(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 713 208 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
22.05.1996 Bulletin 1996/21

(51) Int. Cl.[6]: **G10L 3/00**

(21) Application number: **95118142.9**

(22) Date of filing: **17.11.1995**

(84) Designated Contracting States:
DE FR GB

(30) Priority: **21.11.1994 US 342494**

(71) Applicant: **ROCKWELL INTERNATIONAL CORPORATION**
Seal Beach, California 90740-8250 (US)

(72) Inventors:
- **Su, Huan-Yu**
  Irvine, California 92714 (US)
- **Li, Tom Hong**
  Middletown, New Jersey 07748 (US)

(74) Representative: **Wagner, Karl H. et al**
WAGNER & GEYER
Patentanwälte
Gewürzmühlstrasse 5
D-80538 München (DE)

(54) Pitch lag estimation system

(57) A pitch estimation device and method utilizing a multi-resolution approach to estimate speech pitch lag. The system includes sampling the speech 602 and applying, alternately, a discrete Fourier transform 606 and squaring the result 608. A DFT on the squared amplitude is then performed 610 to transform the speech samples into another domain. An initial pitch lag can then be found with lower resolution. After getting the low-resolution pitch lag estimate, a refined algorithm is applied 618 to get a higher-resolution pitch lag. The refined algorithm is based on minimizing the prediction error in the time domain. The refined pitch lag then can be used directly in the speech coding.

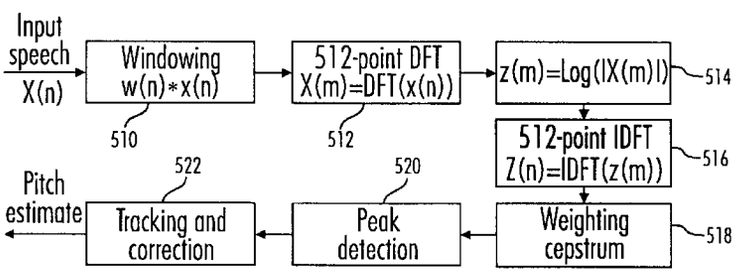

Fig. 5

EP 0 713 208 A2

Description

BACKGROUND OF THE INVENTION

Signal modeling and parameter estimation play increasingly important roles in data compression, decompression, and coding. To model basic speech sounds, speech signals must be sampled as a discrete waveform to be digitally processed. In one type of signal coding technique, called linear prediction coding (LPC), the signal value at any particular time index is modeled as a linear function of previous values. A subsequent signal is thus linearly predicted according to an earlier value. As a result, efficient signal representations can be determined by estimating and applying certain prediction parameters to represent the signal. Presently, LPC techniques are being used for speech coding involving code excited linear prediction (CELP).

It is recognized that pitch information is a reliable indicator and representative of sounds for coding purposes. Pitch describes a key feature or parameter of a speaker's voice. Because human speech is generally not easily mathematically quantifiable, speech estimation models which can effectively estimate the speech pitch data provide for more accurate and precise coded and decoded speech. In current speech coding models, however, such as certain CELP (e.g., vector sum excited linear prediction (VSELP), multi-pulse, regular pulse, algebraic CELP, etc.) and MBE coder/decoders ("codecs"), pitch estimation is often difficult due to the need for high precision and low complexity of the pitch estimation algorithm.

Several pitch lag estimation schemes are used in conjunction with the abovementioned codecs: a time domain approach, frequency domain approach, and cepstrum domain approach. The precision of the pitch estimation has a direct impact on the speech quality due to the close relationship between pitch lag and speech reproduction. In CELP coders, speech generation is based on predictions -- long-term pitch prediction and short-term linear prediction. Figure 1 shows a speech regeneration block diagram of a typical CELP coder.

To compress speech data, it is desirable to extract only essential information to avoid transmitting redundancies. Speech can be grouped into short blocks, where representative parameters can be identified in all of the blocks. As indicated in Figure 1, to generate good quality speech, a CELP speech coder must extract LPC parameters 110, pitch lag parameters 112 (including lag and its coefficient), and an optimal innovation code vector 114 with its gain parameter 116 from the input speech to be coded. The coder quantizes the LPC parameters by implementing appropriate coding schemes. The indices of quantization of each parameter comprise the information to be stored or transmitted to the speech decoder. In CELP codecs, determination of pitch prediction parameters (pitch lag and pitch coefficients) is performed in the time domain, while in MBE codecs, pitch parameters are estimated in the frequency domain.

After LPC analysis, the CELP encoder determines an appropriate LPC filter 110 for the current speech coding frame (usually taken about 10-40 ms). The LPC filter is represented by the equation:

$$A(z) = 1 - a_1 z^{-1} - a_2 z^{-2} - \ldots - a_{np} z^{-np}$$

or the nth sample can be predicted by

$$\hat{y}(n) = \sum_{k=1}^{np} a_k * y(n-k)$$

where np is the LPC prediction order (usually approximately 10), y(n) is sampled speech data, and n represents the time index. The LPC equations above describe the estimation of the current sample according to the linear combination of the past samples. A perceptual weighting filter based on the LPC filter which models the sensitivity of the human ear is then defined by

$$W(z) = \frac{A(z/\gamma_1)}{A(z/\gamma_2)}$$

where $0<\gamma_2<\gamma_1\leqq 1$

To extract the desired pitch parameters, the pitch parameters which minimize the following weighted coding error energy must be calculated for each coding subframe, where one coding frame may be divided into several coding subframes for analysis and coding:

$$d = \|T - \beta P_{Lag} H - \alpha C_i H\|^2$$

where T is the target signal which represents the perceptually filtered input signal, and H is the impulse response matrix of the filter W(z)/A(z). P_{Lag} is the pitch prediction contribution having pitch lag "Lag" and prediction coefficient β which is uniquely defined for a given lag, and C_i is the codebook contribution associated with index i in the codebook and its corresponding gain α. Typically, the pitch of human speech varies from 2 ms - 20 ms. Thus, when the speech is sampled at an 8 KHz sampling rate, the pitch lag corresponds roughly to 20 - 147 samples. In addition, i takes values between 0 and Nc-1, where Nc is the size of the innovation codebook.

A one-tap pitch predictor and one innovation codebook are assumed. Typically, however, the general form of the pitch predictor is a multi-tap scheme, and the general form of the innovation codebook is a multi-level vector quantization, or utilizes multiple innovation codebooks. More particularly, in speech coding, one-tap pitch predictor indicates that the current speech sample can be predicted by a past speech sample, while the multi-tap predictor means that the current speech sample can be predicted by multiple past speech samples.

Due to complexity concerns, sub-optimal approaches have been used in speech coding schemes. For example, pitch lag estimation may be performed by simply evaluating the possible lag values in the range between L_1 and L_2 samples to cover 2.5 ms - 18.5 ms. Consequently, the estimated pitch lag value is determined by maximizing the following:

$$\underset{Lag\in[L_1,L_2]}{Max}\ \frac{(TH^T P_{Lag}^T)^2}{\left\|P_{Lag}H\right\|^2} \qquad \text{Eqn. (1)}$$

Even though this time domain approach may enable the determination of the real pitch lag, for female speech having a high pitch frequency, the pitch lag found by Eqn. (1) may not be the real lag, but a multiple of the real lag. To avoid this estimation error, additional processes are necessary to correct the estimation error (e.g., lag smoothing) at the cost of undesirable complexity.

However, such excess complexity is a significant drawback of using the time domain approach. For example, the time domain approach requires at least 3 million operations per second (MOPs) to determine the lag using integer lag only. Moreover, if pitch lag smoothing and a fractional pitch lag are used, the complexity is more likely approximately 4 MOPs. In practice, approximately 6 million digital signal processing machine instructions per second (DSP MIPs) are required to implement full range pitch lag estimation with acceptable precision. Thus, it is generally accepted that pitch estimation requires 4-6 DSP MIPs. Although there exist other approaches which can reduce the complexity of pitch estimation, such approaches often sacrifice quality.

