A COMPARISON OF PITCH EXTRACTION METHODOLOGIES FOR DOLPHIN VOCALIZATION

Xanadu C. Halkias and Daniel P. W. Ellis
LabRosa, Columbia University, Department of Electrical Engineering,
1300 S.W. Mudd, 500 West 120th Street, New York, NY, 10027

ABSTRACT

When collecting and analyzing marine mammal vocalizations one of the most important goals is to automatically extract the pitch/fundamental frequency of the collected calls. In dolphins we can assume that there are two main pitched sounds: whistles, which can be described as tonal AM-FM signals, and bursts, which can be described as highly harmonic signals. There are three main difficulties with pitch extraction on dolphin vocalizations that arise from the nature of the data. First, most underwater recordings are restricted to a low signal-to-noise ratio due to reflections, hardware noise and other interferences. This constitutes a big challenge for most existing pitch trackers. Second, one has to take into account the significant differences in the frequency range of bottlenose dolphin vocalizations compared to humans. Finally, dolphin whistles and bursts generally are emitted in two distinct frequency ranges, which result in different modes in the analysis data. In this work we compare our novel pitch extraction approach with two widely popular algorithms. Our approach uses hierarchy-based hidden Markov models (HMM) with cepstral coefficients as features. We quantitatively compare the performance of our algorithm with Yin, which is based on a modified autocorrelation method and get_f0, a popular off-the-shelf pitch tracker that utilizes linear predictive coefficients (LPC) and dynamic programming. Our approach outperforms the comparative methods by at least a factor of 10%.

SOMMAIRE

Pour la collecte et l'analyse de vocalises de mamifères marins, l'extraction de la fondamentale est une étape cruciale. Dans le cas des dauphins, nous pouvons considérer qu'il y a deux types de sons voisés : les chants qui peuvent être décrits comme des tonalités AM-FM, et les rafales ("bursts") constituées de signaux hautement harmoniques. La première des trois difficultés pour extraire le timbre est le très faible rapport signal sur bruit dû aux réflections multiples et autres interférences. La seconde consiste à appréhender les résolutions harmoniques sur le signal de cétacés par rapport aux traitements connus en parole par exemple. Dans ce papier, nous testons notre nouvelle méthode d'extraction de timbre sur un modèle Chaîne de Markov Cachée Hièrarchique à partir de coefficient cepstraux. Nous comparons nos résultats à la méthode YIN basée sur un calcul d'autocorrélation, et à get_f0 qui est extracteur de timbre classique par programmation dynamique utilisant des coefficients LPC. Nous montrons que notre méthode apporte un gain de 10% par rapport à ces méthodes.

1. INTRODUCTION

When analyzing dolphin vocalizations, one of the most important tasks is the extraction of the fundamental frequency/pitch of the desired calls. Several methodologies for attempting to automatically extract pitch exist as scientists are extensively studying this problem, especially with respect to human speech and musical recordings.

Most existing packages used by researchers in the analysis of dolphin vocalizations require manual interaction for the extraction of the desired calls. Moreover, they do not extract the pitch at a per-frame level; rather they provide a frequency range that is manually obtained. These packages, such as Ishmael [1] and Raven [2] are widely used in the field and have been valuable tools for onsite researchers.

In order to resolve the problem of pitch extraction on dolphin vocalizations without manual interaction we utilize methodologies that have been effectively applied in the fields of speech and music processing. One such technique is Hidden Markov Models (HMM) [3, 4]. HMM's can be used either directly on the unprocessed spectrogram of the audio or in combination with the extraction of descriptive features e.g. mel frequency cepstral coefficients. Another class of algorithms that have been used widely in speech processing is based on the auto correlation of the signal or some transformed variation of it.

These different algorithms need domain engineering in order to take into account the intricacies of dolphin recordings as compared to human speech. As discussed in the introduction, there exist two main differentiating issues when analyzing dolphin vocalizations. The first and arguably most important difference is that dolphin recordings exhibit extremely low signal to noise ratios, which require a careful selection of robust features to be used.

