EM-based Gain Adaptation for Probabilistic Multipitch Tracking

Michael Wohlmayr, Franz Pernkopf

Signal Processing and Speech Communication Laboratory
Graz University of Technology, Austria
michael.wohlmayr@tugraz.at, pernkopf@tugraz.at

Abstract

We introduce an EM algorithm for automatic speaker gain adaptation, and use this approach for probabilistic multipitch tracking. We derive a lower bound on the log-likelihood of the gain parameters and use a fast pruning method to make lower bound optimization efficient. We evaluate the performance of gain adapted multipitch tracking on the GRID database, where 3000 speech mixtures were generated for each mixing level. For gain differences in the range of zero up to 18dB, the proposed method achieves almost the same performance as for the case where the gain is assumed to be known.

Index Terms: Multipitch tracking, factorial hidden Markov model, mixture maximization, gain adaptation

10.21437/Interspeech.2011-518

1. Introduction

Pitch is a significant cue for speech analysis and corresponds to the perceived tonal height of voiced utterances of speech. Estimation and tracking of pitch is an important and ongoing research area in speech and audio signal processing. Well performing algorithms exist for the case of single speaker speech [1][2], however tracking the pitch of multiple concurrent speakers from single-channel recordings is a more challenging task. Accurate estimation of pitch contours and their reliable assignment to sources/speakers can provide relevant cues for single-channel speech separation and computational auditory scene analysis. In [3], we proposed a statistical approach for multipitch tracking. The (unknown) pitch dynamics are modeled by a Markov chain for each speaker, and pitch dependent characteristics of speech are modeled by Gaussian mixture models (GMMs). The usage of the mixture maximization (MIXMAX) interaction model [4] allows to combine single Markov chains to form a factorial hidden Markov model (FHMM) [5]. Given a speech mixture, pitch trajectories are obtained by inference on the FHMM. The system competes with a state-of-the-art multi-pitch tracking system [6], and additionally assigns the extracted pitch trajectories to individual speakers when speaker-dependent models are available. However, so far we assumed that the gain of each speaker in a mixture matches the speaker model obtained during training. As will be shown in the experiments, a mismatch between speaker model and actual speaker gain results in a degraded tracking performance. In this work, we propose an EM algorithm to adapt speaker models to the actual gain of each speaker in a speech mixture.

In a related framework for monaural speech separation, methods for gain adaptation based on iterated tracking and derivative-free optimization have been introduced [7] [8]. In [9], a variational framework for approximate inference in conjunction with the MIXMAX model was developed, which was further used for gain adaptation. Similar to [9], the method in this work uses a variational lower bound to derive a novel EM algorithm for gain adaptation. The resulting algorithm makes use of a fast pruning method we proposed earlier in [10], which efficiently constructs an upper bound on MIXMAX observation likelihoods and thus allows fast pruning of unlikely state combinations.

The paper is organized as follows: In section 2, we give a short review on the statistical approach for multipitch tracking proposed in [3], and extend this approach by gain adapted speaker models. In section 3, we introduce an EM algorithm to adapt the gain of each speaker model. Experimental results are presented in section 4, where we evaluate the performance of the proposed adaptation method. Finally, section 5 concludes the paper.

2. Probabilistic Multipitch Tracking Model

For simplicity we consider the case of two interfering speakers, although this approach can easily be extended to more speakers. The pitch trajectory of an individual speaker is modelled by a Markov chain, where pitch is discretized into 170 states. State '1' refers to 'no pitch' (i.e. unvoiced or silent), and the state values '2'-'170' encode frequency values ranging from 80 to 500Hz. Specifically, the pitch value corresponding to state $x \in \{2, ..., 170\}$ is given as $f_0 = \frac{16000}{30+x}$Hz. Similar to [6], this results in a non-uniform quantization, where low pitch values have a more fine-grained resolution than high pitch values. The individual Markov chains are combined using the framework of FHMMs, which enables the tracking of the states of multiple Markov processes evolving in parallel over time. The available observations are considered as a joint effect of all single Markov processes. Fig. 1 shows the FHMM considered here, represented as a factor graph. We denote by $x_i^{(t)}$ the hidden state of the i^{th} Markov chain (corresponding to the i^{th} speaker) at time frame t. Realizations of observed random variables at t are collected in a D-dimensional vector $\boldsymbol{y}^{(t)} \in \mathbb{R}^D$. In order to infer the most likely state sequences for both speakers, a Viterbi decoder for FHMMs is applied [3].

