Filter-based Discriminative Autoencoders for Children Speech Recognition

Chiang-Lin Tai[1], Hung-Shin Lee[1], Yu Tsao[2], and Hsin-Min Wang[1]
[1]*Institute of Information Science, Academia Sinica*
[2]*Research Center for Information Technology Innovation, Academia Sinica*
{taijohnny38, hungshinlee}@gmail.com

***Abstract*—Children speech recognition is indispensable but challenging due to the diversity of children's speech. In this paper, we propose a filter-based discriminative autoencoder for acoustic modeling. To filter out the influence of various speaker types and pitches, auxiliary information of the speaker and pitch features is input into the encoder together with the acoustic features to generate phonetic embeddings. In the training phase, the decoder uses the auxiliary information and the phonetic embedding extracted by the encoder to reconstruct the input acoustic features. The autoencoder is trained by simultaneously minimizing the ASR loss and feature reconstruction error. The framework can make the phonetic embedding purer, resulting in more accurate senone (triphone-state) scores. Evaluated on the test set of the CMU Kids corpus, our system achieves a 7.8% relative WER reduction compared to the baseline system. In the domain adaptation experiment, our system also outperforms the baseline system on the British-accent PF-STAR task.**

***Index Terms*—children speech recognition, autoencoders**

I. Introduction

Today, the technology of automatic speech recognition (ASR) is mature enough to be applied to the daily life of adults. However, children need it but cannot benefit as much as adults. According to the research in [1], the word error rate (WER) of children speech recognition can reach 5 times that of adults. The first reason is the lack of children corpora. It is usually much easier to collect transcribed adult speech from news broadcasts and regular recordings. But the above scenarios are rare for children. By 2016, there were only 13 children speech corpora that contained partial or complete word transcriptions [2]. To address the limitation caused by the lack of children resources, many efforts have been made to jointly use a large adult speech corpus and a relatively small children speech corpus for training acoustic models for children speech recognition [3], [4], [5], [6], [7]. Experiments have shown that the joint training approach (a.k.a. multi-condition training) can reduce WER compared to the training method that uses children speech alone.

In addition to changes in volume, prosody, and articulation that make adult and children voices different, research has shown that the pitch of children speech is not only higher but has greater changes. This phenomenon indicates that the acoustic features of high-pitched speech are not sufficient to train children ASR, although the acoustic features of low-pitched speech can be supplemented from adult corpora. Figure 1 shows the pitch distribution of 283 adult speakers of the WSJ corpus and 76 child speakers of the CMU Kids corpus. The pitch value of a person in Hz is derived by averaging the estimated pitch values of a series of voiced frames in all utterances he/she spoke. Obviously, the distribution of pitch in the group of children is wider, twice that of male or female adults. This phenomenon indicates that the features derived from the high-pitched speech are insufficient for training children ASR, but the features from low-pitched speech can be supplemented by adult speech.

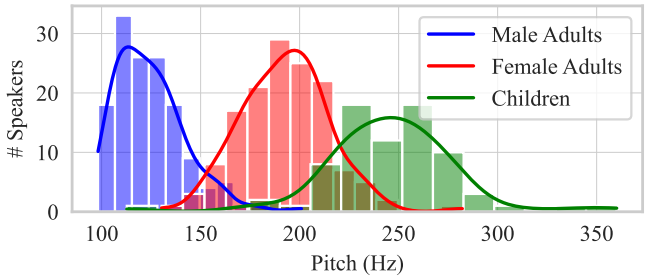


Fig. 1. Kernel density estimate (KDE) plots of the pitch distributions in the CMU Kids and WSJ corpora. The horizontal axis represents the average pitch computed in every 25 Hz interval for each speaker. The numbers of male adults, female adults, and children are 142, 141 and 76, respectively.

Moreover, the studies in [8], [3] show that children's Mel-frequency cepstrum coefficients (MFCCs) are not immune to the effect of pitch, especially in the case of higher pitches. In [3], the algorithm for feature extraction was reformed to filter out the high-pitched components according to the speaker's pitch level. In [4], pitch scaling was used to adjust the pitch of the child downward, so that the pitch variation of the adjusted child's speech could correspond to the pitch range of the adult. In [5], not only pitch adaptation was considered, but the children speaking rate was modified. In [9], vocal tract length normalization was performed to compensate the spectral variation caused by the difference in the vocal tract length among adults and children. The above methods focus on feature-level manipulation, with the purpose of reducing the inter-speaker acoustic variability.