In MBE coders, an important member in the class of sinusoidal coders, coding parameters are extracted and quantized in the frequency domain. The MBE speech model is shown in Figures 2-4. In the MBE voice encoder/decoder ("vocoder"), described in Figures 2 and 3, the fundamental frequency (or pitch lag) 210, voiced/unvoiced decision 212, and spectral envelop 214 are extracted from the input speech in the frequency domain. The parameters are then quantized and encoded into a bit stream which can be stored or transmitted.

In the MBE vocoder, to achieve high speech quality, the fundamental frequency must be estimated with high precision. The estimation of the fundamental frequency is performed in two stages. First, an initial pitch lag is searched within the range of 21 samples to 114 samples to cover 2.6 - 14.25 ms at the sampling rate of 8000 Hz by minimizing a weighted mean square error equation 310 (Figure 3) between the input speech 216 and the synthesized speech 218 in the frequency domain. The mean square error between the original speech and the synthesized speech is given by the equation:

$$E = \frac{1}{2\pi}\int_{-\pi}^{\pi} G(\omega)\left|S(\omega) - \hat{S}(\omega)\right| d\omega$$

where $S(\omega)$ is the original speech spectrum, $\hat{S}(\omega)$ is the synthesized speech spectrum, and $G(\omega)$ is a frequency-dependent weighting function. As shown in Figure 4, a pitch tracking algorithm 410 is used to update the initial pitch lag estimate 412 by using the pitch information of neighboring frames.

The motivation for using this approach is based upon the assumption that the fundamental frequency should not change abruptly between neighboring frames. The pitch estimates of the two past and two future neighbor frames are

used for the pitch tracking. The mean-square error (including two past and future frames) is then minimized to find a new pitch lag value for the current frame. After tracking the initial pitch lag, a pitch lag multiple checking scheme 414 is applied to eliminate the multiple pitch lag, thus smoothing the pitch lag.

Referring to Figure 4, in the second stage of the fundamental frequency estimation, pitch lag refinement 416 is employed to increase the precision of the pitch estimate. The candidate pitch lag values are formed based on the initial pitch lag estimate (i.e., the new candidate pitch lag values are formed by adding or subtracting some fractional number from the initial pitch lag estimate). Accordingly, a refined pitch lag estimate 418 can be determined among the candidate pitch lags by minimizing the mean square error function.

However, there are certain drawbacks to frequency domain pitch estimation. First, the complexity is very high. Second, the pitch lag must be searched within the range of 20 and 114 samples covering only 2.5 - 14.25 ms to limit the window size to 256 samples to accommodate a 256-point FFT. However, for very low pitch frequency talkers, or for speech having a pitch lag beyond 14.25 ms, it is impossible to gather a sufficient number of samples within a 256-sample window. Moreover, only an averaged pitch lag is estimated over a speech frame.

Using cepstrum domain pitch lag estimation (Figure 5), which was proposed by A.M. Noll in 1967, other modified methods were proposed. In cepstrum domain pitch lag estimation, approximately 37 ms of speech are sampled 510 so that at least two periods of the maximum possible pitch lag (e.g., 18.5 ms) are covered. A 512-point FFT is then applied to the windowed speech frame (at block 512) to obtain the frequency spectrum. taking the logarithm 514 amplitude of the frequency spectrum, another 512-point inverse FFT 516 is applied to get the cepstrum. A weighting function 518 is applied to the cepstrum, and the peak of the cepstrum is detected 520 to determine the pitch lag. A tracking algorithm 522 is then implemented to eliminate any pitch multiples.

Several drawbacks of the cepstrum pitch detection method can be observed however. For example, the computational requirement is high. To cover the pitch range between 20 and 147 samples at an 8 KHz sampling rate, the 512-point FFT must be performed twice. The precision of the estimate is inadequate since the cepstrum pitch estimate will provide only the estimate of an averaged pitch lag over the analysis frame. However, for low bit rate speech coding, it is critical for the pitch lag value to be estimated over a shorter time period. As a result, the cepstrum pitch estimate is almost never used today for high-quality low bit rate speech coding. Thus, because of the limitations of each approach mentioned before, a means for efficient pitch lag estimation is desired to meet the needs of high-quality low bit rate speech coding.

SUMMARY OF THE INVENTION

Accordingly, it is an object of the present invention to provide a pitch estimation system incorporating multi-resolution analysis for speech coding, requiring minimal complexity and greater precision. In particular embodiments, the present invention is directed to a device and method of speech coding using CELP techniques, as well as a variety of other speech coding and recognition systems. Consequently, better results are provided with fewer computational resources, while maintaining the necessary high precision.

These and other objects are accomplished, according to an embodiment of the invention, by a pitch lag estimation scheme which quickly and efficiently enables the accurate reproduction and regeneration of speech. The pitch lag is extracted for a given speech frame and then refined for each subframe. After a minimum number of speech samples have been obtained by sampling speech directly, a Discrete Fourier Transform (DFT) is applied, and the resultant amplitude is squared. A second DFT is then performed. Accordingly, an accurate initial pitch lag for the speech samples within the frame can be determined between the possible minimum value of 20 samples and the maximum lag value of 147 samples at the 8 KHz sampling rate. After obtaining the initial pitch lag estimate, time domain refinement must be performed for each subframe to further improve the estimation precision.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram of a CELP speech model.
Figure 2 is a block diagram of an MBE speech model.
Figure 3 is a block diagram of an MBE encoder.
Figure 4 is a block diagram of pitch lag estimation in an MBE vocoder.
Figure 5 is block diagram of a cepstrum-based pitch lag detection scheme.
Figure 6 is an operational flow diagram of pitch lag estimation according to an embodiment of the present invention.
Figure 7 is a flow diagram of pitch lag estimation according to another embodiment of the present invention.
Figure 8 is a diagrammatic view of speech coding according to the embodiment of Figure 6.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

A pitch lag estimation scheme in accordance with a preferred embodiment of the present invention is indicated generally in Figures 6, 7, and 8. First, N speech samples {x(n), n = 0, 1, ..., N-1} are gathered. (Step 602 of Figure 6) For example, N may equal 320 speech samples to accommodate a typical 40 ms speech window at an 8000 Hz sampling rate. The value of N is determined by the roughly estimated speech period, wherein at least two periods are generally required to generate the speech spectrum. Thus, N must be greater than twice the maximum possible pitch lag, where {x(n), n = 0, 1, ..., N-1}. In addition, a Hamming window 604 or other window which covers at least two pitch periods is preferably implemented.

An N-point DFT is applied in step 606 over {x(n), n = 0, 1, ..., N-1} to get amplitude {Y(f), f = 0, 1, ..., N-1}, where

$$Y(f)=\sum_{n=0}^{N-1} x(n)e^{-j2\pi nf/N} \qquad \text{for f = 0, 1, ... , N-1} \qquad \text{Eqn. (2).}$$

Y(f) is then squared in step 608 according to:

$$G(f)=|Y(f)|^2 \qquad \text{for f = 0, 1, ..., N-1} \qquad \text{Eqn. (3).}$$

A second N-point DFT is applied to G(f) in Step 610 to obtain

$$C(n)=\sum_{f=0}^{N-1} G(f)e^{-j2\pi nf/N} \qquad \text{for n = 0, 1, ... , N-1} \qquad \text{Eqn. (4).}$$

It will be recognized that, according to embodiments of the present invention, C(n) is unlike the conventional cepstrum transformation in which the logarithm of G(f) is used in Eqn. (4) rather than the function G(f). This difference is generally attributable to complexity concerns. It is desirable to reduce the complexity by eliminating the logarithmic function, which otherwise requires substantially greater computational resources. In addition, upon comparison of pitch lag estimation schemes using cepstrum or the C(n) function, varying results have been obtained only for unvoiced or transition segments of the speech. For example, for unvoiced or transition speech, the definition of pitch is unclear. It has been said that there is no pitch in transition speech, while others say that some prediction can always be designated to minimize the error.

