applied sciences

MDPI

Article

Augmented Latent Features of Deep Neural Network-Based Automatic Speech Recognition for Motor-Driven Robots

Moa Lee and Joon-Hyuk Chang *

Department of Electronic Engineering, University of Hanyang, Seoul 04763, Korea; uiooiu11@hanyang.ac.kr
* Correspondence: jchang@hanyang.ac.kr

Received: 12 May 2020; Accepted: 30 June 2020 ; Published: 2 July 2020

Abstract: Speech recognition for intelligent robots seems to suffer from performance degradation due to ego-noise. The ego-noise is caused by the motors, fans, and mechanical parts inside the intelligent robots especially when the robot moves or shakes its body. To overcome the problems caused by the ego-noise, we propose a robust speech recognition algorithm that uses motor-state information of the robot as an auxiliary feature. For this, we use two deep neural networks (DNN) in this paper. Firstly, we design the latent features using a bottleneck layer, one of the internal layers having a smaller number of hidden units relative to the other layers, to represent whether the motor is operating or not. The latent features maximizing the representation of the motor-state information are generated by taking the motor data and acoustic features as the input of the first DNN. Secondly, once the motor-state dependent latent features are designed at the first DNN, the second DNN, accounting for acoustic modeling, receives the latent features as the input along with the acoustic features. We evaluated the proposed system on LibriSpeech database. The proposed network enables efficient compression of the acoustic and motor-state information, and the resulting word error rate (WER) are superior to that of a conventional speech recognition system.

Keywords: automatic speech recognition; human-robot interaction; deep learning; bottleneck layer; latent feature; bottleneck network

1. Introduction

Voice as a form of communication can be acquired virtually everywhere and is the most natural and intuitive means of communication between humans or humans and machines. Recently, the humanoid robots such as Softbank's Pepper [1], MIT's home robot JIBO [2], and Intel's Jimmy [3] have been developed thanks to the considerable advances in the automatic speech recognition (ASR) technology. Many recent studies on this ASR technology, as an indispensable part for the humanoid robots, have been actively carried out, but there still retains a challenging problem of isolating ambient noise. The humanoid robots, additionally, generate the strong ego-noise [4], which results in a significant factor deteriorating the recognition performance. Indeed, since the distance between the microphone and the internal parts of the robot that generate the ego-noise is much closer than the distance between the microphone and the speaker, the ego-noise is loudly recorded into the microphone causing performance degradation. The robustness to the ambient noise has been one of the main issues in ASR technology to this day, but the robustness to the ego-noise has not been fully addressed.

In general, DNN-based speech enhancement systems for dealing with the ambient noise are trained using two types of targets including masking-based targets and mapping-based targets. In the first case, the model attempts to describe the time–frequency (T-F) relationships of clean speech to background interference and make a binary classification decision on T-F units, such as ideal binary

mask estimation [5]. Furthermore, in [6], DNNs were used to estimate a smoothed ideal ratio mask (IRM) in the Mel frequency domain for robust ASR. In the second case, the DNN learns a complex mapping function from noisy to clean speech using multi-condition training data [7–9]. However, such conventional DNN-based speech enhancement systems generally require very deep models or large databases to cope with as much ambient noise as possible, which is not suitable for embedded systems such as the humanoid robots. In addition, these systems have limitations in using motor-state information as an auxiliary feature.

As for the ego noise suppression, spectral subtraction [10] can be one possible way. A DNN-based framewise prediction approach is developed to estimate the noise spectra using angular velocity when a robot manipulates a joint [11]. The estimated noise spectra are then subtracted from the target signal spectra to clean up the noisy speech signal. One of the problems in this approach is that the ASR performance gets quite poorer since the noise power estimates are different from the original ones especially when the ego-noise is non-stationary. Several researchers have tackled this problem by predicting and subtracting the ego-noise using templates. For instance, [12] proposed a way to predict the ego-noise using motion-like gesture and walking pattern template obtained from a pre-recorded motor noise that corresponds to the motion pattern. Along with the labeled motion command, the appropriate ego-noise template matched to the latest motion is selected from the template database and used for subtraction. A small set of noise template database are extended in [4] to a larger ego-noise space in which the template database is enhanced by incorporating more information related to the joints such as angular positions, velocities and accelerations. Next, [13] employed the motor data to use the intrinsic harmonic structure of the ego-noise and incorporated the ego-noise harmonics into a multichannel dictionary-based approach for the ego-noise reduction. These studies on the ego-noise reduction have shown that the instantaneous motor data of the humanoid robots can be used as a secondary information source for dealing with the ego-noise. Recently, we originally devised an idea in [14] to use the motor on/off state as the auxiliary information when designing the acoustic model of the DNN-based speech recognition system for the humanoid robots. However, the auxiliary information for the motor-state is simply designed as a one-hot vector, so the performance gain turns out to be limited.

