OMNI-AVSR: TOWARDS UNIFIED MULTIMODAL SPEECH RECOGNITION WITH LARGE LANGUAGE MODELS

Umberto Cappellazzo[♠] *Xubo Liu*[♣] *Pingchuan Ma*[♣] *Stavros Petridis*[♠] *Maja Pantic*[♠]

[♠] Imperial College London, UK [♣] Meta AI, UK

ABSTRACT

Large language models (LLMs) have recently achieved impressive results in speech recognition across multiple modalities, including Auditory Speech Recognition (ASR), Visual Speech Recognition (VSR), and Audio-Visual Speech Recognition (AVSR). Despite this progress, current LLM-based approaches typically address each task independently, training separate models that raise computational and deployment costs while missing potential cross-task synergies. They also rely on fixed-rate token compression, which restricts flexibility in balancing accuracy with efficiency. These limitations highlight the need for a unified framework that can support ASR, VSR, and AVSR while enabling elastic inference. To this end, we present Omni-AVSR, a unified audio-visual LLM that combines efficient multi-granularity training with parameter-efficient adaptation. Specifically, we adapt the matryoshka representation learning paradigm to efficiently train across multiple audio and visual granularities, reducing its inherent training cost. Furthermore, we explore three LoRA-based strategies for adapting the backbone LLM, balancing shared and task-specific specialization. Experiments on LRS2 and LRS3 show that Omni-AVSR achieves comparable or superior accuracy to state-of-the-art baselines while training a single model at substantially lower training and deployment costs. The model also remains robust under acoustic noise, and we analyze its scaling behavior as LLM size increases, providing insights into the trade-off between performance and efficiency.

Index Terms— Audio-Visual Speech Recognition, Multimodal LLMs, Matryoshka Representation Learning

I. INTRODUCTION

Auditory Speech Recognition (ASR) [1]–[3] often degrades in noisy environments such as crowded areas or subways. To address this limitation, Audio-Visual Speech Recognition (AVSR) [4]–[6] incorporates visual cues, such as lip movements, which remain unaffected by acoustic noise, thereby enhancing recognition robustness and accuracy. Early AVSR methods relied on modality-specific encoders and handcrafted fusion strategies [7]–[9]. The introduction of Transformers [10] significantly advanced performance [11]–[13], spurring research into multimodal learning paradigms such as self-supervision [14]–[16], ASR-to-AVSR distillation [17], [18], and cross-modal complementarity [19], [20].

More recently, Multimodal Large Language Models (MLLMs) have demonstrated that integrating modalities such as vision and speech significantly extends the capabilities of LLMs, yielding state-of-the-art results across diverse tasks [21]–[26]. Building on this progress, several studies have applied LLMs to ASR, Visual Speech Recognition (VSR), and AVSR, with promising results [27]–[32].

However, most existing approaches treat each task **in isolation**, training *separate* models for ASR, VSR, and AVSR. This not only increases computational cost and complexity but also overlooks potential synergies across tasks. In contrast, studies across multiple domains have demonstrated the feasibility of unified multi-task multimodal LLMs [33]–[37]. While some attempts have been made to unify ASR, VSR, and AVSR, these either rely on costly student-teacher pseudo-labeling frameworks [38] or underperform compared to task-specific models [39], [40].

Motivated by these limitations, we introduce `Omni-AVSR`, a *unified* audio-visual LLM capable of performing ASR, VSR, and AVSR within a **single framework**. To adapt the backbone LLM to all tasks in a parameter-efficient manner, we propose three *LoRA*-based methods. Furthermore, we adapt and optimize the *matryoshka representation learning* paradigm [32], [41], [42] for our setting, enabling **efficient multi-granularity training** while mitigating its inherent computational cost. This allows the number of tokens to be dynamically adjusted at inference according to resource availability and task requirements. To the best of our knowledge, `Omni-AVSR` is the *first* audio-visual LLM that supports ASR, VSR, and AVSR jointly while enabling elastic inference under a single set of weights.

Our contributions are summarized as follows: **(1)** We provide a comprehensive evaluation of `Omni-AVSR` on the LRS2 and LRS3 benchmarks, showing that it achieves comparable or superior WER results across all three tasks. Unlike prior methods that support only joint ASR–VSR–AVSR within a single model, only multi-granularity, or neither, Omni-AVSR simultaneously supports both within a single framework, substantially reducing training and deployment costs. **(2)** We demonstrate that `Omni-AVSR` remains competitive with state-of-the-art methods under both clean and noisy conditions. **(3)** We conduct scaling experiments to analyze the trade-off between LLM size, performance, and computational efficiency.

