Quaternion Neural Networks for Multi-channel Distant Speech Recognition

Xinchi Qiu[1]*, Titouan Parcollet*[1]*, Mirco Ravanelli*[3]*, Nicholas Lane*[1,2]*, Mohamed Morchid*[4]

[1]University of Oxford, United-Kingdom
[2]Samsung AI, Cambridge, United-Kingdom
[3]Mila, Université de Montréal, Canada
[4]LIA, Avignon University, France

xinchi.qiu@wolfson.ox.ac.uk

Abstract

Despite the significant progress in automatic speech recognition (ASR), distant ASR remains challenging due to noise and reverberation. A common approach to mitigate this issue consists of equipping the recording devices with multiple microphones that capture the acoustic scene from different perspectives. These multi-channel audio recordings contain specific internal relations between each signal. In this paper, we propose to capture these inter- and intra- structural dependencies with quaternion neural networks, which can jointly process multiple signals as whole quaternion entities. The quaternion algebra replaces the standard dot product with the Hamilton one, thus offering a simple and elegant way to model dependencies between elements. The quaternion layers are then coupled with a recurrent neural network, which can learn long-term dependencies in the time domain. We show that a quaternion long-short term memory neural network (QLSTM), trained on the concatenated multi-channel speech signals, outperforms equivalent real-valued LSTM on two different tasks of multi-channel distant speech recognition.

Index Terms: distant speech recognition, quaternion neural networks, multi-microphone speech recognition.

1. Introduction

State-of-the-art speech recognition systems perform reasonably well in close-talking conditions. However, their performance degrades significantly in more realistic distant-talking scenarios, since the signals are corrupted with noise and reverberation [1, 2]. A common approach to improve the robustness of distant speech recognizers relies on the adoption of multiple microphones [3, 4]. Multiple microphones, either in the form of arrays or distributed networks, capture different views of an acoustic scene that are combined to improve robustness.

A common practice is to combine the microphones using signal processing techniques such as beamforming [5]. The goal of beamforming is to achieve spatial selectivity (*i.e.*, privilege the areas where a target speaker is speaking), limiting the effects of both noise and reverberation. One way to perform spatial filtering is provided by the delay-and-sum beamforming, which simply performs a time alignment followed by a sum of the recorded signals [6]. More sophisticated techniques are filter-and-sum beamforming [7], that filters the signal before summing them up, and super-directive beamforming [8], which further enhances the target speech by suppressing the contributions of the noise sources from other directions.

An alternative that is gaining significant popularity is End-to-end (E2E) multi-channel ASR [9, 10, 11, 12, 13, 14]. Here, the core idea is to replace the signal processing part with an end-to-end differentiable neural network, that is jointly trained with the speech recognizer. It will make the speech processing pipeline significantly simpler, and different modules composing the whole system match better with each other. The most straightforward approach is concatenating the speech features of the different microphones and feeding them to a neural network [15]. However, this approach forces the network to deal with very high-dimensional data, and might thus make learning the complex relationships between microphones difficult due to numerous independent neural parameters. To mitigate this issue, it is common to inject prior knowledge or inductive biases into the model. For instance, [16, 17] suggested to use complex linear projection to perform filtering and pooling in the frequency domain, and [11] suggested an adaptive neural beamformer that performs filter-and-sum beamforming using learned filters. In all aforementioned works, the microphone combination is not implemented with an arbitrary function, but a restricted pool of functions. This introduces a regularization effect that helps the convergence of the speech recognizer.

In this paper, we propose a novel approach to model the complex inter- and intra- microphone dependencies that occur in multi-microphone ASR. Our inductive bias relies on the use of quaternion algebra. Quaternions extend complex numbers and define four-dimensional vectors composed of a real part and three imaginary components. The standard dot product is replaced with the Hamilton product that offers a simple and elegant way to learn dependencies across input channels by sharing weights across them. More precisely, Quaternion Neural Networks (QNN) have recently been the object of several research efforts focusing on image processing [18, 19], 3D sound event detection [20] and single-channel speech recognition [21]. To the best of our knowledge, our work is the first that proposes the use of quaternions in a multi-microphone speech processing scenario, which is a particularly suitable application. Our approach combines the speech features extracted from different channels into four different dimensions of a set of quaternions (Section 2.3). We then employ a Quaternion Long-Short Term Memory (QLSTM) neural network [21]. This way, our architecture not only models the latent intra- and inter- microphone correlations with the quaternion algebra, but also jointly learns time-dependencies with recurrent connections.

