SPECTROGRAM-BASED FORMANT TRACKING VIA PARTICLE FILTERS

Yu Shi and Eric Chang

Microsoft Research Asia

{i-yshi,echang}@microsoft.com

ABSTRACT

This paper presents a particle-filtering method for estimating formant frequencies of speech signals from spectrograms. First, frequency bands corresponding to the analyzed formants are extracted via a two-step dynamic programming based algorithm. A particle-filtering method is then used to accurately locate formants in every formant areas based on the posterior pdf described by a set of support points with associated weights. Formant trajectories of voiced frames of a group of 81 utterances were manually tracked and labeled, partly for model training and partly for algorithm evaluation. In the experiments, the proposed method obtains average estimation errors of 72, 115, and 113 Hz for the first three formants, respectively, whereas LPC based method induces 118, 172, and 250 Hz deviations. The experimental results show that the formants estimated by the proposed method are quite reliable and the trajectories are more accurate than LPC.

1. INTRODUCTION

Formant frequency is one of useful speech parameters to be specified by a vocal tract shape or its movements in various pronunciations. However, capturing and tracking formants accurately from natural speech is difficult because of the variety of speech sounds. Typically, formant-tracking algorithms have three phases [1] [6]: signal conditioning (preemphasis), frame-dependent formant candidates generation, and tracking. For the second phase, linear prediction analysis (LPC) based methods have received considerable attention. Root-finding algorithms are employed to find the zeros of the LPC polynomial, or local maxima of the LPC envelope are searched using peak-picking techniques. However, the problem with root-finding algorithms is that the determination of formant frequencies and bandwidths is only successful for complex-conjugate poles and not for real poles, while peak-picking techniques are vulnerable to merged formants and spurious peaks.

This paper proposes a new formant tracking method based on gray-scale spectrograms. It is well known that the horizontal bands in spectrograms with higher energy show the formant positions. This is why many researchers are capable of distinguishing different phones just from spectrograms. In this paper, three spectrogram-based features consistent with the human sense are used to describe the likelihood of a frequency value to a formant at a certain time frame. In order to provide an optimum set of frequency segments each of which covers and concentrates on one formant spectrum region, a two step frequency range segmentation scheme based on dynamic programming is performed first.

Additionally, the developed algorithm is an application of sequential Monte Carlo methods (also known as particle filters) to track horizontal bands with higher energy in spectrogram. Particle filters were introduced to the vision community in the form of the CONDENSATION algorithm [9]. Improvements of a technical nature to it were provided by Isard and Black [10] (importance sampling). Recently, particle filtering has become a popular way to infer time-varying properties of a scene from images. The algorithm has already seen applications to nonlinear and non-Gaussian Bayesian tracking of various targets [5] [7] [8] [12]. Moreover, particle filtering and smoothing has also been introduced into the audio and speech enhancement community [4] [13]. In these applications, the speech signal is modeled as time-varying autoregressive (TVAR) equivalent process submerged in white Gaussian noise. By using sequential particle methods, an SNR improvement of the speech signals was achieved. Particle filtering has attracted much interest because it offers a framework for dynamic state estimation where the underlying probability density functions (pdfs) need not be Gaussian and state and measurement equations can be nonlinear. These situations are commonly encountered in vision and speech. In addition, the method has the ability for recovering from tracking misses in intermediate frames.

While particle filtering has many advantages in target tracking, to the best of our knowledge, the use of particle filters for formant tracking has not been proposed yet. Another contribution of this paper is that a set of formant trajectories are manually labeled, based on which performance of different formant estimation algorithms are evaluated numerically and easily. The proposed method is used in experimental tests that are carried out on Aurora2 clean speech database. According to the results, the presented approach produces reliable estimates of formant frequencies.

