Combining the Glottal Mixture Model (GLOMM) with UBM for Speaker Recognition

Paul M. Baggenstoss
Fraunhofer FKIE, Fraunhoferstrasse 20
53343 Wachtberg, Germany
Email: p.m.baggenstoss@ieee.org

Abstract—**We present an iterative algorithm to extract the voice source waveform from recordings of speech for speaker identification. The method detects glottal closings, then constructs a speaker-dependent library of glottal pulse waveforms by clustering data windows centered on the linear prediction error time-series at the glottal closures. With the voice source modeled as scaled and shifted glottal pulses, the algorithm iteratively determines the vocal tract parameters in each frame. In experiments, we combine the extracted voice source information with a universal background model (UBM). Using the TIMIT data corpus and a 200-speaker population size, we demonstrate a factor of three speaker recognition error reduction.**

I. Introduction and Previous Work

A. Voice source information for speaker recognition

Most recent work in speaker recognition relies on front-end processing that extracts short-time spectral information such as MFCCs [1]. These features were developed for speech recognition and are not optimized to capture the speaker-dependent voice source information. Attempts to add voice source information back into speaker-recognition systems to improve them have met limited success [1], [2], [3], probably due to the difficulty of estimating the voice source waveform itself.

The most basic principle in voice source estimation is linear prediction [3]. The linear predictive coding (LPC) coefficients are readily estimated using classical methods. In addition to providing a good approximation to the vocal tract filter (VTF), the prediction error waveform is a first-order approximation to the voice source waveform (VSW), which approximates the derivative of the glottal flow [2]. There are, however, dependencies that are difficult to disentangle. LPC is estimated based on the spectrum containing the product of VSW and VTF spectra. The effects of VSW contamination in LPC can be removed by seeking to estimate LPC only during the time that the glottis is closed [2]. This is also subject to error since the method relies on estimating the glottal closed phase, and can fail if the vocal folds do not close completely or quickly.

B. Proposed Approach: Glottal Mixture Model (GLOMM)

Instead of attempting to reconstruct the VSW accurately, we define the VSW more loosely as all speaker-dependent effects that cannot be attributed to the all-pole filter and that repeat with each glottal closure (every pitch period). We then construct the VSW for a given speaker from scaled and shifted glottal pulses from a speaker-dependent glottal pulse library. Knowledge of the speaker's glottal pulse library can then be used to augment speaker-regognition statistics based on short-time spectral information. Terminology used in this paper is tabulated below:

LPETS	**Linear prediction error time-series** obtained by LPC.
n/a	**Glottal closure.** Detected peak in the LPETS assumed to be caused by closure of the glottis.
n/a	**Glottal data window**. A length $2Q+1$ window centered on the LPETS precisely at the time of the detected glottal closure.
n/a	**Glottal pulse**. One of the library of fixed waveforms attributed to a given speaker's voice production. They are obtained by clustering glottal data windows.
VSW	**Voice Source Waveform**. The voice driving function, assuming vocal tract is all-pole filter.
n/a	**Synthetic VSW**. Estimate of the VSW made up of scaled and shifted glottal pulses.

II. Algorithm Description

A. Linear Prediction-Error Time-Series (LPETS)

Assume we have data from multiple speakers, consisting of multiple utterances of each speaker. Let the speech data be divided into 50% overlapped Hanning-weigthed frames of size $N_{\rm FFT}$ samples. Independently in each frame, we estimate LPC of order P using classical methods (auto-correlation followed by Levinson-Durbin). Linear prediction coding (LPC) is a widely-used approach in speech and time-series analysis. It is well known that the linear prediction error time-series (LPETS) is a first-order approximation to the voice-source waveform, i.e. the vocal tract input waveform when the vocal tract is seen as an all-pole filter [3]. The LPETS is obtained in each frame in the frequency-domain by multiplying the DFT of the input data by the DFT of the prediction-error filter, then taking the inverse DFT.

Glottal closures are marked by a sharp increase in prediction error. To more robustly detect the glottal closing instant, we use hilbert envelope of LPETS [4]. So, prior to the final inverse-DFT, we zero the DFT bins above Nyquist so that the LPETS will be complex (analytic). Using overlap-add, the frames of the analytic LPETS are combined to re-construct the full analytic LPETS of each utterance. An example is shown in Figure 1, which shows the magnitude of the analytic LPETS.

