DETECTION OF PITCHED/UNPITCHED SOUND USING PITCH STRENGTH CLUSTERING

Arturo Camacho
Computer and Information Science and Engineering Department
University of Florida
Gainesville, FL 32611, USA
acamacho@cise.ufl.edu

ABSTRACT

A method for detecting pitched/unpitched sound is presented. The method tracks the pitch strength trace of the signal, determining clusters of pitch and unpitched sound. The criterion used to determine the clusters is the local maximization of the distance between the centroids. The method makes no assumption about the data except that the pitched and unpitched clusters have different centroids. This allows the method to dispense with free parameters. The method is shown to be more reliable than using fixed thresholds when the SNR is unknown.

1. INTRODUCTION

Pitch is a perceptual phenomenon that allows ordering sounds in a musical scale. However, not all sounds have pitch. When we speak or sing, some sounds produce a strong pitch sensation (e.g., vowels), but some do not (e.g., most consonants). This classification of sounds into pitched and unpitched is useful in applications like music transcription, query by humming, and speech coding.

Most of the previous research on pitched/unpitched (P/U) sound detection has focused on speech. In this context, the problem is usually referred as the voiced/unvoiced (V/U) detection problem, since voiced speech elicits pitch, but unvoiced speech does not. Some of the methods that have attempted to solve this problem are pitch estimators that, as an aside, make V/U decisions based on the degree of periodicity of the signal [3,7,8,11][1]. Some other methods have been designed specifically to solve the V/U problem, using statistical inference on the training data [1,2,10]. Most methods use static rules (fixed thresholds) to make the V/U decision, ignoring possible variations in the noise level. To the best of our knowledge, the only method deals with non-stationary noise makes strong assumptions about the distribution of V/U sounds[2], and requires the determination of a large number of parameters for those distributions [5].

The method presented here aims to solve the P/U problem using a dynamic two-means clustering of the pitch strength trace. The method favors temporal locality of the data, and adaptively determines the clusters' centroids by maximizing the distance between them. The method does not make any assumption about the distribution of the classes except that the centroids are different. A convenient property of the method is that it dispenses with free parameters.

2. METHOD

2.1. Formulation

A reasonable measure for doing P/U detection is the pitch strength of the signal. We estimate pitch strength using the SWIPE′ algorithm [4], which estimates the pitch strength at (discrete) time n as the spectral similarity between the signal (in the proximity of n) and a sawtooth waveform with missing non-prime harmonics and same (estimated) pitch as the signal.

In the ideal scenario in which the noise is stationary and the pitch strength of the non-silent regions of the signal is constant, the pitch strength trace of the signal looks like the one shown in Figure 1(a). Real scenarios differ from the ideal in at least four aspects: (i) the transitions between pitched and non-pitched regions are smooth; (ii) different pitched utterances have different pitch strength; (iii) different unpitched utterances have different pitch strength; and (iv) pitch strength within an utterance varies over time. All these aspects are exemplified in the pitch strength trace shown in Figure 1(b).

The first aspect poses an extra problem which is the necessity of adding to the model a third class representing transitory regions. Adding this extra class adds significant complexity to the model, which we rather avoid and

[1] Pitch strength and degree of periodicity of the signal are highly correlated.

[2] It assumes that the autocorrelation function at the lag corresponding to the pitch period is a stochastic variable whose p.d.f. follows a normal distribution for unvoiced speech, and a reflected and translated chi-square distribution for voiced speech.

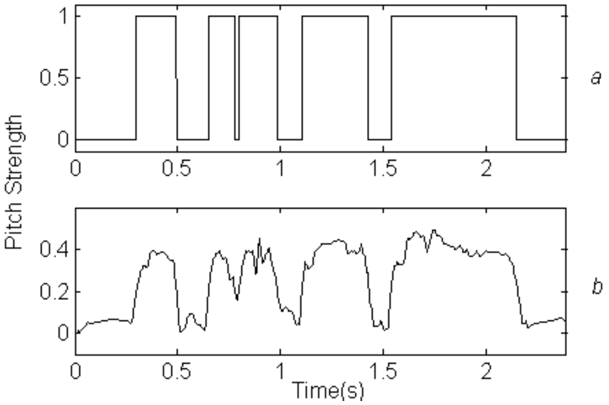

Figure 1. Pitch strength traces. (a) Ideal. (b) Real.

instead opt for assigning samples in the transitory region to the class whose centroid is closest. The second and third aspects make the selection of a threshold to separate the classes non trivial. The fourth aspect makes this selection even harder, since an utterance whose pitch strength is close to the threshold may oscillate between the two classes, which for some applications may be even worst than assigning the whole utterance to the wrong class.

