OMNIINSTRUCT: INSTRUCTION-GUIDED AUDIO-VIDEO EDITING VIA UNIVERSAL DATASET AND REWARD OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Instruction-guided audio-video editing aims to jointly modify visual and auditory content according to natural-language instructions while preserving source content and cross-modal coherence. While existing methods have achieved promising progress, two challenges still remain: (1) the absence of a large-scale, high-quality dataset covering diverse audio-video editing tasks, and (2) the inadequate generalization ability of editing models on open-domain scenarios. To tackle the first challenge, we introduce OmniInstruct-500K, a high-quality dataset comprising 500K audio-video editing triplets across local, global, and speech editing categories constructed through an automated pipeline. Based on this dataset, we propose OmniInstruct, an end-to-end framework that first adapts a pretrained joint audio-video generator into an instruction-guided editor through task-aware supervised fine-tuning. To tackle the second challenge, we further introduce Edit-NFT, a reinforcement learning post-training method that employs editing-oriented rewards for instruction following, source preservation, perceptual quality, and audio-visual synchronization, and routes reward-wise advantages to the corresponding modality branches. Extensive experiments show our dataset and framework surpass prior methods on quantitative and qualitative metrics, yielding stronger instruction following, editing quality and open-domain generalization.

1 INTRODUCTION

Building on recent breakthroughs in modern text-to-video (T2V) frameworks (Yang et al., 2025b; Kong et al., 2024; Wan et al., 2025), video editing techniques have rapidly advanced for practical scenarios such as film production and advertising. Typical video editing methods (Geyer et al., 2024; Cheng et al., 2024; Jiang et al., 2025) often focus on modifying visual content while neglecting audio effects. For instance, when we use them to alter a video's background from an indoor scene to a seaside setting, the edited results struggle to provide accompanying audio feedback. To address this limitation, joint audio-video editing (Lin et al., 2026; Guo et al., 2026) has emerged as a promising research direction and gained rapid popularity. In particular, the instruction-guided video editing paradigm offers the most user-friendly interaction; thus, this work focuses on this setting.

Instruction-guided audio-video editing (Lin et al., 2026; Zheng et al., 2026a;b; Chen et al., 2026b) requires modifying both visual content and corresponding audio according to user instructions. Despite many promising advances, there remain two notable challenges. First, the existing source-target-instruction training pair synthesis pipelines either rely entirely on mask-guided generation, which confines edits to localized regions (Zheng et al., 2026a;b), or focus solely on speech editing (Chen et al., 2026b). This makes it difficult for more general edits, such as changing the global environment (e.g., weather) and background. Second, current solutions typically model video editing as a supervised translation from an input video and editing instruction to an edited output. However, generally speaking, reliance on training data with limited editing samples fundamentally limits their generalization, leading to performance degradation under open-domain settings.

To deal with the first challenge, this paper proposes an automatic data curation pipeline, which brings a large-scale, high-quality, open-domain dataset named OmniInstruct-500K, tailored for instruction-guided joint audio-video editing. As seen in Fig. 1, the dataset comprises approximately 500K

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

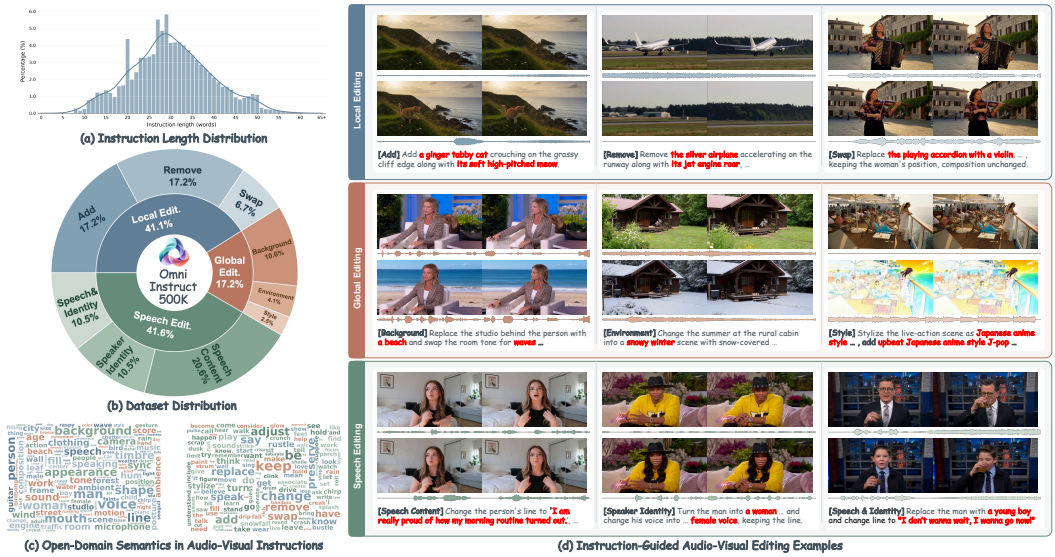


Figure 1: Overview of OmniInstruct-500K. We show instruction length and editing-type distributions, open-domain instruction semantics, and representative instruction-guided editing examples.

