ROBUST PITCH ESTIMATION USING AN EVENT BASED ADAPTIVE GAUSSIAN DERIVATIVE FILTER

Amol Shah

Electrical Engineering
Stanford University
Stanford, California 94305
amolshah@stanford.edu

Ravi P. Ramachandran

Electrical and
Computer Engineering
Rowan University
Glassboro, NJ 08028
ravi@rowan.edu

Michael A. Lewis

Electrical and
Computer Engineering
City University of New York
New York, NY 10031

ABSTRACT

In the development of practical speech processing algorithms, the ability to automatically and accurately determine the pitch period in noisy environments remains a fundamental obstacle. In this paper, we propose a new pitch detection algorithm based on an iterative adaptive smoothing approach using a Gaussian Derivative filter which is the sum of a zeroth and second order Hermite function. We refer to this new algorithm as the Adaptive Gaussian Derivative Filter (AGDF). The AGDF pitch detector works under varying noise conditions, with variable pitch periods and for different speakers. We compare the performance of the AGDF method to the approach based on the Dyadic Wavelet Transform (DyWT) and the pitch prediction (PP) formulation for speech subjected to different noise conditions and signal to noise ratios (SNR). The results show that the AGDF is slightly better than the DyWT pitch detection scheme and significantly outperforms the PP approach.

1. INTRODUCTION

Pitch is a very important parameter in the analysis and synthesis of speech. Accurate and robust pitch determination is difficult especially when speech is subjected to noise [1]. The difficulty arises from the irregular and variable nature of the speech signal and random noise interference. The human vocal tract varies tremendously from person to person. In fact, the pitch period of humans can vary from 2.5 ms to 25 ms. Also, the pitch period can vary depending on the emotional state, accents and other perceptual variables [2]. In telephony, the pitch period of the signal can be affected by noise, phase distortion or bandwidth reduction of the signal. Therefore, developing an algorithm that can perform well for different speakers of diverse ethnic backgrounds, for different applications and under different environmental conditions is greatly needed. Pitch period estimation is also an excellent pre-processing block for speech enhancement systems using comb filtering [3]. An accurate pitch estimate leads to an accurate comb filter and successful removal of the noise in the speech signal. This enhancement system can in turn be used to make speaker recognition systems more robust [3]. In this paper, we introduce the Adaptive Gaussian Derivative Filter (AGDF) which adaptively smooths the signal and determines the pitch period under severe noise conditions (low signal to noise ratios (SNR)) and various noise conditions (white, colored and babble). As demonstrated later, the performance with our approach is slightly better than the pitch detection method based on dyadic wavelets (DyWT) [4][5] and significantly better than the pitch prediction (PP) formulation [6].

2. AGDF ALGORITHM

The Gaussian Derivative Filter (GDF) [7] was shown to give better edge enhancement and noise suppression in digital images. Edge enhancement in images is an event based detection that is analogous to enhancement and detection of periodic peaks in the speech signal. This is what motivated us to use GDF's for pitch (event) detection. The GDF $g(x)$ is the linear combination of Hermite functions $h_0(x)$ and $h_2(x)$ as described by

$$g(x) = c_0 h_0(x) + c_2 h_2(x) \tag{1}$$

where

$$h_n(x) = \frac{1}{\sqrt{n!\,2^n}} \frac{d^n}{d(x/\sigma)^n} \frac{1}{\sigma\sqrt{\pi}} \exp(-x^2/\sigma^2) \tag{2}$$

The coefficients c_0 and c_2 weight the two Hermite functions and σ is a scaling parameter.

Since the pitch period varies over different temporal regions of the speech signal, we must divide the signal into frames and adaptively set the GDF parameters to extract the period. Hence, we have the adaptive GDF or AGDF. Within a speech frame, the AGDF parameters are iteratively computed and the filter is applied to the signal. The pitch period is determined from the filtered or smoothed signal. The general idea behind adaptive smoothing is to apply a versatile operator which adapts itself to the local topography of the signal to be smoothed. This principle has been adopted in the AGDF algorithm to smooth the speech in order to enhance the pitch peaks for ease in detection.