2. PROBLEM FORMULATION

We define the frequency position of the k-th formant at time t as $F_t^{(k)}$. Based on the Bayesian rule, to locate this position is to calculate the expectation $\bar{F}_t^{(k)}$ given the formant spectrum region sequence $R_{1:t}^{(k)}$ up to time t

$$\hat{F}_t^{(k)} = \bar{F}_t^{(k)} = \mathrm{E}[F_t^{(k)} | R_{1:t}^{(k)}] \tag{1}$$

where the posterior pdf $p(F_t^{(k)} | R_{1:t}^{(k)})$ is often unknown and non-Gaussian.

The speech data we have analyzed consist of a total of 81 clean female sentences randomly selected from the Aurora2 speech database. The speech signals are sampled at $f_s = 8$ KHz. Two kinds of spectrograms are generated in this paper. One is wide-band for formant labeling and the other is narrow-band for both model training and formant tracking. For wide-band spectrograms, the window size and frame step size are taken as 5 ms and 1 ms, whereas for narrow-band, they are 20 ms and 10 ms, respectively. Preemphasis and Hamming window are used before FFT for both types of spectrogram. In the formant labeling process, after being converted from log-energy to gray scale ranging from 0 to 255,

0-7803-7663-3/03/$17.00 ©2003 IEEE

the spectrograms are saved into bitmaps, based on which formant trajectories are then directly labeled on a special screen using a special pen. Voiced phone boundaries of the analyzing utterances are also manually labeled with the aid of spectrograms. In model training and formant tracking, the speech signals are limited to the frequency band $[f_u, f_s/2 - 200]$ Hz as in [14], for antialiasing and for reducing influence of the strong fundamental components on the estimation of $F^{(1)}$. According to the average vocal tract length of adult females, f_u takes as 300 Hz and the lowest three formants $F^{(1)}$-$F^{(3)}$ are estimated in the frequency band 0-4 KHz.

3. PARTICLE FILTERS FOR FORMANT TRACKING

Particle filtering is a technique for implementing a recursive Bayesian filter by Monte Carlo simulations. The key idea is to represent the required posterior pdf by a set of random samples with associated weights and to compute estimates based on them. As the number of samples becomes infinite, the Monte Carlo characterization becomes an equivalent representation to the usual functional description of the posterior pdf, and the particle filter approaches the optimal Bayesian estimate.

3.1. Particle filters in general

In this subsection, following [3], a general framework of the particle filtering methods is described. Consider the following dynamic system modelled in a state space form as

$$\begin{aligned} \mathbf{x}_t &= f_t(\mathbf{x}_{t-1}, \mathbf{v}_{t-1}) && (2) \\ \mathbf{z}_t &= h_t(\mathbf{x}_t, \mathbf{u}_t) && (3) \end{aligned}$$

where $\mathbf{x}_t$ is the state variable, $\mathbf{z}_t$ is the observation, and $\mathbf{v}_t$ and $\mathbf{u}_t$ are state and observation noises. These variables are either scalars or vectors. In the system, both state and measurement equations, f_t and h_t, are non-linear functions.

Let $\mathbf{z}_{1:t} = (\mathbf{z}_1, \cdots, \mathbf{z}_t)$. From a Bayesian perspective, the tracking problem is to recursively calculate some degree of belief in the state $\mathbf{x}_t$ at time t, taking different values, given the data $\mathbf{z}_{1:t}$ up to time t, and the optimal solution is

$$\hat{\mathbf{x}}_t = \bar{\mathbf{x}}_t = \mathrm{E}[\mathbf{x}_t|\mathbf{z}_{1:t}] = \int_{\mathbf{x}_t} \mathbf{x}_t p(\mathbf{x}_t|\mathbf{z}_{1:t}) \mathrm{d}\mathbf{x}_t \qquad (4)$$

Thus it is required to construct the pdf $p(\mathbf{x}_t|\mathbf{z}_{1:t})$.

