Pitch tracking using harmonic model

Ing. Igor Szöke
FIT BUT Czech Republic

ABSTRACT: Pitch tracking is important in some speech processing applications (e.g. synthesis, coding, parametrization, ...) We need robust pitch tracker producing reliable smooth pitch line and voiced/unvoiced decision. Both are used in speech synthesis with fundamental frequency modifications. A resonance theory of voice origination is used for candidates computation. Next a dynamic programming is used for best path retrieval. There is a constant for setting smoothness of a pitch line. Some experiments and further work are discussed in the conclusion.

1 Introduction

Presented pitch tracker was designed for use in harmonic and noise speech model (HNM). We use this model in text-to-speech system [4] (for corpus pitch linearization [5]) and in very low bit rate coder [6].

1.1 Harmonic and noise model

Parametric models are usually used in text-to-speech systems. These models are needed for their ability to change speech fundamental frequency and speech rate. One of parametric models is harmonic and noise model [3]. The model splits speech signal \( O(t) \) into two components, a harmonic part \( H(t) \) and a noise part \( N(t) \). The harmonic component originates from the resonances of fundamental frequency in cavities (i.e. mouth, nose and throat) [1]. The noise component originates from the air friction against obstacles.

Any fundamental frequency determination or voiced/unvoiced decision imprecisions are noticeable while listening speech synthesized by HNM. Another problem raises when determined pitch is not smooth. Voiced parts are pitch-synchronous windowed so each window contains one pitch period. Unvoiced parts are windowed too, but they have no pitch so windows have constant length (as was presented in [3]), see picture 7b. This causes some pitch discontinuities between voiced and unvoiced parts. If there is imprecision in voiced/unvoiced decision, bad artefact can be heard there.

We aimed to create a pitch tracker, which can produce precision and smooth pitch line over voiced and unvoiced parts of speech with precise V/UV decision.

1.2 Resonance theory

The resonance theory of voice origination states, that voiced speech is composed of many harmonics of fundamental frequency. That harmonics originate by resonating of fundamental frequency in cavities (buccal, nose, throat). See example of voiced speech spectrum in figure 1.

2 The algorithm

Speech signal is usually windowed (e.g. 10ms length rectangular windows). If we want to find voiced parts of speech (voiced frames), we must find the harmonic component. The
harmonic component is composed of harmonics $F_0^n$ of fundamental frequency. So we can synthesize speech signal of $n$ harmonics of fundamental frequency $F_0$. But we do not know the $F_0$. It usually lies in interval $(50\,\text{Hz}, 350\,\text{Hz})$ for adults. So we can try all frequencies and we will look for minimal difference (minimal error) between original speech signal and synthesized one (see figure 2).

2.1 Error computation

As was said we try to minimize error between original speech signal $O(t)$ and synthesized harmonic speech signal $S_{F_0}(t)$ (for given fundamental frequency $F_0$). We must know parameters of harmonic components (amplitudes and phases) to do this. The determination of parameters is done by FFT. For one fundamental frequency $F_0$ we pick for example 10 harmonics parameters and we synthesize speech signal $S_{F_0}(t)$ from them.

$$S_{F_0}(t) = \sum_{i=1}^{n} A_i \cos(iF_0 + \varphi_i), \quad (1)$$

where amplitudes $A_i$ and phases $\varphi_i$ are given by FFT. Next step is to minimize the error between $O(t)$ and $S_{F_0}(t)$ to avoid imprecisions caused by different energies of those two signals. We can write

$$G_{F_0} = \frac{\sum_t O(t)S_{F_0}(t)}{\sum_t S_{F_0}^2(t)}, \quad (2)$$

where $G_{F_0}$ is a gain and

$$E_{F_0} = \sum_t O(t) - S_{F_0}(t)G_{F_0}, \quad (3)$$

where $E_{F_0}$ is error for given $F_0$ and it is minimal now. The error minimization affects only the amplitude of $S_{F_0}(t)$, not the phase.

2.1.1 Error line for one frame

We can compute errors $E_{F_0}$ for fundamental frequencies in interval $F_0 \in (50\,\text{Hz}, 350\,\text{Hz})$. Errors for one frame and some number of harmonics $n \in (1, 5)$ are shown in picture 3. As the number of harmonic components in synthesized speech increases, local minima in harmonics of pitch appear. Examples of error lines for voiced (contains harmonic component) and unvoiced (only noise component) are given in figure 4. Variance of noise error line is negligible.

2.1.2 Error map for speech sample

As we can see in picture 3, there are some local minima in harmonics of pitch period (halves of fundamental frequencies). Sometimes it happens that this local minimum overrides the
Figure 2: Original speech signal $O(t)$ (black) and synthesized speech signal $S_{F_0}(t)$ (gray). Signal was synthesized from 10 harmonics. a) $S_{F_0}(t)$ for fundamental frequency $F_0 = 118Hz$ (halved $F_0$). b) $S_{F_0}(t)$ for fundamental frequency $F_0 = 200Hz$ (bad $F_0$). c) $S_{F_0}(t)$ for fundamental frequency $F_0 = 235Hz$ (correct $F_0$).