In this paper, we propose to employ the latent features that improve the speech recognition performance by utilizing the internal knowledge about whether the motor is running or not. The first DNN is carefully designed to create the motor-state dependent latent features, and the input is determined by concatenating the motor-state data in addition to the acoustic features contaminated by the ego-noise. After the preliminary training is accomplished at the first DNN to result in the latent features, these are fed into the second DNN designed for acoustic modeling under the ego-noise environments. Subsequently, to fully represent the complex relationship between the audio signal and senones, the second DNN is trained with the inputs containing both the latent features and acoustic features. In order to verify the performance of our approach against the existing methods, experiments are extensively conducted on the LibriSpeech corpus. The experimental results showed better performance in terms of the WER reduction than the baseline models including the mapping-based speech enhancement, the method using one-hot encoded input [14], and the method using only acoustic features.

The rest of the paper is organized as follows: In Section 2, we describe the related works. Details about the model architecture are explained in Section 3. The experimental results are shown in Section 4. We conclude with a discussion in Section 5.

2. Bottleneck Network

In the past several years, the latent features extracted from the bottleneck layer have been widely used in many tasks such as speech recognition [15–17], audio classification [18,19], speech synthesis [20] and speaker recognition [21]. The latent features are generated from a multi-layer perceptron (MLP) or DNN with a middle bottleneck layer having a small number of hidden units compared to the

other hidden layers. This special hidden layer creates a constriction in the network to compress the task-related information into a low dimensional representation. Therefore, the latent features can be considered as nonlinear transformation and dimensionality reduction of the input features.

One may wonder how to derive the latent features in both the unsupervised and supervised method. In the unsupervised manner, an autoencoder with several hidden layers is trained to reconstruct the inputs as shown in Figure 1, and does not need explicit lables to train on. In this figure, the autoencoder has three layers: input, output and hidden layer. The input vector $\mathbf{x}$ is encoded to the hidden vector $\mathbf{h}$ of the autoencoder by a nonlinear activation function σ, with the learned weight matrix $\mathbf{W}^{(1)}$ and the bias vector $\mathbf{b}^{(1)}$. Then, the input is decoded from the hidden vector to produce a reconstructed vector $\tilde{\mathbf{x}}$ using the learned weight matrix $\mathbf{W}^{(2)}$ and bias vector $\mathbf{b}^{(2)}$. The autoencoder parameter $\theta = (\mathbf{W}^{(1)}, \mathbf{b}^{(1)}), (\mathbf{W}^{(2)}, \mathbf{b}^{(2)})$ is learned using the back-propagation algorithm by minimizing the mean square error (MSE) loss.

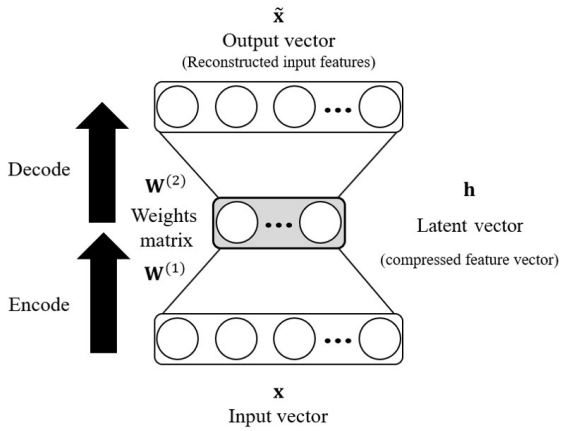

Figure 1. Structure of the autoencoder model.

Further, a stacked autoencoder can be used to extract latent features, which are progressively encoded using successive hidden layers. Then, fine-tuning on the entire stack of hidden layers is performed using the back-propagation algorithm. This allows each hidden layer to exhibit different levels of representation for the input feature.

