HMM-BASED PSEUDO-CLEAN SPEECH SYNTHESIS FOR SPLICE ALGORITHM

Jun Du, Yu Hu, Li-Rong Dai, Ren-Hua Wang

University of Science and Technology of China, Hefei, P. R. China, 230027

{unuedjwj,jadefox}@ustc.edu, {lrdai,rhw}@ustc.edu.cn

ABSTRACT

In this paper, we present a novel approach to relax the constraint of stereo-data which is needed in a series of algorithms for noise-robust speech recognition. As a demonstration in SPLICE algorithm, we generate the pseudo-clean features to replace the ideal clean features from one of the stereo channels, by using HMM-based speech synthesis. Experimental results on aurora2 database show that the performance of our approach is comparable with that of SPLICE. Further improvements are achieved by concatenating a bias adaptation algorithm to handle unknown environments. Relative word error rate reductions of 66% and 24% are achieved over the baseline systems in the clean-training and multi-training conditions, respectively.

Index Terms: noisy speech recognition, SPLICE, HMM-based speech synthesis

1. INTRODUCTION

With the progress of automatic speech recognition (ASR), the noise robustness of speech recognizers attracts more and more attentions for practical recognition systems. Many techniques [1] have been proposed to handle the difficult problem of mismatch between training and application conditions. However, the performance achieved by most of them are unable to reach that under matched training and testing conditions. Recently, a feature compensation technique called Stereo-based Piecewise Linear Compensation for Environments (SPLICE) [2] is demonstrated that this performance limit could be surpassed. SPLICE is an extension of the feature compensation techniques [3, 4] developed at Carnegie Mellon University (CMU) in the past decade. Requirement of stereo-data and handling unseen environments are two main obstacles in SPLICE algorithm.

First, to remove the requirement of stereo-data, variations of SPLICE are proposed. In [5, 6], *stochastic vector mapping* (SVM), which represents the mapping from the "corrupted" speech to "clean" by a simple transformation, is a generalized definition of SPLICE. And a joint training of the parameters of SVM function and HMMs is implemented by adopting maximum likelihood (ML) or minimum classification error (MCE) criteria. MMI-SPLICE [7] is much like SPLICE, but without the need for target clean features. Instead of learning a speech enhancement function, it learns to increase recognition accuracy directly with a maximum mutual information (MMI) objective function. FMPE [8], a kind of discriminatively trained features, is related with SPLICE to a certain extent [9]. To handling unseen environments, a unsupervised online adaptation of SVM is presented in [10].

The motivation of our approach is to relax the constraint of recorded stereo-data from a new viewpoint: pseudo-clean features generated by exploiting HMM-based synthesis method [11, 12] is used to replace the ideal clean features from one of the stereo channels in SPLICE. Experimental results of clean training condition on aurora2 show that this pseudo clean features are even more effective than the ideal clean features. Moreover, a simple ML-based bias adaptation algorithm to handle the mismatch between training and testing, which yields consistent improvements of different testing sets on aurora2, is proposed. As a extension, this method of generating the pseudo clean features can be used in any algorithms [4, 13] like SPLICE where the stereo-data is needed.

The rest of this paper is organized as follows. First a review of SPLICE is given in section 2. In section 3, we propose our modifications. Experimental results are discussed in section 4. Finally in section 5, we give our conclusions.

2. REVIEW OF SPLICE

The flowchart of HMM-based pseudo-clean speech synthesis for SPLICE is illustrated in Fig. 1. In this section, we will give a brief description of SPLICE module. SPLICE is a general framework used to model and remove the effect of any consistent degradation of speech cepstra. The probabilistic formulation is described below.

