Interspeech 2022
18-22 September 2022, Incheon, Korea

Pushing the limits of raw waveform speaker recognition

Jee-weon Jung[1,*], *You Jin Kim*[1,*], *Hee-Soo Heo*[1], *Bong-Jin Lee*[1], *Youngki Kwon*[1], *Joon Son Chung*[2]

[1]Naver Corporation, South Korea
[2]Korea Advanced Institute of Science and Technology, South Korea
{jeeweon.jung, youjin.kim117}@navercorp.com, joonson@kaist.ac.kr

Abstract

In recent years, speaker recognition systems based on raw waveform inputs have received increasing attention. However, the performance of such systems are typically inferior to the state-of-the-art handcrafted feature-based counterparts, which demonstrate equal error rates under 1% on the popular VoxCeleb1 test set. This paper proposes a novel speaker recognition model based on raw waveform inputs. The model incorporates recent advances in machine learning and speaker verification, including the Res2Net backbone module and multi-layer feature aggregation. Our best model achieves an equal error rate of 0.89%, which is competitive with the state-of-the-art models based on handcrafted features, and outperforms the best model based on raw waveform inputs by a large margin. We also explore the application of the proposed model in the context of self-supervised learning framework. Our self-supervised model outperforms single phase-based existing works in this line of research. Finally, we show that self-supervised pre-training is effective for the semi-supervised scenario where we only have a small set of labelled training data, along with a larger set of unlabelled examples.

Index Terms: raw waveform, speaker recognition, speaker verification, self-supervised learning, semi-supervised learning

1. Introduction

Despite the increasing popularity of end-to-end machine learning, the use of handcrafted features (such as MFCCs or mel-filterbanks) remain prevalent in many speech processing tasks [1–6]. However, models that directly operate upon raw waveforms are becoming more common across several speech-related fields with the recent advances in deep learning. The majority of the existing works in the area have used vanilla convolutional layers or parameterised filterbank layers to process raw waveform inputs [7–10].

The trend towards end-to-end models can also be observed in the speaker recognition literature. Since the first raw waveform speaker recognition model that employs a VGGNet-styled vanilla convolutional layer [11], several studies now adopt direct modelling from raw waveforms [12–16].

However, most raw waveform speaker recognition models suffer from performance degradation compared to handcrafted feature-based counterparts. Even the latest architecture demonstrates equal error rate (EER) of 1.29% [16], whereas the widely adopted ECAPA-TDNN architecture and its variants have consistently reported EERs under 1% [2, 17, 18]. We therefore propose a new model architecture combining several recent advances in deep learning together with RawNet2 to overcome this challenge.

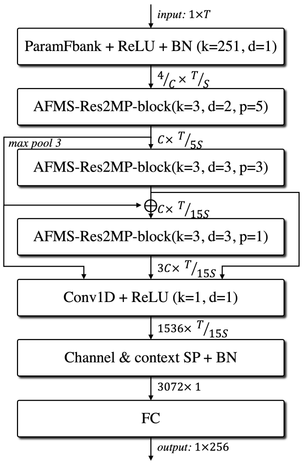
input: 1×T
ParamFbank + ReLU + BN (k=251, d=1)
4/C × T/S
AFMS-Res2MP-block(k=3, d=2, p=5)
max pool 3
C× T/5S
AFMS-Res2MP-block(k=3, d=3, p=3)
C× T/15S
AFMS-Res2MP-block(k=3, d=3, p=1)
3C× T/15S
Conv1D + ReLU (k=1, d=1)
1536× T/15S
Channel & context SP + BN
3072× 1
FC
output: 1×256

Figure 1: *The RawNet3 architecture. It is in a hybrid form of the ECAPA-TDNN [2] and the RawNet2 [32] with additional features including logarithm and normalisation. k, d, p, C, S, and ⊕ correspond to kernel length, dilation, max pooling size, number of channels, stride size of the parameterised filterbank layer, and element-wise addition.*

Meanwhile, self-supervised learning has arisen as an alternative to the currently dominant supervised learning paradigm which requires human-annotated labels. Various frameworks in self-supervised learning train the model without ground truth labels (i.e., unlabelled), leveraging *pretext* tasks which can be solved by learning the inherent data properties [19]. The prevailing methods in self-supervised learning take the form of either contrastive- [7, 20, 21], reconstruction- [22, 23], or non-negative (positive pairs only)-based learning [24–26].

