Do VSR Models Generalize Beyond LRS3? (Supplementary Material)

Yasser Abdelaziz Dahou Djilali[1,2] Sanath Narayan[1] Eustache Le Bihan[1]
Haithem Boussaid[1] Ebtesam Almazrouei[1] Merouane Debbah[1]
[1]Technology Innovation Institute, UAE [2]Dublin City University, Ireland

In this supplementary, we present additional analysis related to our proposed test set WildVSR. Additional details regarding the data collection are given in Section A followed by additional results on ASR models and word-level lip-reading in Section B. Finally, we present additional details regarding the training budget calculation for the VSR models in Section C.

A. Additional Details on Data Collection

Keywords selection. In the course of our study, the tool was configured to systematically extract video IDs from YouTube, utilizing a predefined array of significant keywords as the fundamental criteria for data selection. These keywords, derived from diverse trending and popular thematic categories, directed the tool to gather data across a wide spectrum of content. These keywords are: *knowledge*, *history*, *conference*, *beauty*, *dialogue*, *news*, *talk*, *interview*, *sport*, *health*, *technology*, *conversation*, *cooking*, *lesson*, *tips*, *reading*, *challenges*, *travel*, *course*, *games*. As such, this comprehensive dataset provided a rich substrate for our subsequent investigations, enabling a deeper understanding of the various parameters that govern the content popularity on the platform.

Similarity measures. As shown in Figure A.1, the similarity across our test set is relatively low (visually represented by darker colors in the matrix) signifing that the embeddings produced by the VGG-Face model are notably distinct for different test set images. This in turn implies that the VGG-Face model has successfully captured a wide array of facial feature representations, making it capable of distinguishing between different individuals effectively. Moreover, the high diversity within the similarity matrix also indicates that the models are fairly tested without a specific focus on a given facial category.

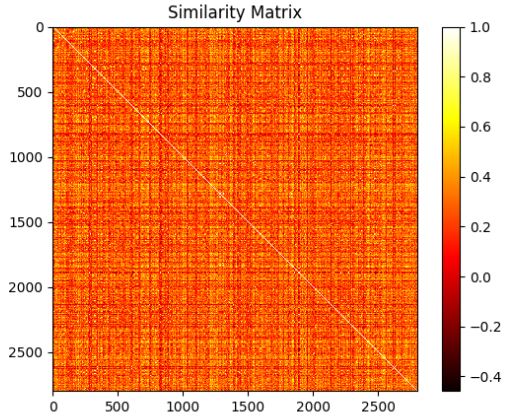

Figure A.1. The similarity matrix of the face embeddings using VGG-Face across the test set samples. It can be seen that the similairy is low across our test set, showing the diversity.

B. Additional Results

ASR models: We benchmark the Wav2Vec2.0 [1] and Whisper [8] on both LRS3 and our test sets. Wav2vec2.0 achieved 6.2 and 23.4 WER scores on both datasets respectively. In comparison, Whisper exhibits even higher performance with scores less than 4 WER on both test sets. These low WER scores signify the models' ability to accurately transcribe speech, capturing the spoken words with remarkable precision. Moreover, the small standard deviation observed for both models suggests that their performance is consistently reliable, with minimal variation in recognition errors across different samples. However, Wav2vec2.0 follows the same trend as VSR models on our test set, deviating by 12 WER points from the LRS3 score. This discrepancy while not being attributed to visual features responsible for VSR models performance, is more likely to be influenced by the sequence of phonemes in our test set. It is possible that the transcriptions in our test set contain more challenging or complex sequences of phonemes, which may pose difficulties for the VSR models and result in a drop in their performance. Unlike Whisper, Wav2vec2.0 relies solely on character-level CTC decoding without the use of

Label: THERE IS BOARDING BUSES EVERY DAY AND THESE IMPROVEMENTS WILL ADD EIGHT BUS BAYS AND REDESIGNED BUS LANES
Wav2vec2.0: BOARDING BUSSES EVERY DAY AND THESE IMPROVEMENTS WILL ADD EIGHT BUSBAS AND REDESIGNED BUS LANES
AUTO-VSR: SPORTING BUSINESS EVERY DAY AND THESE IMPROVEMENTS WITHOUT A BUS BASE AND WE NEED TO GET A BUS LIST

