REDUCING F0 FRAME ERROR OF F0 TRACKING ALGORITHMS UNDER NOISY CONDITIONS WITH AN UNVOICED/VOICED CLASSIFICATION FRONTEND

Wei Chu, Abeer Alwan

Department of Electrical Engineering
University of California, Los Angeles
Los Angeles, 90024
{weichu, alwan}@ee.ucla.edu

ABSTRACT

In this paper, we propose an F0 Frame Error (FFE) metric which combines Gross Pitch Error (GPE) and Voicing Decision Error (VDE) to objectively evaluate the performance of fundamental frequency (F0) tracking methods. A GPE-VDE curve is then developed to show the trade-off between GPE and VDE. In addition, we introduce a model-based Unvoiced/Voiced (U/V) classification frontend which can be used by any F0 tracking algorithm. In the U/V classification, we train speaker independent U/V models, and then adapt them to speaker dependent models in an unsupervised fashion. The U/V classification result is taken as a mask for F0 tracking. Experiments using the KEELE corpus with additive noise show that our statistically-based U/V classifier can reduce VDE and FFE for the pitch tracker TEMPO [1] in both white and babble noise conditions, and that minimizing FFE instead of VDE results in a reduction in error rates for a number of F0 tracking algorithms, especially in babble noise.

Index Terms— Fundamental Frequency, Pitch Tracking, Noise Robustness, Evaluation Metrics, Unvoiced/Voiced Classification

1. INTRODUCTION

Accurate fundamental frequency (F0) tracking in quiet and in noise is important for speech applications, such as speech coding, analysis, synthesis, and recognition.

Two types of error metrics are commonly used [2]. The first is Voicing Decision Error (VDE) [3]:

$$VDE = \frac{N_{V\rightarrow U} + N_{U\rightarrow V}}{N} \times 100\% \quad (1)$$

where N is the number of the frames in the utterance. The second is F0 value estimation error which is called the Gross Pitch Error (GPE):

$$GPE = \frac{N_{F0E}}{N_{VV}} \times 100\% \quad (2)$$

where N_{VV} is the number of frames which both the F0 tracker and the ground truth consider to be voiced, N_{F0E} is the number of frames for which

$$|\frac{F0_{i,estimated}}{F0_{i,reference}} - 1| > \delta\% \quad (3)$$

where i is the frame number, and δ is a threshold which is typically 20.

It is desirable for an F0 tracking algorithm to reduce the VDE and GPE at the same time. The error of an F0 tracking method is ususally presented as an error pair: (GPE, VDE). But some algorithms have low GPE, but higher VDE compared to other algorithms. We propose an error metric called the F0 Frame Error (FFE) which takes both GPE and VDE into consideration. We plot the GPE-VDE curve as a Receiver Operating Characteristics (ROC) curve to show the trade-off between GPE and VDE. With the help of the FFE and the GPE-VDE curve, we can compare the performance of F0 trackers in a unified framework.

Several F0 tracking packages: Get_F0 [4], Praat [5], TEMPO [1], and YIN [6] estimate F0 tracks reliablely when processing clean speech or speech with clear U/V boundaries [7]. When speech is processed over a noisy channel or in an office environment, however we are not guaranteed ideal clean conditions, let alone obtaining a reliable U/V mask.

Most F0-tracking algorithms make U/V decisions based on the values of energy-based or harmonic-based features exceeding certain thresholds or not. Under different noisy conditions, one has to adjust these thresholds carefully in order to avoid performance degradation. To improve the accuracy and overcome the instability of these U/V detection methods that rely on thresholds, we introduce a model-based U/V classification frontend whose output can be taken as an U/V mask for any F0 tracker. With the help of the model-based method, parameters are automatically learned and adjusted during model training and unsupervised adaptation. Reliable U/V boundary information results in improved F0 tracking.