Self-supervised learning is also an actively ongoing field of research in speaker recognition [27–31]. However, none of these works has yet adopted an end-to-end architecture that directly operates upon raw waveform, leaving the potential open.

In this paper, we investigate various components of the raw waveform speaker recognition by exploring the aforementioned issues. Concretely, we make the below contributions:

- We propose a new raw waveform speaker recognition architecture, namely RawNet3, that demonstrates EER under 1% in the VoxCeleb1 evaluation protocol;
- We explore raw waveform speaker verification model with a self-supervised learning framework for the first time and outperform contrastive-based existing works;
- We demonstrate the effectiveness of self-supervised pre-training under semi-supervised learning scenario.

2. Proposed architecture

Figure 1 illustrates the proposed model architecture, named *RawNet3* for brevity. The architecture of RawNet3 is inspired

[*] Equal contribution

10.21437/Interspeech.2022-126

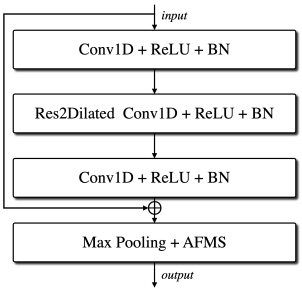

Figure 2: *The AFMS-Res2MP-block of the RawNet3 architecture. AFMS refers to the extended feature map scaling module of RawNet2.*

by techniques used in RawNet2 [32] and ECAPA-TDNN [2]. We first apply pre-emphasis to the raw waveform following [33] and feed it through an instance normalisation layer [34]. Then the output is processed into a time-frequency representation using parameterised analytic filterbanks [10] where complex-valued filterbanks are learned. This layer is an extension of the sinc-convolution layer [9], which has been adopted in RawNet2 [32], where real-valued parameterised filterbanks are learned. At this step, the extent of sequence compression is configured via the stride size, where a smaller stride slows processing but produces better performance. In contrast, a bigger stride delivers faster processing but degrades performance. The default kernel length and stride are 251 and 48, respectively, generating approximately 15 ms window and 3 ms shift per frame.

Three backbone blocks with residual connections digest the parameterised analytic filterbank layer output. The outputs of the three blocks are concatenated identical to the ECAPA-TDNN architecture. We also input the summation of the first and the second block outputs to the third block. This is similar to but slightly different from the ECAPA-TDNN, which inputs all previous block outputs as the following block input. We apply max pooling of sizes 5 and 3 to the first two backbone blocks, different to ECAPA-TDNN, following the RawNet2 architecture. The additional reduction of sequence length is mandatory in raw waveform speaker verification models because the sequence length is extremely longer than the handcraft feature-based model counterparts. The adoption of max pooling is based on our previous empirical results that it mitigates overfitting in raw waveform speaker verification models [35]. Each backbone block, illustrated in Figure 2 and referred to as AFMS-Res2MP-block, is based on Res2Net [36]. It is similar to the backbone block of the ECAPA-TDNN except for two modifications: (i) α-feature map scaling (AFMS) from RawNet2 is applied in place of squeeze-excitation [37] based on previous empirical results, and (ii) we optionally apply max pooling before applying AFMS.

After the three backbone AFMS-Res2MP-blocks, one convolution layer with batch normalisation [38] positions. For supervised learning with classification loss (Section 3.1), the classification head exists as the output layer. For self-supervised learning with the DINO framework (Section 3.2), additional DINO head layers are added.

Comparison with RawNet2 architecture. Several different design choices have been adopted in RawNet3 compared to the previous RawNet2 architecture. First, the parameterised analytic filterbank layer [10] is utilised instead of sinc-convolution layer [9]. Second, log and mean normalisation is applied to the analytic filterbank output. Third, the number of backbone blocks and their connections have been adapted, following the ECAPA-TDNN alike topology. Last, the channel and context-dependent statistic pooling replaces a uni-directional gated recurrent unit layer [39].