Accordingly, once C(n) is determined (step 610), the pitch lag for the given speech frame can be found in step 614 by solving the following:

$$Lag = \arg\left[\underset{n\in[L_1,L_2]}{Max}\sum_{i=n-M}^{n+M} C(i)\bullet W(i-n+M)\right] \qquad \text{Eqn. (5)}$$

where arg [·] determines the variable n which satisfies the inside optimization function, L_1 and L_2 are defined as the minimum and maximum possible pitch lag, respectively. For speech coding convenience, it is desirable for the difference between L_2 and L_1 to be a power of 2 for the binary representation. In preferred embodiments, L_1 and L_2 take values of 20 and 147, respectively, to cover the typical human speech pitch lag range of 2.5 to 18.375 ms, where the distance between L_1 and L_2 is a power of 2. W(i) is a weighting function, and 2M+1 represents the window size. Preferably, {W(i) = 1, i = 0, 1, ..., 2M}, and M = 1.

Although the resultant pitch lag is an averaged value, it has been found to be reliable and accurate. The averaging effect is due to the relatively large analysis window size; for a lag of 147 samples, the window size should be at least twice as large as the lag value. Undesirably, however, with such a large window, signals from some speakers, such as female talkers who typically display a small pitch lag, may contain 4-10 pitch periods. If there is a change in the pitch lag, the proposed pitch lag estimation only produces an averaged pitch lag. As a result, the use of such an averaged pitch lag in speech coding could cause severe degradation in speech estimation and regeneration.

Due to relatively quick changes of pitch information in speech, most speech coding systems based on the CELP model evaluate and transmit the pitch lag once per subframe. Thus, in CELP type speech coding in which one speech frame is divided into several speech subframes which are typically 2-10 ms long (16-80 samples), pitch information is updated in each of the subframes. Accordingly, correct pitch lag values are needed only for the subframes. The pitch lag estimated according to the above scheme, however, does not have sufficient precision for accurate speech coding due to the averaging effect.

Thus, in particular embodiments of the present invention, to improve the estimation precision, a refined search based on the initial pitch lag estimate is performed in the time domain (Step 618). A simple autocorrelation method is performed around the averaged Lag value for the particular coding period, or subframe:

$$Lag = \arg\left[\underset{n\in[Lag-m,Lag+m]}{Max} \sum_{i=k}^{k+l-1} x(i)x(i-n) \right] \qquad \text{Eqn. (6)}$$

where arg [·] determines the variable n which satisfies the inside optimization function, k denotes the first sample of the subframe, l represents the refine window size and m is a searching range. To determine an accurate pitch lag value, the refine window size should be at least one pitch period. The window, however, should not be too large to avoid the effects of averaging. For example, preferably, l = Lag + 10, and m = 5. Thus, according to the time domain refinement of Eqn. 6, a more precise pitch lag can be estimated and applied to the coding of the subframe.

In operation, although the Fast Fourier Transform (FFT) is sometimes more computationally efficient than the general DFT, the drawback of using an FFT is that the window size must be power of 2. For example, it has been shown that the maximum pitch lag of 147 samples is not a power of 2. To include the maximum pitch lag, a window size of 512 samples is necessary. However, this results in a poor pitch lag estimation for female voices due to the averaging effect, discussed above, and the large amount of computation required. If a window size of 256 samples is used, the averaging effect is reduced and the complexity is less. However, to use such a window, a pitch lag larger than 128 samples in the speech cannot be accommodated.

To overcome some of these problems, an alternative preferred embodiment of the present invention utilizes a 256-point FFT to reduce the complexity, and employ a modified signal to estimate the pitch lag. The modification of the signal is a down sampling process. Referring to Figure 7, N speech samples are gathered (Step 702), with N being greater than twice the maximum pitch lag, {x(n), n = 0, 1, ..., N-1}. The N speech samples are then down-sampled into 256 new analysis samples (Step 704) using linear interpolation, according to:

$$y(i) = x([i \cdot \lambda]) + \{x([i \cdot \lambda]+1) - x([i \cdot \lambda])\}(i \cdot \lambda - [i \cdot \lambda]) \qquad \text{for } i = 0, 1, ..., 255$$

where $\lambda = N/256$, and the values within the brackets, i.e., $[i \cdot \lambda]$, denote the largest integer value not greater than $i \cdot \lambda$. A Hamming window, or other window, is then applied to the interpolated data in step 705.

In step 706, the pitch lag estimation is performed over y(i) using a 256-point FFT to generate the amplitude Y(f). Steps 708-710 are then carried out similarly to those described with regard to Figure 6. In addition, however, G(f) is filtered (step 709) to reduce the high frequency components of G(f) which are not useful for pitch detection. Once the lag of y(i), i.e., Lag_y, is found (step 714) according to Eqn. (5), it is rescaled in step 716 to determine the pitch lag estimate:

$$Lag = Lag_y \cdot \lambda$$

In summary, the above procedure to find in initial pitch estimate for the coding frame is as follows:

(1) subdividing the standard 40 ms coding frame into pitch subframes 802 and 804, each pitch subframe being approximately 20 ms long;
(2) taking N = 320 speech samples such that the pitch analysis window 806 is positioned at the center of the last subframe, and find the lag for that subframe using the proposed algorithm; and

(3) determining initial pitch lag values for the pitch subframes.

Time domain refinement is then performed in step 718 over the original speech samples x(n). Thus, in embodiments of the present invention, pitch lag values can be accurately estimated while reducing complexity, yet maintaining good precision. Using FFT embodiments of the present invention, there is no difficulty in handling pitch lag values greater than 120.

More particularly, time domain refinement is performed over the original speech samples. For example, first, the 40 ms coding frame is divided into eight 5 ms subframes 808, as shown in Figure 8. Initial pitch lag estimates lag_1 and lag_2 are the lag estimates for the last coding subframe of each pitch subframe in the current coding frame. Lag_0 is the refined lag estimate of the second pitch subframe in the previous coding frame. The relationship among lag_1, lag_2, and lag_0 is shown in Figure 8.

The initial pitch lags lag_1 and lag_2 are refined first to improve their precision (step 718 in Figure 7) according to:

$$lag_i = \arg\left[\max_{n\in[lag_i-M,\,lag_i+M]} \sum_{k=N_i}^{N_i+L-1} x(k)\bullet x(k-n)\right] \qquad \text{for i = 1, 2}$$

where N_i is the index of the starting sample in the pitch subframe for its pitch lag_i. Preferably, M is selected to be 10, L is lag_i + 10, and i indicates the index of the pitch subframe.

Once the refinement of initial pitch lags is finished, the pitch lags of the coding subframes can be determined. The pitch lags of the coding subframes are estimated by linearly interpolating lag_1, lag_2, and lag_0. The precision of the pitch lag estimates of the coding subframes is improved by refining the interpolated pitch lag of each coding subframe according to the following procedure. If $\{lag_I(i), i = 0, 1, ..., 7\}$ represents the interpolated pitch lags of coding subframes based on the refined initial pitch estimates lag_1, lag_2, and lag_0, $lag_I(i)$ is determined by:

$$\underset{\sim}{lag}(i) = \begin{cases} lag_0 + (lag_1 - lag_0)*\dfrac{i+1}{4} & i = 0, 1, 2, 3 \\ lag_1 + (lag_2 - lag_1)*\dfrac{i-3}{4} & i = 4, 5, 6, 7 \end{cases}$$

Because the precision of the pitch lag estimates given by linear interpolation is not sufficient, further improvement may be required. For the given pitch lag estimates $\{lag_I(i), i = 0, 1, ..., 7\}$, each $lag_I(i)$ is further refined (step 722) by:

$$lag(i) = \arg\left[\max_{n\in[\underset{\sim}{lag}(i)-M,\,\underset{\sim}{lag}(i)+M]} \sum_{k=N_i}^{N_i+L-1} x(k)\bullet x(k-n)\right] \qquad i = 0, 1, ..., 7$$

where Ni is the index of the starting sample in the coding subframe for pitch lag(i). In the example, M is chosen to be 3, and L equals 40.

