EVALUATION OF MULTIPLE-F0 ESTIMATION AND TRACKING SYSTEMS

Mert Bay **Andreas F. Ehmann** **J. Stephen Downie**
International Music Information Retrieval Systems Evaluation Laboratory
University of Illinois at Urbana-Champaign
{mertbay, aehmann, jdownie}@illinois.edu

ABSTRACT

Multi-pitch estimation of sources in music is an ongoing research area that has a wealth of applications in music information retrieval systems. This paper presents the systematic evaluations of over a dozen competing methods and algorithms for extracting the fundamental frequencies of pitched sound sources in polyphonic music. The evaluations were carried out as part of the Music Information Retrieval Evaluation eXchange (MIREX) over the course of two years, from 2007 to 2008. The generation of the dataset and its corresponding ground-truth, the methods by which systems can be evaluated, and the evaluation results of the different systems are presented and discussed.

1. INTRODUCTION

A key aspect of many music information retrieval (MIR) systems is the ability to extract useful information from complex audio, which may then be used in a variety of user scenarios such as searching and organizing music collections. Among these extraction techniques, the goal of multiple fundamental frequency (multi-F0) estimation is to extract the fundamental frequencies of all (possibly concurrent) notes within a polyphonic musical piece. The extracted representations usually either take the form of a 1) list of pitches vs. time; or, 2) a MIDI-like representation that contains individual notes and their onset and offset times. These representations represent an intermediary between the audio and the score. While automatic transcriptions systems concern themselves with generating the actual score of music being analyzed, the intermediate representation generated by multi-F0 systems is useful in its own right. Such information can be very useful for other MIR systems as higher level features: to define the structure of the song, to make a better search or recommendation based on the score, or for F0-guided source separation. Recently, there has been great interest in multi-F0 estimation.

To understand the current state of art, starting in 2007, the MIREX [3] organized a multi-F0 evaluation task. This task can be considered as an evolution and superset of the previous MIREX audio melody extraction tasks. For more information on audio melody extraction, we refer the reader to [13]. The MIREX multiple-F0 task consists of two subtasks built around the two pitch representations mentioned earlier. The first subtask is called *Multiple-F0 Estimation* (MFE). In MFE, systems are required to return a list of active pitches at fixed time steps (analysis frames) of a polyphonic recording. The second subtask is called *Note Tracking* (NT). In the NT subtask, systems are required to return the note F0, onsets and offsets of note events in the polyphonic mixture, similar to a piano-roll representation.

The MIREX multiple-F0 task attracted many researchers from around the world. In the 2007 MFE subtask, there were a total of 16 algorithms from 12 labs. For the NT subtask, there were 11 algorithms from 7 labs. In 2008, there were a total of 15 algorithms from 10 labs for MFE and 13 algorithms from 8 labs for NT.

This paper serves to discuss the current performance of multi-F0 systems and to analyze the results of the MIREX algorithm evaluations. The paper is organized as follows. The rest of Section 1 describes the main approaches and challanges to MFE and NT. Section 2 describes the evaluation process. Section 2.1 describes the dataset and Section 2.2 defines the evaluation metrics. Section 3 discusses the results and some approaches from the MIREX 2007 and 2008 MFE and NT subtasks. Section 4 provides some concluding remarks.

1.1 An Overview of Multiple-F0 Estimation and Note Tracking Methods

There are many methods for F0 estimation and note tracking and an in-depth coverage of the many possible techniques is beyond the scope of this paper. Instead, we will provide a very brief overview of methods. Table 1 shows the participants of the MIREX 2007 and 2008 MFE and NT subtasks and their proposed methods. All systems use a time-frequency representation of the input signal as a front-end. The time-frequency representations include short-time Fourier transforms [1,2,6,10,11,13,15], auditory filter banks [16, 17], wavelet decompositions [5] and sinusoidal analysis [18]. Characteristics of the spectrum such as harmonicity [5, 10, 14, 17, 19], spectral smoothness [11], onset synchronicity of harmonics [18] are often used to extract

F0s either by grouping harmonics together or calculating scores for different F0 hypotheses.

