MELODY EXTRACTION BASED ON A SOURCE-FILTER MODEL USING PITCH CONTOUR SELECTION

Juan J. Bosch
Music Technology Group,
Universitat Pompeu Fabra, Spain
juan.bosch@upf.edu

Emilia Gómez
Music Technology Group,
Universitat Pompeu Fabra, Spain
emilia.gomez@upf.edu

ABSTRACT

This work proposes a melody extraction method which combines a pitch salience function based on source-filter modelling with melody tracking based on pitch contour selection. We model the spectrogram of a musical audio signal as the sum of the leading voice and accompaniment. The leading voice is modelled with a Smoothed Instantaneous Mixture Model (SIMM), and the accompaniment is modelled with a Non-negative Matrix Factorization (NMF). The main benefit of this representation is that it incorporates timbre information, and that the leading voice is enhanced, even without an explicit separation from the rest of the signal. Two different salience functions based on SIMM are proposed, in order to adapt the output of such model to the pitch contour based tracking. Candidate melody pitch contours are then created by grouping pitch sequences, using auditory streaming cues. Finally, melody pitch contours are selected using a set of heuristic rules based on contour characteristics and smoothness constraints. The evaluation on a large set of challenging polyphonic music material, shows that the proposed salience functions help increasing the salience of melody pitches in comparison to similar methods. The complete melody extraction methods also achieve a higher overall accuracy than state-of-the-art approaches when evaluated on both vocal and instrumental music.

1. INTRODUCTION

The task of melody extraction from polyphonic music recordings has been generally approached with salience-based or separation-based methods [1]. Salience-based approaches compute a frame-based pitch salience function, while separation-based approaches attempt to isolate the melody source from the mixture more or less explicitly. Melody oriented pitch salience functions should ideally only contain a peak at the frequency corresponding to the melody pitch present at a given instant.

The most commonly used pitch salience function is harmonic summation [2]. This approach is computationally inexpensive and has been used successfully in a variety of forms for predominant melody extraction [3, 4] or multiple pitch estimation [5]. More recently, probabilistic approaches based on decomposition models such as Non-negative Matrix Factorisation (NMF) have gained more interest [6, 7], especially within source separation scenarios.

Salamon and Gómez [3] propose a salience function based on harmonic summation [2], computed as the sum of the weighted energies found at integer multiples (harmonics) of each of the considered frequencies. Durrieu et al. [6] propose a salience function within a separation-based approach using a Smoothed Instantaneous Mixture Model (SIMM), as detailed in Section 2. There are important differences between the salience functions obtained with SIMM (H_{f_0}) and harmonic summation (HS). H_{f_0} is much more sparse, and has a larger range of values, since the method does not prevent values to be very high or very low. Figure 3 shows a comparison of both salience functions for one of the excerpts used for evaluation: (a) shows the pitch salience function obtained with SIMM, and (b) corresponds to HS, which is more dense and smooth, and has a smaller range of values.

Melody extraction methods exploit salience functions for pitch tracking, relying on the energetic predominance of melody pitches and on melody contour smoothness, using e.g. streaming rules [4] Hidden Markov Models (HMM) [7, 8], or pitch contour characteristics [3]. Finally, frames are classified as voiced or unvoiced (containing a melody pitch or not respectively), using static or dynamic thresholds [4, 7], or exploiting pitch contour salience distribution [3]. For instance, Durrieu et al. [8] use an empirically chosen fixed threshold, such that voiced frames represent more than 99.95% of the leading instrument energy. Salamon [9] proposed a generative model to distinguish melody from non-melody contours, and Bittner [10] proposed a discriminative classifier based on contour features. While both approaches learnt from training data, none of them increased the overall accuracy obtained with the method based on heuristic rules [3].

Main challenges in melody extraction deal with more complex music material [1], with melodies played by different instruments, harmonised melodic lines, or music that features "ensemble" sounds, typically found when several performers play or sing in unison. Some characteristics of such sounds is the fluctuation of the pitches, known as voice flutter, typically found in orchestral and choral music [11]. A key step towards the development of more advanced algorithms and a more realistic evaluation are large and open annotated databases. Recent works presented datasets for melody extraction with such char-

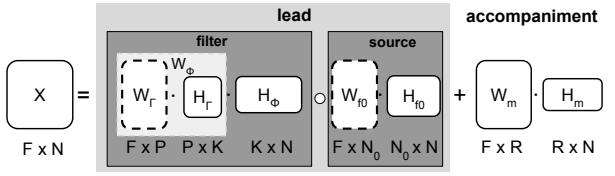

Figure 1: SIMM model. Dashed lines refer to the matrices which are fixed, while the rest are iteratively estimated

acteristics, e.g. in a variety of genres and instrumentation (MedleyDB) [12], and in orchestral music (Orchset) [13]. These datasets allow broader definitions of melody than the one used in Music Information Retrieval Evaluation eXchange (MIREX) [14], since they are not restricted to a single instrument. Results on both datasets generally drop significantly in comparison to results on simpler datasets used in MIREX [12, 13, 15].

Previous works [13, 15] have shown the benefits of using separation-based approaches such as the [8] for pitch estimation on orchestral data. However, voicing detection was identified as a key aspect to improve in their method [8]. In this work, we address some of the mentioned challenges, by combining separation and salience-based methods [16, 17], as presented in Section 2. In Section 3 we present the methodology to evaluate pitch salience functions and melody extraction methods, and we present and discuss the results in Section 4.