Next, almost all information in human speech exists in the range below 4kHz and 20 kHz for music. These ranges could be considered the low register for dolphins that can vocalize above 90kHz. This suggests three problems with using off-the-shelf algorithms designed for speech. Parameters such as filter cutoffs and domain-specific tuning curves must be modified to accommodate the revised frequency range. Also, the wider signal bandwidth of interest necessitates a much higher dimensionality of the feature space. Most importantly, however, is that the frequency range of harmonic content present in a dolphin vocalization can be much higher than the Nyquist rate of most commonly used underwater recording devices. Unfortunately, this means much of the high-frequency content of the dolphin calls is lost during recording.

Finally, there are differences in the frequency ranges of the various types of dolphin vocalizations. It is thus possible to cluster a call based on vocalization type, and using a different classifier for each type. This suggests a hierarchical or two-stage pitch extraction system. Knowing the frequency ranges of each type of call allows us to build classifiers with a lower feature space dimensionality.

In this paper we proceed by providing an explanation of three methodologies used for pitch extraction in section 2. Section 3 summarizes the experimental results, and finally in section 4 we discuss the implications of the nature of dolphin calls on the design of pitch extraction algorithms.

Algorithm	Feature	Classifier
Cepstrum+HHMM	256 Cepstral coefficients	Hierarchy HMM
YIN	Modified autocorrelation	Local minimum
get_f0	LPC residual	Dynamic Programming

Table 1: Description of pitch extraction methodologies

2. PITCH EXTRACTION METHODS

Three algorithms are used in order to achieve a comparative result in the desired pitch extraction task. Our novel approach consists of the use of a hierarchy/decision based HMM with the use of cepstral coefficients. The second algorithm, YIN [6], is widely used in speech processing for single pitch extraction and is based on a modified autocorrelation method. Finally, indicative results from get_f0 [7, 8], a popular off-the-shelf pitch tracker are obtained. Table 1 summarizes the three algorithms and the features used.

2.1 Cepstral coefficients with hierarchically driven hidden Markov models (HMM)

Hidden Markov models (HMM) [3, 4] have been extensively used in many natural sequences such as speech, language and handwriting. They provide us with a valuable tool for the analysis and extraction of information of time dependent data.

As previously discussed, there needs to be a robust selection of features that will be able to overcome the inherent low SNR present in the recordings. In this work the use of the cepstrum is preferred given its ability to highlight the pitch of a given signal. Through the existing literature the cepstrum [5] has been successfully employed in speech to obtain the desired pitch. It assumes a source-filter model and provides a homomorphic deconvolution thus separating the detailed excitation part of the signal from the broad, filter part.

We utilize the real cepstrum using the observation that a pitch peak will appear at the high *n* coefficients. The real cepstrum is defined as the inverse Fourier transform of the log magnitude of the spectrum of our signal. This can be seen in Equation 1.

$$c_n = \int \log|X(e^{j\omega})| \cos n\omega \, d\omega \qquad (1)$$

Figures 2, 3 show an example of the cepstral coefficients for one call.

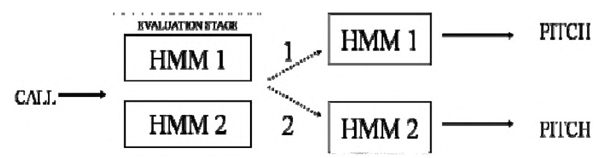

Figure 1: Description of HHMM system

We remove, as is common, the first coefficient, which captures the average energy of the original signal.

Arguably, the use of the spectrogram or even a normalized version of it could have been a suitable feature, it is clear that it will not provide a good noise suppression feature given that noise will be distributed across all frequencies, thus interfering with the task at hand.

Domain engineering suggests the existence of narrowband clusters of calls within the wide frequency range of vocalizations. In this work, analysis of the data as will be seen in section 4.1 dictated the use of a decision level. The idea behind this implementation is that there is an inherent bimodality in the data that can be taken advantage of with the use of a hierarchy. Initially, two HMM's are created with different number of hidden states that correspond to different frequency ranges of the calls. For every input vector both HMM's are evaluated using the forward algorithm and the one that gives us the highest likelihood is activated for the implementation of Viterbi decoding [3, 4], thus obtaining the most likely path across the hidden states.

Figure 1 provides a schematic description of the utilized system. It is evident from Figure 1 that two different HMM's are used each of them with different parameters representing two different frequency ranges respectively.

