Incorporating Visual Information in Audio Based Self-Supervised Speaker Recognition

Danwei Cai, *Student Member, IEEE* Weiqing Wang, and Ming Li, *Senior Member, IEEE*

***Abstract*—The current success of deep learning largely benefits from the availability of large amount of labeled data. However, collecting a large-scale dataset with human annotation can be expensive and sometimes difficult. Self-supervised learning thus attracts many research interests to train models without labels. In this paper, we propose a self-supervised learning framework for speaker recognition. Combining clustering with deep representation learning, the proposed framework generates pseudo labels for the unlabeled dataset and learns speaker representation without human annotation. Our method starts with training a speaker representation encoder with contrastive self-supervised learning. Clustering on the learned representation generates pseudo labels, which are used as the supervisory signal for the subsequent training of the representation encoder. The clustering and representation learning process is performed iteratively to bootstrap the discriminative power of the deep neural network. We apply this self-supervised learning framework to both single modal audio data and multi-modal audio-visual data. For audio-visual data, audio and visual representation encoders are employed to learn representations of the corresponding modality. A cluster ensemble algorithm is then used to fuse the clustering results of the two modalities. The complementary information in multi-modalities ensures a robust and fault-tolerant supervisory signal for audio and visual representation learning. Experimental results show that our proposed iterative self-supervised learning framework outperforms previous works with self-supervision by large margins. Training with single modal audio data on the development set of VoxCeleb 2, our proposed framework achieves an equal error rate (EER) of 2.8% on the original test trials of VoxCeleb 1. When training with additional visual modality, the EER further reduces to 1.8%, which is only 20% higher than the fully supervised audio-based system with an EER of 1.5%. Also, experimental analysis shows that the proposed framework generates pseudo labels that are highly correlated to ground truth labels.**

***Index Terms*—Self-supervised learning, self-labeling, clustering, speaker recognition, audio-visual data**

I. Introduction

REPRESENTATION learning is to extract useful information of perceptual data such as audio, image, or video when building classifiers or other predictors [1]. Over the past decade, deep learning has facilitated representation learning by training deep neural networks (DNN) with massive labeled data. For example, in speaker recognition, a DNN is trained to map audio data to a discriminative feature space by classifying speakers in training data. Under this deep learning setup, a large-scale dataset is required to obtain great model generalizability and discriminative representation space. However, manually annotating labels for a large-scale dataset is expensive and sometimes difficult. Learning from unlabeled data can significantly reduce the cost of developing machine learning models for new applications. Self-supervised learning thus emerges as an increasingly popular framework to train models without labels.

Self-supervised learning aims to design pretext or proxy tasks for DNNs to learn model parameters without annotated data [2]. For example, in natural language processing, a proxy task can be predicting the next or a randomly masked element of a sequence [3]. In visual representation learning, most proxy tasks fall into two classes: generative or discriminative. Generative approaches learn to generate or model pixels in the image space. Examples include colorizing images [4], solving jigsaw puzzles [5], predicting the patch context [6], predicting rotations [7], inpainting patches [8], and so on. Recently, discriminative approaches based on contrastive self-supervised learning (CSL) emerge and show promising results in visual representation learning [9, 10, 2]. It performs instance discrimination for the unlabeled data and learns the representation using metric learning-based objectives similar to supervised learning methods [11]. Typically, self-supervised learning with pretext tasks is used as a pre-training method for other downstream tasks.

As an alternative, self-labeling using self-supervised learning aims at learning a deep neural network together with discovering the data labels [12, 13]. Self-labeling algorithms can be viewed as a combination of feature learning with clustering. Starting from the randomly initialized feature representation, the clustering algorithm derives pseudo labels as the supervisory signal iteratively and updates network parameters. Since self-labeling methods rely on the initial feature representations of the network, they are sensitive to the initialization of network parameters [12].

In our previous work on self-supervised speaker representation learning [14, 15], we proposed a two-stage iterative labeling framework. In the first stage, contrastive self-supervised learning (CSL) pre-trains the speaker embedding network. CSL allows the network to learn a meaningful representation for the first clustering round instead of random initialization. The second stage is an iterative process of clustering and representation learning. A clustering algorithm generates pseudo labels of the training data with the learned speaker representation, and the network is trained with these labels in a supervised manner. The clustering algorithm can discover the intrinsic structure of the representation of the unlabeled

D. Cai, W. Wang, and M. Li are with the Department of Electrical and Computer Engineering, Duke University, Durham, NC, 27705, USA, e-mail: {danwei.cai, weiqing.wang, ming.li369}@duke.edu.

M. Li is also with Data Science Research Center at Duke Kunshan University, Kunshan, China.

Corresponding author: Ming Li.

data, providing meaningful supervisory signals compared to contrastive learning, which draws negative samples uniformly from the training data without label information. The idea behind the proposed framework is to take advantage of the DNN's ability to learn from data with label noise and bootstrap its discriminative power. Different from other works on self-labeling using self-supervised learning [12, 13], our method decouples clustering and representation learning: representation is trained until converged before clustering.

While the evaluation phase of speaker recognition only allows audio data, multi-modal audio-visual data can be used for training. The usage of the additional modality could be beneficial for representation learning since different modalities contain complementary information. In this work, we extend the iterative labeling framework to multi-modal audio-visual data. Specifically, a visual encoder is added to learn face representations from the visual modality. Clustering on the representations of the multi-modal data gives pseudo labels from the audio and the visual modality. We employ a cluster ensemble algorithm to fuse the pseudo labels from different modalities. This fused pseudo label is then used to train both audio and visual encoders. With the clustering ensemble algorithm, information in one modality can flow to the other, and confirmation bias in self-training within a single modality is avoided.

We evaluate our proposed framework on the Voxceleb [16, 17] dataset. Experimental results show that the proposed self-supervised framework outperforms prior works by large margins in both single and multiple modality settings. Also, our method is capable of labeling the video data with high accuracy.

The key contributions of this work are summarized as follows:

1) We develop an iterative framework for self-supervised speaker representation learning using single modal audio data. The framework generates pseudo labels that are highly correlated to ground truth labels.
2) We extend the proposed self-supervised learning-based speaker recognition framework to audio-visual training data. The multi-modal information helps to generate more meaningful pseudo labels compared to a single modality.
3) The proposed framework greatly shrinks the performance gap between self-supervised and fully supervised speaker recognition. Our method obtains the best speaker recognition performance among published literature under a self-supervised setting to the best of our knowledge.

II. Related Works

A. Deep speaker recognition

Automatic speaker recognition analyzes a given speech and recognizes the speaker's identity using signal processing and pattern recognition algorithms. Over the past few years, deep learning methods have greatly improved the performance of speaker recognition systems [18, 19, 20]. In general, speaker recognition, especially speaker verification, is evaluated under the open-set setting, where speakers in the testing set are different from those in the training set. Therefore, the goal of a speaker network is to learn discriminative representations from the training speakers. It is common to train a speaker classification network and extract the speaker representation from the output of the intermediate layer.

Typically, a deep speaker framework consists of three parts: a frame-level local pattern extractor, an utterance-level encoding layer, and a fully connected layer. A local pattern extractor learns speaker representation from the spectral feature sequence of varying lengths, producing a frame-level representation. Common network architectures for local pattern extractor lie in two categories: time-delayed neural network (TDNN) [18, 21] and convolutional neural network (CNN) [19, 22]. TDNN is actually 1-dimensional CNN with a 1-dimensional kernel whose receptive field covers the whole frequency axis in one time frame of the time-spectral feature map. The encoding layer encodes the frame-level sequence into an utterance-level representation. The most common encoding method is the average pooling layer [18], which aggregates the mean or (and) standard deviation statistics from the frame-level representation. Other encoding layers include attentive pooling layer [23, 21], learnable dictionary encoding layer [24], and dictionary-based NetVLAD layer [25, 26]. After that, fully connected layers take the utterance-level representation as an input to further abstract the speaker information and classify the training speakers.

B. Audio-visual speaker recognition

Because of the complementary nature of audio and visual data, audio-visual biometrics methods have drawn attention in the research community. Several fusion strategies have been proposed at different stages of model training, i.e., early, middle and late fusion. Early fusion takes audio features and visual images as joint input and generates joint representation containing multi-modal information [27]. Middle fusion combines audio and visual representations after the independent encoding of the two modalities [27, 28, 29, 30]. Late fusion performs score fusion of uni-modal systems from different modalities [31].

Audio-visual learning based on fusion strategies generally requires multi-modal data at both the training and evaluation phases. As an alternative, the cross-modal method allows greater flexibility for unimodal testing. The basic idea of cross-modal method is to map data from multi-modalities into a shared latent feature space and achieve cross-modal retrieval [32, 33, 34, 35].

C. Self-supervised speaker representation learning

Proxy tasks for self-supervised speaker representation learning fall into generative and discriminative methods. Stafylakis *et al.* [36] propose to learn speaker representation via reconstructing the acoustic features of a target speech, given the decoded phone sequence and the inferred speaker representation of another speech segment from the same utterance. Although speaker label is not required, this method employs a phone decoder trained with supervision and is not strictly self-supervised.

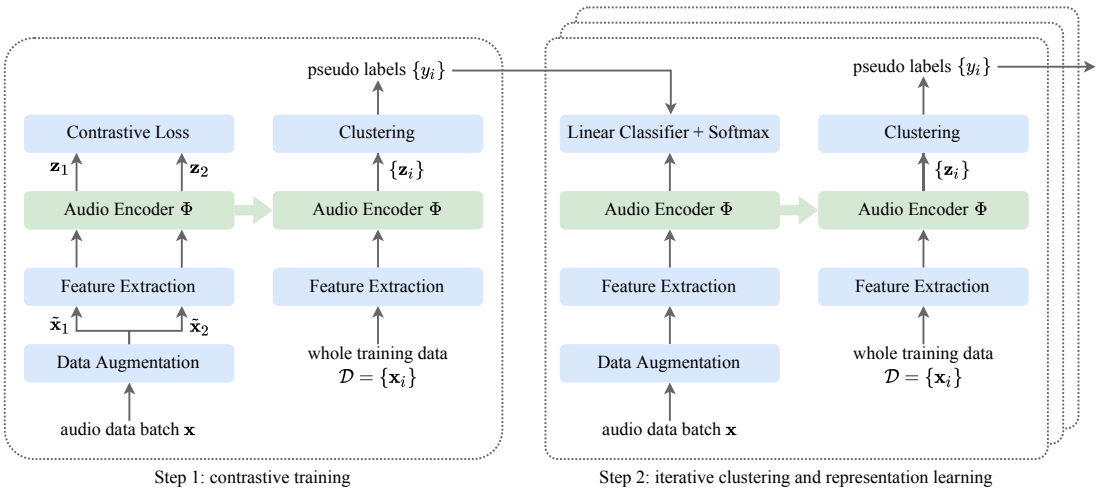


Fig. 1: The proposed iterative framework for self-supervised speaker representation learning.

