Hearing Lips in Noise: Universal Viseme-Phoneme Mapping and Transfer for Robust Audio-Visual Speech Recognition

Yuchen Hu[1], **Ruizhe Li**[2], **Chen Chen**[1], **Chengwei Qin**[1], **Qiushi Zhu**[3], **Eng Siong Chng**[1]
[1]Nanyang Technological University, Singapore [2]University of Aberdeen, UK
[3]University of Science and Technology of China, China
{yuchen005@e., chen1436@e., chengwei003@e., aseschng@}ntu.edu.sg,
ruizhe.li@abdn.ac.uk, qszhu@mail.ustc.edu.cn

Abstract

Audio-visual speech recognition (AVSR) provides a promising solution to ameliorate the noise-robustness of audio-only speech recognition with visual information. However, most existing efforts still focus on audio modality to improve robustness considering its dominance in AVSR task, with noise adaptation techniques such as front-end denoise processing. Though effective, these methods are usually faced with two practical challenges: 1) lack of sufficient labeled noisy audio-visual training data in some real-world scenarios and 2) less optimal model generality to unseen testing noises. In this work, we investigate the noise-invariant visual modality to strengthen robustness of AVSR, which can adapt to any testing noises while without dependence on noisy training data, *a.k.a.*, unsupervised noise adaptation. Inspired by human perception mechanism, we propose a universal viseme-phoneme mapping (UniVPM) approach to implement modality transfer, which can restore clean audio from visual signals to enable speech recognition under any noisy conditions. Extensive experiments on public benchmarks LRS3 and LRS2 show that our approach achieves the state-of-the-art under various noisy as well as clean conditions. In addition, we also outperform previous state-of-the-arts on visual speech recognition task[1].

1 Introduction

The world surrounding us involves multiple modalities, including vision, audio, text, etc., which complement each other and jointly comprise human perception (Baltrušaitis et al., 2018; Zhu et al., 2021b). Audio-visual speech recognition (AVSR) leverages both audio and visual modalities to understand human speech, which provides a promising solution to ameliorate the noise-robustness of audio-only speech recognition with noise-invariant lip movement information (Sumby and Pollack, 1954).

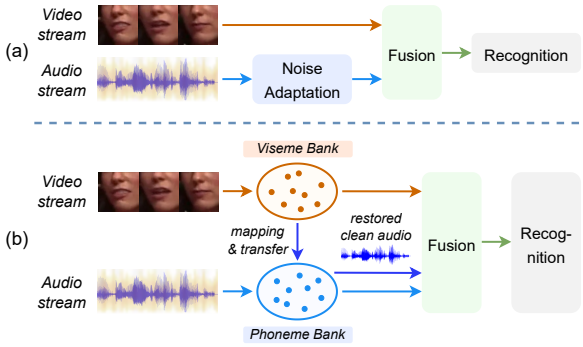


Figure 1: Illustration of noisy audio-visual speech recognition. (a) Mainstream AVSR approaches with noise adaptation. (b) Our framework constructs viseme-phoneme mapping for modality transfer, which restores clean audio from visual signals to enable speech recognition under any noisy conditions.

However, most existing efforts still focus on audio modality to improve noise-robustness considering its dominance in AVSR, where audio modality contains much richer information to represent speech content than visual modality (Sataloff, 1992; Ren et al., 2021). Current mainstream approaches introduce noise adaptation techniques to improve robustness[2], inspired by robust speech recognition (Wang et al., 2020). Most of them leverage noise-corrupted training data to strengthen robustness (Afouras et al., 2018a; Ma et al., 2021b; Song et al., 2022), and recent works extend it to self-supervised learning scheme (Shi et al., 2022b; Hsu and Shi, 2022). Based on that, latest works introduce speech enhancement as front-end to denoise before recognition (Xu et al., 2020; Hong et al., 2022). Despite the effectiveness, these methods are usually faced with two practical challenges. First, they require abundant labeled noisy audio-visual data for network training, which is not always available in some real-world scenarios (Lin et al., 2021; Chen et al., 2022). Second, the well-trained model may not adapt to new-coming noise scenes in practical applications[2], resulting in less optimal model

[1]Code is available at https://github.com/YUCHEN005/UniVPM.

[2]Experimental analysis are in §A.1 and §4.2.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 15213–15232

generality (Meng et al., 2017). Therefore, our research idea in this paper is leveraging visual modality to develop a general noise-robust AVSR system while without dependence on noisy training data.

We may gain some inspirations from human perception mechanism of noisy audio-visual speech. Neuroscience studies (Nath and Beauchamp, 2011) find that human brain will unconsciously rely more on the lip movement to understand speech under noisy conditions (*a.k.a.*, McGurk Effect, McGurk and MacDonald, 1976). During this process, instead of directly recognizing lip movement, human brain will first transfer it to speech signal in auditory cortex for further understanding (Bourguignon et al., 2020; Mégevand et al., 2020). With prior knowledge of lip-audio mapping, human brain can restore informative clean audio from lip movement under any noisy conditions to aid in speech understanding (Bernstein et al., 2004; Aller et al., 2022).

Motivated by above observations, we propose a universal viseme-phoneme[3] mapping approach (UniVPM) to implement modality transfer, which can restore clean audio from lip movement to enable speech recognition under any noisy conditions. We first build two universal memory banks to model all the visemes and phonemes via online balanced clustering. Based on that, an adversarial mutual information estimator is proposed to construct strong viseme-phoneme mapping, which enables final lip-to-audio modality transfer via retrieval. As a result, our system can adapt well to any testing noises while without noisy training data. Empirical results show the effectiveness of our approach. Our contributions are summarized as:

- We present UniVPM, a general noise-robust AVSR approach investigated on visual modality, which can adapt to any testing noises while without dependence on noisy training data, *a.k.a.*, unsupervised noise adaptation.
- We build two universal banks to model all the visemes and phonemes via online balanced clustering, followed by an adversarial mutual information estimator to construct strong mapping between them, which enables modality transfer to restore clean audio from lip movement for speech recognition under any noises.
- Our UniVPM outperforms previous state-of-the-arts on LRS3 and LRS2 benchmarks. Extensive experiments also show its superiority on visual speech recognition (VSR) task.

[3]Phoneme is the phonetic base unit (from clean audio), and viseme is the visual equivalent of phoneme.

2 Related Work

Audio-Visual Speech Recognition. AVSR provides a promising solution to noise-robust speech recognition with the noise-invariant visual modality (Afouras et al., 2018a). However, most existing efforts still focus on audio modality to improve robustness considering its dominance in AVSR task (Sataloff, 1992; Ren et al., 2021). Mainstream approaches introduce noise adaptation techniques to strengthen robustness, where most of them leverage noise-corrupted data to improve network training (Afouras et al., 2018a; Ma et al., 2021b; Pan et al., 2022; Shi et al., 2022b; Hsu and Shi, 2022), and recent works further introduce speech enhancement as front-end to denoise before recognition (Xu et al., 2020; Hong et al., 2022). Despite the effectiveness, these methods require abundant labeled noisy audio-visual training data that is not always available in some real scenarios, and they may not adapt to the new-coming noise scenes in practical applications. In this work, we investigate the visual modality to develop a general noise-robust AVSR approach while without dependence on noisy training data, *a.k.a.*, unsupervised noise adaptation.

Memory Network. Memory network (Weston et al., 2014) presents a long-term memory component that can be read from and written in with inference capability. Miller et al. (2016) introduces key-value memory structure where key memory is used to address a query and the retrieved output is obtained from value memory using the address. Since this scheme can remember selected information, it is effective for augmenting features in many tasks, including video prediction (Lee et al., 2021), cross-modal retrieval (Song et al., 2018; Chen et al., 2020a), lip reading (Kim et al., 2021a, 2022) and talking face generation (Park et al., 2022). Despite the advances, the memory network is prone to overfitting when handling imbalanced distributed data, *a.k.a.*, long tail[4] (Liu et al., 2019), which may fail to model the minority classes well. In this work, we propose to build two memory banks via online balanced clustering to model all the visemes and phonemes equally, *i.e.*, universal.

Viseme-Phoneme Mapping. Viseme-phoneme mapping is important to many visual-audio learning tasks, including speech recognition (Chan et al.,

[4]Phoneme distribution in English is a long tail, see §A.4.

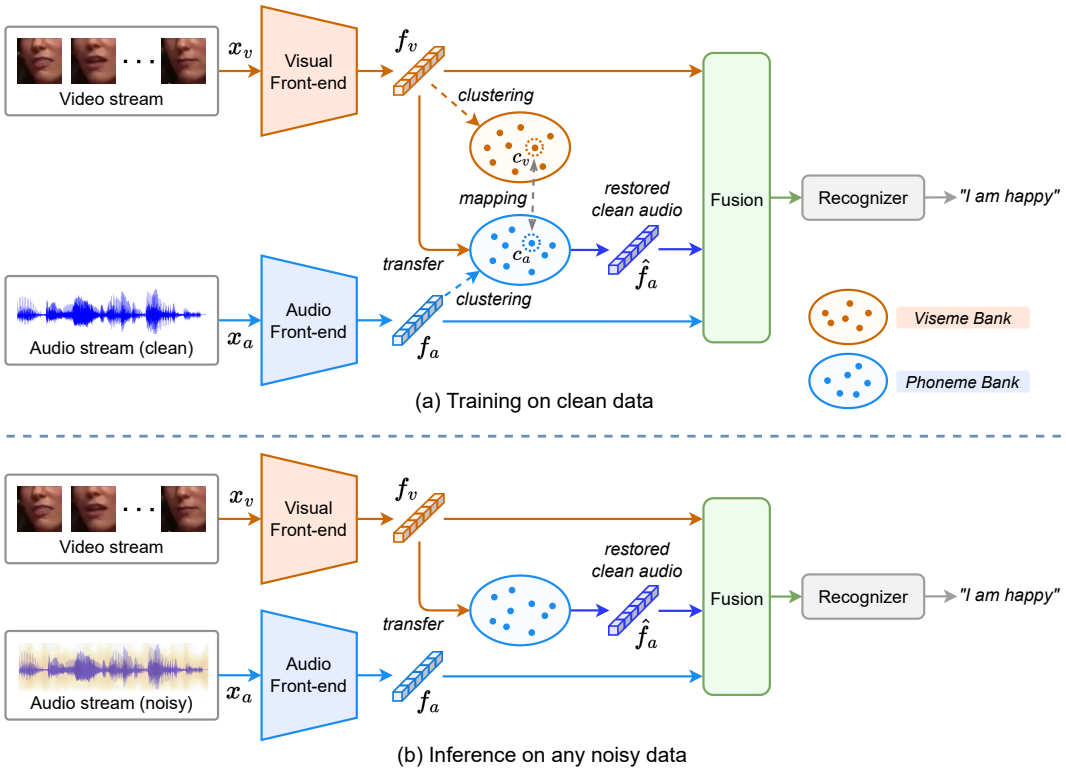


Figure 2: Illustration of our proposed UniVPM. (a) Training on clean audio-visual data to construct universal viseme-phoneme mapping. (b) Inference on any noisy data with restored clean audio from modality transfer.