II. OMNI-AVSR

The goal of `Omni-AVSR` is to train a single unified LLM-based model capable of performing ASR, VSR, and AVSR. At the same time, it enables flexible control of audio–visual granularity at inference according to resource constraints. In this way, `Omni-AVSR` supports multiple modalities and granularities within a single set of weights, while reducing training and deployment costs and achieving performance on par with, or even surpassing, state-of-the-art models trained independently for specific tasks or granularities.

Following prior audio-visual LLMs [27]–[29], [32], `Omni-AVSR` comprises pre-trained audio and video encoders, projection layers, and an LLM backbone (see Fig. 1a). In the next sections, we detail how `Omni-AVSR` is endowed with **1)** explicit control over audio-visual granularities during inference and **2)** the ability to jointly support ASR, VSR, and AVSR within a single model.

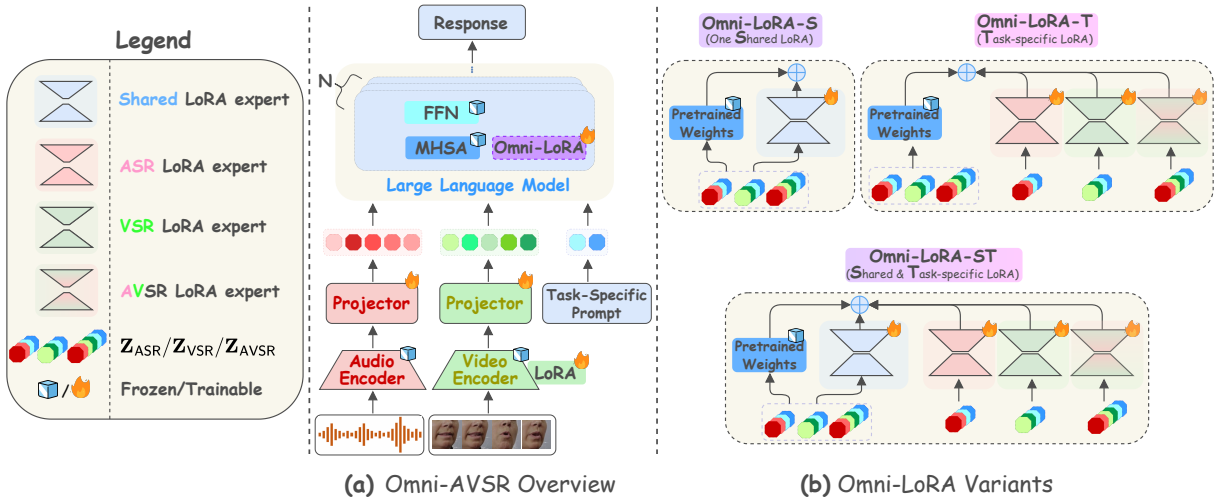

Fig. 1: Overview of **(a)** the proposed Omni-AVSR model and **(2)** its Omni-LoRA variants. Audio and video inputs are encoded by pre-trained modality-specific encoders and compressed by applying selected audio and video rates before projection into the LLM space. Omni-AVSR explores three LoRA-based LLM adaptation strategies: 1) *Omni-LoRA-S* defines a single LoRA module for both ASR, VSR, and AVSR; 2) *Omni-LoRA-T* dedicates task-specific LoRAs; 3) *Omni-LoRA-ST* makes use of both a shared LoRA and task-specific LoRA modules.

II-A. Multi-Granularity via Efficient Matryoshka Training

Given an audio waveform $\mathbf{a}$ and its corresponding lip movement video $\mathbf{v}$, we process them with a pre-trained audio encoder (*e.g.*, Whisper [43]) and video encoder (*e.g.*, AV-HuBERT [14]) to obtain audio and visual token sequences, $\mathbf{Z}^a$ and $\mathbf{Z}^v$, respectively. Reducing token granularity lowers computational cost and improves efficiency when feeding audio–visual tokens into an LLM. In AVSR, temporal continuity across modalities creates redundancy, yet most compression methods rely on fixed rates, limiting adaptability to performance and resource trade-offs [27]–[29]. While finer-grained tokens enhance recognition accuracy, they substantially increase inference cost due to the quadratic complexity of Transformers.