Discriminative approaches based on contrastive learning have recently shown promising results [37, 38]. To learn speaker representations that are invariant to channel variabilities, multiple methods such as augmentation adversarial training [39], equilibrium learning [40], and channel-invariant training [41] are proposed.

In this work, we take a step further beyond contractive self-supervised learning. A clustering algorithm is used to discover the intrinsic structure of the representation of the unlabeled data and generate pseudo labels to learn the representation encoder discriminatively.

D. Self-supervised multi-modal representation learning

Audio-visual self-supervised representation learning utilizes the video data's multi-modal information, i.e., images and sound, to learn representations. Given the audio and visual streams, semantic correspondence [42, 43, 44, 45, 46] and synchronized timing of content [47, 48] are commonly used as a supervisory signal in the existing literature. Other works perform within-clip sound localization [49, 50] or audio-separation [51] using this multi-modal information. Also, the cross-modal semantic correspondence has been leveraged in the application of speaker/person representation learning[52, 53].

This paper focuses on using single modal audio data or multi-modal audio-visual data to learn speaker representation under a self-supervised setting. Also, the testing data only contains audio signals. Our goal is to obtain pseudo labels for an unlabeled dataset using the multi-modal information and bootstrap the discriminative power of the representation encoders for both audio and visual modalities. Since the multi-modal data is only used to discover the supervisory signal for model training, multi-modal data is not necessarily required at the testing phase.

E. Self-supervised track of VoxSRC

The VoxCeleb Speaker Recognition Challenge (VoxSRC) has been held since 2019 annually to: (i) promote new research in speaker recognition; (ii) evaluate the current state of the art through public evaluations; and (iii) provide open-source data that can be used by the research community [54, 55, 56]. In 2020, VoxSRC developed the new track of self-supervised speaker verification. The challenge dataset [16, 17] is a multi-modal dataset, both audio and visual modalities are allowed for system development in the self-supervision track.

The proposed two-stage iterative labeling framework in this paper was submitted in VoxSRC 2020 (single modality system) [14] and VoxSRC 2021 (multi-modality system) [57]. A similar iterative framework based on single modality of audio data was also developed by Thienpondt *et al.* in VoxSRC 2020 [58]. In VoxSRC 2021, participants developed their systems based on the two-stage framework with single modality of audio data. Two improvements from two different teams are highlighted here:

1) In [59], a non-contrastive self-supervised method, distillation with no labels (DINO), is used as the initial model. This new method is shown to outperform the previous contrastive learning method.
2) In [60], two parallel branches of neural network are trained with different data augmentation setups. The pseudo labels generated by different branches are exchanged for the next round of training.

III. Iterative Labeling Framework for Self-Supervised Speaker Representation Learning

This section describes the proposed iterative labeling framework for self-supervised speaker representation learning. We illustrate the proposed framework in figure 1.

- Stage 1: contrastive training
  - Train an audio encoding network with contrastive self-supervised learning.
  - With this encoding network, extract representations for the whole training data. Perform a clustering algorithm on these representations to generate pseudo labels.
- Stage 2: iterative clustering and representation learning

- Train a new encoding network with a classification layer and cross-entropy loss using the generated pseudo labels.
- With the new encoding network, extract representations and perform clustering to generate new pseudo labels.
- Repeat stage 2 with limited rounds.

A. Contrastive self-supervised learning

We employ the contrastive self-supervised learning (CSL) framework similar to the framework in [2, 61] to learn an initial audio representation. Let $\mathcal{D} = \{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$ be an unlabeled dataset with N data samples, CSL assumes that each data sample defines its own class and perform instance discrimination. During training, we randomly sample a mini-batch $\mathcal{B} = \{\mathbf{x}_1, \cdots, \mathbf{x}_M\}$ of M data samples from $\mathcal{D}$. For data point $\mathbf{x}_i$, two different audio segments are randomly cropped from the original audio before data augmentation. Then stochastic data augmentation is performed to generate two correlated views, i.e., $\tilde{\mathbf{x}}_{i,1}$ and $\tilde{\mathbf{x}}_{i,2}$, resulting $2M$ data points in total for a mini-batch. $\tilde{\mathbf{x}}_{i,1}$ and $\tilde{\mathbf{x}}_{i,2}$ are considered as a positive pair and other $2(M-1)$ data points $\{\tilde{\mathbf{x}}_{j,k} | j \neq i, k = 1, 2\}$ are negative examples for $\tilde{\mathbf{x}}_{i,1}$ and $\tilde{\mathbf{x}}_{i,2}$.

During training, a neural network encoder Φ extracts representations for the $2M$ augmented data samples,

$$\mathbf{z}_{i,j} = \Phi(\tilde{\mathbf{x}}_{i,k}), \ k \in \{1, 2\} \tag{1}$$

After that, contrastive loss identifies the positive example $\tilde{\mathbf{x}}_{i,1}$ (or $\tilde{\mathbf{x}}_{i,2}$) among the negative examples $\{\tilde{\mathbf{x}}_{j,k} | j \neq i, k = 1, 2\}$ for $\tilde{\mathbf{x}}_{i,2}$ (or $\tilde{\mathbf{x}}_{i,1}$). We adapt the contrastive loss from SimCLR [2] as:

$$\mathcal{L}_{\text{CSL}} = \frac{1}{2M} \sum_{i=1}^{M} (l_{i,1} + l_{i,2}) \tag{2}$$

$$l_{i,j} = -\log \frac{\exp(\cos(\mathbf{z}_{i,1}, \mathbf{z}_{i,2})/\tau)}{\sum_{k=1}^{M} \sum_{l=1}^{2} \mathbb{1}_{\substack{k \neq i \\ l \neq j}} \exp(\cos(\mathbf{z}_{i,j}, \mathbf{z}_{k,l})/\tau)} \tag{3}$$

where $\mathbb{1}$ is an indicator function evaluating 1 when $k \neq i$ and $l \neq j$, cos denotes the cosine similarity and τ is a temperature parameter to scale the similarity scores. $l_{i,j}$ can be interpreted as the loss for anchor feature $\mathbf{z}_{i,j}$. It computes positive score for positive feature $\mathbf{z}_{i,(j+1)\text{mod}2}$ and negative scores across all $2(M-1)$ negative features $\{\mathbf{z}_{k,j} | k \neq i, j = 1, 2\}$.

The encoder encodes the audio segments at the utterance level; thus, the learned representation may contain variability factors of semantic content, speaker identity, channel, and language all together. Since the contrastive loss performs instance discrimination for two segments in an utterance, the variability factor of semantic content, which varies in different segments, may be reduced. Other unwanted variabilities such as channel and language, which remain unchanged to varying segments within an utterance, may still be preserved in the learned representation.

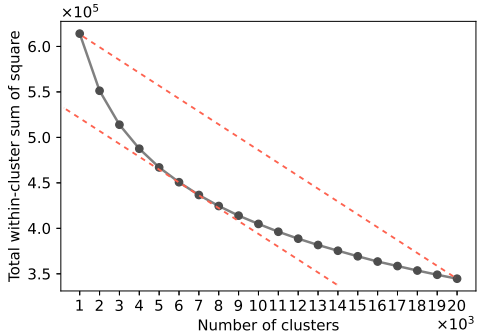


Fig. 2: Within-cluster sum of square W of a clustering procedure versus the number of clusters K employed.

B. Generating pseudo labels by clustering

1) K-means clustering: Given the learned representations of the training data, we employ a clustering algorithm to generate cluster assignments and pseudo labels. In this paper, we use the well-known k-means algorithm because of its simplicity, fast speed, and capability with a large dataset.

Let the learnt representation in d-dimensional feature space $\mathbf{z} \in \mathbb{R}^d$, k-means learns a centroid matrix $\mathbf{C} \in \mathbb{R}^{d \times K}$ and the cluster assignment $y_i \in \{1, \cdots, K\}$ for representation $\mathbf{z}_i$ with the following learning objective

$$\min_{\mathbf{C}} \frac{1}{N} \sum_{i=1}^{N} \min_{y_i} \|\mathbf{z}_i - \mathbf{C}_{y_i}\|_2^2 \tag{4}$$

where $\mathbf{C}_{y_i}$ is the y_i^{th} column of the centroid matrix $\mathbf{C}$. The optimal assignments $\{y_1, \cdots, y_N\}$ are used as pseudo labels.

In contrastive self-supervised learning, negative samples are drawn uniformly from the training data without label information, which brings false negative samples into training. However, with the clustering algorithm, the intrinsic structure of the unlabeled data is mined, providing a meaningful supervisory signal to train the representation encoder discriminatively.

2) Determine the number of clusters: As mentioned before, the learned representations may contain variability factors of channel and language in addition to the variability of speaker identity. The choice of the number of clusters may affect the content of the resulting clusters. For example, clustering with two classes may result in male and female classes; clustering with 100 classes may result in classes with different languages (suppose that the dataset contains about 100 languages).

