Deep Residual Networks with Auditory Inspired Features for Robust Speech Recognition

F. de-la-Calle-Silos, C. Peláez-Moreno, A. Gallardo-Antolín

Signal Theory and Communications Department,
Universidad Carlos III, Leganés (Madrid), Spain
{fsilos, carmen, gallardo}@tsc.uc3m.es

Abstract

The introduction of Deep Neural Networks (DNN) based acoustic models has become the new state of the art of speech recognition systems. The main reason for this is their lower recognition error rates in comparison with the traditional GMM-based systems. However, the problem of robustness in noisy environments still exits. Deep Residual networks (ResNet), a special type of DNNs, are popular in computer vision due to their increasing number of convolutional layers and ease of optimization, achieving a better performance in almost all the standard image recognition datasets. In this paper, a Deep Residual Network architecture is proposed, allowing ResNets to be used in speech recognition tasks where the network input is small in comparison with the image dimensions for which they were initially designed. Although the proposed model improves robustness against noisy conditions itself we also introduce a modification of the well-known Power Normalized Cepstral Coefficients (PNCC) as input to the ResNet with the aim of creating a noise invariant representation of the acoustic space. Experiments show that deep residual learning in conjunction with these features provides substantial improvements in recognition accuracy in both, mismatched and matched conditions, in comparison to a conventional baseline in the Aurora-4 dataset.
Index Terms: noise robustness, deep neural networks, residual networks, speech recognition, deep learning, features.

1. Introduction

In spite of the recent achievements of Automatic Speech Recognition (ASR) systems, their performance is still worse than that of humans in noisy or reverberant environments. The important leap in performance that ASR has experienced in last years is mostly due to the introduction of new acoustic models based on Deep Neural Networks (DNNs) ([1], [2], [3]). Nevertheless, they still suffer from important deterioration in noisy environments and when unseen data is present in the test set, i.e. mismatch conditions. A broad range of techniques have been proposed aimed at solving this problem, but the performance is still far from that of high Signal-to-Noise Ratio (SNR) scenarios [4].

The problem of mismatch training can be addressed following one of these two approaches: a) proposing novel neural network structures and b) robust feature extraction.

With respect to the first issue, a very effective example of novel neural network structures that generalize better on unseen data are Convolutional Neural Networks (CNN) [5]. CNN are simply neural networks where a convolution replaces the general matrix multiplication (feed-forward) in at least one of the layers. The convolution operation can be seen as a feature map that uses a filter. This way, this layer processes the image (spectrogram or any other time-frequency representation in the case of speech recognition) with filters whose parameters are learnt through back-propagation and convex optimization. The main advantages of CNN are: a) parameter sharing, allowing a small number of parameters since the filters are shared throughout the whole input image and b) preservation of the local correlations of the spectrogram.

CNN have become the state of the art in computer vision [6, 7, 8, 9] and also have been employed in speech recognition [10, 11, 12, 13] including contributions to robust speech recognition [14, 15]. It is worth noting that in [11] the application of a very deep convolutional neural network to noise speech recognition provides an important enhancement. In particular the authors present various network architectures based on the well known VGGNet [8], using small 3×3 convolutional filters and 2×2 pooling layers in a network with a high number of layers: in particular, they employ 10 convolutional layers and 4 fully connected layers, obtaining high recognition scores in multi-condition training (the mismatch training problem is not addressed).

Recently in the computer vision community, Residual Networks (ResNet) [9, 16] have been shown to improve the VGGNet baseline, by increasing the number of convolutional layers through the inclusion of shortcut connections. In this paper we put forward that ResNets can improve speech recognition rates in noisy conditions given that they are capable to more effectively model the speech variability of data.

Regarding the second issue, robust feature extraction has been traditionally employed to create invariant representations of the speech signal most of the times inspired in the Human Auditory System (HAS). For example, the classical Mel-Frequency Cepstral Coefficients (MFCC) [17] and Perceptual Linear Processing (PLP) features [18], as well as Gammatone-based Coefficients (GTC) [19] or the more recently proposed Power-Normalized Cepstral Coefficients [20, 21]. Additionally, well-known techniques for reducing the train-test mismatch can also be applied like mean normalization and mean variance normalization which improve the DNN performance. Of course, training with noisy data is an effective approach but in some situations where the test conditions are unknown it can be impractical.

