ELASTIC SPECTRAL DISTORTION FOR LOW RESOURCE SPEECH RECOGNITION WITH DEEP NEURAL NETWORKS

Naoyuki Kanda, Ryu Takeda and Yasunari Obuchi

Central Research Laboratory, Hitachi Ltd.
1-280, Higashi-koigakubo, Kokubunji-shi, Tokyo 185-8601, Japan
naoyuki.kanda.kn@hitachi.com, ryu.takeda.qh@hitachi.com, yasunari.obuchi.jx@hitachi.com

ABSTRACT

An acoustic model based on hidden Markov models with deep neural networks (DNN-HMM) has recently been proposed and achieved high recognition accuracy. In this paper, we investigated an elastic spectral distortion method to artificially augment training samples to help DNN-HMMs acquire enough robustness even when there are a limited number of training samples. We investigated three distortion methods—vocal tract length distortion, speech rate distortion, and frequency-axis random distortion—and evaluated those methods with Japanese lecture recordings. In a large vocabulary continuous speech recognition task with only 10 hours of training samples, a DNN-HMM trained with the elastic spectral distortion method achieved a 10.1% relative word error reduction compared with a normally trained DNN-HMM.

Index Terms— Deep neural network, speech recognition, elastic distortion

1. INTRODUCTION

Making an accurate speech recognizer with limited training resources is currently a key issue in the speech recognition field because most languages do not have rich annotated corpora and making such corpora is costly. Recently, deep neural networks (DNNs) have proven highly effective for several recognition tasks [1, 2, 3]. For example, Seide et al. showed that acoustic models based on a combination of DNN and hidden Markov models (DNN-HMMs [2]) achieved much higher accuracy than conventional Gaussian mixture model-based acoustic models (GMM-HMM). In their study [2], a DNN-HMM trained by 309 hours of speech achieved almost the same (and sometimes better) accuracy as a GMM-HMM trained by 2000 hours of speech. Our experiment (discussed in section 4) also showed that a DNN-HMM trained by only 10 hours of speech achieved almost the same accuracy as a GMM-HMM trained by 270 hours of speech. Therefore, acoustic modeling based on DNNs could be a driving force for speech recognition applications in the low resource languages.

On the other hand, one previous study [4] showed that DNNs had a severely degraded performance when the acoustic properties of test samples were very different from those of the training samples. Our assumption is that with only limited training samples, DNNs cannot obtain sufficient robustness against several types of distortions, which could be obtained if such distortions were observed in the training samples. Another study [5] showed that, with plenty of training samples, DNNs could obtain robustness against vocal tract length differences. Therefore, we can expect that artificially augmenting the training samples will help DNNs acquire enough robustness against several types of distortions.

In this paper, an elastic spectral distortion method that artificially augments training samples by varying the spectral properties of original training data is investigated. In the character recognition field, an artificial augmentation of training samples by distorting original samples, called the "elastic distortion" method, has already been proposed and widely used. For example, in one study [6], training samples were artificially augmented by rotating and expanding original samples, and the neural networks trained with those augmented samples achieved much higher recognition accuracy. However, the speech spectrum has a time axis and a frequency axis, each with clearly different properties, and some distortion methods, such as image rotation, cannot be applied to it.

We investigated three distinct spectral distortion methods: vocal tract length distortion, speech rate distortion, and frequency-axis random distortion. Several experiments were conducted to evaluate the effect of the spectral elastic distorion method with Japanese lecture recordings. Some studies have already used DNNs to tackle low resource speech recognition. Those works focused on how to transfer the acoustics of other resource-rich languages to the target language [7, 8] or to make the language-independent feature front-end [9]. Compared to those methods, our approach has advantages that it can be applied to all types of networks and there is no need to prepare another resouces. Very recently, Jaitly and Hinton proposed an almost same method as vocal tract length distortion, and showed its effectiveness in TIMIT task with DNN-based acoustic models [10][1]. Compared to the paper [10], our work includes two additional distortion methods and shows their effectiveness even for large vocabulary continuous speech recognition tasks.

In Section 2, we give an overview of DNN-based acoustic models. In Section 3, the three spectral distortion meth-

[1]We became aware of the paper [10] after the acceptance notification.

ods we investigated are described. Finally, in Section 4, the evaluation results with a Japanese lecture recognition task are presented.

