LEARNING AUDIO-VISUAL SPEECH REPRESENTATION BY MASKED MULTIMODAL CLUSTER PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Video recordings contain correlated audio and visual information, providing a strong signal for speech representation learning from the audio stream and the speaker's lip movements. We introduce Audio-Visual Hidden Unit BERT (AV-HuBERT), a self-supervised representation learning framework for audio-visual speech, which masks multi-stream video input and predicts automatically discovered and iteratively refined multimodal hidden units. AV-HuBERT learns powerful audio-visual speech representation benefiting both lip-reading and automatic speech recognition. On the largest public lip-reading benchmark LRS3 (433 hours), AV-HuBERT achieves 34.6% WER with only 30 hours of labeled data, 1% away from the former state-of-the-art approach (33.6%) trained with a thousand times more transcribed video data (31K hours) (Makino et al., 2019). The lip-reading WER is further reduced to 30.6% when using all 433 hours of labeled data from LRS3, achieving a new state-of-the-art result using an order-of-magnitude less video data for pre-training and fine-tuning. Using our audio-visual representation on the same benchmark for audio-only speech recognition leads to a 40% relative WER reduction over the state-of-the-art performance (1.4% vs 2.3%). Our models and codes will be publicly available.

1 INTRODUCTION

Human perception of speech is intrinsically multimodal, involving audition and vision. The speech production is accompanied by the movement of lips and teeth, which can be visually interpreted to understand speech. Visual cues of speech not only play an essential role in language learning for pre-lingual children (Meltzoff & Moore, 1977; Davies et al., 2008), but also improve speech understanding in noisy environment (Sumby & Pollack, 1954) and provide patients of speech impairment with means of communication. Furthermore, perceptual studies (McGurk & MacDonald, 1976) have shown that such visual cues can alter the perceived sound.

For machine learning models, the tight coupling between audio and visual lip movement information emerges as a natural source for supervision to learn speech representations, which has not been extensively utilized yet in the self-supervised speech representation learning literature. Recent successful representation learning frameworks for speech (e.g., APC (Chung et al., 2019), CPC (Oord et al., 2018; Kharitonov et al., 2021), wav2vec 2.0 (Baevski et al., 2020; Hsu et al., 2021b), DeCoAR2.0 (Ling & Liu, 2020), HuBERT (Hsu et al., 2021c;a)) are mostly built entirely on audio. The fundamental research question addressed in this paper is whether a self-supervised audio-visual speech representation learned from the lip movement information, alongside the audio signal in video recordings, captures cross-modal correlations and improves downstream performance for visual speech recognition (i.e., lip reading) and automatic speech recognition (ASR) tasks. Existing ML models for lip-reading rely heavily on text transcriptions to achieve an acceptable level of accuracy. The state-of-the-art lip-reading model (Makino et al., 2019) requires 31K hours of transcribed video data for training. Such large amounts of labeled data are expensive and hard to obtain for most of the world's 7,000 languages. The benefit from a robust visual speech representation learning framework goes beyond lip-reading. Additionally, it can benefit a vast range of applications, including but not limited to keyword spotting in sign language (Albanie et al., 2020), speech enhancement (Xu et al., 2020) and talking face generation (Chen et al., 2018).

In this paper, we present Audio-Visual Hidden Unit BERT (AV-HuBERT), a multimodal self-supervised speech representation learning framework. It encodes masked audio and image sequences into audio-visual features via a hybrid ResNet-transformer architecture to predict the pre-determined sequence of discrete cluster assignments. The target cluster assignments are initially generated from signal processing-based acoustic features (e.g., MFCC) and iteratively refined using the features learned by the audio-visual encoder via k-means clustering. AV-HuBERT simultaneously captures linguistic and phonetic information for unmasked regions from both the lip-movement and audio streams into its latent representations, then encodes their long-range temporal relationships to solve the masked-prediction task.

The contextualized representations learned by AV-HuBERT show excellent transferability to the lip-reading task, where only the visual modality is available. Pre-training on audio and visual input streams led to substantially better results than only visual input. In the low-resource setup using only 30 hours of labeled data from LRS3 (Afouras et al., 2018b), our model achieves a lip-reading WER of 34.6%, 1% away from the previous state-of-the-art model (33.6%) trained on 31,000 hours of transcribed videos (Makino et al., 2019). When the pre-trained AV-HuBERT model is fine-tuned using the complete 433 hours from LRS3, it sets a new lip-reading WER record of 30.6%. In addition, we show that the multimodal clusters derived from AV-HuBERT can be used to pre-train a HuBERT model for audio-based speech recognition, outperforming the previous state-of-the-art model (2.3%) and the unimodal HuBERT pre-trained on audio clusters (1.7%) by a large margin (1.4%).

2 Related Work

The strong correlation between video modalities provides an effective means for self-supervised representation learning on videos, which has been explored in many prior works and is still an active research area. This work draws inspiration from two lines of previous research:

Multimodal general video representation focuses on learning self-supervised audio-visual representations of general videos to solve high-level semantic tasks, e.g., action recognition and audio event detection (Arandjelović & Zisserman, 2017; Bruno et al., 2018; Morgado et al., 2021). Owens & Efros (2018) learns a multimodal network to predict whether the audio and visual streams of a video are temporally synchronized while Pham et al. (2019) applies a cyclic translation between different modalities. Piergiovanni et al. (2020) learns the visual representation through a multi-tasking framework that incorporates a series of pretext tasks such as reconstruction and temporal ordering prediction. Sharing our work's inspiration of DeepClustering (Caron et al., 2018), XDC (Alwassel et al., 2020) learns cross-modal representations through predicting cross-modal or cluster assignments. In contrast to XDC, AV-HuBERT is trained with a BERT-like masked prediction loss, which forces the model to learn the structure within the multimodal input and was shown in Hsu et al. (2021c) to be more resilient to bad cluster assignments compared to unmasked cluster prediction.

Semi- and self-supervised audio-visual speech representation learning focuses on improving lip-reading with untranscribed audio-visual speech data. To solve isolated visual word recognition, Chung et al. (2020) learns visual embeddings using a contrastive loss based on audio-visual synchronization. Ma et al. (2021a) learns visual speech representations by minimizing the distance between its latent features and off-the-shelf audio embeddings. Using an external supervised ASR to transcribe unlabeled audio, Afouras et al. (2020) trains their lip-reading model on the augmented labeled and pseudo-labeled data. Unlike Ma et al. (2021a) and Afouras et al. (2020), our model is trained from scratch and encouraged to learn contextualized representations with a masked prediction task. Moreover, our method does not rely on any external pre-trained models.

3 Method

3.1 Preliminary: Audio HuBERT

Our research builds on Audio HuBERT (Hsu et al., 2021a) which is a self-supervised learning framework for speech and audio. It alternates between two steps: feature clustering and masked prediction. In the first step, a discrete latent variable model (e.g., k-means) is applied to a sequence of acoustic frames $\mathbf{A}_{1:T}$ producing a sequence of frame-level assignments $\mathbf{z}^a_{1:T}$. Clusters of signal-

processing-based acoustic features, e.g., Mel-frequency cepstral coefficients (MFCC), exhibit non-trivial correlations with the inherent acoustic units of speech inputs. Using $(\mathbf{A}_{1:T}, \mathbf{z}^a_{1:T})$ pairs, the second step learns new feature representations by minimizing a masked prediction loss, similar to masked language modeling in BERT (Devlin et al., 2019). The pressure to predict cluster assignments of masked audio regions forces the model to learn good local acoustic representations for unmasked regions and long-range temporal dependencies between latent features. Repeating these two pre-training steps improves the quality of learned representations through better and fine-grained cluster assignments $\mathbf{z}^a_{1:T}$.

3.2 Single-Modal & Cross-Modal Visual HuBERT

Single-modal Visual HuBERT: The first way to extend Audio HuBERT to the visual domain is by generating targets using visual features. Formally, given an image sequence $\mathbf{I}_{1:T}$, we first cluster the image features into a sequence of discrete units $\mathbf{z}^i_{1:T}$ via k-means: $z^i_t = \text{k-means}(G(\mathbf{I}_t)) \in \{1, 2, ..., V\}$, where G is an visual feature extractor and V is the codebook size. The cluster assignments $\mathbf{z}^a_{1:T}$ serve as the prediction targets of the model. Initially, G can be an engineered image feature extractor such as Histogram of Oriented Gradients (HoG), analogous to MFCC in audio HuBERT. The intermediate layers of the HuBERT model are used as G in later iterations.