Our QLSTM achieves promising results on both a simulated version of TIMIT and the DIRHA corpus [22], which are characterized by the presence of significant levels of non-stationary noises and reverberation. In particular, we outperform both a beamforming baseline (15% relative improvement) and a real-valued model with the same number of parameters (8% relative improvement). In the interest of reproducibility, we release the code under PyTorch-Kaldi [23] [1].

[1] https://github.com/mravanelli/pytorch-kaldi/

2. Methodology

This section first describes the quaternion algebra (Section 2.1) and quaternion long short-term memory neural networks (Section 2.2). Finally, the quaternion representation of multi-channel signals is introduced in Section 2.3.

2.1. Quaternion Algebra

A quaternion is an extension of a complex number to the four-dimensional space [24]. A quaternion Q is written as:

$$Q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}, \tag{1}$$

with a, b, c, and d four real numbers, and 1, **i**, **j**, and **k** the quaternion unit basis. In a quaternion, a is the real part, while $b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$ with $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$ is the imaginary part, or the vector part. Such definition can be used to describe spatial rotations. In the same manner as complex numbers, the conjugate Q^* of Q is defined as:

$$Q^* = a - b\mathbf{i} - c\mathbf{j} - d\mathbf{k}, \tag{2}$$

and a unitary quaternion (*i.e.* whose norm is equal to 1) is defined as:

$$Q^{\triangleleft} = \frac{Q}{\sqrt{a^2 + b^2 + c^2 + d^2}}. \tag{3}$$

The Hamilton product between $Q_1 = a_1 + b_1\mathbf{i} + c_1\mathbf{j} + d_1\mathbf{k}$ and $Q_2 = a_2 + b_2\mathbf{i} + c_2\mathbf{j} + d_2\mathbf{k}$ is determined by the products of the basis elements and the distributive law:

$$\begin{aligned} Q_1 \otimes Q_2 = &(a_1a_2 - b_1b_2 - c_1c_2 - d_1d_2) \\ &+ (a_1b_2 + b_1a_2 + c_1d_2 - d_1c_2)\mathbf{i} \\ &+ (a_1c_2 - b_1d_2 + c_1a_2 + d_1b_2)\mathbf{j} \\ &+ (a_1d_2 + b_1c_2 - c_1b_2 + d_1a_2)\mathbf{k}. \end{aligned} \tag{4}$$

Analogously to complex numbers, quaternions also have a matrix representation defined in a way that quaternion addition and multiplication correspond to a matrix addition and a matrix multiplication. An example of such matrix is:

$$Q_{\text{mat}} = \begin{bmatrix} a & -b & -c & -d \\ b & a & -d & c \\ c & d & a & -b \\ d & -c & b & a \end{bmatrix}. \tag{5}$$

Following this representation, the Hamilton product can be written as a matrix multiplication as follow:

$$Q_1 \otimes Q_2 = \begin{bmatrix} a_1 & -b_1 & -c_1 & -d_1 \\ b_1 & a_1 & -d_1 & c_1 \\ c_1 & d_1 & a_1 & -b_1 \\ d_1 & -c_1 & b_1 & a_1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \\ c_2 \\ d_2 \end{bmatrix}. \tag{6}$$

Using the matrix representation of quaternions turns out to be particularly suitable for computations on modern GPUs compared to the less efficient object programming.