2022), lip reading (Ren et al., 2021) and lip-to-speech synthesis (Prajwal et al., 2020). Among them, cross-modal distillation is a popular technique to transfer knowledge from viseme to phoneme (Afouras et al., 2020; Zhao et al., 2020; Ren et al., 2021). Other works design specific neural networks to learn their mapping (Qu et al., 2019; Kim et al., 2021b). Recent studies introduce self-supervised learning to capture correlations between visemes and phonemes (Qu et al., 2021; Ma et al., 2021a). Though effective, these methods are often challenged by the ambiguity of homophenes (Bear and Harvey, 2017) where one lip shape can produce different sounds. To this end, we propose an adversarial mutual information estimator to construct strict viseme-phoneme mapping with the strong distinguishing ability of adversarial learning.

3 Methodology

3.1 Overview

The overall framework of proposed UniVPM is illustrated in Fig. 2. During training, we first send the input video and clean audio streams into two front-ends for processing, which generates modality sequences $f_v, f_a \in \mathbb{R}^{T\times D}$, where T is number of frames and D is embedding dimension. These frames are sent into two memory banks to model all the visemes and phonemes, using an online balanced clustering algorithm where each cluster center represents a specific viseme or phoneme. Then, we propose an adversarial mutual information estimator to construct strong mapping between corresponding visemes and phonemes. Based on that, we finally implement modality transfer via retrieval to restore clean audio from visual signals, which enables speech recognition under any testing noises.

3.2 Online Balanced Clustering

Clustering is a widely used knowledge discovery technique to partition a set of data points into homogeneous groups, which has a variety of applications such as data mining (Fayyad et al., 1996). Among them, K-Means algorithm (MacQueen, 1967) is the most well-known and popular one. However, it cannot be directly applied for our viseme and phoneme clustering due to imbalanced data distribution (see §A.4). This may challenge K-Means clustering according to uniform effect (Xiong et al., 2006). As shown in Fig. 3 (a), most cluster centers gather in the majority data class (*i.e.*, over-fitting), leaving the minority class not well modeled.

To this end, we propose an Online Balanced Clustering algorithm in Alg. 1 to model all the visemes and phonemes equally from input frames.

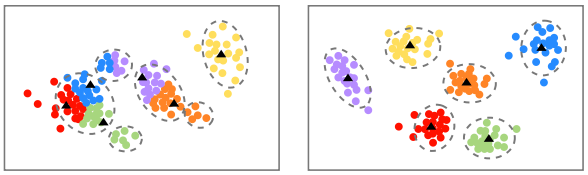
Algorithm 1 Online Balanced Clustering.

Require: Streaming data D, number of clusters N, maximum cluster size S_{max}.

1: Initialize an empty memory bank $\mathcal{B}$ and a list of empty cluster banks $\{\mathcal{B}_1, \mathcal{B}_2, ..., \mathcal{B}_N\}$.
2: **while** $len(\mathcal{B}) \leq N$ **do**
3: Receive new batch data d from D
4: Append all frame samples in d to bank $\mathcal{B}$
5: **end while**
6: Initialize a list of cluster centers $\{c_1, c_2, ..., c_N\}$ from $\mathcal{B}$ using K-MEANS++ Algorithm (2006)
7: **for** batch data $d \in D$ **do**
8: Append all frame samples in d to bank $\mathcal{B}$
9: $\{\mathcal{B}_1, ..., \mathcal{B}_N\} = \text{RE-ALLOCATE}(\mathcal{B}, \{c_1, ..., c_N\})$
10: $\{c_1, ..., c_N\} = \text{RENEW-CENTERS}(\{\mathcal{B}_1, ..., \mathcal{B}_N\})$
11: Calculate average cluster size $S_{avg} = len(\mathcal{B})/N$
12: Threshold cluster size $S_{thr} = \min(S_{avg}, S_{max})$
13: **for** $i = 1, 2, ..., N$ **do**
14: **if** $len(\mathcal{B}_i) > S_{thr}$ **then** ▷ UNDERSAMPLING
15: Maintain the S_{thr}-nearest samples to c_i in $\mathcal{B}_i$
16: Update $\mathcal{B}$ accordingly
17: **else** ▷ OVERSAMPLING
18: Set a random weight $\alpha \in (0, 1)$
19: Find the nearest sample d_{near} to c_i in $\mathcal{B}_i$
20: $d_{new} = d_{near} \cdot \alpha + c_i \cdot (1 - \alpha)$
21: $\mathcal{B}_i.append(d_{new})$
22: Update $\mathcal{B}$ accordingly
23: **end if**
24: **end for**
25: **end for**

First, we set the number of clusters N to 40, following the amount of English phonemes (Phy, 2022). Then, we set a maximum cluster size S_{max} (*i.e.*, number of samples in each cluster) to control the total memory. We also initialize an empty bank $\mathcal{B}$ as an overall cache, as well as a list of empty banks $\{\mathcal{B}_1, \mathcal{B}_2, ..., \mathcal{B}_N\}$ to cache each cluster.

The proposed algorithm is executed in three steps, center initialization, K-Means clustering and re-sampling. First, we collect the first few batches of data frames into $\mathcal{B}$ to initialize N dispersed cluster centers $\{c_1, c_2, ..., c_N\}$, using K-Means++ algorithm (Arthur and Vassilvitskii, 2006). Second, we add the current batch data to bank $\mathcal{B}$ and employ vanilla K-Means algorithm to re-allocate each sample in the bank to the nearest cluster center, after that the new cluster centers would be updated. Finally, we propose a re-sampling strategy to balance the size of different clusters as well as control the total memory of bank $\mathcal{B}$, by setting a threshold cluster size S_{thr} (line 12 in Alg. 1). For those clusters with more than S_{thr} samples (*i.e.*, majority cluster), we perform undersampling by only maintaining the S_{thr} nearest samples to cluster center. In contrast, for the minority clusters with less samples than threshold, we propose oversampling to interpolate a new sample between center and the nearest sample with a random weight, inspired by SMOTE algorithm (Chawla et al., 2002). In this way, as illustrated in Fig. 3 (b), the resulted clusters would be balanced-sized and separated to better represent each of the visemes and phonemes.



(a) Online clustering (b) Online balanced clustering

Figure 3: t-SNE visualization of clustered phonemes from (a) online clustering (with random pruning to keep fixed cluster size, details are in §C.3), and (b) our proposed online balanced clustering. We randomly select six clusters for visualization, and black triangle denotes the cluster center. Dashed ellipses highlight the real phoneme classes, which are confirmed by pre-trained phoneme recognition model (Phy, 2022).

3.3 Adversarial Mutual Information Estimator

After clustering visemes and phonemes in banks, we propose an Adversarial Mutual Information Estimator (AMIE) to construct strong mapping between them. Mutual Information (MI) is a commonly used measure to explore the coherence between two distributions, which is, however, historically difficult to estimate. Recently, Belghazi et al. (2018) propose a Mutual Information Neural Estimation (MINE) approach to approximate MI lower bound with neural network. Based on that, we propose an adversarial learning approach to maximize the MI between visemes and phonemes, in order to construct strict mapping between them and thus alleviate the ambiguity of homophenes.

3.3.1 Preliminary Theory of MINE

Mutual information measures the mutual dependency between two probability distributions,

$$I(X, Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}, \quad (1)$$

where $p(x, y)$ is the joint probability distribution of X and Y, and $p(x)$ and $p(y)$ are the marginals.

Therefore, the mutual information can be written in terms of Kullback-Leibler (KL-) divergence:

$$I(X, Y) = D_{KL}(p(x, y) \parallel p(x)p(y)), \quad (2)$$

where D_{KL} is defined as:

$$D_{KL}(p \parallel q) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)}, \quad (3)$$

Furthermore, the KL-divergence admits the Donsker-Varadhan (DV) representation (Donsker and Varadhan, 1983; Belghazi et al., 2018):

$$D_{KL}(p \parallel q) = \sup_{T:\Omega \to \mathbb{R}} \mathbb{E}_p[T] - \log(\mathbb{E}_q[e^T]), \tag{4}$$

where the supremum is taken over all functions T on $\Omega \subset \mathbb{R}^d$ to guarantee two finite expectations. Therefore, we have the MI lower bound:

$$I(X, Y) \geq I_\Theta(X, Y), \tag{5}$$

where I_Θ is the neural information measure,

$$\begin{aligned} I_\Theta(X, Y) = &\sup_{\theta \in \Theta} \mathbb{E}_{p(x,y)}[T_\theta(x, y)] \\ &- \log(\mathbb{E}_{p(x)p(y)}[e^{T_\theta(x,y)}]), \end{aligned} \tag{6}$$

and T_θ denotes a trainable neural network.

3.3.2 Proposed AMIE

Based on MINE, we propose an Adversarial Mutual Information Estimator to explore and maximize the mutual information between clustered visemes and phonemes. As illustrated in Fig. 2 and 4, given a visual sequence f_v, we send each frame of it into viseme bank to find the nearest cluster center c_v, which forms the viseme sequence $s_v \in \mathbb{R}^{T \times D}$. Similarly, we obtain a phoneme sequence s_a to represent audio features f_a. The neural network T_θ then feeds $\{s_v, s_a\}$ to output a scalar for MI estimation, where T_θ is a 3-layer classifier with output as a 1-dimensional scalar. Furthermore, since we do not concern the accurate value of MI when maximizing it, we employ Jensen-Shannon (JS) representation (Hjelm et al., 2018) to approximate KL-divergence in Eq. 4, which has been proved with more stable neural network optimization. Therefore, the mutual information between clustered visemes and phonemes is estimated as:

$$\begin{aligned} I_\Theta^{JS}(s_v, s_a) = &\sup_{\theta \in \Theta} \mathbb{E}_{p(s_v,s_a)}[-\mathrm{sp}(-T_\theta(s_v, s_a))] \\ &- \mathbb{E}_{p(s_v)p(s_a)}[\mathrm{sp}(T_\theta(s_v, \tilde{s}_a))], \end{aligned} \tag{7}$$

where $\tilde{s}_a$ is the shuffle-ordered version of s_a that subjects to the marginal distributions of phonemes, and $\mathrm{sp}(z) = \log(1 + e^z)$ is the softplus function.

As stated in Belghazi et al. (2018), the neural network T_θ can be used to estimate MI between generated data (s_v, s_a in our case) by directly trained on them. However, this will suffer a lot from the poor quality of generated data at early training stage. One feasible scheme (Zhu et al., 2021a) is to train

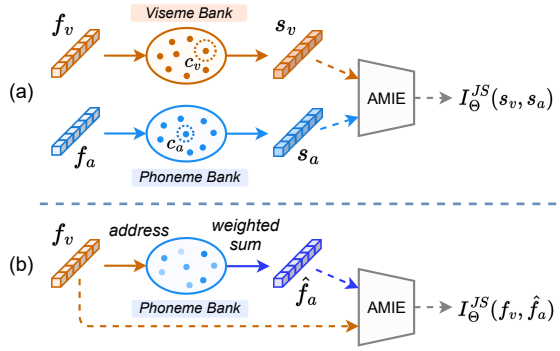


Figure 4: Illustration of (a) viseme-phoneme mapping via AMIE, and (b) modality transfer via retrieval.