In the supervised approach, latent features are created by the MLP trained to predict the class label (e.g., phoneme states) as shown in Figure 2, where the MLP is the feed-forward neural network made of an input layer, output layer, and at least one hidden layer. Usually, for the classification task, the softmax function is employed to convert the values of arbitrary ranges into a probabilistic representation. The learning process is accomplished to minimize the prediction error with respect to the parameter $\theta = (\mathbf{W}^{(1)}, \mathbf{b}^{(1)}), (\mathbf{W}^{(2)}, \mathbf{b}^{(2)}), \cdots, (\mathbf{W}^{(L)}, \mathbf{b}^{(L)})$, and a cross entropy error function [22] is generally selected as the loss function in the MLP. While the supervised method carries out the subsequent classification task, the classification information is compressed into the excitation of the latent units as much as possible via the continuous training. Hence, the latent features contain the expressive power to show more invariant and discriminative capability.

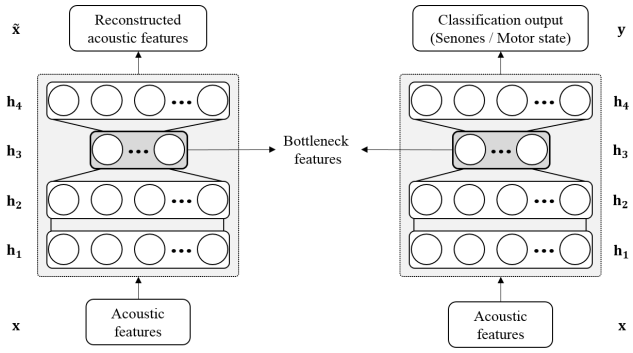

Figure 2. Extracting latent features using unsupervised (**left**) and supervised (**right**) methods.

3. Proposed Method

Since the instantaneous motor-state information of the robot is intuitively effective for dealing with the ego-noise, in this work, we attempt to fuse acoustic features obtained from spoken utterances and the motor-state data into a single framework. To this end, we propose a way to use the two distinct motor-states, one for the operating motor and the other for idle motor, which are brought into the auxiliary features as shown in Figure 3. The motor data are inherently acquired from the robot in such a way that the robot is driving like shaking its body or not in accordance with the motor-state alternating between "on" and "off". This auxiliary feature provides a very useful information, instantaneously changing, for the internal state of the intelligent robot. Then, the input $\mathbf{x}' = [\mathbf{x}; \mathbf{o}]$ is comprised of the conventional spectral features with the auxiliary feature and used for the latent feature training as depicted in Figure 4.

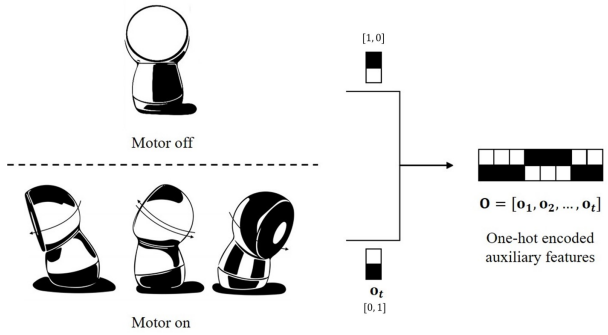

Figure 3. Extracting auxiliary features from the motor on/off state information for every frame ("Motor off" state with fan noise only and "Motor on" state with additional movement noise).

Here, the elementary question raises about how to fuse the motor data with the acoustic feature such as MFCC. For this, we introduce a way to exploit the latent feature as proposed in [12,16]. This key idea is accomplished by training the first DNN, designed for preliminary acoustic modeling, with the fused features as displayed in Figure 4. For training, three different targets are placed at the output, including senones, motor-state, and MFCC. The cross-entropy between the network output and target is minimized by using senones and motor-states as the labels, and the MSE loss is minimized to restore a clean version of MFCC. The created latent features can be thought of as an embedding features that contain information about the motor operation in combination with the acoustic features. Next, this latent features are fed into the second DNN, which is responsible for primary acoustic modeling as shown in Figure 4, illustrating the overall architecture of the proposed system. Eventually, latent features are appended to the acoustic features and then applied as the second DNN input together. There seems to be a possibility that the acoustic modeling capabilities can be improved due to the additional input that more effectively represents the information about whether the motor is

operating or not under the acoustic circumstances, which will be verified in detail in the following experimental section.