2.2. Quaternion Long Short-Term Memory Networks

Equivalently to standard LSTM models, a QLSTM consists of a forget gate f_t, an input gate i_t, a cell input activation vector $\tilde{C}_t$, a cell state C_t and an output gate o_t. In a QLSTM layer, however, inputs x, hidden states h_t, cell states C_t, biases b, and weight parameters W are quaternion numbers. All multiplications are thus replaced with the Hamilton product. Different activation functions defined in the quaternion domain can be used

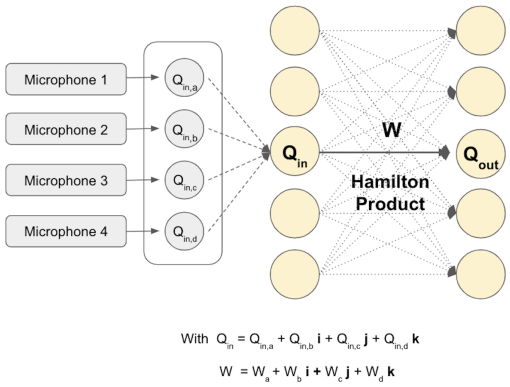

Figure 1: *Illustration of the integration of multiple microphones with a quaternion dense layer. Each microphone is encapsulated by one component of a set of quaternions. All the neural parameters are quaternion numbers.*

[25, 19]. In this work, we follow the split approach defined as:

$$\alpha(Q) = \alpha(a) + \alpha(b)\mathbf{i} + \alpha(c)\mathbf{j} + \alpha(d)\mathbf{k}, \tag{7}$$

where α is any real-valued activation function (*i.e.* ReLU, Sigmoid, ...). Indeed, fully quaternion-valued activation functions have been demonstrated to be hard to train due to numerous singularities [19]. Then, the output layer is commonly defined in the real-valued space to be combined with traditional loss functions (*e.g.* cross-entropy) [26] due to the real-valued nature of the labels implied by the considered speech recognition task. Therefore, a QLSTM layer can be summarised with the following equations:

$$\begin{aligned} f_t &= \sigma(W_{fh} \otimes h_{t-1} + W_{fx} \otimes x_t + b_f), \\ i_t &= \sigma(W_{ih} \otimes h_{t-1} + W_{ix} \otimes x_t + b_i), \\ \tilde{C}_t &= tanh(W_{Ch} \otimes h_{t-1} + W_{Cx} \otimes x_t + b_C), \\ C_t &= f_t \otimes C_{t-1} + i_t \otimes \tilde{C}_t, \\ o_t &= \sigma(W_{oh} \otimes h_{t-1} + W_{ox} \otimes x_t + b_o), \\ h_t &= o_t \otimes tanh(C_t), \end{aligned} \tag{8}$$

with two split activations σ and $tanh$ as described in Eq. 7. As shown in [21], QLSTM models can be trained following the quaternion-valued backpropagation through time. Finally, weight initialisation is crucial to train deep neural networks effectively [27]. Hence, a well-adapted quaternion weight initialisation process [21, 28] is applied. Quaternion neural parameters are sampled with respect to their polar form and a random distribution following common initialization criteria [27, 29].

2.3. Quaternion Representation of Multi-channel Signals

We propose to use quaternion numbers in a multi-microphone speech processing scenario. More precisely, quaternion numbers offer the possibility to encode up to four microphones (Fig. 1). Therefore, common acoustic features (*e.g.* MFCCs, FBANKs, ...) are computed from each microphone signal $M_{1,2,3,4}$, and then concatenated to compose a quaternion as follows:

$$Q = M_{1,a} + M_{2,b}\mathbf{i} + M_{3,b}\mathbf{j} + M_{4,b}\mathbf{k} \tag{9}$$

Internal relations are captured with the specific weight sharing property of the Hamilton product. By using Hamilton products, quaternion-weight components are shared through multiple quaternion-input, creating relations within the elements as demonstrated in [21]. More precisely, real-valued network inputs are treated as a group of uni-dimensional elements that could be related to each other, potentially decorrelating the four microphone signals. Conversely, quaternion networks consider each time frame as an entity of four related elements. Hence, internal relations are naturally captured and learned through the process. Indeed, a small variation in one of the microphone would result in an important change in the internal representation affecting the encoding of the three other microphones.

It is worth noticing that four microphones may be limiting for realistic applications. For instance, the latest CHIME-6 challenge [30] proposes various recordings obtained from six microphones in different scenarios. This difficulty could be easily avoided by considering these tasks as a special case of higher algebras, such as octonions (eight dimensions) or sedenions (sixteen dimensions). Nevertheless, this paper proposes to first consider four dimensions to evaluate the viability of the application of high-dimensional neural networks for distant and multi-microphone ASR. Finally, quaternion neural networks are known to be more computationally intensive than real-valued neural networks. Indeed, the Hamilton product involves 28 basic operations compared to 1 for a standard product. Nonetheless, the training time can be reduced with the matrix representation defined in Eq.(6), and can be drastically improved with simple linear algebra properties [31].