T_θ on real data (f_v, f_a in our case) and then estimate MI on generated data, but this suffers from the ambiguity of homophenes (see Fig. 8). To this end, we propose AMIE with adversarial learning to estimate and maximize the MI between corresponding visemes and phonemes, which can construct strict viseme-phoneme mapping without ambiguity.

Inspired by GAN (Goodfellow et al., 2014), we design the AMIE as discriminator and the viseme-phoneme banks as generator. Based on that, the adversarial loss is defined as:

$$\begin{aligned} \mathcal{L}_{GAN} &= \mathcal{L}_D + \mathcal{L}_G \\ &= I_\Theta^{JS}(f_v, f_a) + [-I_\Theta^{JS}(s_v, s_a)], \end{aligned} \tag{8}$$

Our framework employs an adversarial learning strategy for optimization, where D and G play a two-player minimax game as detailed in Alg. 2. As a result, the estimated MI between corresponding visemes and phonemes would be maximized to construct mapping relationships. The strong distinguishing ability of adversarial learning enables strict viseme-phoneme mapping to overcome the ambiguity of homophenes, as shown in Fig. 5.

3.4 Modality Transfer

With constructed viseme-phoneme mapping, we can finally implement modality transfer to restore clean audio from lips. As shown in Fig. 4, given the visual sequence f_v and clustered phoneme centers $\{c_a^1, c_a^2, ..., c_a^N\}$, we calculate an addressing score $\mathcal{A}^{i,j}$ to indicate the probability that the i-th visual frame corresponds to the j-th phoneme cluster:

$$\mathcal{A}^{i,j} = \frac{\exp(\langle f_v^i, c_a^j \rangle / \tau)}{\sum_{k=1}^{N} \exp(\langle f_v^i, c_a^k \rangle / \tau)}, \tag{9}$$

where $\langle \cdot, \cdot \rangle$ denotes cosine similarity, τ is temperature weight. The restored clean audio frames are:

$$\hat{f}_a^i = \sum_{j=1}^{N} \mathcal{A}^{i,j} \cdot c_a^j, \tag{10}$$

Method	PT Type	FT Type	Babble, SNR (dB) = -10	-5	0	5	10	avg	Speech, SNR (dB) = -10	-5	0	5	10	avg	Music + Natural, SNR (dB) = -10	-5	0	5	10	avg	Clean ∞
RNN-T (2019)	-	Clean	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	4.5
Hyb-AVSR (2021b)	-	Noisy	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2.3
TM-seq2seq (2018a)	-	Noisy	-	-	42.5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	7.2
EG-seq2seq (2020)	-	Noisy	38.6	31.1	25.5	24.3	20.7	28.0	-	-	-	-	-	-	-	-	-	-	-	-	6.8
u-HuBERT (2022)	Noisy	Noisy	-	-	4.1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1.2
AV-HuBERT (2022b)	Clean	Clean	72.6	30.9	9.8	2.9	2.1	23.7	93.4	71.6	22.1	6.1	2.7	39.2	24.1	10.9	3.6	2.4	1.9	8.6	1.42
		Noisy	30.0	15.2	5.9	2.7	1.9	11.1	15.9	7.5	3.9	2.4	1.9	6.3	12.1	5.9	3.1	2.2	1.8	5.0	1.40
	Noisy	Clean	39.4	14.5	5.2	2.7	2.0	12.8	18.8	5.1	3.1	2.3	1.9	6.2	11.4	5.0	2.8	2.2	1.8	4.6	1.54
		Noisy	28.4	13.4	5.0	2.6	1.9	10.3	11.4	4.6	2.9	2.2	1.8	4.6	9.7	4.7	2.5	1.9	1.8	4.1	1.40
UniVPM (ours)	Clean	Clean	37.5	17.1	6.9	2.6	1.9	13.2	20.4	9.6	4.9	3.6	2.3	8.2	14.2	6.8	3.1	2.1	1.8	5.6	1.31
		Noisy	28.1	13.8	5.1	2.2	1.7	10.2	14.5	6.7	3.3	2.1	1.7	5.7	10.7	5.2	2.7	1.9	1.6	4.4	1.22
	Noisy	Clean	32.6	12.6	4.4	2.3	1.7	10.7	17.0	4.4	2.7	2.1	1.6	5.6	10.1	4.3	2.4	1.9	1.6	4.1	1.25
		Noisy	**26.8**	**12.1**	**4.0**	**2.1**	**1.6**	**9.3**	**10.4**	**4.1**	**2.5**	**2.0**	**1.6**	**4.1**	**8.7**	**4.1**	**2.1**	**1.7**	**1.5**	**3.6**	**1.18**

Table 1: WER (%) of proposed UniVPM and prior works on LRS3 benchmark. "PT Type" / "FT Type" denote pre-training / finetuning data type. "SNR" is signal-to-noise ratio. All the noisy data contains MUSAN (2015) noise.

Method	PT Type	FT Type	Babble, SNR (dB) = -10	-5	0	5	10	avg	Speech, SNR (dB) = -10	-5	0	5	10	avg	Music + Natural, SNR (dB) = -10	-5	0	5	10	avg	Clean ∞
TM-seq2seq (2018a)	-	Noisy	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	8.5
Hyb-RNN (2018)	-	Noisy	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	7.0
LF-MMI TDNN (2020)	-	Clean	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5.9
Hyb-AVSR (2021b)	-	Noisy	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	3.7
MoCo+w2v2 (2022)	-	Noisy	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2.6
AV-HuBERT (2022b)	Clean	Clean	65.2	33.6	10.9	5.6	3.8	23.8	88.2	57.8	20.6	7.5	4.0	35.6	27.3	13.3	6.7	4.0	3.4	10.9	2.57
		Noisy	33.2	16.3	7.6	4.6	3.7	13.1	14.9	9.5	6.2	4.5	3.8	7.8	13.9	9.0	4.9	3.9	3.2	7.0	2.38
	Noisy	Clean	36.9	18.6	8.1	4.8	3.5	14.4	24.6	9.7	4.8	3.6	3.4	9.2	15.2	8.4	5.1	3.8	3.1	7.1	2.44
		Noisy	32.7	14.9	6.4	4.5	3.4	12.4	9.0	5.9	3.9	3.5	3.0	5.1	12.5	6.0	4.4	3.5	3.0	5.9	2.33
UniVPM (ours)	Clean	Clean	38.3	19.0	9.2	5.0	3.5	15.0	21.1	12.2	7.8	5.4	3.9	10.1	16.3	10.4	5.6	3.6	3.2	7.8	2.30
		Noisy	30.4	14.4	6.6	**4.1**	3.4	11.8	12.4	8.3	5.5	4.2	3.6	6.8	12.4	7.9	4.3	3.4	3.0	6.2	2.17
	Noisy	Clean	33.7	16.2	6.7	4.2	**3.2**	12.8	19.8	7.6	4.0	3.2	3.1	7.5	13.4	7.3	4.5	3.4	2.9	6.3	2.24
		Noisy	**30.1**	**13.7**	**5.7**	**4.1**	**3.2**	**11.4**	**7.5**	**5.1**	**3.4**	**3.1**	**2.8**	**4.4**	**10.9**	**5.0**	**3.8**	**3.1**	**2.8**	**5.1**	**2.16**

Table 2: WER (%) of proposed UniVPM and prior works on LRS2 benchmark.

To supervise the quality of restored audio $\hat{f}_a = \{\hat{f}_a^i\}_{i=1}^T$, we first employ AMIE to maximize the MI between $\hat{f}_a$ and f_v, where Eq. 8 is rewritten as:

$$\mathcal{L}_{GAN} = I_\Theta^{JS}(f_v, f_a) + [-I_\Theta^{JS}(s_v, s_a) - I_\Theta^{JS}(f_v, \hat{f}_a)], \tag{11}$$

along with a reconstruction loss $\mathcal{L}_{rec} = \|\hat{f}_a - f_a\|_2$ to enable restoration of high-quality clean audio.

3.5 Optimization

The UniVPM is optimized in an end-to-end manner (see Alg. 2), with the final training objective as:

$$\mathcal{L} = \mathcal{L}_{ASR} + \lambda_{GAN} \cdot \mathcal{L}_{GAN} + \lambda_{rec} \cdot \mathcal{L}_{rec} + \lambda_{var} \cdot \mathcal{L}_{var}, \tag{12}$$

where $\mathcal{L}_{ASR}$ denotes the downstream speech recognition loss. $\mathcal{L}_{var}$ is a variance regularization term to disperse the clustered viseme and phoneme centers, which aims to ease their mapping construction. λ_{GAN}, λ_{rec} and λ_{var} are weighting parameters.

4 Experiments

4.1 Experimental Setup

Datasets. Our experiments are conducted on two large-scale public datasets, LRS3 (Afouras et al., 2018b) and LRS2 (Chung et al., 2017). LRS3 dataset collects 433 hours of transcribed English videos from TED & TEDx talks. LRS2 contains 224 hours of video speech from BBC programs.

Configurations and Baselines. The proposed UniVPM is implemented based on AV-HuBERT with similar configurations, which are detailed in §B.3. We also select some mainstream AVSR approaches as baselines for comparison, *e.g.*, u-HuBERT (Hsu and Shi, 2022), and details are presented in §B.7.

4.2 Main Results

Audio-Visual Speech Recognition. Table 1 compares the AVSR performance of our UniVPM with prior methods on LRS3 benchmark. Under clean training data, the proposed UniVPM (in purple shades) significantly outperforms AV-HuBERT baseline, and it achieves comparable performance to the AV-HuBERT trained on noisy data, where the restored clean audio plays the key role and implements our original motivation of unsupervised noise adaptation. Based on that, available noisy training data further improves the robustness[5], where our best results achieve new state-of-

[5]Noisy training pipeline of UniVPM is shown in Fig. 9.