As for the methods applied to the model itself, researchers have focused on adaptive [10], [7] or multi-task learning [7] of acoustic models. In [10], the transfer learning strategies for adapting the acoustic and pronunciation variability were discussed separately. In [7], two different softmax layers were optimized with adult speech and children speech respectively to differentiate the phonemes of children speech from those of

adult speech. Nonetheless, any additional adjustments to the model cannot avoid the risk of catastrophic forgetting or overfitting due to the scarcity of the children corpus. To reduce the inter-speaker distinction, another potential approach is to provide auxiliary information, such as speaker embeddings (e.g., i-vectors and x-vectors) [11], [12], [13] and prosodic features (e.g., pitch and loudness) [14], to the model.

In [15], [16], discriminative autoencoder-based (DcAE) acoustic modeling was proposed to separate the acoustic feature into the components of phoneme, speaker and environmental noise. Such model-space innovation takes a great advantage of unsupervised learning to extract pure phonetic components from the acoustic feature to better recognize speech. Inspired by this creativity, we combine the strength of the auxiliary information, i.e., the i-vector and pitch-related vector (called p-vector in this paper [17]) into autoencoder-based acoustic modeling to deal with very high-pitched speech (mainly children's speech). Because the use of i-vector and/or p-vector in DcAE-based acoustic modeling can be regarded as a filtering mechanism with inducers to purify the phonetic information in the acoustic feature, our model is called filter-based DcAE (f-DcAE for short).

II. Proposed Model

A. Filtering Mechanism for Acoustic Modeling

Our ASR model belongs to the Gaussian mixture models (GMM)/deep neural networks (DNN)/hidden Markov models (HMM) topology and is developed based on the `chain` structure of the Kaldi toolkit. The basic DNN training process takes the MFCCs and i-vectors as input with two loss functions: cross-entropy (CE) for frame-level training and lattice free maximum mutual information (LF-MMI) [18] for sentence-level training. The main function of using i-vectors for acoustic modeling is to eliminate factors such as speaker variation and channel mismatch in the acoustic features, thereby purifying the phonetic embedding. In this study, considering the extremely high-pitched characteristics of children, in addition to the i-vector, the p-vector is also used to reduce the impact of pitch on model adaptation.

To import the advantage of unsupervised learning in [15], [16] into our acoustic modeling, the DNN model for generating the emission probabilities of the output labels can be regarded as the encoder, and the decoder is used to reconstruct the acoustic features, as shown in Figure 2. The output of the penultimate layer of the encoder is supposed to be a pure phoneme-related vector without any information irrelevant to the phonetic content. Therefore, it can be regarded as a latent embedding vector representing the phonetic information (called s-code below). The s-code is concatenated with the i-vector, the p-vector, or the fusion of the two as the input of the decoder to reconstruct the original MFCCs.

Generally speaking, the encoder functions to identifying senones, i.e., triphone-states. (There are a total of 2,864 senones in this study.) In our proposed model, the encoder functions further like a senone filter that uses speaker and pitch information as guidance to remove impurities that are

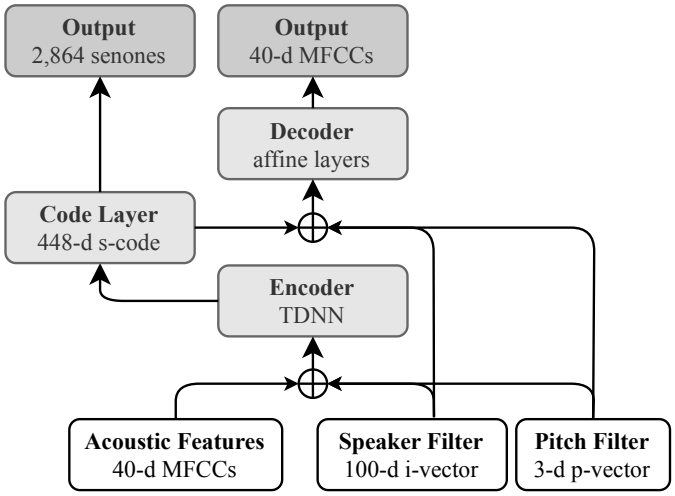


Fig. 2. The structure of f-DcAE, where $\bigoplus$ denotes the concatenation operator, and the two outputs (black boxes) are associated with three kinds of objective functions. There is no need to use the decoder in the recognition phase.