3. Experimental Protocol

A perturbed speech and multi-channel TIMIT [32] version presented thereafter is first used as a preliminary task to investigate the impact of the Hamilton product. Then, the DIRHA dataset [33] is used to verify the scalability of the proposed approach to more realistic conditions.

3.1. TIMIT Dataset

The TIMIT corpus contains broadband recordings of 630 speakers of eight main dialects of American English, each reading ten phonetically rich sentences. The training dataset consists of the standard 3696 sentences uttered by 462 speakers, while the testing one consists of 192 sentences uttered by 24 speakers. A validation dataset composed of 400 sentences uttered by 50 speakers is used for hyper-parameter tuning.

In our experiments, we created a multi-channel simulated version of TIMIT using the impulse responses measured in [34, 35][2]. The reference environment is a living room of a real apartment with an average reverberation time T_{60} of 0.7 seconds. The considered four microphones (*i.e.* $LA2$, $LA3$, $LA4$, $LA5$) are placed on the ceiling of the room. Data are created considering all the different positions, and different positions are used for training and testing data. We also integrate a single-channel signal obtained with delay-and-sum beamforming as a baseline comparison [6]. Input features consist of 40 Mel filters bank energies (FBANK) with no deltas extracted with Kaldi [36]. To show that the obtained gain in performance is independent of the input features, we also propose 13 MFCC coefficients as an alternative set of features.

[2]Perturbation can be re-created following: `https://github.com/SHINE-FBK/DIRHA_English_wsj`

3.2. DIRHA Dataset

To validate our model in a more realistic scenario, a set of experiments is also conducted with the larger DIRHA-English corpus [22]. Equivalently to the generated TIMIT dataset, the reference context is a domestic environment characterized by the presence of non-stationary noise and acoustic reverberation. Training is based on the original Wall-Street-Journal-5k (WSJ) corpus (*i.e.* consisting of 7138 sentences uttered by 83 speakers) contaminated with a set of impulse responses measured in a real apartment [37, 38]. Both a real and a simulated dataset are used for testing, each consisting of 409 WSJ sentences uttered by six native American speakers. Note that a validation set of 310 WSJ sentences is used for hyper-parameter tuning. Only the first four microphones of the circular array are used in our experiments to fit the quaternion representation. A single-channel signal obtained with delay-and-sum beamforming is also proposed as a baseline comparison [6]. It is worth noting that we also used 13 MFCC coefficients as features in comparison to FBANKs to evaluate the robustness of the model to the input representation.

3.3. Neural Network Architectures

We decided to fix the number of neural parameters to 5M for both LSTM and QLSTM following the models studied in [21]. Therefore, the QLSTM model is composed of 4 bidirectional QLSTM layers followed by a linear layer with a softmax activation function for classification. Output labels are the different HMM states of the Kaldi decoder. Each of the QLSTM layers consists of 128 quaternion nodes. Although there are $128 * 4 = 512$ real-valued nodes in total, there are only $128 * 128 * 4$ real-valued weight parameters, due to the weight sharing property of quaternion neural networks. The LSTM model is composed of 4 bidirectional LSTM layers of size 290 (*i.e.* ensuring the same number of neural parameters as the QLSTM) followed by the same linear layer to obtain posterior probabilities. A dropout rate of 0.2 is applied across all (Q)LSTM layers. Quaternion parameters are initialised with the specific initialisation defined in [21], while LSTM parameters are initialised with the Glorot criterion [27].

Training is performed with the RMSPROP optimizer with vanilla hyper-parameters and an initial learning rate of $1.6e^{-3}$ over 24 epochs. The learning-rate is halved every time the loss on the validation set increases, ensuring an optimal convergence. Finally, both LSTM and QLSTM are manually implemented in PyTorch to alleviate any variation due to different implementations.