Successful combinations of robust features and DNN backends to address the mismatch problem have been proposed in numerous works, as for example [22] where a review of different feature extraction strategies is presented showing that manually designed (as opposed to automatically learnt) feature extraction is still relevant or [23] where a specific feature set is tailored to a DNN architecture.

In this paper, we propose a modification of the Power-Normalized Cepstral Coefficients [21] that takes into account the masking properties [24] of the HAS and outputs a filter-

bank like representation, that allows us to increase the recognition rates when used in conjunction to CNN and other DNN architectures and in particular, ResNets.

The remainder of this paper is organized as follows: Section 2 introduces deep residual learning and our proposed architecture that adapts the original computer vision ResNet to speech recognition. Our modification of PNCCs is presented in Sections 3. Sections 4 and 5 contain, respectively, the experimental results achieved in comparison with other state of the art techniques and a discussion about them. Finally, we draw some conclusions and further lines of research in Section 6.

2. Deep Residual Learning

Deep Residual learning addresses the problem of degradation when the number of layers in a network is high. In a vanilla network (standard backpropagation trained) the stacked layers directly try to fit the underlying mapping. On the contrary, in ResNets the layers goal is to fit a *residual* function. The resulting residual mapping is more amenable for optimization since it is easier to push a residual to zero than to try to fit an underlying mapping.

Being $H(x)$ the mapping to be fit, where x denotes the input of the first layer of the residual block, ResNets try to fit the mapping: $F(x) = H(x) - x$, and therefore the original function becomes $H(x) = F(x) + x$. This can be implemented by the addition of shortcut connections among the layers, as can be observed in Figure 1. The shortcut connections perform an identity mapping and the inputs are added to the output of the stacked layers. All this architecture is differentiable and therefore can be trained with traditional backpropagation.

The original Deep ResNet of [9] aimed at solving computer vision problems consists of several residual units (Figure 1) stacked together, where each residual unit consists of two convolutional layers with 3×3 filter sizes, batch normalization [25] applied after each convolutions and ReLU [26] activation functions after the first convolution and after the shortcut connections addition operation. Combining residual connections and batch normalization simplifies the training process since even when the weight matrix has small parameters (a typical cause of vanishing gradients) the addition of the input (to compute the residual) produces a more stable gradient across the network.

Deep ResNets architecture is easy to implement: when a layer output map has the same dimensions than the input, a simple addition is performed, however if the layer output map is halved, the number of convolutional filters needs to be doubled. Thus, to perform feature map halving convolutional layers with a stride of 2 are applied instead of the more usual pooling layers. The original ResNet is built by stacking residual units, with a final global average pooling layer, a 1000 units fully-connected layer and a final softmax output.

Our proposed architecture (Figure 2) adapts the original ResNet to speech recognition by taking into account the lower dimensions of the input in comparison with those of images. The input dimension in our case is 17×64 since we depart from a filter bank with 64 filters and a temporal context window of 17 frames. Thus, the ResNet is built by stacking as many residual units as to reduce the temporal dimension to one and the stride is applied every other unit. As in the original ResNet every time the stride is performed, the layer feature maps are doubled. This gives us a total of 6 residual units with 512 feature maps in the final layer. A final average pooling is performed to obtain an output size of 512 to finish with a fully connected layer of 1000 ReLU units and softmax output.

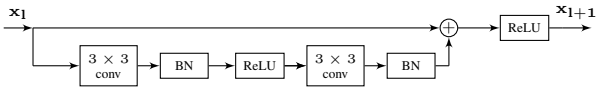

Figure 1: *A typical residual unit. Bach Normalization (BN) and ReLU activation function are applied after each convolution.*

3. Robust Features in Deep Learning-Based Speech Recognition

The Power-Normalized Cepstral Coefficients [21] (PNCC) are based on the use of a power-law non-linearity that replaces the traditional logarithmic non-linearity used in MFCC coefficients, a noise-suppression algorithm based on asymmetric filtering that removes background excitation, and a module that carries out temporal masking.

In this paper, we use a modification of the previously presented PNCC technique, where the modeling of the masking behavior of the HAS is used to enhance the robustness of the feature extraction stage [24]. This modeling consists of a nonlinear filtering of the PNCC spectrum, applied simultaneously on both the frequency and time domains, by processing it using mathematical morphology operations as if it were an image. The *structuring element* used in the morphology operation is designed to closely resembles the masking phenomena taking place in the cochlea.