Method	PT Type	FT Type	Meeting, SNR (dB) = -10	-5	0	5	avg	Cafe, SNR (dB) = -10	-5	0	5	avg	Resto, SNR (dB) = -10	-5	0	5	avg	Station, SNR (dB) = -10	-5	0	5	avg
Finetuned on DEMAND Noise																						
AV-HuBERT (2022b)	Clean	Clean	33.2	11.7	4.3	3.1	13.1	26.0	8.5	2.9	2.0	9.9	63.5	30.4	11.0	3.9	27.2	20.1	7.0	4.7	2.5	8.6
		Noisy	10.6	5.2	2.9	2.5	5.3	10.1	4.3	2.3	1.8	4.6	27.8	14.4	4.9	2.6	12.4	7.6	4.5	2.9	2.0	4.3
	Noisy	Clean	17.7	7.1	4.0	2.9	7.9	16.0	5.8	2.7	1.9	6.6	49.5	19.5	6.2	3.1	19.6	11.8	5.9	3.7	2.2	5.9
		Noisy	10.2	4.8	2.7	2.4	5.0	9.4	4.0	2.2	1.8	4.4	23.5	13.2	4.4	**2.4**	10.9	7.2	**4.3**	2.9	1.8	4.1
Finetuned on MUSAN Noise																						
AV-HuBERT (2022b)	Clean	Clean	33.2	11.7	4.3	3.1	13.1	26.0	8.5	2.9	2.0	9.9	63.5	30.4	11.0	3.9	27.2	20.1	7.0	4.7	2.5	8.6
		Noisy	13.9	6.3	3.3	2.8	6.6	13.6	5.1	2.6	1.9	5.8	36.1	17.5	5.7	2.9	15.6	9.9	5.3	3.5	2.1	5.2
	Noisy	Clean	17.7	7.1	4.0	2.9	7.9	16.0	5.8	2.7	1.9	6.6	49.5	19.5	6.2	3.1	19.6	11.8	5.9	3.7	2.2	5.9
		Noisy	13.2	5.5	3.2	2.7	6.2	12.4	4.8	2.3	1.8	5.3	33.7	16.1	5.1	2.6	14.4	9.8	5.1	3.5	1.9	5.1
UniVPM (ours)	Clean	Clean	12.8	5.3	3.1	2.7	6.0	12.1	4.9	2.3	1.7	5.3	32.8	15.8	5.0	2.8	14.1	9.5	5.0	3.6	2.1	5.1
		Noisy	10.0	4.7	2.7	2.4	5.0	9.6	4.0	2.2	**1.6**	4.4	24.9	13.3	4.7	2.6	11.4	7.0	**4.3**	2.9	1.8	4.0
	Noisy	Clean	11.9	5.1	3.0	2.6	5.7	10.8	4.6	2.2	1.7	4.8	27.4	14.8	4.9	2.6	12.4	8.3	4.7	3.2	1.8	4.5
		Noisy	**9.7**	**4.6**	**2.6**	**2.3**	**4.8**	**9.0**	**3.8**	**2.1**	**1.6**	**4.1**	**22.6**	**12.9**	**4.3**	**2.4**	**10.6**	**6.9**	**4.3**	**2.8**	**1.7**	**3.9**

Table 3: WER (%) on unseen testing noises with LRS3 benchmark. Testing noises "Meeting", "Cafe", "Resto" and "Station" are from DEMAND dataset (2013). Pre-training noise are from MUSAN dataset.

Method	Finetune Mode	Unlabeled Data (hrs)	Labeled Data (hrs)	WER (%)
TM-seq2seq (2018a)	AV	-	1,519	58.9
EG-seq2seq (2020)	AV	-	590	57.8
Hyb-AVSR (2021b)	AV	-	590	43.3
RNN-T (2019)	AV	-	31,000	33.6
Distill-PT (2022)	V	1,021	438	31.5
u-HuBERT (2022)	AV	2,211	433	27.2
AV-HuBERT (2022a)	AV	1,759	433	34.7
	V	1,759	433	28.6
+ Self-Training	V	1,759	433	26.9
UniVPM (ours)	AV	1,759	433	**26.7**

Table 4: WER (%) results of visual speech recognition (VSR) on LRS3 benchmark. "Finetune Mode" denotes the input modality during finetuning stage.

Method	B	S	M	N	Clean	VSR
AV-HuBERT (2022b)	23.7	39.2	10.7	6.4	1.42	34.7
Effectiveness of Online Balanced Clustering						
Memory Network (2022)	20.6	29.5	9.2	6.1	1.39	32.0
Online Clustering	19.3	22.9	8.7	5.9	1.37	31.2
Online Balanced Clustering	**13.2**	**8.2**	**6.1**	**5.1**	**1.31**	**26.7**
Effectiveness of AMIE						
None	22.3	35.4	10.4	6.0	1.39	31.8
Contrastive Learning	21.5	29.2	9.7	5.8	1.37	30.1
MINE (2018)	18.6	20.1	8.3	5.5	1.34	28.8
AMIE w/o Adv. Learning	17.0	17.9	7.7	5.4	1.33	28.2
AMIE	**13.2**	**8.2**	**6.1**	**5.1**	**1.31**	**26.7**
Analysis of Adversarial Learning						
$\mathcal{L}_{GAN}$ w/ $I(s_v, s_a)$	15.4	14.6	7.2	5.3	1.32	27.8
$\mathcal{L}_{GAN}$ w/ $I(f_v, \hat{f}_a)$	17.7	22.0	8.8	5.6	1.36	29.2
$\mathcal{L}_{GAN}$ w/ $I(s_v, s_a) + I(f_v, \hat{f}_a)$	**13.2**	**8.2**	**6.1**	**5.1**	**1.31**	**26.7**
Analysis of Regularization						
None	17.2	20.4	8.0	5.7	1.36	30.3
UniVPM w/ $\mathcal{L}_{rec}$	14.3	11.5	6.5	5.3	1.33	27.4
UniVPM w/ $\mathcal{L}_{var}$	15.6	14.6	7.2	5.4	1.33	28.5
UniVPM w/ $\mathcal{L}_{rec} + \mathcal{L}_{var}$	**13.2**	**8.2**	**6.1**	**5.1**	**1.31**	**26.7**

Table 5: Ablation study. 'B', 'S', 'M', 'N' denote average-SNR results on four MUSAN noises in Table 1. "Adv." denotes "Adversarial". The four ablations are all based on full UniVPM and independent with each other.

the-art in various noisy as well as clean conditions. Furthermore, we can also observe similar results on LRS2 dataset as shown in Table 2.

Table 3 further compares the performance of UniVPM with AV-HuBERT on unseen testing noises, which are sampled from DEMAND (Thiemann et al., 2013) dataset. First, when AV-HuBERT is finetuned and tested both on DEMAND noise, good WER performance can be achieved. However, if it is finetuned on MUSAN noise and then tested on unseen DEMAND noise, the performance would degrade a lot. In comparison, our UniVPM finetuned on clean data (purple shades) achieves significant improvement and surpasses the AV-HuBERT finetuned on MUSAN noise, which further verifies the strong generality of our model. Furthermore, when finetuned on MUSAN noise, our UniVPM even outperforms the AV-HuBERT finetuned on in-domain DEMAND noise, which highlights the superiority of our approach on unseen test noises.

Visual Speech Recognition. To further verify the effectiveness of UniVPM, we evaluate its VSR performance by discarding the input audio modality during inference, as shown in Table 4. In this case, with restored clean audio from lip movements, the proposed UniVPM significantly outperforms AV-HuBERT baseline (34.7%→26.7%). Although the visual-only training and self-training strategies improve AV-HuBERT's results, our UniVPM still defines new state-of-the-art on LRS3 benchmark.

4.3 Ablation Study

Table 5 presents the ablation study of components in UniVPM. The four parts of ablation are independent, *i.e.*, each study is conducted where other three components are kept same as full UniVPM.

Effect of Online Balanced Clustering. In UniVPM, our online clustering baseline outperforms the memory network with learnable embeddings, indicating the superiority of clustering technique

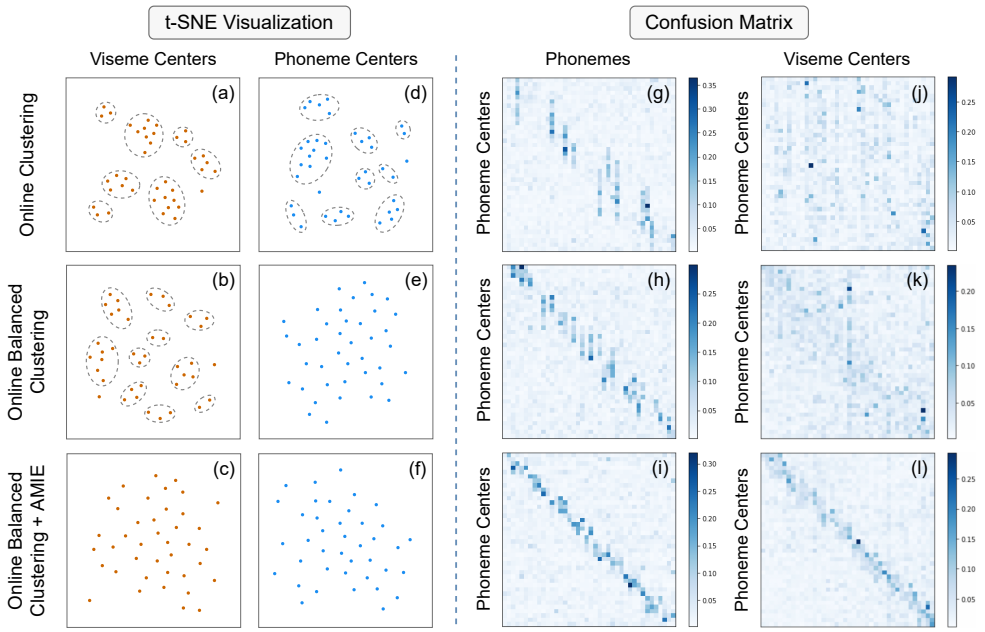


Figure 5: Left panel: t-SNE visualization of clustered viseme and phoneme centers (ellipses highlight the undesirably gathered centers). Right panel: confusion matrix of phoneme matching and viseme-phoneme mapping. In (g)-(i), the vertical axis indicates phoneme center IDs and the horizontal axis indicates real phonemes predicted by pre-trained model (Phy, 2022), while in (j)-(l) the horizontal axis indicates viseme center IDs.

in representing visemes and phonemes. Based on that, our proposed online balanced clustering achieves significant improvement by modeling all the visemes and phonemes equally without over-fitting, which is further shown in Fig. 5.

Effect of AMIE. As presented in Table 5, AMIE plays the key role in the promising performance of UniVPM by constructing strong viseme-phoneme mapping. As a comparison, the contrastive learning baseline only provides limited improvement, and MINE performs better by maximizing the estimated MI between visemes and phonemes. Based on that, our proposed AMIE introduces JS representation to stabilize system optimization, which improves performance but still suffers from the ambiguity of homophenes. To this end, our adversarial learning approach achieves further improvement by constructing strict viseme-phoneme mapping without ambiguity, as shown in Fig. 8.

Analysis of Adversarial Learning. As illustrated in Eq. 11, there are two key components in adversarial learning, *i.e.*, $I(s_v, s_a)$ that constructs viseme-phoneme mapping and $I(f_v, \hat{f}_a)$ that supervises the quality of restored clean audio. Results in Table 5 indicate that viseme-phoneme mapping is the most important, and the supervision on restored clean audio also improves the AVSR performance.

Analysis of Regularization. According to Eq. 12, $\mathcal{L}_{rec}$ and $\mathcal{L}_{var}$ are two auxiliary terms for regularization, where the former supervises the quality of restored audio, and the latter disperses clustered viseme and phoneme centers to ease their mapping construction. Both of them are proved with positive contributions to the gains of performance.