Furthermore, the linear interpolation of pitch lag is critical in unvoiced segments of speech. The pitch lag found by any analysis method tends to be randomly distributed for unvoiced speech. However, due to the relatively large pitch subframe size, if the lag for each subframe is too close to the initially determined subframe lag (found in step (2) above), an undesirable artificial periodicity that originally was not in the speech is added. In addition, linear interpolation provides a simple solution to problems associated with poor quality unvoiced speech. Moreover, since the subframe lag tends to be random, once interpolated, the lag for each subframe is also very randomly distributed, which guarantees voice quality.

It should be noted that the objects and advantages of the invention may be attained by means of any compatible combination(s) particularly pointed out in the items of the following summary of the invention and the appended claims.

SUMMARY OF INVENTION

1. A system for estimating pitch lag for speech quantization and compression, the speech being defined by a plurality of speech samples, wherein the estimation of a current speech sample is determined in the time domain according to a linear combination of past samples, the system comprising:
means for applying a first discrete Fourier transform (DFT) to the samples, the first DFT having an associated amplitude;
means for squaring the amplitude of the first DFT;
means for applying a second DFT over the squared amplitude;
means for determining an initial pitch lag value according to the time domain-transformed speech samples; and
means for coding the speech samples according to the refined pitch lag value.

2. The system wherein the initial pitch lag value has an associated prediction error, the system further comprising means for refining the initial pitch lag value, wherein the associated prediction error is minimized.

3. The system further comprising:
means for grouping the plurality of speech samples into a current coding frame;
means for dividing the coding frame into multiple pitch subframes;
means for subdividing the pitch subframes into multiple coding subframes;
means for estimating initial pitch lag estimates lag_1 and lag_2 which represent the lag estimates, respectively, for the last coding subframe of each pitch subframe in the current coding frame;
means for refining the pitch lag estimate lag_0 of the second pitch subframe in the previous coding frame;
means for linearly interpolating lag_1, lag_2, and lag_0 to estimate pitch lag values of the coding subframes; and
means for further refining the interpolated pitch lag of each coding subframe.

4. The system further comprising means for downsampling the speech samples to a downsampling value for approximate representation by fewer samples.

5. The system wherein the initial pitch lag value is scaled according to the equation: Lag_{scaled} = Number speech samples/Downsampling value .

6. The system wherein the means for refining the initial pitch lag value comprises autocorrelation.

7. The system further comprising:
speech input means for receiving the speech samples;
a computer for processing the refined pitch lag value to reproduce the input speech as coded speech; and
speech output means for outputting the coded speech.

8. A speech coding apparatus for reproducing and coding input speech, the speech coding apparatus operable with linear prediction coding (LPC) parameters and an innovation codebook representing a plurality of vectors which are referenced to excite speech reproduction to generate speech, the speech coding apparatus comprising:
speech input means for receiving the input speech;
a computer for processing the input speech, wherein the computer includes:
means for segregating a current coding frame within the input speech,
means for dividing the coding frame into plural pitch subframes,
means for defining a pitch analysis window having N speech samples, the pitch analysis window extending across the pitch subframes,
means for estimating an initial pitch lag value for each pitch subframe,
means for dividing each pitch subframe into multiple coding subframes,
wherein the initial pitch lag estimate for each pitch subframe represents the lag estimate for the last coding subframe of each pitch subframe in the current coding frame, and
means for linearly interpolating the estimated pitch lag values between the pitch subframes to determine a pitch lag estimate for each coding subframe, and
means for refining the linearly interpolated lag values of each coding subframe; and
speech output means for outputting speech reproduced according to the refined pitch lag values.

9. The apparatus wherein the computer further includes

means for downsampling the N speech samples to a downsampling value X for representation by fewer samples, and

means for scaling the pitch lag value such that the scaled lag value $Lag_{scaled} = N/X$.

10. The apparatus further comprising sampling means which sample the input speech at a sampling rate R, wherein the N speech samples are determined according to the equation $N = R * X$.

11. The apparatus wherein X = 25 ms, R = 8000 Hz, and N = 320 samples.

12. The apparatus wherein each coding frame has a length of approximately 40 ms.

13. A method for estimating pitch lag for speech quantization and compression, the speech being defined by a plurality of speech samples, wherein the estimation of a current speech sample is determined in the time domain according to a linear combination of past samples, the method comprising the steps of:

applying a first discrete Fourier transform (DFT) to the samples, the first DFT having an associated amplitude;

squaring the amplitude of the first DFT;

applying a second DFT over the squared amplitude of the first DFT;

determining an initial pitch lag value according to the time domain-transformed speech samples, the initial pitch lag value having an associated prediction error;

refining the initial pitch lag value using autocorrelation, wherein the associated prediction error is minimized; and

coding the speech samples according to the refined pitch lag value.

14. The method further comprising the steps:

grouping the plurality of speech samples into a current coding frame;

dividing the coding frame into multiple pitch subframes;

subdividing the pitch subframes into multiple coding subframes;

estimating initial pitch lag estimates lag_1 and lag_2 which represent the lag estimates, respectively, for the last coding subframe of each pitch subframe in the current coding frame;

refining the pitch lag estimate lag_0 of the second pitch subframe in the previous coding frame;

linearly interpolating lag_1, lag_2, and lag_0 to estimate pitch lag values of the coding subframes; and

further refining the interpolated pitch lag of each coding subframe.

15. The method further comprising the step of downsampling the speech samples to a downsampling value for approximate representation by fewer samples.

16. The method further comprising the step of scaling the initial pitch lag value according to the equation: Lag_{scaled} = Number speech samples/Downsampling value.

17. The system further comprising the steps of:

receiving the speech samples;

processing the refined pitch lag value to reproduce the input speech as coded speech; and

outputting the coded speech.

18. A speech coding method for reproducing and coding input speech, the speech coding apparatus operable with linear prediction coding (LPC) parameters and an innovation codebook representing pseudo-random signals which form a plurality of vectors which are referenced to excite speech reproduction to generate speech, the speech coding method comprising the steps of:

receiving and processing the input speech;

processing the input speech, wherein the step of processing includes:

determining a speech coding frame within the input speech,

subdividing the coding frame into plural pitch subframes,

defining a pitch analysis window having N speech samples, the pitch analysis window extending across the pitch subframes,

roughly estimating an initial pitch lag value for each pitch subframe,

dividing each pitch subframe into multiple coding subframes, such that the initial pitch lag estimate for each pitch subframe represents the lag estimate for the last coding subframe of each pitch subframe, and

interpolating the estimated pitch lag values between the pitch subframes for determining a pitch lag estimate

for each coding subframe, and
refining the linearly interpolated lag values; and
outputting speech reproduced according to the refined pitch lag values.

19. The method wherein the step of processing further includes the steps of
downsampling the N speech samples to a downsampling value X for representation by fewer samples, and
scaling the pitch lag value such that the scaled lag value $Lag_{scaled} = N/X$.

20. The method further comprising the steps of sampling the input speech at a sampling rate R, such that the N speech samples are determined according to the equation $N = R * X$.

