SINUSOID EXTRACTION AND SALIENCE FUNCTION DESIGN FOR PREDOMINANT MELODY ESTIMATION

Justin Salamon, Emilia Gómez and Jordi Bonada, *

Music Technology Group
Universitat Pompeu Fabra, Barcelona, Spain
{justin.salamon,emilia.gomez,jordi.bonada}@upf.edu

ABSTRACT

In this paper we evaluate some of the alternative methods commonly applied in the first stages of the signal processing chain of automatic melody extraction systems. Namely, the first two stages are studied – the extraction of sinusoidal components and the computation of a time-pitch salience function, with the goal of determining the benefits and caveats of each approach under the specific context of predominant melody estimation. The approaches are evaluated on a data-set of polyphonic music containing several musical genres with different singing/playing styles, using metrics specifically designed for measuring the usefulness of each step for melody extraction. The results suggest that equal loudness filtering and frequency/amplitude correction methods provide significant improvements, whilst using a multi-resolution spectral transform results in only a marginal improvement compared to the standard STFT. The effect of key parameters in the computation of the salience function is also studied and discussed.

1. INTRODUCTION

To date, various different methods and systems for automatic melody extraction from polyphonic music have been proposed, as evident by the many submissions to the MIREX automatic melody extraction evaluation campaign[1]. In [1], a basic processing structure underlying melody extraction systems was described comprising three main steps – multi-pitch extraction, melody identification and post-processing. Whilst alternative designs have been proposed [2], it is still the predominant architecture in most current systems [3, 4, 5, 6]. In this paper we focus on the first stage of this architecture, i.e. the multi-pitch extraction. In most cases this stage can be broken down into two main steps – the extraction of sinusoidal components, and the use of these components to compute a representation of pitch salience over time, commonly known as a *Salience Function*. The salience function is then used by each system to determine the pitch of the main melody in different ways.

Whilst this overall architecture is common to most systems, they use quite different approaches to extract the sinusoidal components and then compute the salience function. For extracting sinusoidal components, some systems use the standard Short-Time Fourier Transform (STFT), whilst others use a multi-resolution transform in an attempt to overcome the time-frequency resolution trade-off inherent to the FFT [7, 8, 9]. Some systems apply filters to the audio signal in attempt to enhance the spectrum of the melody before performing spectral analysis, such as bandpass [7] or equal loudness filtering [6]. Others apply spectral whitening to make the analysis robust against changes in timbre [3]. Finally, given the spectrum, different approaches exist for estimating the peak frequency and amplitude of each spectral component.

Once the spectral components are extracted, different methods have been proposed for computing the time-frequency salience function. Of these, perhaps the most common type is based on harmonic summation [3, 4, 5, 6]. Within this group various approaches can be found, differing primarily in the weighting of harmonic peaks in the summation and the number of harmonics considered. Some systems also include a filtering step before the summation to exclude some spectral components based on energy and sinusoidality criteria [8] or spectral noise suppression [10].

Whilst the aforementioned systems have been compared in terms of melody extraction performance (c.f. MIREX), their overall complexity makes it hard to determine the effect of the first steps in each system on the final result. In this paper we aim to evaluate the first two processing steps (sinusoid extraction and salience function) alone, with the goal of understanding the benefits and caveats of the alternative approaches and how they might affect the rest of the system. Whilst some of these approaches have been compared in isolation before [9], our goal is to evaluate them under the specific context of melody extraction. For this purpose, a special evaluation framework, data-sets and metrics have been developed. In section 2 we described the different methods compared for extracting sinusoidal components, and in section 3 we describe the design of the salience function and the parameters affecting its computation. In section 4 we explain the evaluation framework used to evaluate both the sinusoid extraction and salience function design, together with the ground truth and metrics used. Finally, in section 5 we provide and discuss the results of the evaluation, summarised in the conclusions of section 6.

2. METHODS FOR SINUSOID EXTRACTION

The first step of many systems involves obtaining spectral components (peaks) from the audio signal, also referred to as the *front end* [7]. As mentioned earlier, different methods have been proposed to obtain the spectral peaks, usually with two common goals in mind – firstly, extracting the spectral peaks as accurately as possible in terms of their frequency and amplitude. Secondly, some systems attempt to enhance the amplitude of melody peaks whilst suppressing that of background peaks by applying some pre-filtering. For the purpose of our evaluation we have divided this process into three main steps, in each of which we consider two or three alternative approaches proposed in the literature. The alternatives considered at each step are summarised in Table 1.

* This research was funded by the Programa de Formación del Profesorado Universitario of the Ministerio de Educación de España, COFLA (P09-TIC-4840-JA) and DRIMS (TIN2009-14247-C02-01-MICINN).

[1] http://www.music-ir.org/mirex/wiki/MIREX_HOME

Table 1: *Analysis alternatives for sinusoid extraction.*

Filtering	**Spectral Transform**	**Frequency/Amplitude Correction**
none Equal Loudness	STFT MRFFT	none Parabolic Interpolation Phase Vocoder

2.1. Filtering

As a first step, some systems filter the time signal in attempt to enhance parts of the spectrum more likely to pertain to the main melody, for example band-pass filtering [7]. For this evaluation we consider the more perceptually motivated equal loudness filtering. The equal loudness curves [11] describe the human perception of loudness as dependent on frequency. The equal loudness filter takes a representative average of these curves, and filters the signal by its inverse. In this way frequencies we are perceptually more sensitive to are enhanced in the signal, and frequencies we are less sensitive to are attenuated.