A large cross-section of techniques use nonnegative matrix factorization (NMF) to decompose the observed magnitude spectrum into a sparse basis. Fundamental frequencies can then be determined for each basis vector, and the onsets/offsets are computed from the amplitude weight of each basis throughout a piece. Some systems follow classification approaches which attempt to find pre-trained notes in the mixture. In general, it is possible to categorize the methods used into two groups in terms of how they approach polyphony. In the first group, systems extract F0s for the predominant source in the polyphonic mixture. The source is subsequently canceled or suppressed and the next predominant F0 is estimated. This procedure goes on iteratively until all sources are estimated. In the second group, systems attempt to estimate all F0s jointly.

2. EVALUATION

Extracting pitch information from polyphonic music is a difficult problem. This is why we choose to subdivide the task into the two MFE and NT subtasks. MFE defines a lower level representation for multiple-F0 systems. In this subtask, the systems estimate the F0s of active sources for each analysis frame.In many multi-F0 systems, frame-level F0 estimation is a precursor to the NT subtask. In the NT subtask, the systems are required to report the F0, onset and offset times of every note in the input mixture. Originally, additional timbre-tracking subtasks were envisioned for the MIREX multi-F0 task. Timbre tracking requires that the systems return the F0 contour and the notes of each individual source (e.g., oboe, flute, etc.) separately. However these subtasks were canceled due to lack of participation.

2.1 Creating the Dataset and the Ground-truth

The MIREX multi-F0 dataset consists both of recordings of a real-world performance and pieces generated from MIDI. The real-world performance is a recording of *L. van Beethoven Variations from String Quartet Op.18 N.5.* which is adapted and arranged for a woodwind quintet which consists of bassoon, clarinet, flute, horn and oboe. The piece was chosen due to its highly contrapuntal nature where the lines of each instrument are fairly different but sound harmonious when played together. Also, the predominant melodies alternate between instruments. The recording was done at the School of Music at the University of Illinois at Urbana-Champaign. First, the members of the quintet were recorded playing together where each performer was close mic'ed. Second, each part was then recorded in complete isolation while the performer listened to and played along with the other parts previously recorded through headphones. The rerecording was done in isolation because there was significant bleed through of other sources into each instruments microphone during the ensemble recording. The MIREX 2007 dataset consisted of five different 30-second sections that were chosen from the nine minute recording. The MIREX 2008 data set added two more 30-second sections for a total of seven. The sections were chosen based on high activity of all sources. The isolated instruments from those sections were mixed to form mixtures starting from duet (two polyphony) to quintet (five polyphony). This results in four clips per section where each clip is generated by introducing an extra instrument to the mixture. There was no normalization during mixing, so each source's loudness in the mixture depends on how it was performed by the musician.

To create the ground-truth set, monophonic pitch detectors were used on the isolated instrument tracks using a 46 ms window and a 10 ms hop size. The pitch detectors used were *Wavesurfer*, *Praat* and *YIN*. The pitch contours generated were manually inspected and corrected by experts to get rid of common monophonic pitch detector errors such as voiced / unvoiced detection and octave errors. To create the ground-truth for the NT subtask, the isolated instrument recordings were annotated by hand to determine each note's onset, offset and its F0 by inspecting the extracted monophonic pitch contour, the time domain amplitude envelope and the spectrogram of the recording.

The second, MIDI-based, portion of the dataset comes from two different sources. The first set was generated by [18] by creating monophonic tracks rendered and synthesized from MIDI files using real instrument samples from the RWC database [8]. The monophonic tracks were created such that no notes overlap so that each frame in the track is strictly monophonic. The ground-truth for MFE was extracted using *YIN*. The ground-truth for the NT subtask was generated using the MIDI file. Two 30-seconds sections with 4 clips from two to five polyphony were used from this data. The second set, which was used only for the note tracking subtask, was generated by [12] by recording a MIDI-controlled *Disklavier* playback piano. Two one-minute clips were used from this dataset for the note tracking subtask. The ground-truth was generated using the MIDI files.