To address this, Llama-MTSK [32] exploits the matryoshka representation learning (MRL) principle [42] to flexibly control audio-visual granularity at inference time based on user requirements. During training, token sequences at varying granularities are generated by applying G audio compression rates $\{a_1, a_2, \cdots, a_{\mathsf{G}}\}$ and L video compression rates $\{v_1, v_2, \cdots, v_{\mathsf{L}}\}$ to the input streams. For AVSR, each of the resulting $\mathsf{G} \cdot \mathsf{L}$ audio-visual sequences is fed to the LLM, requiring $\mathsf{G} \cdot \mathsf{L}$ forward/backward passes per batch.

However, when extended to Omni-AVSR, which must also support ASR and VSR, the cost grows further: G passes for ASR, L for VSR, and G·L for AVSR. *This leads to prohibitive computational overhead and potential interference among multiple objectives*. To overcome this limitation, we introduce a **key modification**: during training, we randomly select one audio rate a_i and one video rate v_j at each iteration, yielding compressed sequences $\mathbf{Z}^{a_i}$ and $\mathbf{Z}^{v_j}$. This reduces the number of forward/backward LLM passes to only three, one per task, instead of $\mathsf{G}+\mathsf{L}+\mathsf{G}\cdot\mathsf{L}$. The resulting compressed sequences are then passed through modality-specific projection layers to match the LLM embedding dimension and concatenated with task-specific text tokens X_t^{P}, where $t \in \{\mathsf{ASR}, \mathsf{VSR}, \mathsf{AVSR}\}$ and X_t^{P} encodes both the task prompt and the transcription. Therefore, we obtain: $\mathbf{Z}_{\mathsf{ASR}} = [\mathbf{Z}^{a_i}, X_{\mathsf{ASR}}^{\mathrm{P}}]$, $\mathbf{Z}_{\mathsf{VSR}} = [\mathbf{Z}^{v_j}, X_{\mathsf{VSR}}^{\mathrm{P}}]$, and $\mathbf{Z}_{\mathsf{AVSR}} = [\mathbf{Z}^{a_i}, \mathbf{Z}^{v_j}, X_{\mathsf{AVSR}}^{\mathrm{P}}]$. This strategy preserves the flexibility of MRL at inference while substantially reducing its training cost.

II-B. Joint ASR-VSR-AVSR Training Formulation

Omni-AVSR is trained by averaging the auto-regressive next token prediction loss for each task for each input data. The LLM predicts the response $\mathbf{Y} = \{y_s\}_{s=1}^{S}$ conditioned on the multimodal input tokens, where S represents the number of tokens of the ground truth transcription. Accordingly, for each task-specific sequence $\mathbf{Z}_t$, the probability of the target $\mathbf{Y}$ is computed by $p(\mathbf{Y}|\mathbf{Z}_t) = \prod_{s=1}^{S} p_\theta(y_s|\mathbf{Z}_t, y_{<s})$, and the corresponding loss is defined as $\mathcal{L}_t = -\log p(\mathbf{Y}|\mathbf{Z}_t)$, where $y_{<s}$ is the generated output sequence up to token $s-1$, θ is the trainable parameters, and $t \in \{\mathsf{ASR}, \mathsf{VSR}, \mathsf{AVSR}\}$. Overall, the final objective we train on is:

$$\mathcal{L}_{\mathsf{OMNI}} = \lambda_{\mathsf{ASR}}\mathcal{L}_{\mathsf{ASR}} + \lambda_{\mathsf{VSR}}\mathcal{L}_{\mathsf{VSR}} + \lambda_{\mathsf{AVSR}}\mathcal{L}_{\mathsf{AVSR}}, \tag{1}$$

where λ_{ASR}, λ_{VSR}, λ_{AVSR} are task-specific weights.

II-C. Efficient LLM Adaptation via Omni-LoRA

In Omni-AVSR, following prior works [27]–[29], [31], [32], the pre-trained LLM is kept frozen while low-rank LoRA modules [44] are employed to parameter-efficiently fine-tune it. Given our multi-task setting, we explore three configurations: **1)** Omni-LoRA-**S**, **2)** Omni-LoRA-**T**, and **3)** Omni-LoRA-**ST**, illustrated in Fig. 1b. These variants allow us to systematically investigate the trade-off between parameter sharing and task specialization within Omni-AVSR.

The Omni-LoRA-**S** variant employs a *single* **S**hared LoRA module to adapt the query and value projection matrices of each LLM self-attention layer across ASR, VSR, and AVSR tasks. Specifically, a frozen pre-trained weight matrix W is decomposed into low-rank factors with down-projection parameters $W_{down} \in \mathbb{R}^{d\times r}$ and up-projection parameters $W_{up} \in \mathbb{R}^{r\times d}$, where $r \ll d$. Given an input $\mathbf{Z}_t$ for task t, the output is computed as: $\mathbf{O}_t = \mathbf{Z}_t W + \alpha(\mathbf{Z}_t W_{down})W_{up}$, where α is a scaling hyperparameter.