Visualizations. Fig. 5 presents t-SNE visualization and confusion matrixes to further verify the effectiveness of UniVPM. First, the online clustering baseline generates gathered viseme and phoneme centers due to over-fitting, where only several majority phonemes are modeled as shown in (g). Our proposed online balanced clustering alleviates such over-fitting issue and generates separated phoneme centers, which can cover most of the real phonemes as illustrated in (h). However, we can still observe gathered viseme centers due to homophenes, and the ambiguity of viseme-phoneme mapping is also shown in (k). To this end, our proposed AMIE effectively alleviates the ambiguity of homophenes thanks to the strong distinguishing ability of adversarial learning, which constructs strict viseme-phoneme mapping in (l). Meanwhile, we also observe dispersed viseme centers in (c), which can distinguish the same visemes that correspond to different phonemes. In addition, real phonemes are also better modeled by clustering as shown in (i).

Evaluation of Modality Transfer. Table 6 further reports phoneme match accuracy to evaluate the quality of restored clean audio. We observe that online clustering baseline can hardly restore cor-

Method	VSR WER (%)	Phoneme Match Acc. (%)
AV-HuBERT (2022a)	34.7	-
+ Online Clustering	33.5	14.2
+ Online Balanced Clustering	31.8	31.0
+ AMIE (UniVPM)	**26.7**	**67.5**

Table 6: Evaluation of restored clean audio in terms of phoneme match accuracy on LRS3 test set. It is calculated with predicted phonemes for restored audio and real clean audio by pre-trained model (Phy, 2022).

rect phonemes, and the proposed online balanced clustering improves the accuracy but still limited by the ambiguity of homophenes. Furthermore, our proposed AMIE significantly improves the quality of restored clean audio with strict viseme-phoneme mapping, which also yields better VSR result.

5 Conclusion

In this paper, we propose UniVPM, a general robust AVSR approach motivated from visual modality via unsupervised noise adaptation. UniVPM constructs universal viseme-phoneme mapping to implement modality transfer, which can restore clean audio from visual signals to enable speech recognition under any noises. Experiments on public benchmarks show that UniVPM achieves state-of-the-art under various noisy as well as clean conditions. Further analysis also verifies its effectiveness on VSR task.

Limitations

We state two points of limitations and future work in this section. First, the UniVPM combines both restored clean audio and original input audio for downstream speech recognition, while without any trade-off to weight them. For example, under extremely noisy conditions the restored clean audio plays a more important role, while in less noisy scenarios the original audio may provide more valid information. Some weighting strategies to select the most effective audio information could benefit the downstream speech recognition. Second, the proposed clustering and viseme-phoneme mapping are actually unsupervised schemes, so that it could be promising to extend our UniVPM to the popular self-supervised learning framework, in order to make full use of the abundant unlabeled data.

Ethics Statement

All the data used in this paper are publicly available and are used under the following licenses: the Creative Commons BY-NC-ND 4.0 License and Creative Commons Attribution 4.0 International License, the TED Terms of Use, the YouTube's Terms of Service, and the BBC's Terms of Use. The data is collected from TED and BBC and contain thousands of speakers from a wide range of races. To protect the anonymity, only mouth area of speaker is visualized wherever used in the paper.

Acknowledgements

This research is supported by KLASS Engineering & Solutions Pte Ltd and the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No.: AISG2-100E-2023-103). The computational work for this article was partially performed on resources of the National Supercomputing Centre, Singapore (https://www.nscc.sg).

References

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

Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. 2018b. Lrs3-ted: a large-scale dataset for visual speech recognition. *arXiv preprint arXiv:1809.00496*.

Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. 2020. Asr is all you need: Cross-modal distillation for lip reading. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2143–2147. IEEE.

Máté Aller, Heidi Solberg Økland, Lucy J MacGregor, Helen Blank, and Matthew H Davis. 2022. Differential auditory and visual phase-locking are observed during audio-visual benefit and silent lip-reading for speech perception. *Journal of Neuroscience*, 42(31):6108–6120.

David Arthur and Sergei Vassilvitskii. 2006. k-means++: The advantages of careful seeding. Technical report, Stanford.

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

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in Neural Information Processing Systems*, 33:12449–12460.

Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. 2018. Multimodal machine learning: A

survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443.

Helen L Bear and Richard Harvey. 2017. Phoneme-to-viseme mappings: the good, the bad, and the ugly. *Speech Communication*, 95:40–67.

Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. 2018. Mutual information neural estimation. In *International conference on machine learning*, pages 531–540. PMLR.

Lynne E Bernstein, Edward T Auer Jr, and Sumiko Takayanagi. 2004. Auditory speech detection in noise enhanced by lipreading. *Speech Communication*, 44(1-4):5–18.

Mathieu Bourguignon, Martijn Baart, Efthymia C Kapnoula, and Nicola Molinaro. 2020. Lip-reading enables the brain to synthesize auditory features of unknown silent speech. *Journal of Neuroscience*, 40(5):1053–1065.

David M Chan, Shalini Ghosh, Debmalya Chakrabarty, and Björn Hoffmeister. 2022. Multi-modal pre-training for automated speech recognition. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 246–250. IEEE.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

Chen Chen, Nana Hou, Yuchen Hu, Shashank Shirol, and Eng Siong Chng. 2022. Noise-robust speech recognition with 10 minutes unparalleled in-domain data. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4298–4302. IEEE.

Hui Chen, Guiguang Ding, Xudong Liu, Zijia Lin, Ji Liu, and Jungong Han. 2020a. Imram: Iterative matching with recurrent attention memory for cross-modal image-text retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12655–12663.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. 2020b. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*.

Yang Chen, Weiran Wang, I-Fan Chen, and Chao Wang. 2020c. Data techniques for online end-to-end speech recognition. *arXiv preprint arXiv:2001.09221*.

Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. 2017. Lip reading sentences in the wild. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pages 3444–3453. IEEE.

Monroe D Donsker and SR Srinivasa Varadhan. 1983. Asymptotic evaluation of certain markov process expectations for large time. iv. *Communications on Pure and Applied Mathematics*, 36(2):183–212.

Angela Fan, Edouard Grave, and Armand Joulin. 2019. Reducing transformer depth on demand with structured dropout. In *International Conference on Learning Representations*.

Usama M Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, and Ramasamy Uthurusamy. 1996. Advances in knowledge discovery and data mining. American Association for Artificial Intelligence.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *NeurIPS*.

Alex Graves. 2012. Sequence transduction with recurrent neural networks. *arXiv preprint arXiv:1211.3711*.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376.

Anmol Gulati, James Qin, Chiu Chung-Cheng, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. 2020. Conformer: Convolution-augmented transformer for speech recognition. In *Interspeech*, pages 5036–5040.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. 2018. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*.

Joanna Hong, Minsu Kim, and Yong Man Ro. 2022. Visual context-driven audio feature enhancement for robust end-to-end audio-visual speech recognition. In *23rd Annual Conference of the International Speech Communication Association, INTERSPEECH 2022*, pages 2838–2842. International Speech Communication Association.

Wei-Ning Hsu and Bowen Shi. 2022. u-hubert: Unified mixed-modal speech pretraining and zero-shot transfer to unlabeled modality. In *Advances in Neural Information Processing Systems*.

Minsu Kim, Joanna Hong, Se Jin Park, and Yong Man Ro. 2021a. Multi-modality associative bridging through memory: Speech sound recollected from face video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 296–306.

Minsu Kim, Joanna Hong, and Yong Man Ro. 2021b. Lip to speech synthesis with visual context attentional gan. *Advances in Neural Information Processing Systems*, 34:2758–2770.

Minsu Kim, Jeong Hun Yeo, and Yong Man Ro. 2022. Distinguishing homophenes using multi-head visual-audio memory for lip reading. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada*, volume 22.

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

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75.

Sangmin Lee, Hak Gu Kim, Dae Hwi Choi, Hyung-Il Kim, and Yong Man Ro. 2021. Video prediction recalling long-term motion context via memory alignment learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3054–3063.

Hsin-Yi Lin, Huan-Hsin Tseng, Xugang Lu, and Yu Tsao. 2021. Unsupervised noise adaptive speech enhancement by discriminator-constrained optimal transport. *Advances in Neural Information Processing Systems*, 34:19935–19946.

Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. 2019. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2537–2546.

Yanhua Long, Yijie Li, Hone Ye, and Hongwei Mao. 2017. Domain adaptation of lattice-free mmi based tdnn models for speech recognition. *International Journal of Speech Technology*, 20(1):171–178.

Pingchuan Ma, Rodrigo Mira, Stavros Petridis, Björn W Schuller, and Maja Pantic. 2021a. Lira: Learning visual speech representations from audio through self-supervision. *arXiv preprint arXiv:2106.09171*.

Pingchuan Ma, Stavros Petridis, and Maja Pantic. 2021b. End-to-end audio-visual speech recognition with conformers. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7613–7617. IEEE.

Pingchuan Ma, Stavros Petridis, and Maja Pantic. 2022. Visual speech recognition for multiple languages in the wild. *arXiv preprint arXiv:2202.13084*.

J MacQueen. 1967. Classification and analysis of multivariate observations. In *5th Berkeley Symp. Math. Statist. Probability*, pages 281–297.

Takaki Makino, Hank Liao, Yannis Assael, Brendan Shillingford, Basilio Garcia, Otavio Braga, and Olivier Siohan. 2019. Recurrent neural network transducer for audio-visual speech recognition. In *2019 IEEE automatic speech recognition and understanding workshop (ASRU)*, pages 905–912. IEEE.

Harry McGurk and John MacDonald. 1976. Hearing lips and seeing voices. *Nature*, 264(5588):746–748.

Pierre Mégevand, Manuel R Mercier, David M Groppe, Elana Zion Golumbic, Nima Mesgarani, Michael S Beauchamp, Charles E Schroeder, and Ashesh D Mehta. 2020. Crossmodal phase reset and evoked responses provide complementary mechanisms for the influence of visual speech in auditory cortex. *Journal of Neuroscience*, 40(44):8530–8542.

Zhong Meng, Zhuo Chen, Vadim Mazalov, Jinyu Li, and Yifan Gong. 2017. Unsupervised adaptation with domain separation networks for robust speech recognition. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 214–221. IEEE.

Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-value memory networks for directly reading documents. *arXiv preprint arXiv:1606.03126*.

Audrey R Nath and Michael S Beauchamp. 2011. Dynamic changes in superior temporal sulcus connectivity during perception of noisy audiovisual speech. *Journal of Neuroscience*, 31(5):1704–1714.

Xichen Pan, Peiyu Chen, Yichen Gong, Helong Zhou, Xinbing Wang, and Zhouhan Lin. 2022. Leveraging unimodal self-supervised learning for multimodal audio-visual speech recognition. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4491–4503, Dublin, Ireland. Association for Computational Linguistics.