2.1. Two assumptions of speech modeling and degradation

The first assumption is that the noisy speech cepstral vector follows the distribution of mixture of Gaussians:

$$\begin{aligned} p(\boldsymbol{y}_t) &= \sum_m p(\boldsymbol{y}_t|m)p(m) \\ p(\boldsymbol{y}_t|m) &= \mathcal{N}(\boldsymbol{y}_t;\boldsymbol{\mu}_m,\boldsymbol{\Sigma}_m) \end{aligned} \quad (1)$$

The second assumption is that conditional distribution for clean vector $\boldsymbol{x}_t$ given the noisy speech vector $\boldsymbol{y}_t$ in each component m is Gaussian whose mean vector is a linear transformation of $\boldsymbol{y}_t$ with the bias vector $\boldsymbol{r}_m$ as follows:

$$p(\boldsymbol{x}_t|\boldsymbol{y}_t,m) = \mathcal{N}(\boldsymbol{x}_t;\boldsymbol{y}_t+\boldsymbol{r}_m,\boldsymbol{\Gamma}_m) \quad (2)$$

2.2. SPLICE training

First, the GMMs of noisy speech in each environment are trained using standard EM algorithm. Then, if stereo-data is available, the bias parameters $\boldsymbol{r}_m$ can be trained using maximum likelihood criterion:

$$\begin{aligned} \boldsymbol{r}_m &= \frac{\sum_t p(m|\boldsymbol{y}_t)(\boldsymbol{x}_t-\boldsymbol{y}_t)}{\sum_t p(m|\boldsymbol{y}_t)} \\ p(m|\boldsymbol{y}_t) &= \frac{p(\boldsymbol{y}_t|m)p(m)}{\sum_l p(l|\boldsymbol{y}_t)p(l)} \end{aligned} \quad (3)$$

where this training procedure requires a set of stereo-data (two channels). One channel contains the clean speech, and the other contains time-synchronized noisy speech. The requirement of stereo-data is a main disadvantage of SPLICE.

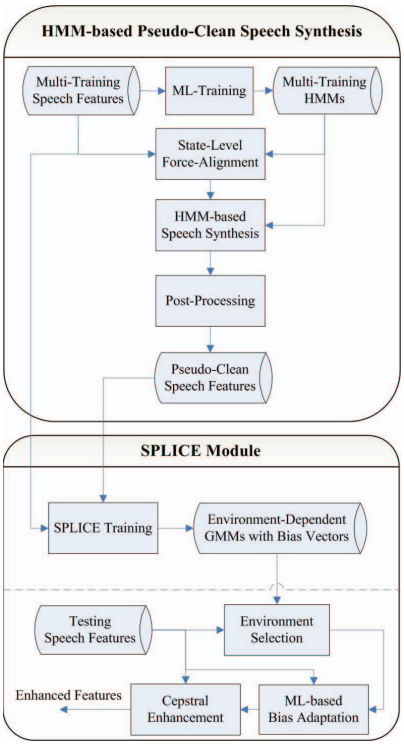

Fig. 1. System overview

2.3. Environment selection

In denoising stage, first we should select the corresponding environment for the current utterance in Bayesian framework:

$$E^* = \arg\max_E p(E|\boldsymbol{Y}) = \arg\max_E p(\boldsymbol{Y}|E)p(E) \tag{4}$$

where $\boldsymbol{Y}$ is the sequence of the noisy speech vectors in the current utterance, $p(E)$ is set equally for all E.

2.4. Cepstral Enhancement

After environment selection, MMSE estimation of clean speech, which is the conditional expectation of clean speech given the observed noisy speech, can be derived:

$$\hat{\boldsymbol{x}}_t = E_x[\boldsymbol{x}_t|\boldsymbol{y}_t] = \boldsymbol{y}_t + \sum_m p(m|\boldsymbol{y}_t)\boldsymbol{r}_m \tag{5}$$

3. OUR MODIFICATIONS

The main objective of this section is to introduce HMM-based pseudo-clean speech synthesis and ML-based bias adaptation as shown in Fig. 1.

3.1. State-level force-alignment

Imagine the scenario that we only have multi-training set, which consists of various speech from different noisy environments. It is hard to collect the ideal time-synchronized clean speech. So first, using multi-training speech features, we can get ML-trained HMMs. To some extent, multi-training HMMs is noise-robust. Then state-level force-alignment of multi-training features is performed. With this state sequence and multi-training HMMs, we can do the following HMM-based speech synthesis.