Each HMM is defined through parameters, λ that are extracted from the data. In this work both HMM's are continuous, implying that each state, q, can be represented with a single Gaussian probability density function:

$$\begin{aligned} &\lambda_1 = (\pi_1, A_1, E_1), \text{where:} \\ &\pi_1 = (\pi_{11}, \pi_{12}, \dots \pi_{1N}),\ N = 1,2,\dots 18 \\ &A_1 = \{a_{1ij}\}_{i,j=1,2,\dots N}, \text{where } a_{1ij} = P(q_{1t} = j \mid q_{1t-1} = i) \\ &E_{1N}(o) = N(o \mid \mu_{1N}, \sigma_{1N}), N = 1,2,\dots 18 \end{aligned} \tag{2}$$

$$\begin{aligned} &\lambda_2 = (\pi_2, A_2, E_2), \text{where:} \\ &\pi_2 = (\pi_{21}, \pi_{22}, \dots \pi_{2M}),\ M = 1,2,\dots 32 \\ &A_2 = \{a_{2ij}\}_{i,j=1,2,\dots M}, \text{where } a_{2ij} = P(q_{2t} = j \mid q_{2t-1} = i) \\ &E_{2M}(o) = N(o \mid \mu_{2M}, \sigma_{2M}), M = 1,2,\dots 32 \end{aligned} \tag{3}$$

where the states sets q_1, q_2 represent the frequency ranges of approximately 2.2kHz-11kHz and 440Hz-740Hz respectively. A noise state is also added for every HMM in order to capture the lack of pitch in a particular frame. Each state, q represents a pitch delay number that can be directly mapped to a specified frequency. Also, π_1, π_2 define the priors for state sets q_1, q_2 respectively as obtained from the statistics of the ground truth data. A_1, A_2 define the transition matrices for each HMM directly obtained from our ground truth and E_1, E_2 are the emission distributions for each state set. These are single 256 dimensional Gaussian distributions obtained from the extracted cepstral coefficients.

Once the parameters of the HMM's have been extracted we proceed to evaluate every call in order to identify its frequency range. This is shown in Equations 4, 5. The last stage of the system employs Viterbi decoding [3, 4] in order to find the most likely path across the evaluated state set, thus extracting the desired pitch at every frame, Equation 6.

$$p(Y_1) = \sum_{q_1} p(Y_1 \mid q_1) p(q_1) \tag{4}$$

$$p(Y_2) = \sum_{q_2} p(Y_2 \mid q_2) p(q_2) \tag{5}$$

$$\begin{aligned} &\text{if } p(Y_1) > p(Y_2) \text{ then} \\ &V = \max_{q_{1 \dots T}} p(q_{11}, q_{12}, \dots q_{1t-1}, q_{1t} = i, Y_1, Y_2 \dots Y_T \mid \lambda_1) \end{aligned} \tag{6}$$

Where $Y_t, t = 1,2$ is a sequence of observations, $q_t, t = 1,2$ is a sequence of the hidden states, $q_t, t = 1,2 \dots T$ is the maximum probability state path and λ_1 are the parameters of the HMM.

2.2 YIN: A fundamental frequency estimator

Yin, created by de Chevigne and Kawahara [6], is a widely used algorithm for the estimation of the fundamental frequency/pitch of speech or monophonic musical sounds. It is based on a modified autocorrelation method and is extremely successful in extracting single pitches. Its popularity is also enhanced by the fact that it is a relatively simple and efficient algorithm, thus minimizing the computational cost.

Since our goal is to extract the fundamental frequency of dolphin vocalizations we can assume that our signal x_t is periodic with period T.

As mentioned previously YIN is based on the autocorrelation of the signal as defined in Equation 7.

$$r_t(\tau) = \sum_{j=t+1}^{t+W} x_j x_{j+\tau} \tag{7}$$

where $r_t(\tau)$ is the autocorrelation at lag τ calculated at time t and W is the integration window size.

We can also see that Equation 6 holds if we take the square and average over a window, W. This implies that a difference function can be formed where an unknown period may be found while searching for those values of τ for which the function is zero. The function is seen in Equation 8.

$$d_t(\tau) = \sum_{j=1}^{W} (x_j - x_{j+\tau})^2 \tag{8}$$

One of the problems that the difference function creates is that it has the value of zero at zero lag and often times a non-zero value at the lag corresponding to the period due to imperfections in the periodicity. This indicates that the method will fail since it will always choose for the zero lag. In order to alleviate the above problem, the method proposes the use of the cumulative mean normalized difference function instead of the one in Equation 8. This new function is shown in Equation 9.

$$d'_t(\tau) = \begin{cases} 1, \text{if } \tau = 0 \\ d_t(\tau) \Big/ \left[(1/\tau) \sum_{j=1}^{\tau} d_t(j) \right], \text{otherwise} \end{cases} \tag{9}$$

This new function is actually one at zero lag and stays large at small lags.