To perform the masked prediction task, the model first encodes $\mathbf{I}_{1:T}$ using a ResNet architecture into an intermediate visual feature sequence $\mathbf{f}^v_{1:T}$, which is then corrupted into $\tilde{\mathbf{f}}^v_{1:T}$ via a binary mask M. Specifically, $\forall t \in M$, $\tilde{\mathbf{f}}^v_t$ is replaced with a learned masked embedding. We adopt the same strategy in Hsu et al. (2021a) to generate span masks. The masked visual features $\tilde{\mathbf{f}}^v_{1:T}$ are encoded into a sequence of contextualized features $\mathbf{e}_{1:T}$ via a transformer encoder followed by a linear projection layer. The loss is then applied over the masked regions and optionally over unmasked regions (controlled by the parameter α):

$$\begin{aligned} \mathbf{p}_t &= \text{Softmax}(\mathbf{W}\mathbf{e}_t + \mathbf{b}), 1 \leq t \leq T \\ L &= -\sum_{t \in M} \log p_t(z^i_t) - \alpha \sum_{t \notin M} \log p_t(z^i_t) \end{aligned} \tag{1}$$

Where ($\mathbf{W} \in \mathbb{R}^{d \times V}$, $\mathbf{b} \in \mathbb{R}^V$) are parameters of the projection layer which maps features into logits for the cluster assignments.

Cross-modal Visual HuBERT: The single-modal visual HuBERT aims to learn visual speech representation through gradually refined image features. However, it does not employ the audio stream of the video. Presumably, audio features, e.g., MFCC or a pre-trained audio HuBERT model, are more informative about phonetics than vanilla image features (e.g., HoG), which mainly capture low-level image primitives, thus benefiting visual speech representation more. To this end, we train an audio encoder based on the aligned audio frame sequence $\mathbf{A}_{1:T}$ in parallel to the visual encoder. The iterative training alternates between the two encoders. In each iteration, an audio encoder E^a is utilized to generate target cluster assignments $\mathbf{z}^a_{1:T}$. The visual encoder E^v is trained subsequently with $(\mathbf{I}_{1:T}, \mathbf{z}^a_{1:T})$. The $\mathbf{z}^a_{1:T}$ is further used to train the next iteration of the audio encoder E^a for refinement. The cross-modality visual HuBERT can be seen as modeling visual inputs by implicitly distilling knowledge from the audio stream, where $\mathbf{z}^a_{1:T}$ represents the audio-side knowledge. We hypothesize that the audio feature is more favorable to speech representation learning than the visual feature, which is validated in the Section D.1.

Critical for the lip-reading downstream task, the masked prediction objective used by HuBERT forces the model to capture temporal relationships, which facilitates prediction of homophemes, which are groups of sounds with identical visual shapes (e.g., 'p'-'b', 'f'-'v', 'sh'-'ch') that are impossible to distinguish using a single image frame.

3.3 Audio-visual HuBERT

Our primary model in this work is Audio-Visual HuBERT (AV-HuBERT), shown in figure 1, which is trained iteratively by alternating between feature clustering and masked prediction in a similar way to the Visual HuBERT but with four main differences:

Audio-visual input: The AV-HuBERT model consumes both acoustic and image frames for the masked prediction training, which enables better modeling and distillation of the correlations be-

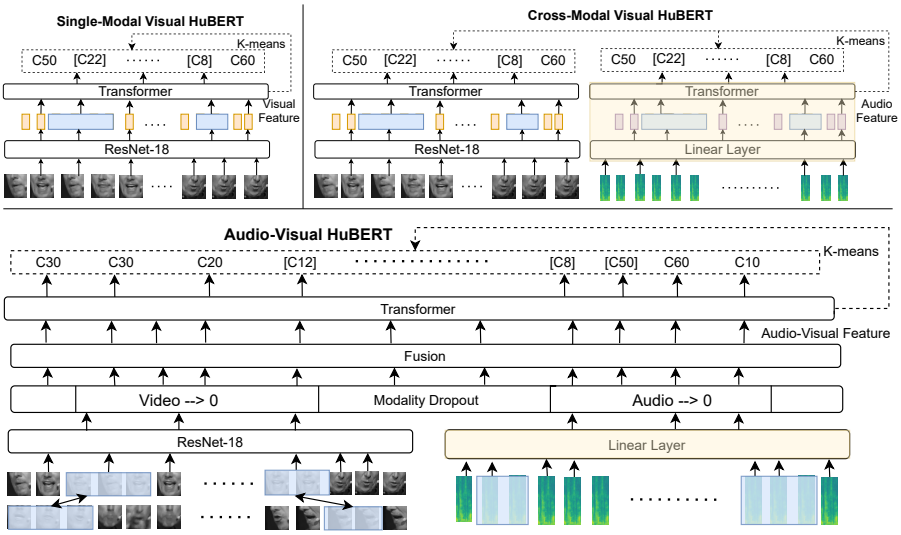


Figure 1: Illustration of our models for visual speech representation learning. The yellow region denotes the audio-specific modules which are removed during fine-tuning for the downstream lip-reading task.

tween the two modalities. Specifically, image sequences and acoustic features pass through their modality-specific encoders to produce intermediate visual and audio features, which are then fused and fed into a shared backbone transformer encoder to predict masked cluster assignments. The targets are generated from clustering audio features or features extracted from the previous iteration of the AV-HuBERT model. When fine-tuned for lip-reading, we drop the audio input to work solely with the visual input. The input discrepancy is addressed by modality dropout described next.

Modality dropout: Audio-visual speech recognition models can relate audio input to lexical output more effortlessly than the visual input stream, as observed in the literature (Afouras et al., 2018a; Ma et al., 2021b). This causes the audio modality to dominate model decisions. The problem is aggravated in our setting because the target cluster assignments are initially generated from acoustic features. To prevent the model's over-reliance on the audio stream in our joint model, we only use a linear layer to encode acoustic input to force the audio encoder to learn simple features.

Additionally, before fusing audio and visual inputs into the backbone joint transformer encoder, dropout is applied to mask the full features of one modality; we refer to it as modality dropout. With a probability p_m, both modalities are used as input. When only modality is used, the audio stream is selected with a probability of p_a. Formally, given the encoded audio and visual feature sequence $\mathbf{f}^a_{1:T}$ and $\mathbf{f}^v_{1:T}$, equation 2 shows feature fusion equipped with modality dropout:

$$\mathbf{f}^{av}_{1:T} = \begin{cases} \text{Fuse}(\mathbf{f}^a_t, \mathbf{f}^v_t)_{1\leq t\leq T} & \text{with } p_m \\ \text{Fuse}(\mathbf{f}^a_t, \mathbf{0})_{1\leq t\leq T} & \text{with } (1-p_m)p_a \\ \text{Fuse}(\mathbf{0}, \mathbf{f}^v_t)_{1\leq t\leq T} & \text{with } (1-p_m)(1-p_a) \end{cases} \tag{2}$$

where Fuse is the fusion operator and $\mathbf{f}^{av}_{1:T}$ is the fused feature sequence. We use concatenation as the default fusion operator. The modality dropout prevents the model from degenerating into an audio-focused one and implicitly encourages its visual encoder to learn from audio input during pre-training, thus boosting the knowledge transfer from audio to video. Furthermore, since the fine-tuning and inference phases use the visual stream alone (no audio input), this modality dropout mechanism bridges the gap between pre-training (multimodal) and fine-tuning/inference (single-modality). A similar dropout mechanism is used in prior work (Zhang et al., 2019a; Makino et al., 2019) for audio-visual speech recognition to increase the robustness. We verify the modality dropout effectiveness in Section C.

Audio-visual clustering: One benefit of pre-training on both modalities is the ability to generate multimodal cluster assignments that serve as target labels for the masked prediction task of the next iteration. In contrast to the Cross-modal Visual HuBERT where targets are generated from

audio-based features or a prior Audio HuBERT model, the targets for AV-HuBERT are naturally multimodal after the first iteration. Lip movement sequences provide complementary information to the audio stream. Combining both modalities produces cluster assignments of higher quality for AV-HuBERT, as shown in Section D.1.

Masking by substitution: We propose a novel masking strategy for AV-HuBERT that masks segments in the visual stream by substituting them with random segments from the same video. More formally, given an input video $\mathbf{I}^v_{1:T}$, an imposter video $\mathbf{I}^{v,f}_{1:T_f}$ and a mask consisting of n intervals $M = \{(s_i, t_i)\}_{1 \leq i \leq n}$, we corrupted $\mathbf{I}^v_{1:T}$ into $\tilde{\mathbf{I}}^v_{1:T}$ by setting:

$$\tilde{\mathbf{I}}^v_{s_i:t_i} = \mathbf{I}^{v,f}_{p_i:p_i+t_i-s_i}, \forall 1 \leq i \leq n \tag{3}$$

where p_i is a randomly selected integer from the interval $[0, T_f - t_i + s_i]$. Our motivation for replacing video segments is to keep the masked region smooth temporally, compared to random frames replacement. Using masked prediction with real-video substitutions, the model first needs to *identify the fake frames* and then *infer the labels belonging to the original video segment*. As the "filled-in" frames are from real videos rather than learned masking vectors, the fake segment detection sub-task improves the learned features of the ResNet encoder. Compared to vanilla masking, the substitution masking preserves the visual "naturalness" and avoids gaps in the video sequence during pre-training and fine-tuning. We show an ablation comparing different replacement setups in Section C.