Claims

1. A system for estimating pitch lag of a plurality of speech samples, the system comprising:
means for applying a first discrete Fourier transform (DFT) 606 to the samples, the first DFT having an associated amplitude;
means for squaring the amplitude 608 of the first DFT 606;
means for applying a second DFT 610 over the squared amplitude 608;
means for determining an initial pitch lag value 614 according to the time domain-transformed speech samples; and
means for coding the speech samples according to the pitch lag value 614.

2. The system of claim 1, further comprising:
means for grouping the plurality of speech samples 602 into a current coding frame;
means for dividing the coding frame into multiple pitch subframes 802, 804;
means for subdividing the pitch subframes 802, 804 into multiple coding subframes 808;
means for estimating initial pitch lag estimates lag_1 and lag_2 which represent lag estimates, respectively, for the last coding subframe of each pitch subframe in the current coding frame;
means for estimating pitch lag estimate lag_0 which represents the lag estimate for the last coding subframe 808 of the previous coding frame;
means for refining 718 the pitch lag estimate lag_0;
means for linearly interpolating lag_1, lag_2, and lag_0 to estimate pitch lag values of the respective coding subframes 808; and
means for further refining 722 the interpolated pitch lag of each coding subframe 808.

3. The system of claim 1, further comprising means for downsampling 704 the speech samples to a downsampling value for approximate representation by fewer samples, wherein the initial pitch lag value is scaled 716 according to the equation: Lag_{scaled} = Number speech samples/Downsampling value.

4. The system of claim 1, further comprising:
speech input means for receiving the speech samples;
a computer for processing the refined pitch lag value to reproduce the input speech as coded speech; and
speech output means for outputting the coded speech.

5. A speech coding apparatus for reproducing and coding input speech, the speech coding apparatus operable with linear prediction coding (LPC) parameters and a codebook representing a plurality of vectors which are referenced to excite speech reproduction to generate speech, the speech coding apparatus comprising:
speech input means 602 for receiving the input speech;
a computer for processing the input speech, wherein the computer includes:
means for segregating a current coding frame within the input speech,
means for dividing the coding frame into plural pitch subframes 802, 804,
means for defining a pitch analysis window 806 having N speech samples, the pitch analysis window extending across the pitch subframes 802, 804,
means for estimating an initial pitch lag value 714 for each pitch subframe 802, 804,
means for dividing each pitch subframe 802, 804 into multiple coding subframes 808, wherein the initial pitch lag estimate for each pitch subframe 802, 804 represents the lag estimate for the last coding subframe 808 of each pitch subframe 802, 804 in the current coding frame, and
means for linearly interpolating 720 the estimated initial pitch lag values 714 between the pitch subframes 802, 804 to determine a pitch lag estimate for each coding subframe 808, and

means for refining 722 the linearly interpolated lag values 720 of each coding subframe; and
speech output means for outputting speech reproduced according to the refined pitch lag values 722.

6. The apparatus of claim 5, wherein the computer further includes
means for downsampling 704 the N speech samples 702 to a downsampling value X for representation by fewer samples, and
means for scaling 716 the pitch lag value such that the scaled lag value $Lag_{scaled} = N/X$.

7. The apparatus of claim 5, further comprising sampling means which sample the input speech at a sampling rate R, wherein the N speech samples are determined according to the equation $N = R * X$.

8. A method for estimating pitch lag for speech quantization and compression, the speech being defined by a plurality of speech samples, wherein the estimation of a current speech sample is determined in the time domain according to a linear combination of past samples, the method comprising the steps of:
applying a first discrete Fourier transform (DFT) 606 to the samples, the first DFT having an associated amplitude;
squaring the amplitude 608 of the first DFT 606;
applying a second DFT 610 over the squared amplitude 608 of the first DFT 606;
determining an initial pitch lag value 614 according to the time domain-transformed speech samples, the initial pitch lag value having an associated prediction error;
refining the initial pitch lag value 618 using autocorrelation, wherein the associated prediction error is minimized; and
coding the speech samples according to the refined pitch lag value.

9. The method of claim 8, further comprising the steps of:
grouping the plurality of speech samples into a current coding frame;
dividing the coding frame into multiple pitch subframes 802, 804;
subdividing the pitch subframes into multiple coding subframes 808;
estimating initial pitch lag estimates lag_1 and lag_2 714 which represent the lag estimates, respectively, for the last coding subframe 808 of each pitch subframe 802, 804 in the current coding frame;
estimating a pitch lag lag_0 from the last coding subframe 808 of the previous coding frame;
refining 718 the pitch lag estimate lag_0 of the second pitch subframe in the preceding coding frame;
linearly interpolating 720 lag_1, lag_2, and lag_0 to estimate pitch lag values 714 of the coding subframes 808; and
further refining 722 the interpolated pitch lag of each coding subframe 808.

10. The method in any of the preceding claims, further comprising the steps of:
downsampling 704 the speech samples to a downsampling value for approximate representation by fewer samples;
scaling 716 the initial pitch lag value according to the equation: Lag_{scaled} = Number speech samples/Downsampling value.

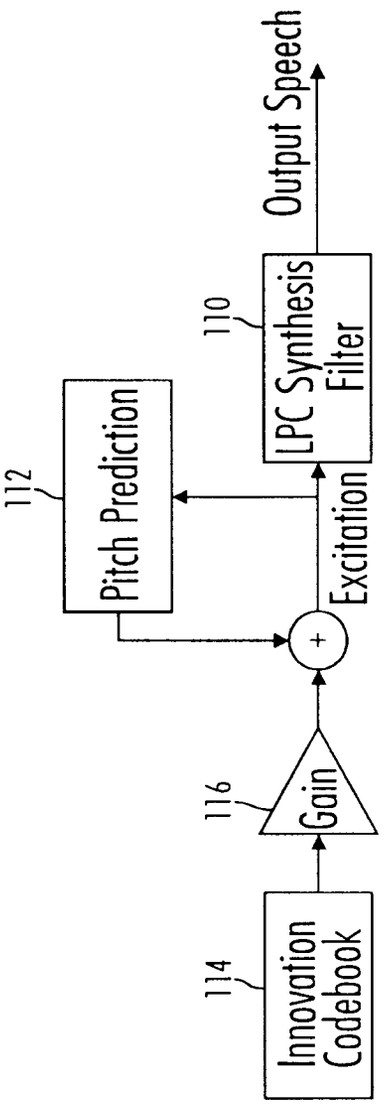
Output Speech
LPC Synthesis Filter
110
Pitch Prediction
112
Excitation
+
Gain
116
Innovation Codebook
114