To determine the optimal number of clusters, we employ the simple 'elbow' method [62]. It calculates the total within-cluster sum of squares W for the clustering outputs with different numbers of clusters K:

$$W = \sum_{i=1}^{N} \|\mathbf{z}_i - \mathbf{C}_{y_i}\|_2^2 \tag{5}$$

The total within-cluster sum of squares W curve is plotted according to a sequence of K in ascending order. Figure 2 shows an example of such a curve. W decreases as K increases and the decrease of W flattens from some K onwards, forming an 'elbow' of the curve. Such 'elbow' indicates that additional

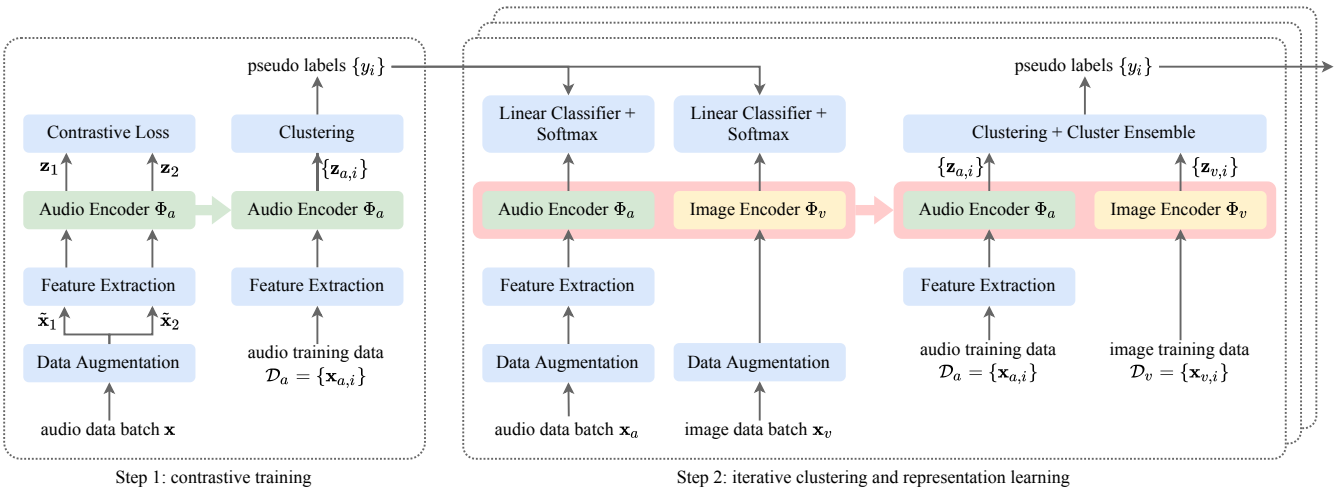


Fig. 3: The proposed iterative framework for self-supervised speaker representation learning using multi-modal data.

clusters beyond such K contribute little intra-cluster variation; thus, the K at the 'elbow' indicates the appropriate number of clusters.

To find such 'elbow', we draw an auxiliary line connecting the first and the last points of the W-K curve. The auxiliary line and the W-K curve together form a closed shape. This closed shape can be abstracted as a triangle. The 'elbow' of the W-K curve corresponds to the lower-left vertex of the triangle. In this case, we draw a tangent line of the W-K curve parallel to the auxiliary line. The contact on the W-K curve has the longest distance from the auxiliary line and can be considered an 'elbow'. In figure 2, the number of clusters can choose between 5,000 and 7,000.

This 'elbow' method is not exact, and the optimal number of clusters can be subjective [63]. Still, it provides a meaningful way to help to determine the optimal number of clusters and is successfully used in different applications [64, 65]. More analysis of this method can be found in [66, 67].

C. Learning with pseudo labels

With the generated pseudo labels $\{y_1, \cdots, y_N\}$ for training data $\mathcal{D} = \{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$, the neural network encoder Φ can be discriminatively trained with a parametrized classifier $g_W(\cdot)$ which predicts the labels for the representation vector $\mathbf{z}_i = \Phi(\mathbf{x}_i)$. The parameters $\{\Phi, W\}$ are jointly trained with the cross-entropy loss:

$$\mathcal{L}_{\text{classifier}} = -\sum_{i=1}^{N}\sum_{k=1}^{K} \log\left(p(k|\mathbf{x}_i)q(k|\mathbf{x}_i)\right) \tag{6}$$

$$p(k|\mathbf{x}_i) = \frac{\exp(g_{W\,k}(\mathbf{z}_i))}{\sum_{j=1}^{K}\exp(g_{W\,j}(\mathbf{z}_i))} \tag{7}$$

where $q(k|\mathbf{x}_i) = \delta_{k,y_i}$ is the ground-truth distribution over labels for data sample $\mathbf{x}_i$ with label y_i, δ_{k,y_i} a Dirac delta which equals to 1 for $k = y_i$ and 0 otherwise, $g_{W\,j}(\mathbf{z}_i)$ is the j^{th} element ($j \in \{1, \cdots, K\}$) of the class score vector $g_W(\mathbf{z}_i) \in \mathbb{R}^K$, K is the number of the pseudo classes.

D. Dealing with label noise: label smoothing regularization

One problem with the generated pseudo labels is label noise which degrades the generalization performance of deep neural networks. We apply label smoothing to deal with label noise to mitigate this problem.

Label smoothing is a regularization method to estimate the marginalized effect of label noise during training. It prevents a DNN from assigning full probability to the training samples with noisy label [68, 69]. Specifically, for a training example $\mathbf{x}$ with label y, label smoothing regularization replaces the label distribution $q(k|\mathbf{x}) = \delta_{k,y}$ in equation (6) with

$$q'(k|\mathbf{x}) = (1-\epsilon)\delta_{k,y} + \frac{\epsilon}{K} \tag{8}$$

where ϵ is a smoothing parameter.

IV. Incorporating Visual Information in Self-Supervised Speaker Representation Learning

Given a dataset of multi-modal data with audio- and visual-modality, the representation of each modality is learned independently following the method in section III. Clustering is performed on the representations of each modality to generate pseudo-labels for both audio and visual data. Cluster ensemble is then used to fuse pseudo-labels generated by different modalities. The proposed framework is illustrated in figure 3.

- Stage 1: contrastive training
  - Train an audio encoding network using contrastive self-supervised learning.
  - With this encoding network, extract representations for the audio data. Perform a clustering algorithm on these audio representations to generate pseudo labels.
- Stage 2: iterative clustering and representation learning
  - With the generated pseudo labels, train audio, and visual encoding networks independently in a supervised manner.
  - With the audio encoding network, extract audio representations and perform clustering to generate pseudo audio labels.

- With the visual encoding network, extract visual representations and perform clustering to generate pseudo visual labels.
- Fuse the audio and visual pseudo labels using a cluster ensemble algorithm.
- Repeat stage 2 with limited rounds.

A. Representation learning

Given a multi-modal dataset $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ with audio-modality $\mathcal{D}_a = \{\mathbf{x}_{a,1}, \mathbf{x}_{a,2}, \cdots, \mathbf{x}_{a,N}\}$ and visual-modality $\mathcal{D}_v = \{\mathbf{x}_{v,1}, \mathbf{x}_{v,2}, \cdots, \mathbf{x}_{v,N}\}$, encoders for each modality is trained independent with the generated pseudo labels from last training round. For each video sample $\mathbf{x}_i$, its audio part $\mathbf{x}_{a,i}$ and visual part $\mathbf{x}_{v,i}$ share the same generated pseudo label.

The audio encoder Φ_a is discriminatively trained with the audio classifier $g_{W_a}(\cdot)$ using cross-entropy loss in equation (6). The audio representation is extracted as

$$\mathbf{z}_a = \Phi_a(\mathbf{x}_a) \tag{9}$$

Same procedure is applied to the visual encoder Φ_v and the visual classifier $g_{W_v}(\cdot)$. The visual representation is extracted as

$$\mathbf{z}_v = \Phi_v(\mathbf{x}_v) \tag{10}$$

B. Clustering

Clustering is applied independently on both audio representations $\{\mathbf{z}_{a,i}|i = 1, \cdots, N\}$ and visual representations $\{\mathbf{z}_{v,i}|i = 1, \cdots, N\}$. Audio and visual pseudo labels ($\{y_{a,i}|i = 1, \cdots, N\}$ and $\{y_{v,i}|i = 1, \cdots, N\}$) are thus obtained for further aggregation.

Considering that the audio and the visual representations contain complementary information from different modalities, we apply an additional clustering on the joint representations to generate more robust pseudo labels. Given the audio representation $\mathbf{z}_a$ and the visual representation $\mathbf{z}_v$, the joint representation is formed as

$$\mathbf{z}_\jmath = (\mathbf{z}_a, \mathbf{z}_v) \tag{11}$$

Joint pseudo labels $\{y_{\jmath,i}|i = 1, \cdots, N\}$ is then generated by clustering on joint representations.

C. Cluster ensemble

We use simple voting strategy [70, 71] to fuse the three clustering outputs, i,e., $\{y_{a,i}\}$, $\{y_{v,i}\}$ and $\{y_{\jmath,i}\}$. Since the cluster labels in different clustering outputs are arbitrary, cluster correspondence should be established among different clustering outputs. This starts with a contingency matrix $\Omega \in \mathbb{R}^{K\times K}$ for the referenced clustering output $\{y_{\text{ref},i}\}$ and the current clustering output $\{y_{\text{cur},i}\}$, where K is the number of clusters. Each entry $\Omega_{l,l'}$ represents the co-occurence between cluster l of the referenced clustering output and cluster l' of the current clustering output,

$$\begin{aligned} \Omega_{l,l'} &= \sum_{i=1}^{N} \omega(i) \\ \omega(i) &= \begin{cases} 1 & y_{\text{ref},i} = l, y_{\text{cur},i} = l' \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{12}$$

Cluster correspondence is solved by the following optimization problem,

$$\max_{\Theta} \sum_{l=1}^{K} \sum_{l'=1}^{K} \Omega_{l,l'} \Theta_{l,l'} \tag{13}$$

where $\Theta \in \mathbb{R}^{K\times K}$ is the correspondence matrix for the two clustering outputs. $\Theta_{l,l'}$ equals to 1 if cluster l in the reference clustering output corresponds to cluster l' in the current clustering output, 0 otherwise. This optimization can be solved by the Hungarian algorithm [72].

We select the joint pseudo labels as the reference clustering output and calculate cluster correspondence for the audio and visual pseudo labels. A globally consistent label set is obtained after the re-labeling process. Majority voting is then employed to determine a pseudo consensus label for each data sample in the multi-modal dataset.

V. Experimental Setups

A. Dataset

The experiments are conducted on VoxCeleb, which is an audio-visual dataset consisting of short video clips extracted from interview videos [16, 17].