B. Glottal closure detection

Glottal closures are detected by peak detection of the magnitude-analytic LPETS. Using a median filter, a time-

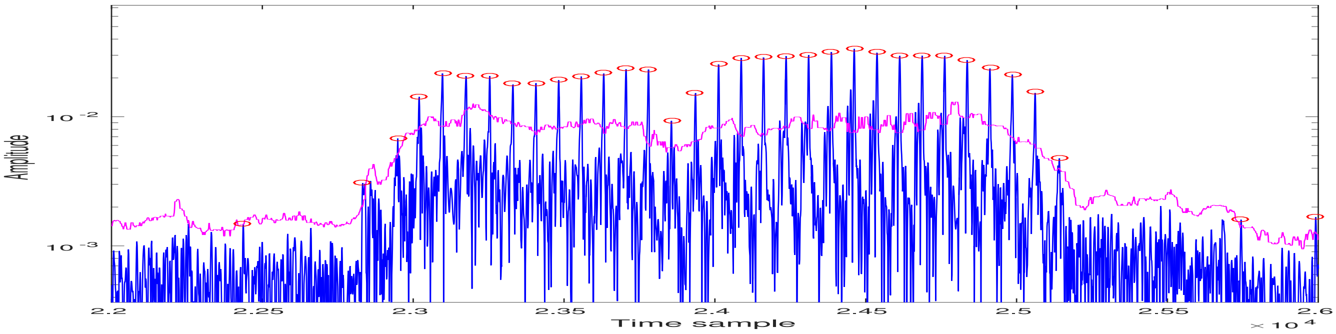

Fig. 1. Dark blue: The magnitude of the analytic LPETS for a portion of an utterance reconstructed using overlap-add from individual frames. Pink: the time-varying median-filtered threshold. Red circles: detected glottal closures.

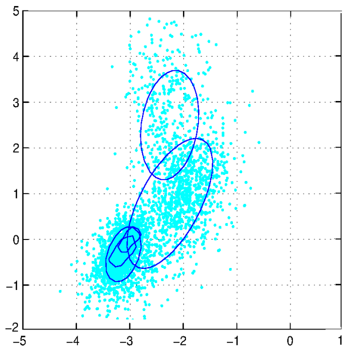

Fig. 2. Illustration of clustering in the feature space using all available training data for one speaker.

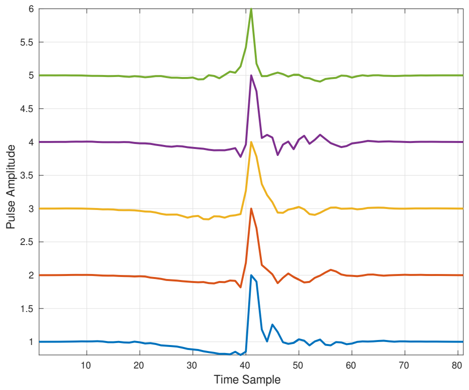

Fig. 3. The glottal pulse library for TIMIT speaker “msfh1”.

varying peak-detection threshold is formed (magenta line in Figure 1). Local maxima exceeding the threshold are then detected (red circles in Figure 1). These are the approximate times of the glottal closures. To get the most accurate estimate of the glottal closure, we used parabolic interpolation to estimate the precise peak time with sub-sample accuracy.