4. Results and Discussions

The results on the distant multi-channel TIMIT dataset are reported in Table 1. From this comparison, it emerges that QLSTM with four microphones outperforms the other approaches. Our best QLSTM model, in fact, obtains a PER of 28.7% against a PER of 30.2% achieved with a standard real-value LSTM. In both cases, the best performance is obtained with FBANK features. Interestingly, Table 1 shows that the concatenation of the four input signals with a real-valued LSTM outperforms the delay-and-sum beamforming approach. Similar achievements have already emerged in previous works on multi-channel ASR [15] and can be due to the ability of modern neural networks to obtain disentangled and informative representations from noisy inputs.

We can now investigate in more detail the role played by the quaternion algebra on learning cross-microphone dependen-

Table 1: *Results expressed in terms of Phoneme Error Rate (PER) percentage (i.e lower is better) of both QLSTM and LSTM models on the TIMIT distant phoneme recognition task with different acoustic features. Results are from an average of* 5 *runs.*

Models	**Signals**	**Test (FBANK)**	**Test (MFCC)**
QLSTM	1 microphone copied	32.1 ± 0.02	34.2 ± 0.13
LSTM	1 microphone	32.3 ± 0.14	35.0 ± 0.23
LSTM	beamforming	31.1 ± 0.11	33.4 ± 0.07
LSTM	4 microphones	30.2 ± 0.16	32.8 ± 0.09
QLSTM	**4 microphones**	$\mathbf{28.7 \pm 0.06}$	$\mathbf{30.4 \pm 0.11}$

Table 2: *Results expressed in terms of Word Error Rate (WER) (i.e lower is better) of both QLSTM and LSTM based models on the DIRHA dataset with different acoustic features. 'Test Sim.' corresponds to the simulated test set of the corpus, while "Test Real" is the set composed of real recordings.*

Models	**Signals**	**Test Real (MFCC)**	**Test Sim. (MFCC)**	**Test Real (FBANK)**	**Test Sim. (FBANK)**
LSTM	beamforming	35.1	33.7	35.0	33.0
LSTM	4 microphones	32.7	26.4	31.6	26.3
QLSTM	**4 microphones**	**29.8**	**23.8**	**29.7**	**23.4**

cies. One way to do it is to overwrite the quaternion dimensions with the features extracted from the same microphone (see the first row of Table 1). In this case, we expect that our QLSTM will fail to learn cross-microphone dependencies, simply because we have a single feature vector replicated multiple times. For a fair comparison, the aforementioned experiment is conducted by selecting the best microphone of the array (*i.e.* LA4).

From the first and the second rows of Table 1, one can note that both single-channel QLSTM and LSTM perform roughly the same. As expected, in fact, the single-channel QLSTM is not able to model useful dependencies when the quaternion dimensions are dumped with the same feature vector. Nonetheless, switching to four-channel signal brings an average PER improvement of 3.6% for the QLSTM compared to 2.1% for the LSTM, showing a higher gain obtained on multiple channels with the QLSTM. This illustrates the ability of QLSTM to better capture latent relations across the different microphones.

To provide some experimental evidence on a more realistic task, we evaluate our model with the DIRHA dataset. The results obtained in Table 2 confirm the trend observed with TIMIT. Indeed, Word Error Rates (WER) of 29.8% and 23.8% are obtained for the QLSTM on the real and simulated test sets respectively, compared to 32.7% and 26.4% for the equivalent real-valued LSTM. The same remark holds while feeding our models with FBANK features with a best WER of 29.7 obtained with the QLSTM compared to 31.6. As a side note, the accuracies reported on Table 2 are slightly worse compared to the ones given in [23]. Indeed, the latter work includes a specific batch-normalisation that is not applied in our experiments due to the very high complexity of the Quaternion Batch-Normalisation (QBN) introduced in [39]. As a matter of fact, the current equations of the QBN induce an increase of the VRAM consumption by a factor of 4. As expected, WER observed on the real test set are also higher than those on the simulated one, due to more complex and realistic perturbations.

As shown in both TIMIT and DIRHA experiments, the performance improvement observed with the QLSTM is independent of the initial acoustic representation, implying that a similar increase of accuracy may be expected with other acoustic features such as fMLLR or PLP. Interestingly, the single-channel beamforming approach gives the worst results among all the investigated methods on both TIMIT and DIRHA.