In order to develop an adaptive method to find the AGDF parameters c_0, c_2 and σ, we first set $c_0 + c_2 = 1$ without loss of generality (note that c_0 and c_2 are not constrained to be either positive or negative). The factor σ has a greater influence on the spatial and frequency bandwidth of $g(x)$ and hence, is chosen for adaptive optimization. We empirically found that setting $c_0 = 0.65$ gave the best results particularly when speech is subjected to noise. In implementing the AGDF algorithm, the speech is partitioned into 30 ms frames and a set of pitch periods are found for each frame. Successive frames have a 20 ms overlap thereby making each 30 ms frame uniquely specify a 10 ms segment that is centered about the frame. As described in more detail later, there is one pitch period

0-7803-7448-7/02/$17.00 ©2002 IEEE

estimate for each 10 ms segment that is derived from the estimates of the 30 ms frame the segment comes from and from the estimates derived for the two neighboring frames. A finite impulse response (FIR) discrete GDF, namely, $g(n)$ is obtained by finely sampling $g(x)$ from $x = -20$ to $x = 20$ in steps of 0.1. The discrete GDF is a linear phase FIR filter.

We now describe the AGDF algorithm for one 30 ms frame.

1. For the GDF function $g(x)$, we set $c_0 = 0.65$ and the initial condition for the parameter σ is set to be $\sigma = 0.5$. Let $niter$ be the number of iterations. The initial value of $niter$ is $niter = 0$.
2. The parameter $niter$ is incremented by 1. If $niter = 20$, we stop the algorithm and use the most recent pitch estimates as the final estimates. The discrete GDF function $g(n)$ is convolved with the speech frame to get the signal $r(n)$.
3. The absolute maximum value of $r(n)$ is determined and denoted as $rmax$. A set of estimated pitch periods of $r(n)$ are taken by (1) picking the local peaks (or maxima) of $r(n)$ for which $|r(n)| > 0.7\ rmax$, (2) finding the corresponding time indices of these local peaks and (3) finding the difference in the time indices between successive local peaks. For example, if two successive peaks of r(n) are at $n = 10$ and $n = 75$, the estimated pitch period is 65 samples.
4. Pitch period estimates below 2.5 ms and above 25 ms are discarded. There are two peaks that will have led to the discarded estimate. Of these two peaks, the pitch peak that has a lower absolute signal amplitude is ignored. The pitch periods are recalculated.
5. From the set of pitch period estimates, an average pitch period and a standard deviation are calculated. The standard deviation reflects how much the pitch period varies in a particular frame. For clean speech, we can expect a maximum variation of about 10% above and below the average pitch period. For noisy speech, the variation is much more and in this case, we have to repeatedly apply the GDF with an updated σ to get a reliable estimate. If the standard deviation is above 20% of the average pitch period, the GDF function is recomputed using the new updated value $\sigma + \epsilon$ where ϵ is a small number (usually around 0.1) and we go back to Step 2. If the standard deviation is below 20% of the average pitch period, the pitch periods are approximately equally spaced indicating that the pitch period estimation is good and the algorithm terminates. Note that if only one pitch period estimate is found, no standard deviation is calculated and the algorithm terminates.
6. Upon termination of the algorithm, a set of pitch period estimates are recorded. If no estimates are found, the pitch period for this frame is made equal to 0.

Candidate pitch period estimates for each 30 ms frame are found. A final postprocessing step is executed as described below. As mentioned earlier, the 20 ms overlap between successive frames specifies a 10 ms segment that is centered about a particular frame. We get one pitch period estimate for each 10 ms segment by first gathering the pitch period estimates of the 30 ms frame the segment comes from, the first one-third of the estimates derived for the next frame and the last one-third of the estimates derived for the previous frame. If there is only one pitch period estimate for the next and/or previous frame, that one estimate is taken. Otherwise, the number of estimates taken for the next or previous frame is the nearest integer greater than or equal to one-third the total number of estimates found. The median of all gathered estimates defines the pitch period for the 10 ms segment of interest. In computing the median, any estimate of 0 is not counted. This postprocessing step takes advantage of the fact that the pitch period varies slowly from frame to frame, eliminates outliers due to algorithm imperfection and mitigates pitch doubling and tripling.

3. OTHER METHODS

We compare the new AGDF method to the approach based on the dyadic wavelets (DyWT) [5] and to the pitch prediction (PP) formulation [6] that is used in speech coding.