2. ACOUSTIC MODELING BASED ON DEEP NEURAL NETWORKS

2.1. An overview of deep neural networks

A deep neural network (DNN) is a multi-layer perceptron model that has several (normally more than three) layers. In the past, it was believed that deep structures made training the network parameters so difficult that most of the parameters became easily trapped at a poor local optimum. However, Hinton et al. [11] proposed an efficient layer-wise initialization method of parameters and showed that deep structures could produce much better results than shallow ones. Recently, DNNs have achieved state-of-the-art performance in many recognition tasks such as image recognition [1] and speech recognition [2, 3].

In neural networks, the l-th layers' value $\mathbf{z}^l = (z_1^l, ..., z_{N_l}^l)^T$ (N_l indicates the number of nodes in the l-th layer) is propagated to the $(l+1)$-th layer as

$$a_j^{l+1} = \mathbf{w}_j^l \cdot \mathbf{z}^l + b_j^l \tag{1}$$

$$z_j^{l+1} = h(a_j^{l+1}), \tag{2}$$

where $\mathbf{w}_j^l = (w_{j,1}^l, ..., w_{j,N_l}^l)$ and b_j^l indicate the weight and bias parameters in the l-th layer, respectively. Input layer's value $\mathbf{z}^1$ is initialized by input feature vector $\mathbf{x} = (x_1, ..., x_D)^T$ as $\mathbf{z}^1 = \mathbf{x}$. Function h indicates the activation function of each node. In this paper, a sigmoid function is used as an activation function in the hidden layers.

$$h(a_j^{l+1}) = \frac{1}{1 + \exp(-a_j^{l+1})} \tag{3}$$

In the output layer, softmax is used as the activation function.

$$p(y_j = 1|\mathbf{x}) = h(a_j^{L+1}) = \frac{\exp(a_j^{L+1})}{\sum_{k=1}^{N^{L+1}} \exp(a_k^{L+1})} \tag{4}$$

Neural networks are trained by maximizing a posteriori probability over training samples.

$$\mathcal{L} = \sum_i \log p(y_{(k_i)} = 1|\mathbf{x}_i) \tag{5}$$

Here, k_i indicates a reference of the i-th training data. Parameters $(\mathbf{w}_j^l, b_j^l)$ can be optimized by using numerical optimization techniques like a stochastic gradient decent, as

$$(\mathbf{w}_j^l, b_j^l) \leftarrow (\mathbf{w}_j^l, b_j^l) + \eta \frac{\partial \mathcal{L}}{\partial (\mathbf{w}_j^l, b_j^l)}, \tag{6}$$

where η is an update parameter. Gradients can be efficiently calculated by the back-propagation method [12].

2.2. Acoustic models based on deep neural networks

Recently, acoustic models that utilize DNNs to represent output probabilities of the hidden Markov model (HMM), which are called DNN-HMMs, have been proposed and have shown high speech recognition accuracy [2]. In this paper, we use a DNN-HMM for the acoustic modeling.

Conventional HMMs use GMMs to represent an output probability $p(\mathbf{x}|s)$ of the feature vector $\mathbf{x}$ from state s. In the DNN-HMM model [2], GMMs are replaced with DNNs to represent the output probability. According to Bayes' theorem, an output probability $p(\mathbf{x}|s)$ can be calculated from a posteriori probability $p(s|\mathbf{x})$ produced by DNN, as

$$p(\mathbf{x}|s) = \frac{p(s|\mathbf{x})}{p(s)} \cdot p(\mathbf{x}), \tag{7}$$

where $p(s)$ indicates a generative probability of state s and can be calculated from training samples. In contrast, p(**x**) indicates a generative probability of input $\mathbf{x}$, which does not affect the probability difference between states and can be ignored in decoding times.

2.3. Dropout training method

The dropout training method [13] has been proposed to prevent over-fitting of networks. In this method, each output from hidden layers is forced to be zero with a probability of $\gamma\%$ in the training phase. When using the neural networks, each weight parameter is multiplied $(100 - \gamma)\%$ instead of dropout. This process regards one network as a combination of many weak classifiers. The dropout training method has been found to be very effective for improving accuracy in both image recognition [1] and speech recognition tasks [13].