The edges between nodes in Fig. 1 indicate a conditional dependency between random variables. Specifically, the dependency of hidden variables between two consecutive time instances is defined for each Markov chain by the transition probability $p(x_i^{(t)}|x_i^{(t-1)})$. The transitions $p(x_i^{(t)}|x_i^{(t-1)})$ and priors $p(x_i^{(1)})$ are determined by maximum-likelihood estimation and parameter smoothing on pitch-labeled training data.

At each time frame t, the FHMM models the feature vector $\boldsymbol{y}^{(t)}$ extracted from the mixture signal by the observation probability $p(\boldsymbol{y}^{(t)}|x_1^{(t)}, x_2^{(t)})$. The design of $p(\boldsymbol{y}^{(t)}|x_1^{(t)}, x_2^{(t)})$ is guided by the insight that feature vectors based on the log-spectrogram (or alternatively the magnitude spectrogram)

Copyright © 2011 ISCA

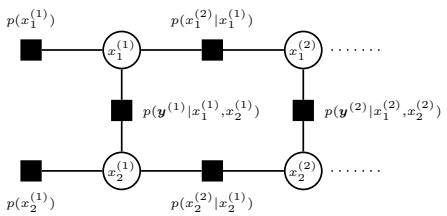


Figure 1: A factorial HMM shown as a factor graph [11]. Factor nodes are depicted as shaded rectangles together with their functional description. Hidden variable nodes are shown as circles. Here, observed variables $\boldsymbol{y}^{(t)}$ are absorbed into factor nodes.

can be approximated by an *interaction model* f: $\boldsymbol{y}^{(t)} \approx f(\boldsymbol{s}_1^{(t)}, \boldsymbol{s}_2^{(t)})$, where $\boldsymbol{s}_i^{(t)}$ is the feature vector for a single speaker. In [3], we investigated two different interaction models for multipitch tracking, namely the mixture-maximization (MIXMAX) model [4] and a linear interaction model. The MIXMAX interaction model was originally proposed in [4] for noise robust speech recognition. The underlying assumption is that each particular time-frequency cell of a mixed-speech spectrogram is dominated by a single speaker, which is valid with high probability due to the sparse nature of speech. In this work, we employ the MIXMAX approximation, and extend the probabilistic model by a mechanism for gain adaptation.

2.1. Gain Adapted Observation Model

In the following, we assume that the time-domain mixture signal $\tilde{y}[n]$ is a weighted instantaneous mixture of single speaker source signals $\tilde{s}_i[n]$:

$$\tilde{y}[n] = \tilde{b}_1 \tilde{s}_1[n] + \tilde{b}_2 \tilde{s}_2[n]. \tag{1}$$

Without loss of generality, we assume that $\tilde{s}_i[n]$ represents a source signal with unit-variance, and $\tilde{b}_i$ denotes a positive gain factor related to speaker i. In this work, we make the simplifying assumption that the speaker-related gain factors do not change over time. In log-magnitude domain, the MIXMAX model approximates (1) as the elementwise maximum of the single-speaker contributions:

$$\boldsymbol{y}^{(t)} \approx \max(\boldsymbol{s}_1^{(t)} + b_1, \boldsymbol{s}_2^{(t)} + b_2). \tag{2}$$

Here, $\boldsymbol{y}^{(t)}$ and $\boldsymbol{s}_i^{(t)}$ denote the short-time log-spectrum of the speech mixture and the source signals at time frame t, respectively, and the gain parameter $b_i = \log(\tilde{b}_i)$ is additive in log-domain. We interpret the MIXMAX approximation as a probabilistic interaction model, i.e. $p(\boldsymbol{y}^{(t)}|\boldsymbol{s}_1^{(t)}, \boldsymbol{s}_2^{(t)}) = \delta\left(\boldsymbol{y}^{(t)} - \max(\boldsymbol{s}_1^{(t)}, \boldsymbol{s}_2^{(t)})\right)$, where $\delta(\cdot)$ denotes the Dirac delta.