The Omni-LoRA-**T** variant instead defines *separate* **T**ask-specific LoRA modules, with parameters W_{down}^t and W_{up}^t specialized to each task. The output is then computed as: $\mathbf{O}_t = \mathbf{Z}_t W + \alpha(\mathbf{Z}_t W_{down}^t)W_{up}^t$. Finally, Omni-LoRA-**ST** combines both **S**hared *and* **T**ask-specific LoRA modules, yielding: $\mathbf{O}_t = \mathbf{Z}_t W +$

Table I: ASR, VSR, AVSR results in terms of WER (%) across different audio and video compression rates (e.g., (4,2)). The best results for each specific task, rate and dataset are shown in **bold**.

Method	ASR		VSR		AVSR				Avg
	(4)	(16)	(2)	(5)	(4,2)	(4,5)	(16,2)	(16,5)	
LRS2 Dataset									
Llama-AVSR [27]	3.3	4.3	26.9	30.0	**2.5**	2.6	3.9	4.6	9.8
Llama-MTSK [32]	**2.5**	**3.9**	**26.7**	28.5	**2.5**	2.5	3.7	4.0	**9.3**
Llama-MT	2.6	4.1	27.2	28.8	**2.5**	**2.4**	**3.5**	**3.9**	9.4
`Omni-AVSR-S`	2.8	5.0	27.8	28.5	2.7	2.6	3.8	4.0	9.6
`Omni-AVSR-T`	2.7	4.5	26.8	**28.3**	2.6	2.7	3.9	4.0	9.4
`Omni-AVSR-ST`	2.7	4.8	27.8	29.5	**2.5**	2.7	3.9	4.2	9.8
LRS3 Dataset									
Llama-AVSR [27]	1.1	2.0	27.4	29.5	1.1	1.2	2.0	2.1	8.3
Llama-MTSK [32]	**1.0**	2.0	26.9	27.8	**1.0**	**1.0**	1.9	2.0	8.0
Llama-MT	**1.0**	2.1	27.2	28.4	**1.0**	**1.0**	**1.8**	**1.9**	8.0
`Omni-AVSR-S`	1.1	2.4	**26.6**	27.4	1.1	**1.0**	1.9	2.0	**7.9**
`Omni-AVSR-T`	1.2	**1.9**	26.7	27.8	1.2	1.2	2.0	2.2	8.0
`Omni-AVSR-ST`	1.2	2.0	26.8	**27.1**	**1.0**	1.1	**1.8**	**1.9**	**7.9**

Table II: Computational cost analysis in terms of **1)** the number of trained models and **2)** LLM forward/backward passes required to cover all tasks and rates in training. Here, T denotes the number of tasks, while C_A/C_V denotes the number of *audio*/*video* rates.

Method	# Trained Models	# LLM F/B Passes
Llama-AVSR [27]	$C_A + C_V + C_AC_V$	$C_A + C_V + C_AC_V$
Llama-MTSK [32]	T	$C_A + C_V + C_AC_V$
Llama-MT	C_AC_V	$T(C_AC_V)$
`Omni-AVSR`	**1**	**T**

$\alpha(\mathbf{Z}_t W_{down})W_{up} + \alpha(\mathbf{Z}_t W^t_{down})W^t_{up}$. During **training**, Omni-LoRA-T and Omni-LoRA-ST activate all task-specific modules. At **inference**, however, *only* the module corresponding to the selected task is used, ensuring efficiency.

III. EXPERIMENTS AND RESULTS

III-A. Experiment Settings

Datasets. We conduct experiments on LRS2 [45] and LRS3 [46] datasets. LRS2 includes 225 hours of footage from BBC programs. LRS3 contains 433 hours of English video clips from TED talks.

Pre-Processing. We follow [18], [27], [32] for the datasets pre-processing. For video, we crop the mouth region of interests (ROIs) through a bounding box of 96×96. Each frame is normalised by subtracting the mean and dividing by the standard deviation of the training set. Audio data undergo z-normalisation per utterance.

`Omni-AVSR` Details. We use AV-HuBERT Large as the visual encoder and Whisper medium as the audio encoder. The projection layers consist of two linear layers with a ReLU activation in between. For the LLM backbone, we adopt LLaMA 3.2-1B [47] in our main experiments. Following prior work [27], [28], [32], both the LLM and video encoder are fine-tuned via LoRA modules applied to the query and value projection matrices with rank 64. We evaluate three `Omni-AVSR` variants, depending on the LoRA configuration used: `Omni-AVSR-S`, `Omni-AVSR-T`, and `Omni-AVSR-ST`.

