BAG OF ARCS: NEW REPRESENTATION OF SPEECH SEGMENT FEATURES BASED ON FINITE STATE MACHINES

Shinji Watanabe†, Yotaro Kubo, Takanobu Oba, Takaaki Hori, and Atsushi Nakamura

NTT Communication Science Laboratories, NTT Corporation, Kyoto, Japan

ABSTRACT

This paper proposes a new feature representation, Bag Of Arcs (BOA) for speech segments. A speech segment in BOA is simply represented as a set of counts for unique arcs in a *finite state machine*. Similar to the Bag Of Words model (BOW), BOA disregards the order of arcs, and thus, efficiently models speech segments. A strong motivation to use BOA is provided by a fact that the BOA representation is tightly connected to the output of a Weighted Finite State Transducer (WFST) based ASR decoder. Thus, BOA directly represents elements in the search network of a WFST-based ASR decoder, and can include information about context-dependent HMM topologies, lexicons, and back-off smoothed n-gram networks. In addition, the counts of BOA are accumulated by using the WFST decoder output directly, and we do not require an additional overhead and a change of decoding algorithms to extract the features. Consequently, we can combine the ASR decoder and post-processing without a process to extract word features from the decoder outputs or re-compiling WFST networks. We show the effectiveness of the proposed approach for some ASR post-processing applications in utterance classification experiments, and in speaker adaptation experiments by achieving absolute 1% improvement in WER from baseline results. We also show examples of latent semantic analysis for BOA by using latent Dirichlet allocation.

***Index Terms*—** Speech segment feature, finite state machine, Bag Of Arcs (BOA), speaker recognition, utterance classification

1. INTRODUCTION

The number of speech archives has grown hugely as storage capacity has increased and cloud services have been widespread. This has led to the need for an efficient technique for modeling speech segments for classification and utilization of speech processing including Automatic Speech Recognition (ASR). This paper proposes a new representation of a speech feature, which we call Bag Of Arcs (BOA) for speech segments. A speech segment in BOA is simply represented as a set of counts for unique arcs in a *finite state machine* used in an ASR decoder [1, 2].

The proposed BOA model is a generalized extension of the Bag Of Words (BOW) model, which is widely employed for document classification in the field of natural language processing [3–5]. The BOW model is used as a generative model of a document in a naive Bayes classifier and latent topic model. The BOW representation is also used as a feature in the Support Vector Machine (SVM). Similar to BOW, BOA disregards the order of arcs, and thus, efficiently models speech segments. A strong motivation to use BOA is provided by a fact that the BOA representation is tightly connected to the output of a Weighted Finite State Transducer (WFST) based ASR decoder. Namely, BOA directly represents elements in the search network of a WFST-based ASR decoder, and can include information about context-dependent HMM topologies, lexicons, and back-off smoothed n-gram networks, whereas BOW only possesses word unigram information. In addition, the counts of BOA are accumulated by using the WFST decoder output directly, and we do not require an additional overhead and a change of decoding algorithms to extract the features. This tight connection enables us to simply combine the ASR decoder and post-processing of the ASR outputs via the BOA representation. Suppose, for example, a type of post-processing of ASR based on the conventional BOW representation, we require a step of extracting word features from the decoder outputs. In addition, when we reflect the post-processing results to the ASR decoder (e.g., language model adaptation), we have to re-compile WFST networks using the obtained BOW model, which is time-consuming in the large vocabulary speech recognition setup. These overhead processes are not required for the proposed BOA representation. Moreover, the proposed approach has the potential to be used in various spoken language applications (e.g., spoken document retrieval), and other finite state machine based processings than ASR (e.g., statistical machine translation, and text summarization).

We show the effectiveness of the proposed approach for some ASR post-processing applications in utterance classification experiments and speech recognition experiments using speaker adaptation. We also show examples of latent semantic analysis for BOA by using Latent Dirichlet Allocation (LDA) [5], which provided topic word categorization different from the BOW case, as a result of reflecting the information about context-dependent HMM topologies and lexicons.

2. FORMULATION

This section deals with speech segments in a probabilistic manner to make it possible to use machine learning techniques[1]. In this paper, we assume a speech segment to be an utterance or segment that has some linguistic meaning.