C. Glottal Data Windows and Clustering

For each detected glottal closure, we extracted a glottal data window of length $2Q + 1$, from the analytic LPETS, where Q is the glottal data window half-length. We used a value of Q giving a glottal data window length of 0.01 sec for both male and female speakers. Since we have estimated the peak time with sub-sample accuracy, the glottal data window can be sub-sample time-shifted (in the frequency domain) so that the peak occurs at exactly the $Q + 1$-th sample of the glottal data window. It is well known that a speaker’s VSW is significantly influenced by emotional factors, stress, and loudness. Therefore, to estimate the VSW, we cluster the glottal data windows before averaging to obtain a library of potential glottal pulse waveforms. Separately for each speaker, we estimated the glottal pulse library as follows: (a) We collected glottal data windows that were time-corrected and normalized to have a peak magnitide value of 1 exactly at sample $Q+1$. (b) We performed principal component analysis (PCA) to remove noise from the collected pulses, taking the largest $D = 30$ singular vectors. (c) A Gaussian mixture of $M = 5$ components was then used to cluster the data in the D-dimensional space formed by projecting the glottal data windows onto the singular vectors. The clustering itself on the feature space is visualized in Figure 2 created by projecting the glottal data windows onto the first two singular vectors. It is clear that natural clusters have formed in the scatter diagram and that the Gaussian mixture has properly identified some clusters. The center of each cluster is defined by the 30-dimensional mean of the corresponding Gaussian mixture component. The cluster centers are then projected back to the original window length by multiplying by the basis set of singular vector, forming the library of glottal pulses. The resulting library of glottal pulses is illustrated in Figure 3.

D. Synthetic VSW

The next step is to construct a synthetic VSW using the speaker’s library of glottal waveforms. Each glottal data window is “classified” using the Gaussian mixture model as one of the M library pulses. A synthetic VSW is then formed by scaling and time-shifing the library pulses to correspond to the amplitude and time of each detected glottal pulse. The process is illustrated in Figure 4. On the top of the figure, we see a section of the LPETS (real part). We have indicated with red circles the location of detected glottal closures, and placed a number indicating which glottal library pulse best matches. In the lower graph is the synthetic VSW formed from delayed and scaled glottal library pulses from the speaker’s library. Note that not only are the glottal closures detected, but also

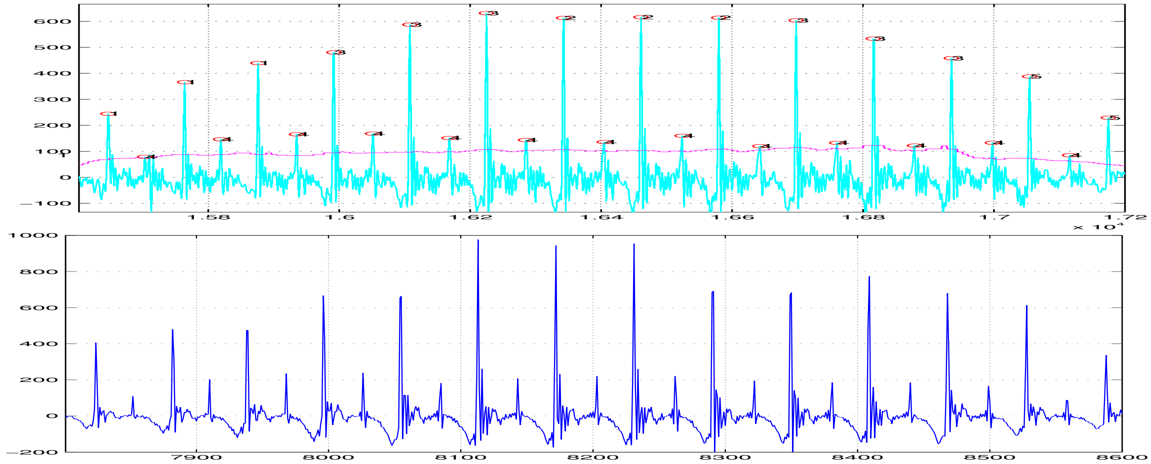

Fig. 4. Example of synthetic VSW. Top: Section of LPETS (real part) reconstructed from from multiple frames. Bottom: synthetic VSW. This plot was made after the tenth iteration of the GLOMM algorithm.

the error peaks at the period half-times, which can occur due to laryngealization or glottal opening [4].

E. LPC re-estimation

Next, the synthetic VSW is re-segmented into 50% overlapped Hanning-weighted frames. In Figure 5 (bottom) we see a hanning-weighted VSW frame created from VSW in Figure 4. This produces a synthetic VSW for each frame. Then, to complete the GLOMM algorithm, the LPC coefficients are re-estimated independently in each frame. But instead of using the classical ACF/Levinson method, the LPC coefficients are estimating by optimization of the fit between the input data time-series in the frame and the synthetic time-series produced by passing the synthetic VSW through the all-pole filter corresponding to the LPC coefficients.