2. METHOD

We propose a melody extraction method, based on the combination of a salience function based on a source-filter model [6] with melody tracking based on pitch contour selection (PCS) [3]. Our intention is to obtain both an accurate pitch estimation and a good voicing detection, based on their results in [15], and in MIREX.

We propose two different salience functions which aim at adapting the characteristics of H_{f_0} to a melody tracking stage based on pitch contour selection. The first salience function (CB) combines two salience functions: one based on SIMM (H_{f_0}) [6, 8] and another one based on harmonic summation (HS) [3]. The second approach (EW) uses an estimate of the energy of the melody. Both approaches employ Gaussian filtering, since we hypothesise that such smoothing is useful to make melody pitches more salient, particularly in the case of "ensemble" sounds.

We reuse code from Durrieu's source-filter model implementation [1] and Essentia [2] [18], an open source library for audio analysis with a slightly different implementation of [3] compared to MELODIA [3] . Our source code is available for research reproducibility [4] .

2.1 Pitch salience function based on SIMM

Following [6], we model the spectrum of the signal as the lead instrument plus accompaniment: $\hat{X} = \hat{X}_v + \hat{X}_m$,

[1] https://github.com/wslihgt/separateLeadStereo
[2] http://essentia.upf.edu
[3] http://mtg.upf.edu/technologies/melodia
[4] https://github.com/juanjobosch/SourceFilterContoursMelody

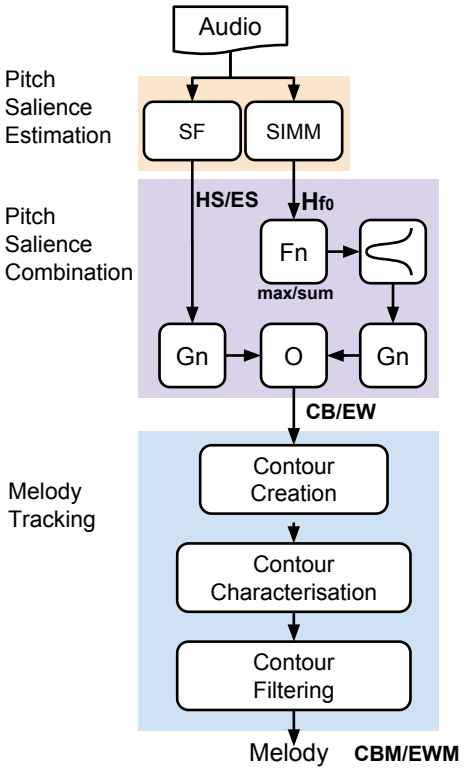

Figure 2: **Left**: Proposed method schema. SIMM: Smoothed Instantaneous Mixture Model (outputs H_{f_0}); SF: salience function, either Harmonic Summation (outputs HS) or Energy-based Salience (outputs ES); Fn: Frame-wise normalisation; Gn: Global normalisation; o: Hadamard product; Gaussian symbol: Gaussian filtering. Combining H_{f_0} with HS we obtain CB. Combining it with ES we obtain EW. CBM and EWM denote the complete melody extraction methods.

where $\hat{X}$ represents the modelled spectrum. The lead instrument is modelled as: $\hat{X}_v = X_\Phi \circ X_{f_0}$, where X_{f_0} corresponds to the source, X_Φ to the filter, and the symbol $\circ$ denotes the Hadamard product. Both source and filter are decomposed into basis and gains matrices as $X_{f_0} = W_{f_0} H_{f_0}$ and $X_\Phi = W_\Phi H_\Phi$ respectively. The filter basis matrix W_Φ is further decomposed into a weighted sum of smooth spectral atoms: $W_\Gamma H_\Gamma$. H_{f_0} corresponds to the pitch activations of the source, which can also be understood as a representation of pitch salience [6]. The accompaniment spectrum is modelled as: $\hat{X}_m = \hat{W}_m \hat{H}_m$, leading to Equation 1.

$$X \approx \hat{X} = (W_\Gamma H_\Gamma H_\Phi) \circ (W_{f_0} H_{f_0}) + W_m H_m \quad (1)$$

Several parameters of the algorithm need to be specified: the number of bins per semitone (U_{st}), the number of possible elements of the accompaniment (R), the number of atomic filters in W_Γ (K), and the maximum number of iterations (N_{iter}). Parameter estimation is based on Maximum-Likelihood, with a multiplicative gradient method [8], updating parameters in the following order for each iteration: H_{f_0}, H_Φ, H_m, W_Φ and W_m. Figure 1 represents the blocks of the Smoothed Instantaneous Mixture Model.

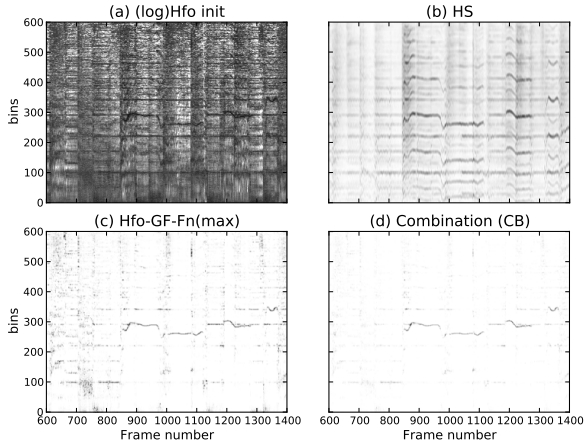

Figure 3: Time-frequency pitch salience representation of an excerpt from "MusicDelta_Beatles.wav" (MedleyDB) with (a) SIMM: $log_{10}(H_{f_0})$ is represented, to reduce the range of values for visualisation purposes) (b) Harmonic Summation: HS (c) H_{f_0} (max) normalised per frame and gaussian filtered (d) Combination (CB).