Fig. 1

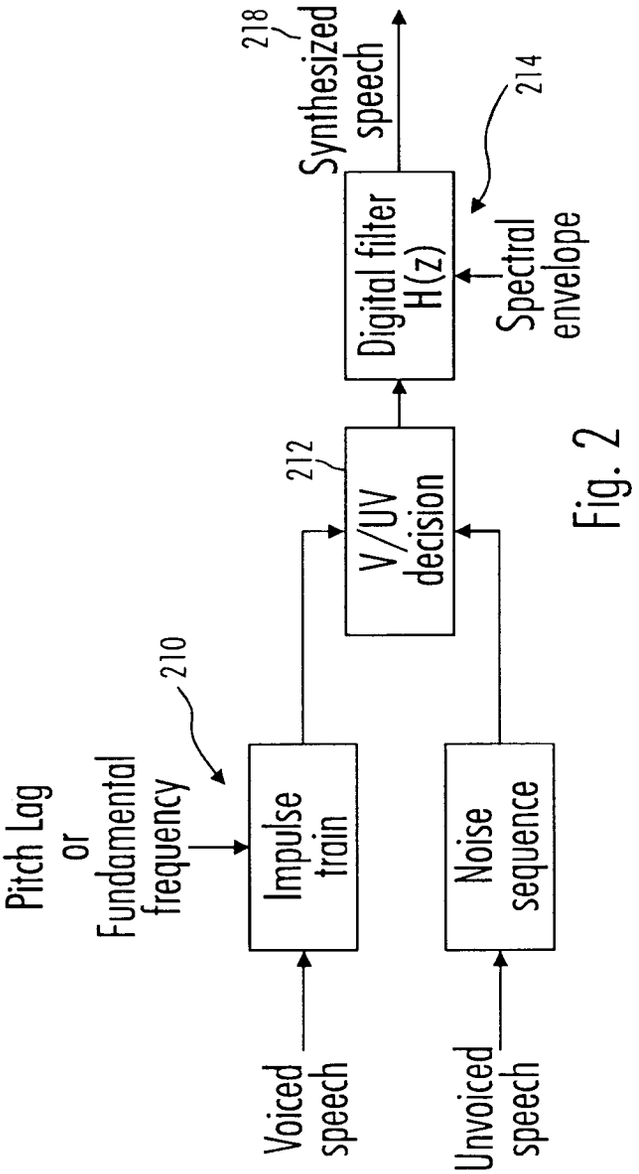
Synthesized speech
218
Digital filter H(z)
214
Spectral envelope
V/UV decision
212
Pitch Lag or Fundamental frequency
210
Impulse train
Noise sequence
Voiced speech
Unvoiced speech

Fig. 2

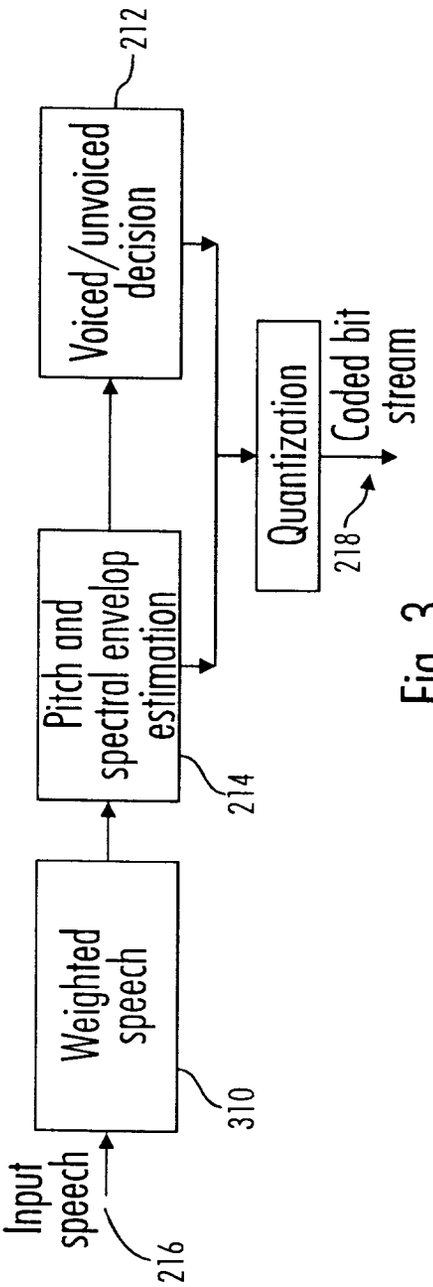
Input speech
216
Weighted speech
310
Pitch and spectral envelop estimation
214
Voiced/unvoiced decision
212
Quantization
218
Coded bit stream

Fig. 3

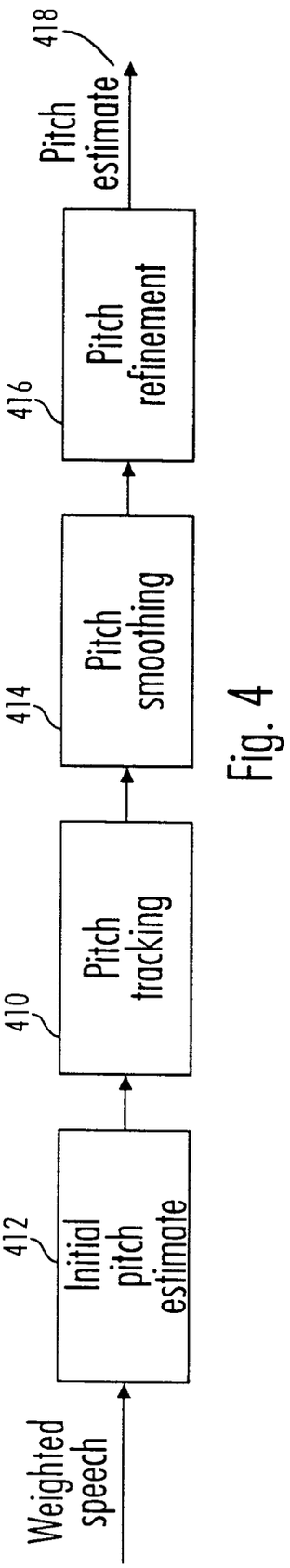
Weighted speech
Initial pitch estimate
412
Pitch tracking
410
Pitch smoothing
414
Pitch refinement
416
Pitch estimate
418

Fig. 4

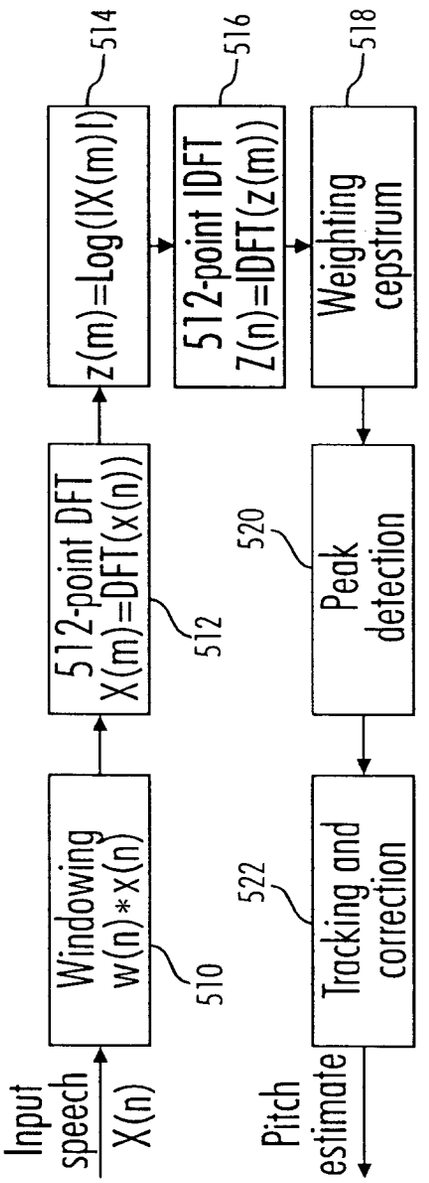
Input speech
X(n)
Windowing
w(n)*x(n)
510
512-point DFT
X(m)=DFT(x(n))
512
z(m)=Log(|X(m)|)
514
512-point IDFT
Z(n)=IDFT(z(m))
516
Weighting cepstrum
518
Peak detection
520
Tracking and correction
522
Pitch estimate