2. FFE AND GPE-VDE CURVE

Consider F0 tracking on an utterance of N frames shown in Fig. 1 where the F0 values are set to be 0 Hz. When the tracked F0 contour is compared to the ground truth, there can only exist 3 possible types of error in any frame i:

- U→V Error: an unvoiced frame is classified as a voiced frame;
- V→U Error: a voiced frame is classified as an unvoiced frame;

Supported in part by NSF

Table 1. Phonemes and Sounds to U and V Dictionary

	Stops	Affricates and Fricatives	Nasals and Vowels	Semivowels and Glides	Others
U	p(cl) t(cl) k(cl) bcl dcl gcl q	ch s f th sh	-	hh	epi h pau
V	b d g dx	jh z v zh dh	m n ng em en eng nx iy ih eh ey ae aa aw ay ah ao oy ow uh uw ux er ax ix axr ax-h	l r el w y hv	-

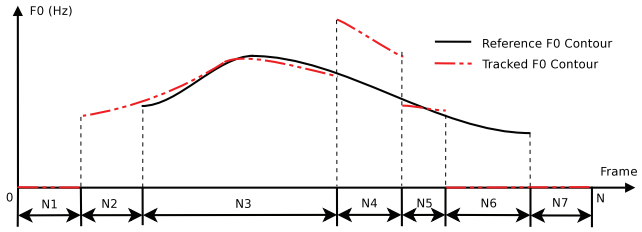

Fig. 1. F0 Tracking Contour over Time for an utterance of N frames

- F0 Value Estimation Error:

$$|F0_{i,estimated}/F0_{i,reference} - 1| > \delta\%.$$

In Fig. 1, the F0 tracker made U→V errors over N_2 frames, F0 value estimation errors over N_4 frames, and V→U errors over N_6 frames. We propose an F0 Frame Error (FFE) metric which sums the three types of errors mentioned above:

$$\begin{aligned} FFE &= \frac{\text{\# of error frames}}{\text{\# of total frames}} \times 100\% \\ &= \frac{N_{U\to V} + N_{V\to U} + N_{F0E}}{N} \times 100\%. \end{aligned} \quad (4)$$

FFE is also a combination of GPE and VDE:

$$\begin{aligned} FFE &= \frac{N_{F0E}}{N} \times 100\% + \frac{N_{U\to V} + N_{V\to U}}{N} \times 100\%. \\ &= \frac{N_{VV}}{N} \times GPE + VDE \end{aligned} \quad (5)$$

Therefore, FFE takes both GPE and VDE into consideration making the comparison of different F0 trackers possible.

A GPE-VDE curve is effective in showing the relationship between GPE and VDE. When tweaking the parameters of the F0 tracker, we can obtain a set of (GPE, VDE) pairs. $(\text{GPE}_i, \text{VDE}_i)$ is a minimum point if and only if there exists no j that satisfies $\text{GPE}_j < \text{GPE}_i$ and $\text{VDE}_j < \text{VDE}_i$ at the same time. When plotting all the minimum points, we can obtain a GPE-VDE curve.

3. UNVOICED/VOICED CLASSIFICATION

Since VDE is part of the FFE, in this section, we focus on developing a robust model-based U/V classification frontend in order to reduce the VDE.

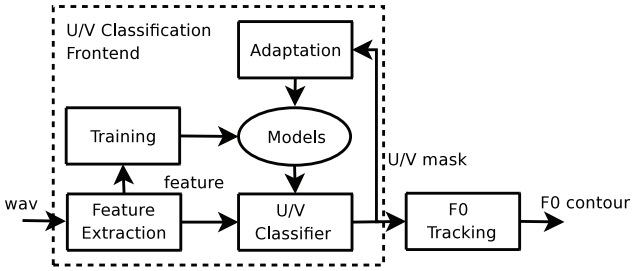

Fig. 2. U/V Classification Frontend for F0 Trackers

There have been several model-based techniques for Voice Activity Detection (VAD) [8] [9] [10], but they primarily distinguish voiced frames from unvoiced frames.

The flowchart of the proposed U/V classifier and its relationship to the subsequent F0 tracker are shown in Fig. 2.

The structure of our U/V classifier is similar to the common Hidden Markov Model (HMM) based phone recognizer in the maximum likelihood based model training and Viterbi decoding. The setting and performance of the classifier will be presented in the experimental section. In the following, we introduce the acoustic modeling of the U/V models and unsupervised speaker adaptation.

3.1. Unvoiced/Voiced Acoustic Modeling

Two acoustic models were trained, one for unvoiced sounds (U) and the other for voiced sounds (V). The mapping from sounds to U and V is shown in Table 1. The phone symbols appeared in the table are used in the TIMIT phone level transcription. 'pau' is a pause, 'epi' is an epenthetic silence, 'h#' is the begin/end marker (non-speech events).