2.2 Evaluation Methods and Metrics

This section describes the evaluation methods used in MIREX 2007 and 2008. The MFE and NT subtasks have different methods for evaluation.

2.2.1 Multi-F0 Estimation Evaluation

As mentioned earlier, the multi-F0 task represents a frame-level estimation of F0s where submitted systems were required to report active F0s every 10 ms. Many different metrics are used to evaluate this subtask. We begin by defining precision, recall and F-Measure as:

$$Precision = \frac{\sum_{t=1}^{T} TP(t)}{\sum_{t=1}^{T} TP(t) + FP(t)} \quad (1)$$

$$Recall = \frac{\sum_{t=1}^{T} TP(t)}{\sum_{t=1}^{T} TP(t) + FN(t)} \quad (2)$$

$$\textit{F-measure} = \frac{2 \times precision \times recall}{precision + recall} \quad (3)$$

Systems	Code	Front End	F0-Est Method	Note Tracking method	Ref
Cont	AC	STFT	NMF with sparsity constraints	NMF with sparsity constraints	[2]
Cao, Li	CL	STFT	Subharmonic sum, cancel-iterate	N/A	[1]
Yeh et al.	YRC	Sinusoidal an.	Joint Estimation based on spectral features	HMM tracking	[18]
Poliner, Ellis	PE	STFT	SVM classification	HMM tracking	[13]
Leveau	PL	Matching pursuit	Matching Pursuit with harmonic atoms	N/A	[10]
Raczyński et al.	SR	Constant-Q trans.	Harmonicity constrained NMF	N/A	[14]
Durrieu et al.	DRD	STFT	GMM source model, cancel-iterate	N/A	[4]
Emiya et al.	EBD	STFT	Derived from note tracking	HMM Tracking	[6]
Egashira et al.	EOS	Wavelets	Derived from note tracking	EM fit of Harmonic Temp. Models	[5]
Groble	STFT	MG	Scoring on pre-trained pitch models.	N/A	[9]
Pertusa, Iñesta	PI	STFT	Joint Estimation based on spectral features	Merge notes	[11]
Reis et al.	RFF	STFT	Derived from note tracking	Genetic Alg.	[15]
Ryynänen, Klapuri	RK	Auditory model	Derived from note tracking	HMM note and key models	[16]
Vincent et al.	EBD	ERB filter-bank	Derived from note tracking	Harmonicity constrained NMF	[17]
Zhou, Reiss	ZR	RTFI	N/A	Harmonic grouping, onset detection	[19]

Table 1. Summary of submitted multi-F0 and note tracking systems.

Since not all sources are active during any given analysis frame, the number of F0s in each time step of the ground-truth varies with time. For that reason, TP, FP and FN are defined as a function of time (frame index, t) as follows: "true positives" $TP(t)$ are calculated for frame t, based on the number F0s that correctly correspond between the ground-truth F0 set and the reported F0 set for that frame. "False positives" $FP(t)$ are calculated as the number of F0s detected that do not exist in the ground-truth set for that frame. The notion of "false negatives" $FN(t)$ however, becomes more problematic. We first begin by defining the notion of a negative. We define negatives based on the maximum polyphony of a each musical clip. Therefore, a quartet clip has a polyphony of four. Negatives in the ground-truth for each frame are calculated as the difference of the total polyphony and the number of F0s in the ground-truth. Similarly, the number of negatives for each frame in the reported F0 transcriptions are the difference between the total polyphony and the number of reported F0s. Therefore, the false negatives for each frame, $FN(t)$, is calculated as the difference between the number of reported negatives at frame t and the number of negatives in the ground-truth at frame t. Therefore, false negatives represent the number of active sources in the ground-truth that are not reported. The $TP(t)$,$FP(t)$ and $FN(t)$ are summed across all frames to calculate the total number of TPs, FPs and FNs for a given musical clip. From these measures, we can calculate an overall accuracy score as:

$$Accuracy = \frac{\sum_{t=1}^{T} TP(t)}{\sum_{t=1}^{T} TP(t) + FP(t) + FN(t)} \quad (4)$$