Fig. 5

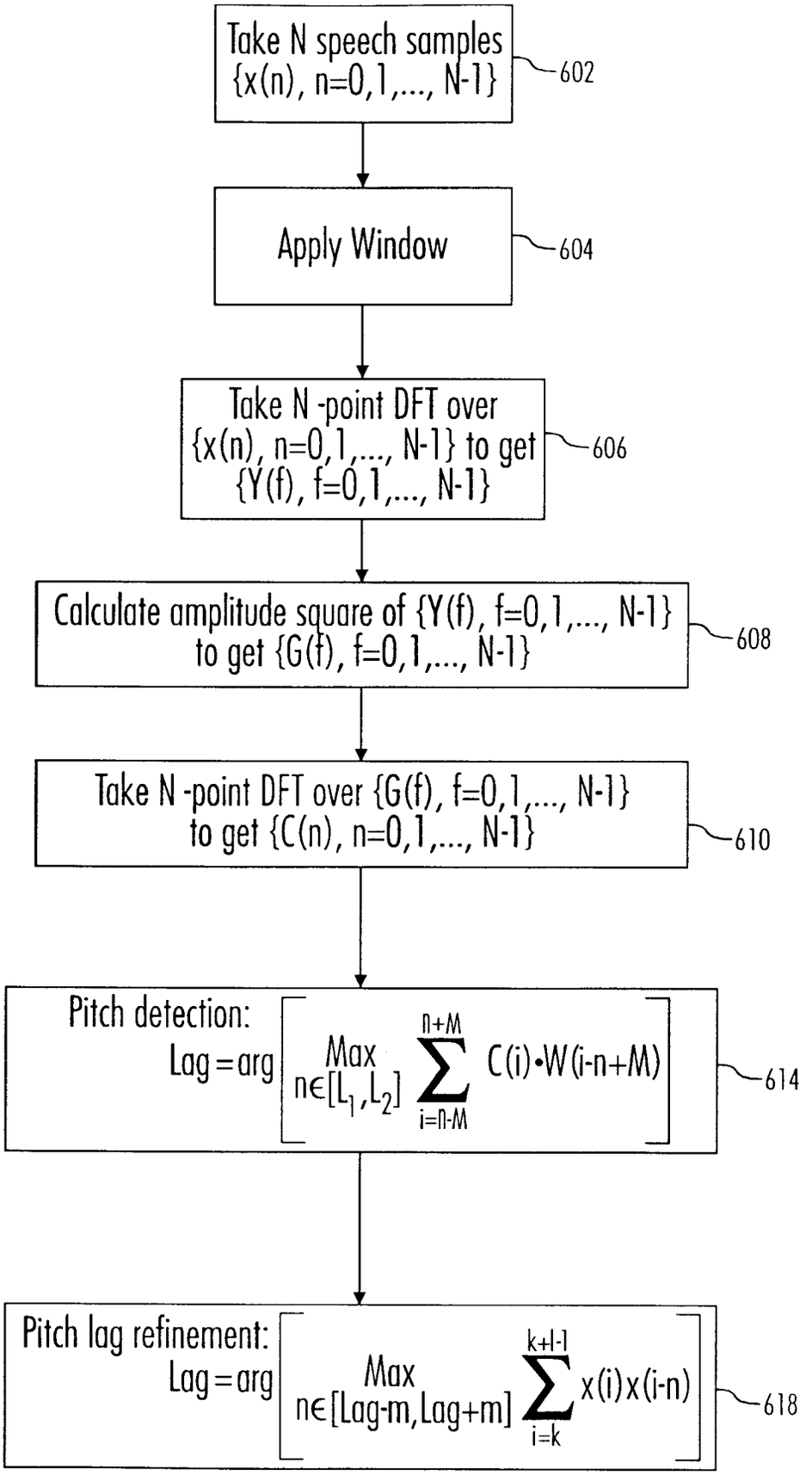
Take N speech samples {x(n), n=0,1,..., N-1}
602
Apply Window
604
Take N -point DFT over {x(n), n=0,1,..., N-1} to get {Y(f), f=0,1,..., N-1}
606
Calculate amplitude square of {Y(f), f=0,1,..., N-1} to get {G(f), f=0,1,..., N-1}
608
Take N -point DFT over {G(f), f=0,1,..., N-1} to get {C(n), n=0,1,..., N-1}
610
Pitch detection:
Lag = arg [Max n∈[L1,L2] Σ i=n-M to n+M C(i)•W(i-n+M)]
614
Pitch lag refinement:
Lag = arg [Max n∈[Lag-m,Lag+m] Σ i=k to k+l-1 x(i)x(i-n)]
618

Fig. 6

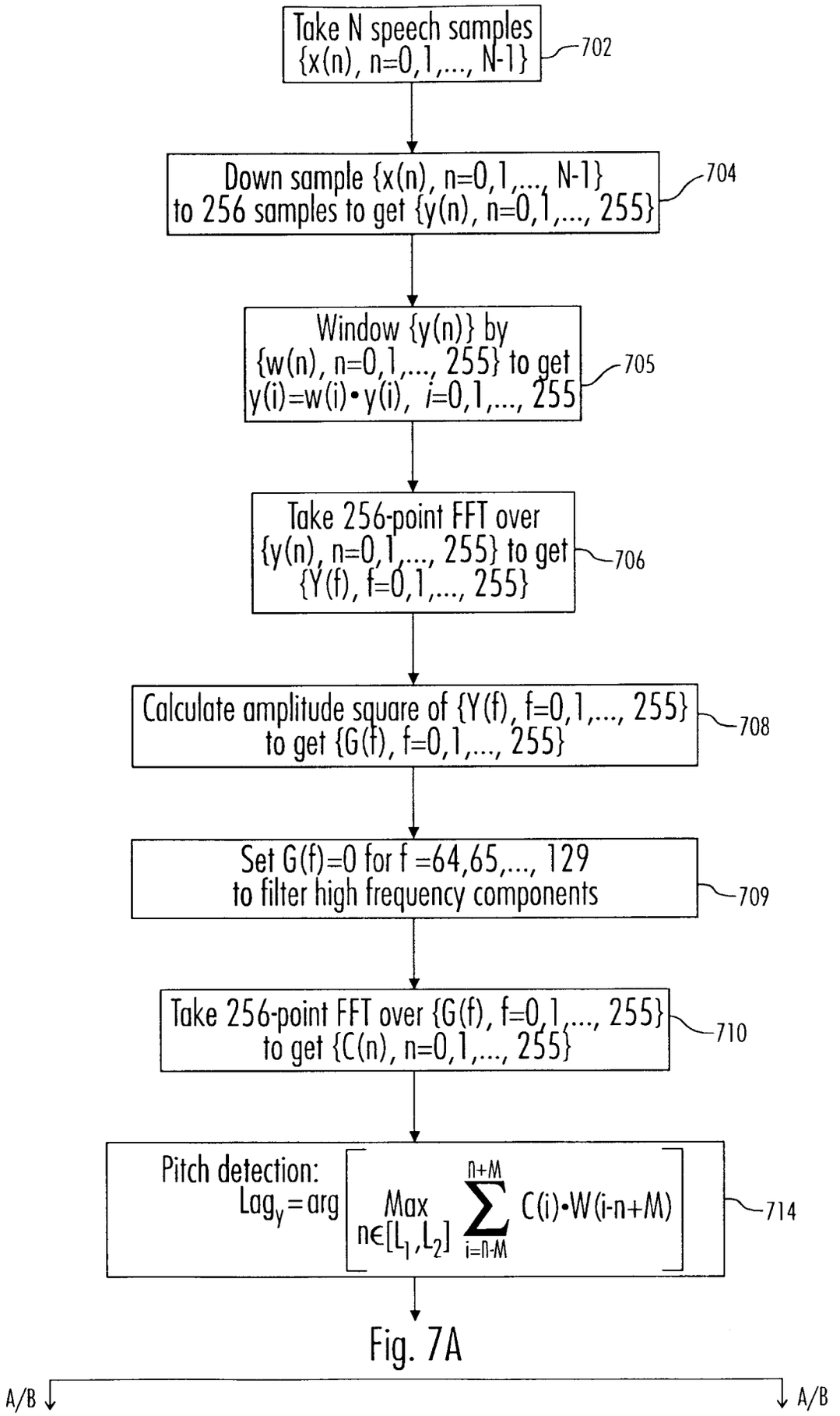
Take N speech samples {x(n), n=0,1,..., N-1}
702
Down sample {x(n), n=0,1,..., N-1} to 256 samples to get {y(n), n=0,1,..., 255}
704
Window {y(n)} by {w(n), n=0,1,..., 255} to get y(i)=w(i)•y(i), i=0,1,..., 255
705
Take 256-point FFT over {y(n), n=0,1,..., 255} to get {Y(f), f=0,1,..., 255}
706
Calculate amplitude square of {Y(f), f=0,1,..., 255} to get {G(f), f=0,1,..., 255}
708
Set G(f)=0 for f =64,65,..., 129 to filter high frequency components
709
Take 256-point FFT over {G(f), f=0,1,..., 255} to get {C(n), n=0,1,..., 255}
710
Pitch detection:
$Lag_y = \arg \left[\max_{n \in [L_1, L_2]} \sum_{i=n-M}^{n+M} C(i) \cdot W(i-n+M) \right]$
714
A/B
A/B

