ROBUST TIME DELAY ESTIMATION IN NOISY REVERBERANT ENVIRONMENTS WITH A PROBABILISTIC GRAPHICAL MODEL

Taesu Kim[1,3†], *Hagai Attias*[2], *Soo-Young Lee*[3†], *and Te-Won Lee*[1]

[1]Institute for Neural Computation, University of California, San Diego, La Jolla, CA 92093, USA

[2]Golden Metallic Inc, P.O. Box 475608, San Francisco, CA 94147, USA

[3]Deptartment of Biosystems and Brain Science Research Center, Korea Advanced Institute of Science and Technology, Daejeon 305-701, Republic of Korea

ABSTRACT

This paper presents a fast new algorithm for estimating the relative time delay between a microphone pair in noisy reverberant conditions. The algorithm is derived from a novel approach to this problem, which we formulate in the framework of probabilistic graphical models. In this approach, we construct a probabilistic model of the microphone signals, using models of speech and noise as building blocks. The reverberation filter coefficients and the relative time delay appear as model parameters, and are estimated by our algorithm from data. The resulting delay estimate is Bayes optimal and takes into account noise and reverberation in a principled manner. We demonstrate very good performance on data from real and simulated room environments.

1. INTRODUCTION

Time delay estimation for speaker localization is an important problem in a number of domains. In video conferencing, the location of the current speaker is required for the camera to turn toward them and stabilize them at the center of the frame. In speech enhancement using a steerable microphone array, the speaker location is required to steer the array in the precise angle for optimal noise cancellation. Given the relative time delay of arrival of the speaker signal between a pair of microphones, the speaker location can be inferred using a simple geometric method.

However, whereas in ideal conditions the relative delay may be estimated from the peak in the cross-correlation function between the microphone signals, in realistic conditions the estimate obtained by this method becomes very inaccurate. The reason is that several factors contribute to distorting the temporal correlation structure of the microphone signals. First, sound sources other than the speaker interfere with the speaker signal; such sources have their own temporal structure that differs from that speaker's. Second, as the speaker signal propagates toward the microphones, reverberation effects caused by echoes, multipath propagation, and medium response modify the temporal correlations of the signal. Third, the temporal correlation of the signal itself changes in time due to the non-stationary nature of speech. These factors combine to modify the cross-correlation between the microphone signals, and make it difficult to estimate the time delay from it.

Several methods have been proposed to improve the performance of the cross-correlation based estimator. Most of them attempt to reduce the effects of noise and reverberation using the generalized cross-correlation (GCC) function, which is the cross-correlation function between filtered versions of the signals [1]. The different proposals focus on optimal ways to choose the filters [2][3][4]. We briefly review the two most popular methods, which we use as benchmarks, in Section 4.1.

Here we take a completely new approach to the problem of robust time delay estimation. We use the framework of probabilistic graphical models, which has been developed over the last decade or so in the field of machine learning [5], and is starting to have an impact on speech and signal processing (see, e.g., [6],[7],[8]). In this framework, one builds a model of the probability distribution of the observed data variables. It is also termed a generative model, because it describes the data in terms of the mechanism that generated them. In our case, we model the joint distribution of the observed microphone signals in terms of the unobserved signal originating from the speaker, distorted by unknown linear filtering due to reverberation as it propagated, contaminated by additive noise, and reaches the microphone with an unknown relative time delay.

We use a detailed probabilistic model of speech signals, which we train offline on a large dataset of clean speech, as one component of our model. We also use a probabilistic model of the noise. The reverberation filter coefficients and relative time delay appear as parameters in our model. These parameters are estimated by maximum likelihood using an iterative expectation maximization (EM) algorithm. The idea is that, since the model describes explicitly the different sources of variability in the microphone data, the resulting time delay estimate would be robust to those sources, including noise and reverberation. Moreover, whereas the model may be somewhat involved, the required computations done by the estimation algorithm are completely straightforward and performed fast an efficiently using FFT.