There are several more steps that can be employed in order to ensure a better estimate and these steps can be seen in detail in [5]. Overall, the desired pitch can be obtained by picking the smallest value of the lag/pitch delay, τ that gives the minimum d'. An example of the YIN function for a specific call as well as the cepstral coefficients for the same call is shown in Figures 2-4.

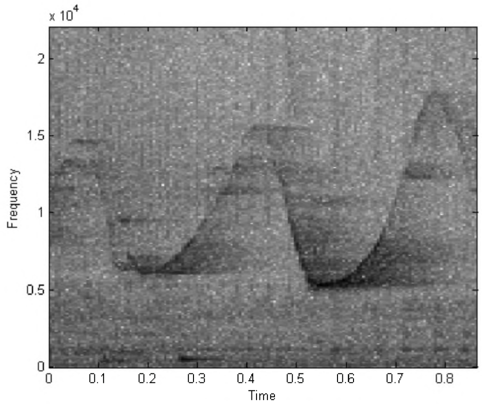

Figure 2: Whistle example

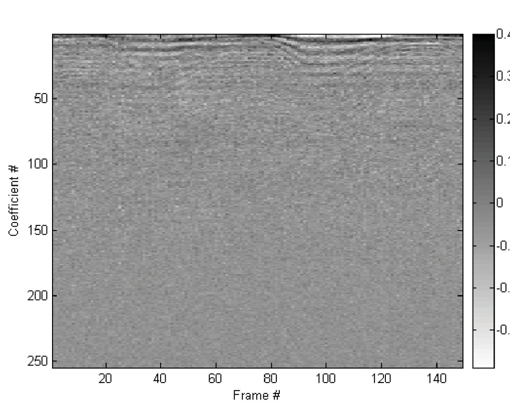

Figure 3: Cepstral coefficients

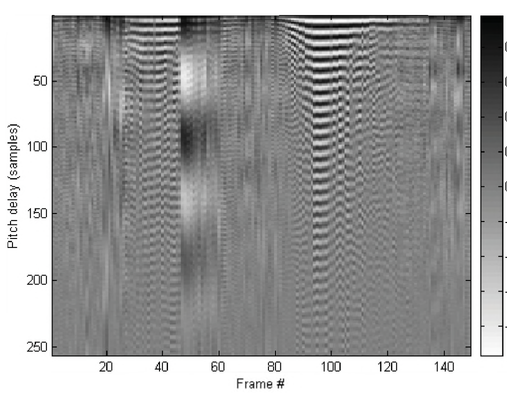

Figure 4: YIN coefficients

There are several more steps that can be employed in order to ensure a better estimate and these steps can be seen in detail in [5]. Overall, the desired pitch can be obtained by picking the smallest value of the lag/pitch delay, τ that gives the minimum d'. An example of the YIN function for a specific call as well as the cepstral coefficients for the same call is shown in Figures 2-4.

2.3 Get_f0: A software package for pitch extraction in speech

Get_f0 is one of the most popular pitch tracking algorithms. It is part of a widely used software package called Entropic Signal Processing Systems (ESPS) and Waves [8]. The majority of researchers in speech processing are familiar with this package. It is based on Doddington's and Secrest's 1983 algorithm [7] for pitch tracking in speech systems.

This method utilizes the linear prediction coding (LPC) residual error signal in order to extract the desired pitch candidate. LPC is based on the source filter model as seen for the cepstrum in section 2.1. This indicates that we theoretically expect that the residual signal will provide us with the excitation information.

To best alleviate some problems of high frequency noise, the authors devise and employ a de-emphasis filter as a pre-processing tool, whereby they low pass filter the residual signal in voiced regions of speech and high pass filter in unvoiced regions. These filters need to be redesigned for dolphin vocalizations.