Fig. 7A

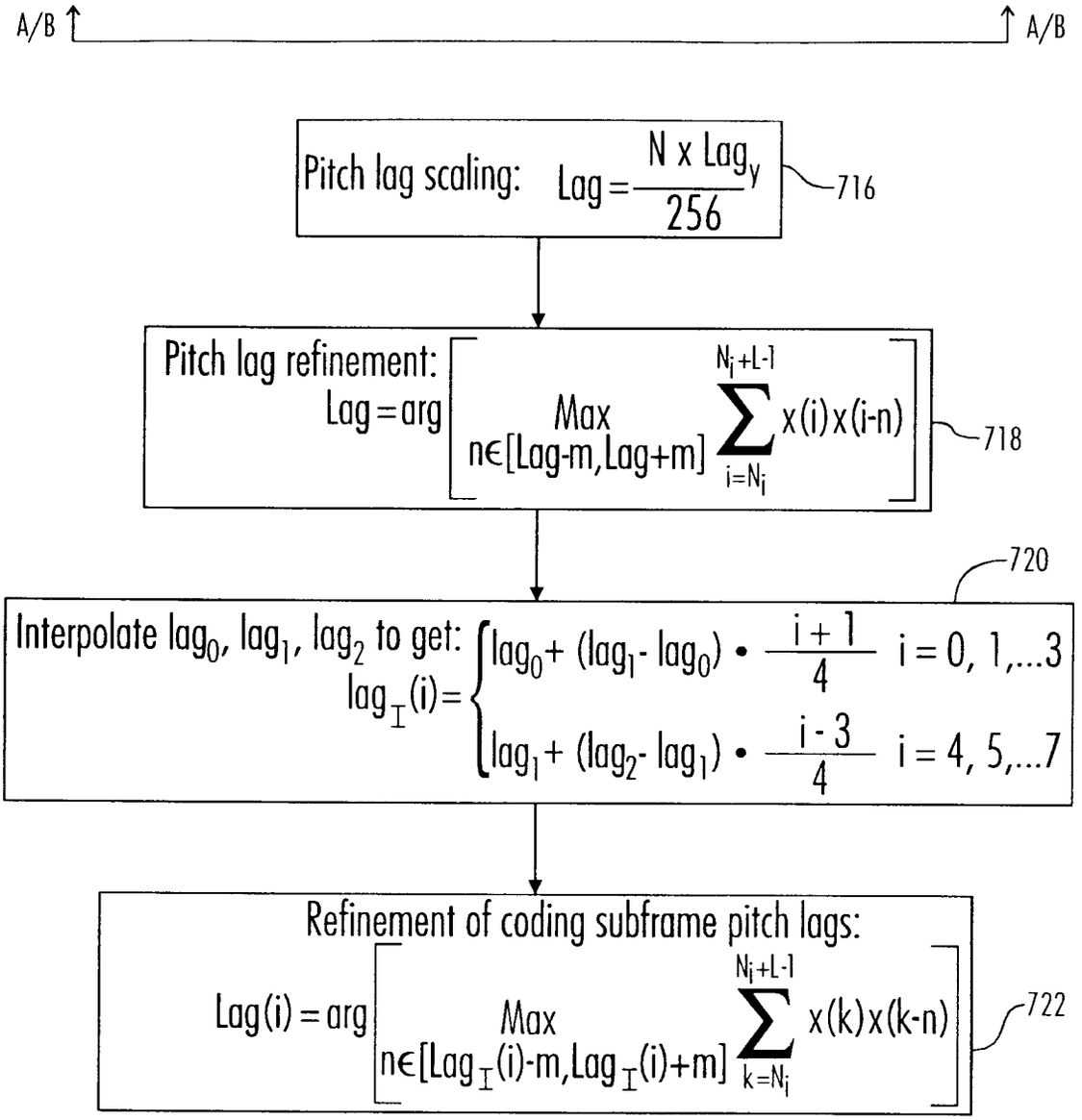
A/B
A/B
Pitch lag scaling: $Lag = \frac{N \times Lag_y}{256}$
716
Pitch lag refinement: $Lag = \arg \left[\underset{n \in [Lag-m, Lag+m]}{Max} \sum_{i=N_i}^{N_i+L-1} x(i)x(i-n) \right]$
718
720
Interpolate lag_0, lag_1, lag_2 to get: $lag_I(i) = \begin{cases} lag_0 + (lag_1 - lag_0) \cdot \frac{i+1}{4} & i = 0, 1, \ldots 3 \\ lag_1 + (lag_2 - lag_1) \cdot \frac{i-3}{4} & i = 4, 5, \ldots 7 \end{cases}$
Refinement of coding subframe pitch lags:
$Lag(i) = \arg \left[\underset{n \in [Lag_I(i)-m, Lag_I(i)+m]}{Max} \sum_{k=N_i}^{N_i+L-1} x(k)x(k-n) \right]$
722

Fig. 7B

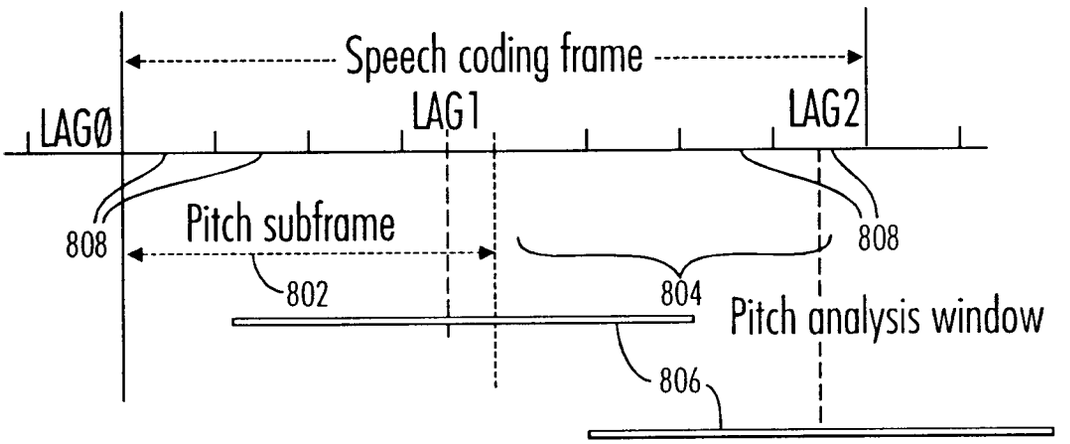
Speech coding frame
LAGØ
LAG1
LAG2
808
Pitch subframe
808
802
804
Pitch analysis window
806

Fig. 8