editing triplets and covers nine distinct types spanning local editing, global editing, and speech editing. Our automated data construction pipeline integrates rigorous multi-dimensional data filtering, LLM-powered diverse instruction generation, and modality-decoupled expert editing pipelines to produce fine-grained editing samples with rich meta-information. We further adopt a hybrid verification scheme to ensure the visual-audio fidelity, temporal synchronization, instruction following and source consistency of the edited samples.

Furthermore, this paper introduces OmniInstruct, an end-to-end framework for instruction-guided audio-video editing, which improves model performance by two designs. First, we adapt a pre-trained joint audio-video generator LTX-2.3 (HaCohen et al., 2026) into an instruction-guided editor through task-aware supervised fine-tuning, where Mixture-of-Experts LoRA modules are embedded within the DiT blocks. Second, inspired by DiffusionNFT (Zheng et al., 2026c) and OmniNFT (Zhang et al., 2026a), we introduce a post-training stage named Edit-NFT, which further optimizes the editor toward successful edits via modality-aware reinforcement learning. Concretely, we sample a group of edited candidates for each source pair and instruction, score video and audio separately on instruction following, source consistency, and perceptual quality, and add one shared audio–video synchronization term. Group-relative advantages are then routed to the responsible modality rather than pooled into a single scalar. In summary, our main contributions are:

- We build OmniInstruct-500K, a large-scale open-domain dataset with 500K diverse audio-video editing triplets spanning nine editing types.
- We propose an end-to-end instruction-guided editing framework, dubbed OmniInstruct, which performs a modality-routed reinforcement learning post-training paradigm after supervised fine-tuning.
- Extensive experiments demonstrate that our dataset and framework yield clear improvements over strong baselines, particularly in instruction following and editing quality, and enhance the open-domain generalization of instruction-guided audio-video editing.

2 Related Works

Instruction-guided Video Editing. Instruction-guided editing originates from the image domain (Brooks et al., 2023; Zhang et al., 2023a; Labs et al., 2025; Wu et al., 2025a), where models follow textual instructions to edit an image. Recently, instruction-guided video editing (Zi et al., 2026; Long et al., 2026; Bai et al., 2026; Wu et al., 2025b; He et al., 2025) has attracted growing attention owing to its intuitive interaction paradigm. For instance, Señorita-2M (Zi et al., 2026) constructs a large-scale editing dataset by employing four expert models, each specialized in a dis-

tinct editing task. CoinVE-200K (Long et al., 2026) supports multiple editing intents through joint understanding within a single video.

Joint Audio-Video Generation and Editing. Advances in image (AI, 2022a; Esser et al., 2024; FLUX, 2024; AI, 2022b; OpenAI, 2026) and video (Yang et al., 2025b; Kong et al., 2024; Wan et al., 2025; Zhang et al., 2023b) generation have spurred growing interest in audio-video synthesis. Early approaches typically employed cascaded Audio-to-Video (A2V) (Luo et al., 2023; Gao et al., 2025) or Video-to-Audio (V2A) (Zhang et al., 2026b; Cheng et al., 2025; Benita et al., 2025) pipelines. More recently, closed-source systems (OpenAI, 2025; DeepMind, 2025; ByteDance, 2026) have demonstrated strong synchronized audio-video generation through joint modeling. Open-source models such as OVI (Low et al., 2025) and LTX-2 (HaCohen et al., 2026) adopt twin-backbone DiT architectures (Peebles & Xie, 2023), while MiniMax-H3 (MiniMaxAI, 2026) supports joint generation but remains limited in instruction-guided editing.

Parallel efforts have explored joint audio-video editing. AvED (Lin et al., 2026) proposes zero-shot cross-modal denoising guided by textual prompts and patch-level audio-visual cues. CoherentAVEdit (Ishii et al., 2025) follows a cascaded pipeline that first edits the video and then adapts the audio to the visual changes. AVI-Edit (Zheng et al., 2026a) targets audio-synchronized video instance editing using a granularity-aware mask refiner. More recent methods, InstructAV2AV (Zheng et al., 2026b) and JAVEdit (Chen et al., 2026b), develop end-to-end instruction-guided joint editing pipelines and construct datasets containing 80K and 100K editing pairs, respectively.