Our approach for solving the P/U detection problem is the following. At every instant of time *n*, we determine the optimal assignment of classes (P/U) to samples in the neighborhood of *n*, using as optimization criterion the maximization of the distance between the centroids of the classes. Then, we label *n* with the class whose pitch-strength centroid is closer to the pitch strength at time *n*.

To determine the optimal class assignment for each sample *n'* in the neighborhood of *n*, we first weight the samples using a Hann window of size 2*N*+1 centered at *n*:

$$w_N(n'-n) = 1 + \cos\left(\frac{\pi(n'-n)}{N+1}\right), \qquad (1)$$

for $|n'-n| \leq N$, and 0 otherwise.

We represent an assignment of classes to samples by the membership function $\mu(n') \in \{0,1\}$ $(\forall n')$, where $\mu(n') = 1$ means that the signal at *n'* is pitched $(\forall n')$, and $\mu(n') = 0$ means that the signal at *n'* is unpitched $(\forall n')$. Given an arbitrary assignment μ of classes to samples, an arbitrary *N*, and a pitch strength time series *s*(*n'*), we determine the centroid of the pitched class in the neighborhood of *n* as

$$c_1(n,\mu,N) = \frac{\sum_{n'=-N}^{N} \mu(n'+n) s(n'+n) w_N(n')}{\sum_{n'=-N}^{N} \mu(n'+n) w_N(n')}, \qquad (2)$$

the centroid of the unpitched class as

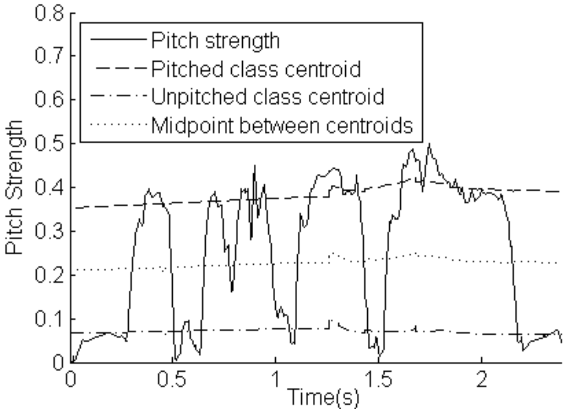

Figure 2. Pitch and unpitched classes centroids and their midpoint.

$$c_0(n,\mu,N) = \frac{\sum_{n'=-N}^{N} \left[1-\mu(n'+n)\right] s(n'+n) w_N(n')}{\sum_{n'=-N}^{N} \left[1-\mu(n'+n)\right] w_N(n')}, \qquad (3)$$

and the optimal membership function and parameter *N* as

$$[\mu^*(n), N^*(n)] = \underset{[\mu,N]}{\arg\max}\, c_1(n,\mu,N) - c_0(n,\mu,N) \cdot \qquad (4)$$

Finally, we determine the class membership of the signal at time *n* as

$$m(n) = \left[\frac{s(n) - c_0(n,\mu^*(n),N^*(n))}{c_1(n,\mu^*(n),N^*(n)) - c_0(n,\mu^*(n),N^*(n))} > 0.5\right], \qquad (6)$$

where [·] is the Iverson bracket (i.e., it produces a value of one if the bracketed proposition is true, and zero otherwise).