3.2. HMM-based speech synthesis

Our problem is corresponding to the **Case 1** discussed in [11]. Multi-training HMMs are denoted as λ, and the speech parameter vector sequence to be determined is described as:

$$\boldsymbol{O} = \left[\boldsymbol{o}_1^\top, \boldsymbol{o}_2^\top, ..., \boldsymbol{o}_T^\top\right]^\top \tag{6}$$

we assume that $\boldsymbol{o}_t$ consists of the static cepstral feature vector $\boldsymbol{c}_t$ and dynamic feature vectors $\Delta\boldsymbol{c}_t$,$\Delta^2\boldsymbol{c}_t$, that is, $\boldsymbol{o} = \left[\boldsymbol{c}_t^\top, \Delta\boldsymbol{c}_t^\top, \Delta^2\boldsymbol{c}_t^\top\right]^\top$, where dynamic features and static features should satisfy some constraints:

$$\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C} \tag{7}$$

$\boldsymbol{C}$ is the sequence of static cepstral vectors, the transformation matrix $\boldsymbol{W}$ is decided by the relation between static and dynamic features.

On the other hand, from the state-level force-alignment, the state sequence for all frames can be given:

$$\boldsymbol{S} = \{s_1, s_2, ..., s_T\} \tag{8}$$

and we calculate the mean and covariance of state s_t as follows:

$$\begin{aligned} \boldsymbol{\mu}_{s_t} &= \sum_m p(m|\boldsymbol{o}_t)\boldsymbol{\mu}_{s_t m} \\ \boldsymbol{\Sigma}_{s_t} &= \sum_m p(m|\boldsymbol{o}_t)(\boldsymbol{\Sigma}_{s_t m} + \boldsymbol{\mu}_{s_t m}\boldsymbol{\mu}_{s_t m}^\top) - \boldsymbol{\mu}_{s_t}\boldsymbol{\mu}_{s_t}^\top \end{aligned} \tag{9}$$

which can be considered as the "average parameters" of all Gaussian components in state s_t.

For given λ and $\boldsymbol{S}$, our target is to maximize likelihood function $p(\boldsymbol{O}|\boldsymbol{S},\lambda)$ with respect to $\boldsymbol{O}$ under the condition Eq. 7. The objective function can be written as:

$$\log p(\boldsymbol{O}|\boldsymbol{S},\lambda) = -\frac{1}{2}\boldsymbol{O}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{O} + \boldsymbol{O}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{U} + K \tag{10}$$

where

$$\begin{aligned} \boldsymbol{\Sigma}^{-1} &= \text{diag}\left[\boldsymbol{\Sigma}_{s_1}^{-1}, \boldsymbol{\Sigma}_{s_2}^{-1}, ..., \boldsymbol{\Sigma}_{s_T}^{-1}\right] \\ \boldsymbol{U}^{-1} &= \left[\boldsymbol{\mu}_{s_1}^\top, \boldsymbol{\mu}_{s_2}^\top, ..., \boldsymbol{\mu}_{s_T}^\top\right]^\top \end{aligned} \tag{11}$$

the constant K is independent of $\boldsymbol{O}$, by optimizing Eq. 10, we obtain a set of equations:

$$\boldsymbol{W}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{W}\boldsymbol{C} = \boldsymbol{W}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{U}^\top \tag{12}$$

By utilizing the special structure of $\boldsymbol{W}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{W}$, Eq. 12 can be solved efficiently in a time-recursive manner by the QR decomposition.

3.3. Post-processing

If we directly use the above synthesized pseudo-clean features, the recognition performance of SPLICE is not promising. As the post-processing, gain normalization should be taken based on the observation that there are big differences among the dynamic range of clean speech features, noisy speech features and synthesized

speech features, respectively. So a simple cepstral gain normalization (CGN) [14] is adopted:

$$\boldsymbol{x}_t^{\text{syn}} = (\boldsymbol{o}_t - \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{o}_t)/(\max_{1\le t\le T}\boldsymbol{o}_t - \min_{1\le t\le T}\boldsymbol{o}_t) \quad (13)$$

where $\boldsymbol{x}_t^{\text{syn}}$ is the final synthesized pseudo-clean features applied to SPLICE algorithm.