3. ELASTIC SPECTRAL DISTORTION FOR ACOUSTIC MODELING

In this section, we describe an artificial augmentation method of training samples by distorting original speech samples. In the character recognition field, artificial distortion of training data has already been proposed. For example, one study [6] showed that artificially augmenting training samples by rotating and expanding original samples can greatly improve the recognition accuracy. However, because the speech spectrum has time and frequency axes, which have clearly distinct properties, some distortion methods, such as image rotation, cannot be applied to it. In this study, we investigated three distinct spectral distortion methods: vocal tract length distortion, speech rate distortion, and frequency-axis random distortion. An overview of each distortion method is shown in Fig. 1.

3.1. Vocal tract length distortion

Vocal tract length distortion methods artificially augment the training samples by varying the vocal tract length of original training samples. This is achieved by applying vocal tract

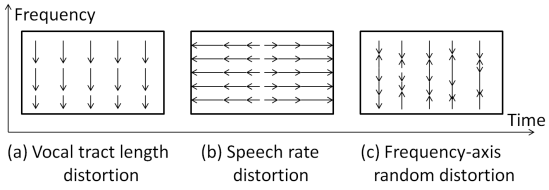


Fig. 1. *An overview of the three elastic spectral distortion methods. Rectangles indicate the spectrum and arrows within the rectangles indicate the direction of distortion.*

length normalization (VTLN) [14] to training samples with randomly selected warping factors every time we fetch training samples. Note that VTLN is normally applied to reduce inter-speaker variety of test or training samples, while in vocal tract length distortion, it is applied to augument variety of training samples. In this paper, we used the same simple linear frequency warping as that implemented in HTK [15].

As discussed in section 1, we assume that with only limited training samples, DNNs cannot obtain sufficient robustness against vocal tract length differences, which could be obtained if there were enough training samples. We expect that artificially augmenting the training samples by varying the vocal tract length will help DNNs acquire enough robustness even when there is a limited number of training samples.

3.2. Speech rate distortion

By the same discussion as described above, we assume that DNNs cannot obtain sufficient robustness against speech rate differences if there is only a limited number of training samples. To complement the training samples, we investigated a speech rate distortion method that artificially augments the training samples by varying the speech rate of the original training samples. Every time we fetch training samples, we change the speech rate of the samples with a randomly selected varying factor. We used Praat software [16] to change only the speech rate (and not the spectral properties).

Note that we need to re-make alignments of the input features and output states of DNNs whenever we change the speech rate. For simplicity, we randomly selected a varying factor from a fixed set of values, which enabled us to perform speech rate distortion with limited types of alignment information corresponding to a fixed number of varying factors.

3.3. Frequency-axis random distortion

We also investigated a frequency-axis random distortion method that simulates oscillations in the spectral domain. To realize a random distortion, we first generate a uniformly distributed random number between -1 and 1 for each time-frequency bin of the spectrum:

$$r(f,t) \sim U(-1,1), \tag{8}$$

where f and t indicate an index of frequency and time, respectively. Then, distortion factor $\delta(f,t)$ is calculated by averaging $r(f,t)$ in the small time-frequency region:

$$\delta(f,t) = \frac{\lambda}{(2p+1)(2q+1)} \sum_{f'=f-p}^{f+p} \sum_{t'=t-q}^{t+q} r(f',t'), \tag{9}$$

where $\lambda(>0)$ is the parameter that controls the magnitude of distortion. Parameters p and q are introduced for smoothing the randomness of the frequency axis and time axis, respectively. Note that while $r(f',t')$ is calculated according to Eq. 8 even if t' is negative or larger than maximum time index, $r(f',t')$ is set to 0 if f' is negative or larger than maximum frequency index, which makes $\delta(f,t)$ small when it is nearby the frequency boundary. Spectral power after distortion $\tilde{S}(f,t)$ is calculated from the original spectral power $S(f,t)$, as

$$\tilde{S}(f,t) = S(f+\delta(f,t),t) \tag{10}$$

Because $\delta(f,t)$ is not an integer, we calculate $\tilde{S}(f,t)$ as an interpolation of the adjacent frequency's power.