This is a measure of overall performance bounded between 0 and 1 where 1 corresponds to perfect transcription. However, it does not explain the types of errors that can happen. Therefore, we turn our attention to measures which better identify the types of errors multi-F0 systems make. We first note that not every instrument is active at every time frame. For example, an instrument in the mixture might be inactive through most of a piece's duration and active for only a relatively short amount of time. There are different kind of errors that can happen in estimating and reporting F0 candidates. An F0 of a source can be missed altogether, substituted with a different F0, or an extra F0 can be inserted ("false alarm" or false positive). To explain these types of errors, a measure called the frame-level transcription error score defined by [7] and used for music transcription by [12] is used. The benefit of this error measure is that this single error score can be decomposed into the three aforementioned types of errors, namely a miss, substitution, or false alarm. The total error score is defined as

$$E_{tot} = \frac{\sum_{t=1}^{T} \max(N_{ref}(t), N_{sys}(t)) - N_{corr}(t)}{\sum_{t=1}^{T} N_{ref}(t)} \quad (5)$$

where $N_{ref}(t)$ is the number of F0s in the ground-truth list for frame t, $N_{sys}(t)$ is the number of reported F0s and $N_{corr}(t)$ is the number of correct F0s for that frame. This error counts the number of returned F0s that are not correct (they are either extra or substituted F0s) and the number of F0s that are missed. The total error is calculated by summing the frame level errors and normalizing by the the total number of F0s in the ground-truth. The maximum bound of this error score is directly correlated with the number of F0s returned. Not returning anything will result in a score of 1 while perfect transcription will yield a score of 0. However, the total error is not necessarily bounded by 1. This total error can be decomposed into the sum of three sub-errors. The substitution error is defined as

$$E_{sub} = \frac{\sum_{t=1}^{T} \min(N_{ref}(t), N_{sys}(t)) - N_{corr}(t)}{\sum_{t=1}^{T} N_{ref}(t)} \quad (6)$$

The substitution error counts the number of ground-truth F0s for each frame that were not returned, but some other incorrect F0s were returned instead. These types of errors can be considered substitutions. This score is bounded between 0 and 1.

Missed errors are defined as

$$E_{miss} = \frac{\sum_{t=1}^{T} \max(0, N_{ref}(t) - N_{sys}(t))}{\sum_{t=1}^{T} N_{ref}(t)} \quad (7)$$

which counts the number of F0s in the ground-truth that were missed by the system with no substitute F0s being returned. This error is also bounded between 0 and 1.

False alarms are defined as

$$E_{fa} = \frac{\sum_{t=1}^{T} \max(0, N_{sys}(t) - N_{ref}(t))}{\sum_{t=1}^{T} N_{ref}(t)} \quad (8)$$

which counts the number of extra F0s returned that are not substitutes. Every extra F0 after the number of F0s in the ground-truth list is counted as false alarm. The upper bound of this error depends on the number of F0s returned. All errors are normalized by the total number of F0s in the ground-truth. The error is good measure for this task because it enables us to explain different types of errors and can also provide a single measure for comparison.