The U/V models are left-to-right HMMs with 3 emitting states, and 256 Gaussian components per mixture model. A word net containing unvoiced and voiced nodes with a bigram language model attached to the directed arcs between the nodes was constructed. The U/V decision can be adjusted by tuning the language model. For example, increasing $P(voiced)$ or $P(voiced|unvoiced)$ would make the decoder prone to making more voiced hypotheses.

3.2. Unsupervised Speaker Adaptation

It is difficult for the speaker independent (SI) models trained on American English corpus (TIMIT) to accurately depict the distribution of unseen data - the test set (KEELE) composed of different British English speakers. Therefore, we apply offline unsupervised Maximum Likelihood Linear Regression (MLLR) speaker adaptation to adapt the initial SI models to speaker dependent (SD) models [11]. In SD model adaptation for speaker s, a global transformation style can apply a linear transformation $\mathbf{W}_s$ to all the mean vectors of the Gaussians: $\hat{\mu}_s = \mathbf{W}_s \mu_s$. A regression class tree based

Table 2. VDE of the U/V Classifier Using the KEELE Corpus (SNR = 0 dB, **SI**: speaker independent models, **GSD/RSD**: global style/regression tree style adapted models, **MFCC** and **ETSI** are the features used in the classifier)

VDE	White Noise		Babble Noise	
	MFCC	ETSI	MFCC	ETSI
SI	11.57%	10.84%	30.70%	26.27%
GSD	10.98%	9.81%	27.61%	**22.48%**
RSD	**10.18%**	**9.14%**	**27.23%**	23.54%

transformation style uses a binary regression tree to decide whether μ_s^i of node i should be adapted by a separate transformations $\mathbf{W}_s^i$ or a same global transformation $\mathbf{W}_s$. In the U/V classification task, the regression tree is composed of a base node connected to two leaves which are unvoiced and voiced nodes. For a speaker s, the global style adaptation uses all the data to train a global transformation $\mathbf{W}_s$, Regression tree based adaptation needs to use the decoding results to attach the data to leaf node i, and then use the attached data to train a transformation $\mathbf{W}_s^i$ for the leaf node i.

4. EXPERIMENTS

In this section, we compare the FFE, GPE-VDE curve and the traditional (GPE, VDE) pair using the KEELE corpus [12]. The corpus contains a simultaneous recording of speech and laryngograph signals for a phonetically-balanced text which was read by 5 male and 5 female speakers. The total length of the recoding is 5 min 37 s.

The SI U/V models are trained from the TIMIT training corpus (approximately 4 hours). For feature extraction, both Mel-Frequency Cepstral Coefficients (MFCCs) and the ETSI [13] frontend are used. ETSI features are more noise robust for the Aurora 2 task [14] than MFCCs.

Noise is artificially added to both TIMIT and KEELE corpora. To test the robustness of the F0 tracker under different noise conditions, the program FaNT [14] was used to employ white and babble noise segments from the NOISEX92 [15] corpus to corrupt the speech to a Signal-to-Noise Ratio (SNR) of 0 dB.

Table 2 shows the VDE of the proposed model-based U/V classifier with different features before and after adaptation. Unsupervised speaker adaptation is effective in minimizing the mismatch between training and test data. ETSI features are always better than MFCC features before and after adaptation. For the white noise cases, the VDE of the regression class tree based adaptation (RSD) is lower than that of global adaptation (GSD). In the babble noise case, the GSD resulted in slightly better performance for ETSI features. This could be because babble noise is more correlated with the underlying speech signal than white noise is.

The U/V classification result was then used as a mask for F0 trackers. Since Get_F0 and Praat do not have the option of directly using an U/V mask, the effect of the mask is only tested on TEMPO. The U/V decoder using ETSI features and SD models is used for both noise conditions. To take advantage of the decoder that has the lowest VDE, regression tree style adaptation is used under white noise, but global style adaptation is used under babble noise.