4. EVALUATION

4.1. Experimental settings

Large vocabulary continuous speech recognition (LVCSR) experiments were conducted using the Corpus of Spontaneous Japanese (CSJ), which is a collection of Japanese lecture recordings. As evaluation data, 2.4 hours of lectures featuring 10 speakers (5 male and 5 female) were used.

As training data for the acoustic models, 10 hours of lecture recordings (5 hours of male speech and 5 hours of female speech) were used to simulate a low resource scenario[2]. As a reference, we also evaluated acoustic models with 270 hours of lecture recordings[3]. As a language model, a 3-gram language model with 65,000 vocabulary words trained from a transcription of 2,671 lectures was used. A WFST-based one-pass decoder was used for LVCSR decoding.

4.2. Baseline 1: Evaluation of GMM-HMM models

We first evaluated conventional GMM-HMM acoustic models. We used 2,734 tied-state triphones as the units of acoustic modeling. 13 mel-frequency cepstral coefficients (MFCCs), delta coefficients, and delta-delta coefficients with mean and variance normalization per utterance were used as features. In 10-hour settings, each state was represented by eight mixtures of Gaussians and parameters were trained by a maximum likelihood (ML) criterion. In 270-hour settings, each

[2]Japanese is of course not a resource restricted language, but it is beneficial to simulate a low resource scenario because we can compare the results of the proposed method with normal DNNs with several sizes of training samples. This comparison is presented in Section 4.5.

[3]This content was all conference lectures from the CSJ, except the evaluation data.

Table 1. Word accuracy with 10 hours of training data.

Model	Feature (dim)	Word Acc. (%)
GMM-ML	MFCC (39)	71.7
DNN	MFCC (39)	80.1
	logMFB (75)	81.2

Table 2. Word accuracy with 270 hours of training data.

Model	Feature (dim)	Word Acc. (%)
GMM-ML	MFCC (39)	80.2
GMM-MPE	MFCC (39)	82.3
DNN	MFCC (39)	86.6
	logMFB (75)	87.3

Table 3. Effect of dropout (10-hour).

Model	Frame Acc.	Word Acc. (%)
DNN	46.6	81.2
DNN + Dropout	51.8	82.3

Table 4. Effect of vocal tract length distortion (10-hour).

Model	Distortion Ratio $[\alpha_{min}, \alpha_{max}]$	Frame Acc. (%)	Word Acc. (%)
DNN	-	46.6	81.2
DNN	[0.9, 1.1]	48.2	82.7
	[0.85, 1.15]	48.2	82.9
	[0.8, 1.2]	48.0	82.7

state was represented by 32 mixtures of Gaussians and parameters were trained by both ML and minimum phone error (MPE) criteria.

The GMM-HMM results are listed in the upper row of Table 1 (10-hour training) and Table 2 (270-hour training). Among the GMM-HMMs, the best result was obtained by the MPE-trained GMM (GMM-MPE) with 270-hour training data; it achieved a word accuracy of 82.3%.

4.3. Baseline 2: Evaluation of DNN-HMM models

Next, we evaluated DNN-HMM acoustic models. In these experiments, neural networks with 7-hidden layers, each of which has 2048 nodes, were used. Output units of the DNNs were 2,734 tied-state triphones, the same as those used in the GMM-HMM models. With 10 hours of training speech samples, the mini-batch size [17] was set to 128, and the initial update parameter η was set to 0.05 and 0.01 in the pre-training and fine-tuning, respectively. We used a discriminative pre-training [5] method for the pre-training and the AdaGrad method [18] for scheduling the update parameters[4]. With 270 hours of training speech samples, the mini-batch size was set to 1024 and the initial update parameters η were set to 0.05 in both pre-training and fine-tuning. Other settings were the same as those used with the 10 hours of training speech samples.

As input features of DNN, we used 39 MFCCs, the same as used in the GMM-HMM models. We also evaluated 75 log mel-filter bank features (logMFBs), which consisted of 25 log mel-filter bank coefficients (including one log energy feature) and delta and delta-delta coefficients. Both MFCCs and logMFBs were mean and variance normalized. We concatenated features of the previous 5 frames and following 5 frames for a total of 429 (= 39 x 11) MFCCs or 825 (= 75 x 11) logMFBs input into the DNNs.