It is defined that $\{\mathbf{x}_t^n, w_t^n\}_{n=1}^{N_s}$ is a random measure that characterizes the posterior pdf $p(\mathbf{x}_t|\mathbf{z}_{1:t})$, where $\{\mathbf{x}_t^n, n = 1, \cdots, N_s\}$ is a set of support points with associated normalized weights $\{w_t^n, n = 1, \cdots, N_s\}$ such that $\sum_{n=1}^{N_s} w_t^n = 1$. Then the posterior pdf at time t is approximated as

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) \approx \sum_{n=1}^{N_s} w_t^n \delta(\mathbf{x}_t - \mathbf{x}_t^n) \qquad (5)$$

where

$$w_t^n \propto \frac{p(\mathbf{x}_t^n|\mathbf{z}_{1:t})}{q(\mathbf{x}_t^n|\mathbf{z}_{1:t})} \qquad (6)$$

The *proposal function* $q(\cdot)$ is called an importance density from which $\mathbf{x}_t^n$ are easily generated. In recursive calculation, the weight-update equation is

$$w_t^n \propto w_{t-1}^n \frac{p(\mathbf{z}_t|\mathbf{x}_t^n) p(\mathbf{x}_t^n|\mathbf{x}_{t-1}^n)}{q(\mathbf{x}_t^n|\mathbf{x}_{1:t-1}^n, \mathbf{z}_t)} \qquad (7)$$

The most common and convenient choice of importance density is the conditional prior

$$q(\mathbf{x}_t^n|\mathbf{x}_{1:t-1}^n, \mathbf{z}_t) = q(\mathbf{x}_t^n|\mathbf{x}_{t-1}^n, \mathbf{z}_t) = p(\mathbf{x}_t^n|\mathbf{x}_{t-1}^n) \qquad (8)$$

since it is intuitive and simple to implement, yielding

$$w_t^n \propto w_{t-1}^n p(\mathbf{z}_t|\mathbf{x}_t^n) \qquad (9)$$

Then the weights need to be normalized. Subsequently, the optimal Bayesian estimate of state $\mathbf{x}_t$ is calculated as follows

$$\hat{\mathbf{x}}_t = \int_{\mathbf{x}_t} \mathbf{x}_t \sum_{n=1}^{N_s} w_t^n \delta(\mathbf{x}_t - \mathbf{x}_t^n) \mathrm{d}\mathbf{x}_t = \sum_{n=1}^{N_s} w_t^n \mathbf{x}_t^n \qquad (10)$$

A common problem with particle filtering is the degeneracy phenomenon, which is that the variance of the importance weights only increase over time. To reduce the effects of degeneracy, a resampling process is used. The basic idea of resampling is to eliminate particles that have small weights and to concentrate on particles with large weights. The resampling step involves selecting a number of, say N_n, children for each particle $\mathbf{x}_t^n$ such that $\sum_{n=1}^{N_s} N_n = N_s$. There is a variety of resampling schemes with varying performance in terms of the variance of the particles . The residual resampling [2] which has smaller variance of the particles and is computationally cheaper is used in this paper.

3.2. Formant tracking using particle filters

Taking the unknown k-th formant $F_t^{(k)}$ as the state variable and the formant spectrum region $R_t^{(k)}$ as the observation, the problem described in (1) is solved via the particle filtering method. Based on the framework described in the previous subsection and choosing (8) as the importance density, the optimal estimation in (1) becomes

$$\hat{F}_t^{(k)} = \mathrm{E}[F_t^{(k)}|R_t^{(k)}, \hat{F}_{t-1}^{(k)}] \qquad (11)$$

and therefore, we need to train the prior $p(F^{(k)})$, conditional prior $p(F_t^{(k)}|F_{t-1}^{(k)})$ and likelihood $p(R^{(k)}|F^{(k)})$, respectively. The likelihood $p(R^{(k)}|F^{(k)})$ is used to measure if the spectrum local features $L_{F^{(k)}}$ on frequency $F^{(k)}$ are similar to those of the key point in terms of the appearance. It is simplified to

$$p(R^{(k)}|F^{(k)}) = p(L_{F^{(k)}}|F^{(k)}) \qquad (12)$$

In this paper, all of the pdfs are modeled and learned as Gaussian or products of Gaussians. In details, the prior and conditional prior pdfs are written as