Further details about the implementation of the filter can be found here[2]. It is worth noting that in the low frequency range the filter acts as a high pass filter with a high pass frequency of 150Hz. In our evaluation two alternatives are considered – equal loudness filtering, and no filtering[3].

2.2. Spectral Transform

As previously mentioned, a potential problem with the STFT is that it has a fixed time and frequency resolution. When analysing an audio signal for melody extraction, it might be beneficial to have greater frequency resolution in the low frequencies where peaks are bunched closer together and are relatively stationary over time, and higher time resolution for the high frequencies where we can expect peaks to modulate rapidly over time (e.g. the harmonics of singing voice with a deep vibrato). In order to evaluate whether the use of a single versus multi-resolution transform is significant, two alternative transforms were implemented, as detailed below.

2.2.1. Short-Time Fourier Transform (Single Resolution)

The STFT can be defined as follows:

$$X_l(k) = \sum_{n=0}^{M-1} w(n) \cdot x(n + lH) e^{-j\frac{2\pi}{N}kn}, \qquad (1)$$

$$l = 0, 1, \ldots \text{ and } k = 0, 1, \ldots, N-1$$

where $x(n)$ is the time signal, $w(n)$ the windowing function, l the frame number, M the window length, N the FFT length and H the hop size. We use the Hann windowing function with a window size of 46.4ms, a hop size of 2.9ms and a $\times 4$ zero padding factor. The evaluation data is sampled at $f_S = 44.1$kHz, giving $M = 2048$, $N = 8192$ and $H = 128$.

Given the FFT of a single frame $X(k)$, peaks are selected by finding all the local maxima k_m of the normalised magnitude spectrum $X_m(k)$:

$$X_m(k) = 2\frac{|X(k)|}{\sum_{n=0}^{M-1} w(n)}. \qquad (2)$$

Peaks with a magnitude more than 80dB below the highest spectral peak in an excerpt are not considered.

2.2.2. Multi-Resolution FFT

We implemented the multi-resolution FFT (MRFFT) proposed in [8]. The MRFFT is an efficient algorithm for simultaneously computing the spectrum of a frame using different window sizes, thus allowing us to choose which window size to use depending on whether we require high frequency resolution (larger window size) or high time resolution (smaller window size). The algorithm is based on splitting the summations in the FFT into smaller sums which can be combined in different ways to form frames of varying sizes, and performing the windowing in the frequency domain by convolution. The resulting spectra all have the same FFT length N (i.e. smaller windows are zero padded) and use the Hann windowing function. For further details about the algorithm the reader is referred to [8].

In our implementation we set $N = 8192$ and $H = 128$ as with the STFT so that they are comparable. We compute four spectra $X_{256}(k)$, $X_{512}(k)$, $X_{1024}(k)$ and $X_{2048}(k)$ with respective window sizes of $M = 256$, 512, 1024 and 2048 samples (all windows are centered on the same sample). Then, local maxima (peaks) are found in each magnitude spectrum within a set frequency range as in [8], using the largest window (2048 samples) for the first six critical bands of the Bark scale (0-630Hz), the next window for the following five bands (630-1480Hz), the next one for the following five bands (1480-3150Hz) and the smallest window (256 samples) for the remaining bands (3150-22050Hz). The peaks from the different windows are combined to give a single set of peaks at positions k_m, and (as with the STFT) peaks with a magnitude more than 80dB below the highest peak in an excerpt are not considered.

2.3. Frequency and Amplitude Correction

Given the set of local maxima (peaks) k_m, the simplest approach for calculating the frequency and amplitude of each peak is to directly use its spectral bin and FFT magnitude (as detailed in equations 3 and 4 further down). This approach is limited by the frequency resolution of the FFT. For this reason various correction methods have been developed to achieve a higher frequency precision, and a better amplitude estimation as a result. In [12] a survey of these methods is provided for artificial, monophonic stationary sounds. Our goal is to perform a similar evaluation for real-world, polyphonic, quasi-stationary sounds (as is the case in melody extraction). For our evaluation we consider three of the methods discussed in [12], which represent three different underlying approaches:

2.3.1. Plain FFT with No Post-processing

Given a peak at bin k_m, its sine frequency and amplitude are calculated as follows:

$$\hat{f} = k_m \frac{f_S}{N} \qquad (3)$$

$$\hat{a} = X_m(k_m) \qquad (4)$$

[2] http://replaygain.hydrogenaudio.org/equal_loudness.html

[3] Spectral whitening/noise suppression is left for future work.

Note that the frequency resolution is limited by the size of the FFT, in our case the frequency values are limited to multiples of $f_S/N = 5.38$Hz. This also results in errors in the amplitude estimation as it is quite likely for the true peak location to fall between two FFT bins, meaning the detected peak is actually lower (in magnitude) than the true magnitude of the sinusoidal component.