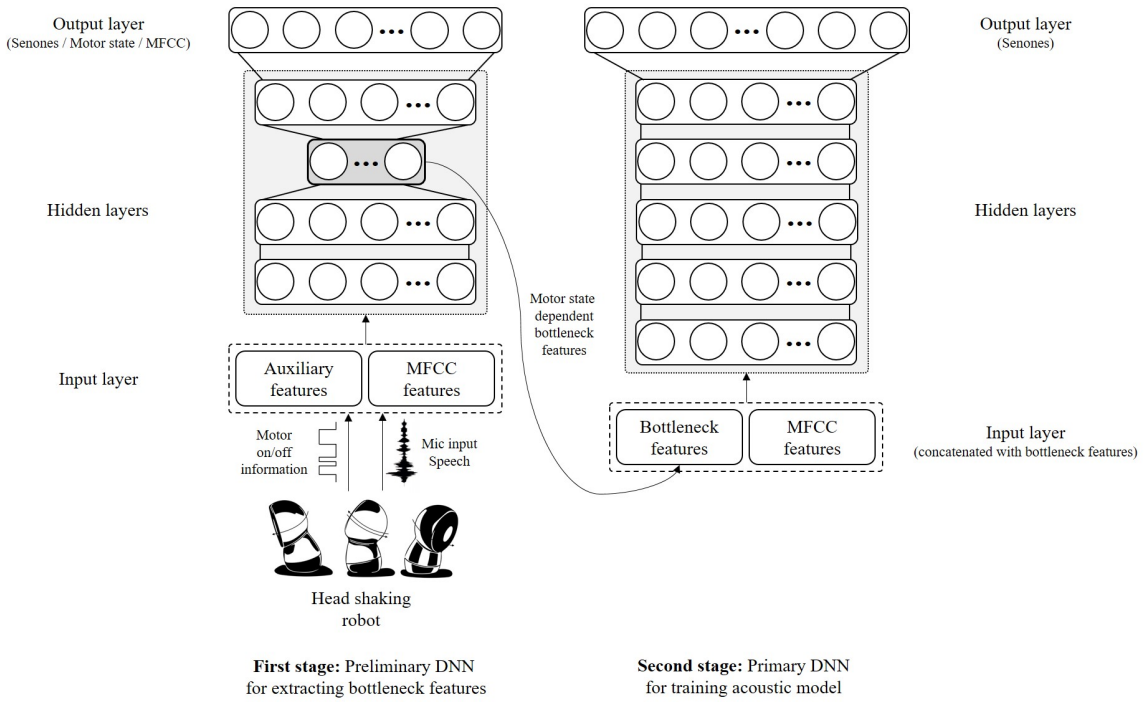

Figure 4. Framework of the proposed ASR system employing latent features.

4. Experiments and Results

4.1. Corpus Description

In order to evaluate the proposed approach, we conducted experiments with the JIBO humanoid robot which has 3 full-revolute axes [2] as a brief specification of the robot introduced in Table 1. We mainly consider the scenario in which humans interact with the robot while the robot shakes its head and body. To simulate noisy environments for the humanoid robot, we recorded the ego-noises using the single microphone located at the front side of the head under the anechoic environment. These noise signals involve two types of the ego-noise including fan and movement noises. The first type of the ego-noise labeled "Motor Off " only considers the fan noise, while the second type labeled as "Motor On" considers both the fan and movement noise. The mixing was conducted at various signal-to-noise ratio (SNR) levels including 5 dB and 15 dB with 1 m distance between the speaker and robot. Walking robots typically use a motor with a large torque, so low SNR levels between -5 dB and 0 dB are assumed [4]; a walking robot like ASIMO [23] requires motors with large torque because they run, climb stairs or avoid obstacles like human. However, since the JIBO robot is a social robot for home, its motion is limited and it uses a motor with lower torque. Furthermore, social robots generally operate in an indoor environment, so there is little external noise. Therefore, extreme conditions like an SNR level less than 0 dB were not considered. These mixtures were then used to train and evaluate the ego-noise robust ASR algorithms described above. Our experiments were performed on the LibriSpeech database [24] which is composed of three distinct training subsets of 100,360, and 500 h of speech respectively, representing a total training set of 960 h of read English speech. In our experiments, we used 100 h subset and further augmented the database via simulating the ego-noises that we recorded in reality. We evaluated the models on the test sets provided by the LibriSpeech corpus: test-clean and test-other. The difference between "clean" and "other" is the quality of the audio and its corresponding transcription. The quality of the "clean" is higher than that of the

"other" [24]. The waveform sampling rate of the corpus and the recorded noises was set to 16 kHz. We then evaluated the proposed algorithm in terms of WER under the aforementioned environments. Since we focus on the ego-noise in this work, the ambient noise was excluded, but, the acoustic noise can be successfully addressed in a unified DNN framework in the end.