3. Adopted frameworks

3.1. Supervised learning: classification with AAM-softmax

The two most prevailing supervised learning frameworks in speaker verification are classification and metric learning. We adopt the classification-based training framework. In the classification framework, a model is trained as a closed set speaker identification model with a classification head that has a dimensionality equal to the number of speakers in the train dataset (i.e., the d-vector framework [40]). A categorical cross-entropy objective function is adopted, which calculates the loss by comparing the classification head where softmax non-linearity is applied with the one-hot ground truth label. After the training is complete, the classification head is removed, and the model is used as a speaker embedding extractor.

Specifically, we adopt the AAM-softmax objective function [41], also known as the ArcFace. The AAM-softmax can enforce larger gaps between the nearest speakers. Let $\mathbf{x}_i$, y_i, and $\mathbf{W}$ be the speaker embedding, its corresponding label, and the weight matrix of the classification head where i is the index of an utterance within a mini-batch of size N, $0 < i < N$. The AAM-softmax loss is formulated as:

$$\mathcal{L}_A = -\frac{1}{N}\sum_{i=1}^{N} log \frac{e^{s(cos(\theta_{y_i,i})+m)}}{e^{s(cos(\theta_{y_i,i})+m)} + \sum_{j=1,j\neq y_i}^{N} e^{s\,cos(\theta_{j,i})}} \tag{1}$$

where $cos(\theta_{j,i})$ is the dot product between $l2$ normalised $\mathbf{x}_i$ and $\mathbf{W}_j$, s and m are the hyper-parameters for a scale factor and a margin, respectively. Readers are guided to [41] for details.

3.2. Self-supervised learning: DINO

The DINO framework [25] is one of the most competitive frameworks in the recent self-supervised learning literature. It compares multiple different views generated from a single utterance like the BYOL [24] framework and employs a self-distillation method similar to the data2vec [26] framework.

DINO involves a teacher and a student network with an identical architecture but different parameters. Multi-crop training is utilised where a set of views (V), two different global (a_1^g and a_2^g) and several local views (a^l), of an utterance are exploited. The teacher digests only global views, whereas the student digests all views. Only the parameters of the student network are updated using the loss function. The parameters of the teacher network are updated as an exponential moving average of the student networks.

In the DINO framework, the outputs of both teacher and student networks are sharpened via dedicated temperatures to avoid representation collapse during training. Centring is additionally applied to the teacher output for the same purpose, which subtracts centre from the teacher output. The centre is updated using an exponential moving average. Centring uniformly flattens the probability distribution while sharpening narrow-downs the width of the distribution. It is argued that a good balance between centring and sharpening hyper-parameters leads to successful training without requiring architecture differentiation. The DINO loss is then defined as:

$$\mathcal{L}_D = \sum_{a\in\{a_1^g,a_2^g\}} \sum_{a'\in V,a'\neq a} H(P_t(a), P_s(a')), \tag{2}$$

Table 1: *Results on supervised learning using the AAM-softmax [41] objective function. Trained on VoxCeleb1&2 development sets. The two numbers in* Hz *denote frame resolutions after the first parameterised filterbank and the last max pooling layer.*

Configurations	**EER(%)**	**minDCF**
RawNet2 [32]	2.48	N/R
RawNet3 (stride=48)	1.05	0.0763
− param fbank log	1.27	0.0852
− param fbank norm	1.22	0.0838
− param fbank log&norm	1.23	0.0927
− ch&context stat pool	1.45	0.0975
→ stride=10, 1600Hz→106Hz	**0.89**	0.0669
→ stride=16, 1000Hz→66Hz	0.90	**0.0593**
→ stride=24, 666Hz→44Hz	0.96	0.0773
→ stride=64, 250Hz→16Hz	1.11	0.0851
→ stride=96, 166Hz→11Hz	1.31	0.0937

where P_t and P_s correspond to the teacher and the student network outputs and $H(\cdot)$ is the cross-entropy. Readers are guided to [25] for full details.

4. Experiments

We present three sets of experiments using the proposed RawNet3 model under various frameworks: (i) supervised learning using ground truth label-based classification (Table 1); (ii) self-supervised learning using the DINO framework (Table 2); and additionally, (iii) semi-supervised learning (Table 3). In the semi-supervised learning scenario, we first pre-train the model using the DINO self-supervised learning framework. We then fine-tune the model using supervised learning with ground truth label-based classification. We also provide comparison with recent models in the literature in Tables 4 and 5.