2.2.2 Note Tracking Evaluation

In the note tracking subtask, systems are required to return a list of notes where each note is designated by its F0, onset and offset time. The evaluation of this subtask is more straightforward then the frame-level subtask. We can think of the ground-truth list as a fixed collection of events where each event is defined by three variables, F0, onset and offset. Due to the difficulty of detecting offsets in a highly polyphonic mixture, the evaluations were calculated using two different scenarios. In the first scenario, a returned note event is assumed to be correct if its onset is within a +/-50 millisecond range of a ground-truth onset and its F0 is within +/- a quarter tone (3%) of the ground-truth pitch. Here, the offset times are ignored. In the second scenario, in addition to the previous onset and pitch requirements, the correct returned note is required to have an offset time within 20% of ground-truth note's duration around the ground-truth note's offset value, or within 50 milliseconds of the ground-truth note's offset, whichever is larger. For these two cases, precision, recall and F-measure are calculated where true positives are defined as the returned notes that conform to the previously mentioned requirements and false positives were defined as the ones that do not. We also define an additional measure called Overlap Ratio (OR). The OR for a ith correct note in the returned list is defined as

$$\text{OR}_i = \frac{min(t_{i,off}^{ref}, t_{i,off}^{sys}) - max(t_{i,on}^{ref}, t_{i,on}^{sys})}{max(t_{i,off}^{ref}, t_{i,off}^{sys}) - min(t_{i,on}^{ref}, t_{i,on}^{sys})} \quad (9)$$

where $t_{i,off}^{sys}$ and $t_{i,on}^{sys}$ are the offset and the onset times of the correctly returned note and $t_{i,off}^{ref}$ and $t_{i,on}^{ref}$ are the offset and onset times of the corresponding ground-truth note. An average OR score is a good measure of how much the correct returned note overlaps with the corresponding ground-truth note. This information is especially useful when the correct notes are calculated based on the onset only.

3. RESULTS AND DISCUSSION

The evaluation results of two iterations of the MIREX multi-F0 estimation task (2007-2008) are presented here. We first turn our attention to the frame-level MFE subtask. Figure 1 shows the precision, recall, and accuracy scores for all submitted MFE systems over the two years. In general, systems have improved in accuracy over the course of the two years.

In Figure 2, a bar graph of the total error is shown for each of the systems. Each total error bar is subdivided into the three types of errors that constitute it namely, miss errors, substitution errors, and false alarm errors. It is evident that different systems present different trade-offs in terms of the types of errors. Referring back to Fig. 1, one can see that some systems have a very high precision compared to their accuracy such as those by PI, EBD and PE [6, 11, 13]. PI has the highest precision in both years. The reason behind this is that most of the F0s reported by these systems are correct, but they tend to under-report and miss a lot of active F0s in the ground-truth. This type of behavior is also evident in Fig. 2. While PI systems have the lowest total error score, there are very few false alarms compared to miss errors. PI achieves a low number of local false positives by taking into account a temporal salience of each combination of pitches. The results are post-processed by either merging/ignoring note events or using a weighted directed acyclic graph (wDAG).

Similarly, EBD and PE use hidden Markov models for temporal smoothing, and also have a relatively high miss error. RK [16] and YRC [18] have balanced precision, recall, as well as a balance in the three error types, and as a result, have the highest accuracies for MIREX 07 and MIREX 08, respectively. On the other hand, some systems like half of the CL submissions, have a high recall compared to their precision accuracy. CL returned a fixed (maximum) number of F0s for every frame regardless of the input polyphony in order to maximize recall.

The top two submissions share similar approaches. Both YRC and PI(1,2) generate a pool of candidate F0s for each frame and combine the candidates into hypotheses to jointly evaluate the present F0s. YRC first estimates an adaptive noise level, and extracts sinusoidal components. The algorithm then extracts F0 candidates until all the sinusoidal components are explained in the signal, as well as a polyphony inference stage that estimates the number of concurrent sources. All combinations of F0 candidates are evaluated by a score function based on smoothness and harmonicity, among others, and the best set is chosen. Finally, a tracking method is performed by first connecting F0 candidates across frames to establish candidate trajectories and then pruning them using HMMs. PI takes a similar approach in that, once again, joint F0 hypotheses are evaluated using saliency scores based on properties such as spectral smoothness and candidate loudness. Post-processing either takes into account local signal characteristics taken from adjacent frames or uses wDAGs for F0 note merging or pruning. The top performing algorithm from 2007, RK uses an auditory inspired model for anal-