We now mathematically describe the LPC re-estimation approach. Let $\mathbf{x}$ be the un-weighted input data of the frame. Let function $h(\,)$ represent the Hanning-weighting operation. Thus, $h(\mathbf{x})$ is the Hanning-weighted input data. Let $\mathbf{u}$ be the corresponding frame of the synthetic source waveform that we have constructed and is assumed to be fixed. Let $\mathbf{y}$ be the vocal-tract filtered synthetic source waveform. In the time domain, the vocal tract filtering is

$$y_t = \sum_{i=1}^{P} a_i y_{t-i} + u_i. \tag{1}$$

This operation is much easier to do in the frequency domain. We have

$$Y_k = U_k / A_k, \quad 1 \leq k \leq N_{\mathrm{FFT}},$$

where the capital letter quantities are the DFT coefficients of the corresponding lower-case quantities. The Hanning-weighted $\mathbf{y}$, i.e. $h(\mathbf{y})$, should closely approximate $h(\mathbf{x})$ up to an unknown scale factor. If we model the approximation error between $h(\mathbf{x})$ and $h(\mathbf{y})$ as Gaussian, we obtain the Gaussian distribution:

$$\begin{aligned} L(\mathbf{x}; \mathbf{a}, \sigma^2, c) &= (2\pi)^{-N/2} \det(\mathbf{R})^{-1/2} \\ &\cdot \exp\left\{-\tfrac{1}{2}\left[h(\mathbf{x}) - c\, h(\mathbf{y})\right]' \mathbf{R}^{-1} \left[h(\mathbf{x}) - c\, h(\mathbf{y})\right]\right\} \end{aligned} \tag{2}$$

where c is a scale factor and covariance matrix $\mathbf{R}$ conforms to the autoregressive process defined by $\mathbf{a}$ and σ^2. We call this a *dual-mode model* because the LPC coefficients "$\mathbf{a}$" enter into the model twice, first coherently through (1) as a waveform filter, and second incoherently through covariance matrix $\mathbf{R}$, which is the theoretical covariance of the autoregressive process corresponding to coefficients $\mathbf{a}$. In effect, this assumes all the voice energy passes through the vocal tract. This assumption does not hurt, even when there is no voiced speech, and no detected glottal closures, because the LPC parameters are not used for speaker identification.

The LPC coefficients in each frame are re-estimated by maximizing (2) over the LPC coefficients. And, finally, the GLOMM algorithm repeats with formation of the LPETS (Section II-A) using the updated LPC coefficients. Typically the algorithm stops changing significantly after about three to five iterations. After several iterations, the waveform match between $h(\mathbf{x})$ and $h(\mathbf{y})$ can be strikingly good. In Figure 5 (top), we see $h(\mathbf{x})$ and $h(\mathbf{y})$ overlaid ($h(\mathbf{y})$ is shown in darker color). The match is very close. Equally impressive is the match between the synthetic VSW and the LPETS Figure 5 (bottom).

The statistical model (2) can be used not just to estimate the LPC coefficients, but can also serve as a likelihood function. Let the total log-likelihood be

$$S = \sum_k \log L(\mathbf{x}_k; \hat{\mathbf{a}}_k, \hat{\sigma}_k^2, \hat{c}_k), \tag{3}$$

where k ranges over the available data segments for the speaker, and $\hat{\mathbf{a}}_k, \hat{\sigma}_k^2, \hat{c}_k$ are the frame-dependent parameter estimates. Intuitively, S is the measure of fit between the data and the GLOMM model, which assumes the data is generated by glottal pulses from the given library, passed through an all-pole filter. As the algorithm iterates, the glottal pulse library improves and S generally increases.