Figure 2 illustrates how the classes' centroids and their midpoint vary over time for the pitch strength trace in Figure 1(b). Note that the centroid of the pitched class follows the tendency to increase over time that the overall pitch strength of the pitched sounds have in this trace. Note also that the speech is highly voiced between 0.7 and 1.4 sec (although with a gap at 1.1 sec). This makes the overall pitch strength increase in this region, which is reflected by a slight increase in the centroid of both classes in that region. The classification output for this pitch strength trace is the same as the one shown in Figure 1(a), which consists of a binary approximation of the original pitch strength trace.

2.2. Implementation

For the algorithm to be of practical use, the domains of *N* and μ in Equation 4 need to be restricted to small sets. In our implementation, we define the domain of *N* recursively, starting at a value of 1 and geometrically increasing its value by a factor of $2^{1/4}$, until the size of the pitch strength trace is reached. Non-integer values of *N* are rounded to the closest integer.

The search of μ^* is performed using the Loyd's algorithm (a.k.a. *k*-means) [6]. Although the goal of Loyd's algorithm is to minimize the variance within the classes, in practice it tends to produce iterative increments in the distance between the centroids of the classes as well, which is our goal. We initialize the pitched class centroid to the maximum pitch strength observed in the window, and the unpitched class centroid to the minimum pitch strength observed in the window. We stop the algorithm when μ reaches a fixed point (i.e., when it stops changing) or after 100 iterations. Typically, the former condition is reached first.

2.3. Postprocessing

When the pitch strength is close to the middle point between the centroids, undesired switchings between classes may occur. A situation that we consider unacceptable is the adjacency of a pitched segment to an unpitched segment such that the pitch strength of the pitched segment is completely below the pitch strength of the unpitched segment (i.e., the maximum pitch strength of the pitched segment is less than the minimum pitch strength of the unpitched segment). This situation can be corrected by relabeling one of the segments with the label of the other. For this purpose, we track the membership function $m(n)$ from left to right (i.e., by increasing n) and whenever we find the aforementioned situation, we relabel the segment to the left with the label of the segment to the right.

3. EVALUATION

3.1. Data Sets

Two speech databases were used to test the algorithm: Paul Bagshaw's Database (PBD) (available online at http://www.cstr.ed.ac.uk/research/projects/fda) and Keele Pitch Database (KPD) [9], each of them containing about 8 minutes of speech. PBD contains speech produced by one female and one male, and KPD contains speech produced by five females and five males. Laryngograph data was recorded simultaneously with speech and was used by the creators of the databases to produce fundamental frequency estimates. They also identified regions where the fundamental frequency was inexistent. We regard the existence of fundamental frequency equivalent to the existence of pitch, and use their data as ground truth for our experiments.

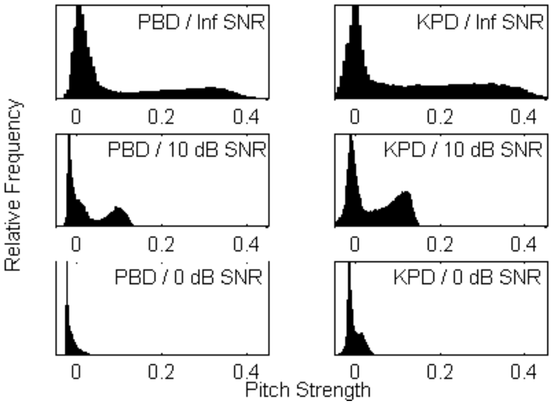

Figure 3. Pitch strength histogram for each database/SNR combination.

3.2. Experiment Description

We tested our method against an alternative method on the two databases described above. The alternative method consisted in using a fixed threshold, which is commonly used in the literature [3,7,8,11]. Six different pitch strength thresholds were explored: 0, 0.01, 0.02, 0.05, 0.10, and 0.20., based on the plots of Figure 3. This figure shows pitch strength histograms for each of the speech databases, at three different SNR levels: ∞, 10, and 0 dB.

3.3. Results

Table 1 shows the error rates obtained using our method (dynamic threshold) and the alternative methods (fixed thresholds) on the PBD database, for seven different SNRs and the six proposed thresholds. Table 2 shows the error rates obtained on the KPD database. On average, our method performed best in both databases (although for some SNRs some of the alternative methods outperformed our method, they failed to do so at other SNRs, producing overall a larger error when averaged over all SNRs). These results show that our method is more robust to changes in SNR.