3.1. Dyadic Wavelet Approach

The dyadic wavelet transform (DyWT) has been used for image analysis [8], image coding [9] and pitch detection of speech [5]. The wavelet function used to obtain the transform is a cubic spline which is in turn the first derivative of a smoothing function [5]. The general properties of the cubic spline wavelet are particularly good for obtaining the DyWT and using it for speech analysis. The multiresolution properties of the DyWT make it very attractive since the signal can be examined at different levels of detail. In addition, the modulus of the DyWT of a signal exhibits local maxima around the points of discontinuity [5]. In speech, there are sharp signal discontinuities at the points of glottal closure which correspond to the pitch pulses.

As for the AGDF algorithm, the speech is partitioned into 30 ms frames and a set of pitch periods are found for each frame. Successive frames have a 20 ms overlap. One pitch period estimate is found for the 10 ms segment that is centered about the frame. The steps of the DyWT method for a particular 30 ms frame are summarized as follows:

1. The DyWT of a 30 ms speech frame is computed on a dyadic scale corresponding to 2^j for $j = 4$ and $j = 5$.
2. The global maximum value of the DyWT for $j = 4$ and $j = 5$ are found.
3. A set of estimated pitch periods are taken by (1) picking the local peaks (or maxima) of the DyWT which exceed the threshold value equal to 0.8 of the global maximum, (2) finding the corresponding time indices of these local peaks and (3) finding the difference in the time indices between successive local peaks. This is done for the DyWT obtained for both $j = 4$ and $j = 5$.
4. A check is made to see whether the number of local maxima and their locations of the DyWT for $j = 4$ and $j = 5$ match. If the locations match, the pitch periods obtained for $j = 4$ are chosen. If they do not match, a comparison of the DyWT for $j = 5$ and $j = 6$ is performed as above. Now, if there is a match in the locations of the local maxima, the pitch periods for $j = 5$ are chosen. If there is still no match, no pitch period is found for this frame.
5. Pitch period estimates below 2.5 ms and above 25 ms are discarded. There are two peaks that will have led to the discarded estimate. Of these two peaks, the pitch peak that has a lower absolute DyWT amplitude is ignored. The pitch periods are recalculated.

SNR (in dB)	Additive white noise	Colored noise	Babble noise
30	100 100 99.7	100 100 99.7	100 100 99.7
20	100 100 93.8	100 100 94.3	100 100 93.6
10	99.9 99.9 80.6	99.9 99.9 81.8	99.8 99.9 82.1
5	95.3 90.9 79.8	99.7 99.7 77.0	98.5 98.5 79.7
0	74.4 68.9 79.1	99.0 93.2 68.9	95.3 92.1 70.4
-5	71.1 69.9 66.1	88.0 85.9 62.8	77.0 72.1 60.9
-10	59.5 59.5 62.9	72.5 72.3 62.3	62.5 63.3 58.3

Table 1. Relative accuracy as a percentage for the pitch estimates of synthetic speech (vowel /a/ with 10 ms pitch period) corrupted by different types of noise. Entries along a row refer to the performance of the AGDF, DyWT and PP methods, respectively.

SNR (in dB)	Additive white noise	Colored noise	Babble noise
30	100 100 98.9	100 100 98.4	100 100 98.4
20	100 100 90.2	99.9 99.9 90.3	99.9 99.9 89.2
10	100 100 70.9	99.8 99.8 79.3	99.7 99.7 76.7
5	99.4 99.4 70.4	99.8 99.7 73.4	99.5 99.5 72.5
0	98.6 97.5 66.6	99.4 94.5 70.9	99.3 98.3 72.4
-5	87.6 86.5 65.6	98.5 90.9 70.0	94.7 93.5 65.5
-10	68.4 64.5 63.0	88.7 85.3 63.9	75.5 69.3 60.2

Table 2. Relative accuracy as a percentage for the pitch estimates of synthetic speech (vowel /u/ with 17.5 ms pitch period) corrupted by different types of noise. Entries along a row refer to the performance of the AGDF, DyWT and PP methods, respectively.

If no pitch estimates are found, the pitch period for this frame is made equal to 0. Candidate pitch period estimates for each 30 ms frame are found. As for the AGDF method, a final postprocessing step of taking the median of all gathered estimates from the current frame and neighboring frames defines the pitch period for the 10 ms segment of interest.