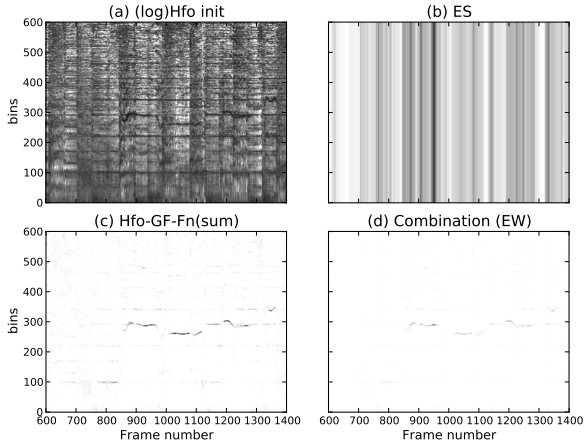

Figure 4: Time-frequency pitch salience representation of an excerpt from "MusicDelta_Beatles.wav" (MedleyDB) with (a) SIMM: $log_{10}(H_{f_0})$ is represented, to reduce the range of values for visualisation purposes) (b) Energy-based matrix: ES (c) H_{f_0} normalised per frame and gaussian filtered (d) Combination (EW).

2.2 CB: Combination with Harmonic Summation

In order to adapt H_{f_0} for pitch contour based tracking, we first propose to combine it with a Harmonic Summation salience function (HS), since pitch contour tracking, was originally adapted to this kind of representation [3]. The computation of HS starts with a Short Time Fourier Transform (STFT) as time-frequency transformation, applies Equal-Loudness Filters (ELF), finds spectral peaks positions and magnitudes, and then refines them using parabolic curve fitting (as implemented in Essentia).

We normalize and combine the considered pitch salience functions $HS(k,i)$ and $H_{f_0}(k,i)$, where *k* indicates the frequency index (bin) and *i* the frame index. The process is illustrated in Figure 2: 1) **Global normalization** (Gn) of HS, dividing all elements by their maximum value $max_{k,i}(HS(k,i))$. 2) **Frame-wise normalization** (Fn) of H_{f_0}. For each frame *i*, divide $H_{f_0}(k,i)$ by $max_k(H_{f_0}(k,i))$. 3) **Convolution in the frequency axis** *k* of H_{f_0} with a Gaussian filter to smooth estimated activations. The filter has a standard deviation of 0.2 semitones. 4) **Global normalization** (Gn), whose output is $\widetilde{H_{f_0}}$ (see Figure 2 (c)). 5) **Combination** by means of element-wise product: $S_c = \widetilde{H_{f_0}} \circ HS$ (see Figure 3 (d)).

2.3 EW: Energy-based normalisation

In order to reduce the range of salience values of H_{f_0}, one possibility would be to simply normalise each frame with the maximum salience. The drawback of this solution is that high salience values also appear in unvoiced frames, which would make voicing detection based on pitch contour selection a complicated task. In order to reduce the salience of unvoiced parts, we employ a frame-wise energy estimate of the melody line, using the method in [8]. For energy estimation, a HMM is employed, where each state corresponds to one bin of the pitch salience function (H_{f_0}), and the probability of each state corresponds to the estimated salience. Pitch continuity is considered in the transition probabilities, favouring smoothness in pitch trajectories. The energy of the melody source for each frame i (E_i), is then computed using the decoded pitch sequence and the matrix decomposition computed before.

The estimated energy is then used to create a matrix (ES) with the same size as H_{f_0}, in which all bins in one frame are equal to the estimated energy in that frame: $ES(k,i) = E_i, \forall k$. ES is then combined with H_{f_0} to create the salience function EW, following the same steps introduced in Section 2.2 (see Figure 2), with the difference that in the frame-wise normalisation (Fn), $H_{f_0}(k,i)$ is divided by $\sum_k H_{f_0}(k,i)$, instead of the maximum value, also following Durrieu's approach. Figure 4 illustrates the combination.

2.4 Melody tracking

From the proposed lead-enhanced salience functions, we create pitch contours by grouping continuous sequences of salience peaks, following [3]. Several parameters need to be set (default values used here are presented between brackets). Salience peaks are first filtered per frame: peaks below a threshold factor τ_+ (0.9) of the highest salience peak are filtered out. Secondly, peaks are filtered if their salience is below $\mu_s - \tau_\sigma \cdot \sigma_s$, where μ_s and σ_s are the mean and standard deviation of the salience of remaining peaks (in all frames). τ_σ (0.9) determines the accepted degree of deviation below mean salience. Contours are created by grouping peaks which are close in time and frequency, with several parameters: the minimum allowed contour duration (100 ms); maximum allowed pitch change during 1 ms time period (27.56 cents) and maximum allowed gap duration (tc = 50 ms). Default parameters here are the same as in [3], except for tc. We analyse the effect of some of these parameters in Section 4.3.