2.3.2. Parabolic Interpolation

This method improves the frequency and amplitude estimation of a peak by taking advantage of the fact that in the magnitude spectrum of most analysis windows (including the Hann window), the shape of the main lobe resembles a parabola in the dB scale. Thus, we can use the bin value and magnitude of the peak together with that of its neighbouring bins to estimate the position (in frequency) and amplitude of the true maximum of the main lobe, by fitting them to a parabola and finding its maximum. Given a peak at bin k_m, we define:

$$A_1 = X_{dB}(k_m-1), A_2 = X_{dB}(k_m), A_3 = X_{dB}(k_m+1), \quad (5)$$

where $X_{dB}(k) = 20\log_{10}(X_m(k))$. The frequency difference in FFT bins between k_m and the true peak of the parabola is given by:

$$d = 0.5\frac{A_1 - A_3}{A_1 - 2A_2 + A_3}. \quad (6)$$

The corrected peak frequency and amplitude (this time in dB) are thus given by:

$$\hat{f} = (k_m + d)\frac{f_S}{N} \quad (7)$$

$$\hat{a} = A_2 - \frac{d}{4}(A_1 - A_3) \quad (8)$$

Note that following the results of [12], the amplitude is not estimated using equation 8 above, but rather with equation 11 below, using the value of d as the bin offset $\kappa(k_m)$.

2.3.3. Instantaneous Frequency using Phase Vocoder

This approach uses the phase spectrum $\phi(k)$ to calculate the peak's instantaneous frequency (IF) and amplitude, which serve as a more accurate estimation of its true frequency and amplitude. The IF is computed from the phase difference $\Delta\phi(k)$ of successive phase spectra using the phase vocoder method [13] as follows:

$$\hat{f} = (k_m + \kappa(k_m))\frac{f_S}{N}, \quad (9)$$

where the bin offset $\kappa(k)$ is calculated as:

$$\kappa(k) = \frac{N}{2\pi H}\Psi\left(\phi_l(k) - \phi_{l-1}(k) - \frac{2\pi H}{N}k\right), \quad (10)$$

where Ψ is the principal argument function which maps the phase to the $\pm\pi$ range.

The instantaneous magnitude is calculated using the peak's spectral magnitude $X_m(k_m)$ and the bin offset $\kappa(k_m)$ as follows:

$$\hat{a} = \frac{1}{2}\frac{X_m(k_m)}{W_{Hann}\left(\frac{M}{N}\kappa(k_m)\right)}, \quad (11)$$

where W_{Hann} is the Hann window kernel:

$$W_{Hann}(\kappa) = \frac{1}{2}\frac{\text{sinc}(\kappa)}{1-\kappa^2}, \quad (12)$$

and *sinc* is the normalised sinc function. To achieve the best phase-based correction we use $H = 1$, by computing at each hop (of 128 samples) the spectrum of the current frame and of a frame shifted back by one sample, and using the phase difference between the two.

3. SALIENCE FUNCTION DESIGN

Once the spectral peaks are extracted, they are used to construct a salience function - a representation of frequency salience over time. For this study we use a common approach for salience computation based on harmonic summation, which was used as part of a complete melody extraction system in [6]. Basically, the salience of a given frequency is computed as the sum of the weighted energy of the spectral peaks found at integer multiples (harmonics) of the given frequency. As such, the important factors affecting the salience computation are the number of harmonics considered N_h and the weighting scheme used. In addition, we can add a relative magnitude filter, only considering for the summation peaks whose magnitude is no less than a certain threshold γ (in dB) below the magnitude of the highest peak in the frame. Note that the proposed salience function was designed as part of a system which handles octave errors and the selection of the melody pitch at a later stage, hence whilst the salience function is designed to best enhance melody salience compared to other pitched sources, these issues are not addressed directly by the salience function itself.

Our salience function covers a pitch range of nearly five octaves from 55Hz to 1.76kHz, quantized into $n = 1\ldots600$ bins on a cent scale (10 cents per bin). Given a frequency f_i in Hz, its corresponding bin $b(f_i)$ is calculated as:

$$b(f_i) = \left\lfloor \frac{1200\left(\log_2(\frac{f_i}{13.75}) - 0.25\right) - 2100}{10} + 1 \right\rfloor. \quad (13)$$

At each frame the salience function $S(n)$ is constructed using the spectral peaks p_i (with frequencies f_i and linear magnitudes m_i) found in the frame during the previous analysis step. The salience function is defined as:

$$S(n) = \sum_{h=1}^{N_h}\sum_{p_i} e(m_i)\cdot g(n,h,f_i)\cdot(m_i)^\beta, \quad (14)$$

where β is a parameter of the algorithm, $e(m_i)$ is a magnitude filter function, and $g(n, f_i, h)$ is the function that defines the weighting scheme. The magnitude filter function is defined as:

$$e(m_i) = \begin{cases} 1 & \text{if } 20\log_{10}(m_M/m_i) < \gamma, \\ 0 & \text{otherwise,} \end{cases} \quad (15)$$

where m_M is the magnitude of the highest peak in the frame. The weighting function $g(n, f_i, h)$ defines the weight given to peak p_i, when it is considered as the h^{th} harmonic of bin n:

$$g(n,h,f_i) = \begin{cases} \cos^2(\delta\cdot\frac{\pi}{2})\cdot\alpha^{h-1} & \text{if } |\delta| \le 1, \\ 0 & \text{if } |\delta| > 1, \end{cases} \quad (16)$$

where $\delta = |b(f_i/h) - n|/10$ is the distance in semitones between the harmonic frequency f_i/h and the centre frequency of bin n and α is the harmonic weighting parameter. The threshold for δ means that each peak contributes not just to a single bin of the salience function but also to the bins around it (with $\cos^2$ weighting). This avoids potential problems that could arise due to the quantization of the salience function into bins, and also accounts for inharmonicities.

In sections 4 and 5 we will examine the effect of each of the aforementioned parameters on the salience function, in attempt to select a parameter combination most suitable for a salience function targeted at melody extraction. The parameters studied are the weighting parameters α and β, the magnitude threshold γ and the number of harmonics N_h used in the summation.