$$\begin{aligned} p(F^{(k)}) &\sim \mathcal{N}(F^{(k)}; \mu_{F^{(k)}}, \sigma_{F^{(k)}}) && (13) \\ p(F_t^{(k)}|F_{t-1}^{(k)}) &\sim \mathcal{N}(F_t^{(k)}; F_{t-1}^{(k)}, \sigma_{F_{t|t-1}^{(k)}}) && (14) \end{aligned}$$

and the likelihood distribution is decoupled into

$$p(L_{F^{(k)}}|F^{(k)}) = p(E_{F^{(k)}}|F^{(k)}) p(A_{F^{(k)}}|F^{(k)}) p(B_{F^{(k)}}|F^{(k)}) \qquad (15)$$

where $E_{F^{(k)}}$ means the average gray scale in the frequency band centered at $F^{(k)}$, and $A_{F^{(k)}}$ and $B_{F^{(k)}}$ are the rate-of-descent of the average gray scale when moving the band to a little bit lower and higher frequencies, respectively:

$$\begin{aligned} A_{F^{(k)}} &= E_{F^{(k)}}/E_{F^{(k)}-F_{\mathrm{sh}}} && (16) \\ B_{F^{(k)}} &= E_{F^{(k)}}/E_{F^{(k)}+F_{\mathrm{sh}}} && (17) \end{aligned}$$

where F_{sh} denotes the shift of $F^{(k)}$. In this paper, the frequency band covers 250 Hz and F_{sh} equals 180 Hz. The three pdfs in (15) are also modeled as Gaussian distributions.

By using the formant trajectories manually labeled, all the parameters in above Gaussian distributions are easily learned. Then the optimal estimate of the k-th formant $\hat{F}_t^{(k)}$ is computed by using the particle filtering method frame by frame. The iteration of the algorithm is described by Algorithm 1. In this paper, the number of particles N_s is set to 1000.

```
Algorithm 1: Particle Filters based Formant Tracking
For $t = 1 : T$
  • For $n = 1 : N_s$, assign weights: $w_t^n = N_s^{-1}$
  • For $n = 1 : N_s$,
      - Draw $F_t^{(k)}(n) \sim \begin{cases} p(F_t^{(k)}|F_{t-1}^{(k)}(n)) & t > 1 \\ p(F^{(k)}) & t = 1 \end{cases}$
      - Calculate $w_t^n = p(L_{F_t^{(k)}(n)}|F_t^{(k)}(n))$
  • For $n = 1 : N_s$, normalize weights: $\tilde{w}_t^n = w_t^n / \sum_{m=1}^{N_s} w_t^m$
  • Estimate $\hat{F}_t^{(k)} = \sum_{n=1}^{N_s} \tilde{w}_t^n F_t^{(k)}(n)$
  • Residual resampling
```

4. DYNAMIC PROGRAMMING FOR FREQUENCY RANGE SEGMENTATION

So far we have considered the tracking algorithm of the single k-th formant $F_t^{(k)}$ in region $R_t^{(k)}$ only. We now assume that K formant regions exist in the whole frequency axis, with boundaries $f_u \leq f_1^s < f_1^e \leq \cdots \leq f_K^s < f_K^e \leq f_s/2 - 200$ where f_k^s and f_k^e are the starting and ending position of the k-th segment. Similar to [15], we use a dynamic programming based algorithm for finding the optimum segment boundaries, but with a different objective function. The algorithm consists of two steps, a rough segmentation and a more accurate second step.