Se Jin Park, Minsu Kim, Joanna Hong, Jeongsoo Choi, and Yong Man Ro. 2022. Synctalkface: Talking face generation with precise lip-syncing via audio-lip memory. In *36th AAAI Conference on Artificial Intelligence (AAAI 22)*. Association for the Advancement of Artificial Intelligence.

Stavros Petridis, Themos Stafylakis, Pingchuan Ma, Georgios Tzimiropoulos, and Maja Pantic. 2018. Audio-visual speech recognition with a hybrid ctc/attention architecture. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 513–520. IEEE.

Vitou Phy. 2022. Automatic Phoneme Recognition on TIMIT Dataset with Wav2Vec 2.0. If you use this model, please cite it using these metadata.

KR Prajwal, Rudrabha Mukhopadhyay, Vinay P Namboodiri, and CV Jawahar. 2020. Learning individual speaking styles for accurate lip to speech synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13796–13805.

Leyuan Qu, Cornelius Weber, and Stefan Wermter. 2019. Lipsound: Neural mel-spectrogram reconstruction for lip reading. In *INTERSPEECH*, pages 2768–2772.

Leyuan Qu, Cornelius Weber, and Stefan Wermter. 2021. Lipsound2: Self-supervised pre-training for lip-to-speech reconstruction and lip reading. *arXiv preprint arXiv:2112.04748*.

Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. 2019. Rapid learning or feature reuse? towards understanding the effectiveness of maml. In *International Conference on Learning Representations*.

Sucheng Ren, Yong Du, Jianming Lv, Guoqiang Han, and Shengfeng He. 2021. Learning from the master: Distilling cross-modal advanced knowledge for lip reading. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13325–13333.

Robert T Sataloff. 1992. The human voice. *Scientific American*, 267(6):108–115.

Bowen Shi, Wei-Ning Hsu, Kushal Lakhotia, and Abdelrahman Mohamed. 2022a. Learning audio-visual speech representation by masked multimodal cluster prediction. In *International Conference on Learning Representations*.

Bowen Shi, Wei-Ning Hsu, and Abdelrahman Mohamed. 2022b. Robust self-supervised audio-visual speech recognition. *arXiv preprint arXiv:2201.01763*.

David Snyder, Guoguo Chen, and Daniel Povey. 2015. Musan: A music, speech, and noise corpus. *arXiv preprint arXiv:1510.08484*.

Ge Song, Dong Wang, and Xiaoyang Tan. 2018. Deep memory network for cross-modal retrieval. *IEEE Transactions on Multimedia*, 21(5):1261–1275.

Qiya Song, Bin Sun, and Shutao Li. 2022. Multimodal sparse transformer network for audio-visual speech recognition. *IEEE Transactions on Neural Networks and Learning Systems*.

William H Sumby and Irwin Pollack. 1954. Visual contribution to speech intelligibility in noise. *The journal of the acoustical society of america*, 26(2):212–215.

Joachim Thiemann, Nobutaka Ito, and Emmanuel Vincent. 2013. The diverse environments multi-channel acoustic noise database (demand): A database of multichannel environmental noise recordings. In *Proceedings of Meetings on Acoustics ICA2013*, volume 19, page 035081. Acoustical Society of America.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Zhong-Qiu Wang, Peidong Wang, and DeLiang Wang. 2020. Complex spectral mapping for single-and multi-channel speech enhancement and robust asr. *IEEE/ACM transactions on audio, speech, and language processing*, 28:1778–1787.

Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R Hershey, and Tomoki Hayashi. 2017. Hybrid ctc/attention architecture for end-to-end speech recognition. *IEEE Journal of Selected Topics in Signal Processing*, 11(8):1240–1253.

Jason Weston, Sumit Chopra, and Antoine Bordes. 2014. Memory networks. *arXiv preprint arXiv:1410.3916*.

Hui Xiong, Junjie Wu, and Jian Chen. 2006. K-means clustering versus validation measures: a data distribution perspective. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 779–784.

Bo Xu, Cheng Lu, Yandong Guo, and Jacob Wang. 2020. Discriminative multi-modality speech recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14433–14442.

Jianwei Yu, Shi-Xiong Zhang, Jian Wu, Shahram Ghorbani, Bo Wu, Shiyin Kang, Shansong Liu, Xunying Liu, Helen Meng, and Dong Yu. 2020. Audio-visual recognition of overlapped speech for the lrs2 dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6984–6988. IEEE.

Jisi Zhang, Catalin Zorila, Rama Doddipatla, and Jon Barker. 2022. On monoaural speech enhancement for automatic recognition of real noisy speech using mixture invariant training. *arXiv preprint arXiv:2205.01751*.

Pengfei Zhang, Jianru Xue, Cuiling Lan, Wenjun Zeng, Zhanning Gao, and Nanning Zheng. 2019. Eleatt-rnn: Adding attentiveness to neurons in recurrent neural networks. *IEEE Transactions on Image Processing*, 29:1061–1073.

Ya Zhao, Rui Xu, Xinchao Wang, Peng Hou, Haihong Tang, and Mingli Song. 2020. Hearing lips: Improving lip reading by distilling speech recognizers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6917–6924.

Hao Zhu, Huaibo Huang, Yi Li, Aihua Zheng, and Ran He. 2021a. Arbitrary talking face generation via attentional audio-visual coherence learning. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 2362–2368.

Hao Zhu, Man-Di Luo, Rui Wang, Ai-Hua Zheng, and Ran He. 2021b. Deep audio-visual learning: A survey. *International Journal of Automation and Computing*, 18(3):351–376.

Mode	PT Type	FT Type	Babble, SNR (dB) = -10	-5	0	5	10	avg	Speech, SNR (dB) = -10	-5	0	5	10	avg	Music + Natural, SNR (dB) = -10	-5	0	5	10	avg	Clean ∞
A	Clean	Clean	99.3	89.6	43.9	11.0	3.7	49.5	102.5	93.8	63.5	24.1	10.7	58.9	58.6	35.9	13.9	5.4	2.6	23.3	1.55
		Noisy	98.2	65.6	17.0	5.3	2.7	37.8	94.3	73.8	46.3	22.9	9.7	49.4	43.4	18.0	6.5	3.2	2.1	14.6	1.50
	Noisy	Clean	98.3	77.6	23.0	7.3	2.9	41.8	87.3	62.9	41.0	22.2	8.9	44.5	43.4	19.3	7.1	3.4	2.5	15.1	1.62
		Noisy	97.5	62.3	15.7	5.1	2.6	36.6	81.7	56.2	37.3	19.0	8.3	40.5	38.7	15.1	5.7	3.1	2.3	13.0	1.60
AV	Clean	Clean	72.6	30.9	9.8	2.9	2.1	23.7	93.4	71.6	22.1	6.1	2.7	39.2	24.1	10.9	3.6	2.4	1.9	8.6	1.42
		Noisy	30.0	15.2	5.9	2.7	1.9	11.1	15.9	7.5	3.9	2.4	1.9	6.3	12.1	5.9	3.1	2.2	1.8	5.0	1.40
	Noisy	Clean	39.4	14.5	5.2	2.7	2.0	12.8	18.8	5.1	3.1	2.3	1.9	6.2	11.4	5.0	2.8	2.2	1.8	4.6	1.54
		Noisy	28.4	13.4	5.0	2.6	1.9	10.3	11.4	4.6	2.9	2.2	1.8	4.6	9.7	4.7	2.5	1.9	1.8	4.1	1.40

Table 7: WER (%) of AV-HuBERT on LRS3 benchmark. “Mode” denotes the input modality during both finetuning and inference stages, “PT Type" denotes the pre-training data type, “FT Type" denotes the finetuning data type, and “avg” denotes the average performance on all SNRs.

A Supplementary Experimental Analysis

A.1 Analysis of the Noise-Robustness of AVSR

Table 7 presents the performance of AV-HuBERT to analyze the noise-robustness of AVSR system. First, as the original motivation of AVSR, the visual modality significantly improves the audio-only speech recognition performance under various noisy as well as clean testing conditions, especially the low-SNR environments. However, most existing efforts still focus on audio modality to improve robustness considering its dominance in AVSR task. The reason is the inherent information insufficiency of visual modality to represent speech content. Mainstream approaches introduce noise adaptation techniques to strengthen robustness, where most of them leverage noise-corrupted data to improve network training (Afouras et al., 2018a; Ma et al., 2021b; Pan et al., 2022; Shi et al., 2022b; Hsu and Shi, 2022). As shown in Table 7, available noisy training data significantly improves the AVSR performance in different testing conditions. However, this strategy is usually faced with two practical challenges. First, it requires abundant labeled noisy audio-visual training data, which is not always available in some real-world scenarios (Meng et al., 2017; Long et al., 2017; Lin et al., 2021; Chen et al., 2022). For instance, in scenarios like theatre, it is valuable to develop a AVSR system but costly to obtain sufficient training data. Second, as it is impossible to cover all the real-world noises in training data, when some unseen noises appear in practical testing scenarios, the well-trained model may not perform well as shown in Table 3, resulting in less optimal model generality (Meng et al., 2017). Above two challenges motivate this work. With unsupervised noise adaptation investigated on visual modality, our proposed UniVPM improves the AVSR performance under clean training data to a comparable level to the state-of-the-art AV-HuBERT trained on noisy data in various noisy as well as clean testing conditions, as shown in Table 1, 2, and 3. Moreover, available noisy training data can further improve the robustness of UniVPM and yield new state-of-the-arts on both LRS3 and LRS2 benchmarks.

A.2 Analysis of Limited In-domain Noisy Audio-Visual Data

According to §1 and §A.1, the first challenge of audio modality-based robust AVSR is the limited in-domain noisy audio-visual data, which leads to domain mismatch between training and testing (Meng et al., 2017; Long et al., 2017; Lin et al., 2021; Chen et al., 2020c, 2022). Actually there are two methods of obtaining such data, *i.e.*, collection and simulation. First, we can collect and transcribe amounts of noisy audio-visual data under real-world scenarios, but that is extremely time-consuming and laborious, and to our best knowledge there is currently no such public dataset. Second, as there is sufficient clean transcribed audio-visual data (Afouras et al., 2018b; Chung et al., 2017), we can collect in-domain noise to simulate noisy audio-visual data. However, this data augmentation method can only partially alleviate but not resolve the domain mismatch problem (Zhang et al., 2022). What is worse, the in-domain noise data is also not always available in all the real-world scenarios (Meng et al., 2017; Long et al., 2017; Chen et al., 2020c, 2022).