2.1. Generative model of speech segments

Let $\mathbf{X}(u) = \{\mathbf{x}_t \in \mathbb{R}^D | t = 1, \cdots, T(u)\}$ be a D dimensional observation vector (e.g. a Mel-Frequency Cepstral Coefficient (MFCC)) sequence of speech segment u. $T(u)$ denotes the number of frames in speech segment u. Now, we deal with the problem of classifying speech segment u to category C_u, and consider the posterior distribution $p(C_u|\mathbf{X}(u))$ by reference to the generative model of a document in the naive Bayes classifier [3].

Since we cannot generally obtain an N-length word sequence $\mathbf{W} = \{w_n \in \mathbb{N} | n = 1, \cdots, N\}$ given speech inputs unlike BOW, we use the probabilistic sum rule to introduce the word sequence $\mathbf{W}$ as follows:

$$p(C_u|\mathbf{X}(u)) \approx \sum_{\mathbf{W}} p(C_u|\mathbf{W}) p(\mathbf{W}|\mathbf{X}(u)), \quad (1)$$

†Shinji Watanabe is now with Mitsubishi Electric Research Laboratories (MERL), Cambridge Massachusetts, USA.

[1]We can also use non-probabilistic features based on our proposed BOA representation.

Here, we use the Bayes theorem and conditional independence assumption, and approximate that $p(C_u|\mathbf{W}) \approx p(C_u|\mathbf{W}, \mathbf{X}(u))$. Eq. (1) is expressed by the posterior distribution given a word sequence used in the BOW formulation [3] and the posterior distribution of a word sequence, which can be obtained with an ASR decoder. For simplicity, we use the Viterbi approximation ($\hat{\mathbf{W}} = \text{argmax}_{\mathbf{W}}\, p(\mathbf{W}|\mathbf{X}(u))$ and $p(\hat{\mathbf{W}}|\mathbf{X}(u)) = \delta_{\mathbf{W}\hat{\mathbf{W}}}$)[2] and obtain the following equation:

$$p(C_u|\mathbf{X}(u)) \approx p(C_u|\hat{\mathbf{W}}) = \prod_{n=1}^{\hat{N}(u)} p(\hat{w}_n|C_u)p(C_u). \quad (2)$$

From the equation, the posterior distribution of a speech segment can be approximately represented by the word distribution of the 1-best recognition result for $\mathbf{X}(u)$ by using an ASR decoder.

2.2. Finite state machine based representation

A probabilistic speech recognition approach obtains an appropriate word sequence from the conditional distribution $p(\mathbf{W}|\mathbf{X}(u))$, i.e., $\text{argmax}_{\mathbf{W}}\, p(\mathbf{W}|\mathbf{X}(u))$. However, it is difficult to deal directly with $p(\mathbf{W}|\mathbf{X}(u))$, and the joint distribution $p(\mathbf{W}, \mathbf{X}(u))$ is generally used, which is obtained by composing acoustic, lexicon, and language model networks.

WFST-based decoders deal with this joint distribution by considering the composed network statically in a decoding graph. For example, for general speech recognition, we prepare a transducer network $\mathcal{R}$ composed of an HMM network, a phoneme-context network, a lexicon network, and a grammar network. Then, WFST decoders focus on a path in WFST $\mathcal{R}$, which corresponds to arc sequence ($\mathbf{a} = \{a_1, \cdots, a_m, \cdots a_M\} \in \mathcal{R}$), instead of HMM state and word sequences. Arc a_m has an input symbol $i[a_m]$ consisting of an HMM state id, an output symbol $o[a_m]$ consisting of a word id, and weight $h[a_m]$[3]. Therefore, arc sequence $\mathbf{a}$ represents a transducer from HMM state sequence $\mathbf{s} = \{i[a_1], \cdots, i[a_M]\}$ to word sequence $\mathbf{W} = \{o[a_1], \cdots, o[a_M]\}$. Then, the decoding process becomes a search problem concerned with finding an appropriate arc sequence $\bar{\mathbf{a}}$ among all possible $\mathbf{a}$ in WFST $\mathcal{R}$:

$$\bar{\mathbf{a}} = \underset{\mathbf{a}\in\mathcal{R}}{\text{argmax}}\, w(\mathbf{X}(u), \mathbf{a}). \quad (3)$$

$w(\mathbf{X}(u), \mathbf{a})$ is the unnormalized likelihood score given observation vectors $\mathbf{X}(u)$ and hypothesized arc sequence $\mathbf{a}$. If we normalize $w(\mathbf{X}(u), \mathbf{a})$ for all possible arc sequences in recognition network $\mathcal{R}$, this score becomes a conditional distribution probability. The corresponding word sequence is obtained by $\bar{\mathbf{W}} = \{o[\bar{a}_1], \cdots, o[\bar{a}_M]\}$.