Reinforcement Learning for Generative Models. Motivated by the success of GRPO (Guo et al., 2025) in large language models, reinforcement learning has emerged as an effective post-training approach for improving the visual fidelity and semantic consistency of generative models (Luo et al., 2026; Jiang et al., 2026). Reward-guided policy optimization has been applied to both image (Jiang et al., 2026; Liu et al., 2026a) and video generation (Xue et al., 2025; Zhang et al., 2026a). DiffusionNFT (Zheng et al., 2026c) introduces Diffusion Negative-aware FineTuning, an online RL paradigm that directly optimizes diffusion models along the forward process through flow matching. OmniNFT (Zhang et al., 2026a) further extends negative-aware fine-tuning to joint audio-visual generation. Building on these advances, we extend modality-routed reinforcement learning from joint generation to instruction-guided audio-video editing with editing-oriented rewards.

3 OMNIINSTRUCT-500K

We present OmniInstruct-500K, a large-scale and high-quality dataset containing 500K triplets for instruction-guided audio-video editing. OmniInstruct-500K spans nine editing types across local, global, and speech editing categories and covers diverse real-world audio-visual scenarios. Detailed definitions and representative examples of these editing types are provided in Appendix B. As illustrated in Fig. 2, our automated data construction pipeline comprises four stages.

Data Collection and Processing. We collect speech-centric videos from SpeakerVid (Zhang et al., 2025) and general sound-event videos from VGGSound (Chen et al., 2020) and AudioSet (Gemmeke et al., 2017), and standardize them into five-second, 1280×720 clips. We then filter out clips with poor visual quality (aesthetics, motion, sharpness) or audio fidelity, as well as those with semantic or temporal audio-visual misalignment. To recover valuable long-tail semantics from low-quality clips, we rewrite their captions with an LLM (Yang et al., 2025a), regenerate the clips with MiniMax-H3 (MiniMaxAI, 2026), and re-apply the same filtering.

Instruction Generation. For each source clip, Qwen3-Omni (Xu et al., 2025) extracts structured annotations, including salient subjects, visual and audio descriptions, and ASR transcripts. Based on these annotations, Qwen3-235B (Yang et al., 2025a) selects a suitable editing type and generates a joint audio-video editing instruction along with fine-grained auxiliary annotations for subsequent expert editing. Finally, Qwen3-Omni (Xu et al., 2025) verifies the consistency between the source content and the generated instruction, filtering out infeasible or hallucinated cases.

Adaptive Expert Pipelines. We decompose joint audio-video editing into visual and audio operations along a shared timeline. Given the source video, source audio, and editing instruction, an editing agent interprets the instruction, formulates an editing plan, and routes the task to appropriate visual and audio experts to produce coherent editing targets. For local removal, Grounded-SAM-2 (Ren et al., 2024) localizes the target, while MiniMax-Remover (Zi et al., 2025) and SAM-

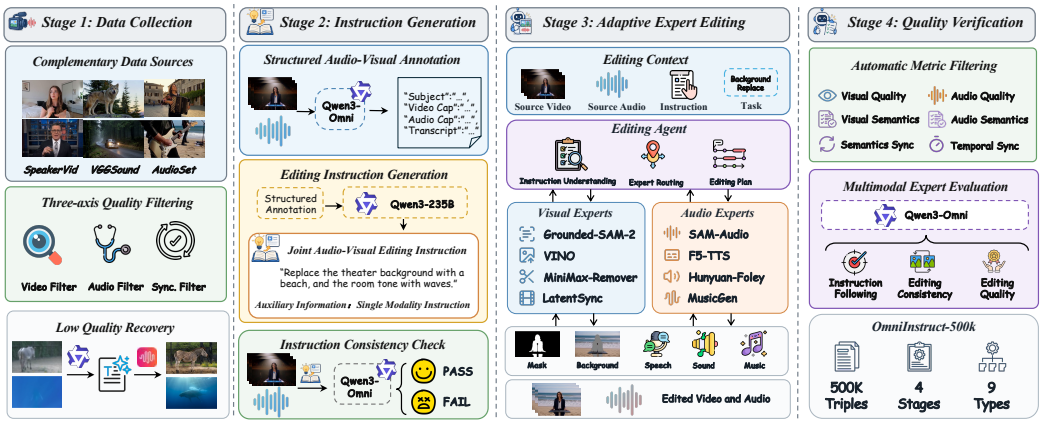


Figure 2: **Overview of the automated data construction pipeline for OmniInstruct-500K.** Stage 1 performs data collection and quality filtering. Stage 2 generates diverse editing instructions and verifies consistency. Stage 3 employs an editing agent to coordinate adaptive visual and audio experts for target synthesis. Stage 4 applies two-level quality verification to construct the final dataset.

Audio (Shi et al., 2025) remove its visual and auditory content, respectively. For background replacement, VINO (Chen et al., 2026a) edits the visual background, and HunyuanVideo-Foley (Shan et al., 2025) synthesizes the corresponding ambient sound, which is then mixed with the foreground audio isolated by SAM-Audio. For speech-content replacement, F5-TTS (Chen et al., 2025) synthesizes the target speech using the source speaker's timbre, and LatentSync (Li et al., 2024) synchronizes the lip motion with the edited speech. The remaining editing types are handled by task-specific combinations of these experts, e.g., MusicGen (Copet et al., 2024) synthesizes style-specific background music for style transfer.