As presented in Table 1, in case of no available in-domain noise, our UniVPM achieves comparable performance to previous state-of-the-art trained on in-domain noise. When in-domain noise is available, our UniVPM directly outperforms previous state-of-the-art, which breaks out the limit of data augmentation and moves one step forward to the real noisy data training setting (*i.e.*, oracle). In addition, Table 3 further investigates the cases with out-of-domain training noise, where our UniVPM even

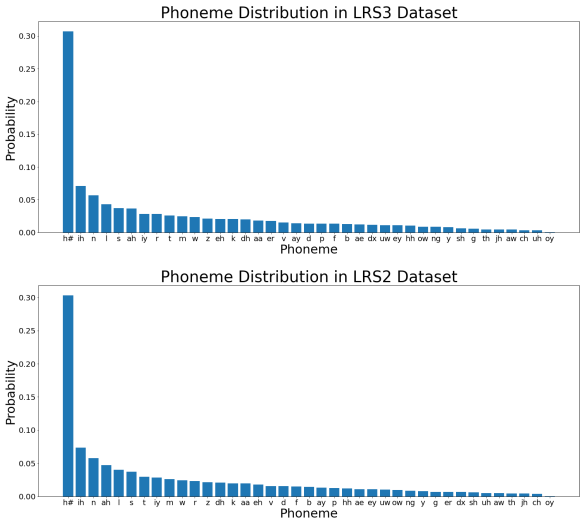


Figure 6: Phoneme distributions in LRS3 and LRS2 datasets. Pre-trained phoneme recognition model (Phy, 2022) is used for statistics, where speech is recognized into 44 phonemes, with 39 of them visualized in figures and another 5 special phonemes eliminated (*i.e.*, '|', '[UNK]', '[PAD]', '<s>', '</s>').

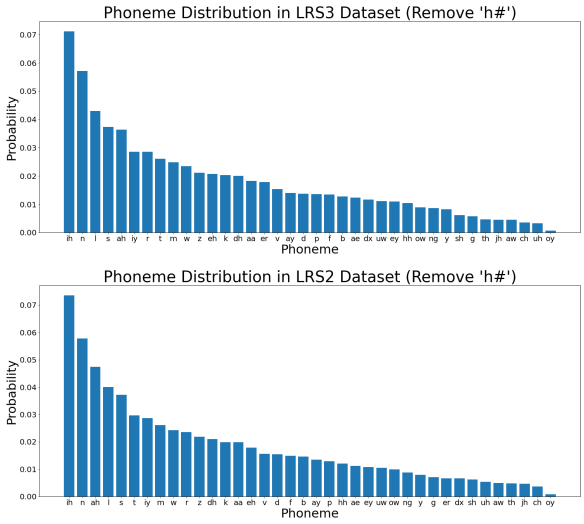


Figure 7: Phoneme distributions without 'h#'.

surpasses previous state-of-the-art trained on in-domain noise. As a result, our proposed approach effectively alleviates the limitation of in-domain noisy data in audio modality-based robust AVSR.

A.3 Analysis of UniVPM from Meta-Learning Perspective

The main idea of our proposed UniVPM can also be explained from meta-learning perspective (Raghu et al., 2019), *i.e.*, learn how to learn. In AVSR task, considering the inherent information sufficiency of visual modality to represent speech content (Sataloff, 1992; Ren et al., 2021), the key factor of its robustness is still the informative audio modality. However, audio is usually interfered by background noise during practical inference. Therefore, the key of improving robustness is to gain sufficient knowledge from clean audio in training stage, and meta-learning exactly tells AVSR how to learn from the clean audio. Motivated by this idea, we leverage clean audio-visual data to train the core modules of UniVPM, *i.e.*, viseme and phoneme banks, where video serves as "prompt" and clean audio serves as "meta". In particular, our UniVPM learns the mapping between visemes and phonemes, which then enables modality transfer to restore clean audio against testing noises. Here the viseme-phoneme mapping defines *how to learn from clean audio*. Therefore, we only need video "prompt" during inference to access the clean audio "meta", which enables UniVPM to adapt to any testing noises.

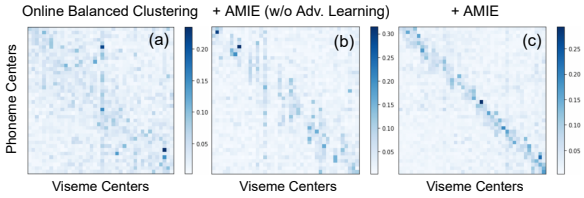


Figure 8: Confusion matrix of viseme-phoneme mapping in (a) Online Balanced Clustering, (b) Online Balanced Clustering + AMIE (without adversarial learning) and (c) Online Balanced Clustering + AMIE.

A.4 Analysis of Phoneme Distribution in LRS3 and LRS2 Datasets

Fig. 6 presents the phoneme distribution in LRS3 and LRS2 datasets. We can observe that in both datasets, the phoneme obeys a long-tail distribution (Liu et al., 2019) with head classes including 'h#', 'ih', 'n', 'l', 's', 'ah', etc. For better visualization, Fig. 7 removes the dominant phoneme 'h#' and also presents a long-tail distribution. Therefore, the neural network trained on these data is prone to over-fitting to head phoneme classes, resulting in less satisfactory performance on tail classes.

LRS3 and LRS2 are both large-scale English reading speech datasets recorded with thousands of speakers from a wide range of races, so that they can be roughly representative of the phoneme distribution of English language.

B Experimental Details

B.1 Datasets

LRS3[6] (Afouras et al., 2018b) is currently the largest public sentence-level lip reading dataset,

[6] https://www.robots.ox.ac.uk/~vgg/data/lip_reading/lrs3.html

which contains over 400 hours of English video extracted from TED and TEDx talks on YouTube. The training data is divided into two parts: pretrain (403 hours) and trainval (30 hours), and both of them are transcribed at sentence level. The pretrain part differs from trainval in that the duration of its video clips are at a much wider range. Since there is no official development set provided, we randomly select 1,200 samples from trainval as validation set ($\sim$ 1 hour) for early stopping and hyper-parameter tuning. In addition, it provides a standard test set (0.9 hours) for evaluation.

LRS2[7] (Chung et al., 2017) is a large-scale publicly available labeled audio-visual (A-V) datasets, which consists of 224 hours of video clips from BBC programs. The training data is divided into three parts: pretrain (195 hours), train (28 hours) and val (0.6 hours), which are all transcribed at sentence level. An official test set (0.5 hours) is provided for evaluation use. The dataset is very challenging as there are large variations in head pose, lighting conditions and genres.

B.2 Data Preprocessing

The data preprocessing for above two datasets follows the LRS3 preprocessing steps in prior work (Shi et al., 2022a). For the audio stream, we extract the 26-dimensional log filter-bank feature at a stride of 10 ms from input raw waveform. For the video clips, we detect the 68 facial keypoints using dlib toolkit (King, 2009) and align the image frame to a reference face frame via affine transformation. Then, we convert the image frame to gray-scale and crop a 96×96 region-of-interest (ROI) centered on the detected mouth. During training, we randomly crop a 88×88 region from the whole ROI and flip it horizontally with a probability of 0.5. At inference time, the 88×88 ROI is center cropped without horizontal flipping. To synchronize these two modalities, we stack each 4 neighboring acoustic frames to match the image frames that are sampled at 25Hz.

B.3 Model Configurations

Front-ends. We adopt the modified ResNet-18 from prior work (Shi et al., 2022a) as visual front-end, where the first convolutional layer is replaced by a 3D convolutional layer with kernel size of 5×7×7. The visual feature is flattened into an 1D vector by spatial average pooling in the end. For audio front-end, we use one linear projection layer followed by layer normalization (Ba et al., 2016).

UniVPM. The viseme and phoneme banks contain $N = 40$ clusters, following the amount of English phonemes (Phy, 2022), *i.e.*, 39 regular phonemes and one special phoneme '`[PAD]`' that indicates silence. It is worth mentioning that the actual amount of visemes is less than phonemes due to homophene phenomenon, *i.e.*, one-to-many lip-audio mapping (Bear and Harvey, 2017), but in this work we set same number of clusters to construct a strict one-to-one viseme-phoneme mapping, as shown in Fig. 5 and Fig. 8. The cluster capacity S_{max} in Alg. 1 is set to 20, and the temperature τ in Eq. 9 is set to 0.1.

Speech Recognition. The downstream speech recognition model follows AV-HuBERT (Shi et al., 2022b) with 24 Transformer (Vaswani et al., 2017) encoder layers and 9 decoder layers, where the embedding dimension/feed-forward dimension/attention heads in each Transformer layer are set to 1024/4096/16 respectively. 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., 2019) at a rate of 0.1.

The total number of parameters in our UniVPM and AV-HuBERT baseline are 478M and 476M.

B.4 Data Augmentation

Following prior work (Shi et al., 2022b), we use many noise categories for data augmentation to simulate noisy training data. We select the noise categories of "`babble`", "`music`" and "`natural`" from MUSAN noise dataset (Snyder et al., 2015), and extract some "`speech`" noise samples from LRS3 dataset. For experiments on unseen testing noises (see Table 3), we also select the noise categories of "`Meeting`", "`Cafe`", "`Resto`" and "`Station`" from DEMAND noise dataset (Thiemann et al., 2013). All categories are divided into training, validation and test partitions.

During training process, we randomly select one noise category and sample a noise clip from its training partition. Then, we randomly mix the sampled noise with input clean audio, at signal-to-noise ratio (SNR) of 0dB with a probability of 0.25.

At inference time, we evaluate our model on clean and noisy test sets respectively. Specifically, the system performance on each noise type is evaluated separately, where the testing

[7] https://www.robots.ox.ac.uk/~vgg/data/lip_reading/lrs2.html

noise clips are added at five different SNR levels: $\{-10, -5, 0, 5, 10\}dB$. At last, the testing results on different noise types and SNR levels will be averaged to obtain the final noisy WER result.

B.5 Training and Inference

Training. The noisy training data is synthesized by adding random noise from MUSAN (Snyder et al., 2015) or DEMAND (Thiemann et al., 2013) of 0dB at a probability of 0.25. We load the pre-trained AV-HuBERT[8] for front-ends and downstream speech recognition model, and then follow its sequence-to-sequence (S2S) finetuning configurations (Shi et al., 2022b) to train our system. We use Transformer decoder to decode the encoded features into unigram-based subword units (Kudo, 2018), where the vocabulary size is set to 1000. The weighting parameters $\lambda_{GAN}/\lambda_{rec}/\lambda_{var}$ in Eq. 12 are set to 0.1/0.2/0.5, respectively. The entire system is trained for 60K steps using Adam optimizer (Kingma and Ba, 2014), where the learning rate is warmed up to a peak of 0.001 for the first 20K updates and then linearly decayed. The training process takes $\sim$ 2.5 days on 4 NVIDIA-V100-32GB GPUs, where in comparison the AV-HuBERT finetuning takes $\sim$ 1.3 days on 4 NVIDIA-V100-32GB GPUs.

Inference. As shown in Table 1, the testing noises "`Babble`", "`Music`" and "`Natural`" are sampled from MUSAN, and "`Speech`" is drawn from LRS3, following prior work (Shi et al., 2022b). No language model is used during inference. We employ beam search for decoding, where the beam width and length penalty are set to 50 and 1 respectively. All hyper-parameters in our systems are tuned on validation set. Since our experimental results are quite stable, a single run is performed for each reported result.