4.1. Dataset

We use VoxCeleb1 and 2 datasets [3, 42] throughout this paper, which contain utterances collected in the wild, including background noises. The two datasets respectively include utterances from $1,251$ and $6,112$ speakers and more than 340 and $2,440$ hours of speech. VoxCeleb1 is divided into two subsets: the development set, which involves $1,211$ speakers and the evaluation set, which involves 40 speakers. VoxCeleb2 is also divided into two subsets: the development set, which involves $5,994$ speakers and the evaluation set, which involves 118 speakers.

4.2. Data configurations and metrics

For supervised learning experiments, we employ the development sets of the VoxCeleb1 and 2 datasets. For self-supervised learning experiments, the development set of VoxCeleb2 is used. For fine-tuning the self-supervised pre-trained model, i.e., the semi-supervised scenario, we pre-train the model using the VoxCeleb2 development set and fine-tune the model using the VoxCeleb1 development set. The VoxCeleb1-O benchmark evaluation protocol that involves the VoxCeleb1 test set is used to measure the performance for all experiments.

The widely adopted EER (%) is the primary metric. We also report performances using the minimum detection cost function (minDCF) metric. For the minDCF, hyper-parameters of P_{target}=0.05 and $C_{falsealarm} = C_{miss}$=1 are adopted. Lower values depict superior performances for both metrics.

Table 2: *Results on self-supervised learning using the DINO [25] framework. Trained on VoxCeleb2 development set.*

Configurations	**EER(%)**	**minDCF**
RawNet3	**5.74**	**0.3507**
− param fbank log	10.46	0.5775
− param fbank mean norm	8.87	0.4969
− param fbank log&mean norm	9.98	0.5386
+ DINO temp warm-up	5.89	0.4004
+ DINO last layer norm	**5.40**	**0.3396**
→ DINO T momentum 0.99	6.17	0.3987
→ half batch size (400→200)	6.87	0.4513

Table 3: *Results on fine-tuning the pre-trained model. Trained on VoxCeleb1 development set.*

Configurations	**EER(%)**	**minDCF**
RawNet3 (w/ pre-train)	**2.18**	**0.1519**
RawNet3 (w/o pre-train)	2.98	0.2268

4.3. Implementation details

Common. All experiments have been conducted using the PyTorch Python library with four Nvidia V100 GPUs. The model architecture, training recipe and pre-trained model weights will be freely accessible.[1] Unless mentioned otherwise, the parameterised filterbank layer has a kernel length of 251, stride size of 48, and 256 filters. AFMS-Res2MP blocks have 1024 filters, and other hyper-parameters, including kernel length, pool size, and dilation, follow that of Figure 1. The Adam optimiser [43] with SGDR learning rate scheduling [44] is adopted. Weight decay regularisation of $5e^{-5}$ is applied to the model.

Supervised learning. The learning rate is scheduled between $1e^{-3}$ and $5e^{-6}$ and restarts every eight epochs. The model is trained for 40 epochs. AAM-softmax has a margin m of 0.3 and scale s of 30. The model is trained using randomly cropped 3 seconds utterances. The size of a mini-batch is 512. We apply batch augmentation, similar to that of [45] where augmentation methods include waveform masking, speed, noise and reverberation additions.

Self-supervised learning. The learning rate is scheduled between $1e^{-3}$ and $1e^{-5}$ and restarts every 16 epochs. The model is trained for 80 epochs. The size of a mini-batch is 400 unless mentioned otherwise. We adopt two global views of 4 seconds and five local views of 2 seconds for the DINO framework. Augmentation involves noise and reverberations with additional data curriculum augmentation [46]. Temperatures for the teacher and the student RawNet3 models are 0.04 and 0.1, respectively. Momentum values for the teacher model and centre update are 0.987 and 0.9, respectively.

[1] Benchmark reproduction and inference are available in https://github.com/Jungjee/RawNet. Training recipe is available in https://github.com/clovaai/voxceleb_trainer.