Editing Quality Verification. We employ two-level quality verification. First, VTSS (Wang et al., 2024), AudioBox-Aesthetic (Tjandra et al., 2025), CLIP (Radford et al., 2021), CLAP (Wu et al., 2023), ImageBind (Girdhar et al., 2023), and SyncNet (Chung & Zisserman, 2016) assess visual and audio quality, semantic alignment, and cross-modal synchronization. Second, Qwen3-Omni evaluates instruction following, editing consistency, and editing quality. Samples passing both levels form OmniInstruct-500K, with quantitative comparisons provided in Appendix B.

4 METHODS

This paper proposes an end-to-end instruction-guided framework dubbed OmniInstruct, which can perform joint editing on both visual content and corresponding audio according to user instructions. The section is organized as follows: Sec. 4.1 revisits the preliminaries of joint audio–video flow matching, DiffusionNFT (Zheng et al., 2026c), as well as OmniNFT (Zhang et al., 2026a). Sec. 4.2 illustrates our instruction-based supervised fine-tuning, which employs a Mixture-of-Experts LoRA (MoE-LoRA) module for task-aware adaptation. Sec. 4.3 gives the details of our editing-oriented reward optimization based on OmniNFT (Zhang et al., 2026a), which aims to improve instruction following, source consistency, perceptual quality, and audio-visual alignment.

4.1 PRELIMINARIES

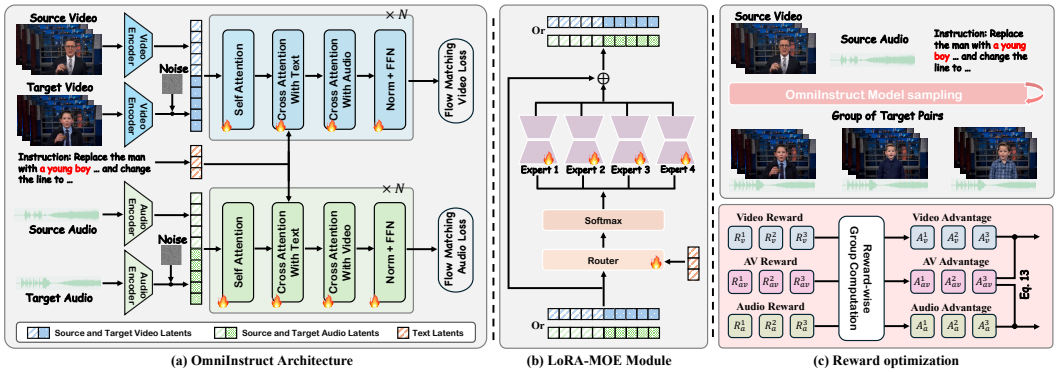
Joint Audio–video Flow Matching. The video latent x^v and the audio latent x^a evolve along a common flow-matching path in the joint audio-video generation model. Two modalities are corrupted by independent standard Gaussians $x_1^m \sim \mathcal{N}(0, I)$, yet they share a single time parameter $t \in [0, 1]$:

$$x_t^m = (1 - t)\, x_0^m + t\, x_1^m, \qquad m \in \{v, a\}. \tag{1}$$

A two-stream network $v_\theta = (v_\theta^v, v_\theta^a)$ estimates one velocity per modality, and the streams exchange information through cross-modal attention. At inference, the two trajectories are integrated synchronously with a deterministic solver,

$$\mathrm{d}x_t^m = v_\theta^m(x_t^v, x_t^a, t, c)\, \mathrm{d}t, \qquad m \in \{v, a\}, \tag{2}$$

where c denotes the text condition.



(a) OmniInstruct Architecture (b) LoRA-MOE Module (c) Reward optimization

Figure 3: (a) OmniInstruct architecture. We extend the joint audio-video generation model LTX 2.3 by using source video and audio as clean reference context. The visual and audio VAE encoders project the source video s_v and audio s_a into tokens, which are then concatenated with the noised latent tokens from target video t_v and audio t_a, respectively. (b) LoRA-MOE Module. We employ a four-expert MoE-LoRA for task-aware SFT. (c) Reward optimization. We sample a group of edited candidates for each source pair and instruction, score video, audio and audio–video synchronization term separately. Group-relative advantages are then routed to the responsible modality.