3.4. ML-based bias adaptation

In order to handle the mismatch between training and testing, a simple ML-based bias adaptation algorithm is proposed. First, for a testing utterance after environment selection, the bias parameters are adapted as follows:

$$\hat{\boldsymbol{r}}_m = \boldsymbol{r}_m + \boldsymbol{b} \quad (14)$$

where $\hat{\boldsymbol{r}}_m$ is the adapted bias vector for the mixture m, $\boldsymbol{b}$ is the global bias shift to describe the mismatch between training and testing. Here we do not use mixture-dependent bias shift for each m because there is not enough data to estimate it accurately. $\boldsymbol{b}$ can be iteratively estimated using the maximum likelihood criterion:

$$b_i' = \frac{\sum_t\sum_m p(m|\boldsymbol{y}_t,\boldsymbol{b})(y_{t,i}-\mu_{m,i})/\sigma_{m,i}^2}{\sum_t\sum_m p(m|\boldsymbol{y}_t,\boldsymbol{b})/\sigma_{m,i}^2} \quad (15)$$

$$p(m|\boldsymbol{y}_t,\boldsymbol{b}) = \frac{p(m)N(\boldsymbol{y}_t;\boldsymbol{\mu}_m+\boldsymbol{b},\boldsymbol{\sigma}_m^2)}{\sum_l p(l)N(\boldsymbol{y}_t;\boldsymbol{\mu}_l+\boldsymbol{b},\boldsymbol{\sigma}_l^2)} \quad (16)$$

where the subscript i denotes the dimensional index, $\boldsymbol{y}_t$ is the feature vector of the current utterance at frame t, $\boldsymbol{\mu}_m$ and $\boldsymbol{\sigma}_m^2$ are the mean and variance vectors of mixture m, respectively. The initial value of $\boldsymbol{b}$ is set to a zero vector. With this adaptation procedure, the bias parameters obtained from the training sets, are globally shifted to the current testing environment.

4. EXPERIMENTS AND RESULTS

4.1. Experimental setup

Our experiments are performed on the aurora2 database. The aurora2 task consists of English digits in the presence of additive noise and linear convolutional distortion. These distortions have been synthetically introduced to clean TIDigits data. Two training conditions (clean-traing/multi-training) and three testing sets (Set A/B/C) are defined by aurora2.

The cespstral features used in this paper are produced by the reference WI007 front-end with some modifications. The WI007 baseline uses a LogE (log-energy) feature and computes cepstra based on the magnitude frequency spectrum. We replace these with cepstral coefficient C0, and the power spectral density. And CMN is applied before SPLICE algorithm. The bias parameters are trained using the pseudo clean and real multi-style data for each of 17 noise conditions as described in [2]. The noisy speech model consists of a mixutre of 256 Gaussians with diagonal covariance matrices. HMMs in the back-end are trained in the manner prescribed by the scripts included with the aurora task. The details of baseline front-end and back-end can be found in [15, 16].

4.2. Experimental results

Table 1& 2 summarize performance comparison among different methods. "P-SPLICE" represents "SPLICE" using synthesized pseudo-clean features, and "P-SPLICE-BA" means "P-SPLICE" with bias adaptation. From these results, several observations can be made. First, all the methods based on SPLICE algorithm outperform "Baseline" system. Second, in clean-training condition, from Table 1, "P-SPLICE" performs better than "SPLICE" on SetA whose noise scenarios are the same as those for bias training while the opposite observation is obtained on SetB which are mismatch noise conditions with training conditions. On SetC the performance of "P-SPLICE" is slightly better than that of "SPLICE". From the viewpoint of different SNRs, "P-SPLICE" is more effective than "SPLICE" under lower SNRs as shown in Table 2. Third, in multi-training conditon, the performance of "P-SPLICE" is a little worse than that of "SPLICE" for different test sets and SNRs. In a word, our "P-SPLICE" method without stereo-data constrain is comparable with SPLICE for both clean-training and multi-training conditions. Finally, "P-SPLICE-BA" using bias adaptation is consistently outperforms "P-SPLICE", although the gain is not significant, which is due to the simple global strategy. The detailed results are listed in Table 3.