Table 4: *Comparison with recent literature of supervised speaker verification. †: calculated with $P_{target} = 0.01$.*

	In Feat	EER(%)	minDCF
Desplanques et al. [2]	MFCC	0.87	0.1066†
Ravanelli et al. [17]	Fbank	0.69	N/R
Kuzmin et al. [18]	Fbank	**0.66**	**0.0640**†
Zhu et al. [12]	Waveform	2.60	0.2390
Li et al. [14]	Waveform	2.31	N/R
Lin et al. [15]	Waveform	1.95	0.2030
Kim et al. [16]	Waveform	1.29	0.1420
Ours−stride=10	Waveform	**0.89**	0.0659
Ours−stride=16	Waveform	0.90	**0.0593**

5. Results

5.1. Supervised learning

Table 1 reports the result of RawNet3 architecture under supervised learning with ground truth label-based classification. Compared to the previous RawNet2 architecture, RawNet3 demonstrates superior performance with EER reduced from 2.48% to 1.05%, showing over 57% improvement relative.

Rows from 3 to 6 report ablation experiments on the RawNet3 architecture, excluding each modified component. Exclusion of either logarithm or mean subtraction normalisation degrades the performance. Channel and context statistical pooling had the most impact as a single component, degrading the EER to 1.45% when excluded.

The last five rows demonstrate a trade-off between computation and performance via different stride sizes in the parameterised filterbank layer. A smaller stride size generates a longer sequence of frames, bringing further performance improvement at the cost of more computation. On the other hand, a larger stride size generates a shorter sequence of frames, resulting in less computation at the cost of performance degradation. With a stride size of 10, the EER is reduced to 0.89%.

5.2. Self-supervised learning

Table 2 delivers the result of the proposed RawNet3 architecture under the DINO self-supervised learning framework. The result in the first row shows an EER of 5.74% without utilising any ground truth label. Rows from 2 to 4 show the results of excluding logarithm, mean normalisation or both after the first parameterised filterbank layer. Compared to supervised learning, logarithm and mean normalisation affected the performance significantly. We analyse that this phenomenon is related to the property of the DINO framework, which does not involve negative pairs. Because comparison is only made among the same utterances with different augmentations, robustness towards channel variation can be weaker, which the logarithm and mean normalisation offer.

Rows 5 and 6 show ablations by testing the officially recommended hyper-parameter changes.[2] Both hyper-parameters tended to act differently than anticipated. Applying warm-up did not affect the performance noticeably, and normalising the last layer (no normalisation is expected to perform better) further improved the EER to 5.40%. The last two rows present additional ablation experiments by changing the teacher network's momentum value and halving the batch size.

Table 5: *Comparison with self-supervised learning models.*

	Framework	EER(%)	minDCF
Huh et al. [27]	AP+AAT	8.65	0.4540
Xia et al. [28]	MOCO+Wav-Aug(ProtoNCE)	8.23	0.5900
Mun et al. [29]	CEL	8.01	N/R
Tao et al. [30]	Contrastive	7.36	N/R
Sang et al. [31]	SSReg	6.99	0.4340
Ours	DINO	**5.40**	**0.3396**

5.3. Semi-supervised learning

Table 3 delivers the result of RawNet3 under the semi-supervised scenario where we assume that the ground truth label only exists for a smaller VoxCeleb1 development set. By solely initialising the RawNet3 model with DINO pre-trained weight parameters, 25% relative performance improvement was observed. EER of 2.18% was obtained with pre-training, whereas EER was 2.98% without the DINO pre-training.

5.4. Comparison with recent literature

Supervised learning. Table 4 compares the RawNet3 trained with ground truth labels with recent handcrafted feature- and raw waveform-based models [2, 12, 14–18]. Compared to the recent state-of-the-art models, our RawNet3 shows a competitive performance of EER 0.89%, whereas EER of the best model [18] is 0.66%. In terms of the minDCF, RawNet3 has a value of 0.0593; however, the values are incomparable due to the difference in P_{target}. Our RawNet3 outperforms the previous best model among raw waveform speaker verification models by 31% relative (1.29% vs 0.89%).

Self-supervised learning. Table 5 compares the proposed RawNet3 trained using the DINO self-supervised framework with other recent works in the literature. All five mentioned works [27–31] can be seen as contrastive learning variants. The DINO framework, which does not adopt negative pairs, demonstrated the best performance.