To extract the candidate pitch at each instance the peaks of the normalized cross-correlation are acquired, Equation 10.

$$C(\tau) = \frac{\sum_{j=0}^{m-1} s(j)s(j+\tau)}{\left(\sum_{j=0}^{m-1} s(j) \sum_{j=0}^{m-1} s(j-\tau)\right)^{1/2}} \qquad (10)$$

Where τ is the lag and m is the number of samples to be correlated. As previously mentioned, the candidate pitch values are the lags at the peaks of $C(k)$ and the "goodness" measure is the corresponding value of $C(k)$ at those lags.

After having extracted the above values, dynamic programming [9] is employed in order to extract a smoother pitch contour. This requires some sort of penalty metric in order to decide what the best path amongst the candidates is. The cumulative penalty for each pitch candidate consists of a transition error in going from one frame to the next. This methodology provides a good pitch extractor specialized for speech.

3. EXPERIMENTAL RESULTS

In this section we provide the comparative experimental results as obtained from the methodologies described in section 2.

It is important to provide information on the data that was used for the experiments. Recordings from captive dolphins were obtained. From these recordings, whistles and bursts were manually extracted so that there would be no overlapping vocalizations. Overall, 110 calls were extracted of balanced type. These calls have a mean duration of 0.5sec and a mean SNR of 9.7dB. The low SNR was partly a result of analog to digital conversion given the lack of high precision hardware at the time of the recordings. The SNR was obtained by averaging the peak SNR, Equation 10, at every frame, which was computed through the short-time autocorrelation function.

$$pSNR = dB\left(\frac{r(\tau=p)}{r(0)-r(\tau=p)}\right) \quad (11)$$

Where $r(0)$ is the energy of the signal plus the noise and $r(\tau=p)$ is the energy of the signal with period at lag $\tau=p$.

To be better able to extract meaningful conclusions, ground truth was obtained by bootstrapping (semi-hand labeling for every frame). Initially, YIN was employed and then visually inspected in order to correct possible errors. Evidently, the extraction of ground truth allows for some errors due to resolution and rounding limitations given that we extract a pitch delay/lag for every frame. It is expected that such ground truth may incorporate some bias in the final results.

After obtaining the ground truth, the analysis of the data indicated an inherent bimodality. That led us to the choice of the hierarchy driven hidden Markov models for our approach. This is clearly shown in Figure 5. Two distinct frequency ranges are evident, thus allowing us to insert a decision level in the system. Arguably, one might explore the reasons for not choosing a single dynamic model/HMM for this task. In several experiments, a single system suffered from erroneous "doublings" and/or "halvings" at a per frame level caused by the fact that the cepstrum captures the presence of noise e.g. hardware, reflection noise.

Table 2 provides the average per frame accuracy for all three methods. It is worth noting that there are three different metrics in our results: Strict rate, which implies that the resulting pitch is an exact match with the ground truth, relaxed rate of ±1 pitch delay (lag), and finally a relaxed rate of ±2 pitch delays (lag). Basically, this implies a soft boundary or range of acceptable error. The relaxed rates correspond to an approximate 1.5% and 3% deviation from the ground truth, which in many applications could be acceptable. The same results are provided schematically in Figure 6. All results are generated using leave one out cross validation, otherwise known as round-robin.

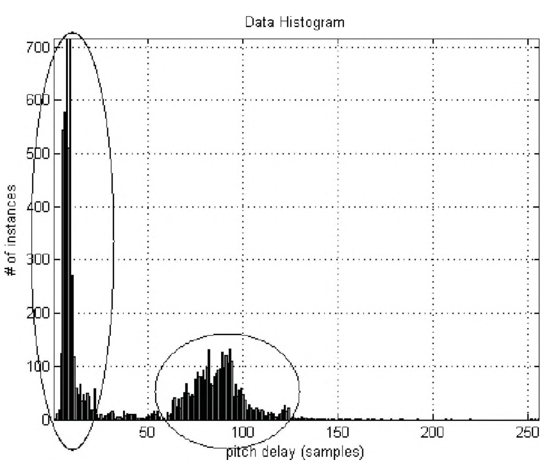

Figure 5: Data histogram. The two ellipses show the two modes of the data

Average per frame accuracy (%)		
HMM cepstrum	Yin	get f0
Strict Rate (%)		
66.12	47.09	29.3
Relaxed Rate ±1 pitch delay (%)		
76.01	54.35	N/A
Relaxed Rate ±2 pitch delay (%)		
77.9	55.11	N/A