Evaluation results for 10 hours of training data and 270 hours of training data are listed in the lower columns of Table 1 and Table 2, respectively. As shown in the tables, DNN-HMM showed great improvement of word accuracy compared with GMM-HMM: DNN-HMM achieved 81.2% and 87.3% word accuracy when logMFB features used, which correspond 33.6% and 28.2% relative error reductions from the GMM-HMM's best results. In our experiments, logMFB features always produced better results than MFCC features, the same as mentioned in the paper [20]. Moreover, the DNN-HMM trained using only 10 hours of speech achieved almost the same word accuracy as the GMM-HMM with 270 hour training data did. These results demonstrate the effectiveness of using DNNs in low resource scenarios.

4.4. Evaluation of dropout training

We evaluated the performance of the dropout method. In this experiment, dropout ratio γ was set to 50%. The update parameter η was set to 0.1 in both pre-training[5] and fine-tuning, and AdaGrad was used to schedule the update parameters. 75 logMFB features were used as input features. Other settings were the same as those described in the previous section. Because so much training time is needed for the dropout method, we only evaluate DNN-HMM with 10 hours of training data.

Evaluation results are shown in Table 3. In this experiment, we show not only word accuracy but also frame accuracy, which in this case means frame-by-frame triphone accuracy. Note that frame accuracy is directly calculated from the outputs of the DNNs and does not include LVCSR procedures. Since the frame accuracy was not affected by LVCSR, it can measure the pure identification performance of DNNs. We found that the dropout method improved the frame accuracy from 46.6% to 51.8%, which corresponds to a relative error reduction of 9.7%. Improvement of word accuracy was slightly smaller than the frame accuracy, and dropout achieved a 1.1-point improvement of word accuracy, which corresponds to a relative error reduction of 5.9%. We thought that the dropout method has a similar effect as the combination of weak classifiers, and therefore it could smooth the differences among output probabilities, which is key in LVCSR.

[4]Other studies [19] have pointed out that AdaGrad can possibly make converged parameters worse. However, in our preliminary experiments, the parameters obtained by AdaGrad were good enough. We used AdaGrad primarily for its fast convergence property.

[5]In the discriminative pre-training method, training samples are normally used only once when initializing each layer (early stopping [5]), but with dropout training, we iterated 20 times in each layer, which produced a slight improvement

Table 7. Effect of elastic spectral distortion (10-hour).

Model	VTL dist.	SR dist.	FR dist.	Dropout	Frame Acc. (%)	Word Acc. (%)
DNN					46.6	81.2
	√				48.2	82.9
		√			47.6	81.4
			√		47.0	81.6
	√	√	√		49.5	83.1
	√	√	√	√	53.4	83.5

VTL: Vocal Tract Length, SR: Speech Rate, FR: Frequency-axis Random

Table 5. Effect of speech rate distortion (10-hour).

Model	Distortion Ratio $[\beta_{min}, \beta_{max}]$	Frame Acc. (%)	Word Acc. (%)
DNN	-	46.6	81.2
DNN	[0.8, 1.2]	47.4	81.3
	[0.7, 1.3]	47.5	81.4
	[0.6, 1.4]	47.6	81.4

Table 6. Effect of frequency-axis random distortion (10-hour).

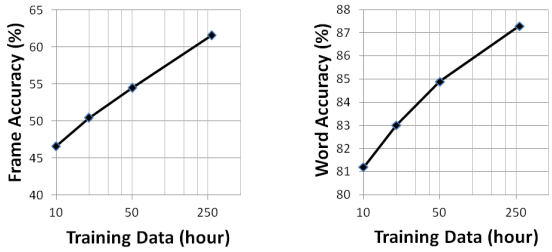
Model	Distortion Ratio λ	Frame Acc. (%)	Word Acc. (%)
DNN	-	46.6	81.2
DNN	100	46.7	81.3
	200	46.9	81.4
	400	47.0	81.6
	800	46.9	81.2

4.5. Evaluation of elastic spectral distortion

Next, we evaluated the elastic spectral distortion methods with 10 hours of training samples. In every beginning of the training epoch, original training samples were fetched and distorted with random factors at utterance level, and features extracted from distorted samples were used for DNN-training in that epoch. We used 75 logMFB as input features. Other settings were the same as in Section 4.3.