The idea of the proposed approach replaces the word-based posterior distribution in Eq. (2) with the arc based distribution as follows:

$$p(C_u|\mathbf{X}(u)) \approx p(C_u|\bar{\mathbf{a}}) \approx \prod_{m=1}^{\bar{M}(u)} p(\bar{a}_m|C_u)p(C_u). \quad (4)$$

$\bar{M}(u)$ is the number of arcs for sequence $\bar{a}$. This representation directly reflects the search network of a WFST-based ASR decoder.

[2]We can also use the lattice or n-best based approaches for the 1-best Viterbi approximation. The word lattice based formulation within a finite state machine framework is discussed in [6].

[3]If we use factorization, input and output symbols may include sequences of HMM state and word ids, respectively, and an epsilon transition (ϕ) is also allowed in input and output symbols.

2.3. Multinomial distribution for arc output distribution

Arc uni-gram model

By setting a multinomial distribution for the output distribution, we can obtain the following distribution:

$$p(C_u|\mathbf{X}(u)) \approx \mathcal{M}(\bar{\mathbf{a}}|\{\theta_l\}_{l=1}^{L}), \quad (5)$$

where

$$\mathcal{M}(\bar{\mathbf{a}}|\{\theta_l\}_{l=1}^{L}) \propto \prod_{m=1}^{\bar{M}(u)} \theta_{s[\bar{a}_m]} = \prod_{l=1}^{L} \theta_l^{\bar{n}_l(u)}. \quad (6)$$

l means a unique index of arcs in a finite state machine, and L is the number of unique arcs. $s[a_m]$ is an arc identifier function to output the corresponding arc index of a_m. $n_l(u)$ denotes a count of arc l that appeared in speech segment u and that is obtained from the decoded result. From the maximum likelihood estimation and by considering a discounting method, we can derive the following multinomial distribution parameter:

$$\theta_l = \frac{\sum_u n_l(u) + \delta}{\sum_u \sum_{l=1}^{L} n_l(u) + \delta L}. \quad (7)$$

In this paper, we use the additive smoothing method with discounting parameter δ. Thus, we can derive the generative model of speech segments by using the arc representation based on the WFST decoding framework, which is used for WFST adaptation (Section 3.2) and latent semantic analysis (Section 3.3).

Latent topic model

This paper considers word clustering based on a latent topic model that uses arc and segment variables instead of word and document variables in the original latent topic model. The generative model represents the occurrence probability of arc sequence $\mathbf{a}$ given segment u without using classification category C_u. This is decomposed into topic and arc probabilities as follows:

$$\begin{aligned}\prod_{m=1}^{M(u)} p(a_m|u) &= \prod_{m=1}^{M(u)} \sum_{k=1}^{K} \underbrace{p(a_m|k,u)}_{\text{Arc probability}} \underbrace{p(k|u)}_{\text{Topic probability}} \\ &= \prod_{m=1}^{M(u)} \sum_{k=1}^{K} \theta_{uks[a_m]}\phi_{uk},\end{aligned} \quad (8)$$

where K is the number of latent topics. We also use the conditional independence assumption for $\mathbf{a}$. In LDA, we set a Dirichlet distribution for the prior distribution of the topic probability ϕ_{uk}, i.e., $p(\phi_{uk}) \propto \prod_{k=1}^{K} \phi_{uk}^{\gamma_k - 1}$ with hyper-parameter γ_k.

2.4. Discriminative approach

Similar to natural language processing, we can use the BOA representation as a feature of discriminative models (e.g., SVM and averaged perceptron [7]). In the discriminative approaches, we can also introduce other features (e.g., acoustic and language scores, Gaussian statistics, duration) in addition to unique arc counts. These are used in the WFST based discriminative model approaches [8, 9].