In the rough segmentation , a set of successive boundaries are found, i.e., $f_k^s = f_{k-1}^e$. To deal with the effects of the fundamental frequency and the possible higher formant presented by an unexpected longer vocal tract, two auxilary segments 0 and $K + 1$ are added. The local probability that one frequency point f_i belongs to segment k, say $R^{(k)}$, is defined as

$$p(f_i \in R^{(k)}) = p(L_{f_i}, f_i|k) = p(L_{f_i}|f_i, k)p(f_i|k) \qquad (18)$$

which is calculated by (15) and (13). The inter-frame continuity probability is taken as $p(f_i|\hat{F}_{t-1}^{(k)}, k)$ and computed by (14). To calculate the intra-frame segment-transition probability, a new variable is established by multiplying the local probability by the inter-frame continuity probability as $P_{f_i}^{(k)} = p(f_i \in R^{(k)})p(f_i|k, \hat{F}_{t-1}^{(k)})$. $P_{f_i}^{(k)}$ is then rescaled into range [0,1] in terms of frequency

$$P_{f_i}^{(k)} = \frac{P_{f_i}^{(k)} - \min_{f_i}(P_{f_i}^{(k)})}{\max_{f_i}(P_{f_i}^{(k)}) - \min_{f_i}(P_{f_i}^{(k)})} \qquad (19)$$

Then the intra-frame transition probability of segment k to $k+1$ at the frequency point f_i is calculated by using the product of three Gaussian distributions

$$p(f_{i+1} \in R^{(k+1)}|f_i \in R^{(k)}) \sim \mathcal{N}(P_{f_i}^{(k)}; \mu, \sigma_1)$$
$$\times \mathcal{N}(P_{f_{i+1}}^{(k+1)}; \mu, \sigma_1) \times \mathcal{N}(|P_{f_i}^{(k)} - P_{f_{i+1}}^{(k+1)}|; \mu, \sigma_2) \qquad (20)$$

The smaller the values of $P_{f_i}^{(k)}$, $P_{f_{i+1}}^{(k+1)}$, and $|P_{f_i}^{(k)} - P_{f_{i+1}}^{(k+1)}|$ are, the more possible a transition takes place, so the parameters in above equation are experimentally set as follows: $\mu = 0$, $\sigma_1 = 0.8$, and $\sigma_2 = 0.1$.

Once the local, inter-frame continuity and intra-frame transition probabilities are obtained, a dynamic-programming algorithm is used for frequency range segmentation within one frame. If we define a pair (f, k) as a point at frequency f and segment k, the dynamic programming algorithm searches the optimal path between starting point $(f_u, 0)$ and the end point $(f_s/2 - 200, K + 1)$ from left to right and bottom to up with the maximum accumulated probability (the sum of all log-probabilities on the path). Segments 0 and $K+1$ can be skipped, i.e., the starting point could be $(f_u, 1)$ and the end point could be $(f_s/2 - 200, K)$. Therefore, the number of segments is K to $K + 2$, and only the segments labeled as 1 to K are used later.

After the rough segmentation, the frequency range is separated into K to $K + 2$ successive parts. The problem of directly using this result is that the likelihood distributions used in the particle filtering step within one segment often have more than one peak, and therefore, the expectation in (11) will deviate from the real value. To make the segments more concentrated on the true formant, another dynamic programming algorithm is used. Suppose there are at most 3 subsegments in one segment. The local probability that one frequency point $f_i \in [f_k^s, f_k^e]$ in segment k belongs to subsegment j, say $R^{(k_j)}$, at time t is defined as

$$p(f_i \in R^{(k_j)}) = p(f_i \in R^{(k)}) \times \mathcal{N}(f_i|k, j) \qquad (21)$$

where the first item at the right side is calculated from (18), and the second item is computed from a Gaussian distribution with mean $\mu_{k_j} = \{f_k^s, (f_k^s + f_k^e)/2, f_k^e\}$ with respect to $j = \{1, 2, 3\}$ and variance $\sigma_k = (f_k^e - f_k^s)/4$. The inter-frame continuity probability is the same as in rough segmentation, and the intra-frame transition probability of subsegment j to $j + 1$ in segment k at the frequency point f_i is calculated using the similar function to (20) with a different variance of $\sigma_1 = 0.2$. Like the rough segmentation, the first and last subsegments ($j = 1$ and 3) can also be skipped. The final subsegment is the one which has the highest value of the variable in (19).