Label: THE OLDER GIRLS WILL TAKE THE PANADOL AND THE YOUNGER ONES WILL NORMALLY GET PERITON OR BENADRYL
Wav2vec2.0: GIRLS WILL TAKE THE PANADOL AND THE YOUNGER ONES WILL NORMALLY GET PARATON SOIER OR BANADEROL
AUTO-VSR: THIS WAS THE MANUAL AND THE ONES WOULD NORMALLY GET PARROTED TO DO IT OR MANDATORY

Label: THATS WHAT SHAITAN COMES AND TELL US HE COMES AND TELL US SURELY OTHERS SHOULD KNOW ABOUT IT COME ON YOU NEED TO
Wav2vec2.0: THAT'S WHAT CHEYPIN COMES AND TELLS US HE COMES AND TELLS US SURELY OTHERS SHOULD KNOW ABOUT IT COME ON YOU NEIDTO
AUTO-VSR: THAT'S WHY SHE SAYS I'M DETACHED I'M DETACHED I'M DETACHED

Label: UP AND THE TRAJECTORY THEN OF THOSE ARTISTS INTO LUDPORE
Wav2vec2.0: UP AND YOU KNOW ER THE TRAJECTORY THEN OF THOSE ARTISTS INTO YOU KNOW LIKE POOR AM
AUTO-VSR: UP AND YOU KNOW THE TRACTOR AT THE END OF THOSE HARNESSES YOU KNOW THEY PULL

Figure A.2. **Qualitative results.** The predictions of the Wav2vec2.0 and Auto-AVSR models on sample sequences from our test set. Wav2Vec2.0 mostly makes errors in terms of near-by homophones, *e.g.*, PARATON *vs*. PERITON in 2^{nd} row, LIKE POOR *vs*. LUDPORE in 4^{th} row. In comparison, the state-of-the-art VSR framework Auto-AVSR predictions deviate significantly from the target speech, *e.g.*, SPORTING BUSINESS *vs*. THERE IS BOARDING BUSES in 1^{st} row.

any language model to ensure valid word predictions. This lack of language modeling support in Wav2vec2.0 could contribute to its higher WER on our test set. Projecting on VSR approaches, we hypothesize that factors beyond visual features alone, such as the sequence of phonemes in our test set are also likely to contribute to the drop in performance.

B.1. WildVSR-Word

To transform our sentence-level test set into a word-level format akin to the LRW (Lip Reading in the Wild) dataset [2], we followed a systematic process. The objective was to ensure that the selected word segments were not only contextually relevant but also well-aligned with the LRW classes.

- Whisper [8] Word Boundaries: The primary challenge in transitioning from sentence to word level lies in identifying accurate word boundaries within continuous spoken sentences. To address this, we made use of whisper word boundaries. These boundaries provided a reliable temporal localization of individual words within the sentences, allowing us to select the start and end times of each word.

- LRW Class Overlap: Given that our aim is to align with the LRW word classes, we performed a filtering operation on the identified word boundaries. Only the words which overlap with the LRW class vocabulary were retained for the next steps. This ensured that our WILDVSR-Word dataset is directly comparable and compatible with existing LRW models.

- Central Frame Extraction: To maintain consistency and ensure the best representation of each word, we centered the segment on the midpoint timestamp of each selected word boundary. From this center point, we cropped video segments to obtain a fixed length of 29 frames. This length was chosen to comply with the LRW creation process.

Table A.1. **Performance comparison on word-level VSR.**

Model	Test sets	
	LRW	WildVSR-W
DCTCN/Boundary [7]	92.1	34.6
DCTCN [7]	89.6	32.5
MSTCN [7]	88.9	29.6

As shown in Table A.1, we tested models from [7] on the resulting dataset, termed "WILDVSR-Word". In fact, we observe a similar drop in accuracy as in sentence-level VSR. The DCTCN drops by 60.0 points, this confirms the generalization issues of VSR models for both sentence and word level.