Table 1. Performance (word accuracy in %) comparison of several methods, averaged over SNRs between 0dB and 20dB across all noise conditions on three different test sets of aurora2 database.

Clean Training - Results				
Methods	Set A	Set B	Set C	Overall
Baseline	63.66	57.83	72.30	63.06
SPLICE	87.65	87.12	86.70	87.25
P-SPLICE	88.43	86.32	86.89	87.28
P-SPLICE-BA	88.59	86.48	87.16	87.46
Multi Training - Results				
Methods	Set A	Set B	Set C	Overall
Baseline	87.74	87.63	85.44	87.24
SPLICE	91.95	89.59	90.89	90.79
P-SPLICE	91.49	88.87	90.39	90.22
P-SPLICE-BA	91.53	88.93	90.48	90.28

Table 2. Performance (word accuracy in %) comparison of several methods, averaged over three test sets of aurora2 database at each SNR.

Clean Training - Results					
Methods	0dB	5dB	10dB	15dB	20dB
Baseline	19.72	43.70	70.61	87.03	94.23
SPLICE	62.96	84.92	93.35	96.83	98.19
P-SPLICE	64.09	84.91	93.27	96.36	97.75
P-SPLICE-BA	64.24	85.31	93.54	96.47	97.75
Multi Training - Results					
Methods	0dB	5dB	10dB	15dB	20dB
Baseline	60.72	86.98	94.42	96.54	97.51
SPLICE	73.25	89.75	95.42	97.40	98.16
P-SPLICE	72.75	89.03	94.84	96.81	97.68
P-SPLICE-BA	72.68	89.13	95.04	96.85	97.71

5. CONCLUSIONS

The modified version of SPLICE is described in this paper. First, we remove the constraint of stereo-data by exploiting HMM-based synthesis method to generate the pseudo-clean speech parameters. Experimental results show that the pseudo-clean features are even more effective than the real clean features in the clean training condi-

Table 3. Detailed results of P-SPLICE-BA on aurora2 database.

Clean Training - Results											
	Set A				Set B				Set C		
	Subway	Babble	Car	Exhibition	Restaurant	Street	Airport	Station	Subway M	Street M	Avg.
Clean	98.71	98.37	98.27	98.61	98.71	98.37	98.27	98.61	98.46	98.28	98.47
20dB	97.76	97.88	97.49	97.99	98.25	97.28	97.79	98.21	97.82	97.07	97.75
15dB	96.62	96.77	96.48	96.79	96.9	95.71	96.9	96.27	96.53	95.71	96.47
10dB	93.98	94.59	94.18	93.55	92.94	92.38	94.48	94.11	93.49	91.69	93.54
5dB	87.72	85.25	88.52	86.21	83.3	82.19	84.94	84.23	88.73	82.04	85.31
0dB	69.73	58.34	72.47	69.55	55.6	60.4	64.24	63.53	68.5	60.04	64.24
-5dB	35.52	20.31	35.76	36.35	20.23	24.79	28.03	28.79	34.39	25.48	28.97
Avg.	89.16	86.57	89.83	88.82	85.40	85.59	87.67	87.27	89.01	85.31	87.46
Multi Training - Results											
	Set A				Set B				Set C		
	Subway	Babble	Car	Exhibition	Restaurant	Street	Airport	Station	Subway M	Street M	Avg.
Clean	97.97	97.67	97.41	97.5	97.97	97.67	97.41	97.5	97.88	97.76	97.67
20dB	97.88	97.55	97.58	98.15	97.94	97.13	97.61	97.99	98.13	97.13	97.71
15dB	97.21	97.25	97.02	97.19	97.27	96.34	96.78	96.73	96.68	95.98	96.85
10dB	95.24	95.92	95.65	96.08	94.01	93.77	95.47	95.12	95.18	93.95	95.04
5dB	91.46	89.6	92.57	90.34	86.09	86.97	87.98	86.98	91.86	87.48	89.13
0dB	78.57	67.38	81.6	76.43	62.85	70.13	71.43	69.98	78.72	69.71	72.68
-5dB	46.24	26.18	49.42	47.02	24.96	34.04	34.03	37.03	44.49	32.92	37.63
Avg.	92.07	89.54	92.88	91.64	87.63	88.87	89.85	89.36	92.11	88.85	90.28