4. EVALUATION

The evaluation is split into two parts. First, we evaluate the different analysis approaches for extracting sinusoids in a similar way to [12]. The combination of different approaches at each step (filtering, transform, correction) gives rise to 12 possible analysis configurations, summarised in Table 2. In the second part, we evaluate the sinusoid extraction combined with the salience function computed using different parameter configurations. In the following sections we describe the experimental setup, ground truth and metrics used for each part of the evaluation.

Table 2: *Analysis Configurations.*

Conf.	Filtering	Spectral Transform	Frequency/Amplitude Correction
1	none	STFT	none
2			Parabolic
3			Phase
4		MRFFT	none
5			Parabolic
6			Phase
7	Eq. Loudness	STFT	none
8			Parabolic
9			Phase
10		MRFFT	none
11			Parabolic
12			Phase

4.1. Sinusoid Extraction

4.1.1. Ground Truth

Starting with a multi-track recording, the ground truth is generated by analysing the melody track on its own as in [14] to produce a per-frame list of f0 + harmonics (up to the Nyquist frequency) with frequency and amplitude values. The output of the analysis is then re-synthesised using additive synthesis with linear frequency interpolation and mixed together with the rest of the tracks in the recording. The resulting mix is used for evaluating the different analysis configurations by extracting spectral peaks at every frame and comparing them to the ground truth. In this way we obtain a melody ground truth that corresponds perfectly to the melody in the mixture, whilst being able to use real music as opposed to artificial mixtures.

As we are interested in the melody, only voiced frames are used for the evaluation (i.e. frames where the melody is present). Furthermore, some of the melody peaks will be masked in the mix by the spectrum of the accompaniment, where the degree of masking depends on the analysis configuration used. Peaks detected at frequencies where the melody is masked by the background depend on the background spectrum and hence should not be counted as successfully detected melody peaks. To account for this, we compute the spectra of the melody track and the background separately, using the analysis configuration being evaluated. We then check for each peak extracted from the mix by the analysis whether the melody spectrum is masked by the background spectrum at the peak frequency (a peak is considered to be masked if the spectral magnitude of the background is greater than that of the melody for the corresponding bin), and if so the peak is discarded.

The evaluation material is composed of excerpts from real-world recordings in various genres, summarised in Table 3.

Table 3: *Ground Truth Material.*

Genre	Excerpts	Tot. Melody Frames	Tot. Ground Truth Peaks
Opera	5	15,660	401,817
Pop/Rock	3	11,760	769,193
Instrumental Jazz	4	16,403	587,312
Bossa Nova	2	7,160	383,291

4.1.2. Metrics

We base our metrics on the ones used in [12], with some adjustments to account for the fact that we are only interested in the spectral peaks of the melody within a polyphonic mixture.

At each frame, we start by checking which peaks found by the algorithm correspond to peaks in the ground truth (melody peaks). A peak is considered a match if it is within 21.5Hz (equivalent to 1 FFT bin without zero padding) from the ground truth. If more than one match is found, we select the peak closest in amplitude to the ground truth. Once the matching peaks in all frames are identified, we compute the metrics R_p and R_e as detailed in Table 4.

Table 4: *Metrics for sinusoid extraction.*

R_p	Peak recall. The total number of melody peaks found by the algorithm in all frames divided by the total number of peaks in the ground truth.
R_e	Energy recall. The sum of the energy of all melody peaks found by the algorithm divided by the total energy of the peaks in the ground truth.
$\overline{\Delta a_{dB}}$	Mean amplitude error (in dB) of all detected melody peaks.
$\overline{\Delta f_c}$	Mean frequency error (in cents) of all detected melody peaks.
$\overline{\Delta f_w}$	Mean frequency error (in cents) of all detected melody peaks weighted by the normalised peak energy.

Given the matching melody peaks, we can compute the frequency estimation error Δf_c and the amplitude estimation error Δa_{dB} of each peak[4]. The errors are measured in cents and dBs respectively, and averaged over all peaks of all frames to give $\overline{\Delta f_c}$ and $\overline{\Delta a_{dB}}$. A potential problem with $\overline{\Delta f_c}$ is that the mean may be dominated by peaks with very little energy (especially at high frequencies), even though their effect on the harmonic summation later on will be insignificant. For this reason we define a third measure $\overline{\Delta f_w}$, which is the mean frequency error in cents where each peak's contribution is weighted by its energy, normalised by the energy of the highest peak in the ground truth in the same frame. The normalisation ensures the weighting is independent of the volume of each excerpt[5]. The metrics are summarised above in Table 4.

4.2. Salience Function Design

In the second part of the evaluation we take the spectral peaks produced by each one of the 12 analysis configurations and use them to compute the salience function with different parameter configurations. The salience function is then evaluated in terms of its usefulness for melody extraction using the ground truth and metrics detailed below.

4.2.1. Ground Truth

We use the same evaluation material as in the previous part of the evaluation. The first spectral peak in every row of the ground truth represents the melody f0, and is used to evaluate the frequency accuracy of the salience function as explained below.

4.2.2. Metrics

We evaluate the salience function in terms of two aspects – frequency accuracy and melody salience, where melody salience should reflect the predominance of the melody compared to the other pitched elements appearing in the salience function. Four metrics have been devised for this purpose, computed on a per-frame basis and finally averaged over all frames.