The pitch conditional distribution of log-spectral features $\boldsymbol{s}_i^{(t)}$ is modeled for each speaker $i \in \{1, 2\}$ by a Gaussian mixture model (GMM):

$$p(\boldsymbol{s}_i^{(t)}|x_i) = p(\boldsymbol{s}_i^{(t)}; \boldsymbol{\Theta}_{i,x_i}) = \sum_{m=1}^{M_{i,x_i}} \alpha_{i,x_i}^m \mathcal{N}\left(\boldsymbol{s}_i^{(t)}; \boldsymbol{\theta}_{i,x_i}^m\right), \tag{3}$$

where variables right to the symbol ; represent the parameters of a distribution. $M_{i,x_i} \geq 1$ is the number of mixture components, and α_{i,x_i}^m corresponds to the weight of each component $m = 1, \ldots, M_{i,x_i}$. These weights are constrained to be positive, $\alpha_{i,x_i}^m \geq 0$, and $\sum_{m=1}^{M_{i,x_i}} \alpha_{i,x_i}^m = 1$. The GMM for pitch x_i is fully specified by the parameter set $\boldsymbol{\Theta}_{i,x_i} = \{\alpha_{i,x_i}^m, \boldsymbol{\theta}_{i,x_i}^m\}_{m=1}^{M_{i,x_i}}$, where $\boldsymbol{\theta}_{i,x_i}^m = \{\boldsymbol{\mu}_{i,x_i}^m, \boldsymbol{\Sigma}_{i,x_i}^m\}$ denotes the mean vector and the diagonal covariance matrix of the m^{th} component. GMM parameters are learned from pitch-labeled speech signals using the EM algorithm [12], and the number of components is selected using the MDL criterion [13]. It is important to note that all speech utterances used for model training were normalized to unit-variance in time-domain. The resulting speaker model still captures energy fluctuations due to natural speaking style. To capture the overall gain difference between speaker model and the actual source component, we express a gain adapted speaker model as $p(\boldsymbol{s}_i^{(t)}|x_i; b_i) = p(\boldsymbol{s}_i^{(t)} - \mathbf{1}b_i|x_i)$, where $\mathbf{1}$ denotes a vector of same dimension as $\boldsymbol{s}_i^{(t)}$ with all elements set to one.

Combining the interaction model with gain adapted speaker models, we obtain the observation probability by marginalization over the unknown single speaker log-spectra:

$$p(\boldsymbol{y}^{(t)}|x_1, x_2; b_1, b_2) = \int\int p(\boldsymbol{y}^{(t)}|\boldsymbol{s}_1, \boldsymbol{s}_2) p(\boldsymbol{s}_1 - \mathbf{1}b_1|x_1) \times p(\boldsymbol{s}_2 - \mathbf{1}b_2|x_2) d\boldsymbol{s}_1 d\boldsymbol{s}_2, \tag{4}$$

Note that we made explicit the dependency of the observation model on gain parameters b_i. For the MIXMAX interaction model and GMM-based state conditional probabilities, the integrals in (4) can be solved in closed-form (see e.g. [3] for a derivation), and we obtain the following observation model:

$$p(\boldsymbol{y}^{(t)}|x_1, x_2; b_1, b_2) = \sum_{m=1}^{M_{1,x_1}} \sum_{n=1}^{M_{2,x_2}} \alpha_{1,x_1}^m \alpha_{2,x_2}^n \times \prod_{d=1}^{D} \left\{ f_{d,t,b_1}^{x_1,m} G_{d,t,b_2}^{x_2,n} + F_{d,t,b_1}^{x_1,m} g_{d,t,b_2}^{x_2,n} \right\}, \tag{5}$$

where we introduced the shorthand symbols $f_{d,t,b_1}^{x_1,m} = \mathcal{N}(y_d^{(t)} - b_1; \theta_{1,x_1}^{m,d})$, $g_{d,t,b_2}^{x_2,n} = \mathcal{N}(y_d^{(t)} - b_2; \theta_{2,x_2}^{n,d})$, $F_{d,t,b_1}^{x_1,m} = \Phi(y_d^{(t)} - b_1; \theta_{1,x_1}^{m,d})$, $G_{d,t,b_2}^{x_2,n} = \Phi(y_d^{(t)} - b_2; \theta_{2,x_2}^{n,d})$. The d^{th} element of $\boldsymbol{y}^{(t)}$ is denoted by $y_d^{(t)}$, $\theta_{i,x_i}^{m,d}$ gives the d^{th} element of the corresponding mean and variance, and $\Phi(y; \theta) = \int_{-\infty}^{y} \mathcal{N}(x; \theta) dx$ denotes the univariate cumulative normal distribution.