tion on aurora2. Then a simple ML-based bias adaptation algorithm to handle the mismatch between training and testing, which yields consistent improvements of different testing sets on aurora2, is proposed. In our future work, we will further study the noise robustness of the HMM-based speech parameters generation, which can be combined with other robust techniques for noisy speech recognition.

6. REFERENCES

[1] Gong, Y., "Speech Recognition in Noisy Environments: A Survey", Speech Communication, vol. 16, no. 3, pp. 261–291, Apr. 1995.

[2] Droppo, J., Deng, L., and Acero A., "Evaluation of the SPLICE Algorithm on the Aurora2 Database", Proc. EUROSPEECH'01, pp. 217–220, 2001.

[3] Acero, A, "Acoustical and Environmental Robustness in Automatic Speech Recognition", Ph.D. thesis, Carnegie Mellon University, 1990.

[4] Moreno, P. J., "Speech Recognition in Noisy Environments", Ph.D. thesis, Carnegie Mellon University, 1996.

[5] Wu, J. and Huo, Q., "An Environment-Compensated Minimum Classification Error Training Approach Based on Stochastic Vector Mapping", IEEE Trans. on Audio, Speech and Language Processing, vol. 14, no. 6, pp. 2147–2155, Nov. 2006.

[6] Huo, Q. and Zhu, D.-L., "A Maximum Likelihood Training Approach to Irrelevant Variability Compensation Based on Piecewise Linear Transformations", Proc. of ICSLP'06, pp. 1129–1132, 2006.

[7] Droppo, J. and Acero A., "Maximum Mutual Information SPLICE Transform for Seen and Unseen Conditions", Proc. EUROSPEECH'05, pp. 989–992, 2005.

[8] Povey, D., Kingsbury, B., Mangu, L., Saon, G., Soltau, H., and Zweig, G., "fMPE: Discriminatively Trained Features for Speech Recognition", Proc. ICASSP'05, 2005, pp. I-961–I-964.

[9] Deng, L., Wu, J., Droppo J., and Acero, A., "Analysis and Comparison of Two Speech Feature Extraction/Compensation Algorithms", IEEE Signal Process. Lett., vol. 12, no. 6, pp. 477–480, Jun. 2005.

[10] Zhu, D.-L. and Huo, Q., "A Maximum Likelihood Approach to Unsupervised Online Adaptation of Stochastic Vector Mapping Function for Robust Speech Recognition", Proc. of ICASSP'07, 2007, pp. IV-773–IV-776.

[11] Tokuda, K., Yoshimura, T., Masuko, T., Kobayashi, T., and Kitamura, T., "Speech Parameter Generation Algorithms for HMM-based Speech Synthesis", Proc. of ICASSP'00, 2000, pp. 1315–1318.

[12] Yan, Z.-J., Soong, F. K., and Wang, R.-H., "Word Graph Based Feature Enhancement for Noisy Speech Recognition", Proc. of ICASSP'07, 2007, pp. IV-373–IV-376.

[13] Cerisara, C. and Daoudi, K., "Evaluation of the SPACE Denoising Algorithm on Aurora2", Proc. of ICASSP'06, 2006, pp. I-521–I-524.

[14] Yoshizawa, S., Hayasaka, N., Wada, N. and Miyanaga, Y., "Cepstral Gain Normalization for Noise Robust Speech Recognition", Proc. of ICASSP'04, 2004, pp. I-209–I-212.

[15] Hirsch, H. G. and Pearce, D., "The AURORA Experimental Framework for the Performance Evaluations of Speech Recognition Systems under Noisy Conditions", ISCA ITRW ASR2000, Paris, September 2000.

[16] Young, S., et al., "The HTK Book", Version 3.2, 2002.