Created contours are then characterised by a set of features: pitch (mean and deviation), salience (mean, standard deviation), total salience, length and presence of vi-

Method	Salience	Description
DUR	H_{f_0}	Source-filter model (SIMM) [8]
SAL	HS	VAMP Implementation of [3]
ESS	HS	Essentia implementation of [3]
CBM	CB	HS+H_{f_0} with PCS
EWM	EW	Energy weighted H_{f_0} with PCS

Table 1: Overview of the evaluated melody extraction methods. PCS: Pitch Contour Selection

brato. Contour features are then exploited for voicing detection, octave error minimisation, and final melody selection. Non-melody contours are filtered out using a voicing detection threshold τ_ν, based on contour salience distribution: $\tau_\nu = \overline{C_s} - \nu \cdot \sigma_{C_s}$ where $\overline{C_s}$ and σ_{C_s} are the contours' salience mean and standard deviation. We focus on the effect of parameter ν (0.2), which controls the amount of filtered contours. For a more detailed explanation, the reader is referred to [3].

Complete melody extraction methods using the proposed salience functions are here denoted as CBM (using CB) and EWM (using EW) (see Figure 2).

3. EVALUATION

We conduct two different kind of evaluation experiments in order to analyse the benefits of combining the proposed salience functions with pitch contour-based tracking. First, the proposed salience functions are evaluated and compared to H_{f_0} and HS in terms of their usefulness for melody extraction. Second, the complete melody extraction approaches are compared to Durrieu et al. [8] (DUR) and two implementations of Salamon and Gómez [3]: VAMP plugin MELODIA (SAL) and Essentia (ESS). Table 1 presents an overview of the evaluated methods. The motivation to conduct the evaluation at two different levels is to better understand the benefits of the combination of salience functions, and the effect on the complete melody extraction method. The pitch resolution (number of bins per semitone) was set to $U_{st} = 10$, and the hop size was 256 samples, except for SAL which is fixed to 128. Sampling rate was 44100 Hz. The frequency limits were set to $f_{min} = 55$ Hz and $f_{max} = 1760$ Hz for all algorithms.

3.1 Datasets

The evaluation is conducted on MedleyDB and Orchset datasets, converted to mono as (left+right)/2. MedleyDB contains 108 melody annotated files (most between 3 and 5 minutes long), with a variety of instrumentation and genres. We use two definitions of melody, **MEL1**: the f_0 curve of the predominant melodic line drawn from a single source (MIREX definition), and **MEL2**: the f_0 curve of the predominant melodic line drawn from multiple sources.

Orchset [5] contains 64 excerpts from symphonies, ballet suites and other musical forms interpreted by symphonic orchestras. The definition of melody in this dataset is not restricted to a single instrument, with all (four) annotators agreeing on the melody notes [15]. The focus is pitch estimation, while voicing detection is less important: the proportion of voiced and unvoiced frames is 93.7/6.3%.

3.2 Salience function evaluation

Salience functions are evaluated from two different perspectives: pitch and salience estimation accuracy. To do so, we compute four different metrics [19], using the ground truth melody.

We start by computing salience function peaks, and then select the peak closest to the ground truth, which is considered as the melody salience peak. The first metric is the frequency error of the salience function Δf_m, computed as the difference (in cents) between the frequency of the melody salience peak and the ground truth f0. The following metrics deal with salience estimation. The first metric (RR_m) is the reciprocal rank score of the melody salience peak amongst the rest of salience peaks (the closer to one the better). The second (S1) is the relative salience of the melody peak in comparison to the highest salience peak in that frame (the closer to one the better). Last metric (S3) computes the salience of the melody peak, divided by the mean salience of the 3 highest peaks (the higher the better). We consider the latter as the single most important measure, since it quantifies the ability of a method to make the melody pitch more salient than the rest of the peaks, which is a key property of a salience function.

3.3 Melody extraction evaluation

Following MIREX methodology, we evaluate melody extraction approaches by comparing the estimated sequence of pitches against a ground truth sequence of melody pitches. All evaluated algorithms were set to report an estimated melody pitch even for frames considered unvoiced. This allows evaluating voicing and pitch estimation separately. Five standard melody extraction metrics [6] are computed using mir_eval [20]: Voicing recall rate (VR): proportion of frames labelled as melody frames in the ground truth that are estimated as melody frames; Voicing false alarm rate (VFA): proportion of frames labelled as non-melody in the ground truth that are mistakenly estimated as melody frames; Raw Pitch Accuracy (RPA): proportion of melody frames in the ground truth for which the estimation is considered correct (within half a semitone of the ground truth); Raw Chroma Accuracy (RCA): measure of pitch accuracy, in which both estimated and ground truth pitches are mapped into one octave, thus ignoring octave errors; Overall Accuracy (OA): proportion of frames that were correctly labelled in terms of both pitch and voicing.

4. RESULTS

4.1 Salience function

In order to have an idea of the variance between excerpts, we compute the mean value of the metrics for each excerpt, and we then visualise evaluation results with a boxplot, as presented in Figure 5. The lower and upper lines of each