Diffusion Negative-aware Fine-Tuning (DiffusionNFT). DiffusionNFT (Zheng et al., 2026c) is an online RL paradigm that optimizes diffusion models directly on the forward process via flow matching. Let v_θ be the current velocity predictor and v^{old} the velocity predictor of the behavior policy that produced the rollout. A group-normalized advantage A is mapped to an optimality probability $r \in [0, 1]$:

$$r = \frac{1}{2} + \frac{1}{2}\,\text{clip}\big[A, -1, 1\big], \qquad A = \frac{R - \mathbb{E}_{\pi^{\text{old}}(\cdot|c)}[R]}{Z_c}, \tag{3}$$

where R denotes the raw reward and $Z_c > 0$ is a normalization factor. The implicit positive and negative policies are defined as

$$v_\theta^+ = (1-\beta)\, v^{\text{old}} + \beta\, v_\theta, \quad v_\theta^- = (1+\beta)\, v^{\text{old}} - \beta\, v_\theta. \tag{4}$$

The model is pulled toward the high-reward policy and pushed away from the low-reward one:

$$\mathcal{L}_{NFT} = \mathbb{E}\big[r\, \|v_\theta^+ - v\|_2^2 + (1-r)\, \|v_\theta^- - v\|_2^2\big], \tag{5}$$

where $v = x_1 - x_0$ is the flow-matching target velocity.

Recently, OmniNFT (Zhang et al., 2026a) extends DiffusionNFT to joint audio-video generation by separately computing video, audio, and synchronization advantages over the same group of G rollouts. It routes the unimodal advantages to their respective branches and shares the synchronization advantage across both:

$$\widetilde{A}_v = A_v + A_{av}, \qquad \widetilde{A}_a = A_a + A_{av}. \tag{6}$$

Each branch then uses its routed advantage $\widetilde{A}_m$ to compute the corresponding optimality probability.

4.2 Instruction-based Supervised Fine-Tuning

We formulate joint audio-video editing as a reference-conditioned audio-video generation problem. Each training sample consists of a source video and audio clip $\{s_v, s_a\}$, a target video and audio clip $\{t_v, t_a\}$, and an editing instruction t_{edit}. The target video frames and target audio are encoded into latent representations by 3D and 1D VAE encoders of LTX backbone (HaCohen et al., 2026) separately, which can be expressed as: $\{z_t^v, z_t^a\} = \{E_v(t_v), E_a(t_a)\}$. The source video and source audio are also encoded into latent tokens and used as clean in-context conditions during the learning of target denoising process. The text instruction is encoded by text encoder (Google, 2025) and provided as semantic conditioning.

As depicted in Fig. 3(a), we adapt the dual-stream LTX-2.3 backbone into an instruction-guided editor through supervised fine-tuning. Specifically, we sample a diffusion timestep and corrupt both the target video and audio latents with Gaussian noise. The clean reference source video and audio

latents are concatenated with the noisy target latents and assigned zero noise level, so that they serve as conditional context rather than denoising targets. The multimodal DiT jointly processes the noisy target tokens, clean source tokens, and text tokens, and predicts the flow-matching targets for both video and audio. We discard the predictions corresponding to the reference tokens and compute the training loss only on the target video and target audio tokens.

For parameter-efficient adaptation, we insert modality-specific LoRA adapters into the attention and feed-forward layers of the video and audio branches. However, standard LoRA provides limited task specialization across heterogeneous local, global, and speech editing tasks. To enable task-aware adaptation, we employ a lightweight MoE-LoRA module in the attention output projection, as illustrated in Fig. 3(b).

For modality $m \in \{v, a\}$, let H^m denote the current latent features and $\bar{C}$ the pooled instruction feature. We map $\bar{C}$ through a modality-specific projector P_m and concatenate it with each latent token. The token-wise routing weights are computed as

$$G^m = \mathrm{softmax}\big(R^m([H^m; P_m(\bar{C})])\big), \tag{7}$$

where R^m is a linear router and the softmax is applied over the expert dimension. Thus, routing depends jointly on the latent content and editing instruction. The frozen attention output projection is augmented with the routed LoRA experts:

$$O^m = H^m W_O^m + \frac{\alpha}{r_e} \sum_{i=1}^{N} G_i^m \odot \mathcal{E}_i^m(H^m), \tag{8}$$

where G_i^m is the token-wise weight and $\mathcal{E}_i^m$ denotes the i-th rank-r_e LoRA expert. We use $N = 4$ experts with $r_e = 32$ and dense routing.

4.3 Edit-NFT: Editing-Oriented Reward Optimization

Although supervised fine-tuning establishes the basic editing capability, it still shows limited generalization. To deal with this, we introduce an editing-oriented reinforcement learning post-training method, dubbed Edit-NFT. As observed in OmniNFT (Zhang et al., 2026a), joint audio-video generation suffers from advantage inconsistency: a candidate may exhibit high visual quality but poor audio fidelity, such that aggregating all rewards into a single advantage can obscure modality-specific feedback. Therefore, we extend the modality-wise advantage routing of OmniNFT from generation to the editing setting. Specifically, for each source pair and editing instruction, we sample a group $\mathcal{G}$ of G candidate edits using different random seeds. We evaluate these candidates using seven reward channels and compute a group-relative advantage for each channel independently. We then route video-specific advantages to the video branch, audio-specific advantages to the audio branch, and the shared synchronization advantage to both.