3. EXPERIMENTS

We show the effectiveness of the proposed approach for lecture classification experiments and speech recognition experiments using speaker adaptation. In the lecture classification experiments, we

Table 1. Lecture classification error rates (%) of BOA (proposal) and BOW.

	BOW	BOA (proposal)
Multi-class SVM (Dev.)	52.36 %	51.36 %
Multi-class SVM (Eval.)	52.48 %	51.42 %
Averaged perceptron (Dev.)	59.26 %	58.36 %
Averaged perceptron (Eval.)	59.20 %	58.46 %

used an MIT OpenCourseWare (OCW) task [10]. In speech recognition experiments, we used MIT-OCW and Corpus of Spontaneous Japanese (CSJ) tasks [11]. We also show examples of latent semantic analysis for BOA with LDA by using MIT-OCW. In all the experiments, we used an utterance unit obtained by voice activity detection as a speech segment.

3.1. Lecture classification

We demonstrated lecture classification task experiments by using the proposed BOA and conventional BOW features, which classified each utterance according to its corresponding lecture. MIT-OCW contains a total of 105 lectures and 57,376 speech utterances. We used 47,376 utterances as a training set, 5,000 utterances as a development set, and the rest 5,000 utterances as an evaluation set. The BOW and BOA were obtained from references (not ASR results) to simplify the experimental discussion. The BOA features were obtained by composing the finite state acceptors, which accept the reference word sequences, and the WFST used for the ASR decoding. In these experiments, we used two types of classifiers, the multi class SVM with a linear kernel and the averaged perceptron. The development set was used to tune the trade-off parameter between the training error and the margin in the multi class SVM, and the number of iterations in the averaged perceptron.

Table 1 shows the lecture classification error rates. This classification problem was a very difficult task because some of the utterances are composed only of fillers, simple words (e.g., "yes", "this one", "you know"), which do not have the information needed for this lecture classification. Even in this situation, both classifiers archived 50 % error rate ranges. The table shows that BOA was superior to BOW by around 1 % in both classifiers' results. The difference between BOW and BOA is that BOA considers the context-dependent HMM state and lexical information in addition to the word information used in BOW. Therefore, we can conclude that the superiority of BOA derives from its sparser representation based on the additional information, which can obtain the classification boundary more separably[4]. In addition, Figure 1 shows the development set results of BOW and BOA for every iteration by using the averaged perceptron. The result shows that BOA steadily improved the performance from BOW in every iteration, and this result also supports the superiority of BOA.

3.2. WFST adaptation

We demonstrated the unsupervised weight vector adaptation of WFSTs in speech recognition based on the BOA representation. The unsupervised weight adaptation of a WFST is realized by employing the following steps:

1) Arc sequences are obtained by using a WFST decoder.
2) The uni-gram probabilities of the multinomial parameters are computed based on Eq. (7).

[4]However, the baseline performance of this topic classification task is rather low, and we need more experiments to validate our conclusion.

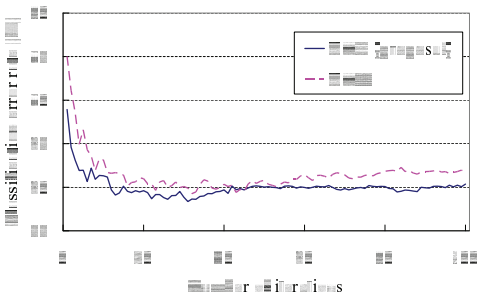

Fig. 1. Lecture classification error rate for each iteration by using averaged perceptron.

Table 2. Experimental conditions for an MIT-OCW task.

Sampling rate/quantization	16 kHz / 16 bit
Observation vector	12 order MFCC with energy
(39 dimensions)	$+\Delta+\Delta\Delta$ (CMS)
Window	Hamming
Frame size/shift	25/10 ms
Num. of phoneme categories	52
Num. of clustered HMM states	2,565 (3 left-to-right HMM states)
Num. of mixture components / state	32
Language model	3-gram (KN discounting)
Vocabulary size	44K

3) The obtained parameters are linearly interpolated with the original weights in the WFST.
4) Recognition results are obtained by using a WFST decoder with the adapted WFST.