F. GLOMM Algorithm Summary

To initialize GLOMM, the data for a given speaker is segmented into frames and the LPC coefficients are obtained

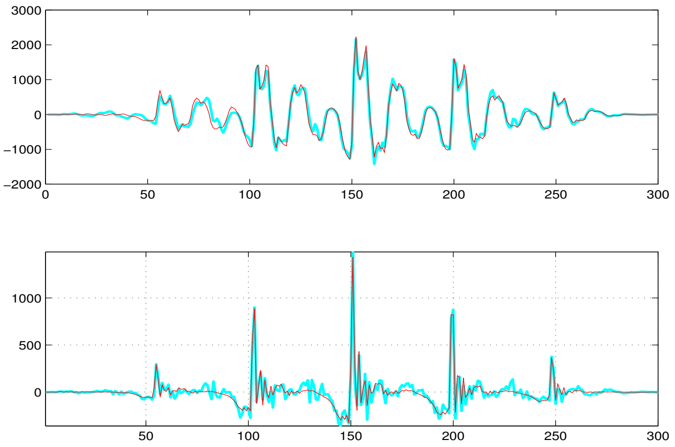

Fig. 5. Top: input data $h(\mathbf{x})$ and $h(\mathbf{y})$ (darker color). Bottom: LPETS and synthetic VSW (darker color), after 10 GLOMM iterations.

separately in each frame using classical methods. In each iteration, we (a) compute the LPETS, (b) detect glottal closures, (c) collect glottal data windows, (d) cluster the glottal data windows to obtain the glottal pulse library, (e) construct the synthetic VSW, (f) segment the synthetic VSW, then (g) re-estimate the LPC in each frame.

G. Speaker-Identification Method

After several iterations of GLOMM, we discard the LPC parameters (which is stored for each data frame). We keep only the glottal pulse library for each speaker. Now, assume we are given a test utterance for the purpose of identifying the true speaker. There are two potential ways to use the glottal pulse libraries to identify a speaker in the "closed" classification problem (where each candidate speaker is known).

In the first method (total likelihood), we run the GLOMM algorithm a few iterations using the candidate speaker's glottal pulse library to determine total likelihood S - when iterating, the glottal pulse library is held fixed and not updated. S is then used as the classifier likelihood statistic.

The second method (glottal pulse matching) is to run GLOMM to extract a glottal pulse library from the test utterance. This glottal pulse library is discarded, but the extracted glottal data windows $\tilde{\mathbf{w}}$ (See section II-C) are kept and matched to the glottal pulse library of each candidate speaker using the GMM.

III. Universal Background Model (UBM)

Our goal is to augment existing speaker-ID methods with additional voice source information. For a state of the art "existing method", we used the universal background model (UBM) [5]. Although the I-vector approach [6] has replaced UBM as state of the art, this is primarily due to better performance of I-vector approaches in the presence of varying recording methods and environments [7]. Since our initial experiments will be conducted using stable recording method and environment, we are justified in using UBM as state of the art.

UBM uses a Gaussian mixture model (GMM) trained on all available speakers. The GMM is trained using a "bag of features" approach where the features (typically MFCC) extracted from short-time Fourier transform analysis of the available utterances are trained without time ordering. Let $\mathbf{z}$ be a feature vector, typically of dimension 19 for MFCC, 38 for MFCC+Δ. The GMM is written

$$p(\mathbf{z};\mathbf{\Lambda}) = \sum_{i=1}^{M} \alpha_i \mathcal{N}(\mathbf{z};\boldsymbol{\mu}_i,\mathbf{\Sigma}_i), \tag{4}$$

where $\mathcal{N}(\mathbf{z};\boldsymbol{\mu}_i,\mathbf{\Sigma}_i)$ is the Gaussian kernel

$$\mathcal{N}(\mathbf{z};\boldsymbol{\mu},\mathbf{\Sigma}) = \frac{(2\pi)^{-N/2}}{|\mathbf{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{z}-\boldsymbol{\mu})'\mathbf{\Sigma}^{-1}(\mathbf{z}-\boldsymbol{\mu})\right\}, \tag{5}$$

and $\mathbf{\Lambda}$ is the collection of UBM parameters

$$\mathbf{\Lambda} = \{\alpha_i, \boldsymbol{\mu}_i, \mathbf{\Sigma}_i,\ \forall i\}.$$

Generally, $\mathbf{\Sigma}_i$ are diagonal covariance matrices. This is called the *universal* background model because it is trained on all speakers. In the process called enrollment, a speaker-dependent GMM is obtained by adapting the UBM to the data of a given speaker [5]. To classify an utterance $[\mathbf{z}_1, \mathbf{z}_2 \ldots \mathbf{z}_K]$ using the UBM (4), we simply apply the maximum likelihood rule:

$$\hat{s} = \arg\max_{s} \left\{ \sum_{k=1}^{K} \log p(\mathbf{z}_k;\hat{\mathbf{\Lambda}}_s) \right\} \tag{6}$$

where $\hat{\mathbf{\Lambda}}_s$ are the speaker-adapted GMMs (by adapting only the means).