5. Conclusion

Summary. This paper proposed to perform multi-channel speech recognition with an LSTM based on quaternion numbers. Our experiments, conducted on multi-channel TIMIT and DIRHA have shown that: 1) Given the same number of parameters, our multi-channel QLSTM significantly outperforms an equivalent LSTM network; 2) the performance improvement is observed with different features, implying that a similar increase of accuracy may be expected with others acoustic representations such as fMLLR or PLP; 3) our QLSTM learns internal latent relations across microphones. Therefore, the initial intuition that quaternion neural networks are suitable for multi-channel distant automatic speech recognition has been verified.
Perspectives. One limitation of the current approach is due to the fact that quaternion neural networks can only deal with four-dimensional input signals. Even though popular devices such as the Microsoft Kinect, or the ReSpeaker are based on 4-microphones arrays, future efforts will focus on generalising this paradigm to an arbitrary number of microphones by considering, for instance, higher dimensional algebras such as octonions and sedenions, or by investigating other methods of weight sharing for multi-channel ASR. Finally, despite recent works on investigating efficient quaternion computations, the current training and inference processes of the QLSTM remain slower than that of a LSTM. Therefore, efforts should be put in developing and implementing faster training procedures.

6. Acknowledgements

This work was supported by the EPRSC through MOA (EP/S001530/), Samsung AI and via the AISSPER project through the French National Research Agency (ANR) under Contract AAPG 2019 ANR-19-CE23-0004-01. We would also like to thank Elena Rastorgueva and Renato De Mori for the helpful comments and discussions.

7. References

[1] M. Wölfel and J. W. McDonough, *Distant speech recognition*. Wiley Online Library, 2009.

[2] J. Li, L. Deng, R. Haeb-Umbach, and Y. Gong, *Robust Automatic Speech Recognition - A Bridge to Practical Applications (1st Edition)*, October 2015.

[3] M. Brandstein and D. Ward, *Microphone arrays: signal processing techniques and applications*. Springer Science & Business Media, 2013.

[4] J. Benesty, J. Chen, and Y. Huang, *Microphone array signal processing*. Springer Science & Business Media, 2008, vol. 1.

[5] W. Kellermann, *Beamforming for Speech and Audio Signals*, 2008.

[6] C. H. Knapp and G. C. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.

[7] M. Kajala and M. Hamalainen, "Filter-and-sum beamformer with adjustable filter characteristics," in *Proc. of ICASSP*, 2001, pp. 2917–2920.

[8] J. Bitzer and K. Simmer, "Superdirective microphone arrays," in *Microphone Arrays*. Springer Berlin Heidelberg, 2001, pp. 19–38.

[9] J. Heymann, L. Drude, C. Boeddeker, P. Hanebrink, and R. Haeb-Umbach, "Beamnet: End-to-end training of a beamformer-supported multi-channel asr system," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5325–5329.

[10] S. Braun, D. Neil, J. Anumula, E. Ceolini, and S.-C. Liu, "Multi-channel attention for end-to-end speech recognition," *2018 Interspeech*, pp. 0–0, 2018.

[11] B. Li, T. N. Sainath, R. J. Weiss, K. W. Wilson, and M. Bacchiani, "Neural network adaptive beamforming for robust multichannel speech recognition," 2016.

[12] T. Ochiai, S. Watanabe, T. Hori, J. R. Hershey, and X. Xiao, "Unified architecture for multichannel end-to-end speech recognition with neural beamforming," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1274–1288, 2017.

[13] X. Xiao, S. Watanabe, H. Erdogan, L. Lu, J. Hershey, M. L. Seltzer, G. Chen, Y. Zhang, M. Mandel, and D. Yu, "Deep beamforming networks for multi-channel speech recognition," in *Proc. of ICASSP*, 2016, pp. 5745–5749.

[14] S. Kim and I. Lane, "End-to-end speech recognition with auditory attention for multi-microphone distance speech recognition," in *Proc. Interspeech 2017*, 2017.

[15] Y. Liu, P. Zhang, and T. Hain, "Using neural network front-ends on far field multiple microphones based speech recognition," in *Proc. of ICASSP*, 2014, pp. 5542–5546.