Unlike the conventional language model adaptation within a WFST framework, we do not need to use a uni-gram rescaling technique [12] or to construct a new language model to obtain the new WFST [13].

We used an MIT-OCW task [10] and a CSJ task [11]. The experimental conditions for the MIT-OCW task are summarized in Table 2. The initial acoustic model was constructed by using variational Bayesian triphone clustering [14] and differenced Maximum Mutual Information (dMMI) training [15]. The evaluation set consisted of 8 lectures (6,989 utterances, 72,159 words, and 7.8 hours). The experimental conditions for the CSJ task are summarized in Table 3. The initial acoustic and language models were trained by discriminative approaches [15, 16]. We used CSJ testset 2 as a development set (10 lectures, 794 utterances, 26,798 words, and 2.2 hours) and CSJ testset 1 as an evaluation set (10 lectures, 977 utterances, 26,329 words, and 2.0 hours). In the CSJ experiment, the utterances were automatically segmented from the lectures using non-linear Kalman filtering based VAD [17]. Both the experiments used a one-pass WFST-based decoder that employs a pair of WFSTs for composition during decoding by a fast on-the-fly composition technique [2]. The discounting parameter, δ in Eq. (7), is set at 0.01 in all the adaptation experiments. The scaling parameters in the linear interpolation in Step 3) were empirically set at 0.02 in the MIT task, and were set at 0.03 in the CSJ task determined by the development set of the CSJ task.

Table 4 shows the Word Error Rates (WERs) obtained by the proposed BOA-based adaptation. It also includes the WERs by the conventional BOW-based adaptation, and by combination of the BOW- and BOA-based adaptations in the CSJ task. As we see from the table, the WERs after the BOA were comparable to those after the BOW. Given the fact that the BOA did not require recompilation of WFSTs (it took several seconds and required several hundred megabytes of memory), we can take the results as meaning that the BOA produced a comparable improvement at much less computational cost than the BOW. In addition, the WERs were

Table 3. Experimental conditions for a CSJ task.

Sampling rate/quantization	16 kHz / 16 bit
Observation vector	12 order MFCC with energy
(39 dimensions)	$+\Delta+\Delta\Delta$ (CMS)
Window	Hamming
Frame size/shift	25/10 ms
Num. of phoneme categories	43
Num. of clustered HMM states	5,000 (3 left-to-right HMM states)
Num. of mixture components / state	32
Language model	3-gram (Good Turing)
Vocabulary size	100K

Table 4. Word error rates (%) for WFST adaptation experiments.

	Baseline	BOW	BOA	BOA+BOW
CSJ-Dev.	17.6 %	16.5 %	16.5 %	16.2 %
CSJ-Eval.	20.9 %	19.9 %	19.8 %	19.6 %
MIT-OCW	27.9 %	-	26.9 %	-

further reduced by the combinatorial approach from those by the separate use of either the BOA or BOW, which suggests the BOA- and BOW-based adaptation could work complementarily. From these results, we can conclude that the proposed BOA representation is advantageous for weight adaptation in WFST frameworks.

3.3. Latent semantic analysis

Finally, we demonstrated a latent semantic analysis in BOA and BOW representations as a preliminary experiment. We used LDA based on the stochastic EM algorithm [13]. The obtained word clusters by BOW and BOA are shown in Tables 5 and 6, respectively. Table 5 shows that topic-wise clustering was achieved by using the BOW model. In contrast, Table 6 shows that the clustering was achieved for the acoustic or phonetic similarity among words by using the BOA model. This is because BOA involves information from HMM state and pronunciation lexicon. Thus, BOA is different from BOW in that it can reflect the hierarchical structure in speech representation for ASR. Future work will apply BOA to topic based language model adaptation [12,13] to show the effectiveness of BOA in a latent semantic analysis in terms of quantitative evaluation.

4. SUMMARY

This paper proposed the Bag Of Arc (BOA) model as a new representation of speech segments based on a finite state machine framework. We provided the formulation of three typical applications of BOA (document classification, adaptation of generative models and latent semantic analysis) used in the conventional Bag Of Word (BOW) model. Experiments showed the superiority of BOA to BOW in these applications. The proposed approach has the potential to be used in various spoken language applications (e.g., spoken document retrieval), and future work will focus on these applications.