Training/Inference Details. Following [18], [27], [32], we augment visual inputs through horizontal flipping, random cropping, and adaptive time masking, while for audio we only apply adaptive time masking. We define the textual prompts as in [27], [28], [32]: "`Transcribe {task_prompt} to text.`", where **`task_prompt`** $\in$ {"`speech`", "`video`", "`speech and video`"}. We set $\lambda_{\text{ASR}} = \lambda_{\text{AVSR}} = 1$ and $\lambda_{\text{VSR}} = 1.5$. We train our `Omni-AVSR` models for 8 epochs with the AdamW optimizer with cosine annealing scheduler and weight decay set to 0.1 using NVIDIA L40 GPUs. The learning rate is 1e-3. For decoding, we use beam search with a beam width of 15 and temperature of 0.6.

Table III: AVSR results on LRS3 across acoustic noise conditions.

Method	SNR (dB)				
	5	2.5	0	-2.5	-5
Compression rates: (4,2)					
Llama-AVSR [27]	2.6	4.1	4.8	12.1	19.1
Llama-MTSK [32]	2.5	3.9	4.8	11.7	18.5
Llama-MT	2.6	3.9	4.4	11.1	17.8
`Omni-AVSR-ST`	2.5	3.8	4.4	11.4	18.0
Compression rates: (16,5)					
Llama-AVSR [27]	4.2	5.8	6.5	14.9	22.1
Llama-MTSK [32]	3.8	5.5	6.0	14.0	20.5
Llama-MT	3.7	5.1	6.0	13.4	20.1
`Omni-AVSR-ST`	3.9	5.3	5.9	13.5	19.5

Table IV: Comparison with state-of-the-art methods using a single model for ASR, VSR, and AVSR on LRS3. ‡u-HuBERT is trained on LRS3 and VoxCeleb2, totaling 1759 hours.

Method	Train Par. (M)	Train Hours	WER ↓ ASR	VSR	AVSR
u-HuBERT [39]‡	325	1759	1.5	29.1	1.3
MultiAVSR [40]	274	433	2.4	31.1	2.5
USR [38]	171	433	1.9	34.3	1.6
`Omni-AVSR-ST` (4,2)	**58**	433	**1.2**	**26.8**	**1.0**
`Omni-AVSR-ST` (16,5)	**58**	433	2.0	27.1	1.9

Audio-Visual Granularities. For fair comparison with prior work [32], we adopt the same compression rates, chosen to capture a spectrum of efficiency–performance trade-offs at inference. Specifically, we use {4,16} for ASR, {2,5} for VSR, and their Cartesian product for AVSR, yielding four audio-visual configurations. Token compression is performed via *average pooling* [32].

Baselines. As shown in Tables I and III, we compare `Omni-AVSR` variants with three main approaches: 1) **Llama-AVSR** [27], which trains a separate model for each task and compression rate; 2) **Llama-MTSK** [32], which enables elastic inference by training on multiple rates but only within a single task; 3) **Llama-MT**, which supports multi-task (**MT**) training across ASR, VSR, and AVSR but requires a separate model for each rate. In contrast, `Omni-AVSR` unifies both elastic inference and multi-task learning within a single framework, subsuming these baselines as special cases. Additional comparisons with AVSR sota methods are provided in Section III-C.

III-B. Main Results

Table I reports the ASR/VSR/AVSR results of our three `Omni-AVSR` variants on LRS2 and LRS3. On LRS2, the task-specific variant `Omni-AVSR-T` achieves the best performance, while on LRS3 all three variants yield comparable results. This difference is likely due to the larger training set of LRS3, which enables lower WERs overall, particularly for ASR and AVSR. Compared with the baselines, we observe the following: **(1)** all `Omni-AVSR` variants consistently outperform Llama-AVSR, which requires a separate model per rate and task; **(2)** `Omni-AVSR-T` on LRS2, and all three variants on LRS3, match or surpass Llama-MTSK and Llama-MT, with `Omni-AVSR-S` and `-T` attaining average WERs as low as 7.9 across tasks on LRS3; **(3)** task-wise, `Omni-AVSR` particularly benefits VSR; and **(4)** performance trends remain consistent across compression rates. These results demonstrate that