For model training, the development set of VoxCeleb 2 is used. The original development set contains 1,092,009 audio files from 5,994 speakers. The corresponding video files from the official VoxCeleb dataset are with a quantity of 1,091,724. The final audio-visual dataset used for training is the intersection of these two parts. We extract face images at one frame per second (fps) from cropped video files. Speaker or face identity labels are not used for model training and are used for experimental analysis purposes only.

For evaluation, the development and test sets of Voxceleb 1 are used. We report the speaker verification results on three trial lists as defined in [17]:

- VoxCeleb 1-O: the original trial list of Voxceleb 1 containing 37,720 trials from 40 speakers.
- Voxceleb 1-E: an extended trial list containing 581,480 trials from 1251 speakers.
- Voxceleb 1-H: a hard trial list containing 552,536 trials from 1190 speakers; all test pairs are within the same language and gender.

Face verification results are also reported using the trial lists described above to test the learned face representation. The cropped face images extracted at one fps are downloaded from the VoxCeleb website[1].

B. Data Augmentation

1) Data augmentation for audio data: Data augmentation is effective for deep speaker representation learning under the settings of supervised learning [73] and contrastive self-supervised learning [37, 39, 2]. We use additive background noise or convolutional reverberation noise for the time-domain waveform. MUSAN dataset [74] is used as the data augmentation dataset. Addictive noises include ambient noise, music, and babble noise. The babble noise is constructed by mixing

[1]Available at https://www.robots.ox.ac.uk/∼vgg/research/CMBiometrics

TABLE I: The network architecture for audio encoder, $\mathbf{C}$(kernal size, stride) denotes the convolutional layer, $[\cdot]$ denotes the residual block; L relates to the duration of the speech and L relates to the number of frequency bins of the Mel spectrogram.

Layer	Output Size	Structure
Input	$1 \times F \times L$	-
Conv1	$16 \times F \times L$	$\mathbf{C}(3 \times 3, 1)$
Residual layer 1	$16 \times F \times L$	$\begin{bmatrix}\mathbf{C}(3 \times 3, 1)\\ \mathbf{C}(3 \times 3, 1)\end{bmatrix} \times 3$
Residual layer 2	$32 \times \frac{F}{2} \times \frac{L}{2}$	$\begin{bmatrix}\mathbf{C}(3 \times 3, 2)\\ \mathbf{C}(3 \times 3, 1)\end{bmatrix} \begin{bmatrix}\mathbf{C}(3 \times 3, 1)\\ \mathbf{C}(3 \times 3, 1)\end{bmatrix} \times 3$
Residual layer 3	$64 \times \frac{F}{4} \times \frac{L}{4}$	$\begin{bmatrix}\mathbf{C}(3 \times 3, 2)\\ \mathbf{C}(3 \times 3, 1)\end{bmatrix} \begin{bmatrix}\mathbf{C}(3 \times 3, 1)\\ \mathbf{C}(3 \times 3, 1)\end{bmatrix} \times 5$
Residual layer 4	$128 \times \frac{F}{8} \times \frac{L}{8}$	$\begin{bmatrix}\mathbf{C}(3 \times 3, 2)\\ \mathbf{C}(3 \times 3, 1)\end{bmatrix} \begin{bmatrix}\mathbf{C}(3 \times 3, 1)\\ \mathbf{C}(3 \times 3, 1)\end{bmatrix} \times 2$
Pooling layer	256	Global statistics pooling
Embedding	128	Fully connected layer

three to eight speech files into one. The signal-to-noise ratios (SNR) are randomly set between 5 to 20 dB. For the reverberation noise, the convolution operation is performed with 40,000 simulated room impulse responses (RIR) in MUSAN. We only use RIRs from small and medium rooms. Data augmentation is performed on the fly at a probability of 0.6 during training.

In contrastive self-supervised learning, we apply a more aggressive data augmentation strategy. In addition to applying a single noise type, we also apply additive and convolutional noise simultaneously for a signal training utterance.

2) Data augmentation for visual data: We sequentially apply these simple augmentations for cropped face images: random cropping followed by resizing to 3×224×224, random horizontal flipping, random color distortions, random grey scaling, and random Gaussian blur. The data augmentation is performed at a probability of 0.6 during training. We normalize each image's pixel value to the range of $[-0.5, 0.5]$ afterward.

C. Audio encoder trained with contrastive self-supervision

We apply contrastive self-supervised learning on audio data to learn speaker representations.

During DNN training, audio waveforms in a data batch are randomly cropped between 2 to 4 seconds. Logarithmical Mel-spectrogram is extracted for each audio signal as an acoustic feature. 40 Mel filters are applied on the spectrogram computed over Hamming windows of 20ms shifted by 10ms to generate the Mel-spectrogram.

While the proposed framework allows various choices of the network architecture, we opt for a residual convolutional network (ResNet) for speaker representation learning [19, 75]. The ResNet takes spectral features as input and produces feature maps at the frame level. A global statistics pooling layer then calculates means and standard deviations for the output feature maps to generate an utterance-level representation. A fully connected layer is employed afterward to extract the 128-dimensional speaker representation. The detailed configuration of the speaker embedding network can be found in table I. Rectified linear unit (ReLU) activation and batch normalization are applied to each convolutional layer in ResNet. Adam optimizer [76] is used to update network parameters with a batch size of 256. The learning rate is initially set to 0.001 and is decreased by 5% every five epochs. The hyper-parameter τ in equation (3) is set to 0.1.

At the evaluation stage, cosine similarity is used to generate a verification score for a test trial.

D. Audio encoder trained with pseudo supervision

The experimental setup for the audio encoder trained with pseudo labels are the same as described in the last section except for the following changes:

- Logarithmical Mel-spectrogram with 80 frequency bins is used as input features.
- The ResNet doubles the feature map channels to increase its modeling ability under the setting of discriminative training.
- A linear layer is used to classify the pseudo speakers using cross-entropy loss. Dropout is added before the classification layer to prevent overfitting [77].
- Network parameters are updated using stochastic gradient descent (SGD) algorithm.
- The learning rate is initially set to 0.1 and is divided by 10 whenever the training loss reaches a plateau.

E. Visual encoder setup

We choose the standard ResNet-34 [75] as the visual encoder. To produce a 128-dimensional representation, a fully connected layer is added between the pooling layer and the final liner layer which classifies face identities. Dropout is applied before the linear classification layer to prevent overfitting.

During training, the network takes the extracted images with the shape of 3×224×244 as inputs. Network parameters are updated with the SGD algorithm with an initial learning rate of 0.1. We divide the learning rate by 10 whenever the training loss reaches a plateau.

During the face based evaluation phase, we average the face representations of the image frames from the same video segment to get a video-level face representation. Cosine similarity is used as the scoring function.

F. Evaluation metric

1) Verification evaluation: For speaker verification and face verification, we report two performance metrics: (1) Equal error rate (EER): the error rate when false acceptance rate and false rejection rate are equal; (2) Minimum detection cost (minDCF): the minimum value of the detection cost function which is defined as a weighted sum of false-reject and false-alarm error rates for some decision threshold [78]. The parameters of the detection cost function are set as: $C_{\text{Miss}} = 1$, $C_{\text{FA}} = 1$, $P_{\text{Target}} = 0.05$.

TABLE II: Verification performance of the proposed self-supervised learning framework on VoxCeleb 1 test trials.

Model	Modality	VoxCeleb 1-O		VoxCeleb 1-E		VoxCeleb 1-H	
		minDCF	EER[%]	minDCF	EER[%]	minDCF	EER[%]
Supervised	Audio	0.097	1.51	0.102	1.59	0.178	3.00
	Visual	0.083	1.46	0.063	1.24	0.092	1.71
Nagrani *et al.* [52]	Audio (train and test); Visual (train)	-	22.09	-	-	-	-
Chung *et al.* [53]	Audio (train and test); Visual (train)	-	17.52	-	-	-	-
Inoue *et al.* [37]	Audio	-	15.26	-	-	-	-
Xia *et al.* [38]	Audio	-	8.23	-	-	-	-
Huh *et al.* [39]	Audio	0.454	8.65	-	-	-	-
Mun *et al.* [40]	Audio	-	8.01	-	-	-	-
Zhang *et al.* [41]	Audio	-	8.28	-	-	-	-
Initial round - CSL	Audio	0.508	8.86	0.570	10.15	0.710	16.20
Training with single modal audio data							
Round 1	Audio	0.257	3.64	0.299	4.11	0.459	7.68
Round 2	Audio	0.214	2.99	0.234	3.41	0.362	6.25
Round 3	Audio	0.190	2.93	0.214	3.23	0.334	5.85
Round 4	Audio	0.184	2.85	0.202	3.16	0.314	5.54
Round 5	Audio	0.173	2.74	0.201	3.08	0.311	5.48
Training with multi-modal audio-visual data							
Round 1	Audio	0.257	3.64	0.299	4.11	0.459	7.68
	Visual	0.345	5.55	0.319	5.15	0.432	8.04
Round 2	Audio	0.146	2.05	0.159	2.36	0.254	4.23
	Visual	0.153	2.27	0.126	1.85	0.170	2.79
Round 3	Audio	0.141	1.93	0.138	2.09	0.231	3.88
	Visual	0.136	1.77	0.108	1.63	0.152	2.48
Round 4	Audio	0.139	1.81	0.139	2.06	0.224	3.80
	Visual	0.178	1.96	0.108	1.61	0.152	2.50
Round 5	Audio	0.142	1.92	0.136	2.03	0.222	3.72
	Visual	0.147	2.19	0.105	1.78	0.145	2.57

2) Clustering evaluation: To evaluate the clustering quality, we adopt three metrics following [46]: the normalized mutual information, the clustering accuracy, and the mean maximal purity per cluster.

Given the ground-truth clustering assignment U and the predictive clustering assignment V, the normalized mutual information (NMI) measures the information shared between U and V and is defined as:

$$\mathrm{NMI}(U,V) = \frac{2 \times I(U;V)}{H(U) + H(V)} \tag{14}$$

where $I(U;V)$ denotes the mutual information between U and V, and $H(\cdot)$ denotes entropy. The NMI ranges from 0 to 1. With two largely independent clustering assignments, NMI becomes 0. When they are insignificant agreement, NMI equals 1.

The clustering accuracy is measured by matching the pseudo labels V to the ground truth labels U. Hungarian algorithm [72] is used to establish label correspondence between U and V.