Table 1. Hardware specifications of JIBO [2].

Hardware	Specifications
Sensors	360 degree sound localization
Movement	Three full-revolute axes
Sound	Two premium speakers
Processor	High-end ARM-based mobile

4.2. Experimental Setup

In our experiments, the Kaldi toolkit [25] was utilized to train both the preliminary DNN extracting the latent feature and the primary DNN for acoustic modeling. The two systems were equally designed with 4 hidden layers and each layer has 512 hidden units except for the bottleneck layer. For the primary DNN, the rectified linear unit (ReLU) activation functions was used in the lower layers and the softmax function was used at the output layer. For the acoustic features, 13 dimensional MFCC features were extracted using 25 ms analysis window with 10 ms frame shift. The baseline models used for comparison with the proposed model in this experiment are described in Table 2. Three baseline models were adopted to evaluate the proposed model. As for the first baseline system, the MFCC features stacked with 11 adjacent frames were used as the DNN input for acoustic modeling. In the second baseline system, the MFCC features and one-hot representation of the motor-state were used as the DNN input with 165 dim (15×11). Furthermore, the third baseline system is a DNN-based spectral mapping system designed to have a total of four hidden layers in order to have similar computational complexity to our proposed model. The spectral mapping model is trained using the 100 h subset contaminated by the ego-noises (5 dB and 15 dB SNR). The enhanced MFCC features are evaluated through the following acoustic model which is identical to the first baseline model. Therefore, the performance of the DNN-based spectral mapping model was verified through evaluation of the reconstructed MFCC features. We measured the performance of the first baseline model to check the performance of the acoustic model to be used in all experiments. In addition, to improve speech recognition performance in an environment where the ego-noise exists, the second and third baseline models use auxiliary features or a feature enhancement model, respectively. Therefore, evaluation in a clean environment without the ego-noise was performed only in the first baseline model. As for the proposed systems, the preliminary DNN was separately constructed with 4 hidden layers at the early stage. To investigate the effect of the various latent features, experiments were performed by varying output features, and bottleneck layer positions. Firstly, the senones, motor-state label and original MFCC features were independently placed in order at the output, then three kinds of the primary DNN yielding the latent features (bDNN) were designed including the bDNN_{senone}, bDNN_{MS}, bDNN_{MFCC}. The sigmoid and tanh activation functions were used for the classification and regression task, respectively. In addition, we varied the placement of the bottleneck layer from the bottom hidden layer (position 1) to the top hidden layer (position 4). For all the preliminary networks, the stacked MFCC and motor-state features ($(13+2) \times 11 = 165$-dim.) were used for the input. Then, the extracted motor-state dependent latent features were combined with the MFCC again and used to train the acoustic model as shown in Figure 4. The complexity of all models is described in Table A1.

4.3. Experimental Results and Analysis

Tables 3 and 4 present the WER results on LibriSpeech database for the "motor off" or "motor on" state, respectively. The values in the first column of these two tables represent the recognition

performance in absence of the ego-noise and indicate the performance of the acoustic model used in all models. This acoustic model was trained using a kaldi [25] script and its performance is similar to that of a p-norm DNN trained on a 100 h subset published by kaldi. (6% to 9% and 20% to 28% on the "clean" and "other" test set, respectively) Furthermore, the values in the last column show the average performance at SNR of 5 dB and 15 dB. On the both cases, it is discovered that the motor-state data yields better recognition performances as reported in [14]. As we expected, the auxiliary features like the latent feature, generated by using the bottleneck layer, yielded superior performance when compared to the DNN [14] using one-hot encoded vectors. It indicates that the bottleneck layer can create more valuable representation of the motor-state data by fusing along with the spectral features and being compressed. In addition, the proposed model was superior to the spectral feature mapping, a DNN-based speech enhancement method. Figure 5 shows the results of the DNN-based spectral feature mapping. According to the evaluation by the following acoustic model, the reconstructed features did not bring a significant performance improvement. Because, the DNN for the spectral feature mapping was designed lighter than the usual DNN-based mapping model to have a similar amount of computation as the proposed preliminary network. For the comparison of the proposed algorithms, the WER was reported by changing the output features among senones, MS, and MFCC, When constructing the preliminary DNN. The results show that the senones are appropriate as the target features for the preliminary DNN. Hence, predicting the senones turns out most relevant for the task of the primary DNN. To be specific, the bDNN_{senone} model achieved relative WER reduction of 2.24% and 5.83% over the baseline_1 model for the cases including "motor off" and "motor on" states. Furthermore, As for the position of latent features, the third layer yielded the superior performance for the bDNN_{senone} showing the best results as displayed in Figure 6.