In order to use the PNCC in the CNN networks, the last stage in which the Discrete Cosine Transform (DCT) is applied on the log filter-bank energies is removed obtaining a filter-bank like representation. This modification allows us to have a bigger input dimension (in particular 40 or 64 filters are used). Note that a high dimensional input representation is required in order to increase the number of convolutional layers.

Another modification needed is to remove the power-law non-linearity, as we have found that the traditional logarithmic operation performs better in conjunction with deep-learning back-ends. This follows the line of [14, 15] where it has been shown that the Mel filter bank log-energies (MelFB) perform better than the traditional MFCC.

The PNCC-based features with the logarithmic non-linearity and without DCT are denoted as Power Normalized Filter Banks (PNFB), whereas these latter features with the inclusion of our modeling of the auditory masking based on Morphology Filtering (MF) will be referred in the remaining part of the paper as MF-PNFB.

4. Experiments

In this section we report on the effectiveness of ResNet and MF-PNFB in robust ASR using the Aurora-4 corpus [27]. Aurora 4 [27] is a medium size vocabulary task based on the Wall Street Journal (WSJ0) corpus.

The experiments were performed using the 16 kHz clean and multi-condition training sets. Each training set consists of 7137 utterances from 83 speakers. The clean training set contains only clean data recorded with a single microphone. On the other hand, the multi-condition training uses different microphones and additive artificial noise. In particular, it is corrupted with six different noises (street traffic, train station, car, babble, restaurant, airport) at SNRs of 10-20dB.

The evaluation set is derived from the WSJ0 5K test set corrupted with the same noises and recorded with different types of microphones, creating a total of 14 test sets with 330 utter-

ances each. Note that the types of noise are shared across multi-condition training and test sets but the SNRs of the data are not. The results presented in this paper are averaged across all the 14 tests sets. The clean and multi-condition development sets were only used for validation of the neural networks training.

Traditional GMM-HMM systems are used as a baseline and to obtain the alignments for training the neural networks using the Kaldi Speech Recognition Toolkit [28]. All the proposed deep learning architectures are built following a hybrid architecture where the neural networks are trained to classify the input acoustic features into classes corresponding to the states of HMMs, so that the state emission likelihoods usually computed with GMM can be replaced by the likelihoods generated by the DNN.

In summary, five acoustic modeling systems are evaluated: a traditional triphone GMM-HMM, a fully connected DNN, a state of the art CNN, a very deep CNN version and the proposed ResNet. All the systems are trained in clean and multi-condition scenarios with two different feature extraction methods: the traditional mel Filter Banks (MelFB) and the proposed modification of the Power Normalized Filter Banks (MF-PNFB). In the first four cases, the static acoustic parameters are composed of 40 filters, whereas in the ResNet case, the number of filters is set to 64.

The triphone GMM-HMM baseline system is trained employing the MFCC or PNCC features (cepstral version), lineal discriminant analysis (LDA) and Maximum Likelihood Linear Transform (MLLT), using the Kaldi Aurora-4 recipe. The training recipe starts building a monophone system and then employs the alignments obtained in this first stage to train an initial triphone system. A final triphone system with LDA and MLLT is subsequently retrained using the alignments of the later triphone system. The clean training data is used to obtain the alignments in the mismatched case and the multi-condition data is used to obtain the alignments in the matched case. The DNN, CNN, ResNet models are trained using the alignments obtained by the final triphone system and mean variance normalization is applied to the input features for every system in a per utterance basis.

The deep neural fully connected network baseline (DNN) is composed of 5 hidden layers with 2028 units in each layer with ReLU activations functions and batch normalization in each layer. The input for this configuration consist of 40 MelFB or MF-PNFB and the corresponding Δ and $\Delta\Delta$ parameters. In this case, an 11 frame context window is used.

The convolutional neural networks architectures can be seen in Figure 2. The classical CNN proposed in [29, 13] is used as a CNN baseline consisting of two convolutional layers with 256 feature maps each one, with 9×9 and 3×3 filter sizes, with a pooling layer in-between, followed by 3 fully connected layers with 2048 hidden units each. ReLU activation functions and batch normalization are employed and he convolutions are performed with overlapping pooling.

The input for this configuration is encoded as a $11 \times 40 \times 3$ where 3 feature maps are used for the features corresponding to the static, Δ and $\Delta\Delta$ parameters including an 11 frame temporal context window.