[16] B. Li, T. N. Sainath, A. Narayanan, J. Caroselli, M. Bacchiani, A. Misra, I. Shafran, H. Sak, G. Pundak, K. K. Chin *et al.*, "Acoustic modeling for google home." in *Interspeech*, 2017, pp. 399–403.

[17] E. Variani, T. N. Sainath, I. Shafran, and M. Bacchiani, "Complex linear projection (clp): A discriminative approach to joint feature extraction and acoustic modeling," 2016.

[18] T. Isokawa, T. Kusakabe, N. Matsui, and F. Peper, "Quaternion neural network and its application," in *Internationalschrank2016deep conference on knowledge-based and intelligent information and engineering systems*. Springer, 2003, pp. 318–324.

[19] T. Parcollet, M. Morchid, and G. Linarès, "A survey of quaternion neural networks," *Artificial Intelligence Review*, pp. 1–26, 2019.

[20] D. Comminiello, M. Lella, S. Scardapane, and A. Uncini, "Quaternion convolutional neural networks for detection and localization of 3d sound events," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 8533–8537.

[21] T. Parcollet, M. Ravanelli, M. Morchid, G. Linarès, C. Trabelsi, R. De Mori, and Y. Bengio, "Quaternion recurrent neural networks," *arXiv preprint arXiv:1806.04418*, 2018.

[22] M. Ravanelli, L. Cristoforetti, R. Gretter, M. Pellin, A. Sosi, and M. Omologo, "The dirha-english corpus and related tasks for distant-speech recognition in domestic environments," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp. 275–282.

[23] M. Ravanelli, T. Parcollet, and Y. Bengio, "The pytorch-kaldi speech recognition toolkit," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6465–6469.

[24] W. R. Hamilton and C. J. Joly, *Elements of quaternions*. Longmans, Green, and Company, 1899, vol. 1.

[25] P. Arena, L. Fortuna, L. Occhipinti, and M. G. Xibilia, "Neural networks for quaternion-valued function approximation," in *Proceedings of IEEE International Symposium on Circuits and Systems-ISCAS'94*, vol. 6. IEEE, 1994, pp. 307–310.

[26] P. Arena, L. Fortuna, G. Muscato, and M. G. Xibilia, "Multilayer perceptrons to approximate quaternion valued functions," *Neural Networks*, vol. 10, no. 2, pp. 335–342, 1997.

[27] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.

[28] T. Parcollet, Y. Zhang, M. Morchid, C. Trabelsi, G. Linarès, R. De Mori, and Y. Bengio, "Quaternion convolutional neural networks for end-to-end automatic speech recognition," *arXiv preprint arXiv:1806.07789*, 2018.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[30] S. Watanabe, M. Mandel, J. Barker, and E. Vincent, "Chime-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings," *arXiv preprint arXiv:2004.09249*, 2020.

[31] A. Cariow and G. Cariowa, "Fast algorithms for quaternion-valued convolutional neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[32] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "Darpa timit acoustic-phonetic continous speech corpus cd-rom. nist speech disc 1-1.1," *NASA STI/Recon technical report n*, vol. 93, 1993.

[33] L. Cristoforetti, M. Ravanelli, M. Omologo, A. Sosi, A. Abad, M. Hagmüller, and P. Maragos, "The dirha simulated corpus." in *LREC*, 2014, pp. 2629–2634.

[34] M. Ravanelli, A. Sosi, P. Svaizer, and M. Omologo, "Impulse response estimation for robust speech recognition in a reverberant environment," in *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*. IEEE, 2012, pp. 1668–1672.

[35] M. Ravanelli and M. Omologo, "On the selection of the impulse responses for distant-speech recognition based on contaminated speech training," in *Proc. of Interspeech*, 2014.

[36] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. CONF. IEEE Signal Processing Society, 2011.

[37] M. Ravanelli, P. Svaizer, and M. Omologo, "Realistic multi-microphone data simulation for distant speech recognition," *arXiv preprint arXiv:1711.09470*, 2017.

[38] M. Ravanelli and M. Omologo, "Contaminated speech training methods for robust dnn-hmm distant speech recognition," *arXiv preprint arXiv:1710.03538*, 2017.

[39] C. J. Gaudet and A. S. Maida, "Deep quaternion networks," in *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018, pp. 1–8.