[5] mtg.upf.edu/download/datasets/orchset

[6] http://www.music-ir.org/mirex/wiki/2014:Audio_Melody_Extraction

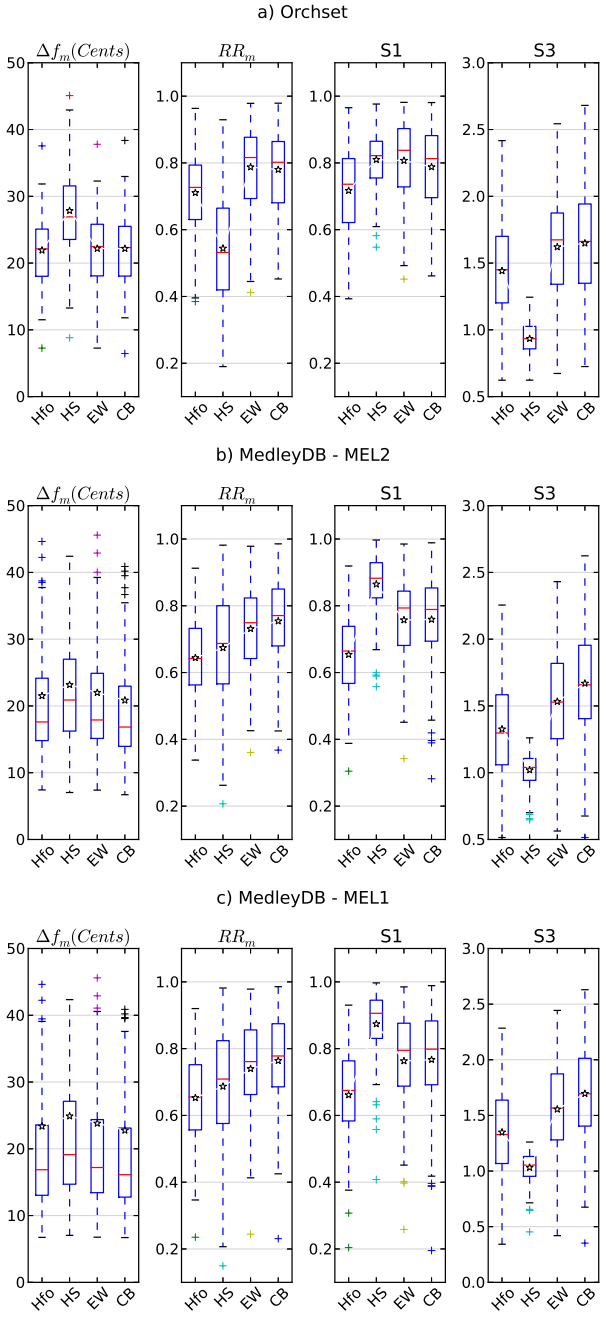

Figure 5: Salience function evaluation results a) Orchset. b) MedleyDB with MEL2 definition c) MedleyDB with MEL1 definition. Mean values represented with a star.

box show the 25th and 75th percentiles of the sample, and the line inside each box represents the median.

Before analysing results, note that the normalisations and energy weighting performed in the proposed EW salience function, do not affect any of the metrics for salience function evaluation. Any difference in results between EW and H_{f_0} is thus only due to the proposed Gaussian filtering performed in each frame of the salience function.

Regarding frequency error (Δf_m), the lowest median value is obtained with CB, but differences amongst all approaches are not significant on MedleyDB (with both melody definitions). In the case of Orchset, H_{f_0} and the proposed methods obtain lower errors than HS. Note that on Orchset, results do not really represent the difference from the closest salience peak and the real melody pitch, since melody notes are played by orchestral sections, and individual instruments contributing to the melody are playing slightly different pitches. Additionally, ground truth pitches in Orchset are actually quantized at the semitone level, since they were derived from MIDI notes, without tuning information.

With regard to salience related metrics, we observe that the reciprocal rank RR_m of the proposed salience functions EW and CB is higher than the rest. Also note that HS performs better on MedleyDB than on Orchset, while H_{f_0} behaves similarly in both datasets. The performance of CB is better than EW on MedleyDB, presumably because of the synergy obtained when combining the two salience functions. In the case of Orchset, the performance of CB in comparison to EW is decreased since HS does not perform as well in orchestral data.

HS obtains the highest mean value of S1 for MEL2 on MedleyDB, however best S3 results are obtained with CB. As previously introduced, S1 compares the salience of the melody peak and the highest salience peak in a frame. S3 measures if the melody peak stands out from the other peaks of the salience function and by how much. These results show that HS achieves a high S1 score because the highest salience peaks do not actually present a high difference between them (the value of both S1 and S3 are close to one). HS obtains a median S3 of less than 1 on Orchset, which attending to the definition of the metric, means that (in average) the salience of the melody peak is smaller than the mean of the three highest peaks. H_{f_0} on the other hand presents a higher difference between the melody peak and the following most salient peaks.

We thus conclude that the proposed combinations do not significantly reduce the estimation error of the melody pitch frequency (Δf_m) with respect to the compared approaches. However, the proposed combined salience function (CB) achieves the highest S3 value, meaning that is the most able to make the melody pitch more salient.

4.2 Melody extraction

After analysing salience functions results, we focus on complete melody extraction methods. Figure 6 shows the results for all metrics obtained with all approaches in both Orchset and MedleyDB with both melody definitions. Results are reported for experiments conducted with the same parameters values as in [3], except for the maximum allowed gap tc = 50 ms. An analysis of the effect of the parameters is presented in Section 4.3.