The very deep CNN is based on the vd6 proposed by [11], where six 3×3 convolutional layers are staked together and non-overlapping pooling is used in convolutional layers. After the first two and four layers a 2 max-pooling operation is performed only in the frequency domain. In addition to the convolutional layers 3 fully connected ones are added. The vd6 only use a 11×40 feature map as the Δ and $\Delta\Delta$ are removed.

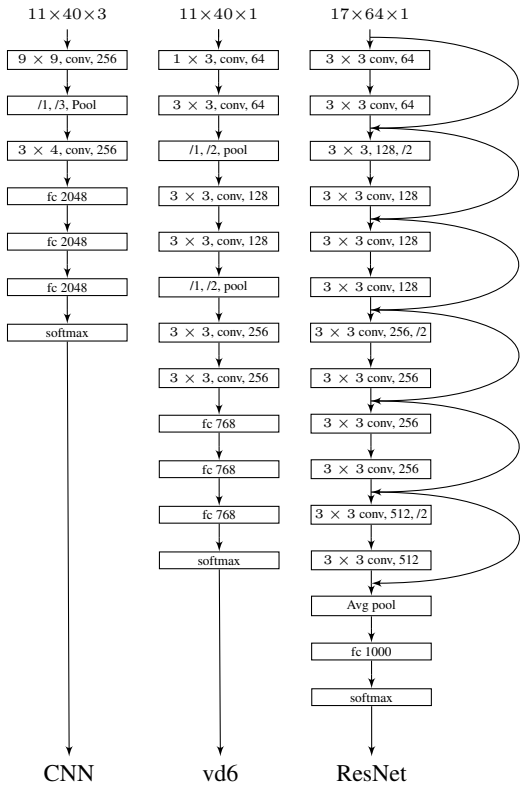

Figure 2: *Block diagrams of the three CNN architectures. The input, convolution and pooling sizes are given in time $\times$ frequency scale. In the ResNet architecture, the stride is denoted as /2 and is applied in both dimensions.*

The last configuration tested is the proposed ResNet architecture as described in Section 2. The input for this architecture is expanded to 17×64 to increase the number of residual layers, i.e. 64 MelFB or PNFB are used with a temporal context of 17 frames.

All the networks have a final softmax output layer whose output size is the number of senones of the final GMM-HMM system described above since we use a hybrid architecture.

The training pipeline used for the deep neural networks is almost the same for all the architectures: an Adam optimizer [30] with cross-entropy as a loss function with an initial learning rate of 0.001, Xavier initialization [31] for all the layers, early stopping with 3 retries of patience is used, where the learning rate is halved if the validation error is greater than the previous epoch. A maximum number of 20 epoch is allowed, no dropout is used, a batch size of 128 utterances is employed for all the networks except for ResNet where 64 is used due computational limitations.

The neural networks are trained using Tensorflow [32]. The connection between Kaldi and Tensorflow can be found on [33, 34]. Also the scripts used to train the networks can be found in [35].

Table 1 shows the recognition results for each of the ASR systems in terms of the Word Error Rate (WER) [%] averaged over all test sets both in clean and multi-condition training and comparing the two parameterizations considered, MelFB and MF-PNFB.

	Model	MelFB	MF-PNFB
Clean Training	GMM-HMM	46.71	30.30
	DNN	41.69	28.10
	CNN	38.20	27.91
	vd6	37.91	27.61
	ResNet	36.22	23.32
Multicondition Training	GMM-HMM	22.11	18.47
	DNN	14.01	14.38
	CNN	12.65	12.26
	vd6	11.70	11.61
	ResNet	10.33	10.11

Table 1: *Recognition results in terms of WER [%] using the Aurora 4 dataset, averaged over all test sets, for all the architectures, and for the two types of features (MelFB and MF-PNFB). Note that for the GMM baseline system, cepstral versions of the features are employed.*

5. Discussion

From the results shown in Table 1, three main conclusions can be drawn.