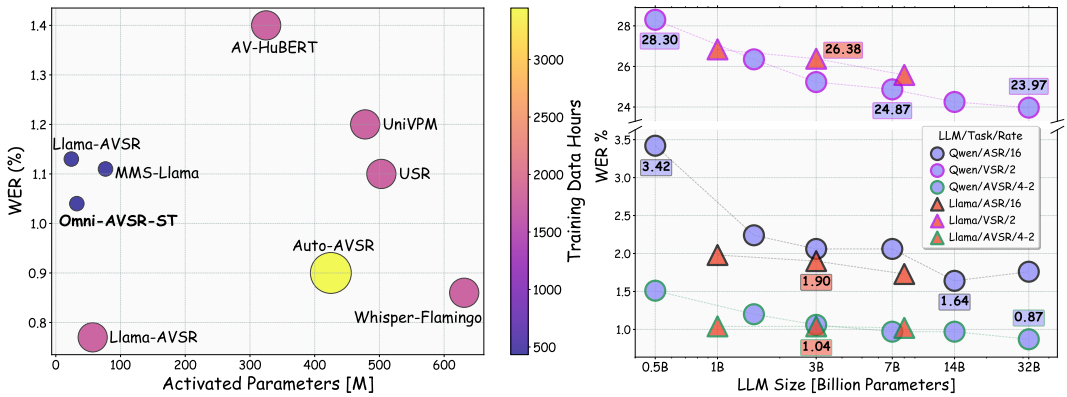

Fig. 2: **Left**: Comparison of Omni-AVSR-ST with state-of-the-art AVSR methods in terms of *WER*, *activated parameters*, and *training data hours* on LRS3. **Right**: *Scaling* trend of Omni-AVSR-ST when we increase the LLM size on LRS3.

Omni-AVSR delivers competitive or superior accuracy while unifying elastic inference and multi-task learning within a single framework.

Beyond delivering strong recognition performance, Omni-AVSR also offers significant computational advantages, as summarized in Table II. **(1)** Omni-AVSR requires training only a *single* model, independent of the number of tasks T (ASR, VSR, and AVSR in our case, so $T = 3$) and the number of audio C_A and video C_V compression rates ($C_A = C_V = 2$ in our setup). In contrast, the other three baselines train multiple models that scale with the number of compression rates (Llama-AVSR and Llama-MT) or with the number of tasks (Llama-MT). **(2)** We further compare the methods in terms of the number of forward/backward passes required over the LLM, since this constitutes the dominant computational cost during training. Llama-AVSR and Llama-MTSK must compute the loss separately for each compression rate and task, requiring C_A passes for ASR, C_V for VSR, and $C_A C_V$ for AVSR. Llama-MT trains one multi-task model for each audio–visual rate pair, which results in $T(C_A C_V)$ passes. In contrast, Omni-AVSR computes the loss only once per task, as it samples a single audio and video rate at each iteration, thus reducing the requirement to just T passes. Overall, Omni-AVSR *requires only a single model and substantially reduces overall training cost compared to all baselines*.

Results under Acoustic Noise. To evaluate the robustness of Omni-AVSR under *noisy conditions*, we inject babble noise from the NOISEX dataset [48] at varying SNRs. As shown in Table III, Omni-AVSR-ST consistently outperforms Llama-AVSR and Llama-MTSK, and remains competitive with Llama-MT across noise levels, often surpassing it at lower SNRs.

Comparison with Other Multi-task Methods.

In Table IV, we compare Omni-AVSR-ST with three state-of-the-art methods that train a single model for ASR, VSR, and AVSR: u-HuBERT [39], MultiAVSR [40], and USR [38]. Unlike Omni-AVSR, these methods do not support elastic inference. At the (4,2) compression setting, Omni-AVSR-ST achieves the best performance across all tasks while requiring significantly fewer parameters and surpassing u-HuBERT, despite the latter being trained on 1759 hours of data (LRS3 + VoxCeleb2 datasets). Even under the more extreme (16,5) compression, Omni-AVSR-ST maintains competitive results within a single set of weights.

III-C. Ablation Studies

AVSR Comparison with Sota Methods. Fig. 2 (left) presents a comparison of Omni-AVSR-ST with recent state-of-the-art approaches on LRS3 for the AVSR task. Baselines include UniVPM [49], USR [38], Whisper-Flamingo [17], Llama-AVSR [27], Auto-AVSR [18], AV-HuBERT [14], and MMS-Llama [29]. Omni-AVSR-ST (evaluated at audio-video rates of (4,2)) achieves competitive WERs while requiring substantially fewer parameters and training data hours than all baselines, within one consistent framework.

Table V: Ablation on the best values of ASR/VSR/AVSR weights.