Video rewards. The video branch is scored by three rewards: *instruction following* (whether the requested editing is applied), *source consistency* (whether unedited content is kept), and *perceptual quality* (whether the result looks natural). We encourage the edited video x_v to be more aligned with edited video descriptions T_{edit} than source descriptions T_{src} by using the following objective:

$$R_{\mathrm{IF}}^v = \mathrm{CLIP}(x_v, T_{\mathrm{edit}}^v) - \mathrm{CLIP}(x_v, T_{\mathrm{src}}^v), \tag{9}$$

where $\mathrm{CLIP}(\cdot,\cdot)$ denotes the CLIP (Radford et al., 2021) similarity. Copying the source video without applying instruction would receive a low score. Besides, *source consistency* measures the similarity between the source and edited videos over visual content that should remain unchanged, denoted by $R_{\mathrm{SC}}^v = \mathrm{CLIP}(s_v, x_v)$. R_{SC}^v is counted only after the video side already shows sufficient instruction following, so a near-copy cannot collect a high consistency score. Then, VideoAlign score (Liu et al., 2025) is used as reward to evaluate *perceptual quality*: $R_{\mathrm{Q}}^v = \mathrm{VideoAlign}(x_v)$.

Audio rewards. The audio branch mirrors the video branch, where *instruction following* contrasts the audio–text similarity against the post-edit and pre-edit audio descriptions,

$$R_{\mathrm{IF}}^a = \mathrm{CLAP}(x_a, T_{\mathrm{edit}}^a) - \mathrm{CLAP}(x_a, T_{\mathrm{src}}^a), \tag{10}$$

where $\mathrm{CLAP}(\cdot,\cdot)$ denotes the CLAP (Wu et al., 2023) similarity. Besides, *source consistency* $R_{\mathrm{SC}}^a = \mathrm{Sim}_a(s_a, x_a)$ measures the content and speaker similarity to the source audio, combining CLAP and WavLM (Chen et al., 2022) embeddings. Then, Audiobox Aesthetics (Tjandra et al., 2025) is employed as the reward for audio *perceptual quality*: $R_{\mathrm{Q}}^a = \mathrm{AES}(x_a)$.

Table 1: Quantitative comparison with state-of-the-art methods on OmniInstruct-Bench. The best and second-best results are highlighted in **bold** and underlined, respectively.

Method	Instruction Following			Source Consistency		Editing Quality				
	CLIP ↑	*CLAP* ↑	*WER* ↓	*SSIM* ↑	*LPAPS* ↓	*FVD* ↓	*FAD* ↓	*LSE-C* ↑	*LSE-D* ↓	*DeSync* ↓
AvED	23.31	21.81	0.69	70.46	4.21	284.9	3.04	1.02	12.81	64.93
CoherentAVEdit	25.06	29.22	1.13	76.35	5.00	267.9	3.62	1.92	11.67	58.45
AVI-Edit	23.87	20.03	0.26	**77.46**	3.96	351.6	8.60	3.50	10.03	62.60
InstructAV2AV	25.15	29.25	0.54	74.10	4.10	277.9	2.96	4.09	10.27	52.93
MiniMax-H3	24.75	31.50	0.47	51.66	3.98	318.8	2.70	**7.85**	**7.28**	**36.93**
OmniInstruct (Ours)	**25.80**	**32.56**	**0.19**	75.10	**3.91**	**248.8**	**2.62**	5.76	8.81	52.67

Audio–video synchronization rewards. Synchronization concerns the relation between the two edited streams and cannot be attributed to either modality alone. We score the edited pair (x_v, x_a) with audio-video alignment models, denoted R_{AV}. Here, SyncNet (Chung & Zisserman, 2016) is adopted when a visible speaking face is present, and Synchformer (Iashin et al., 2024) otherwise:

$$R_{\mathrm{AV}} = \begin{cases} \mathrm{SyncNet}(x_v, x_a), & \text{if } s_v \text{ involves a visible speaking face,} \\ \mathrm{Synchformer}(x_v, x_a), & \text{otherwise.} \end{cases} \tag{11}$$

Group-relative advantages and modality routing. Each reward channel k is standardized inside the group, which can be expressed as:

$$A_k^{(i)} = (R_k^{(i)} - \mu_k^{\mathcal{G}})/(\sigma_k^{\mathcal{G}} + \epsilon), \tag{12}$$

where $\mu_k^{\mathcal{G}}$ and $\sigma_k^{\mathcal{G}}$ are the mean and standard deviation of channel k over group $\mathcal{G}$. The advantage asks only which of the G attempts is better on this edit, not how good a candidate is in isolation.