not related to senones from the input acoustic features. The participation of the decoder is to enhance the training of the senones filter. The significance of feature reconstruction in the decoder lies in the amplification on purifying the phonetic components in the s-code. The similarity between the input acoustic features and the reconstructed features relies on the high-quality s-code and the strong representations of speaker and pitch characteristics, which can be treated as three orthogonal factors of a speech utterance. By introducing a decoder in the training phase, the encoder is further indirectly guided to yield the s-code that better interprets the senone information, while the decoder attempts to reconstruct the original feature from the s-code together with the i-vector and/or the p-vector. The proposed model is called filter-based discriminative autoencoder (f-DcAE). Note that f-DcAE can be implemented on top of various DNN-based acoustic models, such as the long short-term memory (LSTM) [19], multi-head attention-based networks [20], transformers [21], and time-delay neural network (TDNN) [22] by considering them as the encoder. Although adding a decoder will increase the size of the entire model, only the encoder is used in the recognition phase. Therefore, the time complexity of f-DcAE in the recognition process is the same as that of the encoder-only counterpart.

B. Objective Functions

1) Senones-aware CE and LF-MMI: The first ASR-related objective function of the proposed f-DcAE model θ is the cross-entropy (CE) between the predicted senone (triphone-state) scores and ground truth defined as

$$F_{CE} = -\sum_{u=1}^{U}\sum_{t=1}^{T_u} \log p_\theta(\mathbf{s}_{ut}|\mathbf{x}_{ut}), \tag{1}$$

where U is the total number of training utterances, T_u is the total number of frames of utterance u, and $\mathbf{x}_{ut}$ and $\mathbf{s}_{ut}$ are the t-th input frame and the corresponding senone

label derived by GMM-based forced alignment, respectively. The second function relates to maximum mutual information (MMI), designed to maximize the probability of the reference transcription while minimizing the probability of all other transcriptions [23]. As in [24], the LF-MMI is defined by,

$$F_{LF-MMI} = \sum_{u=1}^{U} \log \frac{p_\theta(\mathbf{x}_u|M_{\mathbf{w}_u})P(M_{\mathbf{w}_u})}{p_\theta(\mathbf{x}_u)}, \quad (2)$$

where $\mathbf{w}_u$ is the reference word sequence of u, and the composite HMM graph $M_{\mathbf{w}_u}$ represents all the possible state sequences pertaining to $\mathbf{w}_u$, and is called the numerator graph. The denominator in Eq. (2) can be further expressed as

$$p_\theta(\mathbf{x}_u) = \sum_{\mathbf{w}} p_\theta(\mathbf{x}_u|M_{\mathbf{w}})P(M_{\mathbf{w}}) = p_\theta(\mathbf{x}_u|M_{den}), \quad (3)$$

where M_{den} is an HMM graph that includes all possible sequences of words, and is called the denominator graph. The denominator graph has traditionally been estimated using lattices. This is because the full denominator graph can become large and make the computation significantly slow. More recently, Povey *et al.* derived MMI training of HMM-DNN models using a full denominator graph [18].

2) Reconstruction errors and the final objective function: The most straightforward strategy for evaluating the similarity between reconstructed features and input features is through the mean squared error (MSE) defined as

$$F_{MSE} = \sum_{u=1}^{U}\sum_{t=1}^{T_u} ||\mathbf{x}'_{ut} - \mathbf{x}_{ut}||_2^2, \quad (4)$$

where $\mathbf{x}'_{ut}$ is the reconstructed feature vector of the input feature vector $\mathbf{x}_{ut}$, and $||\cdot||_2^2$ is the 2-norm operator.

The final loss function L to be minimized is the combination of the senone-aware CE, LF-MMI, and reconstruction error:

$$L = \alpha F_{CE} - F_{LF-MMI} + \beta F_{MSE}, \quad (5)$$

where α and β are the regularization penalty and weighting factor for F_{CE} and F_{MSE}, respectively. We set α to 5 followed by most recipes, and β was heuristically determined based on the development set in our experiments.