The audio and visual segments are masked independently using two different masking probabilities m_a and m_v. We hypothesize that the difficulty of the masked prediction task differs for each modality: inferring the masked targets given the audio stream is more straightforward than using the lip movement stream. Setting a high masking probability for acoustic frames is essential to help the whole model capture the language characteristics. On the contrary, setting a high masking probability for the visual input provide the model with more imposter segments than the original ones, hurting its ability to learn meaningful features (studied in Section C of the appendix). Given the output probability $\mathbf{p}_{1:T}$ and target cluster assignments $\mathbf{z}_{1:T}$, the AV-HuBERT pre-training loss is:

$$L = -\sum_{t \in M^a \cup M^v} \log p_t(z_t) - \alpha \sum_{t \notin M^a \cup M^v} \log p_t(z_t) \tag{4}$$

where M^a and M^v are masks for audio and visual stream. α is a hyperparameter weighting the contribution of the unmasked regions in the overall objective.

3.4 Fine-tuning

The proposed pre-training approaches can be fine-tuned for visual speech recognition using any sequence classification loss. In this paper, we focus on fine-tuning with the connectionist temporal classification (CTC) (Graves et al., 2006) and attention-based sequence-to-sequence cross-entropy loss (Bahdanau et al., 2016) (S2S for brevity), which are the most commonly used approaches in the prior visual speech recognition work. Assume that the feature sequence output of our pre-trained model is $\mathbf{e}_{1:T}$ and the ground-truth transcription is $w = w_1, w_2, ..., w_s$. For CTC, a projection layer is used to map the visual feature sequence into the output probabilities: $\mathbf{p}_t = \text{Softmax}(\mathbf{W}^{ft}\mathbf{e}_t + \mathbf{b}^{ft})$, where $\mathbf{W}^{ft} \in \mathbb{R}^{d \times (U+1)}$, $\mathbf{b}^{ft} \in \mathbb{R}^{U+1}$ and U is the output vocabulary size ($+1$: plus blank symbol). The model is trained with CTC loss:

$$L_{ctc} = -\log \sum_{\pi \in \mathcal{B}^{-1}(w)} p(\pi | \mathbf{e}_{1:T}) \tag{5}$$

where $\mathcal{B}$ consists of all possible alignment sequences π corresponding to w. For S2S, a tranformer decoder is appended to the pre-trained encoder to autoregressively decode the feature sequence $\mathbf{e}_{1:T}$ into target probabilities $p(w_t | w_{1:t-1}, \mathbf{e}_{1:T})$. The whole model is trained with cross entropy loss:

$$L_{s2s} = -\sum_{t=1}^{s} \log p(w_t | w_{1:t-1}, \mathbf{e}_{1:T}) \tag{6}$$

4 EXPERIMENT

4.1 SETUP

We conduct experiments on two datasets: LRS3 (Afouras et al., 2018b) with 433 hours of transcribed English videos and VoxCeleb2 (Chung et al., 2018) with 2442 hours of unlabeled multilingual videos. We only use the English portion of VoxCeleb2, which amounts to 1,326 hours of content. The inputs to our backbone model are lip Regions-Of-Interest (ROIs) for the visual stream and log filterbank energy feature for the audio stream. The image encoder is a modified ResNet-18, which has been used in prior work (Ma et al., 2021b; Martinez et al., 2020; Stafylakis & Tzimiropoulos, 2017). The audio encoder is simply a linear projection layer. We consider two model configurations: BASE with 12 transformer blocks and LARGE with 24 transformer blocks. For both BASE and LARGE, each transformer block is of 768 embedding dimension, 3072 feed-forward dimension, and 12 attention heads. The number of parameters in BASE and LARGE are 103M and 188M respectively.

The model uses five iterations of feature clustering and masked prediction during pre-training. See Section A.4, D for details on clustering. For fine-tuning, we use phone targets for the CTC loss and unigram-based subword units (Kudo, 2018) for the S2S loss. For decoding the CTC-trained model, we use a 4-gram language model trained on LRS3 training text. For the S2S fine-tuned model, we rely only on the joint decoder module to incorporate language information, with no external language model employed during inference. More details about the used datasets, data pre-processing, and model training are in Section A.

4.2 MAIN RESULT

Table 1 compares the performance of our AV-HuBERT pre-training approach to previously published supervised, semi-supervised, and self-supervised lip-reading systems using different amounts of labeled and unlabeled data. Since the CTC and S2S fine-tuning approaches have similar trends, only S2S results are shown in table 1. Complete results of CTC fine-tuning can be found in Table B.3.[1]

Using 1,759 hours unlabeled data for pre-training and only 30 hours of labeled data for fine-tuning, AV-HuBERT-Large outperforms almost all the prior supervised lip-reading models and is 1% away from the model in (Makino et al., 2019) which is trained with 1000 times more labeled data. Once we use the whole training set of LRS3 in the fine-tuning stage, our S2S AV-HuBERT LARGE model achieves a new SOTA result, even though the total amount of data used to pre-train and fine-tune our model is about 7% of the data used for training the model in Makino et al. (2019). Many prior models pre-train their visual front-end, e.g., ResNet-18, using word-level annotated lip-reading videos, which is costly to collect since they require word boundary information. In contrast to these models, our models are fully pre-trained from scratch using the proposed approach.

Compared to the semi-supervised approach Jasper-KD (Afouras et al., 2020), which transcribed 334 hours of the English data in VoxCeleb2 using a pre-trained ASR system,[2] our best model achieves 29% lower absolute WER benefiting from VoxCeleb2 for pre-training. Even when limiting our model to the LRS3 data for pre-training and fine-tuning, our model surpasses their semi-supervised system by 18%. Compared to LiRA (Ma et al., 2021a), a recently proposed self-supervised model for lip-reading, AV-HuBERT-Base provides $\sim 17.5\%$ lower absolute WER on average for low-resource and high-resource settings. The implementation details of LiRA are provided in Section A.5.

With the same network architecture, our pre-training approach significantly reduces WER compared to training from scratch, in both low-resource ($92.5\% \rightarrow 34.6\%$, large) and high-resource ($59.1\% \rightarrow 30.5\%$, large) settings. A qualitative view of the improvement can be found in Section E. Additionally, we notice that our AV-HuBERT pre-training helps under a fully supervised setting. Using the LRS3 data only (433 hours), pre-training followed by fine-tuning (41.9%, large) outperforms

[1]The prior work in (Ma et al., 2021b) uses an outdated version of LRS3 (before 2018) with speaker overlap in training and test data, which is no longer publicly available. Its best results on the current version are included in table 1& B.3 . For comparison, we simulate the old closed-speaker setting in Section B.2.

[2]The gap in data amount (334hr vs 1,759hr) is due to the different parameters and ASR model used for filtering non-English data in VoxCeleb2.

Table 1: WER (%) of our models and prior work on the LRS3 dataset. †We re-implemented Ma et al. (2021a) with the same architecture since the author source code was not provided.

Method	Backbone	Criterion	Labeled iso (hrs)	Labeled utt (hrs)	Unlabeled data (hrs)	WER (%)
Supervised						
Afouras et al. (2020)	CNN	CTC	157	433	-	68.8
Zhang et al. (2019b)	CNN	S2S	157	698	-	60.1
Afouras et al. (2018a)	Transformer	S2S	157	1,362	-	58.9
Xu et al. (2020)	RNN	S2S	157	433	-	57.8
Shillingford et al. (2019)	RNN	CTC	-	3,886	-	55.1
Ma et al. (2021b)	Conformer	CTC+S2S	-	433	-	46.9
Ma et al. (2021b)	Conformer	CTC+S2S	157	433	-	43.3
Makino et al. (2019)	RNN	Transducer	-	31,000	-	**33.6**
Semi-Supervised & Self-Supervised						
Afouras et al. (2020)	CNN	CTC	157	433	334	59.8
Ma et al. (2021a)†	Transformer-Base	S2S	-	30	433	71.9
			-	433	1,759	49.6
Proposed (Self-Supervised)						
AV-HUBERT	Transformer-Base	S2S	-	30	-	94.3
			-	30	433	51.8
			-	30	1,759	**46.1**
			-	433	-	60.3
			-	433	433	44.0
			-	433	1,759	**34.8**
	Transformer-Large	S2S	-	30	-	92.5
			-	30	433	47.5
			-	30	1,759	**34.6**
			-	433	-	59.1
			-	433	433	41.9
			-	433	1,759	**30.5**

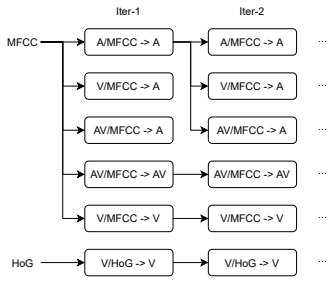
training a lip-reading model from scratch (59.1%, large) to predict the output text. AV-HuBERT, pre-trained on video-audio pairs, learns better fine-grained visual representation than the scratch model trained on video-text pairs. The benefits of AV-HuBERT pre-training in various labeled data setups can be found in Section B.1.