IV. Speaker-Identification Experiments

A. TIMIT Data

The TIMIT speech recognition corpus consists of 630 male and female speakers, each having 10 utterances, averaging about 3 seconds each, and divided into eight "SX" and "SI" utterances and two "SA" utterances. In the speaker identification experiments, we trained on all eight "SX" and "SI" utterances. In each experiment, we selected 200 speakers at random. Then, for each speaker, we selected one of the "SA" utterances at random and classified it against the closed set of 200 speakers. We performed just 2 independent 200 × 200 experiments for a total of 400 individual classification decisions. We used no voice activity dectection, always using the complete utterances.

1) Data pre-processing: The raw TIMIT data is sampled at 16kHz and stored as signed 16-bit quantized data, which is read as a real number in the range [-1,1]. To this data is added independent standard Gaussian noise at three standard deviations: $\sigma \in [0,\ .002,\ .0026, .004$, giving an average SNR of ∞, 16.5 dB, 13.5, and 10.5 dB, respectively. We measured SNR as the total energy in the noise-free utterance divided by the total added noise energy. After adding noise, the data was down-sampled 2:1 to a sample rate of 8 kHz. Noise was added to both training and testing data.

2) UBM implementation: We used an $M = 100$-mixture UBM, trained on all speakers in the training set for all experiments. We used HTK [8] to compute the 19-dimensional MFCC features using a 25 millisecond window with 10 millisecond frame rate. The HTK configuration parameters are TARGETRATE = 100000, WINDOWSIZE = 250000, PREEMCOEF = 0.96, CEPLIFTER = 22, NUMCHANS = 20, NUMCEPS = 19, DELTAWINDOW = 3, ENORMALISE = F, SOURCERATE=1250, SAVECOMPRESSED = T, SAVEWITHCRC = F, USEHAMMING = T, TARGETKIND = MFCC .

3) Hybrid Classifier: As a hybrid GLOMM-UBM classifier, we formed the combined statistic

$$L_s = w_1 L_s^U + w_2 L_s^G + w_3 S_s, \tag{7}$$

where L_s^U is the UBM statistic $L_s^U = \sum_{k=1}^{K} \log p(\mathbf{z}_k; \hat{\mathbf{\Lambda}}_s)$ computed using the speaker-adapted parameters $\hat{\mathbf{\Lambda}}_s$, S_s is the GLOMM total likelihood S_s (equation 3) computed using the glottal pulse library for speaker s, and L_s^G is the total log-likelihood for the extracted glottal data windows using a Gaussian mixture formed from the glottal pulse library of speaker s (See section II-G).

4) Results: With $w_1 = 1$, we optimized the parameters $w2, w3$. The values of $w_2 = .025, w_3 = .43$ were used at all SNRs. In Figure 6, we show the error for the mixture (7) as a function of SNR. Also shown are the individual error performances for UBM only ($w_2 = w_3 = 0$), GLOMM only ($w_1 = 0$). The clean speech case (infinite SNR) is plotted on the X-axis at SNR=35. For clean speech, we measured just one error for GLOMM+UBM in random 400 trials. It is remarkable that GLOMM alone achieves about the same performance as UBM, when GLOMM clearly discards most of the spectral information (LPC is used only to extract VSW). The significant reduction in error when combining GLOMM and UBM (a factor of 3 or more) attests to the independent speaker identity information contained in the VSW.