Comparing the results obtained with the proposed methods, we observe that CBM achieves the best overall accuracy in both datasets. This is specially noticeable in Orchset, partially due to the higher recall. Pitch related accuracies are quite similar in both approaches, especially for MedleyDB. In comparison with other methods, both of the proposed methods yield a higher OA than ESS (baseline), for both datasets and both melody definitions. The OA is also higher in comparison to the rest of the related approaches, for both MEL1 and MEL2 on MedleyDB. In

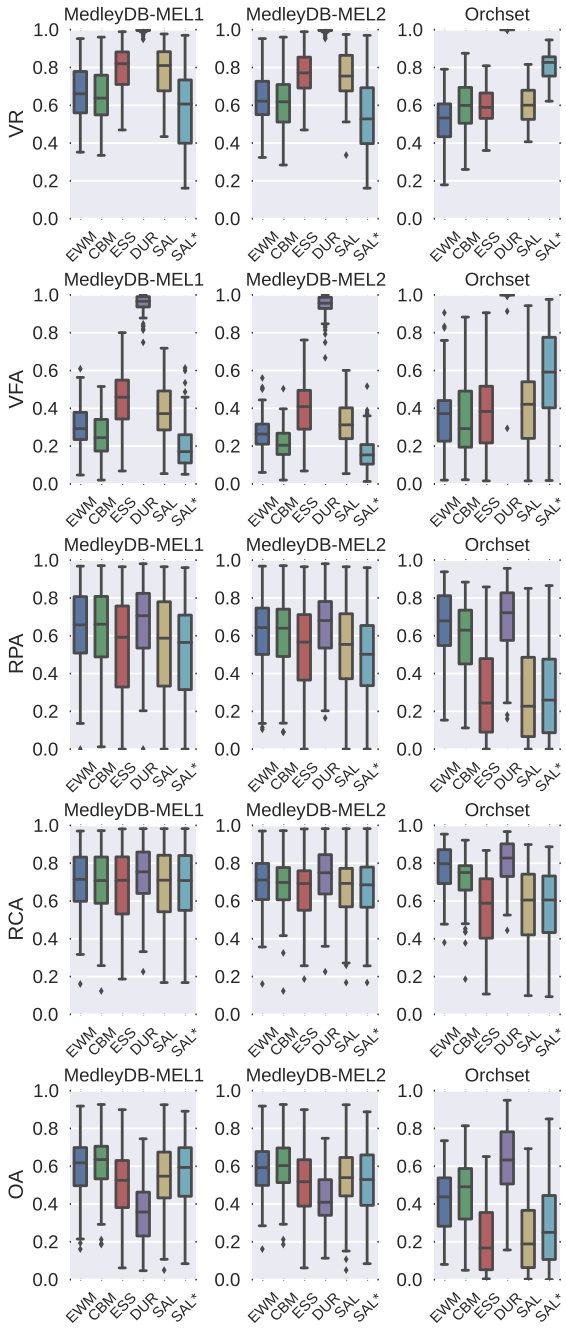

Figure 6: Evaluation results on all metrics, for MedleyDB with both MEL1 and MEL2 definitions and Orchset. "SAL*" denotes the results obtained with SAL with $\nu = -1$ for MedleyDB and $\nu = 1.4$ for Orchset

the case of Orchset, only DUR yields a better OA than the proposed methods, due to a very high recall. Note that DUR always obtains almost perfect recall on all datasets, and very high false alarm rates, since this method outputs almost all frames as voiced. The influence of this fact on the overall accuracy depends on the amount of voiced frames of the dataset. Since Orchset mostly contains voiced frames (93.7%), it is beneficial, but MedleyDB contains full songs with large unvoiced portions, and false alarms in this data considerably reduce the OA.

The proposed approaches achieve a slightly lower RPA in comparison to H_{f_0}. This is related to the fact that Durrieu's method estimates most frames as voiced. Even though it is considered a pitch related metric, RPA is actually also affected by voicing estimation, since it compares estimated pitches with voiced ground truth pitches. If some of the melody contours are not created, or erroneously filtered (e.g. due to a lower salience in comparison to the rest of the contours), this will affect both voicing related metrics and pitch related metrics. That is the case for our proposed methods: while many frames are correctly identified as unvoiced, some contours which correspond to the melody are filtered or simply not created, which decreases pitch related accuracies. However, reducing the voicing false alarm rate helps achieving a better overall accuracy.

SAL and ESS obtain lower pitch related accuracies (RPA, RCA) than the proposed methods, specially in orchestral music. Given that the only difference between them is the salience function, we can conclude that the results are improved thanks to the use of a source-filter model and gaussian filtering. This could be expected from the previously presented salience function evaluation results, since the proposed salience functions are able to make the melody pitch more salient.

Also note that the difference between RCA and RPA is much higher in SAL than in the proposed methods, specially on Orchset. This shows that the kind of signal representation underneath the proposed pitch salience functions is very effective at reducing the amount of octave errors [6, 21].

4.3 Parameter tuning

Previous results can be further improved by adapting melody extraction parameters to the proposed salience functions. We first analysed the influence of Gaussian filtering (see Figure 2) on the complete melody extraction system CBM, by suppressing it from the pitch salience creation process. The effect is quite small on MedleyDB, but it helped improving pitch estimation on Orchset (4% points). This could be due to the small differences in the pitch played by the individual instruments contributing to the melody. As previously observed with the salience function evaluation results, by smoothing H_{f_0} we are able to make more salient the pitches of the notes played by orchestral sections in unison.

Several other parameters affect different parts of the method: salience function creation, contour creation or melody contour selection. Figure 7 shows the effect of the number of iterations (N_{iter}), maximum allowed gap in the contour ($tc \in \{50, 75, 100\}$ ms) and voicing tolerance parameter ($\nu \in \{-1, 0.2, 1, 1.4\}$). For the sake of clarity, we only show results from CBM, since the highest overall accuracy was obtained with this method. Results obtained with other methods are also presented, including the effect of N_{iter} on DUR. Best results in vocal music are obtained with few iterations, but complex data (such as instrumental, and especially orchestral music) benefits from a higher number of iterations.