We start by selecting the peaks of the salience function. The salience peak closest in frequency to the ground truth f0 is considered the melody salience peak. We can then calculate the frequency error of the salience function Δf_m as the difference in cents between the frequency of the melody salience peak and the ground truth f0.

To evaluate the predominance of the melody three metrics are computed. The first is the rank R_m of the melody salience peak amongst all salience peaks in the frame, which ideally should be 1. Rather than report the rank directly we compute the reciprocal rank $RR_m = 1/R_m$ which is less sensitive to outliers when computing the mean over all frames. The second is the relative salience S_1 of the melody peak, computed by dividing the salience of the melody peak by that of the highest peak in the frame. The third metric, S_3, is the same as the previous one only this time we divide the salience of the melody peak by the mean salience of the top 3 peaks of the salience function. In this way we can measure not only whether the melody salience peak is the highest, but also whether it stands out from the other peaks of the salience function and by how much. The metrics are summarised in Table 5.

Table 5: *Metrics for evaluating Salience Function Design.*

Δf_m	Melody frequency error.
RR_m	Reciprocal Rank of the melody salience peak amongst all peaks of the salience function.
S_1	Melody salience compared to top peak.
S_3	Melody salience compared to top 3 peaks.

5. RESULTS

The results are presented in two stages. First we present the results for the sinusoid extraction, and then the results for the salience function design. In both sections, each metric is evaluated for each of the 12 possible analysis configurations summarised in Table 2.

5.1. Sinusoid Extraction

We start by examining the results obtained when averaging over all genres, provided in Table 6. The best result in each column is highlighted in bold. Recall that R_p and R_e should be maximised whilst $\overline{\Delta a_{dB}}$, $\overline{\Delta f_c}$ and $\overline{\Delta f_w}$ should be minimised.

Table 6: *Sinusoid extraction results for all genres.*

Conf.	R_p	R_e	$\overline{\Delta a_{dB}}$	$\overline{\Delta f_c}$	$\overline{\Delta f_w}$
1	**0.62**	**0.88**	3.03	3.17	8.77
2	**0.62**	**0.88**	3.02	2.89	7.20
3	**0.62**	**0.88**	3.02	**2.88**	6.91
4	0.29	0.84	1.43	5.21	9.60
5	0.29	0.84	1.43	4.75	7.99
6	0.31	0.85	1.46	4.35	7.40
7	0.55	**0.88**	2.79	3.47	8.10
8	0.55	**0.88**	2.78	3.16	6.69
9	0.54	**0.88**	2.78	3.13	**6.45**
10	0.27	0.83	**1.41**	5.63	9.04
11	0.27	0.83	**1.41**	5.13	7.58
12	0.27	0.84	1.45	4.84	7.03

We see that regardless of the filtering and transform used, both parabolic and phase based correction provide an improvement in frequency accuracy (i.e. lower $\overline{\Delta f_c}$ values), with the phase based method providing just slightly better results. The benefit of using frequency correction is further accentuated when considering $\overline{\Delta f_w}$. As expected, there is no significant difference between the amplitude error $\overline{\Delta a_{dB}}$ when correction is applied and when it is not, as the error is dominated by the spectrum of the background.

When considering the difference between using the STFT and MRFFT, we first note that there is no significant improvement in frequency accuracy (i.e. smaller frequency error) when using the MRFFT (for all correction options), as indicated by both $\overline{\Delta f_c}$ and $\overline{\Delta f_w}$. This suggests that whilst the MRFFT might be advantageous for certain types of data (c.f. results for opera in Table 7), when averaged over all genres the method does not provide a significant improvement in frequency accuracy.

[4] As we are using polyphonic material the amplitude error may not reflect the accuracy of the method being evaluated, and is included for completeness.

[5] Other weighting schemes were tested and shown to produce very similar results.

When we turn to examine the peak and energy recall, we see that the STFT analysis finds more melody peaks, however, interestingly both transforms obtain a similar degree of energy recall. This implies that the MRFFT, which generally finds less peaks (due to masking caused by wider peak lobes), still finds the most important melody peaks. Whether this is significant or not for melody extraction should become clearer in the second part of the evaluation when examining the salience function.

Next, we observe the effect of applying the equal loudness filter. We see that peak recall is significantly reduced, but that energy recall is maintained. This implies that the filter does not attenuate the most important melody peaks. If, in addition, the filter attenuates some background peaks, the overall effect would be that of enhancing the melody. As with the spectral transform, the significance of this step will become clearer when evaluating the salience function.

Finally, we provide the results obtained for each genre separately in Table 7 (for brevity only configurations which obtain the best result for at least one of the metrics are included). We can see that the above observations hold for the individual genres as well. The only interesting difference is that for the opera genre the MRFFT gives slightly better overall results compared to the STFT. This can be explained by the greater pitch range and deep vibrato which often characterise the singing in this genre. The MRFFT's increased time resolution at higher frequencies means it is better at estimating the rapidly changing harmonics present in opera singing.