III. Experimental Settings

A. Datasets, Features, and Filters

Children speech in our experiments came from the CMU Kids corpus (LDC97S63[1]), which contains 9.1 hours of speech from 151 children. The age range of children is 6–11 years old. The speech content mainly consists of reading sentences from sources such as storybooks or textbooks. The training set (`tr_cmu`) contains 6.34 hours of speech from 76 kids, and the test set (`cmu`) contains 2.75 hours of speech from other 75 kids. The WSJ corpus was used as the adult speech corpus. The training set (`tr_wsj`) contains 81.48 hours of speech from 283 adults, the first test set (`eval92`) contains 0.7 hour of speech, and the second test set (`dev93`) contains 1.08 hours of speech. We combined `tr_cmu` and `tr_wsj` as a joint training set for multi-condition training in the experiments. In addition to the American-accent CMU Kids corpus, the PF-STAR British English children speech corpus [25] was also used in the experiments. The age range of children in PF-STAR is 4–14 years old. It contains 14.2 hours of speech from 152 children. The training set (`tr_pfstar`) contains 8.4 hours of speech from 92 kids, and the test set (`pfstar`) contains 5.82 hours of speech from other 60 kids.

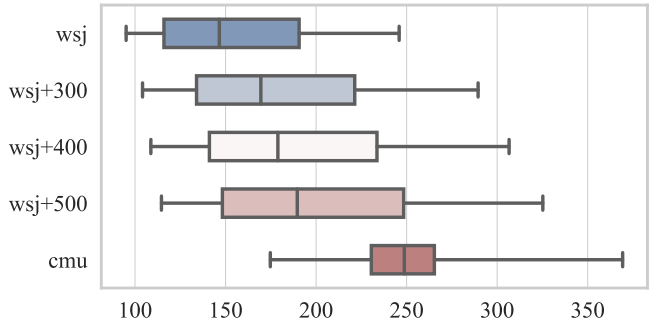


Fig. 3. Box plots of the pitch distributions in Hertz of the simulated children's speech from `eval92` and `dev93` (denoted by `wsj+xxx`) and the test set of the CMU Kids corpus (denoted by `cmu`).

To evaluate the ability of the acoustic model to filter out the effects of pitch, we intentionally created the high-pitched versions of `eval92` and `dev93` to simulate children speech. We used Sound eXchange (SoX[2]) to raise the pitch of adult speech by setting the parameter `pitch` to `+300`, `+400`, and `+500` without changing the speaking rate, where the units of these values are cents, not hertz. Figure 3 shows the pitch distribution of the original test sets and simulated versions of `eval92` and `dev93`. We found that once the pitch was set to `+600` or beyond, the adjusted speech sounded considerably unnatural, not similar to the speech of ordinary children and adults. In addition, the simulated speech was occasionally mixed with sharp mechanical noises, which could hardly represented children speech. Therefore, we used `+300`, `+400`, and `+500` to generate the simulated children test sets.

The size of training data was tripled using speed and volume perturbation. The 40-dimensional high-resolution raw MFCCs were used as the acoustic features.

The i-vector was 100-dimensional and was extracted every 10 frames [26]. We computed the average of pitch, delta-pitch and the normalized cross correlation function every 10 frames to derive a 3-dimensional p-vector [17].

B. Model Configuration

We implemented our baseline model based on TDNN [22], which has been used for children's ASR in [27], under the `chain` setting of the Kaldi toolkit. The TDNN model was composed of one affine layer, eight 448-dimensional TDNN layers, and one 448-dimensional dense layer before the softmax function. It was trained by minimizing the loss function consisting of CE and LF-MMI with the L2 regularization. We applied our training data (i.e., `tr_wsj` + `tr_cmu`) to

[1] https://catalog.ldc.upenn.edu/LDC97S63

[2] http://sox.sourceforge.net/sox.html

TABLE I
WERs (%) AND THEIR RELATIVE REDUCTIONS (%) WITH RESPECT TO VARIOUS TEST SETS.