The right-most column of Tables 1 and 2 shows the (one-tail) *p*-values associated to the difference in the average error rate between our method and each of the alternative methods. Some of these *p*-values are not particularly high compared to the standard significance levels used in the literature (0.05 or 0.01). However, it should be noted that these average error rates are based on 7 samples, which is a small number compared to the number of samples typically used in statistical analyses.

To increase the significance level of our results we combined the data of Tables 1 and 2 to obtain a total of 14 samples per method. The average error rates and their associated *p*-values are shown in Table 3. By using this

	Error rate (%)								
Threshold \ SNR (dB)	0	3	6	10	15	20	∞	Average	*P*-value
0	41.0	11.0	7.4	8.7	13.0	16.0	33.0	18.6	0.10
0.01	51.0	17.0	7.7	7.4	10.0	12.0	23.0	18.3	0.14
0.02	56.0	30.0	9.6	6.9	8.1	9.4	15.0	19.3	0.14
0.05	58.0	57.0	30.0	8.9	6.5	6.6	7.6	24.9	0.09
0.10	58.0	58.0	58.0	39.0	10.0	7.5	5.7	33.7	0.03
0.20	58.0	58.0	58.0	58.0	57.0	36.0	14.0	48.4	0.00
Dynamic	24.0	13.0	9.3	7.7	7.2	7.2	8.4	11.0	

Table 1. Error rates on Paul Bagshaw's Database

	Error rate (%)								
Threshold \ SNR (dB)	0	3	6	10	15	20	∞	Average	*P*-value
0	20.0	12.0	13.0	11.0	23.0	26.0	26.0	18.7	0.04
0.01	29.0	13.0	10.0	12.0	15.0	17.0	17.0	16.1	0.13
0.02	40.0	18.0	11.0	10.0	11.0	12.0	12.0	16.3	0.23
0.05	50.0	43.0	20.0	11.0	8.7	8.6	8.7	21.4	0.13
0.10	50.0	50.0	50.0	28.0	13.1	11.0	9.6	30.2	0.03
0.20	50.0	50.0	50.0	50.0	47.0	32.0	19.0	42.6	0.00
Dynamic	21.0	15.0	12.0	10.0	10.0	10.0	12.0	12.9	

Table 2. Error rates on Keele Pitch Database

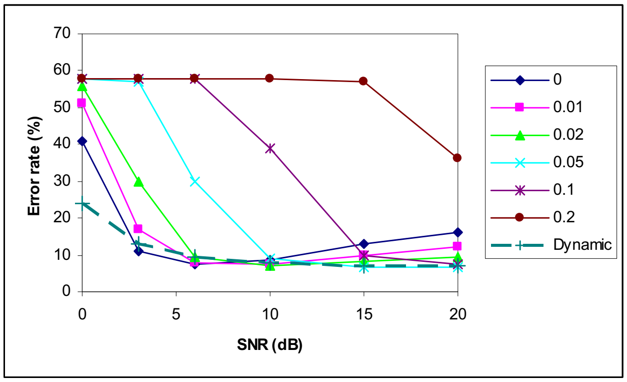

Figure 4. Error rates on Paul Bagshaw's Database

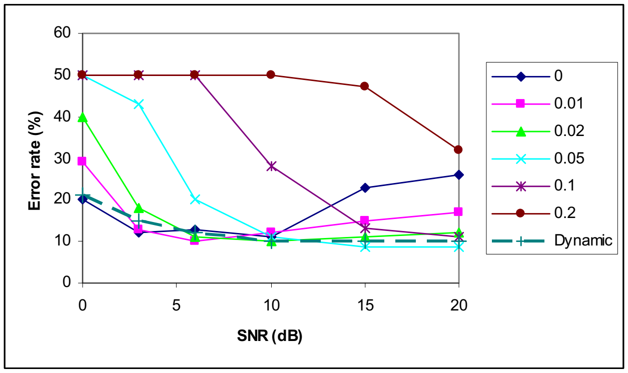

Figure 5. Error rates on Keele Pitch Database

Threshold	Average error rate	*P*-value
0	18.7	0.02
0.01	17.2	0.06
0.02	17.8	0.08
0.05	23.2	0.03
0.10	32.0	0.00
0.20	45.5	0.00
Dynamic	11.9	