Table 2. Model specifications (the MS and BN indicate the motor-state and latent feature, respectively).

Model		Preliminary DNN		Primary DNN	
		Input Feature	Output Feature	Input Feature	Output Feature
Baseline	baseline_1	-		Noisy MFCC	senone
	baseline_2	-		Noisy MFCC and MS	
	baseline_3	Noisy MFCC	Clean MFCC	Reconstructed MFCC	
Proposed	bDNN_{senone}	Noisy MFCC and MS	senone	Noisy MFCC and BN	
	bDNN_{MS}		MS		
	bDNN_{MFCC}		clean MFCC		

Table 3. Performance (WER in %) comparison using LibriSpeech database with 4-gram language model when the motor is off.

	WER (%)							
	Clean	Other	5 dB		15 dB		Average	
			Clean	Other	Clean	Other	Clean	Other
Baseline								
baseline_1	6.65	21.17	7.78	24.77	6.70	21.30	7.24	23.03
baseline_2 [14]	-	-	7.79	24.44	6.67	21.21	7.23	22.83
baseline_3	-	-	7.74	24.74	6.64	21.19	7.19	22.97
Proposed								
bDNN_{senone}	-	-	7.61	24.38	6.66	20.52	**7.14**	**22.45**
bDNN_{MS}	-	-	7.71	24.70	6.84	21.10	7.28	22.90
bDNN_{MFCC}	-	-	7.72	24.74	6.67	21.89	7.20	23.32

Table 4. Performance (WER in %) comparison using LibriSpeech database with 4-gram language model when the motor is on.

	WER (%)							
	Clean	Other	5 dB		15 dB		Average	
			Clean	Other	Clean	Other	Clean	Other
Baseline								
$baseline_1$	6.65	21.17	13.10	37.69	7.43	23.81	10.27	30.75
$baseline_2$ [14]	-	-	12.43	36.58	7.41	23.65	9.92	30.12
$baseline_3$	-	-	12.36	36.24	7.39	23.65	9.88	29.95
Proposed								
$bDNN_{senone}$	-	-	11.32	34.98	7.36	23.60	**9.34**	**29.29**
$bDNN_{MS}$	-	-	12.21	35.76	7.40	23.65	9.81	29.71
$bDNN_{MFCC}$	-	-	12.89	36.10	7.40	23.71	10.15	29.91

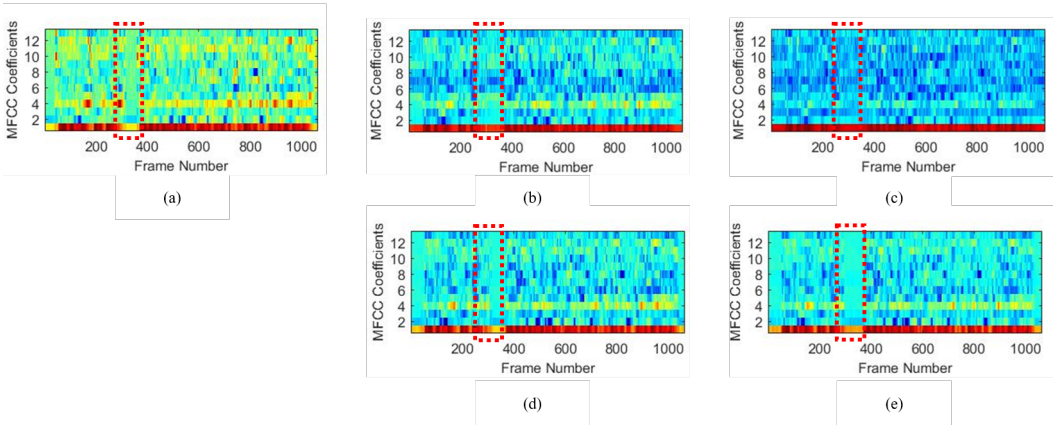

Figure 5. An example of the spectral feature mapping (**a**) clean MFCC feature; (**b**) noisy MFCC feature (SNR 5 dB, "motor off" state) (**c**) noisy MFCC feature (SNR 5 dB, "motor on" state); (**d**) reconstructed MFCC feature (SNR 5 dB, "motor off" state), (**e**) reconstructed MFCC feature (SNR 5 dB, "motor on" state).