This paper is organized as follows. Section 2 describes the probabilistic model. Section 3 presents the EM algorithm for reverberation and time delay estimation, and outline its derivation. Section 4 describe results of the algorithm on simulated and real data. Section 5 concludes with discussion of interesting extensions.

2. PROBABILISTIC GRAPHICAL MODEL

In this section we construct a probabilistic model for speaker localization. The model describes the joint distribution of the observed microphone signals. We start from the mathematical relationship

† Supported by the Chung Moon Soul Center for Bioinformation & Bioelectronics, and by the Brain Neuroinformatics Research Program from Korean Ministry of Science and Technology.

between those signals, the unobserved speaker signal, the environmental noise and reverberation, and the relative time delay of arrival. We then construct probabilistic models for the speaker and noise signals, and use them as building blocks for the final model.

The model has three sets of parameters: speech model parameters, noise model parameters, and reverberation+delay parameters. As described in the section on parameter estimation below, the algorithm in this paper estimates only the last set of parameters from the microphone data. The noise and speech parameters are estimated separately.

The signals involved in the localization problems may be arbitrarily long time series. To facilitate modelling, we divide them into successive N-point frames. Our model describes the microphone signal distribution within a frame, which is typically $25-30$msec long, and assumes that different frames are statistically independent. Parameter estimation involves averaging over all frames.

Notation. We use $\mathcal{N}(z \mid \mu, \gamma) = \sqrt{\gamma/2\pi}\exp[-(\gamma/2)(z-\mu)^2]$ to denote a Gaussian distribution over a random variable z with mean μ and precision γ (precision=1/variance). For a time domain signal $x_n, n = 0:N-1$, we denote its DFT by $X_k = \sum_n \exp(-i\omega_k n)x_n$. With a couple of exceptions (μ_{sk} and γ_{sk} below), a DFT is denoted by capitalizing the letter of its time domain counterpart.

2.1. Microphone Signal Model

Let y_{1n}, y_{2n} be the signals captured by microphones $1, 2$ at time $n = 0 : N-1$, and let x_n be the speaker signal at that time. They are related by linear filtering and additive noise,

$$\begin{aligned} y_{1n} &= h_{1n} \star x_n + u_{1n} \\ y_{2n} &= h_{2,n-\tau} \star x_{n-\tau} + u_{2n} \end{aligned} \tag{1}$$

where h_{im}, $m = 0 : L$ is the impulse response of filter acting on the speaker signal on its way to microphone i, and u_{in} is the noise contaminating the filtered signal at microphone i. τ is the relative time delay of arrival.

Since noise in realistic environments is temporally correlated, to obtain a robust τ estimator it is important to use a noise model that describes such correlations. Here we choose to use an autoregressive model of order q,

$$b_{1n} \star u_{1n} = v_{1n}\ , \qquad b_{2n} \star u_{2n} = v_{2n} \tag{2}$$

where b_{in}, $n = 0 : q$ are the parameters of the AR(q) model for noise i, and v_{in} is an i.i.d. Gaussian signal with precision λ_i. We are using the notation in which $b_{i0} = 1$ and Eq. (2) is $u_{in} = \sum_{m=1}^{q}(-b_{im})u_{in-m} + v_{in}$. We assume that the noise is stationary, i.e., the parameters (b_{im}, λ_i) do not change between frames. We also assume that the noise signals at the different microphones are uncorrelated; this assumption is made for mathematical simplicity and can be easily removed without affecting the tractability of the model.