Relation with iterative clustering. Iterative clustering has been proven to be effective for self-supervised speaker verification models [30, 47]. Our model can be viewed as the initial model of iterative clustering; thus, the model can enjoy the benefits of iterative clustering methods. The RawNet3 model trained using the DINO framework is hence complementary to the existing iterative clustering approaches.

6. Conclusions

This paper proposes RawNet3, a new raw waveform speaker verification model and evaluates it under three scenarios. In supervised learning, the model demonstrates an EER of 0.89%, competitive with state-of-the-art handcrafted feature-based counterparts.Using the DINO self-supervision framework, the model also demonstrates an EER of 5.40%, outperforming existing contrastive learning-based works. Utilising the DINO-trained model as pre-training, fine-tuning with a smaller dataset is also proven to be effective, showing 25% relative improvement to the model trained with random initialisation.

[2] https://github.com/facebookresearch/dino

7. References

[1] D. Snyder, D. Garcia-Romero, G. Sell et al., "X-vectors: Robust dnn embeddings for speaker recognition," in *Proc. ICASSP*, 2018.

[2] B. Desplanques, J. Thienpondt and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *Proc. Interspeech*, 2020.

[3] J. S. Chung, A. Nagrani and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *Proc. Interspeech*, 2018, pp. 1086–1090.

[4] W. Cai, J. Chen, J. Zhang and M. Li, "On-the-fly data loader and utterance-level aggregation for speaker and language recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1038–1051, 2020.

[5] C. H. You, K. A. Lee and H. Li, "An svm kernel with gmm-supervector based on the bhattacharyya distance for speaker recognition," *IEEE Signal processing letters*, vol. 16, no. 1, pp. 49–52, 2008.

[6] N. Dehak, P. J. Kenny, R. Dehak et al., "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010.

[7] A. Baevski, Y. Zhou, A. Mohamed and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. NeurIPS*, 2020, vol. 33, pp. 12449–12460.

[8] T. Sainath, R. J. Weiss, K. Wilson et al., "Learning the speech front-end with raw waveform cldnns," in *Proc. Interspeech*, 2015.

[9] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," in *Proc. SLT*, 2018, pp. 1021–1028.

[10] M. Pariente, S. Cornell, A. Deleforge and E. Vincent, "Filterbank design for end-to-end speech separation," in *Proc. ICASSP*, 2020.

[11] J.-w. Jung, H.-S. Heo, I.-H. Yang et al., "D-vector based speaker verification system using raw waveform cnn," in *Proc. ANIT 2017*, 2017, pp. 126–131.

[12] G. Zhu, F. Jiang and Z. DuanZhu, "Y-vector: Multiscale waveform encoder for speaker embedding," in *Proc. Interspeech*, 2021.

[13] J.-w. Jung, H.-S. Heo, J.-h. Kim et al., "Rawnet: Advanced end-to-end deep neural network using raw waveforms for text-independent speaker verification," in *Proc. Interspeech*, 2019.

[14] J. Li, N. Yan and L. Wang, "Fdn: Finite difference network with hierarchical convolutional features for text-independent speaker verification," *arXiv preprint:2108.07974*, 2021.

[15] W.-W. Lin and M.-W. Mak, "Wav2spk: A simple dnn architecture for learning speaker embeddings from waveforms.," in *Proc. Interspeech*, 2020, pp. 3211–3215.

[16] J.-h. Kim, H.-j. Shim, J. Heo and H.-J. Yu, "Rawnext: Speaker verification system for variable-duration utterances with deep layer aggregation and extended dynamic scaling policies," in *Proc. ICASSP*, 2022.

[17] M. Ravanelli, T. Parcollet, P. Plantinga et al., "Speechbrain: A general-purpose speech toolkit," *arXiv preprint:2106.04624*, 2021.

[18] N. Kuzmin, I. Fedorov and A. Sholokhov, "Magnitude-aware probabilistic speaker embeddings," *arXiv preprint:2202.13826*, 2022.

[19] X. Zhai, A. Oliver, A. Kolesnikov and L. Beyer, "S4l: Self-supervised semi-supervised learning," in *Proc. ICCV*, 2019.

[20] T. Chen, S. Kornblith, M. Norouzi and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. ICML*, 2020, pp. 1597–1607.