C. Additional Details on FLOPs Computation

Here, we detail the approach employed for calculating the training budget (FLOPs) of the VSR/AVSR models in Table. 2 of the main paper. As discussed in Sec. 4 of the paper, we utilize the methodology described in [4] for estimating the compute budget. Accordingly, a transformer model's training compute for a single input token is approximated to be $6N$, where N denotes the number of model parameters. Briefly, it takes around $2N$ compute per token for the forward pass (the backward pass is approximately twice the compute as the forwards pass), resulting in a total of $6N$ compute per token for a single forward-backward computation. Consequently, the total training compute required is $C = 6N \times D$, where D denotes the total number of tokens the model is trained on. For the task of visual speech recog-

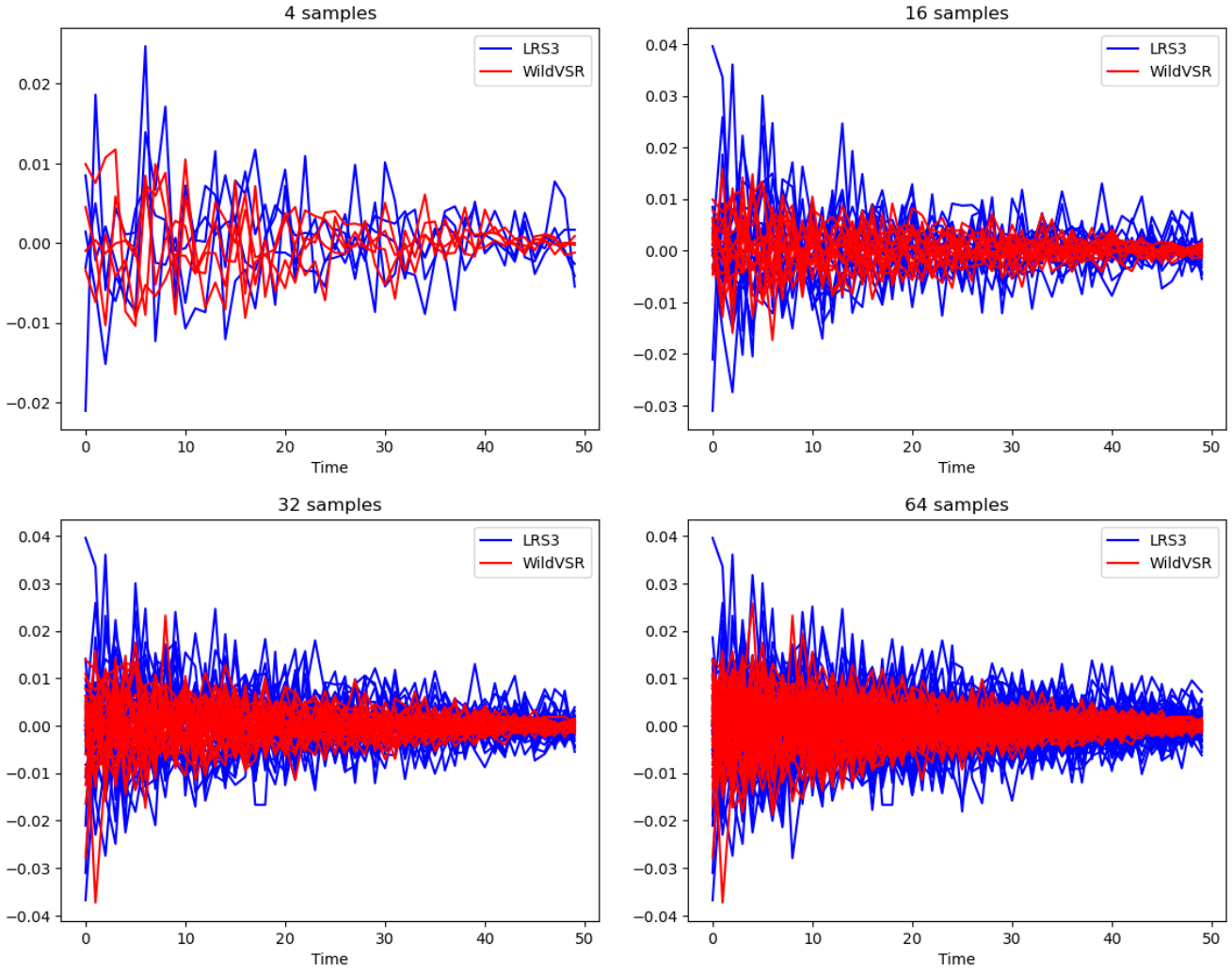

Figure A.3. **Visualization of the dominant spatial mode of the tucker decomposition over time** of Auto-AVSR encoder representations on LRS3 (in blue) and WildVSR (in red) test sets. It can be seen that when adding more samples, the LRS3 representations envelop the WildVSR representations indicating that the salient modes of LRS3 are more compared to WildVSR.

nition, tokens refer to the number of input frames. Next, we describe the calculations for different models reported in Table. 2 of the main paper.

C.1. Fully-supervised Models

Ma *et al.* [6]: This model has a total of 52.5M parameters and is trained on 1459 hours of video data for 50 epochs. The 1459 hours correspond to 131.3M frames (*i.e.*, 1459 $\times$ 3600 seconds $\times$25 fps). Thus, the total compute required for 50 epochs is $6 \times 52.5\text{M} \times (131.3\text{M} \times 50)$, which results in 2.1×10^{18} FLOPs, *i.e.*, 2.1 exaFLOPs.

Auto-AVSR [5]: This model has 250.1M parameters and is trained for 75 epochs. There are three variants of the model trained with different amount of data: 661, 1759 and 3448 hours, corresponding to 59.5M, 158.3M and 310.3M frames, respectively. As a result, the total training compute requirement for these 661, 1759 and 3448 hours variants comes out as 6.7, 17.8 and 34.9 exaFLOPs, respectively.