Filters	None	i-vector		p-vector		i-vector ⊕ p-vector	
Test Set\Model	**Baseline(o)**	**Baseline(i)**	**f-DcAE(i)**	**Baseline(p)**	**f-DcAE(p)**	**Baseline(i+p)**	**f-DcAE(i+p)**
cmu	18.82	17.73	**16.82** (5.13)	18.92	**17.45** (7.77)	17.58	**16.44** (6.48)
eval92+300	2.96	2.68	**2.60** (3.85)	2.98	**2.55** (14.43)	2.89	2.89 (0.00)
eval92+400	3.81	3.51	**3.35** (4.56)	**3.79**	3.95 (-4.22)	3.83	**3.33** (13.05)
eval92+500	5.64	5.30	**5.00** (5.66)	5.37	**5.07** (5.59)	5.46	**5.23** (4.21)
dev93+300	5.88	5.40	**5.10** (5.56)	**5.67**	5.71 (-0.71)	5.42	**5.33** (1.67)
dev93+400	6.87	6.73	**6.47** (3.87)	6.74	**6.45** (4.30)	6.59	**6.27** (4.86)
dev93+500	9.27	9.62	**8.78** (8.73)	8.93	**8.77** (1.79)	9.56	**8.89** (7.00)

TABLE II
WERs (%) OF CMU AND PFSTAR WITH ONE-EPOCH ADAPTATION USING TR_CMU AND TR_PFSTAR, RESPECTIVELY.

Adapt./Test Sets	tr_cmu/cmu		tr_pfstar/pfstar	
One-epoch Adapt.	**no**	**yes**	**no**	**yes**
Baseline(i)	17.73	15.41	71.21	19.28
f-DcAE(i)	16.82	**14.60**	72.03	**18.77**

TABLE III
WERs (%) OF CMU AND PFSTAR WITH TRAINING FROM SCRATCH USING TR_CMU* (TR_WSJ+TR_CMU) AND TR_PFSTAR* (TR_WSJ+TR_PFSTAR), RESPECTIVELY.

Training/Test Sets	tr_cmu*/cmu	tr_pfstar*/pfstar
Baseline(i)	17.73	20.57
f-DcAE(i)	**16.82**	**19.39**

proceed with basic recipes, including feature extraction, GMM training/alignment, and DNN training.

To realize our f-DcAE model in Figure 2, on top of the baseline TDNN model, we added a decoder, which consists of four affine layers, each with 128 nodes. The first affine layer of the decoder was fed with the phonetic embedding (i.e., s-code, the last TDNN layer of the baseline model) extracted by the encoder (the first seven TDNN layers of the baseline model) as well as the i-vector and/or the p-vector.

The 4-gram language model (LM) and an enhanced lexicon, same as those in [22], were used. The perplexities for cmu, pfstar, eval92 and dev93 are 529.7, 693.2, 164.9, and 200.4, respectively. The ratios of the numbers of out-of-vocabulary words to total words for cmu, pfstar, eval92 and dev93 are 0/12180, 61/24838, 2/5700, and 17/8334, respectively. The perplexities of cmu and pfstar stand out dramatically because the LM is trained with news transcripts, and eval92 and dev93 are in the same domain. Hence, there is a domain mismatch between the LM and the two children test sets. We did not use the training transcriptions of the two children's corpora for LM training, because we found that their training and test transcriptions highly overlap each other.

IV. Experimental Results and Discussion

A. TDNN-based Baseline versus f-DcAE

Table I shows the WERs of the TDNN-based baseline and f-DcAE evaluated on different test sets. The model, consisting of a TDNN-based encoder with i-vector but no decoder, is regarded as a baseline that has been widely used in many studies. This baseline is denoted as Baseline(i) in Table I. Baseline(p) denotes the baseline model using p-vector instead of i-vector. Baseline(i+p) denotes the baseline model that uses both i-vector and p-vector. Baseline(o) denotes the baseline model without using any additional embedding. Corresponding to baseline models Baseline(i), Baseline(p), and Baseline(i+p), the proposed models with a decoder are denoted as f-DcAE(i), f-DcAE(p), and f-DcAE(i+p).