Table 7: *Sinusoid extraction results per genre.*

Genre	Conf.	R_p	R_e	$\overline{\Delta a_{dB}}$	$\overline{\Delta f_c}$	$\overline{\Delta f_w}$
Opera	2	**0.73**	0.83	3.74	3.97	7.48
	6	0.59	**0.93**	1.15	**3.66**	6.50
	11	0.53	0.92	**1.08**	3.88	**5.91**
Jazz	3	**0.57**	**0.96**	2.20	**2.33**	6.23
	9	0.56	**0.96**	2.18	2.36	**5.75**
	10	0.20	0.84	**1.57**	7.88	10.95
Pop/Rock	2	**0.54**	**0.84**	3.08	**3.05**	7.71
	3	**0.54**	0.83	3.08	**3.05**	7.43
	9	0.46	**0.84**	2.89	3.37	**6.83**
	11	0.17	0.73	**1.86**	6.73	8.97
Bossa Nova	2	**0.76**	0.91	3.17	**1.95**	5.75
	8	0.56	**0.92**	2.74	2.32	5.48
	9	0.56	**0.92**	2.74	2.36	**5.30**
	10	0.29	0.86	**1.33**	4.19	8.00

5.2. Salience Function Design

As explained in section 3, in addition to the analysis configuration used, the salience function is determined by four main parameters – the weighting parameters α and β, the energy threshold γ and the number of harmonics N_h. To find the best parameter combination for each analysis configuration and to study the interaction between the parameters, we performed a grid search of these four parameters using several representative values for each parameter: $\alpha = 1, 0.9, 0.8, 0.6$, $\beta = 1, 2$, $\gamma = \infty$, 60dB, 40dB, 20dB, and $N_h = 4, 8, 12, 20$. This results in 128 possible parameter combinations which were used to compute the salience function metrics for each of the 12 analysis configurations.

We started by plotting a graph for each metric with a data point for each of the 128 parameter combinations, for the 12 analysis configurations[6]. At first glance it was evident that for all analysis and parameter configurations the results were consistently better when $\beta = 1$, thus only the 64 parameter configurations in which $\beta = 1$ shall be considered henceforth.

5.2.1. Analysis Configuration

We start by examining the effect of the analysis configuration on the salience function. In Figure 1 we plot the results obtained for each metric by each configuration. For comparability the salience function is computed using the same (optimal) parameter values ($\alpha = 0.8$, $\beta = 1$, $\gamma = 40$dB, $N_h = 20$) for all analysis configurations (the parameter values are discussed in section 5.2.2). Configurations that only differ in the filtering step are plotted side by side. Metrics Δf_m, RR_m, S_1 and S_3 are displayed in plots (a), (b), (c) and (d) of Figure 1 respectively.

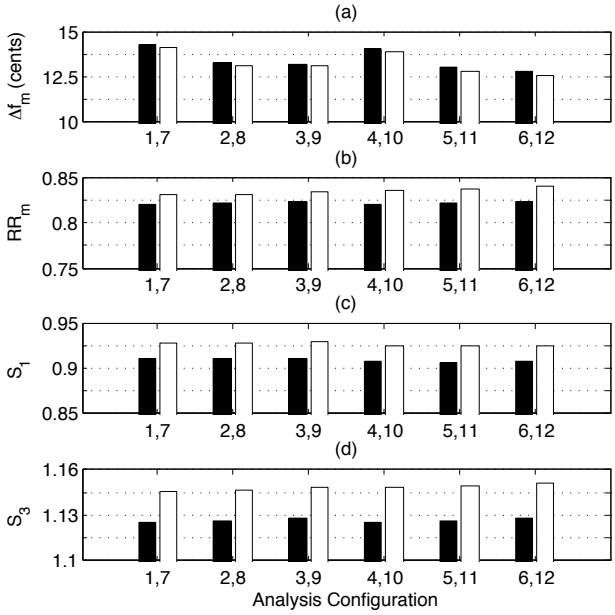

Figure 1: *Salience function design, overall results. Each bar represents an analysis configuration, where white bars are configurations which apply equal loudness filtering. Recall that Δf_m should be minimised whilst RR_m, S_1 and S_3 should be maximised.*

The first thing we see is that for all metrics, results are always improved when equal loudness filtering is applied. This confirms our previous stipulation that the filter enhances the melody by attenuating non-melody spectral peaks. It can be explained by the filter's enhancement of the mid-band frequencies which is where the melody is usually present, and the attenuation of low-band frequencies where we expect to find low pitched instruments such as the bass.

Next we examine the frequency error Δf_m in Figure 1 plot (a). We see that there is a (significant) decrease in the error when either of the two correction methods (parabolic interpolation or phase vocoder) are applied, as evident by comparing configurations 1, 7, 4, 10 (no correction) to the others. Though the error

[6]For brevity these plots are not reproduced in the article but can be found at: http://mtg.upf.edu/node/2023.

using phase based correction is slightly lower, the difference between the two correction methods was not significant. Following these observations, we can conclude that both equal loudness filtering and frequency correction are beneficial for melody extraction.

Finally we consider the difference between the spectral transforms. Interestingly, the MRFFT now results in just a slightly lower frequency error than the STFT. Whilst determining the exact cause is beyond the scope of this study, a possible explanation could be that whilst the overall frequency accuracy for melody spectral peaks is not improved by the MRFFT, the improved estimation at high frequencies is beneficial when we do the harmonic summation (the harmonics are better aligned). Another possible cause is the greater masking of spectral peaks, which could remove non-melody peaks interfering with the summation. When considering the remaining metrics, the STFT gives slightly better results for S_1, whilst there is no statistically significant difference between the transforms for RR_m and S_3. All in all, we see that using a multi-resolution transform provides only a marginal improvement (less than 0.5 cents) in terms of melody frequency accuracy, suggesting it might not necessarily provide significantly better results in a complete melody extraction system.