As seen in Fig. 3(c), the three video rewards supervise only the video branch, the three audio rewards only the audio branch, and synchronization is shared:

$$\begin{aligned} \tilde{A}_v^{(i)} &= w_{\mathrm{IF}}^v A_{\mathrm{IF}}^{v,(i)} + w_{\mathrm{SC}}^v A_{\mathrm{SC}}^{v,(i)} + w_{\mathrm{Q}}^v A_{\mathrm{Q}}^{v,(i)} + w_{\mathrm{AV}} A_{\mathrm{AV}}^{(i)}, \\ \tilde{A}_a^{(i)} &= w_{\mathrm{IF}}^a A_{\mathrm{IF}}^{a,(i)} + w_{\mathrm{SC}}^a A_{\mathrm{SC}}^{a,(i)} + w_{\mathrm{Q}}^a A_{\mathrm{Q}}^{a,(i)} + w_{\mathrm{AV}} A_{\mathrm{AV}}^{(i)}. \end{aligned} \tag{13}$$

The weights w are reward balancing weights.

5 Experiments

5.1 Experimental Setup

Implementation Details. We fine-tune LTX-2.3 (HaCohen et al., 2026), a 22B-parameter joint audio–video diffusion transformer, on paired editing videos drawn from our collected dataset with a base resolution of 480×864. Training follows a rectified flow-matching formulation (Liu et al., 2023; Lipman et al., 2023) and applies equal loss weights to the audio and video branches ($\lambda_{\mathrm{v}}=\lambda_{\mathrm{a}}=1$). We optimize with AdamW (Loshchilov & Hutter, 2017) using a learning rate of 1×10^{-4}, weight decay 1×10^{-2}, and β_1=0.9, β_2=0.999. Training runs for one epoch with a per-device batch size of 1 on 64 GPUs. At inference, we run 30 Euler (Karras et al., 2022) steps to synthesize 121 frames at 24 fps, applying CFG (Ho & Salimans, 2021) of 4.0 and 3.0 for the video and audio branches, respectively. Further details of MoE-LoRA and Edit-NFT training are provided in Appendix A.

Evaluation Protocol. We evaluate our model from three dimensions. (i) Instruction Following. We adopt *CLIP* to evaluate text-video alignment using CLIP-B (Radford et al., 2021). *CLAP* (Wu et al., 2023) is adopted to quantify text-audio alignment. For speech editing, we leverage Whisper (Radford et al., 2023) to transcribe the generated audio, based on which the *WER* metric is computed to measure word-level prediction errors. (ii) Source Consistency. *SSIM* (Wang et al., 2004) is utilized to measure background consistency and preservation. We further adopt *LPAPS* (Iashin & Rahtu, 2021) to evaluate perceptual audio consistency between the source and edited audio. (iii) Editing Quality. We use *FVD* (Unterthiner et al., 2018) and *FAD* (Kilgour et al., 2018) to quantify the overall quality of generated videos and audios. We utilize *DeSync* (Liu et al., 2026b) to quantify the



Figure 4: Qualitative comparison results with state-of-the-art methods for audio-video joint editing.

misalignment between audio and video streams. Furthermore, we adopt *LSE-D* (distance score) and *LSE-C* (confidence score) to assess lip synchronization (Prajwal et al., 2020).

OmniInstruct-Bench. We construct a benchmark of 300 curated samples covering diverse editing types and visual and acoustic scenes. Each sample has human-written annotations, including the editing instruction, audio-visual descriptions, and speech transcripts when applicable. All samples are manually inspected and are disjoint from the training set. Detailed descriptions of the compared baselines are provided in Appendix A.

5.2 Comparison with State-of-the-Art Methods

Qualitative Comparison. As shown in Fig. 4, we compare OmniInstruct with existing audio-video editing methods. AvED (Lin et al., 2026) exhibits severe visual artifacts and temporal flickering and struggles to synthesize the target audio. CoherentAVEdit (Ishii et al., 2025) and AVI-Edit (Zheng et al., 2026a) rely on explicit masks and either fail to follow the instruction or over-edit the target and surrounding content. InstructAV2AV (Zheng et al., 2026b) performs well on speaker-centric edits but generalizes poorly to open-domain scenes and global editing. MiniMax-H3 (MiniMaxAI, 2026) generally follows the instruction but often alters the source identity and background due to its reference-based generation paradigm. In contrast, OmniInstruct accurately edits the target audio-visual content while preserving non-target context, achieving better instruction adherence and source consistency across diverse editing scenarios. More qualitative comparisons and a user study are provided in Appendix D and Appendix C, respectively.