3. EM-based Gain Adaptation

Given a speech mixture, in a first step we adapt the gain parameters by maximizing the log-likelihood of the data. Denoting the sequence of all variables as $\{x^{(t)}\} = \bigcup_{t=1}^{T} \{x_1^{(t)}, x_2^{(t)}\}$ and $\{\boldsymbol{y}^{(t)}\} = \bigcup_{t=1}^{T} \boldsymbol{y}^{(t)}$, the joint distribution of all random variables in the FHMM is given as

$$p(\{x^{(t)}\}, \{\boldsymbol{y}^{(t)}\}; b_1, b_2) = \prod_{k=1}^{2} p(x_k^{(1)}) \prod_{t=2}^{T} p(x_k^{(t)}|x_k^{(t-1)}) \times \prod_{t=1}^{T} p(\boldsymbol{y}^{(t)}|x_1^{(t)}, x_2^{(t)}; b_1, b_2). \tag{6}$$

The log-likelihood of the gain parameters is then obtained by marginalizing over all latent random variables:

$$\log p(\{\boldsymbol{y}^{(t)}\}; b_1, b_2) = \log \sum_{\{x^{(t)}\}} p(\{x^{(t)}\}, \{\boldsymbol{y}^{(t)}\}; b_1, b_2). \tag{7}$$

It is difficult to maximize the log-likelihood directly. Instead, we rely on a common strategy and apply Jensen's inequality to construct a lower bound on (7). For any probability mass function $q(i)$, and any nonnegative function $f(x)$, Jensen's inequality states that

$$\log \sum_i f(x_i) = \log \sum_i q(i) \frac{f(x_i)}{q(i)} \geq \sum_i q(i) \log \frac{f(x_i)}{q(i)}. \tag{8}$$

Introducing two arbitrary probability mass functions q_1 and q_2, we apply this inequality two times to the log-likelihood (7), and obtain the following lower-bound:

$$\begin{aligned}\mathcal{L}(q_1, q_2, b_1, b_2) = &\sum_{\{x^{(t)}\}} q_1(\{x^{(t)}\}) \sum_{t,m,n} q_2(m, n | x_1^{(t)}, x_2^{(t)}) \\ &\times \sum_d \log \left\{ f_{d,t,b_1}^{x_1,m} G_{d,t,b_2}^{x_2,n} + F_{d,t,b_1}^{x_1,m} g_{d,t,b_2}^{x_2,n} \right\} \\ &+ \text{const.}\end{aligned} \tag{9}$$

Here, const. refers to all terms independent of b_1 and b_2. Note that this lower bound is valid for arbitrary probability distributions q_1 and q_2. Maximization of (7) can now be performed using the EM algorithm. Starting with an initial guess for the gain parameters, the following two steps are repeated iteratively:

E-Step: $\{q_1, q_2\} \leftarrow \arg\max_{\{q_1', q_2'\}} \mathcal{L}(q_1', q_2', b_1, b_2)$

M-Step: $\{b_1, b_2\} \leftarrow \arg\max_{\{b_1', b_2'\}} \mathcal{L}(q_1, q_2, b_1', b_2')$

For the sake of computational efficiency, we do not attempt to fully maximize the criterion in either step. In the following, we give details on the proposed update scheme. For more information on the general optimization framework applied in this work, we refer the interested reader to [14].