3.2. Pitch Prediction Formulation

The pitch prediction (PP) formulation models the evolution of the pitch information in the speech signal and allows for distant-sample prediction in the form of a difference equation [6]

$$s(n) = \beta s(n-M) + e(n) \tag{3}$$

where $s(n)$ is the speech signal, $e(n)$ is the prediction error, M is the pitch estimate in samples and β is a prediction coefficient. Minimization of the mean-square value of $e(n)$ over a frame leads to a pitch estimate as the value of M between 20 and 200 (2.5 ms to 25 ms range for 8 kHz sampled speech) that maximizes a correlation function given by [6]

$$\frac{[\sum_{n=1}^{N} s(n)s(n-M)]^2}{\sum_{n=1}^{N} s^2(n-M)} \tag{4}$$

where N is the number of samples in the frame. For the PP method, we use a framesize of 5 ms since it has been found that this small framesize leads to more accurate pitch estimates.

4. RESULTS AND DISCUSSION

We tested the accuracy and robustness of our new AGDF method against the DyWT and PP methods and present the results below. We first used synthesized speech sampled at 8 kHz as was done in [5]. This is convenient since the pitch period can be set apriori and the accuracy of the methods can be compared. For quantifying the performance of the algorithms, we use the relative accuracy (similar to the relative error used in [5]) which is expressed as a percentage as

$$\text{Relative Accuracy} = [1 - \frac{1}{N}\sum_{k=1}^{N} \frac{|x - y_k|}{x}] \times 100\% \tag{5}$$

where N is the number of frames, x is the true pitch period of the synthesized speech and y_k is the estimated pitch period in the kth frame. The robustness of the algorithms are tested by corrupting the synthetic speech with different types of noise, namely, additive white Gaussian, colored and babble. Different signal to noise ratios (SNR) are tested. The colored noise was generated by passing white Gaussian noise through a recursive linear predictive filter computed from a frame of speech corresponding to a sustained vowel. Babble noise was generated by combining the speech signals (different utterances) of 10 interfering speakers (similar to the noise heard at a cocktail party).

Table 1 shows the results for synthesized speech (vowel /a/) having a pitch period of 10 ms. Table 2 shows the results for synthesized (vowel /u/) having a pitch period of 17.5 ms. Both the AGDF and DyWT approaches outperform the PP formulation for SNRs between -5 dB and 10 dB. The AGDF approach is slightly better than the DyWT approach for SNRs less than or equal to 0 dB. The relative accuracy is generally the lowest for the case of white Gaussian noise. Both the AGDF and DyWT algorithms perform better for the higher pitch period of 17.5 ms. The reason appears to be the fact that both algorithms use smoothing. In the process of smoothing, controlling the smoothing of closely positioned peaks in the presence of noise is very difficult to accomplish.

In terms of real speech, we first demonstrate the ability of the AGDF to find the pitch of the vowel /a/ spoken by a male speaker continuously. Since this is real speech, the pitch period of the speaker is not known apriori and can vary slightly with time. The AGDF, DyWT and PP methods reveal that for clean speech, the pitch period is about 57 samples (about 7.1 ms). Figure 1 shows the corresponding pitch tracks obtained by the AGDF method for clean speech, speech corrupted by white Gaussian noise (SNR of 5 dB) and speech corrupted by colored noise (SNR of 5 dB). Some pitch doubling is observed.

We now provide an example of the use of the AGDF algorithm for real conversational speech (spoken by a female) taken from the TIMIT database. The signal consists of several temporal portions of silent, unvoiced and voiced segments. The algorithm will determine the pitch period for all voiced segments and set the pitch period to zero for unvoiced and silent segments. Prior to application of the AGDF algorithm, energy thresholding was used to first discriminate between speech and silent segments. For the speech part, voiced segments are those for which eight of the twelve poles of the linear predictive filter [10] have a magnitude between 0.85 and 1. This approach has been used in the context of speaker recog-

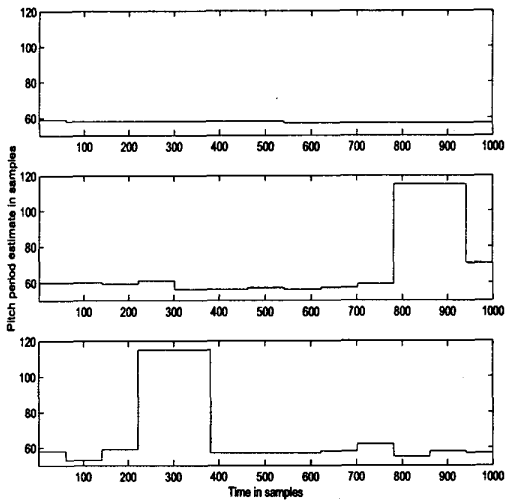

Fig. 1. Pitch tracks obtained by the AGDF method for the vowel /a/ spoken by a male. The top plot shows the pitch track for clean speech. The middle plot shows the pitch track for speech corrupted by white Gaussian noise (SNR of 5 dB). The last plot shows the pitch track for speech corrupted by colored noise (SNR of 5 dB).