B.6 Details of UniVPM Optimization

As detailed in Alg. 2, we design a two-step adversarial learning strategy for UniVPM optimization, where the discriminator and generator play a two-player minimax game. First, we *maximize* $\mathcal{L}_{GAN}$ to update the discriminator, where generator is detached from optimization. According to Eq. 11, maximizing the first term of $\mathcal{L}_{GAN}$ increases the MI between visual and audio sequences, while maximizing the second term is actually decreasing the MI between visemes and phonemes, as well as the MI between visual and restored audio sequences (this is opposite to our desired viseme-phoneme mapping and modality transfer). Second, we freeze discriminator and update the rest network, where *minimizing* $\mathcal{L}_G$ increases the MI between visemes and phonemes, as well as MI between visual and restored audio sequences. In addition, $\mathcal{L}_{ASR}$ optimizes the downstream speech recognition model, $\mathcal{L}_{rec}$ supervise the quality of restored clean audio, and $\mathcal{L}_{var}$ disperses the viseme and phoneme centers to ease their mapping construction. The entire system is trained in an end-to-end manner.

In actual experiments, to save computation cost, we update $\mathcal{B}_v$ and $\mathcal{B}_a$ once every 10 epochs, which has been proved with no affect on the system performance. One can refer to our attached code for more implementation details.

Algorithm 2 UniVPM Optimization.

Require: Training data D_{train} that contains visual-audio pairs (x_v, x_a) and the text transcription y. The UniVPM network θ that consists of visual front-end θ_{vf}, audio front-end θ_{af}, viseme bank $\mathcal{B}_v$, phoneme bank $\mathcal{B}_a$, AMIE θ_{AMIE} and speech recognition model θ_{ASR}. Hyper-parameter weights λ_{GAN}, λ_{rec}, λ_{var}.

Load pre-trained AV-HuBERT for θ_{vf}, θ_{af} and θ_{ASR}, randomly initialize θ_{AMIE}.
Initialize empty banks $\mathcal{B}_v$ and $\mathcal{B}_a$.
while not converged **do**
  for $(x_v, x_a) \in D_{\text{train}}$ **do**
    FORWARD-PROPAGATION:
      $f_v = \theta_{vf}(x_v), f_a = \theta_{af}(x_a)$ ▷ front-ends
      Update $\mathcal{B}_v$ and $\mathcal{B}_a$ according to Alg. 1
      Obtain viseme sequence s_v from f_v and $\mathcal{B}_v$
      Obtain phoneme sequence s_a from f_a and $\mathcal{B}_a$
      Generate restored audio $\hat{f}_a$ in Eq. 9 and 10
      $\hat{y} = \theta_{ASR}(f_v \oplus f_a \oplus \hat{f}_a)$ ▷ recognition
    TRAINING OBJECTIVES:
      $\mathcal{L}_{GAN}$ ($\mathcal{L}_D$ and $\mathcal{L}_G$) in Eq. 11
      $\mathcal{L}_{rec} = \|\hat{f}_a - f_a\|_2$
      $\mathcal{L}_{var} = \text{Var}(c_v^1, ..., c_v^N) + \text{Var}(c_a^1, ..., c_a^N)$
      $\mathcal{L}_{ASR} = \text{CrossEntropy}(\hat{y}, y)$
    BACK-PROPAGATION: ▷ adversarial learning
      UPDATE AMIE: ▷ unfreeze θ_{AMIE}
        $\arg\max_{\theta_{AMIE}} \mathcal{L}_{GAN}$
      UPDATE REST NETWORK: ▷ freeze θ_{AMIE}
        $\arg\min_{\theta \setminus \theta_{AMIE}} \mathcal{L}_{ASR} + \lambda_{GAN} \cdot \mathcal{L}_G + \lambda_{rec} \cdot \mathcal{L}_{rec} + \lambda_{var} \cdot \mathcal{L}_{var}$
  end for
end while

B.7 Baselines

In this section, we describe the baselines for comparison.

- **TM-seq2seq** (Afouras et al., 2018a): TM-seq2seq proposes a Transformer-based AVSR

[8] https://github.com/facebookresearch/av_hubert

system to model the A-V features separately and then attentively fuse them for decoding, and uses cross-entropy based sequence-to-sequence loss as training criterion.

- **TM-CTC** (Afouras et al., 2018a): TM-CTC shares the same architecture with TM-seq2seq, but uses CTC loss (Graves et al., 2006) as training criterion.

- **Hyb-RNN** (Petridis et al., 2018): Hyb-RNN proposes a RNN-based AVSR model with hybrid seq2seq/CTC loss (Watanabe et al., 2017), where the A-V features are encoded separately and then concatenated for decoding.

- **RNN-T** (Makino et al., 2019): RNN-T adopts the popular recurrent neural network transducer (Graves, 2012) for AVSR task, where the audio and visual features are concatenated before fed into the encoder.

- **EG-seq2seq** (Xu et al., 2020): EG-seq2seq builds a joint audio enhancement and multimodal speech recognition system based on RNN (Zhang et al., 2019), where the A-V features are concatenated before decoding.

- **LF-MMI TDNN** (Yu et al., 2020): LF-MMI TDNN proposes a joint audio-visual speech separation and recognition system based on time-delay neural network (TDNN), where the A-V features are concatenated before fed into the recognition network.

- **Hyb-AVSR** (Ma et al., 2021b): Hyb-AVSR proposes a Conformer-based (Gulati et al., 2020) AVSR system with hybrid seq2seq/CTC loss, where the A-V input streams are first encoded separately and then concatenated for decoding.

- **MoCo+w2v2** (Pan et al., 2022): MoCo+w2v2 employs self-supervised pre-trained audio and visual front-ends, *i.e.*, wav2vec 2.0 (Baevski et al., 2020) and MoCo v2 (Chen et al., 2020b), to generate better audio-visual features for fusion and decoding.

- **AV-HuBERT** (Shi et al., 2022a,b): AV-HuBERT employs self-supervised learning to capture deep A-V contextual information, where the A-V features are masked and concatenated before fed into Transformer encoder to calculate masked-prediction loss for pre-training, and sequence-to-sequence loss is then used for finetuning.

- **u-HuBERT** (Hsu and Shi, 2022): u-HuBERT extends AV-HuBERT to a unified framework of audio-visual and audio-only pre-training.

- **Distill-PT** (Ma et al., 2022): Distill-PT proposes a Conformer-based VSR framework with additional distillation from pre-trained ASR and VSR models.

C Clustering Algorithms

C.1 Uniform Effect in K-Means

K-Means (MacQueen, 1967) is the most popular and successful clustering algorithm, where sample re-allocation and center renewal are executed alternatively to minimize the intra-cluster distance. However, Xiong et al. (2006) points out that K-Means algorithm tends to produce balanced clustering result, *a.k.a.*, uniform effect. This preference seriously degrades the performance when the clusters are imbalanced-sized. The consequence is that the center of minority clusters will gradually move to the territory of majority cluster, as illustrated in Fig. 3 (a). In other words, the K-Means algorithm will be over-fitted to majority clusters, leaving the samples in minority clusters not well modeled.

C.2 K-Means++

The performance of K-Means clustering relies on the center initialization, where the vanilla algorithm initialize cluster centers randomly. K-Means++ (Arthur and Vassilvitskii, 2006) is an improved version with dispersed initial centers. It determines cluster centers one by one, and each newly initialized center is pushed as distant as possible to the existed centers. As a result, the K initial cluster centers would separate from each other and benefit the subsequent clustering process.

C.3 Details of Online Clustering Baseline

For comparison, we build an Online Clustering algorithm as baseline. It is similar to Alg. 1 but employs a vanilla random pruning strategy, instead of re-sampling, to control the total memory of the bank. Our strategy is to randomly keep S_{thr} samples in the cluster if its number of samples exceeds S_{thr}. Compared to the proposed Online Balanced Clustering algorithm, this baseline also controls

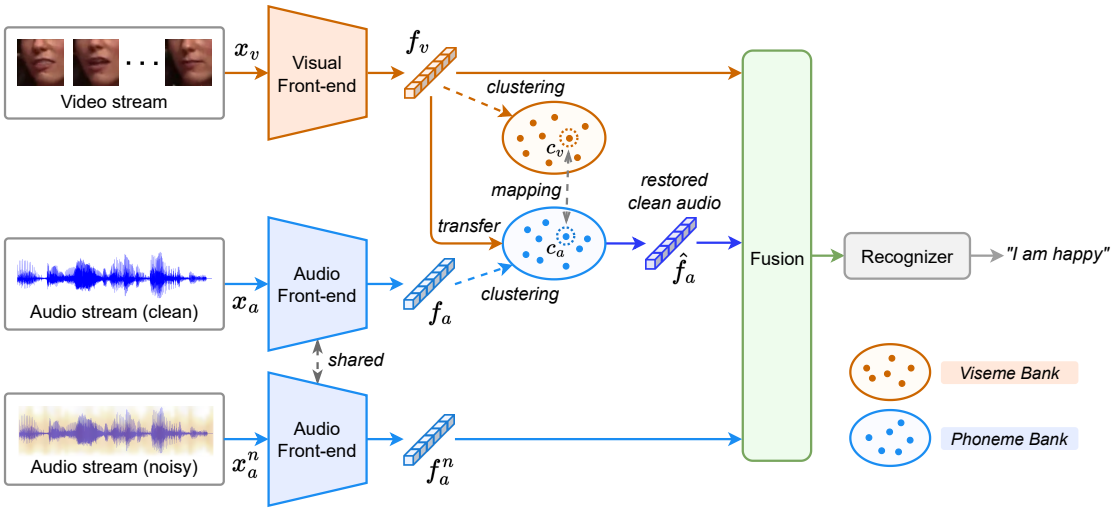


Figure 9: Illustration of noisy training pipeline of UniVPM. Both clean and noisy audio are used for training, where the clean audio is employed for phoneme clustering and the noisy audio is used to improve the system noise-robustness. Compared to Fig. 2, there is an extra data stream of noisy audio to improve robustness.

memory size but ignores the imbalanced clusters, as indicated by the dashed ellipses in Fig. 3 (a).

C.4 Principles of Online Balanced Clustering

According to Alg. 1, the main idea of proposed Online Balanced Clustering is the re-sampling operation to balance cluster sizes. For majority clusters, we perform undersampling to maintain the S_{thr} nearest samples to cluster center, so that the gathered clusters in Fig. 3 (a) can be separated. For minority clusters, we introduce oversampling to interpolate a new sample near the center, so that the minority clusters are highlighted. As a result, all the clusters are balanced-sized and separated from each other as shown in Fig. 3 (b), so that the over-fitting problem is resolved. As a result, all of the visemes and phonemes can get well represented, which enables the subsequent viseme-phoneme mapping construction.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations is after conclusion without section number

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
No response.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C ☑ Did you run computational experiments?

Section 4 and Appendix A

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix B

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 4 and Appendix B

☐ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Not applicable. Left blank.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Appendix B

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.