[21] K. He, H. Fan, Y. Wu et al., "Momentum contrast for unsupervised visual representation learning," in *Proc. CVPR*, 2020.

[22] K. He, X. Chen, S. Xie et al., "Masked autoencoders are scalable vision learners," *arXiv preprint:2111.06377*, 2021.

[23] H. Bao, L. Dong and F. Wei, "Beit: Bert pre-training of image transformers," *arXiv preprint:2106.08254*, 2021.

[24] J.-B. Grill, F. Strub, F. Altché et al., "Bootstrap your own latent-a new approach to self-supervised learning," in *Proc. NeurIPS*, 2020.

[25] M. Caron, H. Touvron, I. Misra et al., "Emerging properties in self-supervised vision transformers," in *Proc. ICCV*, 2021.

[26] A. Baevski, W.-N. Hsu, Q. Xu et al., "Data2vec: A general framework for self-supervised learning in speech, vision and language," *arXiv preprint:2202.03555*, 2022.

[27] J. Huh, H. S. Heo, J. Kang et al., "Augmentation adversarial training for unsupervised speaker recognition," in *NeurIPS workshop on Self-Supervised Learning for Speech and Audio Processing*, 2020.

[28] W. Xia, C. Zhang, C. Weng et al., "Self-supervised text-independent speaker verification using prototypical momentum contrastive learning," in *Proc. ICASSP*, 2021, pp. 6723–6727.

[29] S. H. Mun, W. H. Kang, M. H. Han and N. S. Kim, "Unsupervised representation learning for speaker recognition via contrastive equilibrium learning," *arXiv preprint:2010.11433*, 2020.

[30] R. Tao, K. A. Lee, R. K. Das et al., "Self-supervised speaker recognition with loss-gated learning," *arXiv preprint:2110.03869*, 2021.

[31] M. Sang, H. Li, F. Liu et al., "Self-supervised speaker verification with simple siamese network and self-supervised regularization," *arXiv preprint:2112.04459*, 2021.

[32] J.-w. Jung, S.-b. Kim, H.-j. Shim et al., "Improved rawnet with filter-wise rescaling for text-independent speaker verification using raw waveforms," in *Proc. Interspeech*, 2020, pp. 1496–1500.

[33] J.-w. Jung, H.-S. Heo, I.-H. Yang et al., "A complete end-to-end speaker verification system using deep neural networks: From raw signals to verification result," in *Proc. ICASSP*, 2018.

[34] D. Ulyanov, A. Vedaldi and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint:1607.08022*, 2016.

[35] J.-w. Jung, H.-S. Heo, I.-h. Yang et al., "Avoiding speaker overfitting in end-to-end dnns using raw waveform for text-independent speaker verification," *Proc. Interspeech*, pp. 3583–3587, 2018.

[36] S.-H. Gao, M.-M. Cheng, K. Zhao et al., "Res2net: A new multi-scale backbone architecture," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 2, pp. 652–662, 2019.

[37] J. Hu, L. Shen and G. Sun, "Squeeze-and-excitation networks," in *Proc. CVPR*, 2018, pp. 7132–7141.

[38] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. ICML*, 2015, pp. 448–456.

[39] J. Chung, C. Gulcehre, K. Cho and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *NIPS Workshop on Deep Learning*, 2014.

[40] E. Variani, X. Lei, E. McDermott et al., "Deep neural networks for small footprint text-dependent speaker verification," in *Proc. ICASSP*, 2014, pp. 4052–4056.

[41] J. Deng, J. Guo, N. Xue and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proc. CVPR*, 2019, pp. 4690–4699.

[42] A. Nagrani, J. S. Chung and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," in *Proc. Interspeech*, 2017.

[43] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.

[44] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," in *Proc. ICLR*, 2017.

[45] E. Hoffer, T. Ben-Nun, I. Hubara et al., "Augment your batch: Improving generalization through instance repetition," in *Proc. CVPR*, 2020, pp. 8129–8138.

[46] H.-S. Heo, J.-w. Jung, J. Kang et al., "Self-supervised curriculum learning for speaker verification," *arXiv preprint:2203.14525*, 2022.

[47] D. Cai, W. Wang and M. Li, "An iterative framework for self-supervised deep speaker representation learning," in *Proc. ICASSP*, 2021.