Table 2: Comparative results

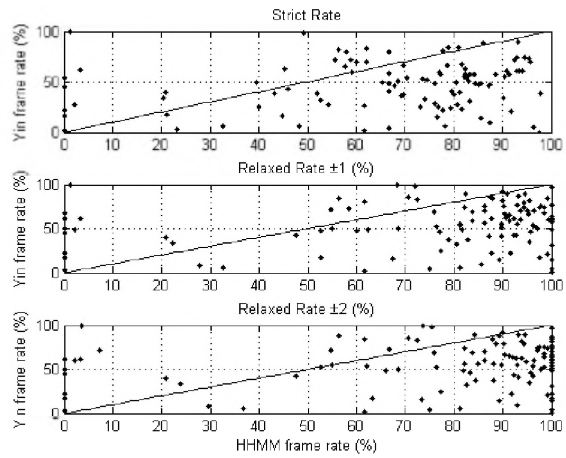

Figure 6: Comparative results of the YIN frame rate vs. the HHMM frame rate for every call

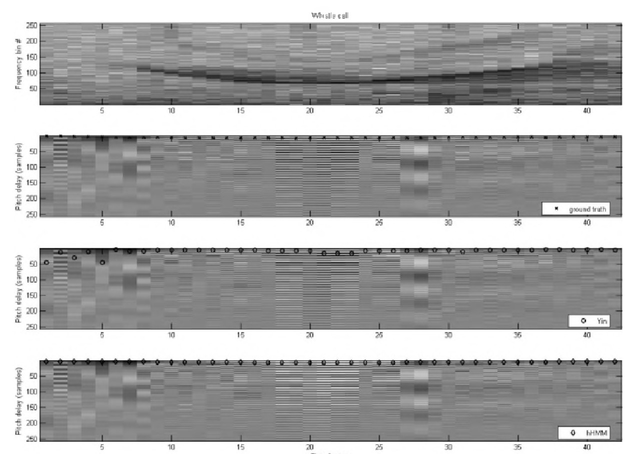

Figure 7: Example of successful pitch extraction using hierarchy HMM

In all cases it is apparent that our novel approach is superior to the baseline algorithms by over 10%. It is also interesting to note that get_f0 fails to give us comparable results for the relaxed rates due to the fact that it is highly tuned for human speech and is not able to track the desired pitch in dolphin vocalizations, which exhibit a much wider frequency range.

Furthermore, Figure 6 provides comparative results for each call for the novel approach of the hierarchy driven HMMs with the cepstral coefficients and the YIN algorithm. As it is clearly seen in the figures there is a shift of the points towards the right side of the plots. This indicates that our methodology is superior and has a higher percentage of calls that are achieving above 80% frame accuracy.

In addition, an interesting fact arises from these plots. There appear to be a constant number of calls that are giving us a near 0% percent match. This discrepancy is caused due to the error that is introduced by the hierarchy. Basically, for these calls the decision of which frequency range they belong to is false, and thus the pitch extraction fails completely.

Lastly, Figures 7, 8 provide indicative examples of success and failure of the implemented algorithms. In both figures, the original spectrogram is shown and the comparative results are overlaid on the short time autocorrelation in order to provide a good visualization tool with results extracted from YIN.

In Figure 7, our implementation is closer to the ground truth where YIN actually exhibits a number of errors due to noise interference that leads to wrong peak picking.

In Figure 8, both YIN and the hierarchy driven HMM fail to extract the desired pitch again, due to the extremely low SNR as well as the ambiguity of the type and range of the call. In this case the evaluation stage of our system fails to classify this call in the correct range.