4.3 AV-HUBERT VS. VISUAL HUBERT

We compare AV-HuBERT against a suite of alternatives, including the Single-modal and Cross-modal Visual HuBERT in table 2. All the models are pretrained on 433 hours of unlabeled data and fine-tuned on 30 hours of labeled data. For this comparison, we use CTC fine-tuning due to its computational efficiency and given their similarity in results trends to S2S.

Table 2: Fine-tuning performance (in WER, %) of AV-HuBERT and visual HuBERT on different target labels. Init: feature in the initial iteration, sub: feature in subsequent iterations. AV: AV-HuBERT, V: Visual-HuBERT, A: Audio-HuBERT.

Model/init→sub	Iteration 1	2	3	4	5
AV/MFCC→AV	71.5	63.6	60.9	58.8	58.2
AV/MFCC→A	71.5	64.3	63.5	-	-
V/MFCC→A	75.4	69.4	69.1	-	-
V/MFCC→V	75.4	72.6	72.3	-	-
V/HoG→V	80.3	80.1	-	-	-



As shown in table 2, target cluster assignments driven from a single modality, either audio or visual, do not provide much WER reduction beyond the second iteration. Training the AV-HuBERT model using targets driven from audio-visual features keeps improving for more iterations and achieves better final performance. As mentioned in Section 3, visual information is complementary to audio input; hence, using both produces higher quality target clusters. Measuring the quality of different target labels using cluster quality metrics such as purity and NMI shows the same trend observed from lip-reading WER (See appendix D.1).

Fixing target labels used for the masked-prediction pre-training, AV-HuBERT (AV/MFCC→A) outperforms the Cross-modal Visual HuBERT (V/MFCC→A) by a large margin. AV-HuBERT effectively transfers knowledge from audio into the visual encoder and the backbone transformer model to benefit visual-only fine-tuning and inference. In contrast to AV-HuBERT, iterative pe-training brings much smaller gains to single-modality visual HuBERT ("V/MFCC→V", "V/HoG→V").

Starting with audio features is critical for learning effective target labels for the masked-prediction task. Phonetic information, which is crucial for lip-reading, is primarily present in the audio stream. All the models considered so far are based on audio feature MFCC clustering in their initial iteration. As is shown in the "V/HoG→V" row, using hand-engineered visual features provides a lousy starting point for iterative learning. Visual features such as HoG mainly incorporate low-level visual cues such as edges and luminance, which is irrelevant to the downstream recognition task. Using features of a higher correlation with phonetic units are more likely to benefit the final lip-reading model. Indeed, clustering MFCC features show much higher purity (30.3%) than HoG clusters (16.4%) if one considers phone labels as the target units, as shown in Table D.1.

4.4 MULTILINGUAL VS. MONOLINGUAL

Our AV-HuBERT approach is agnostic to the spoken language of the video, making it straightforward to apply to multilingual datasets. For low-resource languages, multilingual data can effectively increase the number of unlabeled data points for pre-training. However, multilingual data also enlarges the discrepancy between pre-training and fine-tuning. To study how these two factors affect AV-HuBERT, we compare using monolingual English data only versus multilingual videos in the pre-training phase. In particular, we vary the amount of English data we use in pre-training while the number of non-English utterances is fixed to 1,116 hours in all the settings. For simplicity, we train AV-HuBERT for one iteration with MFCC clusters and fine-tune it with CTC using 30 hours of labeled data.

Table 3: WER (%) with different amounts of unlabeled English utterances in pre-training. Non-En data: 1,116 hours. Labeled data for fine-tuning: 30 hours.

Hours of En data for Pre-Training	0	100	400	800	1759
Pre-train on En only, WER (%)	84.1	77.8	68.9	67.9	**59.9**
Pre-train on En + 1,116 hr of non-En, WER (%)	**70.6**	**68.4**	**67.4**	**66.6**	64.3

As is shown in table 3, using non-English data in pre-training significantly reduces WER when there are no or very little English data in pre-training ($\leq$ 100 hours). As we increase the amount of English data, the gain diminishes because the out-of-domain effect brought by non-English data outweighs the benefit of the overall increase in pre-training data. Using the whole English data only for pre-training is better than combining it with other languages (59.9% vs. 64.3%). This experiment highlights the importance of the domain match between pre-training and fine-tuning data. For zero/low-resource settings, merging data from other languages in pre-training benefits the downstream task. When the unlabeled data from the target language is abundant, limiting the pre-training data to one language is beneficial.

4.5 ASR PERFORMANCE

The multimodal clusters produced by AV-HuBERT, which have higher quality than audio-HuBERT targets, can also benefit speech pre-training. To test our hypothesis, we trained an *audio-HuBERT*, with only audio input during the masked-prediction pre-training, for one iteration with cluster assignments driven from AV-HuBERT features. We also pre-trained an audio-HuBERT from scratch

using clusters driven from the MFCC features for three iterations. The two pre-trained models are evaluated on a downstream ASR task.

Table 4 shows the performance of different models fine-tuned on the ASR task. We only include the performance of S2S fine-tuning for our models as it consistently outperforms the CTC fine-tuning. An audio-HuBERT pre-trained using targets generated by AV-HuBERT features outperforms the vanilla audio-HuBERT in low-resource (30h) and high-resource settings (433h) fine-tuning settings across different model architectures. With the same amount of labeled data, our best model (1.4%) outperforms the prior SOTA (2.3%) even without an external language model during inference.

Given that the AV-HuBERT model utilizes both audio and visual modalities at its input, it can be fine-tuned, in principle, for the ASR downstream task. However, it produced an inferior WER than an audio-HuBERT that does not consume any visual input during pre-training. This observation suggests that, even though the AV-HuBERT features benefit the ASR task, its transformer backbone encoder representations are more biased towards the visual input modality.

Table 4: ASR WER (%) of audio-HuBERT pre-trained with audio-only/audio-visual clusters and their comparison to prior work on the LRS3 dataset.

Method	Backbone	Criterion	LM	Labeled data (hrs)	Unlabeled data (hrs)	WER (%)
Supervised						
Afouras et al. (2018a)	Transformer	S2S	√	1,362	-	8.3
Afouras et al. (2018a)	Transformer	CTC	√	1,362	-	8.9
Xu et al. (2020)	RNN	S2S	-	433	-	7.2
Ma et al. (2021b)	Conformer	CTC+S2S	√	433	-	**2.3**
Self-Supervised						
Hsu et al. (2021a) (A/MFCC→A)	Transformer-Base	S2S	-	30	433	5.4
			-	30	1,759	5.0
			-	433	1,759	2.4
	Transformer-Large	S2S	-	30	433	4.5
			-	30	1,759	3.6
			-	433	1,759	**1.7**
Proposed (Self-Supervised)						
A/MFCC→AV	Transformer-Base	S2S	-	30	433	4.9
			-	30	1,759	3.8
			-	433	1,759	2.0
	Transformer-Large	S2S	-	30	433	4.3
			-	30	1,759	2.9
			-	433	1,759	**1.4**

5 CONCLUSION

We presented multiple pre-training models for visual speech recognition. Our AV-HuBERT model leverages the strong correlation between the audio and lip movement streams for self-supervised audio-visual speech representation learning. Our pre-training approaches iteratively alternate between feature clustering and learning new features through a masked-prediction loss. The AV-HuBERT model consumes masked image and audio frames to predict target cluster assignments. The targets are initially generated from MFCC features and gradually refined through iterative training. Experiments on visual speech recognition show that AV-HuBERT achieves SOTA using 433 hours of text transcriptions, two orders of magnitude less than the 31,000 hours of labeled data used in the prior best approach. When using only one-thousandth of labeled data, the lip-reading performance is 1% away from the prior SOTA. AV-HuBERT also improves the representation for the ASR downstream task. An audio-HuBERT model trained with targets generated by an AV-HuBERT model shows superior performance, achieving the SOTA in the audio-based speech recognition in the LRS3 dataset. As future work, AV-HuBERT can be applied for multilingual lip-reading in low-resource languages. Additionally, our approach can be extended to other applications of visual speech representation, such as speech enhancement and generation.