We first evaluated the vocal tract length distortion method. Every time we fetched training samples, we applied vocal tract length normalization with a random warping factor between $[\alpha_{min}, \alpha_{max}]$ in a 0.05 step size. We tested three patterns, $\{0.8, 1.2\}$, $\{0.85, 1.15\}$, and $\{0.9, 1.1\}$, as distortion ratio $\{\alpha_{min}, \alpha_{max}\}$. The results (shown in Table 4) demonstrate that the vocal tract length distortion works very well: it achieved a 1.6-point improvement of the frame accuracy, and a 1.7-point improvement of the word accuracy. It achieved a maximum word accuracy of 82.9%, which corresponds to a relative error reduction of 9.0%.

Next, we evaluated the speech rate distortion method. Every time we fetched training samples, we changed the speech rate with a random factor between $[\beta_{min}, \beta_{max}]$ in a 0.1 step size. We tested three patterns, $\{0.6, 1.4\}$, $\{0.7, 1.3\}$, and $\{0.8, 1.2\}$, as distortion ratio $\{\beta_{min}, \beta_{max}\}$. Results are shown in Table 5. Speech rate distortion achieved a 1-point improvement of the frame accuracy, but the improvement of the word accuracy was only 0.2 points. It achieved a maximum word accuracy of 81.4%, which corresponds to a relative error reduction of just 1.1%.



(a) Frame Accuracy (b) Word Accuracy

Fig. 2. Frame and word accuracy of normally trained DNNs.

Finally, we evaluated the frequency-axis random distortion method. We set $p = 128$ and $q = 100$ and varied λ from 100 to 800. Results are shown in Table 6. The frequency-axis random distortion method slightly improved both frame accuracy and word accuracy and achieved a maximum 0.4-point improvement on word accuracy.

Through the above experiments, we observed that frame accuracies and word accuracies were not necessarily correlated, especially in the comparison of different distortion methods. For example, the frame accuracy obtained by the speech distortion method at $\{\beta_{min}, \beta_{max}\} = \{0.6, 1.4\}$ was better than that obtained by the frequency-axis random distortion method at $\lambda = 400$; however, the word accuracy obtained by the former method was worse than that obtained by the latter. Note that word accuracies were obtained through large vocabulary continuous speech recognition tasks involving complicated decoding procedures with language model evaluations. We thought that this complexity hides the correlation between frame accuracies and word accuracies.

Table 7 shows the results of a combination of several distortion methods. In this experiment, we set $\{\alpha_{min}, \alpha_{max}\} = \{0.85, 1.15\}$, $\{\beta_{min}, \beta_{max}\} = \{0.85, 1.15\}$ and $\lambda = 400$. Combining all distortion methods resulted in 2.9-point improvements of frame accuracy and 1.9-point improvements of word accuracy, which corresponds to relative error reductions of 5.4 % and 10.1%, respectively. The effect of the distortion methods was nearly additive, especially in terms of frame accuracy; namely, the improvement of frame accuracy (2.9 points) by combining the three distortion methods was

almost the same as the sum of the improvements obtained by each method individually (1.6, 1.0, and 0.4 points, respectively). From above observation, we thought that there can be further improvements by using additional distortion methods.

Also shown in Fig. 2 are additional experimental results of normally trained DNNs with 20, 50, and 270 hours of speech[6]. The horizontal axis indicates the size of the training samples in the log scale. We found that both the frame accuracy and the word accuracy nearly followed the log of the training data sizes. The results demonstrate that the spectral distortion method achieved almost the same accuracy as a DNN with 20 hours of training data in terms of both the frame accuracy and the word accuracy. Therefore, it could be said that the effect of the spectral elastic distortion corresponded to double the amount of training samples.

Finally, we show the results of a combination of the elastic spectral distortion and dropout training in the last row of Table 7. This combination further improved the accuracy; it achieved an additional 3.9-point improvement of frame accuracy and 0.4-point improvement of word accuracy. Unexpectedly, the improvement of word accuracy was much smaller compared to the improvement of frame accuracy, and we thought this phenomena would be same as we discussed in Section 4.4.