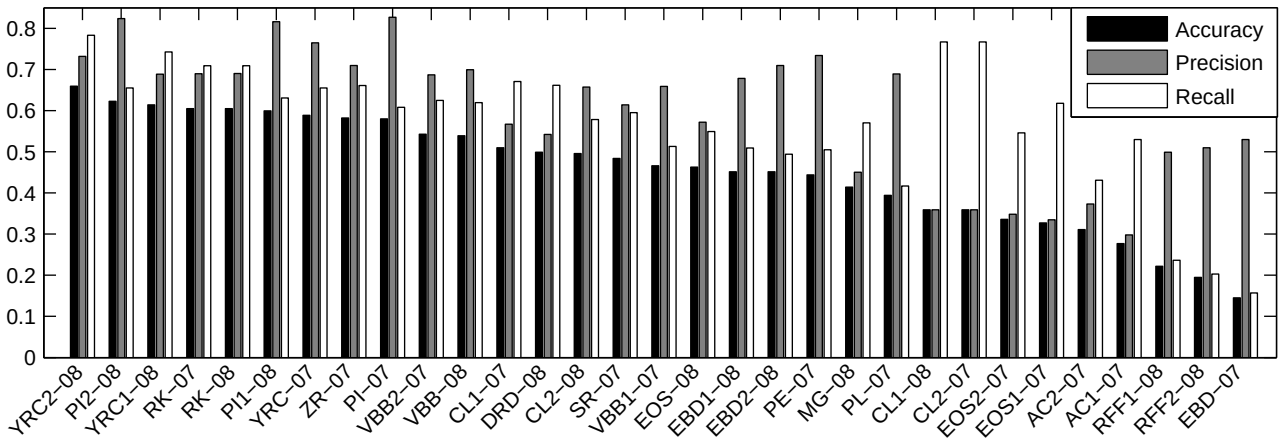

Figure 1. Precision, recall and accuracy for MIREX 07 and MIREX 08 MFE subtask ordered by accuracy.

ysis, and uses HMMs for note models and for note transitions, after a musical key estimation stage, in an attempt to incorporate some musicological information into the process.

For the NT subtask, Fig. 3 shows the precision, recall, and F-measures of the onset-offset based evaluation of the note tracking systems. We notice that in the NT onset-offset evaluation, performance is relatively poor. The likely explanation of this performance stems from the difficulty in properly defining an offset ground-truth in the data sets. In the woodwind data set, offset ground-truth was defined on the monophonic recordings of each track where the offset was labeled at very low loudness. Once mixed, other signals can dominate the low level of a source at the tail end of its decay such that the offset within the mixture is somewhat ambiguous. For the MIDI-generated piano dataset, offset is defined based on the MIDI file, and does not take into account the natural decay and the reverberation of the piano. Therefore, in the woodwind dataset, the offset time may be overestimated, whereas in the MIDI-generated dataset, the offset may be underestimated. Due to the inherent difficulty of properly defining offset, we also evaluate based strictly on note onset. The onset-based evaluation results of the NT subtask can be seen in Fig. 4. More detailed results and significance tests can be found at the MIREX wiki pages. [1]

4. CONCLUSION

Inspecting the methods used and their performances, we cannot make generalized claims as to what type of approach works best. In fact, statistical significance testing showed that the top three methods were not significantly different. However, systems that go beyond simple frame-level estimation methods and incorporate temporal constraints or other note tracking methods seem to perform better. It is plausible that timbral/instrument tracking can improve MFE even more. A future direction for evaluation would then be to add an instrument tracking subtask that

[1] http://www.music-ir.org/mirex/2007/index.php/MIREX2007_Results
http://www.music-ir.org/mirex/2008/index.php/MIREX2008_Results