5.2.2. Salience Function Parameter Configuration

We now turn to evaluate the effect of the parameters of the salience function. In the previous section we saw that equal loudness filtering and frequency correction are important, whilst the type of correction and transform used do not affect the results significantly. Thus, in this section we will focus on configuration 9, which applies equal loudness filtering and uses the STFT transform with phase vocoder frequency correction [7].

In Figure 2 we plot the results obtained for the four metrics using configuration 9 with each of the 64 possible parameter configurations ($\beta = 1$ in all cases) for the salience function. The first 16 datapoints represent configurations where $\alpha = 1$, the next 16 where $\alpha = 0.9$ and so on. Within each group of 16, the first 4 have $N_h = 4$, the next 4 have $N_h = 8$ etc. Finally within each group of 4, each dapatpoint has a different γ value from ∞ down to 20dB.

We first examine the effect of the peak energy threshold γ, by comparing individual datapoints within every group of 4 (e.g. comparing peaks 1-4, 29-32 etc.). We see that (for all metrics) there is no significant difference for the different values of the threshold except for when it is set to 20dB for which the results degrade. That is, unless the filtering is too strict, filtering relatively weak spectral peaks seems to neither improve nor degrade the results.

Next we examine the effect of N_h, by comparing different groups of 4 data points within every group of 16 (e.g. 17-20 vs 25-28). With the exception of the configurations where $\alpha = 1$ (1-16), for all other configurations all metrics are improved the more harmonics we consider. As the melody in our evaluation material is primarily human voice (which tends to have many harmonic partials), this makes sense. We can explain the decrease for configurations 1-16 by the lack of harmonic weighting ($\alpha = 1$) which results in a great number of fake peaks with high salience at integer/sub-integer multiples of the true f0.

Finally, we examine the effect of the harmonic weighting parameter α. Though it has a slight effect on the frequency error, we are primarily interested in its effect on melody salience as indicated by RR_m, S_1 and S_3. For all three metrics, no weighting (i.e. $\alpha = 1$) never produces the best results. For RR_m and S_1 we get best performance when α is between 0.9 and 0.8. Interestingly, S_3 increases continually as we decrease α. This implies that even with weighting, fake peaks at integer/sub-integer multiples (which are strongly affected by α) are present. This means that regardless of the configuration used, systems which use salience functions based on harmonic summation should include a post-processing step to detect and discard octave errors.

[7] Configurations 8, 11 and 12 result in similar graphs.

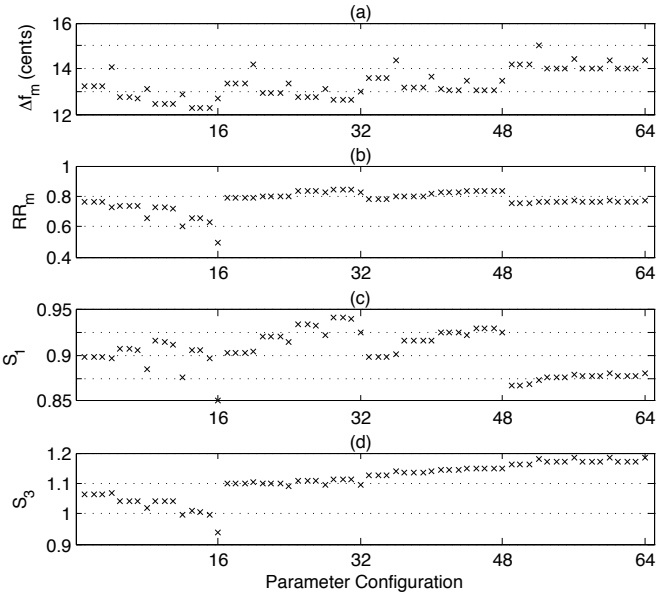

Figure 2: *Salience function design, results by parameter configuration.*

In Figure 3 we plot the metrics as a function of the parameter configuration once more, this time for each genre (using analysis configuration 9). Interestingly, opera, jazz and bossa nova behave quite similarly to each other and to the overall results. For pop/rock however we generally get slightly lower results, and there is greater sensitivity to the parameter values. This is most likely due to the fact that the accompaniment is more predominant in this genre, making it harder for the melody to stand out. In this case we can expect to find more predominant peaks in the salience function which represent background instruments rather than octave errors of the melody. Consequently, S_3 no longer favours the lowest harmonic weighting and, like RR_m and S_1, gives best results for $\alpha = 0.8$ or 0.9.

Following the above analysis, we can identify the combination of salience function parameters that gives the best overall results across all four metrics as $\alpha = 0.8$ or 0.9, $\beta = 1$, $N_h = 20$ and $\gamma = 40$dB or higher.

6. CONCLUSIONS

In this paper the first two steps common to a large group of melody extraction systems were studied - sinusoid extraction and salience function design. Several analysis methods were compared for sinusoid extraction and it was shown that accuracy is improved when frequency/amplitude correction is applied. Two spectral transforms (single and multi-resolution) were compared and shown to perform similarly in terms of melody energy recall and frequency accuracy.