To measure the semantic purity of each pseudo cluster comparing to the ground truth labels, we report the mean maximal purity per cluster,

$$\mathrm{purity} = \frac{1}{K} \sum_{k \in K} \max\left(p\left(y|\hat{y}=k\right)\right) \tag{15}$$

where K is the number of pseudo clusters, $\hat{y}$ represents a pseudo cluster and $p\left(y|\hat{y}=k\right)$ is the distribution of ground-truth clusters under pseudo cluster k. This metric ranges from $\frac{1}{K}$, which corresponds to a random clustering assignment, to perfect matching at 1.

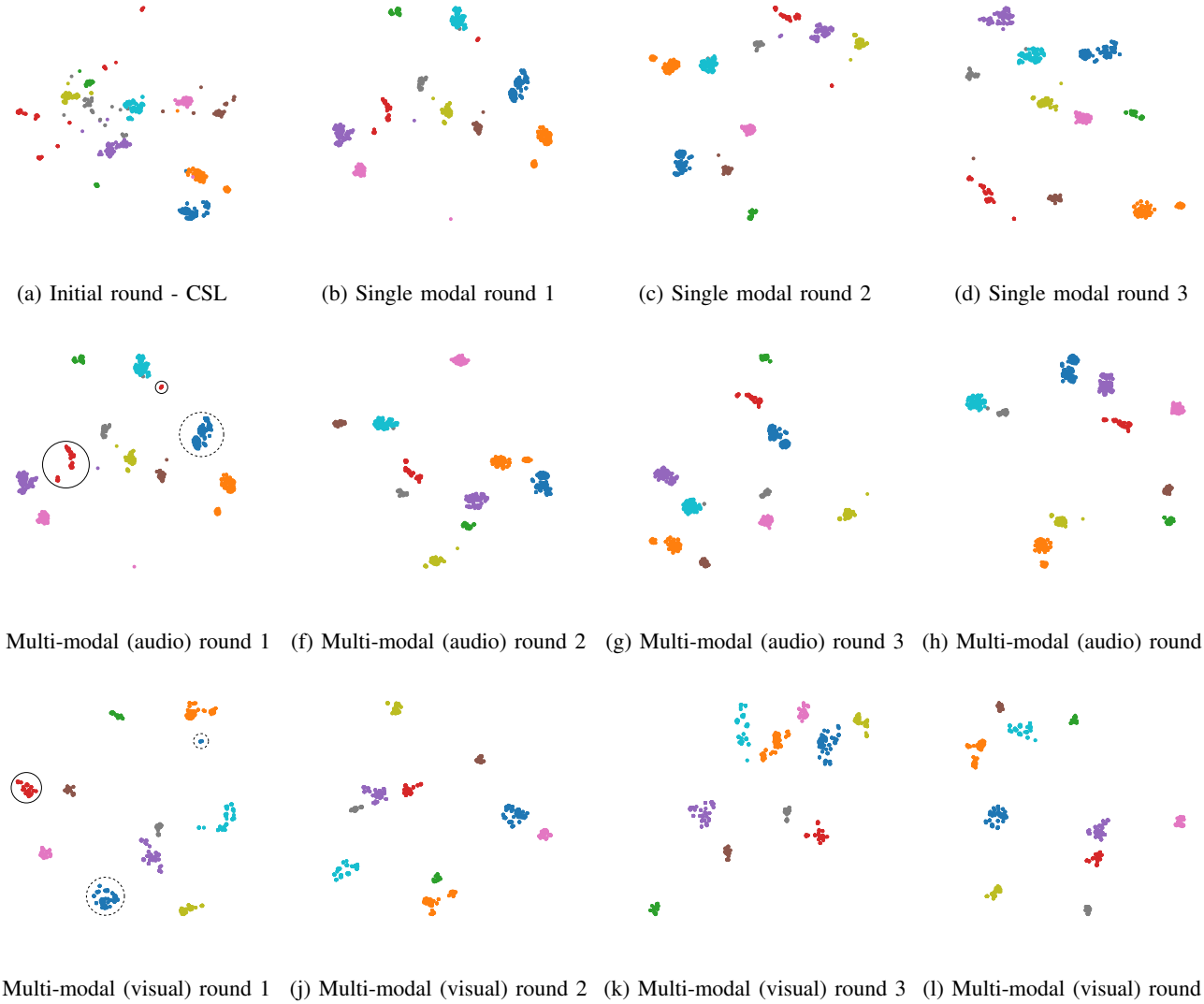
VI. Experimental Analysis

The experiments are performed in five parts. In section VI-A, we report the speaker verification performance of the proposed self-supervised learning framework using both single modal and multi-modal datasets. In section VI-B, labeling qualities are reported. In section VI-C, we discuss how the choice of the number of clusters affects the learned representations. Section VI-D shows that robust training can further improve the performance with our generated pseudo labels. Finally, section VI-E demonstrates the power of self-supervised pre-training on the small-scale labeled dataset.

A. Speaker verification performance

Table II reports the verification results of our proposed framework using both single modal and multi-modal training data. The number of clusters of the *k*-means algorithm is set to 6,000. A detailed analysis of choosing the number of clusters is provided in section VI-C. The performance of the audio speaker verification system and visual face verification system trained with full supervision is also provided in table II for reference.

When training with single modal audio data, one round of clustering and representation learning obtains EER reductions of more than 50% relatively for all the trial lists compared to



(a) Initial round - CSL (b) Single modal round 1 (c) Single modal round 2 (d) Single modal round 3

(e) Multi-modal (audio) round 1 (f) Multi-modal (audio) round 2 (g) Multi-modal (audio) round 3 (h) Multi-modal (audio) round 4

(i) Multi-modal (visual) round 1 (j) Multi-modal (visual) round 2 (k) Multi-modal (visual) round 3 (l) Multi-modal (visual) round 4

Fig. 4: t-SNE visualization of learned representations extracted from the VoxCeleb 1 dataset. Each color corresponds to a different speaker.

the initial round of contrastive self-supervised learning. The performance of the speaker verification system improves with the increase of round number, which shows the effectiveness of the proposed iterative self-supervision framework on speaker representation learning. Three rounds of training provide EERs of 2.93%, 3.23%, and 5.85% on trials of VoxCeleb 1-O, VoxCeleb 1-E, VoxCeleb 1-H, respectively.

Also, we see a trend of performance saturation for both audio modality and audio-visual modalities. With single modal audio data, the relative EER reduction percentages of the current and previous rounds are 53%, 19%, 6% and 5% on VoxCeleb 1-H trials for the first four rounds. With multi-modal audio-visual data, the relative EER reduction percentages are 53%, 45%, 8% and 2% for the first four rounds. The speaker representation network trained with pseudo labels generated from multi-modal data has a larger performance gain thanks to incorporating the complementary information of audio and visual modality.

Although the performance of the audio modality-based system still improves at round 5, the relative performance gain between two concessive rounds shrinks. We can thus safely conclude that the performance of the audio-based system converges. Five rounds of training on audio modality achieves an EER of 2.74% on trials of VoxCeleb 1-O, while two rounds of training on audio-visual modalities achieves an EER of 2.05%. We believe it's worth jumping into the audio-visual modality for maximum efficiency under limited resources.

More training rounds may still improve the verification performance for the audio modality. However, more training rounds require more computational resources due to the iterative nature of the proposed method. We chose to stop the training iteration at round 5 to save computational resources based on the verification performance of the development set.

Figure 4 visualizes the learned representations by using the t-distributed stochastic neighbor embedding (t-SNE) algorithm [79]. Ten random speakers from Voxceleb 1 dataset are

TABLE III: Unsupervised labelling of VoxCeleb 2 development set with different number of clusters K.

Number of clusters		**1,000**			**6,000**			**20,000**		
Model	**Modality**	**NMI**	**Accuracy**	**Purity**	**NMI**	**Accuracy**	**Purity**	**NMI**	**Accuracy**	**Purity**
Supervised	Audio	0.8264	36.42%	46.02%	0.9607	77.20%	90.34%	0.9237	44.87%	96.74%
	Visual	0.8470	36.77%	46.30%	0.9642	77.87%	91.32%	0.9250	46.92%	97.23%
CSL	Audio	0.6189	19.84%	21.58%	0.7586	38.49%	51.45%	0.8114	27.68%	68.21%
Training with single modal audio data										
Round 1	Audio	0.7315	30.15%	34.59%	0.9007	66.04%	78.62%	0.9005	40.51%	90.34%
Round 2	Audio	0.7413	29.82%	33.55%	0.9121	67.22%	81.03%	0.8978	39.71%	89.83%
Round 3	Audio	0.7443	29.82%	33.81%	0.9190	68.64%	82.27%	0.8937	38.69%	88.68%
Round 4	Audio	-	-	-	0.9209	69.07%	82.80%	-	-	-
Round 5	Audio	-	-	-	**0.9230**	**68.93%**	**83.50%**	-	-	-
Training with multi-modal audio-visual data										
Round 1	Audio	0.7315	30.15%	34.59%	0.9007	66.04%	78.62%	0.9005	40.51%	90.34%
	Visual	0.7531	31.21%	42.55%	0.9107	67.94%	79.33%	0.8979	40.95%	88.61%
	Fused	0.7993	32.63%	43.16%	0.9531	77.11%	89.11%	0.9233	47.26%	96.33%
Round 2	Audio	0.7478	31.23%	37.22%	0.9445	73.24%	87.66%	0.9099	42.02%	93.11%
	Visual	0.8121	32.63%	42.50%	0.9502	73.17%	89.04%	0.9069	41.88%	91.72%
	Fused	0.8251	32.59%	44.32%	0.9608	77.34%	90.53%	0.9241	46.90%	96.91%
Round 3	Audio	0.7711	31.81%	38.00%	0.9519	74.68%	88.91%	0.9102	42.06%	93.12%
	Visual	0.8082	32.46%	43.33%	0.9546	73.67%	89.47%	0.9135	42.94%	93.84%
	Fused	0.8311	32.52%	46.00%	**0.9627**	**77.60%**	**90.62%**	0.9249	47.13%	97.20%
Round 4	Audio	-	-	-	0.9523	74.38%	89.32%	-	-	-
	Visual	-	-	-	0.9559	73.23%	89.85%	-	-	-
	Fused	-	-	-	0.9624	76.66%	90.61%	-	-	-
Round 5	Audio	-	-	-	0.9514	74.08%	89.21%	-	-	-
	Visual	-	-	-	0.9571	73.54%	90.17%	-	-	-
	Fused	-	-	-	0.9624	76.91%	90.85%	-	-	-

selected for visualization. The speaker representations learned with the initial round of CSL (figure 4a) are not discriminative enough as speaker subspaces overlap. We can observe that the proposed iterative learning framework keeps optimizing the within-class variance and between-class variance of the learned speaker representations. By comparing the learned speaker representations from the third round, we observe that the representations trained with multi-modal data (figure 4g) are more discriminative than those trained with single modal data (figure 4d). Also, by comparing the feature spaces of speaker representations (figure 4e) and face representations (figure 4i), we observe that the one poorly learned speaker subspace in one modality can be discriminative in the other modality (see the red speaker and the blue speaker marked in figure 4e and 4i). This indicates the complementary property of the audio and visual modality, which allows the label ensemble algorithm to generate noise-tolerant pseudo labels from different modalities.