3.1. E-Step

Keeping the current parameter estimates $\boldsymbol{b}^{(old)} = (b_1^{(old)}, b_2^{(old)})^T$ fixed, the lower bound in (9) is maximized by setting $q_1(\{x^{(t)}\}) = p(\{x^{(t)}\} | \{\boldsymbol{y}^{(t)}\}; \boldsymbol{b}^{(old)})$ and $q_2(m, n | x_1^{(t)}, x_2^{(t)}) = p(m, n | \boldsymbol{y}^{(t)}, x_1^{(t)}, x_2^{(t)}; \boldsymbol{b}^{(old)})$. In this case, the value of the lower bound equals the true log-likelihood at point $\boldsymbol{b}^{(old)}$ [14]. However, the resulting calculations involve a complete pass of the forward-backward algorithm. To reduce computational complexity, we instead set $q_1(\{x^{(t)}\}) = \prod_t p(x_1^{(t)}, x_2^{(t)} | \boldsymbol{y}^{(t)}; \boldsymbol{b}^{(old)})$, i.e. we assume independence between time steps, and obtain the following lower bound:

$$\begin{aligned}\mathcal{L}(q_1, q_2, b_1', b_2') = &\sum_{t,x_1,x_2,m,n} p(x_1, x_2 | \boldsymbol{y}^{(t)}; \boldsymbol{b}^{(old)}) \\ &\times p(m, n, | x_1, x_2, \boldsymbol{y}^{(t)}; \boldsymbol{b}^{(old)}) \\ &\times \sum_d \log \left\{ f_{d,t,b_1'}^{x_1,m} G_{d,t,b_2'}^{x_2,n} + F_{d,t,b_1'}^{x_1,m} g_{d,t,b_2'}^{x_2,n} \right\} \\ &+ \text{const.}\end{aligned} \tag{10}$$

The computational complexity of (10) can be drastically reduced by pruning elements in the pitch-state posterior $p(x_1, x_2 | \boldsymbol{y}^{(t)}; \boldsymbol{b}^{(old)})$. This is done by using an efficient method we proposed in [10], where out of $|x_1| \times |x_2| = 170^2$ entries, at most N of the (probably) largest entries are selected in a fast way, and all remaining elements are set to zero. The resulting posterior is then re-normalized. During the experiments, we set $N = 60$ (for more details, see section 4).

3.2. M-Step

Unfortunately, there is no closed-form solution for $(b_1, b_2)^T$ that maximizes eq. (10). Therefor, we need to rely on gradient based optimization methods. We propose to perform a single step of Newton's method for optimization,

$$\boldsymbol{b}^{(new)} = \boldsymbol{b}^{(old)} - \boldsymbol{H}^{-1} \boldsymbol{\Delta}, \tag{11}$$

where $\boldsymbol{\Delta}$ and $\boldsymbol{H}$ are the gradient and the Hessian, respectively, of the lower bound (10) derived during the previous E-Step. Note that it is not necessary to attain a local maximum of $\mathcal{L}(q_1, q_2, b_1', b_2')$, i.e. in principle a single Newton step is sufficient [14]. A second Newton step would result in more computational effort, which does not pay back in terms of increase in log-likelihood.

4. Experimental Results

We evaluate the proposed method for gain adaptation in terms of pitch tracking accuracy, and compare the results to the baseline method of Wu et al. [6]. Three female and three male speakers were selected from the GRID database [15], where 500 sentences are available per speaker. For each speaker, 490 sentences were used to train speaker dependent GMMs, while the remaining 10 sentences were used for testing. Combining each speaker with every other speaker results in 15 speaker pairs, with 100 combinations of test utterances for each speaker pair. Originally, each test utterance $\tilde{s}[n]$ had the same gain level as the training speech. We created mixtures of two speakers with a predefined gain difference level L from the range $L = \{0, 3, 6, 9, 12, 15, 18\}$ [dB]. For each value of L, and each combination of test utterances $\tilde{s}_1[n]$ and $\tilde{s}_2[n]$, we created two mixtures $\tilde{m}_1[n] = \tilde{b}_L \tilde{s}_1[n] + \tilde{s}_2[n]$ and $\tilde{m}_2[n] = \tilde{s}_1[n] + \tilde{b}_L \tilde{s}_2[n]$, where $\tilde{b}_L = 10^{L/20}$. This results in a total of 3000 test mixtures per gain level L (15 speaker pairs $\times$ 100 combinations per speaker pair $\times$ 2 mixtures). The reference pitch trajectories were obtained directly from the single speech utterances using the RAPT method. [1]

The observed features $\boldsymbol{y}^{(t)}$ are based on the log-spectrogram of the speech mixture. Given an input signal at sampling rate $f_s = 16$kHz, we compute the spectrogram via the 1024 point FFT, using a Hamming window of length 32ms and step size of 10ms. Next, we obtain each observation vector $\boldsymbol{y}^{(t)} \in \mathbb{R}^{64}$ by taking the magnitude-log of spectral bins 2-65, which corresponds to a frequency range up to 1000Hz.