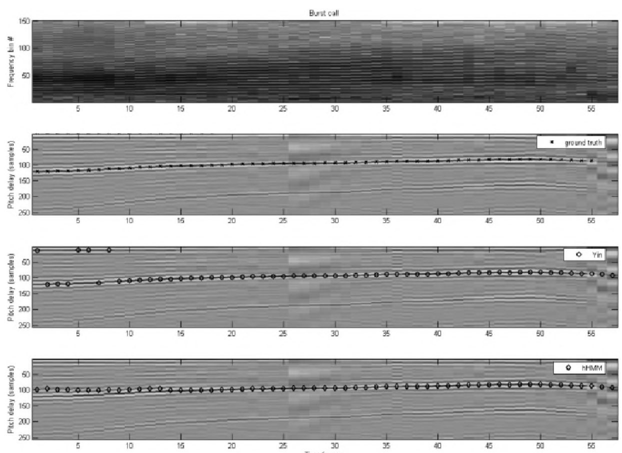

Figure 8: Example of fair pitch extraction using hierarchy HMM

4. CONCLUSIONS

As described in the previous sections, this work provides a comparative view on the success of three different pitch extraction algorithms for dolphin vocalizations. As evident from the experimental results presented in section 3, our novel approach of using hierarchy driven hidden Markov models with cepstral coefficients outperforms the other two popular methods in speech and music, YIN and get_f0.

The success of our approach is based on the idea of the hierarchy, which was implied from the nature of our data as seen in Figure 5. The existing bimodality allowed us to create two different HMM's with two different sets of states. This immediately reduced the state space dimensionality of our system, thus minimizing the computational cost, while alleviating problems when training our model.

It is worth noting that the bimodality in the data needs to be explored in a larger body of data to extract meaningful conclusions in terms of the generic aspect of our method. Our data set needs to be enhanced so that we can extract possible biases from these specific recordings. Moreover, it would be interesting to compare the differences between recordings of captive dolphins versus dolphins in the wild.

Another reason for a larger labeled data set is to avoid pitfalls of over fitting when resorting to training testing methods such as leave one out cross validation.

Also, it should be noted that the hierarchy introduces an extra error term when it comes to deciding which range the call belongs to. Overall, this error accounts for only 4% of the total calls.

The use of the cepstral coefficients as a feature is considered a good match given that it showed a better descriptive feature than using the magnitude of the spectrum.

Also, it appears to be more resilient to noise. Furthermore, we can make assumptions about the location of the pitch peak, thus eliminating a number of coefficients and reducing the dimensionality of the feature space. This could also lead to a more computationally efficient algorithm.

Lastly, it is worth noting that YIN was far superior to get_f0. Its simplicity and efficiency make it a good candidate in simple cases. However, YIN is not so resilient to an increased noise level present in the recordings. Interestingly though, both YIN and our approach utilize dynamic programming/Viterbi, which could be an advantage to get_f0.

Overall, there are several steps that can be taken to improve the algorithm presented in this work. In summary, this paper clearly shows that a choice of good features and the use of a classifier which can be tuned according to a given data set can provide us with very satisfactory results for the task of pitch extraction.

5. ACKNOWLEDGMENTS

The authors would like to thank Diana Reiss who provided us with the recordings of captive dolphins from her own collection. These recordings were hand-labeled by the authors and used for the experiments described in this work. Also, many thanks to Hervé Glotin and Olivier Adam for the French translation of the abstract.

6. REFERENCES

1. ISHMAEL, Integrated System for Holistic Multi-channel Acoustic Exploration and Localization, D. K. Mellinger, U.S. Department of Commerce, at http://pmel.noaa.gov/vents/acoustics/whales/ishmael/

2. RAVEN, Interactive sound analysis software, Cornell Lab of Ornithology at: http://www.birds.cornell.edu/brp/raven/Raven.html
3. R. O. Duda, P. E. Hart and D. G. Stork, *Pattern Classification*, John Wiley & Sons, Inc., second edition 2001
4. L. R., Rabiner and B. H. Juang, "An Introduction to Hidden Markov models", IEEE ASSP Magazine, January, pp. 4-15, 1986
5. A.V. Oppenheim and R. W. Schafer, *Digital Signal Processing*, Englewood Cliffs, NJ, Prentice-Hall, 1975
6. A. de Chevigne and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music", Journal of the Acoustical Society of America, 111 (4), April 2002
7. B. Secrest and G. Doddington, "An integrated pitch tracking algorithm for speech systems", Acoustics and Speech, and Signal Processing, IEEE International conference on ICASSP'83, vol. 8, April 1983, pp. 1352-1355.
8. ESPS/WAVES, Entropic Signal Processing Systems, software package at: http://www.speech.kth.se/software/
9. B. Gold and N. Morgan, *Speech and Audio Signal Processing: Processing and Perception of Speech and Music*. John Wiley & Sons, Inc., New York, 1999.