C.2. SSL Pretrained and Finetuned Models

C.2.1 AV-HuBERT [9]

The AV-HuBERT model has multiple variants corresponding to different model sizes and training data duration. The Base model has encoder and decoder with 103.3M and 57.3M parameters, while the Large model has 325.4M and 151.9M parameters, respectively.

The Base model is pretrained for 5 iterations for 0.4M steps on 32K frame tokens per step (32 GPUs with 1K frame tokens per GPU). Differently, the Large model is initialized from the Base model (after 4 iterations) and further

Table A.2. **Training budget computation for RAVen [3] model variants.** 'ST' refers to self-training, where finetuning is performed on 1759 hours of labeled and pseudo-labeled data. The different data regimes of 30, 433 and 1759 hours translate to 2.7M, 39M and 158.3M frames, respectively. The encoder and decoder sizes are denoted by N_e and N_d. Note that encoder size is doubled for pretraining ($2N_e$) due to the training of both audio and video encoders, while finetuning is with single encoder and decoder ($N_e + N_d$). Also see text in Section C for more details.

Model	**Encoder Size** (M)	**Pretraining** epochs × # Frames (M)	**Decoder Size** (M)	**Finetuning** epochs × # Frames (M)	**Compute** (exaFLOPS) **Pretraining**	**Finetuning**	**Total**
	N_e	D_p	N_d	D_f	$C_p = 6 \cdot 2N_e D_p$	$C_f = 6(N_e + N_d)D_f$	$C = C_p + Cf$
Low-resource Setting							
Base 433h	52.4	150 × 39	10.1	50 × 2.7	3.6	0.05	3.7
Base 1759h	52.4	150 × 158.3	10.1	50 × 2.7	14.9	0.05	14.9
Large 1759h	339.3	150 × 158.3	10.2	50 × 2.7	96.6	0.2	96.8
Large 1759h (w/ ST)	339.3	150 × 158.3	153.3	50 × 158.3	96.6	35.1	131.7
High-resource Setting							
Base 433h	52.4	150 × 39	26.3	75 × 39	3.6	1.4	5.0
Base 1759h	52.4	150 × 158.3	26.3	75 × 39	14.9	1.4	16.3
Large 1759h	339.3	150 × 158.3	153.3	75 × 39	96.6	8.7	105.3
Large 1759h (w/ ST)	339.3	150 × 158.3	153.3	75 × 158.3	96.6	35.1	131.7

Table A.3. **Performance comparison on our proposed test set with varying the attributes**. We report the best variant per model.