λ_{ASR}	λ_{VSR}	λ_{AVSR}	ASR		VSR		AVSR	
			(4)	(16)	(2)	(5)	(4,2)	(16,5)
1	1	1	2.9	5.7	27.0	28.6	2.7	4.4
1	1.5	1	2.7	4.4	26.8	28.3	2.6	4.0
1	2	1	2.7	4.4	27.0	28.5	2.5	4.0

LLM Scaling Trend. We study how scaling the LLM size impacts performance in Fig. 2 (right). Specifically, we evaluate Llama 3.2–1B, 3B, and 3.1–8B [47], as well as Qwen 2.5–0.5B, 1.5B, 3B, 7B, 14B, and 32B [50]. Results are reported for ASR at audio rate 16 (black outline), VSR at video rate 2 (violet outline), and AVSR at (4,2) rates. As shown, performance improves with larger LLMs, with higher gains observed on more challenging tasks (e.g., VSR) or under higher compression (e.g., ASR at rate 16). However, larger models incur greater training cost, memory usage, and slower inference. Overall, LLMs in the 1–3B parameter range represent a favorable trade-off between accuracy and efficiency.

Optimal Task-specific Weights. In Table V, we analyze the impact of varying the loss weight coefficients in Eq. 1 for each task on LRS2. The best performance is obtained with $\lambda_{\text{ASR}} = \lambda_{\text{AVSR}} = 1$ and $\lambda_{\text{VSR}} = 1.5$. Since VSR is the most challenging of the three tasks, assigning it a higher weight leads to improved overall results.

IV. CONCLUSION

In this work, we introduced Omni-AVSR, the first unified audio-visual LLM that jointly supports ASR, VSR, and AVSR while enabling elastic inference under a single set of weights. By combining efficient matryoshka-based multi-granularity training with LoRA adaptation strategies, Omni-AVSR achieves strong performance while reducing training and deployment costs. Experiments on LRS2 and LRS3 show that Omni-AVSR matches or surpasses state-of-the-art baselines, remains robust in noisy conditions, and delivers favorable trade-offs when scaling LLM size. Furthermore, Omni-AVSR provides significant computational savings, requiring only one model and a reduced number of LLM passes during training.

V. REFERENCES

[1] D. Amodei *et al.*, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *ICML*, 2016.

[2] S. Watanabe *et al.*, "Hybrid ctc/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, pp. 1240–1253, 2017.

[3] R. Prabhavalkar *et al.*, "End-to-end speech recognition: A survey," *IEEE/ACM TASLP*, vol. 32, pp. 325–351, 2023.

[4] S. Dupont and J. Luettin, "Audio-visual speech modeling for continuous speech recognition," *IEEE transactions on multimedia*, vol. 2, pp. 141–151, 2000.

[5] G. Potamianos, C. Neti, J. Luettin, and I. Matthews, "Audio-visual automatic speech recognition: An overview," *Issues in visual and audio-visual speech processing*, vol. 22, p. 23, 2004.

[6] S. Petridis *et al.*, "End-to-end audiovisual speech recognition," in *ICASSP*, 2018.

[7] K. Noda *et al.*, "Audio-visual speech recognition using deep learning," *Applied intelligence*, vol. 42, pp. 722–737, 2015.

[8] Y. Mroueh *et al.*, "Deep multimodal learning for audio-visual speech recognition," in *ICASSP*, 2015.

[9] S. Petridis *et al.*, "Audio-visual speech recognition with a hybrid ctc/attention architecture," in *SLT*, 2018.

[10] A. Vaswani *et al.*, "Attention is all you need," *NeurIPS*, 2017.

[11] T. Afouras *et al.*, "Deep audio-visual speech recognition," in *TPAMI*, vol. 44, 2018, pp. 8717–8727.

[12] P. Ma *et al.*, "End-to-end audio-visual speech recognition with conformers," in *ICASSP*, 2021.

[13] D. Serdyuk *et al.*, "Audio-visual speech recognition is worth $32 \times 32 \times 8$ voxels," in *ASRU*, 2021.

[14] B. Shi *et al.*, "Learning audio-visual speech representation by masked multimodal cluster prediction," in *ICLR*, 2022.

[15] ——, "Robust self-supervised audio-visual speech recognition," in *Interspeech*, 2022.

[16] A. Haliassos *et al.*, "Jointly learning visual and auditory speech representations from raw data," in *ICLR*, 2023.

[17] A. Rouditchenko *et al.*, "Whisper-flamingo: Integrating visual features into whisper for audio-visual speech recognition and translation," in *Interspeech*, 2024.

[18] P. Ma *et al.*, "Auto-avsr: Audio-visual speech recognition with automatic labels," in *ICASSP*, 2023.