ETHICAL STATEMENT

All the data used in this paper are publicly available and are used under the following three licenses: the TED terms of use, the Creative Commons BY-NC-ND 4.0 license and Creative Commons Attribution 4.0 International License. Through spot-checking, we find the datasets are gender balanced and cover a wide range of races and ages. However, the distribution of speakers in the data may not be representative of the global human population. Please be cautious of unintended societal, gender, racial and other biases caused by the fact. To maintain anonymity, only the mouth area of a speaker is visualized wherever used in the paper.

REPRODUCIBILITY STATEMENT

Our code and models will be publicly available. In the meantime, we include as many implementation details as we can in the paper.

REFERENCES

The CMU pronouncing dictionary. http://www.speech.cs.cmu.edu/cgi-bin/cmudict.

Triantafyllos Afouras, Joon Son Chung, A. Senior, Oriol Vinyals, and Andrew Zisserman. Deep audio-visual speech recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2018a.

Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. LRS3-TED: a large-scale dataset for visual speech recognition, 2018b. arXiv:1809.00496.

Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. ASR is all you need: Cross-modal distillation for lip reading. In *ICASSP*, 2020.

Samuel Albanie, Gül Varol, Liliane Momeni, Triantafyllos Afouras, Joon Son Chung, Neil Fox, and Andrew Zisserman. BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues. In *ECCV*, 2020.

Humam Alwassel, Dhruv Mahajan, Bruno Korbar, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *NeurIPS*, 2020.

Relja Arandjelović and Andrew Zisserman. Look, listen and learn. In *ICCV*, 2017.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Alexei Baevski, Henry Zhou, Abdel rahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *NeurIPS*, 2020.

Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio. End-to-end attention-based large vocabulary speech recognition. In *ICASSP*, 2016.

Korbar Bruno, Tran Du, and Torresani Lorenzo. Cooperative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 132–149, 2018.

Lele Chen, Zhiheng Li, Ross K. Maddox, Zhiyao Duan, and Chenliang Xu. Lip movements generation at a glance. In *ECCV*, 2018.

Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. Voxceleb2: Deep speaker recognition. In *INTERSPEECH*, 2018.

Soo-Whan Chung, Hong Kang, and Joon Son Chung. Seeing voices and hearing voices: learning discriminative embeddings using cross-modal self-supervision. In *Interspeech*, 2020.

Yu-An Chung, Wei-Ning Hsu, Hao Tang, and James Glass. An unsupervised autoregressive model for speech representation learning. *arXiv preprint arXiv:1904.03240*, 2019.

Rebecca Davies, Evan Kidd, and Karen Lander. Investigating the psycholinguistic correlates of speechreading in preschool age children. *International journal of language and communication disorders*, 44:164–74, 06 2008.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.

Angela Fan, Edouard Grave, and Armand Joulin. Reducing transformer depth on demand with structured dropout. In *ICLR*, 2020.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*, 2006.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *arXiv preprint arXiv:2106.07447*, 2021a.

Wei-Ning Hsu, Anuroop Sriram, Alexei Baevski, Tatiana Likhomanenko, Qiantong Xu, Vineel Pratap, Jacob Kahn, Ann Lee, Ronan Collobert, Gabriel Synnaeve, et al. Robust wav2vec 2.0: Analyzing domain shift in self-supervised pre-training. *arXiv preprint arXiv:2104.01027*, 2021b.

Wei-Ning Hsu, Yao-Hung Hubert Tsai, Benjamin Bolte, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: How much can a bad teacher benefit asr pre-training? In *ICASSP*, 2021c.

Eugene Kharitonov, Morgane Rivière, Gabriel Synnaeve, Lior Wolf, Pierre-Emmanuel Mazaré, Matthijs Douze, and Emmanuel Dupoux. Data augmenting contrastive learning of speech representations in the time domain. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pp. 215–222. IEEE, 2021.

Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10: 1755–1758, 2009.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. In *ACL*, 2018.

Shaoshi Ling and Yuzong Liu. Decoar 2.0: Deep contextualized acoustic representations with vector quantization. *arXiv preprint arXiv:2012.06659*, 2020.

Pingchuan Ma, Rodrigo Mira, Stavros Petridis, Björn Schuller, and Maja Pantic. LiRA: Learning visual speech representations from audio through self-supervision. In *Interspeech*, 2021a.

Pingchuan Ma, Stavros Petridis, and Maja Pantic. End-to-end audio-visual speech recognition with conformers. In *ICASSP*, 2021b.

Takaki Makino, Hank Liao, Yannis Assael, Brendan Shillingford, Basilio Garcia, Otavio Braga, and Olivier Siohan. Recurrent neural network transducer for audio-visual speech recognition. In *Interspeech*, 2019.

Brais Martinez, Pingchuan Ma, Stavros Petridis, and Maja Pantic. Lipreading using temporal convolutional networks. In *ICASSP*, 2020.

Harry McGurk and John MacDonald. Hearing lips and seeing voices. *Nature*, 264:746–748, 1976.

Andrew N. Meltzoff and M. Keith Moore. Imitation of facial and manual gestures by human neonates. *Science*, 198:75–78, 1977.

Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement. In *CVPR*, 2021.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.

Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018.

Hai Pham, Paul Liang, Thomas Manzini, Louis-Philippe Morency, and Barnabas Poczos. Found in translation: Learning robust joint representations by cyclic translations between modalities. In *AAAI*, 2019.

AJ Piergiovanni, Anelia Angelova, and Michael Ryoo. Evolving losses for unlabeled video representation learning. In *CVPR*, 2020.

Mirco Ravanelli, Jianyuan Zhong, Santiago Pascual, Pawel Swietojanski, João L. Monteiro, Jan Trmal, and Yoshua Bengio. Multi-task self-supervised learning for robust speech recognition. In *ICASSP*, 2020.

Brendan Shillingford, Yannis Assael, Matthew Hoffman, Thomas Paine, Cían Hughes, Utsav Prabhu, Hank Liao, Hasim Sak, Kanishka Rao, Lorrayne Bennett, Marie Mulville, Ben Coppin, Ben Laurie, Andrew Senior, and Nando Freitas. Large-scale visual speech recognition. In *Interspeech*, 2019.

Themos Stafylakis and Georgios Tzimiropoulos. Combining residual networks with lstms for lipreading. In *Interspeech*, 2017.

W. H. Sumby and Irwin Pollack. Visual contribution to speech intelligibility in noise. *Journal of the Acoustical Society of America*, 26:212–215, 1954.

Bo Xu, Cheng Lu, Yandong Guo, and Jacob Wang. Discriminative multi-modality speech recognition. In *CVPR*, 2020.

Shiliang Zhang, Ming Lei, Bin Ma, and Lei Xie. Robust audio-visual speech recognition using bimodal dfsmn with multi-condition training and dropout regularization. In *ICASSP*, 2019a.

Xingxuan Zhang, Feng Cheng, and Shilin Wang. Spatio-temporal fusion based convolutional sequence learning for lip reading. In *ICCV*, 2019b.

A Detailed Experimental Setup

A.1 Datasets

LRS3 (Afouras et al., 2018b) is the largest publicly available sentence-level lip reading dataset to date. It consists of over 400 hours of video, extracted from TED & TEDx talks in English from YouTube. In the original dataset, the training data is split into two partitions: pretrain (403 hours) and trainval (30 hours). Both parts are transcribed at the sentence level and come from the same source as the test set. The pretrain set differs from trainval in that the duration of its video clips are of a much wider range and can be shorter or longer than a full sentence. In the low-resource setup, we only use trainval as the labeled data. As no official development set is provided, we randomly select 1,200 sequences from trainval as the validation set (about 1 hour) for early stopping and hyperparameter tuning.

VoxCeleb2 (Chung et al., 2018) is originally created for multilingual audio-visual speaker recognition and it contains over 2,442 hours of utterances of over 6,000 speakers extracted from YouTube videos. The dataset naturally serves our purpose as it does not contain ground-truth transcriptions.

The VoxCeleb2 has substantial domain discrepency to the LRS3 data as its utterances are from multiple languages and includes videos in a larger variety of domains including interviews, talks, excerpts under indoor and outdoor environments. In VoxCeleb2, by default we only use the English portion for pre-training. As no ground-truth language label is given in the VoxCeleb2, we use a simple heuristic to choose the English samples. Specifically, we use an off-the-shelf character-based ASR model (Hsu et al., 2021a) trained on Librispeech which achieves 1.9%/3.5% WER on clean/other test set. We run greedy decoding on the VoxCeleb2 and use the proportion of valid English words to determine if a target utterance is English or not. An utterance is only selected if the proportion of valid English words is higher than 60%. The total amount of unlabeled data after filtering is 1,326 hours.