Table 3. Average error rates using both databases (PBD and KPD)

Threshold	Average error rate	*P*-value
0	15.6	0.00
0.01	14.6	0.05
0.02	15.3	0.05
0.05	21.5	0.00
0.10	33.1	0.00
0.20	50.7	0.00
Dynamic	11.1	

Table 4. Average interpolated error rates using both databases (PBD and KPD)

approach, the p-values were reduced by at least a factor of two with respect to the smallest p-value when the databases were considered individually.

Another alternative to increase the significance of our results is to compute the error rates for a larger number of SNRs. However, the high computational complexity of computing the pitch strength traces and the P/U centroids for a large variety of SNR makes this approach unfeasible. Fortunately, there is an easier approach which consists in utilizing the already computed error rates to interpolate the error rates for other SNR levels. Figures 4 and 5 show curves based on the error rates of Tables 1 and 2 (the error rate curve of our dynamic threshold method is the thick dashed curve). These curves are relatively predictable: each of them starts with a plateau, then the error decrease abruptly to a valley, and finally has a slow increase at the end. This suggests that error levels can be approximated using interpolation.

We used linear interpolation to estimate the error rates for SNRs between 0 dB and 20 dB, using steps of 1 dB, for a total number of 21 steps. Then, we compiled the estimated errors of each database to obtain a total of 42 error rates per method. The average of these error rates and the *p*-values associated to the difference between the average error rate of our method and the alternative methods are shown in Table 4. Based on these *p*-values, all differences are significant at the 0.05 level.

4. CONCLUSION

We presented an algorithm for pitched/unpitched sound detection. The algorithm works by tracking the pitch strength trace of the signal, searching for clusters of pitch and unpitched sound. One valuable property of the method is that it does not make any assumption about the data, other than having different mean pitch strength for the pitched and unpitched clusters, which allows the method to dispense with free parameters. The method was shown to produce better results than the use of fixed thresholds when the SNR is unknown.

5. REFERENCES

[1] Atal, B., Rabiner, L. “A pattern recognition approach to voiced/unvoiced/silence classification with applications to speech recognition”. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24(3), 201-212, June 1976.

[2] Bendiksen, A., Steiglitz, K. “Neural networks for voiced/unvoiced speech classification”, *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Albuquerque, New Mexico, USA, 1990.

[3] Boersma, P. “Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound”, *Proceedings of the Institute of Phonetic Sciences* 17: 97–110. University of Amsterdam.

[4] Camacho, A. “SWIPE: A Sawtooth Waveform Inspired Pitch Estimator for Speech and Music”. Doctoral dissertation, University of Florida, 2007.

[5] Kobatake, H. “Optimization of voiced/Unvoiced decisions in nonstationary noise environments”, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(1), 9-18, Jan 1987.

[6] Lloyd, S. “Least squares quantization in PCM”, *IEEE Transactions on Information Theory*, 28(2), 129-137, Mar 1982.

[7] Markel, J. “The SIFT algorithm for fundamental frequency estimation”, *IEEE Transactions on Audio and Electroacoustics*, 5, 367-377, Dec 1972.

[8] Noll, A. M. “Cepstrum pitch determination”, *Journal of the Acoustical Society of America*, 41, 293-309.

[9] Plante, F., Meyer, G., Ainsworth, W.A. “A pitch extraction reference database”, Proceedings of EUROSPEECH 95, 1995, 837-840.

[10] Siegel, L. J. “A procedure for using pattern classification techniques to obtain a voiced/unvoiced classifier”, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(1), 83-89, Feb 1979.

[11] Van Immerseel, L. M., Martens, J. P. “Pitch and voiced/unvoiced determination with an auditory model”, *Journal of the Acoustical Society of America*, 91, 3511-3526.