B. Unsupervised labeling of unlabeled dataset

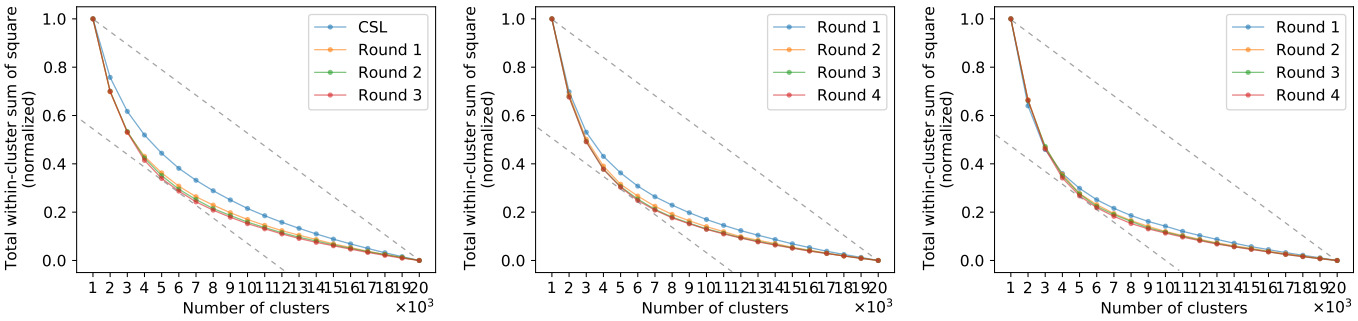
Table III shows the quality of the labels obtained by the proposed framework with both single modal and multi-modal training data. We run *k*-means clustering on both audio representations and visual representations trained with supervision and calculate metrics of clustering quality as upper bounds for reference.

We observe that the iterative training helps to obtain better clustering quality in most experimental settings. Also, the cluster ensemble of audio and visual modality greatly improves all the clustering quality metrics under different settings of the number of clusters. Compared with the single modal system, the multi-modal training system improves the clustering quality by a large margin. With multi-modal training data, our proposed framework obtains a high labeling accuracy of 77.60%, almost the same as the labeling accuracy of the representations learned with supervision.

C. Choosing the number of clusters

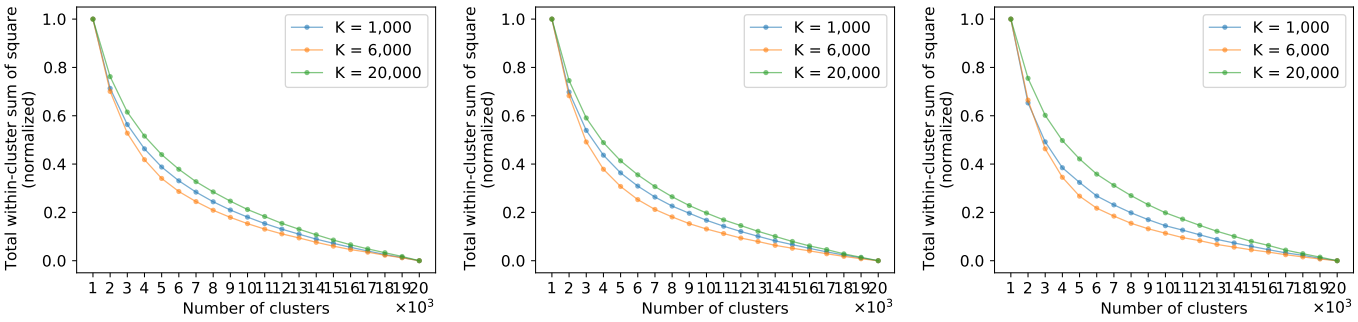
The 'elbow' method introduced in section III-B2 is used to determined the number of clusters for *k*-means algorithm. To determine the number of clusters K for the calculation of the total within-cluster sum of square W, we first attempt to guess the reasonable maximum and minimum of the average cluster size. Given the training data of the VoxCeleb 2 development set with 1,092,009 data samples, the maximum and minimum of the average cluster size could be 1,000 and 50, which lead to minimal and maximal K around 1,000 and 20,000. The W-K curve is then plotted with a step size of 1,000 for K. Figure 2 shows the W-K curve for the representations trained with CSL. Auxiliary line and tangent line as introduced in section III-B2 to help find the 'elbow' point are drawn in the figure. By observing the 'elbow' of the W-K curve, an appropriate number of clusters can choose from 5,000 to 7,000. Figure 5 shows W-K curves of each training round of our proposed framework when the number of clusters is set to 6,000. For better comparison, we normalize the value of W in the W-K curve between 0 and 1. All the W-K curves show a consistent 'elbow' around 5,000 to 7,000.

To understand how the choice of the number of clusters affects the verification performance and labeling quality of



(a) Single modal training (b) Multi-modal training (audio) (c) Multi-modal training (visual)

Fig. 5: Within-cluster sum of square W of a clustering procedure versus the number of clusters K employed. For each training round, the number of clusters is set to 6,000.



(a) Single modal training (b) Multi-modal training (audio) (c) Multi-modal training (visual)

Fig. 6: Within-cluster sum of square W of a clustering procedure versus the number of clusters K employed. Each curve corresponds to the last training round of our learning framework with a particular K.

our proposed framework, we train another two systems with a small number of clusters of 1,000 and a large number of clusters of 20,000. Figure 6 shows W-K curves from the last training round of our learning framework with different K. We observe a relatively consistent 'elbow' around 6,000 from all W-K curves, which indicates that the 'elbow' method can estimate a correct number of clusters even when the system is trained with extremely small or large K.

Table III reports the clustering quality when the proposed framework is trained with a number of clusters of 1,000 and 20,000. We observe a better mean maximal purity per cluster with larger K. Larger K leads to smaller cluster sizes and more semantic-related data samples within a cluster. Although clustering with a number of clusters ($K = 6,000$) that is close to the actual number ($K = 5,994$) gives the best NMI and accuracy, clustering with a small ($K = 1,000$) or large ($K = 20,000$) number of clusters is also able to generate meaningful pseudo labels with reasonable NMIs and accuracies.

Figure 7 presents the verification performance for all training rounds of the proposed framework with different settings of the number of clusters. Training with a number of clusters closed to the correct one gives the best verification performance for both speaker and face verification. Also, training with a small or large number of clusters can obtain performance gain over the CSL system. However, the speaker representation network is more likely to overfit with pseudo labels generated with extreme settings of the number of clusters. We also see that systems with a large K (20,000) outperform those with the estimated K (6,000) at the first training round, which indicates that high cluster purity is essential for the first training round.

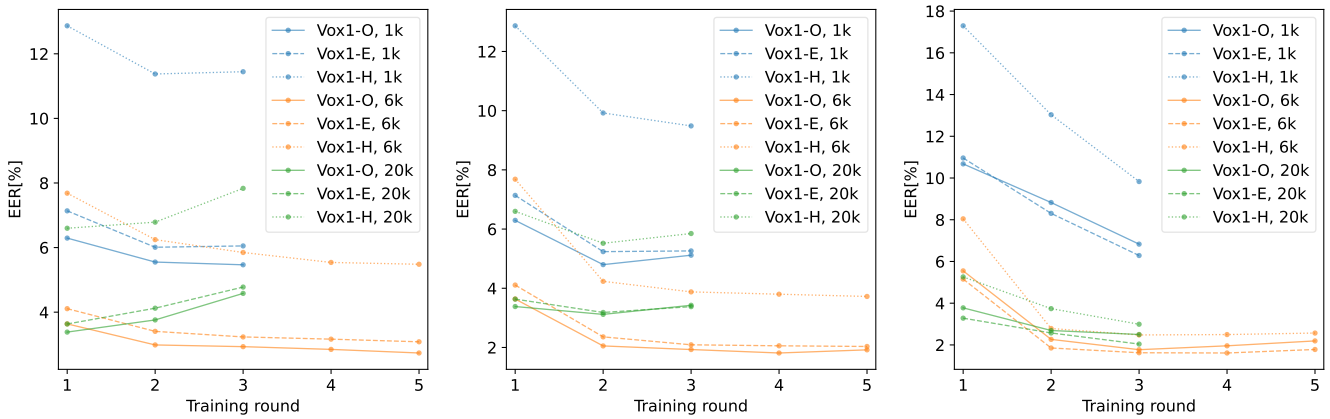
TABLE IV: Self-supervised speaker verification performance (EER %) of robust training on the final pseudo labels.

Model	VoxCeleb 1-O		VoxCeleb 1-E		VoxCeleb 1-H	
	minDCF	EER	minDCF	EER	minDCF	EER
Supervised	0.097	1.51	0.102	1.59	0.178	3.00
Single modal trained	0.190	2.93	0.214	3.23	0.334	5.85
with label smoothing	0.173	2.70	0.201	3.07	0.318	5.42
with SE + AAM	0.179	2.65	0.185	2.89	0.297	5.01
Multi-modal trained	0.139	1.81	0.139	2.06	0.224	3.80
with label smoothing	0.125	1.70	0.125	1.87	0.211	3.44
with SE + AAM	0.120	1.62	0.122	1.87	0.199	3.37

D. Robust training

In this section, we show that speaker verification performance can further improve with the generated pseudo labels. Two robust training methods are applied separately to achieve this goal:

1) Label smoothing regularization is used to deal with label noise of the generated pseudo labels as described in



(a) Single modal training (b) Multi-modal training (audio) (c) Multi-modal training (visual)

Fig. 7: EER of each training round of the proposed framework trained with single modal data or multi-modal data. For each setting of training data, we train the framework with different numbers of clusters, i.e., 1,000, 6,000, 20,000.