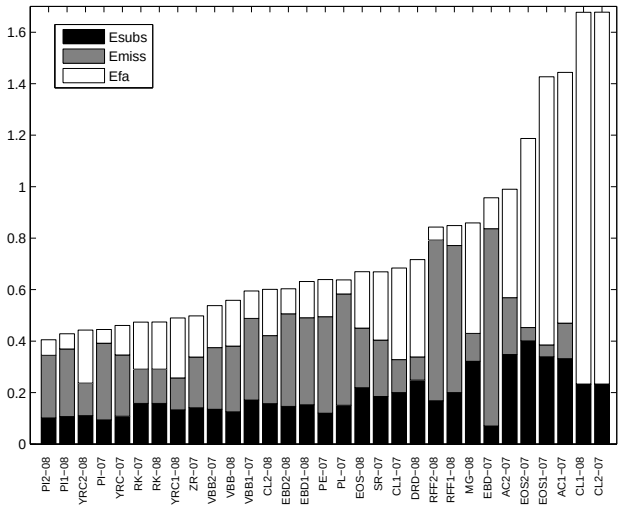

Figure 2. Error scores for MIREX 07 and MIREX 08 MFE subtask ordered by total error.

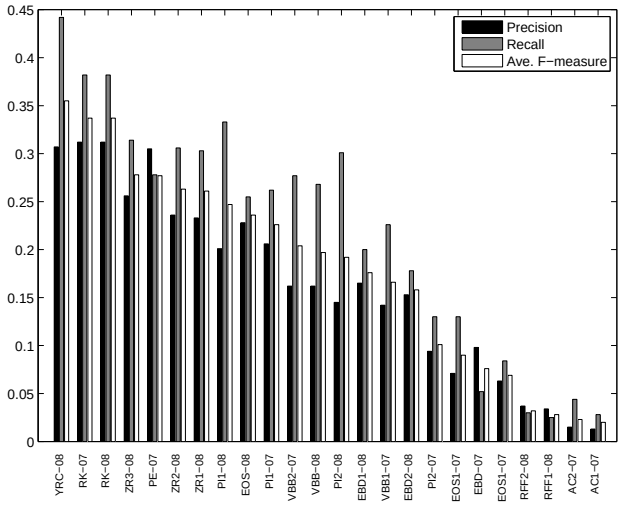

Figure 3. Precision, recall and F-measure based on note onset and offset for the MIREX 07 and MIREX 08 NT subtask.

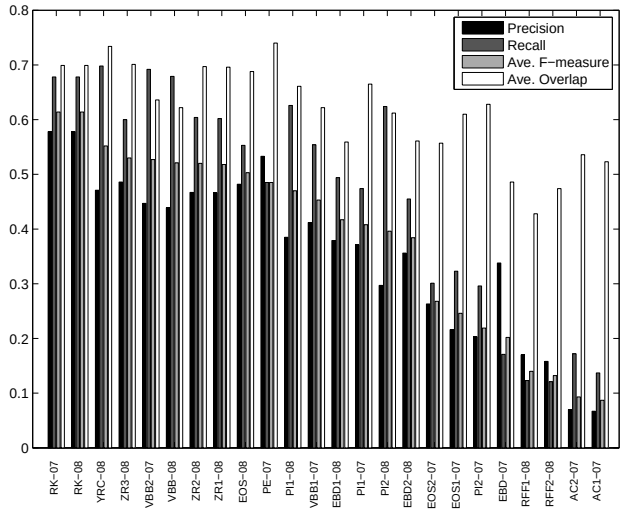

Figure 4. Precision, recall and F-measure based on note onset only for the MIREX 07 and MIREX 08 NT subtask.

would lead to a more complete music transcription task. The music transcription field is advancing but the problem is still far from being solved and there is a great room for improvement.

5. REFERENCES

[1] C. Cao and M. Li. Multiple F0 Estimation in Polyphonic Music, Avaliable at http://www.music-ir.org/mirex/2008/abs/mirex08_multiF0_Cao.pdf.