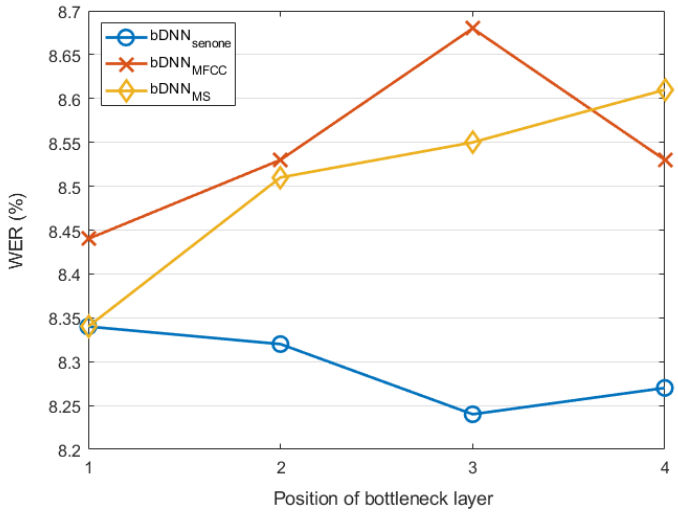

Figure 6. WER as a function of the position of the bottleneck layer on the "clean" test set (The result is the average for two types of the ego-noise).

5. Conclusions

In this paper, we proposed a novel way to incorporate the instantaneous motor on/off state information for the ego-noise robust ASR system. For this, we introduced the bottleneck network to create motor-state dependent latent features to effectively represent the ego-noise along with the acoustic features. These ego-noise adaptive latent features offer a significant improvement than the one-hot encoded motor-state features. In case of the proper target of the bottleneck network, the senones turn out most effective to extract the ego-noise adaptive latent features. Additionally, we compared the effect of the bottleneck layer position and concluded that the third layer was the best. To summarize, we demonstrated that the bottleneck layer offers more valuable representation of the motor data than the previous method. In a future work, more diverse states of the robot could be considered as the motor data, e.g., walking state, right/left arm rotating state, head shaking state, or multiple states

Author Contributions: Conceptualization, J.-H.C.; methodology, M.L.; software, M.L.; validation, M.L., and J.-H.C.; formal analysis, M.L.; investigation, M.L.; writing—original draft preparation, M.L.; writing—review and editing, M.L., and J.-H.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: This work was supported by the research fund of Signal Intelligence Research Center supervised by the Defense Acquisition Program Administration and the Agency for Defense Development of Korea.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Model complexity.

Model	Total Number of Parameters	
	Preliminary DNN	Primary DNN
baseline_1	-	4.4 M
baseline_2	-	4.5 M
baseline_3	2.4 M	4.4 M
DNN_{senone}	2.4 M	4.5 M

References

1. Wang, B. Next Big Future. Available online: https://www.nextbigfuture.com/2016/04/ibm-putting-watson-into-softbank-pepper.html (accessed on 30 June 2020).
2. Rane, P.; Mhatre, V.; Kurup, L. Study of a home robot: Jibo. *IJERT* **2014**, *3*, 490–493.
3. 21st Century Robot. Available online: https://www.21stcenturyrobot.com (accessed on 30 June 2020).
4. Ince, G.; Nakadai, K.; Rodemann, T.; Hasegawa, Y.; Tsujino, H.; Imura, J.I. Ego noise suppression of a robot using template subtraction. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009; pp. 199–204.
5. Wang, Y.; Wang, D. Towards scaling up classification-based speech separation. *IEEE ACM Trans. Audio Speech Lang. Process.* **2013**, *21*, 1381–1390. [CrossRef]
6. Narayanan, A.; Wang, D. Ideal ratio mask estimation using deep neural networks for robust speech recognition. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013.
7. Xu, Y.; Du, J.; Dai, L.R.; Lee, C.H. An experimental study on speech enhancement based on deep neural networks. *IEEE Signal Process. Lett.* **2013**, *21*, 65–68. [CrossRef]
8. Yoshioka, T.; Gales, M.J. Environmentally robust ASR front-end for deep neural network acoustic models. *Comput. Speech Lang.* **2015**, *31*, 65–86. [CrossRef]