Method	**Accent**		**Gender**		**Age**			**Ethnicity**		
	Native	Non-Native	Male	Female	Young	Adult	Old	White	Black	Others
Auto-AVSR [5]	35.1	46.9	38.3	37.4	45.2	38.1	38.5	38.2	42.1	38.1
AV-HuBERT [9, 10]	47.5	55.3	49.2	47.7	52.0	48.4	48.7	48.5	50.0	48.4
RAVen [3]	45.2	55.0	47.5	45.3	54.5	46.4	47.3	46.2	48.7	47.3

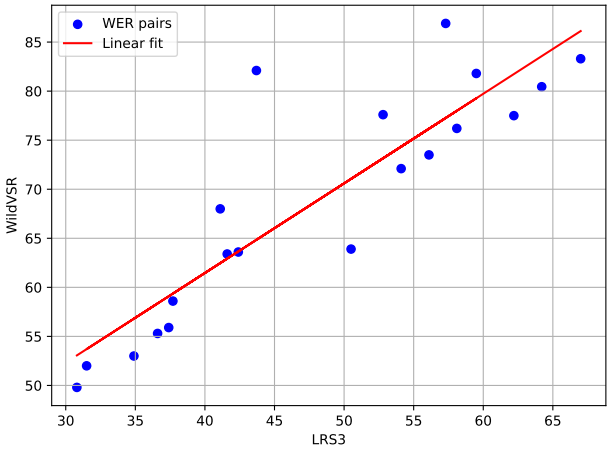

Figure A.4. **Model performance on LRS3 *vs*. WildVSR**. Each data point corresponds to one model in ones we used in the main results. The plots reveal two main phenomena: (i) There is a significant drop in accuracy from LRS3 to WildVSR. (ii) The model WERs closely follow a linear function with slope greater than 1 (1.30). This means that every WER decrease on LRS3 translates into more than one WER point on the new WildVSR test set.

pretrained for 1 iteration for 0.6M steps on 64K frame tokens per step (64 GPUs with 1K frame tokens per GPU). Consequently, pretraining the Base model for one iteration involves $6 \times 103.3\text{M} \times (0.4\text{M} \times 32\text{K})$ FLOPs, equal to 7.9 exaFLOPs. Similarly, pretraining the Large model for one iteration involves $6 \times 325.4\text{M} \times (0.6\text{M} \times 64\text{K})$ FLOPs, equal to 74.9 exaFLOPs. As a result, Base model pretraining on 5 iterations takes 39.6 exaFLOPs, while the Large model pretraining (4 base iterations followed by 1 large iteration) requires 106.6 exaFLOPs.

During finetuning in low-resource setting (30 hours), only the decoder is trained for 18K steps with 8K frame tokens per step (8 GPUs at 1K frames per GPU). Thus, finetuning the Base model in low-resource setting takes $6 \times 57.3\text{M} \times (18\text{K} \times 8\text{K})$ FLOPs, equivalent to 0.05 exaFLOPs. Similarly, finetuning the Large model takes $6 \times 151.9\text{M} \times (18\text{K} \times 8\text{K})$, resulting in 0.13 exaFLOPs. As a result, combining both pretraining and finetuning compute requirements, in the low-resource setting, Base and Large models require 39.7 and 106.7 exaFLOPs, respectively.

In contrast, in the high-resource setting (433 hours), the encoder is trained for 22.5K steps while the decoder is trained for 45K steps with 8K frame tokens per step. Thus, finetuning the Base model in high-resource setting needs $6 \times [(103.3\text{M} \times (22.5\text{K} \times 8\text{K}) + 57.3\text{M} \times (45\text{K} \times 8\text{K})]$ FLOPs, equivalent to 0.23 exaFLOPs. Similarly, finetuning the Large model takes $6 \times [(325.4\text{M} \times (22.5\text{K} \times 8\text{K}) + 151.9\text{M} \times (45\text{K} \times 8\text{K})]$, equal to 0.7 exaFLOPs. Consequently, adding both pretraining and finetuning compute requirements, in the high-resource setting, Base and Large models require 39.9 and 107.3 exaFLOPs, respectively.