A.2 Data Preprocessing

For each video clip, we detect the 68 facial keypoints using dlib (King, 2009) and align each frame to a reference face frame via affine transformation. We crop a 96×96 region-of-interest (ROI) centered on the mouth. Each image frame is converted to grayscale. We randomly cropped 88×88 from the whole ROI and randomly flipped it horizontally with probablity of 0.5 during training. At test time, 88×88 ROI is center cropped and does not go through horizontal flipping. The preprocessing steps remain same as prior works in lip reading (Martinez et al., 2020; Ma et al., 2021b). For the associated audio, we extract the 26-dimensional log filterbank energy feature at a stride of 10 ms from the raw waveform and use it as input to the model. As the image frames are sampled at 25Hz, we stack the 4 neighboring acoustic frames to synchronize the two modalities.

A.3 AV-HuBERT Model Architecture

In the modified ResNet-18 (Ma et al., 2021b; Martinez et al., 2020; Stafylakis & Tzimiropoulos, 2017), the first convolutional layer is substituted by a 3D convolutional layer with kernel size $5 \times 7 \times 7$. The visual feature tensor is flattened into a single-dimensional vector through a 2D average pooling layer in the end. We use one linear projection layer as the audio encoding module. The acoustic features are normalized by per-frame statistics before feeding into the network (Ba et al., 2016). We use a dropout of $p = 0.1$ after the self-attention block within each transformer layer, and each transformer layer is dropped (Fan et al., 2020) at a rate of 0.1.

A.4 Training and Inference

Pretraining Our models are implemented with fairseq (Ott et al., 2019). The whole network is randomly initialized before pre-training. In pre-training, the model is trained for five iterations in total. For the initial iteration, we generate the targets by running k-means clustering algorithm on 39-dimensional MFCC feature (13 coefficients with its first- and second-order derivatives) extracted from raw audio waveform. For the subsequent iterations, the feature from an intermediate layer of the model in the previous iteration is used for clustering. The layer index (one-based) used for clustering in iteration 1-4 are $\{9, 12, 12, 12\}$. The number of features are clustered to $\{100, 100, 500, 1000, 2000\}$ respectively for the 5 iterations. See section D.3 for analysis. To save training time, we always use the BASE model to generate clusters and LARGE model is only trained in the 5th iteration.

We set both p_m and p_a to 0.5 for modality dropout at training time. Both modalities are used at test time for feature extraction. We adopt the strategy used in wav2vec 2.0 (Baevski et al., 2020) to generate masks, where $p\%$ of all frames are randomly selected as start and subsequent l frames are masked. In iteration 1-4, we mask the fused features and set p/l to be 8/10 respectively as we observe such practice generates higher quality cluster assignments (see section D.2). In the last iteration, we set p/l to be 6/5 for video and 8/10 for audio (see section C).

We train the model with Adam (Kingma & Ba, 2015), warming up the learning rate for the first 8% of updates to a peak of 0.002 and then linearly decay it. Videos are batched together to not exceed 1,000 image frames (40 seconds) per GPU. Both BASE and LARGE models are updated for 400K and 600K steps at each iteration, respectively in 433h/1759h unlabeled settings. We train on 32 and 64 V100-GPUs for BASE and LARGE. On average, each iteration takes $\sim 2.0/3.0$ days for BASE and $\sim 2.4/3.6$ days for LARGE in using 433h/1759h unlabeled data for pre-training.

Fine-tuning After pre-training, we fine-tune the AV-HuBERT model on labeled (video, text) pairs. The audio encoder is removed and its output is replaced by a zero-vector. In CTC fine-tuning, a randomly initialized projection layer is added on top of the transformer to map features into phonemes. The lexicon is constructed with CMUDict (cmu). In fine-tuning with S2S, we use a 6-layer/9-layer transformer decoder on BASE and LARGE model to decode features into unigram-based subword units (Kudo, 2018). The vocabulary size in CTC and S2S are 46 and 1000 respectively.

In CTC, the pre-trained model is updated from the initial iteration without any freezing. The model is fine-tuned for 30K/100K steps respectively in 30h/433h setting. In S2S, the pre-trained model (i.e., encoder) is frozen for the first $N\%$ updates . N is 100 and 50 for 30h and 433h labeled setting respectively. The entire model is trained for 18K/45K steps in the 30h/433h setting. Both models are trained with Adam, with the learning rate being warmed up for the first $P\%$ of updates to a peak of 0.001 and linearly decayed. P is tuned among $\{10, 30, 50\}$. All hyperparamters are tuned on the validation set.

Decoding For CTC, we use a 4-gram language model trained on text data in the LRS3 training set. The perplexity of the 4-gram LM on test set is 110.5. No language model is used for S2S decoding. For CTC, we tune the beam width among $\{5, 10, 20, 50, 100, 150\}$, the language model weight among $\{0, 1, 2, 4, 8\}$ and word insertion penalty among $\{\pm 4, \pm 2, \pm 1, 0\}$. For S2S, The beam width and length penalty are tuned among $\{5, 10, 20, 50\}$ and $\{0, \pm 1\}$. The tuning is done on the validation set.

A.5 LiRA Implementation

We re-implemented the LiRA (Ma et al., 2021a) training objective in our framework, as there does not exist publicly available implementations and we aim to focus the comparison on the pre-training objective rather than the architectural difference. We use the same backbone architecture as the BASE AV-Hubert except the output layer being a linear project layer with an output dimension of 256. The 256-dimensional frame PASE+ feature is extracted from the audio with its official implementation in (Ravanelli et al., 2020). The original PASE+ feature is downsampled to 25Hz for synchronization with the visual stream. The pre-trained model is fine-tuned in both CTC and S2S. The optimizer and learning rate schedule in pre-training, hyperparameter search in fine-tuning and decoding remain the same as AV-HUBERT. Note with our implementation, the WER is reduced by 23% ($94.3\% \rightarrow 71.9\%$) using LiRA when the percentage of labeled data is 6.9% (30h labeled, 433h in total), while Ma et al. (2021a) achieves $\sim 10\%$ improvement in a similar setting.

B ADDITIONAL LIP-READING RESULTS

B.1 AMOUNT OF LABELED DATA

Table B.1 shows the effect of pre-training on different amount of labeled data for fine-tuning. We use 433 hours of unlabeled data (LRS3 only) and randomly selected 1, 10 and 100 hours of labeled data for fine-tuning. Overall pre-training brings large and consistent gains across different amount of labeled data. Specifically CTC-based fine-tuning outperforms S2S-based fine-tuning in low-resource settings (1-hour and 10-hour). The larger number of parameters as well as the lack of a language model for decoding makes S2S model more likely to overfit especially when the amount of fine-tuning data is small.

Table B.1: WER (%) in using different amount of labeled data for fine-tuning (BASE, 433 hours unlabeled)

Labeled (hrs)	Unlabeled (hrs)	Criterion	LM	WER (%)	
				w/o pretrain	w/ pretrain
1	433	CTC	4-gram	98.6	**68.8**
		S2S	-	98.9	**92.0**
10	433	CTC	4-gram	90.8	**57.6**
		S2S	-	97.6	**63.1**
100	433	CTC	4-gram	77.8	**54.2**
		S2S	-	84.3	**48.1**

B.2 PERFORMANCE ON SEEN SPEAKERS

The current LRS3 benchmark is under the open-speaker setting, where the speaker identities in training and test set do not overlap. To test the lip reading performance for a fixed set of speakers which is the case for an early versions of LRS3 used before October 2018, we randomly choose 5 groups of utterances from the trainval partition of LRS3 as test set and repeat experiments for each group independently. Each group contains 1322 utterance, which is of the same amount as the original test set. The model we compare is the large AV-HUBERT pre-trained with 1,759 unlabeled data. As is shown in table B.2, the average WER achieved by our model for seen speakers is $18.0 \pm 0.5\%$, which is significantly lower than the WER for unseen speakers (30.5%) under the open-speaker setting.

Table B.2: WER (%) under closed-speaker setting for 5 randomly sampled test sets and their average

	Test set (seen speakers)					AVG
	1	2	3	4	5	
WER (%)	17.4	18.6	18.5	17.5	18.3	18.0 ± 0.5

B.3 FULL RESULTS WITH CTC FINE-TUNING

Table B.3 shows the full results on LRS3, which includes the CTC fine-tuning performance for all the models we implemented. In general, the conclusions we draw from S2S (e.g., the benefits of our pre-training approach in different settings, the improvement over LiRA) in section 4.2 holds for CTC as well.

Table B.3: WER (%) of our models and the comparison with prior works on LRS3-TED dataset.