9. Han, K.; Wang, Y.; Wang, D.; Woods, W.S.; Merks, I.; Zhang, T. Learning spectral mapping for speech dereverberation and denoising. *IEEE ACM Trans. Audio Speech Lang. Process.* **2015**, *23*, 982–992. [CrossRef]
10. Boll, S. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Trans. Acoust. Speech Signal Process.* **1979**, *27*, 113–120. [CrossRef]
11. Ito, A.; Kanayama, T.; Suzuki, M.; Makino, S. Internal noise suppression for speech recognition by small robots. In Proceedings of the Interspeech'2005-Eurospeech, Lisbon, Portugal, 4–8 September 2005.
12. Nishimura, Y.; Nakano, M.; Nakadai, K.; Tsujino, H.; Ishizuka, M. Speech recognition for a robot under its motor noises by selective application of missing feature theory and MLLR. In Proceedings of the ITRW on Statistical and Perceptual Audio Processing, Pittsburgh, PA, USA, 16 September 2006.
13. Schmidt, A.; Löllmann, H.W.; Kellermann, W. A novel ego-noise suppression algorithm for acoustic signal enhancement in autonomous systems. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018.
14. Lee, M.; Chang, J.H. DNN-based Speech Recognition System dealing with Motor State as Auxiliary Information of DNN for Head Shaking Robot. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018.
15. Grézl, F.; Karafiát, M.; Kontár, S.; Cernocky, J. Probabilistic and bottle-neck features for LVCSR of meetings. In Proceedings of the 2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP '07, Honolulu, HI, USA, 15–20 April 2007.
16. Yu, D.; Seltzer, M.L. Improved bottleneck features using pretrained deep neural networks. In Proceedings of the 12th Annual Conference of the International Speech Communication Association, Florence, Italy, 27–31 August 2011.
17. Sainath, T.N.; Kingsbury, B.; Ramabhadran, B. Auto-encoder bottleneck features using deep belief networks. In Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan, 25–30 March 2012 .
18. Zhang, B.; Xie, L.; Yuan, Y.; Ming, H.; Huang, D.; Song, M. Deep neural network derived bottleneck features for accurate audio classification. In Proceedings of the 2016 IEEE International Conference on Multimedia Expo Workshops (ICMEW), Seattle, WA, USA, 11–15 July 2016.
19. Mun, S.; Shon, S.; Kim, W.; Ko, H. Deep Neural Network Bottleneck Features for Acoustic Event Recognition. In Proceedings of the INTERSPEECH 2016, San Francisco, CA, USA, 8–12 September 2016.
20. Wu, Z.; King, S. Improving trajectory modelling for dnn-based speech synthesis by using stacked bottleneck features and minimum generation error training. *IEEE ACM Trans. Audio Speech Lang. Process.* **2016**, *24*, 1255–1265. [CrossRef]
21. Yaman, S.; Pelecanos, J.; Sarikaya, R. Bottleneck features for speaker recognition. In Proceedings of the Odyssey 2012-The Speaker and Language Recognition Workshop, Singapore, 25–28 June 2012.
22. Nasr, G.E.; Badr, E.A.; Joun, C.C. Cross entropy error function in neural networks: Forecasting gasoline demand. In Proceedings of the FLAIRS Conference, Pensacola Beach, FL, USA, 14–16 May 2002.
23. Sakagami, Y.; Watanabe, R.; Aoyama, C.; Matsunaga, S.; Higaki, N.; Fujimura, K. The intelligent ASIMO: System overview and integration. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland, 30 September–4 October 2002.
24. Panayotov, V.; Chen, G.; Povey, D.; Khudanpur, S. Librispeech: An ASR corpus based on public domain audio books. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brisbane, QLD, Australia, 19–24 April 2015.
25. Povey, D.; Ghoshal, A.; Boulianne, G.; Burget, L.; Glembek, O.; Goel, N.; Hannemann, M.; Motlicek, P.; Qian, Y.; Schwarz, P.; et al. The Kaldi speech recognition toolkit. In Proceedings of the IEEE 2011 Workshop on Automatic Speech Recognition and Understanding, Waikoloa, HI, USA, 11–15 December 2011.