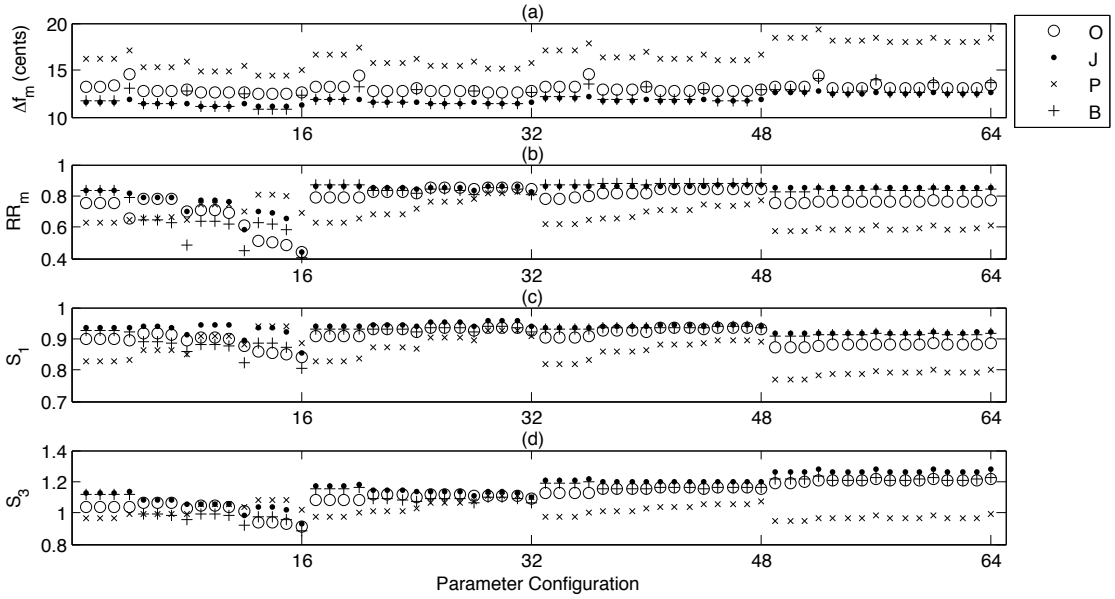

Figure 3: *Per genre results by parameter configuration. Genres are labeled by their first letter – Opera, Jazz, Pop/Rock and Bossa Nova.*

A salience function based on harmonic summation was introduced alongside its key parameters. The different analysis configurations were all evaluated in terms of the salience function they produce, and the effects of the parameters on the salience function were studied. It was shown that equal loudness and frequency correction both result in significant improvements to the salience function, whilst the difference between the alternative frequency correction methods or the single/multi-resolution transforms was marginal. The effect of the different parameters on the salience function was studied and an overall optimal analysis and parameter configuration for melody extraction using the proposed salience function was identified.

7. ACKNOWLEDGMENTS

The authors would like to thank Ricard Marxer, Perfecto Herrera, Joan Serrà and Martín Haro for their comments.

8. REFERENCES

[1] G. E. Poliner, D. P. W. Ellis, F. Ehmann, E. Gómez, S. Steich, and B. Ong, "Melody transcription from music audio: Approaches and evaluation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 4, pp. 1247–1256, 2007.

[2] Jean-Louis Durrieu, Gaël Richard, Bertrand David, and Cédric Févotte, "Source/filter model for unsupervised main melody extraction from polyphonic audio signals," *Trans. Audio, Speech and Lang. Proc.*, vol. 18, pp. 564–575, 2010.

[3] M. Ryynänen and A. Klapuri, "Automatic transcription of melody, bass line, and chords in polyphonic music," *Computer Music Journal*, vol. 32, no. 3, pp. 72–86, 2008.

[4] P. Cancela, "Tracking melody in polyphonic audio," in *4th Music Information Retrieval Evaluation eXchange (MIREX)*, 2008.

[5] Karin Dressler, "Audio melody extraction for mirex 2009," in *5th Music Information Retrieval Evaluation eXchange (MIREX)*, 2009.

[6] J. Salamon and E. Gómez, "Melody extraction from polyphonic music audio," in *6th Music Information Retrieval Evaluation eXchange (MIREX)*, extended abstract, 2010.

[7] 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, pp. 311–329, 2004.

[8] K. Dressler, "Sinusoidal extraction using an efficient implementation of a multi-resolution FFT," in *Proc. of the Int. Conf. on Digital Audio Effects (DAFx-06)*, Montreal, Quebec, Canada, Sept. 2006, pp. 247–252.

[9] P. Cancela, M. Rocamora, and E. López, "An Efficient Multi-Resolution Spectral Transform for Music Analysis," in *Proc. of the 10th Int. Society for Music Information Retrieval Conference (ISMIR)*, Kobe, Japan, 2009, pp. 309–314.

[10] A. P. Klapuri, "Multiple Fundamental Frequency Estimation based on Harmonicity and Spectral Smoothness," in *IEEE Trans. Speech and Audio Processing*, 2003, vol. 11.

[11] D. W. Robinson and R. S. Dadson, "A re-determination of the equal-loudness relations for pure tones," *British Journal of Applied Physics*, vol. 7, pp. 166–181, 1956.

[12] Florian Keiler and Sylvain Marchand, "Survey on extraction of sinusoids in stationary sounds," in *Proc. of the 5th Int. Conf. on Digital Audio Effects (DAFx-02)*, Hamburg, Germany, Sept. 2002, pp. 51–58.

[13] J. L. Flanagan and R. M. Golden, "Phase vocoder," *Bell Systems Technical Journal*, vol. 45, pp. 1493–1509, 1966.

[14] J. Bonada, "Wide-band harmonic sinusoidal modeling," in *Proc. 11th Int. Conf. on Digital Audio Effects (DAFX-08)*, Espoo, Finland, Sept. 2008.