TABLE V: Fine-tune the self-supervised model with labeled data of different speakers.

Fine-tuning data	**None**		**10 speakers**		**100 speakers**		**600 speakers**		**All speakers**	
Pre-trained model	minDCF	EER[%]	minDCF	EER[%]	minDCF	EER[%]	minDCF	EER[%]	minDCF	EER[%]
None	-	-	0.998	25.82	0.581	9.63	0.298	4.09	0.191	2.70
CSL	0.508	8.86	0.489	8.24	0.400	6.14	0.332	4.52	0.294	4.20
Single modal trained	0.190	2.93	0.199	2.90	0.171	2.53	0.151	2.06	0.148	1.91
Multi-modal trained	0.139	1.81	0.144	1.89	0.150	1.92	0.115	1.63	0.116	1.69

section III-D. The hyper-parameter ϵ is set to 0.1 here.

2) Squeeze-Excitation (SE) module [80] is added to improve ResNet. Additive angular margin (AAM) loss [81] is used to learn discriminative representations: the re-scaling factor s is set to 32 and angular margin m is set to 0.2.

Table IV shows experimental results of the above robust training strategies. Compared with the last round results of the proposed framework, label smoothing regularization obtains a relatively 5% gain in terms of EER for all testing trials; SE module with AAM loss obtains a 10% relative gain.

E. Fine-tuning

In this section, the self-supervised learning model is fine-tuned with small-scale labeled datasets. We use the development set of VoxCeleb 1 [16] with 1,211 speakers for fine-tuning. To test the performance of self-supervised pre-training for the smaller dataset, we construct three datasets with 10, 100, 600 speakers randomly selected from VoxCeleb 1 dev set. Results are reported on VoxCeleb 1-O test trials.

We use three pre-trained models, i.e., the model trained with CSL, the model from the last round of the proposed framework trained with single modal data, and the speaker representation model from the last round of the proposed framework trained with multi-modal data. The model trained with CSL uses a ResNet with half feature map channel and 40-dimensional input features. During fine-tuning, the final fully connected layer is replaced to classify the speakers in the small-scale dataset. We firstly freeze the speaker representation encoder and solely train this classification layer until convergence. The remaining training epochs optimize the parameters of the representation encoder and the classification layer simultaneously.

Table V shows the verification results. Fine-tuning with 100 speakers on the model pre-rained with single modal data achieves better performance than the fully supervised model trained with 1,211 speakers. Also, the self-supervised model trained with large-scale multi-modal data outperforms all the fully-supervised models trained with the small-scale dataset. Moreover, fine-tuning can further improve speaker verification performance. With the multi-modal pre-trained model, fine-tuning on VoxCeleb 1 development set obtains a relative EER reduction of 37.4% compared to the counterpart without fine-tuning. We do not see a performance gain when fine-tuning with 10 or 100 speakers. This can be explained by the over-fitting of the small-scale dataset. The pseudo labels generated by our proposed framework are sufficient to train the speaker representation encoder discriminatively, and overfitting hurts the discriminative power of the encoder.

VII. Conclusion

In this paper, we proposed a self-supervised learning framework for speaker recognition. The proposed framework iteratively performs clustering and representation learning, generates pseudo labels for unlabeled training data, and learns speaker representations without human annotation. Several rounds of discriminative training follow the initial training round of contrastive self-supervised learning. Clustering is performed between two rounds of representation learning to generate pseudo labels. The framework exploits DNN's ability to learn from noisy labels and iteratively improves the dis-

criminative power. Considering the complementary property of audio and visual modality, we extend the proposed framework to multi-modal audio-visual data. A visual encoder is added to learn face representations from the visual modality. A cluster ensemble algorithm fuses the pseudo labels from different modalities, avoiding confirmation bias in self-training within a single modality. We evaluate the proposed self-supervised learning framework on the VoxCeleb dataset; experimental results show that our proposed framework outperformed previous works with self-supervision by large margins. With an additional modality of visual data, the proposed framework greatly shrinks the performance gap between self-supervised and fully supervised speaker recognition. Also, experimental analysis shows that the proposed framework generates pseudo labels that are highly correlated to ground truth labels.

REFERENCES

[1] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *TPAMI*, vol. 35, no. 8, pp. 1798–1828, 2013.

[2] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," in *ICML*, 2020, pp. 1597–1607.

[3] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *NAACL-HLT*, 2019, pp. 4171–4186.

[4] R. Zhang, P. Isola, and A. Efros, "Colorful Image Colorization," in *ECCV*, 2016, pp. 649–666.

[5] M. Noroozi and P. Favaro, "Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles," in *ECCV*, 2016, pp. 69–84.

[6] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised Visual Representation Learning by Context Prediction," in *ICCV*, 2015, pp. 1422–1430.

[7] N. Komodakis and S. Gidaris, "Unsupervised Representation Learning by Predicting Image Rotations," in *ICLR*, 2018.

[8] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context Encoders: Feature Learning by Inpainting," in *CVPR*, 2016, pp. 2536–2544.

[9] A. Oord, Y. Li, and O. Vinyals, "Representation Learning with Contrastive Predictive Coding," *arXiv:1807.03748*, 2018.

[10] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum Contrast for Unsupervised Visual Representation Learning," in *CVPR*, 2020, pp. 9729–9738.

[11] A. Kolesnikov, X. Zhai, and L. Beyer, "Revisiting Self-supervised Visual Representation Learning," in *CVPR*, 2019, pp. 1920–1929.

[12] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep Clustering for Unsupervised Learning of Visual Features," in *ECCV*, 2018.

[13] Y. M. Asano, C. Rupprecht, and A. Vedaldi, "Self-Labelling Via Simultaneous Clustering and Representation Learning," in *ICLR*, 2020.

[14] W. Wang, D. Cai, X. Qin, and M. Li, "The DKU-DukeECE Systems for VoxCeleb Speaker Recognition Challenge 2020," *arXiv:2010.12731*, 2020.

[15] D. Cai, W. Wang, and M. Li, "An Iterative Framework for Self-Supervised Deep Speaker Representation Learning," in *ICASSP*, 2021, pp. 6728–6732.

[16] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A Large-Scale Speaker Identification Dataset," in *Interspeech*, 2017, pp. 2616–2620.

[17] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep Speaker Recognition," in *Interspeech*, 2018, pp. 1086–1090.

[18] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "x-vectors: Robust DNN Embeddings for Speaker Recognition," in *ICASSP*, 2018, pp. 5329–5333.

[19] W. Cai, J. Chen, and M. Li, "Exploring the Encoding Layer and Loss Function in End-to-End Speaker and Language Recognition System," in *Speaker Odyssey*, 2018, pp. 74–81.

[20] W. Cai, J. Chen, J. Zhang, and M. Li, "On-the-Fly Data Loader and Utterance-Level Aggregation for Speaker and Language Recognition," *TASLP*, vol. 28, pp. 1038–1051, 2020.

[21] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification," in *Interspeech*, 2020, pp. 3830–3834.

[22] T. Zhou, Y. Zhao, and J. Wu, "ResNeXt and Res2Net Structures for Speaker Verification," in *SLT*, 2021, pp. 301–307.

[23] G. Bhattacharya, J. Alam, and P. Kenny, "Deep Speaker Embeddings for Short-Duration Speaker Verification," in *Interspeech*, 2017, pp. 1517–1521.

[24] W. Cai, Z. Cai, X. Zhang, X. Wang, and M. Li, "A Novel Learnable Dictionary Encoding Layer for End-to-End Language Identification," in *ICASSP*, 2018, pp. 5189–5193.

[25] J. Chen, W. Cai, D. Cai, Z. Cai, H. Zhong, and M. Li, "End-to-end Language Identification using NetFV and NetVLAD," in *ISCSLP*, 2018.

[26] W. Xie, A. Nagrani, J. S. Chung, and A. Zisserman, "Utterance-level Aggregation For Speaker Recognition In The Wild," in *ICASSP*, 2019, pp. 5791–5795.

[27] Y. Qian, Z. Chen, and S. Wang, "Audio-Visual Deep Neural Network for Robust Person Verification," *TASLP*, vol. 29, pp. 1079–1092, 2021.

[28] S. Shon, T.-H. Oh, and J. Glass, "Noise-tolerant Audio-visual Online Person Verification using an Attention-based Neural Network Fusion," in *ICASSP*, 2018, pp. 3995–3999.

[29] S. Shon and J. Glass, "Multimodal Association for Speaker Verification," in *Interspeech 2020*, 2020, pp. 2247–2251.

[30] L. Sarı, K. Singh, J. Zhou, L. Torresani, N. Singhal, and Y. Saraf, "A Multi-View Approach to Audio-Visual Speaker Verification," in *ICASSP*, 2021, pp. 6194–6198.

[31] G. Sell, K. Duh, D. Snyder, D. Etter, and D. Garcia-Romero, "Audio-Visual Person Recognition in Multimedia Data From the Iarpa Janus Program," in *ICASSP*, 2018, pp. 3031–3035.

[32] A. Nagrani, S. Albanie, and A. Zisserman, "Learnable PINs: Cross-Modal Embeddings for Person Identity," in *ECCV*, vol. 11217, 2018, pp. 73–89.

[33] Y. Wen, M. A. Ismail, W. Liu, B. Raj, and R. Singh, "Disjoint Mapping Network for Cross-Modal Matching of Voices and Faces," in *ICLR*, 2019.

[34] S. Nawaz, M. K. Janjua, I. Gallo, A. Mahmood, and A. Calefati, "Deep Latent Space Learning for Cross-modal Mapping of Audio and Visual Signals," in *DICTA*, 2019, pp. 1–7.

[35] R. Tao, R. K. Das, and H. Li, "Audio-visual Speaker Recognition with a Cross-modal Discriminative Network," in *Interspeech*, 2020, pp. 2242–2246.