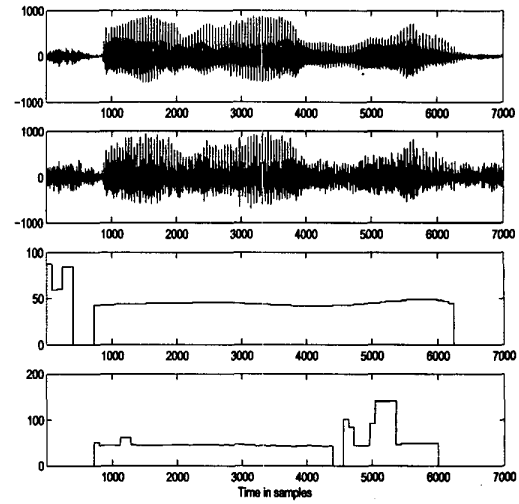

Fig. 2. Pitch tracks obtained by the AGDF method for conversational speech spoken by a female. The top plot shows the clean speech. The second plot shows the speech corrupted by babble noise (SNR of 10 dB). The third plot shows the pitch track for the clean speech. The fourth plot shows the pitch track for speech corrupted by babble noise (SNR of 10 dB).

nition to select only the voiced frames and use them to identify the speaker [11]. The linear predictive analysis was performed using the autocorrelation method [10]. Also, the AGDF method was applied to the linear predictive residual which is the original speech filtered by the linear predictive coefficients. This prior filtering was found to improve the pitch estimates. In fact, it has been demonstrated that the PP formulation as applied to the linear predictive residual is more beneficial for speech coding [6]. Figure 2 shows a conversational speech segment (clean and corrupted with babble noise having an SNR of 10 dB) and the associated pitch tracks. A zero pitch value for low energy unvoiced segments is observed. There are some frames for which estimates due to pitch doubling and tripling occur.

5. SUMMARY AND CONCLUSIONS

In this paper, an Adaptive Gaussian Derivative filter (AGDF), commonly used in image processing, has been introduced for the determination of the pitch period in speech. The cubic spline DyWT wavelet and the pitch prediction (PP) formulation are used for comparison. It is shown that the AGDF method is slightly more robust to noise than the DyWT pitch detector. The AGDF method is much better than the PP approach. The AGDF algorithm can be used for the accurate determination of the pitch period in conversational speech even under noisy conditions.

6. REFERENCES

1. W. Hess, *Pitch Determination of Speech Signals: Algorithms and Devices*, Springer Verlag, 1983.
2. K. R. Schrer, "Speech and emotional states", in *Speech Evaluation in Psychiatry*, edited by G. J. K. Darby, Grune and Stratton, 1981.
3. M. Ramalho, "The pitch mode modulation model and its application in speech processing", in *Modern Methods of Speech Processing*, edited by R. P. Ramachandran and R. J. Mammone, Kluwer Academic Publishers, 1995.
4. M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*, Prentice-Hall, 1995.
5. S. Kadambe and G. F. Boudreaux-Bartels, "Application of the wavelet transform for the pitch detection of speech signals", *IEEE Transactions on Information Theory*, Vol. 38, No. 2, pp. 917–924, March 1992.
6. R. P. Ramachandran and P. Kabal, "Pitch prediction filters in speech coding", *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. 37, No. 4, pp. 467–478, April 1989.
7. M. Basu, "Gaussian derivative model for edge enhancement", *Pattern Recognition*, Vol. 27, No. 11, pp. 1451–1461, 1994.
8. S. G. Mallat and S. Zhong, "Complete signal representation with multiscale edges", Tech. rep. RRT-483-RR-219, Courant Inst. of Math. Sci., December 1989.
9. S. Zhong and S. G. Mallat, "Compact image representation from multiscale edges", *Proc. Third Int. Conf. on Computer Vision*, New York, NY, December 1990.
10. L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Prentice-Hall, 1978.
11. K. T. Assaleh and R. J. Mammone, "New LP derived features for speaker identification", *IEEE Transactions on Speech and Audio Processing*, Vol. 2, pp. 630–638, October 1994.