C.2.2 RAVen [3]

The RAVen model has Base and Large variants trained on different data regimes in the pretraining and finetuning stages. The Base model for low-resource setting has 52.4M and 10.1M parameters in the encoder and decoder. For Base model in high-resource, the encoder is same while the decoder is larger at 26.3M parameters. Similarly, the Large model in low-resource setting has 339.3M and 10.2M parameters for encoder and decoder. While the Large model in high-resource setting has 339.3M and 153.3M parameters for encoder and decoder. The self-trained variant has same sizes as Large variant in high-resource setting. Also, it is important to note that the pretraining involves training the audio and video encoders together (*i.e.*, twice the encoder parameters $2N_e$) and finetuning utilizes a single encoder and decoder ($N_e + N_d$). Furthermore, while 150 epochs are used during pretraining for all models, the low-resource and high-resource models are finetuned for 50 and 75 epochs, respectively. The different data regimes of 30, 433 and 1759 hours translate to 2.7M, 39M and 158.3M frames, respectively.

Table A.2 reports the compute required for different variants of the RAVen model. The Base model pretrained on 433 hours in low-resource setting (30 hour finetuning) utilizes $6 \times (2 \times 52.4\text{M}) \times 39\text{M} \times 150$ FLOPs for pretraining and $6 \times (52.4 + 10.1)\text{M} \times 2.7\text{M} \times 50$ FLOPs for finetuning, resulting in 3.7 exaFLOPs in total. Similarly, the Large model pretrained on 1759 hours and finetuned on 433 hours (high-resource) requires $6 \times (2 \times 339.3\text{M}) \times (158.3\text{M} \times 150)$ FLOPs for pretraining and $6 \times (339.3 + 153.3)\text{M} \times (39\text{M} \times 75)$ FLOPs for finetuning, *i.e.*, a total of 105.3 exaFLOPs. Furthermore, since self-training involves pseudo-labeling the unlabeled data and utilizing them during finetuning, all 1759 hours are used for finetuning. Consequently, self-trained Large models require 35.1 exaFLOPs during finetuning, resulting in 126.1 exaFLOPs requirement for the entire training.

References

[1] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020. 1

[2] Joon Son Chung and Andrew Zisserman. Lip reading in the wild. In *Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part II 13*, pages 87–103. Springer, 2017. 2

[3] Alexandros Haliassos, Pingchuan Ma, Rodrigo Mira, Stavros Petridis, and Maja Pantic. Jointly learning visual and auditory speech representations from raw data. *arXiv preprint arXiv:2212.06246*, 2022. 4, 5

[4] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. 2

[5] Pingchuan Ma, Alexandros Haliassos, Adriana Fernandez-Lopez, Honglie Chen, Stavros Petridis, and Maja Pantic. Auto-avsr: Audio-visual speech recognition with automatic labels. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023. 3, 4

[6] Pingchuan Ma, Stavros Petridis, and Maja Pantic. Visual speech recognition for multiple languages in the wild. *Nature Machine Intelligence*, pages 1–10, 2022. 3

[7] Pingchuan Ma, Yujiang Wang, Stavros Petridis, Jie Shen, and Maja Pantic. Training strategies for improved lip-reading. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8472–8476. IEEE, 2022. 2

[8] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*, 2022. 1, 2

[9] Bowen Shi, Wei-Ning Hsu, Kushal Lakhotia, and Abdelrahman Mohamed. Learning audio-visual speech representation by masked multimodal cluster prediction. *arXiv preprint arXiv:2201.02184*, 2022. 3, 4

[10] Bowen Shi, Wei-Ning Hsu, and Abdelrahman Mohamed. Robust self-supervised audio-visual speech recognition. *arXiv preprint arXiv:2201.01763*, 2022. 4