Method	Backbone	Criterion	Labeled iso (hrs)	Labeled utt (hrs)	Unlabeled data (hrs)	WER (%)
Supervised						
Zhang et al. (2019b)	CNN	S2S	157	698	-	60.1
Afouras et al. (2018a)	Transformer	S2S	157	1,362	-	58.9
Xu et al. (2020)	RNN	S2S	157	433	-	57.8
Afouras et al. (2020)	CNN	CTC	157	433	334	59.8
Shillingford et al. (2019)	RNN	CTC	-	3,886	-	55.1
Ma et al. (2021b)	Conformer	CTC+S2S	157	433	-	43.3
Makino et al. (2019)	RNN	Transducer	-	31,000	-	**33.6**
Semi-Supervised & Self-Supervised						
Afouras et al. (2020)	CNN	CTC	157	433	-	68.8
Ma et al. (2021a)†	Transformer-Base	CTC	-	30	433	72.8
			-	433	1,759	58.4
		S2S	-	30	433	71.9
			-	433	1,759	49.6
Proposed (Self-Supervised)						
AV-HUBERT	Transformer-Base	CTC	-	30	-	83.7
			-	30	433	55.3
			-	30	1,759	**47.3**
			-	433	-	62.5
			-	433	433	49.3
			-	433	1,759	**43.0**
		S2S	-	30	-	94.3
			-	30	433	51.8
			-	30	1,759	**46.1**
			-	433	-	60.3
			-	433	433	44.0
			-	433	1,759	**34.8**
	Transformer-Large	CTC	-	30	-	83.1
			-	30	433	51.6
			-	30	1,759	**41.7**
			-	433	-	61.7
			-	433	433	46.2
			-	433	1,759	**38.8**
		S2S	-	30	-	92.5
			-	30	433	47.5
			-	30	1,759	**34.6**
			-	433	-	59.1
			-	433	433	41.9
			-	433	1,759	**30.5**

C ABLATION STUDIES

The ablation studies in this section are done in the last iteration of the AV-HuBERT, pre-trained with 433 hours of unlabeled data. The model is fine-tuned with 30 hours of labeled data using CTC.

Table C.1: Ablation study for hyper-parameters. The ablations are done in the last iteration of AV-HUBERT. m_a/m_v: the probability of an acoustic/image frame being masked.

Masking				Modality Dropout		Loss	WER	
Where	How	m_a	m_v	p_m	p_a	α	dev	test
Input	Sub (same, seg)	0.8	0.3	0.5	0.5	0.0	46.8	55.3
	Sub (same, frm)						47.2	55.8
	Sub (diff, seg)						47.6	56.1
	Learned Embedding						52.6	57.8
	Gauss. Noise						52.4	57.9
Feature	Learned Embedding						55.2	58.2
Input	Sub (same, seg)	0.8	0.3	0.5	0.5	0.0	46.8	55.3
		0.8	0.8				59.3	61.6
		0.3	0.3				54.9	58.2
Input	Sub (same, seg)	0.8	0.3	0.5	0.5	0.0	46.8	55.3
				1.0	n/a		55.2	57.0
Input	Sub (same, seg)	0.8	0.3	0.5	0.5	0.0	46.8	55.3
						1.0	46.7	55.7

Masking Strategy In the first part of table C.1, we compare the proposed masking strategy against several alternatives. Feature masking applies span mask at feature level and leads to the worst performance, which is due to the leakage of information to ResNet. Directly masking image sequence with random Gaussian noise or a learned embedding slightly improves the performance by preventing the prior issue. However, those artificial frames also corrupts the raw image sequence and enlarges the domain gap in videos between pre-training and fine-tuning. On the other hand, our proposed method achieves better performance. Specifically, using segments from the same utterance ("Sub, (same, seg)") as the imposter leads to the best result compared to sampling from a different utterance ("Sub, (diff, seg)") or sampling non-consecutive frames from the same utterance ("Sub, (same, frm)").[3] The filled-in fake images are visually similar to the raw images and substitution with a segment keeps the temporal smoothness making the replaced segment more realistic, which enforces the ResNet to encode more fine-grained visual details.

Masking probability We set two different masking probabilities for audio and visual stream in the last iteration. The probabilities of an acoustic frame and image frame being masked are 0.8 and 0.3 respectively. As is shown in the second part of table C.1, setting masks for audio and visual stream independently is essential because the optimal masking probability for audio and visual stream are different. Audio encoder tends to deteriorate into a simple acoustic feature extractor when mask length is small. On the other hand, long visual mask will lead to the model lacking context to distinguish between fake and real images.

Modality dropout The third part of table C.1 compares the model performance with and without modality dropout. Randomly dropping audio sequence prevents the model from over-relying on audio for masked prediction and helps the visual representation learning.

Where to compute prediction loss In the last part of table C.1, we compare the choice of masked prediction vs. prediction. The loss weight on unmasked region does not have a large impact on the fine-tuning performance. This is different from the findings in Audio HuBERT Hsu et al. (2021a), where masked prediction leads to much better performance. Given image frames as input, the prediction of cluster assignments, which are mostly determined by the accompanied audio stream, helps encode phonetic information into the visual representation. The task is much less trivial than

[3]To avoid using original frames for substitution, p_i in equation 3 is selected from $[0, 2s_i - t_i] \cup [t_i, T_f - t_i + s_i]$ if the imposter is from the same sequence.

a single-modal model (audio-HuBERT), where setting non-zero weight on unmasked prediction can easily make the model deteriorate into an acoustic feature extractor. In addition, the high quality of targets in the last iteration also makes such prediction more helpful.

D ANALYSIS ON CLUSTERING

D.1 MEASURING CLUSTERING QUALITY

For analysis, we use frame-level phonetic labels as the ground-truth and match its correlation between cluster assignments. The phonetic labels are obtained via forced alignement from a monophone based HMM-GMM ASR model trained on LRS3. In particular, we use clustering purity and Normalize Mutual Information (NMI) as the evaluation metrics. Table D.1 shows that (1) The quality of cluster assignments is consistent with fine-tuning performance across different models (2) Hand-engineered audio feature (MFCC, NMI: 21.5%) has much stronger correlation with phonetic labels than the visual feature (HoG, NMI: 1.6%) (3) Audio-visual clusters (NMI: 44.2%) are of better quality than pure audio-based clusters (NMI: 39.7%). (4) In single-modality visual Hubert (V/HoG→V), feature quality is improved negligibly through iteration training.

Table D.1: Quality of different cluster assignments. Each number is in the format of Purity (NMI). The metrics of cluster assignments and WER in last iteration of each model are in boldface.

Model	Iter	Feature	Target K	Purity (%),↑	NMI (%),↑	WER (%),↓
AV/MFCC→AV (Proposed)	1	MFCC	100	30.3	21.5	71.5
	2	AV/MFCC→AV (it1, L9)	100	47.3	37.7	63.6
	3	AV/MFCC→AV (it2, L12)	500	61.5	42.6	60.9
	4	AV/MFCC→AV (it3, L12)	1000	65.6	43.7	58.8
	5	AV/MFCC→AV (it4, L12)	2000	**68.8**	**44.2**	**58.2**
AV/MFCC→A	1	MFCC	100	30.3	21.5	71.5
	2	A/MFCC→A (it1, L9)	100	47.0	36.7	64.3
	3	A/MFCC→A (it2, L12)	500	**56.5**	**39.7**	**63.5**
V/MFCC→A	1	MFCC	100	30.3	21.5	75.4
	2	A/MFCC→A (it1, L9)	100	47.0	36.7	69.4
	3	A/MFCC→A (it2, L12)	500	**56.5**	**39.7**	**69.1**
V/MFCC→V	1	MFCC	100	30.3	21.5	75.4
	2	V/MFCC→V (it1, L9)	100	32.8	22.9	72.6
	3	V/MFCC→V (it2, L12)	500	**33.0**	**22.8**	**72.3**
V/HoG→V	1	HoG	100	16.4	1.6	80.3
	2	V/HoG→V (it1, L9)	100	**16.4**	**1.8**	**80.1**

D.2 FEATURE MASKING PRODUCES BETTER FEATURES FOR CLUSTERING

In iteration 1-4, we apply the mask in the fused feature. We observe such practice generates targets of higher quality, thus helping future iterations more. Table D.2 shows a comparison between such two different masking strategies. Input-level masking enhances the learning of visual representation (see table C.1) while produces worse audio-visual feature (NMI: 27.2%). In contrast, the two streams of original input are better aligned in feature-level masking which is consistent with the cluster generation process, thus leading to better audio-visual clusters (NMI: 37.7%).