First, we have analysed the influence of Residual Networks (ResNet) on the ASR system performance. As can be observed, when using the conventional MelFB features, the ResNet architecture produces relative error reductions of 13.12% with respect to the plain neural network (DNN), 5.18% with respect to the convolutional network (CNN) and 4.46% with respect to vd6 in clean training conditions. For the multi-condition training scenario and the same features, ResNet also attains the best recognition rate, achieving relative error reductions of 26.27%, 18.34% and 11.71% with respect to, respectively, DNN, CNN and vd6 systems. In all cases, these performance differences are statistically significant. Similar observations can be made when the proposed features MF-PNFB are used, in both, clean and multi-condition training. These results suggest that the proposed ResNet model, which was initially designed for computer vision tasks, is also suitable for speech recognition due to its remarkable generalization capabilities. In fact, ResNet outperforms the other acoustic models considered (GMM-HMM, DNN, CNN and vd6) in mismatched and matched conditions, showing its robustness against noise.

Second, the comparison of MelFB and MF-PNFB was investigated for clean training. As expected, the MF-PNFB features clearly outperforms the MelFB baseline for all the acoustic modelings considered. In particular, for the ResNet architecture the use of MF-PNFB with respect to MelFB obtains a relative error reduction of 35.62%, which is statistically significant. It is worth noting that the result obtained by MelFB in combination to Resnet does not outperform the traditional GMM-HMM with MF-PNCC features. This observation indicates that the performance of ASR systems based on deep neural networks still suffer from important degradations when the mismatch between train and test data is high. Of course, this deterioration can be partially solved in some cases by the application of dataset augmentation techniques if some priors over the test data distribution are available. Nevertheless, when this solution is not feasible, the use of robust acoustic features (in particular, MF-PNFB) in deep neural networks architectures is helpful for reducing the error rate of the ASR system when mismatches between train and test data occur.

Third, the comparison of MelFB and MF-PNFB was evaluated for the multi-condition training scenario. Results show that for deep neural networks ASR systems, the use of MF-PNFB produces small gains with respect to the conventional MelFB. For example, MF-PNFB achieves a relative error reduction of 2.13% with respect to MelFB for the ResNet architecture, although this performance difference is not statistically significant. A reason for this behaviour is that, in general, when the train and test data distributions are similar, the deep architectures can properly extract the more suitable features by themselves and, in consequence, the application of robust techniques on the feature extraction stage does not help significantly. In particular, in the Aurora-4 the multi-condition train set has the same noises at different SNRs than the test set, this allows us to conclude that the deep architectures can generalize under different signal noise scenarios when those particular noises are shown in the training stage. Nevertheless, it is worth mentioning that MF-PNFB do not damage the recognition rates, suggesting that its use is advisable to obtain a performance gain whenever it is plausible that the test data changes drastically from the train data as, for example, in real situations where channel and noise may vary over time.

To conclude, our best system (ResNet + MF-PNFB) attains a better relative error reduction than other state-of-the-art techniques for both, matched and mismatched cases in the Aurora-4 database. In comparison, for instance, we obtain better recognition rates than the features based on Locally-Normalized Filterbanks [23] in clean and multi-condition training, although the DNN-based ASR system in [23] is trained with alignments from the clean set in both scenarios. Also, our system outperforms in a significant amount the very deep convolutional networks (vd6 baseline) presented in [8] for both cases, clean and multi-condition training.

6. Conclusions and Future Work

In this paper, Deep Residual Networks (ResNet) are employed for robust speech recognition in a hybrid ASR system showing a better performance than standard DNNs and state of the art CNNs in both matched and mismatched conditions using the Aurora-4 dataset. This behaviour is due to their well known convergence properties and generalization capabilities that allow a better modeling of speech variability. The other main contribution of this work is the use of robust input features in combination to deep neural network architectures. In particular, a modification of the Power Normalized Cepstral Coefficients (PNCC) with a masking modeling based on morphological filtering is proposed, achieving significant improvements with respect the conventional features when the mismatch between train and test data is high. Further lines of research include the assessment of the ResNet-based ASR system in larger datasets and its combination with recursive hidden units.

7. Acknowledgements

This contribution has been supported by an Airbus Defense and Space Grant (Open Innovation - SAVIER) and Spanish Government-CICYT project TEC2014-53390-P. We are grateful to NVIDIA corporation for supporting our research by donating a GeForce Titan X.

8. References

[1] G. E. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech

recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, 2012.

[2] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 20, no. 1, 2012.

[3] A. Mohamed, G. E. Dahl, and G. E. Hinton, "Acoustic modeling using deep belief networks," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 20, no. 1, 2012.