For each test mixture, we first applied the gain adaptation method proposed in section 3, and then performed multipitch tracking using the gain adapted speaker models, as described in section 2. The EM-algorithm was run for 20 iterations, however we observed convergence after 8 iterations for the majority of speech mixtures. To decrease the computational complexity of both gain adaptation and multipitch tracking, the fast pruning method we proposed earlier in [10] was used: During the E-Step of the proposed gain adaptation method (see section 3.1), we set $N = 60$, i.e. at most 60 out of 170^2 entries of the pitch-state posterior in (10) need to be computed per time frame. In rare cases, this low value of N caused the algorithm to diverge, such that the algorithm was restarted with $N = 120$. For additional speedup, we incorporated only every 5$^{\text{th}}$time frame into calculations. For the computation of the observation model (5) used during multipitch tracking, we used a moderate pruning

[1] An implementation of the RAPT algorithm is provided by the Entropic speech processing system (ESPS) "get_f0" method.

factor of $N = 5000$.

The tracking performance of the algorithm is measured using $\bar{E}_{\text{Total}}$ [3], which is a slight modification of the error measure proposed in [6] to measure the influence of speaker assignment errors: $\bar{E}_{\text{Total}} = E_{01} + E_{02} + E_{10} + E_{12} + E_{20} + E_{21} + \bar{E}_{Gross} + \bar{E}_{Fine} + \bar{E}_{Perm}$. E_{ij} denotes the percentage of time frames, where i pitch points are missclassified as j pitch points. $\bar{E}_{Perm}$ measures the percentage of frames, where the voicing decision is correct, but the pitch values are not assigned to the correct speakers. $\bar{E}_{Gross}$ is the percentage of frames where the voicing decision is correct and no permutation error has occurred, but at least one detected pitch value deviates more than 20% from the reference pitch value. $\bar{E}_{Fine}$ is defined as $\bar{E}^{(1)}_{Fine} + \bar{E}^{(2)}_{Fine}$, where $\bar{E}^{(i)}_{Fine}$ denotes the frequency deviation in percent for speaker i, averaged over frames where no voicing, no gross and no permutation errors have occured.

Figure 2 shows the tracking performance of the gain adapted multipitch tracking method. We compare the results to the two cases where (i) no gain adaptation is performed, and (ii) the speaker models are adapted with the *true* gain applied during mixing ('perfect knowledge'). Furthermore, we show the performance of the multipitch tracking algorithm of Wu et al. [6]. Using the proposed method for gain adaptation, we reach in all cases almost the same performance as with gain levels known a-priori. The tracking performance decreases only moderately with an increasing gain level. The method of Wu et al. also exhibits a good robustness to large differences in gain level; the reason that the overall error is larger than for the proposed method is that the method of Wu et al. is speaker independent, whereas we are using speaker dependent models. Using a subset of 270 speech mixtures, we conducted time measurements on a PC with 6 cores @ 3.2 GHz and 12-GB main memory. All implementations are based on Matlab and MEX, and no multithreading was used. On average, one iteration (i.e. one EM step) of the proposed method for gain adaptation needed $1.78s$ for one second of speech, and subsequent multipitch tracking needed $8.79s$ per second of speech.

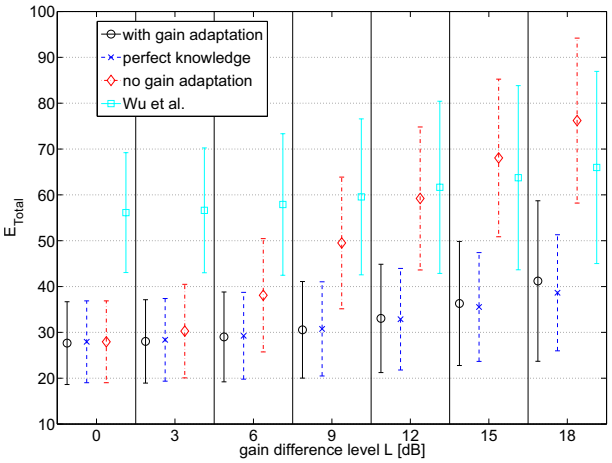


Figure 2: *Multipitch tracking results in terms of $\bar{E}_{Total}$. For each gain difference, we show the performance of 4 different methods: (i) multipitch tracking with EM-based gain adaptation, (ii) multipitch tracking with 'perfect' gain adaptation, i.e. with known gain levels, (iii) multipitch tracking without any gain adaptation and (iv) multipitch tracking using the method of Wu et al. [6].*