Table D.2: Impact of masking strategy on quality of cluster assignments (purity/NMI: quality of cluster assignments used to train the model)

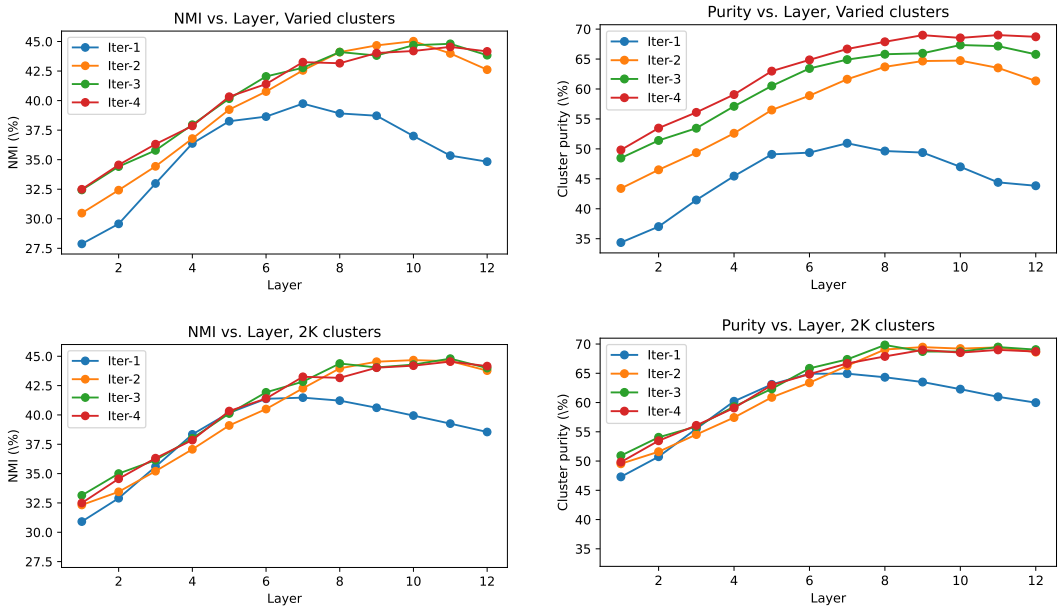
Feature	K	Purity (%)	NMI (%)
MFCC	100	30.3	21.5
AV/MFCC→AV (it1, L9) w/ Feature Masking	100	47.3	37.7
AV/MFCC→AV (it1, L9) w/ Input Masking	100	34.5	27.2

D.3 CLUSTERING QUALITY ACROSS LAYERS

Figure D.1 shows the clustering quality of features of different layers in different iterations. The cluster assignment quality generally improves with more iterations. In the first iteration, features in

the middle layers show higher quality than the other layers. The target for the first iteration (MFCC clusters) is of worse quality, thus later layers that are more correlated with targets do not yield the best cluster. Target quality improves with more training iterations, thus the best feature layer shifts towards the end. Setting a larger number of clusters increases the clustering quality as can be seen from the comparison between "varied clusters" and "2K clusters". In terms of the 12th layer which we choose, the highest NMI (44.2%) is achieved in the last iteration. In addition, more iterations of training improves the overall quality of clusters produced by a model though the highest NMI/purity among all layers does not necessarily increase in later iterations. Therefore, setting a larger number of iterations brings stable gains which are more robust to the index of layer chosen for clustering. It is important as the purity/NMI, whose measurement rely on a supervised model, are not used for hyperparameter search in practice.

Figure D.1: Quality of feature clusters from different layers across different iterations (BASE, 433 hours unlabeled data). (Iter i, Layer j): cluster quality of layer-j feature of iter-i model. Upper row: 100, 500, 1K, 2K clusters for 4 iterations. Bottom row: 2K clusters for all iterations. Purity/NMI of the initial MFCC clusters: 30.3%/21.5%



E QUALITATIVE EXAMPLES

Figure E.2 shows the example outputs from different models. Our self-supervised model is the LARGE AV-HUBERT pre-trained with 1,759 hours unlabeled data and fine-tuned with 433 hours labeled data. The BASE model is the supervised baseline trained with 433 hours labeled data. Both models use the S2S criterion for supervised training and have the same number of parameters. Qualitatively, our approach provides transcriptions with much higher quality. The baseline approach confuses among words of similar sound while our model output more semantically sound sentences. Figure E.2 also shows typical errors made by our model. We noticed many errors are on short sentences. This is mainly because lip reading relies heavily on the context for recognition due to the existence of homophomes. Thus the error rates in lip reading are notably higher in short utterances, which differs from ASR, as can be seen from figure E.1. Substitution among words with homophemes ('fiction' vs. 'vision' in a.4, 'part' vs. 'bunk' in b.5) is another source of error made by the model.

Figure E.1: WER vs. sentence length for lip reading (left) and ASR (right)

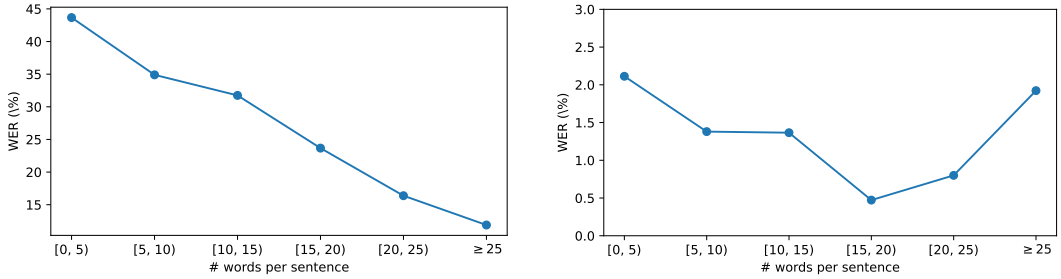
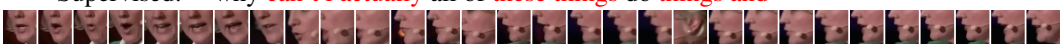


Figure E.2: Transcriptions from different lip-reading models. GT: ground-truth, Proposed: self-supervised model, Supervised: supervised model. Red: wrong words in the output

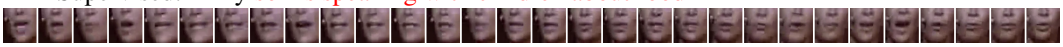
(a) Self-supervised vs. Supervised

(1)	GT:	why not ask all of the states to do that instead
	Proposed:	why not ask all of these things to do that instead
	Supervised:	why can't i actually all of these things do things and

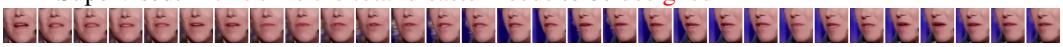


(2)	GT:	indeed we run the risk of making things worse
	Proposed:	indeed we want the risk of making things worse
	Supervised:	in india we roughly receive money in the health world

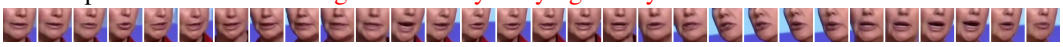
(3)	GT:	my desire to disappear was still very powerful
	Proposed:	my desire to disappear was still very powerful
	Supervised:	my son is speaking with children about food



(4)	GT:	the silent majority does not need to be silent
	Proposed:	the same majority does not need to be silent
	Supervised:	this time the total disaster needs to be designed

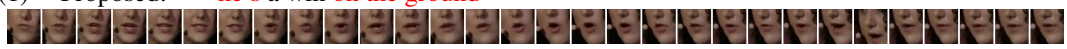


(5)	GT:	mortality is not going down it's going up
	Proposed:	mortality is not going down it's going up
	Supervised:	we're seeing this not only carrying slowly how

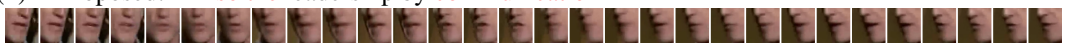


(b) Failure cases

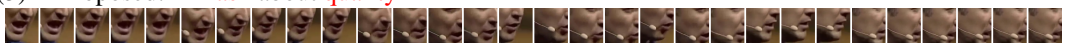
(1)	GT:	it's a win all around
	Proposed:	he's a win on the ground



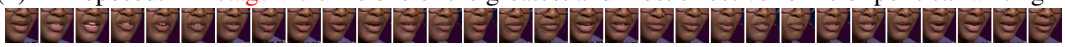
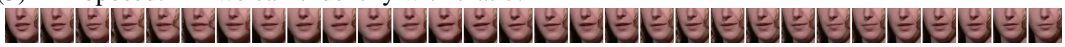
(2)	GT:	sort of leadership by humiliation
	Proposed:	so the leadership by communication



(3)	GT:	is it about equality
	Proposed:	ask about quality



(4)	GT:	science fiction is one of the greatest and most effective forms of political writing
	Proposed:	stage vision is one of the greatest and most effective forms of political writing



(5)	GT:	we can't identify with that part
	Proposed:	we can't identify with that bunk