	86	240	51	3
Solo Plucked String	9	3	81.97%	77.78%	88.09%	380	136	51	7	9
Poly Pitched	10	6	75.70%	89.95%	69.98%	570	54	289	19	0
Solo Brass	2	3	69.09%	71.71%	67.26%	170	40	43	0	8
Complex	15	6	60.25%	86.14%	51.77%	1878	212	1681	87	13
Solo Wind	4	6	47.57%	81.71%	35.40%	96	63	170	1	2
Solo Singing Voice	5	5	29.34%	59.44%	19.85%	44	28	185	1	0
Solo Sustained Strings	6	9	14.74%	90.74%	8.47%	47	7	663	2	0

Table 2: Breakdown over classes

public algorithm available with bbcut2, but not in the algorithm submitted for this contest.

8 Discussion

The algorithms in competition included many variations of Klapuri's psychoacoustically motivated onset detection (Klapuri, 1999), an algorithm whose good qualities were exhibited in a previous comparison of onset detection algorithms (Collins, 2005a). Differences in performance across algorithms are traceable in many respects to threshold parameters chosen to control the tradeoff between false positives and false negatives. All algorithms could surely be improved by optimising this balance with respect to the test set to achieve the best F-measure scores. It is clear in the case of the algorithm I submitted that the algorithm was most likely too conservative in firing. It had however been optimised on a different test set of mostly drum sounds, and it is gratifying to see it perform well on this class, thus demonstrating some freedom from overfitting problems.

The winning algorithm used a machine learning strategy to find the best detection function (one might also use this principle to find the best peak picker), following (Marolt et al., 2002; Kapanci and Pfeffer, 2004). I imagine that performance could be improved further by appropriate auditory frontends for given tasks- the Marolt et al paper is influenced by Smith's work (Smith, 1994, 2001) and further human hearing-like signal processing frontends (Moelants and Rampazzo, 1997) may be investigated, particularly where the segmentation tasks require the discovery of onsets as a human listener would judge music, as for the difficult sustained string and singing cases. Certainly, the efficacy of machine learning techniques to control the awkward parameter optimisation problems occurring in this area is reinforced by this contest. Other algorithms in the contest could no doubt be improved by introducing such techniques, though I suspect the choice of auditory frontend will remain a critical factor. Future algorithms may also use entirely different schemes for different classes of sound event, decided by instrument recognition.

9 Conclusions

The algorithm I submitted performed as predicted, working effectively for percussive sounds but failing on non-percussive onsets, particularly for sustained strings and voice. The reasons for this are further discussed in (Collins, 2005b), where an alternative approach specialised to pitched material is advanced.

ACKNOWLEDGEMENTS

This research is supported by AHRC grant 2003/104481. Many thanks are due to the MIREX testing group and coordination team for all their hard work in overseeing this contest.

References

Nick Collins. A comparison of sound onset detection algorithms with emphasis on psychoacoustically motivated detection functions. In *AES Convention 118*, Barcelona, May 28-31 2005a.

Nick Collins. Using a pitch detector as an onset detector. In *Proc. Int. Symp. on Music Information Retrieval*, London, Sept 11-15 2005b.

Emir Kapanci and Avi Pfeffer. A hierarchical approach to onset detection. In *Proc. Int. Computer Music Conference*, Miami, Florida, October 2004.

Anssi Klapuri. Sound onset detection by applying psychoacoustic knowledge. In *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc. (ICASSP)*, pages 3089–92, 1999.

Matija Marolt, Alenka Kavcic, and Marko Privosnik. Neural networks for note onset detection in piano music. In *Proc. Int. Computer Music Conference*, Gothenberg, Sweden, 2002.

D. Moelants and C. Rampazzo. A computer system for the automatic detection of perceptual onsets in a musical signal. In Antonio Camurri, editor, *KANSEI, The Technology of Emotion*, pages 140–146, Genova, 1997.

Leslie S. Smith. Sound segmentation using onsets and offsets. *Journal of New Music Research*, 23:11–23, 1994.

Leslie S. Smith. Using depressing synapses for phase locked auditory onset detection. In *Int. Conf. on Artificial Neural Networks - ICANN 2001, Lecture Notes in Computer Science 2130 (Springer)*, 2001.