For each F0 tracking package, 500 - 1000 configurations are tested where different parameters are adjusted (e.g., the correlation window length, voicing thresholds). The performance of the F0 tracker under each configuration corresponds to certain values for GPE, VDE, and FFE as shown in Table 3. 'M+' denotes

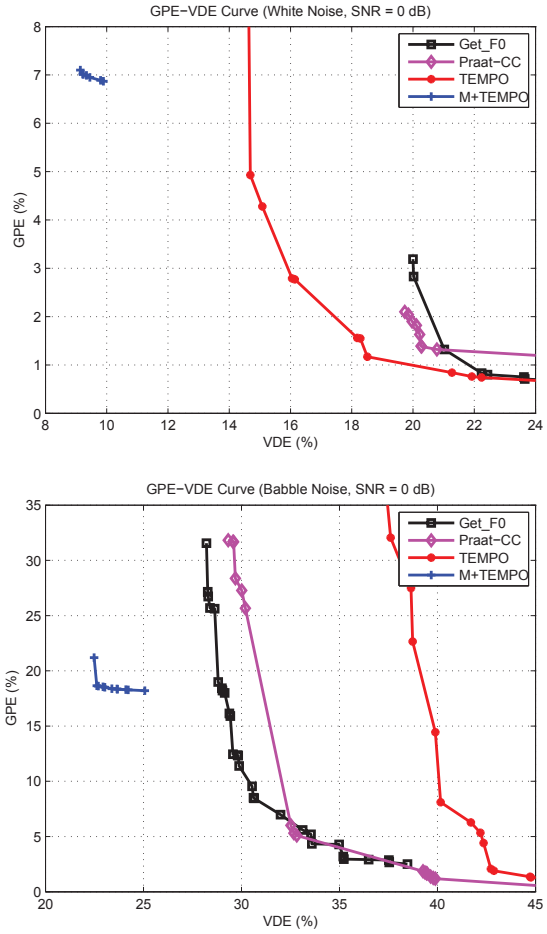

Fig. 3. GPE-VDE Curve (**M+**: using U/V classifier output as a mask)

the U/V mask by the model-based classifier. In white and babble noise, the lowest GPE is achieved by Praat, and the lowest VDE by M+TEMPO. Note that minimizing the FFE results in a significant reduction in GPE. Take TEMPO in white noise for example, when we shift our objective from minimizing the VDE to FFE, the VDE slightly increases from 14.52% to 14.69%, but the GPE significantly decreases from 15.87% to 4.93%. That is also true for Get_F0, Praat, and TEMPO in babble noise. Compared to TEMPO, the FFE of M+TEMPO drops by 24.4% in white noise, and 27.6% in babble noise. It could be inferred that only minimizing the VDE can not guarantee the minimization of the overall FFE, but reducing VDE is helpful for lowering the FFE. Note that the GPE for M+TEMPO is higher than TEMPO when minimizing the FFE.

In the GPE-VDE curve shown in Fig. 3, it can be observed that for every F0 tracker without the U/V mask, GPE decreases when VDE increases. As shown in Eq. 1 and 2, when the VDE increases, it may be due to an increase in the V→U errors resulting in a reduction in N_{VV}. Although the N_{VV} decreases, the N_{F0E} decreases more, for it is easier to estimate the F0 value over the remaining voiced frames with a higher SNR. Since the ratio of N_{F0E} to N_{VV} decreases, the GPE decreases. Take TEMPO in white noise for example, when the VDE increases from 14.69% to 21.92%, the V→U

Table 3. GPE, VDE and FFE for the KEELE Corpus (SNR = 0 dB, **M+**: U/V mask provided by model-based classifier, **min VDE/FFE**: when VDE/FFE is minimized)

		White Noise			Babble Noise		
		GPE	VDE	FFE	GPE	VDE	FFE
Get_F0	min VDE	3.19%	20.00%	21.04%	31.56%	28.21%	37.58%
	min FFE	2.83%	20.02%	20.94%	8.51%	30.65%	32.79%
Praat	min VDE	**2.10%**	19.72%	20.41%	31.82%	29.32%	38.69%
	min FFE	**2.10%**	19.72%	20.41%	**5.31**%	32.67%	33.86%
TEMPO	min VDE	15.87%	14.52%	20.59%	58.05%	36.51%	50.35%
	min FFE	4.93%	14.69%	16.56%	8.11%	40.16%	41.24%
M+TEMPO	min VDE	7.10%	**9.14%**	**12.52%**	18.65%	**22.48%**	**29.86%**
	min FFE	7.10%	**9.14%**	**12.52%**	18.65%	**22.48%**	**29.86%**

error rate increases from to 27.05% to 41.60%, the U→V error rates shift from 1.25% to 0.50%, and the GPE decreases from 4.93% to 0.76%. But for F0 trackers with U/V masks, the VDE is more stable, and the GPE does not change much. Because the F0 tracker has to estimate F0 for every voiced frame indicated by the mask, even if it is a frame with a low SNR. Take M+TEMPO in white noise for example, when the VDE increases from 9.14% to 9.89%, the V→U error rate increases from to 8.60% to 10.63%, the U→V error rate decrease from 9.73% to 9.08%, the GPE slightly decreases from 7.10% to 6.87%.