5. Conclusions

We introduced a novel EM-algorithm for automatic speaker gain adaptation. The computational complexity of the algorithm is reduced by using an efficient method for pruning unlikely MIXMAX state combinations. We applied the algorithm on a large set of speech mixtures with different mixing levels, and evaluated the performance of gain adapted speaker models used for probabilistic multipitch tracking. For mixing levels on a range from zero to 18dB, the multipitch tracking performance is almost equal to the case where the true gain factors are assumed to be known.

6. Acknowledgements

This work was funded by the Austrian Science Fund (project number S10604-N13).

7. References

[1] Talkin D., "A robust algorithm for speech tracking (RAPT)", in Kleijn W.B. and Paliwal K.K. [Ed], Speech Coding and Synthesis, 495-518, Elsevier Science, 1995.

[2] de Cheveigne A. and Kawahara H., "YIN, a fundamental frequency estimator for speech and music", J. Acoust. Soc. Am., vol. 111, no. 4, pp. 1917–1930, 2002.

[3] Wohlmayr M., Stark M. and Pernkopf F., "A Probabilistic Interaction Model for Multipitch Tracking With Factorial Hidden Markov Models", IEEE Transactions on Audio, Speech and Language Processing, vol. 19, no. 4, pp. 799-810, 2011.

[4] Nadas A., Nahamoo D. and Picheny M.A., "Speech recognition using noise-adaptive prototypes", IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 37, no. 10, pp. 1495-1503, 1989.

[5] Ghahramani Z. and Jordan M.I., "Factorial hidden Markov models", Machine Learning, vol. 29, no.2-3, pp. 245-273, 1997.

[6] Wu M., Wang D. and Brown G.J., "A Multipitch Tracking Algorithm for Noisy Speech", IEEE Transactions On Speech and Audio Processing, vol. 11, no. 3, pp. 229-241, 2003.

[7] Virtanen T., "Speech Recognition Using Factorial Hidden Markov Models for Separation in the Feature Space", Proceedings of the 7th Annual Conference of the International Speech Communication Association (Interspeech 2006), 2006.

[8] Radfar M.H., Wong W., Dansereau R.M. and Chan W.-Y., "Scaled Factorial Hidden Markov Models: A New Technique for Compensating Gain Differences in Model-Based Single Channel Speech Separation", International Conference on Acoustics, Speech and Signal Processing (ICASSP 2010), 2010.

[9] Rennie S.J., Hershey J.R and Olsen P.A., "Variational Loopy Belief Propagation for Multi-talker Speech Recognition", Proceedings of the 10th Annual Conference of the International Speech Communication Association (Interspeech 2009), 2009.

[10] Wohlmayr M., Peharz R. and Pernkopf F., "Efficient Implementation of Probabilistic Multi-Pitch Tracking", International Conference on Acoustics, Speech and Signal Processing (ICASSP 2011), 2011.

[11] Kschischangand F.R., Frey B.J. and Loeliger H., "Factor Graphs and the Sum-Product Algorithm", IEEE Transactions On Information Theory, vol. 47, no. 2, pp. 498-519, 2001.

[12] Dempster A., Laird N. and Rubin D., "Maximum likelihood from incomplete data via the EM algorithm", J. Roy. Statist. Soc., vol. B30, pp. 1-38, 1977.

[13] McLachlan G.J. and Basford K.E., "Mixture Models: Inference and Applications to Clustering", Marcel Dekker, 1988.

[14] Minka T.P., "Expectation-maximization as lower bound maximization", Technical Report, 1998, available online: http://research.microsoft.com/en-us/um/people/minka/papers/em.html.

[15] Cooke M.P., Barker M.P., Cunningham S.P. and Shao X., "An audio-visual corpus for speech perception and automatic speech recognition," J. Acoust. Soc. Amer., vol. 120, pp. 2421–2424, Nov 2005.