[36] T. Stafylakis, J. Rohdin, O. Plchot, P. Mizera, and L. Burget, "Self-Supervised Speaker Embeddings," in *Interspeech*, 2019, pp. 2863–2867.

[37] N. Inoue and K. Goto, "Semi-Supervised Contrastive Learning with Generalized Contrastive Loss and Its Application to Speaker Recognition," in *APSIPA*, 2020, pp. 1641–1646.

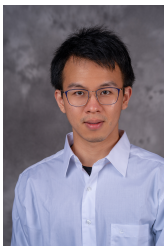
[38] W. Xia, C. Zhang, C. Weng, M. Yu, and D. Yu, "Self-Supervised Text-Independent Speaker Verification Using Prototypical Momentum Contrastive Learning," in *ICASSP*, 2021, pp. 6723–6727.

[39] J. Huh, H. S. Heo, J. Kang, S. Watanabe, and J. S. Chung, "Augmentation Adversarial Training for Unsupervised Speaker Recognition," in *Workshop on Self-Supervised Learning for Speech and Audio Processing, NeurIPS*, 2020.

[40] S. H. Mun, W. H. Kang, M. H. Han, and N. S. Kim, "Unsupervised Representation Learning for Speaker Recognition via

Contrastive Equilibrium Learning," *arXiv:2010.11433*, 2020.
[41] H. Zhang, Y. Zou, and H. Wang, "Contrastive Self-Supervised Learning for Text-Independent Speaker Verification," in *ICASSP*, 2021, pp. 6713–6717.
[42] Y. Aytar, C. Vondrick, and A. Torralba, "Soundnet: Learning Sound Representations from Unlabeled Video," *NeurIPS*, vol. 29, pp. 892–900, 2016.
[43] A. Owens, J. Wu, J. H. McDermott, W. T. Freeman, and A. Torralba, "Ambient Sound Provides Supervision for Visual Learning," in *ECCV*, 2016, pp. 801–816.
[44] M. Patrick, Y. M. Asano, P. Kuznetsova, R. Fong, J. F. Henriques, G. Zweig, and A. Vedaldi, "Multi-modal Self-supervision from Generalized Data Transformations," *arXiv:2003.04298*, 2020.
[45] P. Morgado, N. Vasconcelos, and I. Misra, "Audio-Visual Instance Discrimination with Cross-Modal Agreement," in *CVPR*, 2021, pp. 12 475–12 486.
[46] Y. M. Asano, M. Patrick, C. Rupprecht, and A. Vedaldi, "Labelling Unlabelled Videos from Scratch with Multi-Modal Self-Supervision," in *NIPS*, 2020.
[47] B. Korbar, D. Tran, and L. Torresani, "Cooperative Learning of Audio and Video Models from Self-supervised Synchronization," in *NeurIPS*, 2018, pp. 7774–7785.
[48] A. Owens and A. A. Efros, "Audio-Visual Scene Analysis with Self-Supervised Multisensory Features," in *ECCV*, 2018, pp. 639–658.
[49] D. Hu, F. Nie, and X. Li, "Deep Multimodal Clustering for Unsupervised Audio-Visual Learning," in *CVPR*, 2019, pp. 9248–9257.
[50] A. Senocak, T.-H. Oh, J. Kim, M.-H. Yang, and I. S. Kweon, "Learning to Localize Sound Source in Visual Scenes," in *CVPR*, 2018, pp. 4358–4366.
[51] R. Gao, R. Feris, and K. Grauman, "Learning to Separate Object Sounds by Watching Unlabeled Video," in *ECCV*, 2018, pp. 35–53.
[52] A. Nagrani, J. S. Chung, S. Albanie, and A. Zisserman, "Disentangled Speech Embeddings Using Cross-Modal Self-Supervision," in *ICASSP*, 2020, pp. 6829–6833.
[53] S.-W. Chung, H. G. Kang, and J. S. Chung, "Seeing Voices and Hearing Voices: Learning Discriminative Embeddings Using Cross-Modal Self-Supervision," in *Interspeech*, 2020, pp. 3486–3490.
[54] J. S. Chung, A. Nagrani, E. Coto, W. Xie, M. McLaren, D. A. Reynolds, and A. Zisserman, "VoxSRC 2019: The First VoxCeleb Speaker Recognition Challenge," *arXiv:2201.04583*, 2019.
[55] A. Nagrani, J. S. Chung, J. Huh, A. Brown, E. Coto, W. Xie, M. McLaren, D. A. Reynolds, and A. Zisserman, "VoxSRC 2020: The Second VoxCeleb Speaker Recognition Challenge," *arXiv:2201.04583*, 2020.
[56] A. Brown, J. Huh, J. S. Chung, A. Nagrani, and A. Zisserman, "VoxSRC 2021: The Third VoxCeleb Speaker Recognition Challenge," *arXiv:2201.04583*, 2022.
[57] D. Cai and M. Li, "The DKU-DukeECE System for the Self-Supervision Speaker Verification Task of the 2021 VoxCeleb Speaker Recognition Challenge," *arXiv:2109.02853*, 2021.
[58] J. Thienpondt, B. Desplanques, and K. Demuynck, "The IDLAB VoxCeleb Speaker Recognition Challenge 2020 System Description," *arXiv:2010.12468*, 2020.
[59] J. Cho, J. Villalba, and N. Dehak, "The JHU submission to VoxSRC-21: Track 3," *arXiv:2109.13425*, 2021.
[60] J. Slavíček, A. Swart, M. Klčo, and N. Brümmer, "The Phonexia VoxCeleb Speaker Recognition Challenge 2021 System Description," *arXiv:2109.02052*, 2021.
[61] W. Falcon and K. Cho, "A Framework For Contrastive Self-Supervised Learning And Designing A New Approach," *arXiv:2009.00104*, 2020.
[62] R. L. Thorndike, "Who belongs in the family," *Psychometrika*, 1953.
[63] D. Steinley, "K-Means Clustering: A Half-Century Synthesis," *British Journal of Mathematical and Statistical Psychology*, vol. 59, no. 1, pp. 1–34, 2006.
[64] L. An, S. Yang, and B. Bhanu, "Person Re-Identification by Robust Canonical Correlation Analysis," *IEEE Signal Processing Letters*, vol. 22, no. 8, pp. 1103–1107, 2015.
[65] F. Liu and Y. Deng, "Determine the Number of Unknown Targets in Open World Based on Elbow Method," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 5, pp. 986–995, 2021.
[66] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the Number of Clusters in a Data Set via the Gap Statistic," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 2, pp. 411–423, 2001.
[67] C. Shi, B. Wei, S. Wei, W. Wang, H. Liu, and J. Liu, "A Quantitative Discriminant Method of Elbow Point for the Optimal Number of Clusters in Clustering Algorithm," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, p. 31, 2021.
[68] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *CVPR*, 2016, pp. 2818–2826.
[69] G. Pereyra, G. Tucker, J. Chorowski, Ł. Kaiser, and G. Hinton, "Regularizing Neural Networks by Penalizing Confident Output Distributions," in *ICLR (Workshop)*, 2017.
[70] E. Bauer and R. Kohavi, "An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants," *Machine learning*, vol. 36, no. 1, pp. 105–139, 1999.
[71] L. Lam and S. Suen, "Application of Majority Voting to Pattern Recognition: An Analysis of Its Behavior and Performance," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 27, no. 5, pp. 553–568, 1997.
[72] J. Munkres, "Algorithms for the Assignment and Transportation Problems," *Journal of the society for industrial and applied mathematics*, vol. 5, no. 1, pp. 32–38, 1957.
[73] D. Cai, W. Cai, and M. Li, "Within-Sample Variability-Invariant Loss for Robust Speaker Recognition Under Noisy Environments," in *ICASSP*, 2020, pp. 6469–6473.
[74] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," *arXiv:1510.08484*, 2015.
[75] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *CVPR*, 2016, pp. 770–778.
[76] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *ICLR*, 2015.
[77] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
[78] "NIST 2016 Speaker Recognition Evaluation Plan," 2016. [Online]. Available: https://www.nist.gov/sites/default/files/documents/2016/10/07/sre16_eval_plan_v1.3.pdf
[79] L. Maaten and G. Hinton, "Visualizing Data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. 11, pp. 2579–2605, 2008.
[80] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-Excitation Networks," *CVPR*, 2019.
[81] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," in *CVPR*, 2019, pp. 4685–4694.

Danwei Cai is pursuing his Ph.D. degree in electrical and computer engineering at Duke University. He received his bachelors degree in software engineering and masters degree in electronics and communication engineering from Sun Yet-Sen University in China. His primary research interests are in the area of speech processing, including speech recognition, speaker recognition, speaker diarization, and computational linguistics.



Weiqing Wang received the B.S. (2018) in Computer Science from Sun Yat-sen University, and he is currently a Ph.D. student in Department of Electrical and Computer Engineering at Duke University. His research interests focus on speaker diarization. Before joining Duke ECE, he was a research assistant at Duke Kunshan University (2018-2019), working on automatic piano transcription.

Ming Li received his Ph.D. in Electrical Engineering from University of Southern California in 2013. He is currently an Associate Professor of Electrical and Computer Engineering at Duke Kunshan University. He is also a research scholar at Department of Electrical and Computer Engineering at Duke University. His research interests are in the areas of audio, speech and language processing as well as multimodal behavior signal analysis and interpretation. He has published more than 140 papers and served as the member of IEEE speech and language technical committee, CCF speech dialogue and auditory processing technical committee, CAAI affective intelligence technical committee, APSIPA speech and language processing technical committee. He was the area chair of speaker and language recognition at Interspeech 2016, 2018 and 2020. Works co-authored with his colleagues have won first prize awards at Body Computing Slam Contest 2009, Interspeech Computational Paralinguistic Challenge 2011, 2012 and 2019, ASRU 2019 MGB-5 Challenge, Interspeech 2020 and 2021 Fearless Steps Challenge, VoxSRC 2021 Challenge, M2MeT2022 Challenge. He received the IBM faculty award in 2016, the ISCA Computer Speech and Language 5-years best journal paper award in 2018 and the youth achievement award of outstanding scientific research achievements of Chinese higher education in 2020. He is a senior member of IEEE.