We can now write the probability distribution of the microphone signals conditioned on the speaker signal, at each frame,

$$\begin{aligned} p(y_1 \mid x) &= \prod_n \mathcal{N}(b_{1n} \star y_{1n} \mid b_{1n} \star h_{1n} \star x_n, \lambda_1) \\ p(y_2 \mid x) &= \prod_n \mathcal{N}(b_{2n} \star y_{2n} \mid b_{2,n-\tau} \star h_{2,n-\tau} \star x_{n-\tau}, \lambda_2)\ . \end{aligned} \tag{3}$$

To derive Eqs. (3), start with
$p(u_i) = \prod_n \mathcal{N}(u_{in} \mid \sum_{m=1}^{q}(-b_{im})u_{i,n-m}, \lambda_i)$
$= \prod_n \mathcal{N}(b_{in} \star u_{in} \mid 0, \lambda_i)$, which follows from (2); then substitute $u_{1n} = y_{1n} - h_{1n} \star x_n$ and $u_{2n} = y_{2n} - h_{2,n-\tau} \star x_{n-\tau}$ (Eq. (1)).

Noise model training. The noise parameters (b_{in}, λ_i) are estimated directly from pure noise segments obtained from silent (no speech) parts of the microphone data, prior to applying the estimation algorithm below.

2.2. Speech Signal Model

Speech signals have several important features. They are temporally correlated, non-Gaussian, and non-stationary. Here we use a model that captures all those features. It is a mixture model with S component distributions. Each component $s = 1 : S$ is an autoregressive model of order r,

$$a_{sn} \star x_n = v_n \tag{4}$$

where a_{sn}, $n = 0 : r$ are the parameters of the AR(r) model for component s, and v_n is an i.i.d. Gaussian signal with precision ν_s. As above, in our notation $a_{s0} = 1$ and Eq. (4) is $x_n = \sum_{m=1}^{r}(-a_{sm})x_{n-m} + v_n$. This model divides the frames of the speech signal into clusters, where different clusters have different temporal correlations. A signal consisting of a sequence of frames would typically jump among the clusters due to its non-stationarity. The clustering is soft because of the probabilistic nature of the model.

The probability distribution of the speech signal at each frame is given by

$$p(x \mid s) = \prod_n \mathcal{N}(a_{sn} \star x_n \mid 0, \nu_s)\ , \qquad p(s) = \pi_s \tag{5}$$

where $\pi_s \geq 0$ are the mixing fractions which sum up to unity, $\sum_s \pi_s = 1$. Hence, our speech model is a mixture of AR Gaussians, $p(x) = \sum_s p(x \mid s)p(s)$.

Speech model training. The parameters (a_{sn}, ν_s, π_s) are estimated offline from a large, speaker independent dataset of clean speech signals, prior to applying the estimation algorithm below. For the experiments described in this paper, the dataset included 1000 sentences from the Wall Street Journal, read by 100 male and female native English speakers. The training algorithm used standard EM (omitted) using $S = 256$ clusters, initialized by vector quantization. For more details on training and using similar speech models as building blocks in graphical models of microphone signals, see [6],[7].

2.3. Full Model

We now have the full joint probability distribution that defines our model. It is given by the factored form

$$p(y, x, s) = p(y_1 \mid x)p(y_2 \mid x)p(x \mid s)p(s) \tag{6}$$

which is the product of the above distributions. The model is parametrized by the filter, delay, noise, and speech parameters $(h_{in}, \tau, b_{in}, \lambda_i, a_{sn}, \nu_s, \pi_s)$.

Such a model is also termed a probabilistic generative model, since it has a generative interpretation. In our case, the model describes the following mechanism for generating the observed data. At each frame, component s is selected with probability $p(s)$. Next, speech signal x is sampled from the conditional distribution $p(x \mid s)$. Finally, microphone signals $y_{1,2}$ are sampled from the conditional distributions $p(y_{1,2} \mid x)$.

3. PARAMETER ESTIMATION ALGORITHM

This section presents the algorithm that estimates the reverberation parameters h_{in} and time delay τ. As usual with graphical models, it is an EM algorithm, i.e. an iterative maximum likelihood procedure. Each iteration consists of an E-step and an M-step, where the M-step updates the parameter estimates, and the E-step updates the sufficient statistics used in the M-step.