From Table I, we can see that f-DcAE(i), f-DcAE(p), and f-DcAE(i+p) are always better than Baseline(i), Baseline(p), and Baseline(i+p), respectively, on the cmu test set. The WER is relatively reduced by 5.1%, 7.8%, and 6.5%, respectively. This result shows that no matter which type of additional embedding is used, the proposed DcAE-based filtering mechanism can improve the recognition performance. As for the results of the simulated test sets, we can see that in most cases, f-DcAE(i), f-DcAE(p), and f-DcAE(i+p) are better than Baseline(i), Baseline(p), and Baseline(i+p), respectively. In summary, these results generally confirm the effectiveness of the proposed DcAE-based filtering mechanism for acoustic modeling, especially for the challenging children speech recognition task. Next, we focus on the results of the simulated (pitch-shifted) test sets. From Table I, we can see that the WER has a clear increasing trend with the rise of pitch shift (from +300 to +500). The proposed model with the filtering mechanism can achieve more WER reduction on the test set with more pitch shifts, e.g., f-DcAE(i+p) reduces WER by 7.00% on the dev93+500 test set compared to Baseline(i+p).

B. Adaptation with In-domain and Out-of-domain Data

In this experiment, we studied the effect of model adaptation. We compared Baseline(i) and f-DcAE(i). The models in Table I were used as the seed models and were adapted with one-epoch training by tr_cmu (in-domain data) and tr_pfstar (out-of-domain data). The results are shown in Table II. We can see that adapted with one-epoch training by tr_cmu, Baseline(i) can reduce the WER by 13.09% (from 17.73% to 15.41%), while f-DcAE(i) can reduce the WER by 13.20% (from 16.82% to 14.60%), when evaluated on cmu. Even though their relative WER reductions are comparable, the WER of f-DcAE(i) is lower than that of Baseline(i)

(14.60% vs 15.41%). When evaluated on `pfstar`, we can see that before adaptation, both initial models performed poorly, because `pfstar` is in British accent while the training data `tr_wsj+tr_cmu` is in American accent. With one-epoch adaptation by `tr_pfstar`, the WERs of both models were significantly reduced, and f-DcAE(i) achieved a lower WER than Baseline(i) (18.77% vs 19.28%).

C. Training from scratch for PF-STAR

In this experiment, we investigated whether our proposed model can work well under a specific multi-condition training condition, where the training data contain the speech of American adults and British children. The models were trained from scratch using `tr_wsj+tr_cmu` (both in American accent) or `tr_wsj+tr_pfstar` (the former in American accent and the latter in British accent) with sufficient training epochs. The results are shown in Table III. For the `cmu` task, f-DcAE(i) can reduce the WER by 5.13% compared to Baseline(i) (from 17.73% to 16.82%). For the `pfstar` task, f-DcAE(i) can reduce the WER by 5.74% compared to Baseline(i) (from 20.57% to 19.39%). The result confirmed that an adult training corpus in another accent can help improve the performance of children ASR. In addition, the result once again confirmed that the proposed f-DcAE model is better than the baseline model.

V. Conclusions and Future Work

In this paper, we have proposed a filter-based discriminative autoencoder (f-DcAE) architecture for acoustic modeling. Both the encoder and decoder are fed with the speaker and/or pitch filters. With the help of auxiliary information, the encoder purifies the phonetic information in the input acoustic feature, while the decoder reconstructs the input acoustic feature from the phonetic information extracted by the encoder. Experimental results have shown that our f-DcAE models outperform the counterpart baseline models without using the autoencoder architecture and the filtering mechanism.

In future work, a more non-experimental proof that the code layer (s-code) contains less speaker and pitch information in f-DcAE than the baseline model will be provided. For example, we will try to project each kind of s-codes into a plane using t-SNE [28] to visualize and compare their degrees of separation for speaker or pitch classes. Moreover, we will study another decent feature of f-DcAE, unsupervised pre-training without ASR losses, so that we can make the best use of a large amount of unlabeled children speech for further domain adaptation.

References

[1] A. Potamianos and S. Narayanan, "Robust recognition of children's speech," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 6, pp. 603–616, 2003.

[2] N. F. Chen, R. Tong, D. Wee, P. Lee, B. Ma, and H. Li, "SingaKids-Mandarin: speech corpus of Singaporean children speaking Mandarin Chinese," in *Proc. Interspeech*, 2016.

[3] S. Shahnawazuddin, A. Dey, and R. Sinha, "Pitch-adaptive front-end features for robust children's ASR," in *Proc. Interspeech*, 2016.