5. ACKNOWLEDGMENT

We thank the MIT Spoken Language Systems Group for helping us to perform speech recognition experiments based on MIT-OCW.

Table 5. Top 10 high probability nouns in word probabilities in the Bag of Word (BOW) representation.

(∼*Classical mechanics*)	(∼*Astronomy*)	(∼*(Time) unit*)
m	light	percent
energy	degrees	time
force	angle	dollars
mass	frequency	times
point	energy	minutes
velocity	direction	day
direction	waves	bit
v	sun	year
times	star	hour
speed	speed	half

Table 6. Top 10 high probability words in arc probabilities in the Bag of Arc (BOA) representation. The words were extracted from the output symbols of the WFST arcs.

important	doing	under
forty	engineering	undergraduate
improve	selling	underlying
fourteen	jack	underneath
importance	engineer	unlike
impossible	japanese	undergraduates
importantly	monitoring	underground
improvement	jackson	unquote
ford	arguing	undoubtedly
improving	referring	unpolarized

6. REFERENCES

[1] M. Mohri, F. Pereira, and M. Riley, "Weighted finite-state transducers in speech recognition," in *Proc. ASR'00*, 2000, pp. 97–106.

[2] T. Hori, C. Hori, Y. Minami, and A. Nakamura, "Efficient WFST-based one-pass decoding with on-the-fly hypothesis rescoring in extremely large vocabulary continuous speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1352–1365, 2007.

[3] D. Lewis, "Naive (Bayes) at forty: The independence assumption in information retrieval," *Machine Learning: ECML-98*, pp. 4–15, 1998.

[4] T. Joachims, "Learning to classify text using support vector machines: Methods, theory, and algorithms," *Computational Linguistics*, vol. 29, no. 4, pp. 656–664, 2002.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation.," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[6] C. Cortes, P. Haffner, and M. Mohri, "Rational kernels: Theory and algorithms," *The Journal of Machine Learning Research*, vol. 5, pp. 1035–1062, 2004.

[7] M. Collins, "Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms," in *Proc. EMNLP'02*, 2002.

[8] S. Watanabe, T. Hori, and A. Nakamura, "Large vocabulary continuous speech recognition using WFST-based linear classifier for structured data," in *Proc. of Interspeech'10*, 2010, pp. 346–349.

[9] M. Lehr and I. Shafran, "Learning a discriminative weighted finite-state transducer for speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, pp. 1360–1367, 2011.

[10] J. Glass, T. J. Hazen, S. Cyphers, I. Malioutov, D. Huynh, and R. Barzilay, "Recent progress in the MIT Spoken Lecture Processing Project," in *Proc. Interspeech'07*, 2007, pp. 2553–2556.

[11] K. Maekawa, H. Koiso, S. Furui, and H. Isahara, "Spontaneous speech corpus of Japanese," in *Proceedings of LREC2000*, 2000, vol. 2, pp. 947–952.

[12] D. Gildea and T. Hofmann, "Topic-based language models using EM," in *Proc. Eurospeech'99*, 1999, pp. 2167–2170.

[13] S. Watanabe, T. Iwata, T. Hori, A. Sako, and Y. Ariki, "Topic tracking language model for speech recognition," *Computer Speech and Language*, vol. 25, no. 2, pp. 440–461, 2011.

[14] S. Watanabe, Y. Minami, A. Nakamura, and N. Ueda, "Variational Bayesian estimation and clustering for speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 12, pp. 365–381, 2004.

[15] E. McDermott, S. Watanabe, and A. Nakamura, "Discrminative training based on an integrated view of MPE and MMI in margin and error space," in *Proc. ICASSP'10*, 2010, pp. 4894–4897.

[16] T. Oba, T. Hori, and A. Nakamura, "A study of efficient discriminative word sequences for reranking of recognition results based on n-gram counts," in *Proc. Interspeech'07*, 2007, pp. 1753–1756.

[17] M. Fujimoto, K. Ishizuka, and H. Kato, "Noise robust voice activity detection based on statistical model and parallel non-linear Kalman filtering," in *Proc. ICASSP'07*, 2007, vol. 4, pp. 797–800.