3.1. Reverberation Parameters

The filter coefficients h_{in} are updated by solving the linear equation

$$\begin{aligned} \sum_{n=0}^{L} c_{1,m-n} h_{1n} &= r_{1m} , \quad m = 0 : L \\ \sum_{n=0}^{L} c_{2,m-n} h_{2n} &= \sum_{\tau} q_\tau r_{2,m+\tau} \end{aligned} \tag{7}$$

which can be done efficiently by Levinson recursion. The sufficient statistics involved are defined by

$$\begin{aligned} r_{im} &= E \sum_n (b_{in} \star x_n)(b_{i,n+m} \star y_{i,n+m}) \\ c_{im} &= E \sum_n (b_{in} \star x_n)(b_{i,n+m} \star x_{n+m}) \end{aligned} \tag{8}$$

where E here denotes averaging over the speaker signal x at each frame w.r.t. its posterior distribution $p(x \mid y)$, as well as over frames. r_{im}, c_{im} are computed below. q_τ is the posterior distribution over the delay τ. To derive (7), consider the averaged log distribution of our model $E \log p(y, x, s)$, compute its derivative w.r.t. h_{im}, and set it to zero.

3.2. Time Delay

We consider the posterior distribution $p(\tau \mid y)$ over the time delay

$$q_\tau = p(\tau \mid y) = \frac{1}{z} \exp(\lambda_2 f_{2\tau}) \tag{9}$$

where $z = \sum_\tau \exp(\lambda_2 f_{2\tau})$ is the normalization constant. The time delay estimate is

$$\tau = \arg\max_{\tau'} q_{\tau'} \tag{10}$$

The sufficient statistics involved are defined by

$$f_{im} = \sum_n (b_{in} \star h_{in} \star x_n)(b_{i,n+m} \star y_{i,n+m}) \tag{11}$$

and computed below. To derive the posterior, apply Bayes' rule $p(\tau \mid y) = p(y \mid \tau) p(\tau) / p(y)$, from which, assuming a flat prior $p(\tau) = const.$, it follows that $\log p(\tau \mid y) = E \log p(y_2 \mid x, \tau)$, where E averages over x and frames as in (8).

3.3. Sufficient Statistics

To compute the sufficient statistics r_{im}, c_{im}, f_{im}, we consider their DFT R_{ik}, C_{ik}, F_{ik}, where $R_{ik} = \sum_m \exp(-i\omega_k m) r_{im}$ etc. From (8,11),

$$\begin{aligned} R_{ik} &= \mid B_{ik} \mid^2 Y_{ik} E X_k^\star \\ C_{ik} &= \mid B_{ik} \mid^2 E \mid X_{ik} \mid^2 \\ F_{ik} &= \mid B_{ik} \mid^2 H_{ik}^\star Y_{ik} E X_k^\star . \end{aligned} \tag{12}$$

The averages of X_k and $\mid X_k \mid^2$ are given by

$$\begin{aligned} EX_k &= \sum_s \bar{\pi}_s \mu_{sk} , \\ E \mid X_k \mid^2 &= \sum_s \bar{\pi}_s \mid \mu_{sk} \mid^2 + \frac{N}{\gamma_{sk}} \end{aligned} \tag{13}$$

where the quantities $\bar{\pi}_s, \mu_{sk}, \gamma_{sk}$ are

$$\begin{aligned} \gamma_{sk} &= \lambda_1 \mid B_{1k} H_{1k} \mid^2 + \lambda_2 \mid B_{2k} H_{2k} \mid^2 + \nu_s \mid A_{sk} \mid^2 \\ \mu_{sk} &= \frac{1}{\gamma_{sk}} \left(\lambda_1 \mid B_{1k} \mid^2 H_{1k}^\star Y_{1k} + \lambda_2 \mid B_{2k} \mid^2 H_{2k}^\star Q_k^\star Y_{2k} \right) \\ \bar{\pi}_s &= \frac{1}{z} \pi_s \exp\left[\sum_k \frac{\gamma_{sk}}{2} \mid \mu_{sk} \mid^2 - \frac{1}{2} \log \gamma_{sk} \right] . \end{aligned} \tag{14}$$

Here z is a normalization constant set to ensure $\sum_s \bar{\pi}_s = 1$, and Q_k is the DFT of q_τ (9). Hence, to compute the sufficient statistics, we substitute (13,14) into (12) and apply inverse DFT.