In any case, the influence of pitch salience creation parameters is relatively small in comparison to the influence

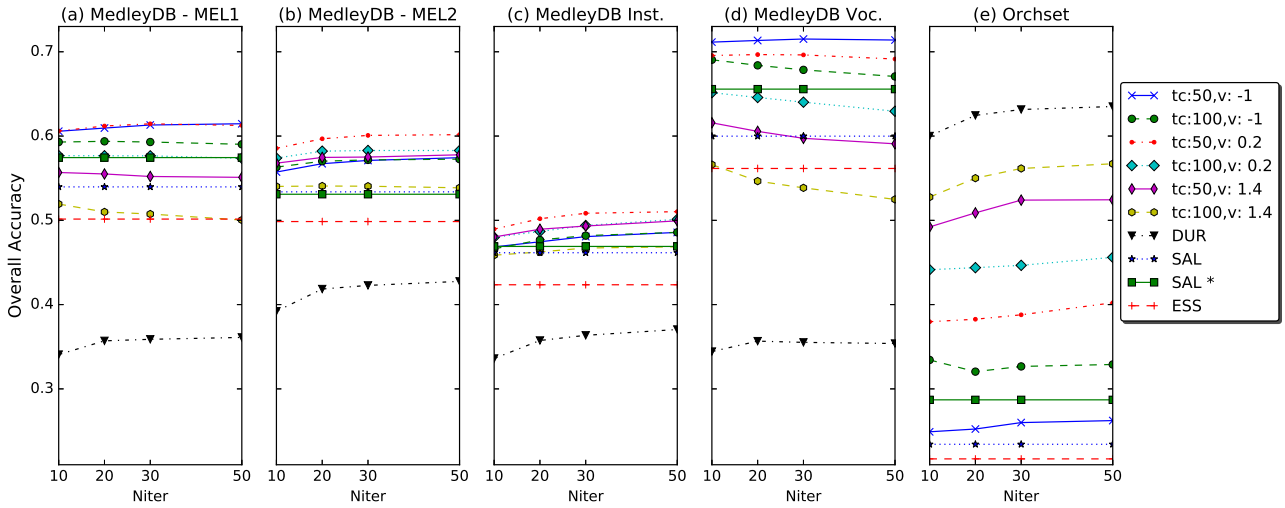

Figure 7: Overall accuracy vs. number of iterations, and influence of time continuity (tc), and voicing (ν) parameters on the results obtained with CBM. Results for DUR, SAL and the baseline (ESS) given as a reference. "SAL*" denotes that $\nu = -1$ for MedleyDB and $\nu = 1.4$ for Orchset (a) MedleyDB (MEL1) definition; (b) MedleyDB (MEL2); (c) MedleyDB (MEL1), instrumental songs; (d) MedleyDB (MEL1), vocal songs; (e) Orchset.

of pitch contour tracking parameters. For instance, OA generally increases considerably when the maximum gap between pitches in a contour is decreased from 100 ms to 50 ms. This is probably due to the noise added in unvoiced frames by the SIMM, which can partially be filtered in the contour creation process. The effect of the voicing parameter (ν) is evident: a higher value increases the voicing threshold and less contours are filtered, which is beneficial in Orchset. Setting a lower threshold is benefitial in MedleyDB with MEL1 definition, since the amount of voiced frames is smaller. Default peak filtering parameter values (τ_σ, τ_+) provided good results in MedleyDB, but OA can be increased up to 60% in Orchset, by increasing τ_σ from 0.9 to 1.3 with $\nu = 1.4$. This allows a higher difference in salience below the salience mean during pitch contour creation, which is appropriate to deal with the higher dynamic range in classical music.

Regarding instrumentation, OA in MedleyDB vocal music is higher than in instrumental, but with the proposed method, we increased it in about 10 and 8 percentage points (pp) over the baseline (ESS) respectively. The improvement is even more evident in Orchset. According to the results, we can conclude that our salience function leads to a better accuracy than HS, for both single instruments and instrument sections.

CBM obtained 25 percentage points (pp) higher OA in MedleyDB (with MEL1 definition, see Figure 7) compared to DUR, and slightly worse in Orchset (around 4 pp with the best parameters mentioned. Additionally, CBM generally needs less iterations (N_{iter}) compared to DUR to achieve the best results, which is very positive given the high computational weight of the estimation algorithm. In comparison to the approach by Salamon et al., we obtained 5 and 30 pp higher accuracy in MedleyDB (with MEL1 definition) and Orchset respectively, using the best voicing parameter for each dataset in both algorithms (CBM and SAL*). This corresponds to about 10% and 100% relative increase, due to the low accuracy of SAL in Orchset.

The selection of parameters has been here performed automatically, but it could be performed automatically by selecting the best performing configuration in a training set. Another possibility is to use a pitch contour classification approach [10], by training a classifier to distinguish between melody and non-melody pitch contours using the proposed salience functions [22].

5. CONCLUSIONS

This paper presents a melody extraction method based on the combination of a source-filter model and pitch contour based tracking. We proposed two different salience functions and we have shown that Gaussian filtering and the combination of a source-filter model with harmonic summation help increasing the salience of melody pitches. The signal representation employed proved to improve pitch estimation accuracy and to reduce octave errors in comparison to harmonic summation. Our complete melody extraction method obtains similar or higher overall accuracy in comparison to similar approaches, when evaluated on a large and varied dataset. This is achieved by accurate voicing detection and pitch estimation.