5. EXPERIMENTAL RESULTS

In this section, we present experimental results of formant estimation in order to illustrate the properties of the proposed algorithm. First, 72 sentences are randomly selected from the total 81 utterances whose formant trajectories have been manually labeled. Model parameters in particle filters are trained on the selected data. The remaining 9 digit strings are used to test the performance of the proposed algorithm. Fig. 1 shows an example of frequency-range segmentation superimposed on the spectrogram. This figure displays the result for the digit string *672* spoken by female talker *BR*. There are two types of horizontal lines for segment boundaries, rough boundaries (solid lines) and accurate boundaries (dash lines). This example indicates that the proposed algorithm for frequency range segmentation is reliable for extracting the exact formant areas.

Fig. 2 presents examples of formant tracks superimposed on the spectrograms. The left column is for digit string *357* spoken by female talker *ES*, whereas the right column is for digit string *1532o* spoken by female talker *AI*. Fig. 2(a) shows the manually labeled trajectories, Fig. 2(b) displays the raw (no-smoothing) formant contours estimated by the particle filtering method proposed

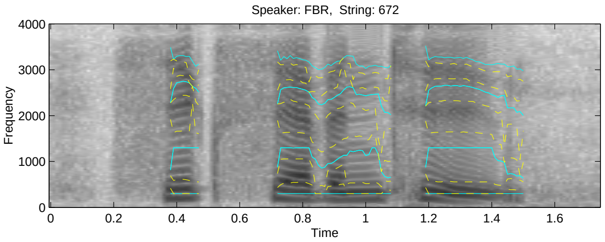
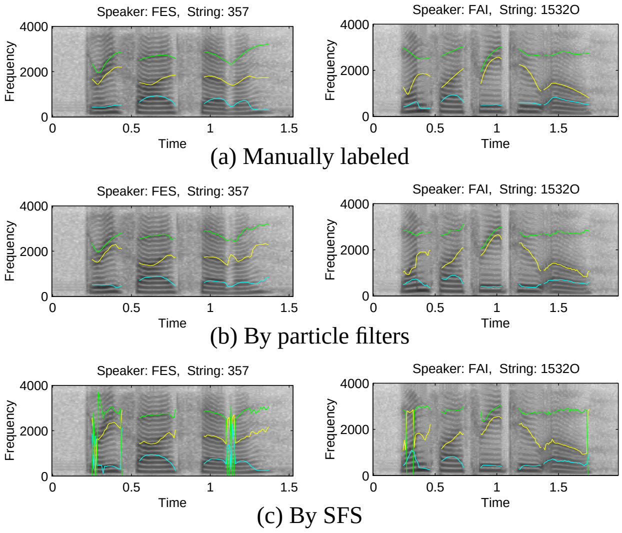


Fig. 1. Example of frequency range segmentation.

in this paper, and Fig. 2(c) gives the corresponding results computed by *Speech Filing System* (SFS) [11] which uses an LPC based method as references. Though most part of two kinds of trajectories in (b) and (c) are very similar to each other, we find more gross errors in (c) than in (b). In contrast with the LPC method, the formant trajectories by the proposed method are continuous and legible. Furthermore, the estimated formant trajectories of the 9 test sentences are compared with the manually labeled formant values on the frame level. The mean and standard deviation of the estimation errors are shown in the first line in Table 1. This experiment has been repeated for 5 times with different training and test data selected at random. Estimation errors by the other 4 experiments are also listed in the Table. As a reference, error information of SFS for each experiment is shown below the corresponding line by particle filters. The average mean errors by particle filtering method are 72, 115, and 113 Hz for the first three formants, where as the average mean errors by SFS are 118, 172, and 250 Hz, respectively. Fig. 2(c) demonstrates that there are several frames which do not have enough formants obtained by SFS due to the shortage of LPC and therefore contain formant alignment errors. To calculate the estimation error properly, formants in such a frame are aligned and the vacancy is filled by a valid value in a near frame. The numbers in the table indicate that the formant frequencies by the proposed method are more accurate than LPC. More experimental results can be found at http://research.microsoft.com/˜echang/projects/particle_formant/particle_formant.htm.