Figure 3: Error lines for signal synthesized using 1, 2, 3, 4 and 5 harmonics. "correct" minimum (for some number of harmonics). To avoid this, we sum error lines for all numbers of harmonics from 1 to 5.

$E_{F_0}^*(t) = \sum_{n=1}^{5} E_{F_0}^n(t), \quad (4)$

where $E_{F_0}^n(t)$ denotes error line for synthesized speech signal containing $n$ harmonics. A "map with valleys and ridges" originates from all $E_{F_0}^*(t)$ for all frames (see figure 5).

Figure 4: Error lines for signal synthesized using 1, ..., 5 harmonics for voiced and unvoiced signal.
Figure 5: 2D map of error. X-axis is time (frames), y-axis is fundamental frequency $F_0$. Black color means low error between original speech signal $O(t)$ and synthetized signal $S_{F_0}(t)$ (for given $F_0$), white color means high error. 1) Path with the lowest level priority. 2) Optimal path. 3) Path with the shortest priority.

2.2 Best path searching

Fundamental frequency $F_0$ can be found by looking for minimal error $\min_{F_0} \{E^{s}_{F_0}(t)\}$ of voiced frames. Sometimes minimal error can be unfortunately half of $F_0$. And we need a pitch (sampling frequency) for unvoiced parts of speech. Dynamic programming can resolve these two problems [2].

We can not select a minimum of error line $E^{s}_{F_0}$, so dynamic programming is used for the best path searching. It has two passes, a forward pass and a backward pass. In the forward pass we sum error for all possible paths. To each $F_0(t+1)$, we can go from $F'_0(t) \in (F_0(t+1) - \Delta F_0, F_0(t+1) + \Delta F_0)$ where $\Delta F_0$ is a maximal possible change of fundamental frequency (e.g. 20Hz). We select the winner $W_{F_0}(t+1)$ for each $F_0(t+1)$ and remember from which $F'_0(t)$ it comes. The criterion for winning is minimal path cost. Each winner $W_{F_0}(t+1)$ has stored sum of errors $E^{s}$ on its path and path length

$$W_{F_0}(t+1) = W_{F'_0}(t) + E^{s}_{F'_0}(t+1) + C\sqrt{(F'_0(t)-F_0(t+1))^2 + 1 \frac{E^r_{F'_0}(t) + E^r_{F_0}(t+1)}{2}}, \quad (5)$$

where:

- $W_{F_0}(t + 1)$ is path cost to actual frame $t + 1$ and fundamental frequency $F_0$
- $W_{F'_0}(t)$ is path cost to the winner (previous frame $t$) and fundamental frequency $F'_0$
- $E^{s}_{F_0}(t+1)$ is actual error (for $(t + 1, F_0)$)
- $C$ is weighting coefficient
- $\sqrt{(F'_0(t)-F_0(t+1))^2 + 1}$ is path length from $(t, F'_0)$ to $(t + 1, F_0)$
- $\frac{E^r_{F'_0}(t) + E^r_{F_0}(t+1)}{2}$ is average altitude (error)
The error function $E^s$ was normalized to interval $\langle 0, 1 \rangle$ for stable results with coefficient $C$. The coefficient $C$ adjusts weight between the lowest path and the shortest path (figure 5). The core of equation 5 is average altitude. When we are in voiced part of speech (we go in valley – the error (altitude) is low), the average altitude switches to lowest path priority. When we go on a ridge (we are in unvoiced part of speech – the error (altitude) is high), the average altitude switches to shortest path priority. This is exactly what we need, precision pitch detection and smooth straight path between voiced parts (figure 5). The backward pass starts at the end (last frame). We find the minimum winner at the end. Now we go back through the map from the winner till the start. This is the reason we have to remember from which point $(t, F'_0)$ we went to point $(t + 1, F_0)$.

### 2.3 Voiced/unvoiced decision

After having the best path through map, we must decide which parts of speech signal are voiced a which are not. The error between original speech signal $O(t)$ and synthesized speech signal $S(t)$ with fundamental frequency $F_0(t)$ is the best way. The error is related to portion of the voiced component in the speech signal. A simple threshold can be used. Frames with error lower than threshold are set voiced others are set unvoiced. Median filtering may be used for smoothing voiced/unvoiced decision function. Example can be seen in figure 6.

![Figure 6](image)

Figure 6: a) 2D map of error with the best path. b) Error run on the best path. After thresholding, we get V/UV decision function which is plotted in picture a).

### 3 Conclusion

Precision way to get fundamental frequency and voiced/unvoiced decision from sample of speech signal has been presented (see figure 7). The method has been successfully implemented and built into HNM synthesis [4] and some tools for pitch modifications (based on the HNM). The improvement was significant. At the moment, we are implementing the method for on-line pitch tracking and perform some optimizations for speed-up without the lost of quality.
Figure 7: a) speech signal, b) fundamental frequency line obtained by standard pitch tracker [3], c) smooth fundamental frequency obtained by our pitch tracker.

Acknowledgements

This research has been partially supported by Grant Agency of Czech Republic under project No. 102/02/0124 and by EC project Multi-modal meeting manager (M4), No. IST-2001-34485.

References