To prove Eqs. (13,14), consider the posterior distribution $p(x, s \mid y)$ over the speaker signal x and state s at each frame. From Bayes' rule $p(x, s \mid y) = p(y, x, s)/p(y)$. For a given s, the conditional $p(x \mid s, y)$ is therefore Gaussian, with mean μ_{sn} (whose DFT is μ_{sk}) and Toeplitz precision matrix $\Gamma_{s,nm} = (1/N) \sum_k \exp(i\omega_k (n - m)) \gamma_{sk}$. Some rearrangement using $\log p(x, s \mid y) = \log p(x \mid s, y) + \log p(s \mid y)$ yields the expression for $p(s \mid y) = \bar{\pi}_s$.

4. EXPERIMENTS

We evaluated the performance of our algorithm using simulated and real data. Simulated data were obtained by simulating impulse responses of a rectangular room using the image model technique [9]. Room reverberation times ranged from 0ms to 100ms. To generate microphone signals, a real speech signal sampled at 16kHz was convolved with the room impulse responses. White Gaussian noise signals were then added to it at SNR between -5dB to 25dB. We then applied a 200ms long Hamming window at 50ms jumps to obtain 320 segments of microphone signals. The estimation algorithm was applied to each segment separately.

Real data were recorded in an office with noise emanating from the fans of two PCs and from AC. The speech signal was played from a speaker located approximately 3m from the microphones at different angles of arrival. The actual time delay was measured separately using long segments of white noise.

4.1. Benchmarks

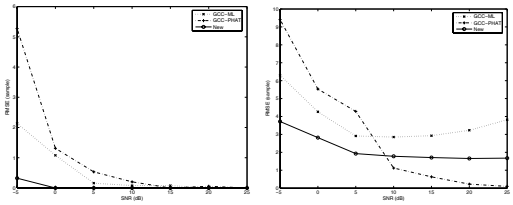
We compared our algorithm to two widely used techniques based in the generalized cross-correlation (GCC) function. The GCC between y_{1n}, y_{2n} is given by

$$C_\tau = \sum_k e^{i\omega_k \tau} W_k X_{1k} X_{2k}^\star \tag{15}$$

where W_k is a weight function. The time delay estimate is τ which maximizes C_τ. In [2], an approximate maximum likelihood (ML) based weight function

$$W_k^{ML} = \frac{|X_{1k}||X_{2k}|}{|U_{1k}|^2 |X_{2k}|^2 + |U_{2k}|^2 |X_{1k}|^2} , \tag{16}$$

where $|U_{1k}|$ and $|U_{2k}|$ are the noise power spectra at the two microphones, which are estimated during non-speech intervals like



(a) Reverberation time 30ms (b) Reverberation time 100ms

Fig. 1. Performance on simulated data

	bias	variance	RMSE
GCC-ML	1.91×10^{-2}	1.88×10^{-2}	1.38×10^{-1}
GCC-PHAT	4.77×10^{-2}	6.48×10^{-2}	2.58×10^{-1}
New	1.72×10^{-1}	1.43×10^{-1}	4.14×10^{-1}

Table 1. Performance on real data. Direction of arrival is 0° and mean SNR is 9.5dB

in our algorithm. The ML weight function has been shown to perform well when the room reverberation time scale is short, but degrades as it increases. It has been proposed that to reduce the effect of room reverberation, one should deemphasize the frequency dependence of the GCC. One way to do that uses the phase transform (PHAT) weight function

$$W_k^{PHAT} = \frac{1}{|X_{1k} X_{2k}^\star|}. \tag{17}$$

The PHAT method has been shown to perform well at low noise levels.