[4] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "Achieving human parity in conversational speech recognition," *CoRR*, vol. abs/1610.05256, 2016. [Online]. Available: http://arxiv.org/abs/1610.05256

[5] Y. LeCun and Y. Bengio, "The handbook of brain theory and neural networks," M. A. Arbib, Ed. Cambridge, MA, USA: MIT Press, 1998, ch. Convolutional Networks for Images, Speech, and Time Series, pp. 255–258. [Online]. Available: http://dl.acm.org/citation.cfm?id=303568.303704

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, p. 2012.

[7] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 1–9.

[8] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778.

[10] O. Abdel-Hamid, A. r. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 10, pp. 1533–1545, Oct 2014.

[11] Y. Qian, M. Bi, T. Tan, and K. Yu, "Very deep convolutional neural networks for noise robust speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 12, pp. 2263–2276, Dec 2016.

[12] L. Tóth, "Convolutional deep maxout networks for phone recognition," in *INTERSPEECH*. ISCA, 2014, pp. 1078–1082.

[13] T. N. Sainath, A. r. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for lvcsr," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 8614–8618.

[14] T. Drugman, Y. Stylianou, L. Chen, X. Chen, and M. J. F. Gales, "Robust excitation-based features for automatic speech recognition," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2015, pp. 4664–4668.

[15] M. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *ICASSP 2013*. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), January 2013.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," *CoRR*, vol. abs/1603.05027, 2016. [Online]. Available: http://arxiv.org/abs/1603.05027

[17] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoust., Speech and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[18] H. Hermansky, "Perceptual linear predictive (PLP) anlysis of speech," *J. Acoustic. Soc. Amer.*, vol. 87, no. 4, pp. 1738–1752, 1990.

[19] H. Yin, V. Hohmann, and C. Nadeu, "Acoustic features for speech recognition based on gammatone filterbank and instantaneous frequency," *Speech Communication*, vol. 53, no. 5, pp. 707 – 715, 2011.

[20] C. Kim and R. M. Stern, "Power-normalized cepstral coefficients (PNCC) for robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, March 2012, pp. 4101–4104.

[21] ——, "Power-normalized cepstral coefficients (PNCC) for robust speech recognition," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 24, no. 7, pp. 1315–1329, July 2016.

[22] V. Mitra, H. Franco, R. Stern, J. V. Hout, L. Ferrer, M. Graciarena, W. Wang, D. Vergyri, A. Alwan, and J. H. Hansen, "Robust features in deep learning-based speech recognition."

[23] J. Fredes, J. Novoa, S. King, R. M. Stern, and N. B. Yoma, "Locally normalized filter banks applied to deep neural-network-based robust speech recognition," *IEEE Signal Processing Letters*, vol. 24, no. 4, pp. 377–381, April 2017.

[24] F. de-la Calle-Silos, F. Valverde-Albacete, A. Gallardo-Antolin, and C. Pelaez-Moreno, "Morphologically filtered power-normalized cochleograms as robust, biologically inspired features for asr," *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, vol. 23, no. 11, pp. 2070–2080, Nov 2015.

[25] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, 2015, pp. 448–456. [Online]. Available: http://jmlr.org/proceedings/papers/v37/ioffe15.html

[26] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, 2010, pp. 807–814.

[27] N. Parihar and J. Picone, "Aurora working group: DSR front end LVCSR evaluation AU/384/02," *Inst. for Signal and Information Process, Mississippi State University, Tech. Rep*, vol. 40, p. 94, 2002.

[28] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, Dec. 2011.

[29] T. N. Sainath, B. Kingsbury, G. Saon, H. Soltau, A. rahman Mohamed, G. Dahl, and B. Ramabhadran, "Deep convolutional neural networks for large-scale speech tasks," *Neural Networks*, vol. 64, pp. 39 – 48, 2015, special Issue on Deep Learning of Representations.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[31] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS10). Society for Artificial Intelligence and Statistics*, 2010.

[32] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/

[33] Y. Miao, "Kaldi+PDNN: Building DNN-based ASR systems with Kaldi and PDNN," *CoRR*, 2014.

[34] V. Renkens. Kaldi with tensorflow neural net. [Online]. Available: https://github.com/vrenkens/tfkaldi

[35] F. de-la Calle-Silos. Personal web. [Online]. Available: http://www.tsc.uc3m.es/ fsilos/