5. CONCLUSION

In this paper, we investigated the elastic spectral distortion method to artificially increase the training samples under low resource scenarios. We investigated three types of distortion method: vocal tract length distortion, speech rate distortion, and frequency-axis random distortion. We evaluated the performance of these methods with a Japanese lecture recognition task and found that they all improved both the frame accuracy and the word accuracy. Combining distortion methods achieved a 10.1 % relative word error reduction compared with a normally trained DNN-HMM in a 10-hour training scenario.

6. REFERENCES

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, 2012, pp. 1106–1114.

[2] Frank Seide, Gang Li, and Dong Yu, "Conversational speech transcription using context-dependent deep neural networks," in *Proc. Interspeech*, 2011, pp. 437–440.

[3] Abdel-rahman Mohamed, George E Dahl, and Geoffrey Hinton, "Acoustic modeling using deep belief networks," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 14–22, 2012.

[4] Jinyu Li, Dong Yu, Jui-Ting Huang, and Yifan Gong, "Improving wideband speech recognition using mixed-bandwidth training data in cd-dnn-hmm," in *Proc. SLT*. IEEE, 2012, pp. 131–136.

[5] Frank Seide, Gang Li, Xie Chen, and Dong Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Proc. ASRU*. IEEE, 2011, pp. 24–29.

[6] Patrice Y Simard, Dave Steinkraus, and John C Platt, "Best practices for convolutional neural networks applied to visual document analysis," in *Proc. ICDAR*, 2003, vol. 2, pp. 958–962.

[7] Pawel Swietojanski, Arnab Ghoshal, and Steve Renals, "Unsupervised cross-lingual knowledge transfer in dnn-based lvcsr," in *Proc. SLT*. IEEE, 2012, pp. 246–251.

[8] Samuel Thomas, Michael L. Seltzer, Kenneth Church, and Hynek Hermansky, "Deep neural network features and semi-supervised training for low resource speech recognition," in *Proc. ICASSP*, 2013, pp. 6704–6708.

[9] Shigeki Matsuda, Xugang Lu, and Hideki Kashioka, "Automatic localization of a language-independent sub-network on deep neural networks trained by multi-lingual speech," in *Proc. ICASSP*. IEEE, 2013, pp. 7359–7362.

[10] Navdeep Jaitly and Geoffrey E Hinton, "Vocal tract length perturbation (VTLP) improves speech recognition," in *Proc. ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013.

[11] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[12] David E Rumelhart, Geoffrey E Hintont, and Ronald J Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[13] George E. Dahl, Tara N. Sainath, and Geoffrey E. Hinton, "Improving deep neural networks for lvcsr using rectified linear units and dropout," in *Proc. ICASSP*. IEEE, 2013, pp. 8609–8613.

[14] P Zhan et al., "Vocal tract length normalization for lvcsr," Tech. Rep., Technical Report CMU-LTI-97-150, Carnegie Mellon Univ, 1997.

[15] Steve Young, Gunnar Evermann, Dan Kershaw, et al., "The htk book (for htk version 3.2)," *Cambridge university engineering department*, 2002.

[16] Paul Boersma, "Praat, a system for doing phonetics by computer," *Glot international*, vol. 5, no. 9/10, pp. 341–345, 2002.

[17] Geoffrey Hinton, "A practical guide to training restricted boltzmann machines," *Momentum*, vol. 9, pp. 1, 2010.

[18] John Duchi, Elad Hazan, and Yoram Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, 2010.

[19] Andrew Senior, Georg Heigold, Marc'aurelio Ranzato, and Ke Yang, "An empirical study of learning rates in deep neural networks for speech recognition," in *Proc. ICASSP*. IEEE, 2013, pp. 6724–6728.

[20] Abdel-rahman Mohamed, Geoffrey Hinton, and Gerald Penn, "Understanding how deep belief networks perform acoustic modelling," in *Proc. ICASSP*. IEEE, 2012, pp. 4273–4276.

[6] DNNs with 20 hours of speech were trained with the parameters used for DNNs with 10 hours of training data, and DNNs with 50 hours of speech were trained with the parameters used for 270 hours of training samples.