4.2. Results

We present performance results in terms of bias, variance, and root-mean-square error (RMSE), which are defined by

$$\begin{aligned} \text{bias} &= |\tau - E[\hat{\tau}]|\,, \qquad \text{variance} = \text{Var}[\hat{\tau}] \\ \text{RMSE} &= \sqrt{\text{bias}^2 + \text{variance}} \end{aligned} \tag{18}$$

where τ is actual time-delay, and $\hat{\tau}$ is estimated time-delay.

Fig. 1 shows the performance of the different methods in simulated room environments with different reverberation time scales, as a function of SNR. At 30ms reverberation time our algorithm outperforms both GCC-ML and GCC-PHAT at all SNRs. At 100ms, our algorithm outperforms both GCC methods at SNR below 10dB, and is dominated by GCC-PHAT at higher SNRs.

Tables 1,2 show performance on real data at different directions of arrival and SNR. Here the results are less conclusive, where our algorithms sometimes outperformed the other methods and sometime underperforms them. Similar results have been obtained in other conditions. We are currently performing extensive experiments in order to get a clear picture of the strengths of the different methods and the differences between them. The results of those experiments will summarized in the final version of the paper.

5. EXTENSIONS

One important extension of our algorithm is to facilitate estimating long reverberation filters. Currently, the estimated filters must be shorter than the signal frames, which are 25ms long. This limits the robustness of the algorithm, since real environments may have reverberation times that $5-10$ frame long or longer. It is not quite trivial to estimate long filters, however, as this requires modifying the computation of sufficient statistics. The reason is that different frames can no longer be considered independent during inference, since they are now coupled by the filters. A similar situation has been treated in [7] using variational techniques [5].

	bias	variance	RMSE
GCC-ML	8.25×10^{-2}	7.60×10^{-2}	2.87×10^{-1}
GCC-PHAT	1.30×10^{-1}	1.14×10^{-1}	3.61×10^{-1}
New	3.17×10^{-2}	5.63×10^{-2}	2.39×10^{-1}

Table 2. Performance on real data. Direction of arrival is 50° and mean SNR is 7.1dB

Another important extension is estimating the noise parameters from the microphone data, rather than rely on the availability of pure noise segments. It is quite straightforward to obtain an update rule for those parameters. Finally, to be robust to non-stationary conditions, the algorithm must be modified to track the filters and noise parameters as they change in time. We are currently working on this extension using a recursive estimation technique.

6. REFERENCES

[1] C. H. Knapp and G. C. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.

[2] J. Adcock M. Brandstein and H. Silverman, "A practical time-delay estimator for localizing speech sources with a microphone array," *Comput. Speech Lang.*, vol. 9, pp. 153–169, 1995.

[3] M. Brandstein, "Time-delay estimation of reverberated speech exploiting harmonic structure," *J. Acoust. Soc. Amer.*, vol. 105, no. 5, pp. 2914–2919, 1999.

[4] B. Champagne S. Bédard and A. Stéphanne, "Effects of room reverberation on time-delay estimation performance," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1994, pp. II 261–264.

[5] M.I. Jordan (Ed.), *Learning in Graphical Models*, MIT Press, 1991.

[6] A. Acero H. Attias, J.C. Platt and L. Deng, "Speech denoising and dereverberation using probabilistic models," in *Adv. Neural Information Processing Systems*, 2001, pp. 758–764.

[7] H. Attias and L. Deng, "A new approach to speech enhancement with a microphone array using em and mixture models," in *Proc. 7th Int. Conf. on Spoken Language Processing*, 2002.

[8] G.-J. Jang and T.-W. Lee, "A maximum likelihood approach to single-channel source separation," *J. Machine Learning Research*, vol. 4, pp. 1365–1392, 2003.

[9] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, pp. 943–950, 1979.