[2] A. Cont, S. Dubnov, and D. Wessel. Realtime multiple-pitch and multiple-instrument recognition for music signals using sparse non-negative constraints. In *Proceedings of the International Conference on Digital Audio Effects (DAFx). Bordeaux, France*, 2007.

[3] J.S. Downie. The music information retrieval evaluation exchange (2005–2007): A window into music information retrieval research. *Acoustical Science and Technology*, 29(4):247–255, 2008.

[4] J.L. Durrieu, G. Richard, and B. David. Singer melody extraction in polyphonic signals using source separation methods. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP 2008*, pages 169–172, 2008.

[5] K. Egashira, N. Ono, and S. Sagayama. Sequential Estimation of Multiple Fundamental Frequencies Through Harmonic-Temporal-Structured Clustering, Avaliable at http://www.music-ir.org/mirex/2008/abs/F0_egashira.pdf.

[6] V. Emiya, R. Badeau, and B. David. Multipitch estimation of inharmonic sounds in colored noise. In *Proc. Int. Conf. Digital Audio Effects (DAFx), Bordeaux, France*, pages 93–98, 2007.

We thank Andrew W. Mellon Foundation for their financial support.

[7] J.G. Fiscus, N. Radde, J.S. Garofolo, A. Le, J. Ajot, and C. Laprun. The rich transcription 2005 spring meeting recognition evaluation. *Lecture Notes in Computer Science*, 3869:369, 2006.

[8] M. Goto. Development of the RWC music database. In *Proceedings of the 18th International Congress on Acoustics (ICA 2004)*, volume 1, pages 553–556, 2004.

[9] M. Groble. Multiple fundamental frequency estimation, Avaliable at http://www.music-ir.org/mirex/2008/abs/F0_groble.pdf.

[10] P. Leveau, D. Sodoyer, and L. Daudet. Automatic Instrument Recognition in a Polyphonic Mixture using Sparse Representations. In *Proc. of Int. Conf. on Music Information Retrieval (ISMIR), Vienne, Autriche*, 2007.

[11] A. Pertusa and J.M. Inesta. Multiple fundamental frequency estimation using Gaussian smoothness. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP 2008*, pages 105–108, 2008.

[12] G.E. Poliner and D.P.W. Ellis. A discriminative model for polyphonic piano transcription. *EURASIP Journal on Advances in Signal Processing*, 2007:1–9, 2007.

[13] G.E. Poliner, D.P.W. Ellis, A.F. Ehmann, E. Gomez, S. Streich, and B. Ong. Melody Transcription From Music-Audio: Approaches and Evaluation. *IEEE Transactions on Audio Speech and Language Processing*, 15(4):1247, 2007.

[14] S.A. Raczynski, N. Ono, and S. Sagayama. Multipitch analysis with harmonic nonnegative matrix approximation. In *Proc. Int. Conf. on Music Information Retrieval (ISMIR)*, pages 381–386, 2007.

[15] G. Reis, N. Fonseca, F.F. de Vega, and A. Ferreira. Hybrid Genetic Algorithm Based on Gene Fragment Competition for Polyphonic Music Transcription. *Lecture Notes in Computer Science*, 4974:305, 2008.

[16] M. P. Ryynanen and A. Klapuri. Polyphonic music transcription using note event modeling. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2005*, pages 319–322, 2005.

[17] E. Vincent, N. Bertin, and R. Badeau. Harmonic and inharmonic nonnegative matrix factorization for polyphonic pitch transcription. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP 2008*, pages 109–112, 2008.

[18] C. Yeh. *Multiple fundamental frequency estimation of polyphonic recordings*. PhD thesis, Ph. D. dissertation, Universit Pierre et Marie Curie, Paris, Jun, 2008.

[19] R. Zhou and J.D. Reiss. A Real-Time Frame-Based Multiple Pitch Estimaiton Method Using The Resonator Time-Frequency Image, Avaliable at http://www.music-ir.org/mirex/2008/abs/F0_zhou.pdf.