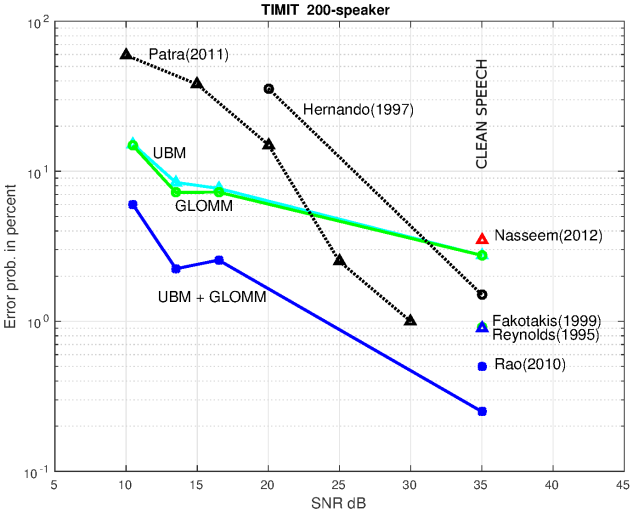

Fig. 6. Performance as a function of SNR - Clean speech plotted at 35 dB.

We found a number of comparative results in the literature for the 200-speaker experiment [9], [10], [11], [12], [13], [3], [14], which we plotted in Figure 6 as bibliography citations. Some slight differences exist in experimental setups. None of them downsample the data to 16 kHz as we do. In [10], an even male/female split is used and training data is clean. In [9], training data is clean. Nossair [13] used 7 training utterances (not 8). Reynolds [14] used a population size of 168 speakers.

V. Conclusions

In this paper, we have presented a means of extracting voice source information from a given speaker as a library of glottal pulses. Using just this glottal pulse library, we have demonstrated comparable speaker-ID performance to UBM in a 200-speaker experiment from TIMIT corpus. However, when combined with UBM, a factor of three reduction in error is demonstrated.

References

[1] J. Gonzalez-Rodriguez, “Evaluating automatic speaker recognition systems: An overview of the nist speaker recognition evaluations (1996-2014),” *Loquens*, vol. 1, no. 1, 2014.

[2] M. Plumpe, T. Quatieri, and D. Reynolds, “Modeling of the glottal flow derivative waveform with application to speaker identification,” *Speech and Audio Processing, IEEE Transactions on*, vol. 7, pp. 569–586, Sep 1999.

[3] R. R. Rao, V. K. Prasad, and A. Nagesh, “Performance evaluation of statistical approaches for text- independent speaker recognition using source feature,” *Computer Science and Networking*, vol. 2, pp. 8–13, Aug 2010.

[4] T. Drugman and T. Dutoit, “Glottal closure and opening instant detection from speech signals,” in *INTERSPEECH 2009*, 2009.

[5] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, “Speaker verification using adapted gaussian mixture models,” *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[6] N. Dehak, P. Kenny, R. Dehak, and P. Ouellet, “Front-end factor analysis for speaker verification,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, pp. 788–798, May 2011.

[7] J. Gonzalez-Rodriguez, “Evaluating automatic speaker recognition systems: An overview of the NIST speaker recognition evaluations (1996-2014),” *Loquens*, vol. 19, pp. 788–798, Jan 2014.

[8] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. A. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book, Version 3.4*. Cambridge University Engineering Department, 2006.

[9] J. Hernando and C. Nadeu, “Cdhmm speaker recognition by means of frequency filtering of filter-bank energies,” *Fifth European Conference on Speech Communication and Technology (EUROSPEECH 1997)*, Sep 1997.

[10] S. Patra and S. K. Acharya, “Hierarchical speaker identification based on latent variable decomposition,” *International Journal of Computer Applications*, vol. 19, Apr 2011.

[11] N. Fakotakis, J. Sirigos, and G. Kokkinakis, “High performance text-independent speaker recognition system based on voiced/unvoiced segmentation and multiple neural nets,” *EUROSPEECH 1999*, 1999.

[12] I. Naseem, R. Togneri, and M. Bennamoun, “A model-based approach to speaker identification using class-specific dictionaries,” *SST, Macquarie University, Sydney, Australia*, Dec 2012.

[13] S. A. Zahorian and Z. B. Nossair, “A neural network clustering technique for text-independent speaker identification,” *Intelligent Engineering Systems through Artificial Neural Networks*, vol. 4, pp. 453–460, Nov 1994.

[14] D. A. Reynolds, M. A. Zissman, T. F. Quatieri, G. C. O’Leary, and B. A. Carlson, “The effects of telephone transmission degradations on speaker recognition performance,” in *International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, 1995.