Future work deals with improving the salience function, in order to further reduce the amount of noise in unvoiced parts, and to improve the adaptation to the contour creation process. We also foresee the use of a supervised method for pitch contour classification and melody tracking.

Acknowledgments

This work is partially supported by the European Union under the PHENICX project (FP7-ICT-601166) and the Spanish Ministry of Economy and Competitiveness under CASAS project (TIN2015-70816-R) and Maria de Maeztu Units of Excellence Programme (MDM-2015-0502).

6. REFERENCES

[1] J. Salamon, E. Gómez, D. Ellis, and G. Richard, "Melody Extraction from Polyphonic Music Signals: Approaches, applications, and challenges," *IEEE Signal Process. Mag.*, vol. 31, pp. 118–134, 2014.

[2] A. Klapuri, "Multiple fundamental frequency estimation by summing harmonic amplitudes," in *Proc. ISMIR*, 2006, pp. 216–221.

[3] J. Salamon and E. Gómez, "Melody extraction from polyphonic music signals using pitch contour characteristics," *IEEE Trans. Audio. Speech. Lang. Processing*, vol. 20, no. 6, pp. 1759–1770, 2012.

[4] K. Dressler, "Towards Computational Auditory Scene Analysis: Melody Extraction from Polyphonic Music," in *Proc. CMMR*, 2012.

[5] ——, "Multiple fundamental frequency extraction for MIREX 2012," in *Music Inf. Retr. Eval. Exch.*, 2012.

[6] J. Durrieu, B. David, and G. Richard, "A musically motivated mid-level representation for pitch estimation and musical audio source separation," *Sel. Top. Signal Process. IEEE J.*, vol. 5, no. 6, pp. 1180–1191, 2011.

[7] B. Fuentes, A. Liutkus, R. Badeau, and G. Richard, "Probabilistic model for main melody extraction using constant-Q transform," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 5357–5360.

[8] J. Durrieu, G. Richard, B. David, and C. Févotte, "Source/filter model for unsupervised main melody extraction from polyphonic audio signals," *Audio, Speech, Lang. Process. IEEE Trans.*, vol. 18, no. 3, pp. 564–575, 2010.

[9] J. Salamon, G. Peeters, and A. Röbel, "Statistical characterisation of melodic pitch contours and its application for melody extraction," in *13th Int. Soc. for Music Info. Retrieval Conf.*, Porto, Portugal, Oct. 2012, pp. 187–192.

[10] R. Bittner, J. Salamon, S. Essid, and J. Bello, "Melody extraction by contour classification," in *Proc. International Society of Music Information Retrieval (ISMIR)*, October 2015.

[11] S. Ternström and J. Sundberg, "Intonation precision of choir singers," *The Journal of the Acoustical Society of America*, vol. 84, no. 1, pp. 59–69, 1988.

[12] R. Bittner, J. Salamon, M. Tierney, M. Mauch, C. Cannam, and J. Bello, "Medleydb: a multitrack dataset for annotation-intensive mir research," in *Proc. ISMIR*, 2014, pp. 155–160.

[13] J. Bosch, R. Marxer, and E. Gómez, "Evaluation and combination of pitch estimation methods for melody extraction in symphonic classical music," *Journal of New Music Research*, DOI: 10.1080/09298215.2016.1182191, 2016.

[14] J. S. Downie, "The music information retrieval evaluation exchange (2005-2007): A window into music information retrieval research," *Acoustical Science and Technology*, vol. 29, no. 4, pp. 247–255, 2008.

[15] J. Bosch and E. Gómez, "Melody extraction in symphonic classical music: a comparative study of mutual agreement between humans and algorithms," in *Proc. 9th Conference on Interdisciplinary Musicology – CIM14*, Berlin, 2014.

[16] C. Hsu and J. Jang, "Singing pitch extraction by voice vibrato/tremolo estimation and instrument partial deletion," in *Proc. ISMIR*, 2010, pp. 525–530.

[17] T. Yeh, M. Wu, J. Jang, W. Chang, and I. Liao, "A hybrid approach to singing pitch extraction based on trend estimation and hidden markov models," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 457–460.

[18] D. Bogdanov, N. Wack, E. Gómez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J. Zapata, and X. Serra, "Essentia: an open source library for audio analysis," *ACM SIGMM Records*, vol. 6, 2014.

[19] J. Salamon, E. Gómez, and J. Bonada, "Sinusoid extraction and salience function design for predominant melody estimation," in *Proc. 14th Int. Conf. Digit. Audio Eff. (DAFx-11), Paris, Fr.*, 2011, pp. 73–80.

[20] C. Raffel, B. McFee, E. Humphrey, J. Salamon, O. Nieto, D. Liang, and D. Ellis, "mir eval: A transparent implementation of common mir metrics," in *Proc. ISMIR*, 2014.

[21] M. Goto, "A Real-Time Music-Scene-Description System: Predominant-F0 Estimation for Detecting Melody and Bass Lines in Real-World Audio Signals," *Speech Communication*, vol. 43, no. 4, pp. 311–329, Sep. 2004.

[22] J. Bosch, R. M. Bittner, J. Salamon, and E. Gómez, "A comparison of melody extraction methods based on source-filter modelling," in *Proc. ISMIR*, New York, Aug. 2016.