[4] W. Ahmad, S. Shahnawazuddin, H. K. Kathania, G. Pradhan, and A. B. Samaddar, "Improving children's speech recognition through explicit pitch scaling based on iterative spectrogram inversion," in *Proc. Interspeech*, 2017.

[5] S. Shahnawazuddin, N. Adiga, and H. K. Kathania, "Effect of prosody modification on children's ASR," *IEEE Signal Processing Letters*, vol. 24, no. 11, pp. 1749–1753, 2017.

[6] S. Shahnawazuddin, K. T. Deepak, G. Pradhan, and R. Sinha, "Enhancing noise and pitch robustness of children's ASR," in *Proc. ICASSP*, 2017.

[7] R. Tong, W. Lei, and M. Bin, "Transfer learning for children's speech recognition," in *Proc. IALP*, 2017.

[8] S. Ghai and R. Sinha, "Exploring the role of spectral smoothing in context of children's speech recognition," in *Proc. Interspeech*, 2009.

[9] R. Serizel and D. Giuliani, "Vocal tract length normalisation approaches to DNN-based children's and adults' speech recognition," in *Proc. SLT*, 2014.

[10] P. G. Shivakumar and P. Georgiou, "Transfer learning from adult to children for speech recognition: evaluation, analysis and recommendations," *Computer Speech & Language*, vol. 63, p. 101077, 2020.

[11] V. Peddinti, G. Chen, D. Povey, and S. Khudanpur, "Reverberation robust acoustic modeling using i-vectors with time delay neural networks," in *Proc. Interspeech*, 2015.

[12] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *Proc. Interspeech*, 2017.

[13] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. ICASSP*, 2018.

[14] H. K. Kathania, S. Shahnawazuddin, N. Adiga, and W. Ahmad, "Role of prosodic features on children's speech recognition," in *Proc. ICASSP*, 2018.

[15] M. H. Yang, H. S. Lee, Y. D. Lu, K. Y. Chen, Y. Tsao, B. Chen, and H. M. Wang, "Discriminative autoencoders for acoustic modeling," in *Proc. Interspeech*, 2017.

[16] P. T. Huang, H. S. Lee, S. S. Wang, K. Y. Chen, Y. Tsao, and H. M. Wang, "Exploring the encoder layers of discriminative autoencoders for LVCSR," in *Proc. Interspeech*, 2019.

[17] P. Ghahremani, B. Babaali, D. Povey, K. Riedhammer, J. Trmal, and S. Khudanpur, "A pitch extraction algorithm tuned for ASR," in *Proc. ICASSP*, 2014.

[18] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," in *Proc. Interspeech*, 2016.

[19] V. Peddinti, Y. Wang, D. Povey, and S. Khudanpur, "Low latency acoustic modeling using temporal convolution and LSTMs," *IEEE Signal Process. Lett.*, vol. 25, pp. 373–377, 2018.

[20] C. X. Qin, W. L. Zhang, and D. Qu, "A new joint CTC-attention-based speech recognition model with multi-level multi-head attention," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2019, no. 1, p. 18, 2019.

[21] Y. Wang, A. Mohamed, D. Le, C. Liu, A. Xiao, J. Mahadeokar, H. Huang, A. Tjandra, X. Zhang, F. Zhang, C. Fuegen, G. Zweig, and M. L. Seltzer, "Transformer-based acoustic modeling for hybrid speech recognition," in *Proc. ICASSP*, 2020.

[22] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proc. Interspeech*, 2015.

[23] K. Veselý, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in *Proc. Interspeech*, 2013.

[24] H. Hadian, H. Sameti, D. Povey, and S. Khudanpur, "Flat-start single-stage discriminatively trained HMM-based models for ASR," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 26, no. 11, pp. 1949–1961, 2018.

[25] A. Batliner, M. Blomberg, S. D'Arcy, D. Elenius, D. Giuliani, M. Gerosa, C. Hacker, M. Russell, S. Steidl, and M. Wong, "The PF_STAR children's speech corpus," in *Proc. Interspeech*, 2005.

[26] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 4, pp. 788–798, 2011.

[27] F. Wu, L. P. García-Perera, D. Povey, and S. Khudanpur, "Advances in automatic speech recognition for child speech using factored time delay neural network," in *Proc. Interspeech*, 2019.

[28] L. V. D. Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Leaerning Research*, vol. 9, pp. 2579–2605, 2008.