[19] J. Hong *et al.*, "Watch or listen: Robust audio-visual speech recognition with visual corruption modeling and reliability scoring," in *CVPR*, 2023.

[20] C. Chen *et al.*, "Leveraging modality-specific representations for audio-visual speech recognition via reinforcement learning," in *AAAI*, 2023.

[21] S. Bai *et al.*, "Qwen2. 5-vl technical report," *arXiv preprint arXiv:2502.13923*, 2025.

[22] E. Fini *et al.*, "Multimodal autoregressive pre-training of large vision encoders," in *CVPR*, 2025.

[23] H. Yao *et al.*, "Dense connector for mllms," in *NeurIPS*, 2024.

[24] Y. Fathullah *et al.*, "Prompting large language models with speech recognition abilities," in *ICASSP*, 2024.

[25] A. Goel *et al.*, "Audio flamingo 3: Advancing audio intelligence with fully open large audio language models," *arXiv preprint arXiv:2507.08128*, 2025.

[26] B. Wu *et al.*, "Step-audio 2 technical report," *arXiv preprint arXiv:2507.16632*, 2025.

[27] U. Cappellazzo *et al.*, "Large language models are strong audio-visual speech recognition learners," in *ICASSP*, 2025.

[28] ——, "Scaling and enhancing llm-based avsr: A sparse mixture of projectors approach," in *Interspeech*, 2025.

[29] J. Yeo *et al.*, "Mms-llama: Efficient llm-based audio-visual speech recognition with minimal multimodal speech tokens," in *ACL Findings*, 2025.

[30] ——, "Zero-avsr: Zero-shot audio-visual speech recognition with llms by learning language-agnostic speech representations," in *ICCV*, 2025.

[31] ——, "Where visual speech meets language: Vsp-llm framework for efficient and context-aware visual speech processing," in *EMNLP Findings*, 2024.

[32] U. Cappellazzo *et al.*, "Adaptive audio-visual speech recognition via matryoshka-based multimodal llms," in *ASRU*, 2025.

[33] X. Zhu *et al.*, "Uni-med: a unified medical generalist foundation model for multi-task learning via connector-moe," in *NeurIPS*, 2024.

[34] Z. Li *et al.*, "UnifiedMLLM: Enabling unified representation for multi-modal multi-tasks with large language model," in *NAACL Findings*, 2025.

[35] J. Wu *et al.*, "Omni-smola: Boosting generalist multimodal models with soft mixture of low-rank experts," in *CVPR*, 2024.

[36] J. Xu *et al.*, "Qwen2. 5-omni technical report," *arXiv preprint arXiv:2503.20215*, 2025.

[37] S. Zhang *et al.*, "Stream-omni: Simultaneous multimodal interactions with large language-vision-speech model," *arXiv preprint arXiv:2506.13642*, 2025.

[38] A. Haliassos *et al.*, "Unified speech recognition: A single model for auditory, visual, and audiovisual inputs," in *NeurIPS*, 2024.

[39] W. Hsu *et al.*, "u-hubert: Unified mixed-modal speech pretraining and zero-shot transfer to unlabeled modality," in *NeurIPS*, 2022.

[40] S. Torrie *et al.*, "Multiavsr: Robust speech recognition via supervised multi-task audio–visual learning," *Electronics*, 2025.

[41] A. Kusupati *et al.*, "Matryoshka representation learning," in *NeurIPS*, 2022.

[42] M. Cai *et al.*, "Matryoshka multimodal models," in *ICLR*, 2025.

[43] A. Radford *et al.*, "Robust speech recognition via large-scale weak supervision," in *ICML*, 2023.

[44] E. Hu *et al.*, "Lora: Low-rank adaptation of large language models." in *ICLR*, 2022.

[45] J. Chung *et al.*, "Lip reading sentences in the wild," in *CVPR*, 2017.

[46] T. Afouras *et al.*, "Lrs3-ted: a large-scale dataset for visual speech recognition," *arXiv preprint arXiv:1809.00496*, 2018.

[47] A. Dubey *et al.*, "The llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, 2024.

[48] A. Varga, "Assessment for automatic speech recognition: Ii. noisex-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Elsevier Speech Commun*, 1992.

[49] Y. Hu *et al.*, "Hearing lips in noise: Universal viseme-phoneme mapping and transfer for robust audio-visual speech recognition," *arXiv preprint arXiv:2306.10563*, 2023.

[50] A. Yang *et al.*, "Qwen2.5 technical report," *arXiv preprint arXiv:2412.15115*, 2024.