It is also shown in Fig. 3 that integrating our model-based U/V classifier into an F0-tracking algorithm can improve its voicing decision accuracy. Take TEMPO and M+TEMPO in white noise for example, after applying the U/V mask, the minimum VDE decreases from 14.52% to 9.14%.

5. CONCLUSIONS

The F0 Frame Error (FFE) and GPE-VDE curve can be used to evaluate the F0 tracking algorithms in a unified framework. The model-based U/V classifier can output robust U/V masks for F0 trackers under both white and babble noise conditions which is helpful for reducing the overall FFE. Minimizing the FFE is more effective than minimizing the VDE alone. Future work will focus on ways of reducing both GPE and VDE for F0 tracking algorithms.

6. ACKNOWLEDGMENT

The authors would thank Hideki Kawahara for providing the TEMPO package, and Georg Meyer for providing the KEELE corpus.

7. REFERENCES

[1] H. Kawahara, H. Katayose, A de Chevigne, and R. Patterson, "Fixed point analysis of frequency to instantaneous frequency mapping for accurate estimation of f0 and periodicity," in *Proc. of EUROSPEECH*, 1999, vol. 6, pp. 2781–2784.

[2] L. Rabiner, M. Cheng, A. Rosenberg, and C. McGonegal, "A comparative performance study of several pitch detection algorithms," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 24, no. 5, pp. 399–418, 1976.

[3] T. Nakatani, S. Amano, T. Irino, K. Ishizuka, and T. Kondo, "A method for fundamental frequency estimation and voicing decision: Application to infant utterances recorded in real acoustical environments," *Speech Communication*, vol. 50, no. 3, pp. 203–214, 2008.

[4] D. Talkin, "Robust algorithm for pitch tracking," *Speech Coding and Synthesis*, pp. 497–518, 1995.

[5] P. Boersma, "Praat, a system for doing phonetics by computer," *Glot International*, vol. 5, no. 9/10, pp. 341–345, 2001.

[6] A. de Cheveigne and H. Kawahara, "Yin, a fundamental frequency estimator for speech and music," *The Journal of the Acoustical Society of America*, vol. 111, no. 4, pp. 1917–1930, 2002.

[7] A de Chevigne and H. Kawahara, "Comparative evaluation of F0 estimation algorithms," in *Proc. of EUROSPEECH*, 2001, pp. 2451–2454.

[8] J. Sohn, N. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Processing Letters*, vol. 6, no. 1, pp. 1–3, 1999.

[9] Y. D. Cho and A. Kondoz, "Analysis and improvement of a statistical model-based voice activity detector," *IEEE Signal Processing Letters*, vol. 8, no. 10, pp. 276–278, 2001.

[10] J.-H. Chang, N. S. Kim, and S. K. Mitra, "Voice activity detection based on multiple statistical models," *IEEE Trans. on Signal Processing*, vol. 54, no. 6, pp. 1965–1976, 2006.

[11] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer Speech and Language*, vol. 9, no. 2, pp. 171–185, 1995.

[12] F. Plante, G. Meyer, and W. A. Ainsworth, "A pitch extraction reference database," in *Proc. of EUROSPEECH*, 1995, pp. 837–840.

[13] ETSI ES 202 050 recommendation, "Speech processing, transmission and quality aspects (stq); distributed speech recognition; advanced front-end feature extraction algorithm; compression algorithms," 2007.

[14] H. Hirsch and D. Pearce, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *Proc. of ASR2000*, 2000, pp. 181–188.

[15] A.P. Varga, H. J. M. Steeneken, M. Tomlinson, and D. Jones, "The NOISEX-92 study on the effect of additive noise on automatic speech recognition," in *Technical report, DRA Speech Research Unit*, 1992.