(a) Manually labeled

(b) By particle filters

(c) By SFS

Fig. 2. Spectrograms and formant frequency contours.

6. CONCLUSION

We have proposed a particle filtering method for estimating formant frequencies of speech signals from spectrograms. First, frequency segments corresponding to the analyzing formants are extracted via a two-step dynamic programming based algorithm. The first step is a rough segmentation which only divides the frequency axis into several parts, whereas the second step extracts from each part the most likely area which the likelihood distribution concentrates on, which is benefit to particle filters. A particle filtering method is then used to accurately locate formants in every segments based on the posterior pdf described by a set of support points with associated weights. In the experiment, formant trajectories of voiced frames of 81 utterances are manually tracked and labeled for both model training and algorithm evaluation. In the experiments, we obtain average estimation errors of 72, 115, and 113 Hz for the first three formants by particle filtering, and 118, 172, and 250 by SFS. The experimental results show that the formants estimated by the proposed method are quite reliable. The trajectories convince us that they are caused by the correct phoneme sequence of a given word. In addition, the proposed method is superior to LPC in the stability of the estimation.

Table 1. Error Information of Formant Estimation

Exp	#frames	$F^{(1)}$(Hz)		$F^{(2)}$(Hz)		$F^{(3)}$(Hz)	
		mean	std	mean	std	mean	std
1	769	73	70	103	136	91	107
S_1	769	117	118	152	324	219	499
2	854	68	58	109	119	106	108
S_2	854	113	111	140	250	199	359
3	880	76	86	110	97	148	236
S_3	880	154	208	283	533	421	875
4	510	63	59	151	271	94	98
S_4	510	106	109	149	286	197	403
5	1003	80	80	103	104	125	167
S_5	1003	101	94	137	233	212	341
Average		72	71	115	145	113	143
S_Average		118	128	172	325	250	495

7. REFERENCES

[1] A. Acero. “Formant analysis and synthesis using hidden Markov models,” *Proc. Eurospeech’99*.

[2] C. Andrieu, N. Freitas, and A. Doucet. “Sequential MCMC for Bayesian Model Selection,” *IEEE Signal Processing Workshop on Higher Order Statistics*, 1999.

[3] M. Arulampalam et al. “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *IEEE Trans. on SP*, 2002, pp. 174-188.

[4] W. Fong et al. “Monte Carlo smoothing with application to audio signal enhancement,” *IEEE Trans. on SP*, 2002, pp. 438-448.

[5] F. Gustafsson et al. “Particle filters for positioning, navigation, and tracking,” *IEEE Trans. on SP*, 2002, pp. 425-437.

[6] X. D. Huang, A. Acero, H. Hon. *Spoken Language Processing*.

[7] C. Hue, J. Cadre, and P. Pérez. “A particle filter to track multiple objects,” *IEEE Workshop on Multi-Object Tracking*, 2001.

[8] C. Hue, J. Cadre, and P. Pérez. “Sequential Monte Carlo methods for multiple target tracking and data fusion,” *IEEE Trans. on SP*, 2002, pp. 309-325.

[9] M. Isard and A. Blake. “CONDENSATION conditional density propagation for visual tracking,” *International J. Computer Vision*, 1998.

[10] M. Isard and A. Blake. “ICONDENSATION: Unifying low-level and high-level tracking in a stochastic framework,” *Proc. ECCV1998*.

[11] http://www.phon.ucl.ac.uk/resource/sfs/.

[12] J. Vermaak et al. “Sequential Monte Carlo fusion of sound and vision for speaker tracking,” *Proc. ICCV’01*.

[13] J. Vermaak et al. “Particle methods for Bayesian modeling and enhancement of speech signals,” *IEEE Trans. on SAP*, 2002, pp. 173-185.

[14] A. Watanabe. “Formant estimation method using inverse-filter control,” *IEEE Trans. on SAP*, 2001, pp. 317-326.

[15] L. Welling and H. Ney. “Formant estimation for speech recognition,” *IEEE Trans. on SAP*, 1998, pp. 36-48.