Quantitative Analysis. In Table 1, OmniInstruct achieves state-of-the-art performance across all instruction-following metrics, demonstrating its effectiveness in executing both visual and audio editing instructions. For source consistency, OmniInstruct achieves the best audio preservation measured by LPAPS, while its visual structural preservation is slightly behind mask-based methods, which benefit from explicit spatial guidance. Regarding editing quality, OmniInstruct achieves the best overall generation quality but underperforms MiniMax-H3 on synchronization-related metrics, likely due to the latter's stronger generative backbone. Overall, OmniInstruct provides a strong balance among instruction following, source consistency, and editing quality. We further evaluate OmniInstruct on the public InsAVE-80K benchmark (Zheng et al., 2026b) without any additional fine-tuning. In Table 4, OmniInstruct demonstrates consistently strong performance across the video, audio, and audio-video evaluation dimensions, further validating its generalization capability.

5.3 Ablation Study

We examine the effects of training data and training strategies. Quantitative results are reported in Tables 3 and 2, with qualitative examples shown in Fig. 5.



Figure 5: Qualitative ablation of Edit-NFT.

Table 2: Ablation study on training strategies.

Model	MoE	V	A	A/V	Rout.	CLIP ↑	CLAP ↑	LPAPS ↓	DeSync ↓
(a)	×	×	×	×	×	25.12	31.40	3.99	53.93
(b)	✓	×	×	×	×	25.43	31.93	4.01	53.27
(c)	✓	✓	×	×	×	<u>25.74</u>	31.56	4.03	53.73
(d)	✓	×	✓	×	×	23.21	<u>32.46</u>	**3.89**	55.83
(e)	✓	×	×	✓	×	24.30	31.74	3.96	<u>52.71</u>
(f)	✓	✓	✓	✓	×	22.32	31.43	4.08	54.68
Ours	✓	✓	✓	✓	✓	**25.80**	**32.56**	<u>3.91</u>	**52.67**

Table 3: Ablation study on training data and data filtering.

Dataset	Scale	CLIP ↑	CLAP ↑	LPAPS ↓	DeSync ↓
InsAVE	80K	23.23	24.94	4.27	59.60
InsAVE	500K	23.11	25.53	4.19	57.33
OmniInstruct	80K	24.49	30.34	4.05	<u>55.62</u>
OmniInstruct*	500K	<u>24.78</u>	<u>30.84</u>	<u>4.04</u>	55.74
OmniInstruct	500K	**25.12**	**31.40**	**3.99**	**53.93**

Table 4: Quantitative comparison on the public InsAVE-80K benchmark.

Method	Video Eval				Audio Eval			A–V Eval			
	FVD ↓	TV-A ↑	TC ↑	SSIM ↑	FAD ↓	TA-A ↑	LPAPS ↓	AV-A ↑	PEAVS ↑	S-C ↑	S-D ↓
AvED	283.3	19.79	95.90	83.30	3.06	29.38	2.31	24.23	<u>2.91</u>	0.90	11.54
CoherentAVEdit	218.3	<u>19.90</u>	96.30	76.68	4.59	<u>31.77</u>	<u>2.03</u>	22.26	2.82	1.34	10.94
AVI-Edit	**183.8**	19.08	<u>96.39</u>	87.95	5.92	30.68	2.64	23.90	2.81	4.00	<u>9.15</u>
InstructAV2AV	194.8	19.46	96.08	**90.69**	<u>2.97</u>	31.58	2.10	<u>24.52</u>	**3.19**	<u>4.25</u>	9.64
OmniInstruct	<u>188.7</u>	**20.18**	**96.68**	<u>89.37</u>	**2.37**	**32.33**	**1.99**	**24.86**	2.87	**4.98**	**9.07**

Training Data. We study the effects of data diversity, scale, and filtering in Table 3, where OmniInstruct* denotes the unfiltered version. At the same 80K scale, OmniInstruct outperforms InsAVE (Zheng et al., 2026b) across all metrics, demonstrating the benefit of higher-quality and more diverse editing pairs. Repeating InsAVE-80K samples to 500K yields little improvement because it increases the training volume without introducing additional diversity. By contrast, scaling OmniInstruct from 80K to 500K with diverse samples improves the overall performance, demonstrating the importance of data diversity rather than repeated exposure alone. Finally, filtering OmniInstruct-500K improves all metrics over its unfiltered counterpart, validating our quality-verification pipeline.

Training Strategy. We analyze the task-aware MoE-LoRA module and the reward design of Edit-NFT in Table 2, where (a) is the SFT baseline with standard LoRA, V, A, and A/V denote the video, audio, and audio-video rewards, and Rout. denotes modality-wise advantage routing. MoE-LoRA improves instruction following and synchronization, with a slight trade-off in source consistency. The individual video, audio, and audio-video rewards improve different aspects of editing but introduce cross-objective trade-offs, while combining them without routing leads to substantial degradation. Modality-wise advantage routing alleviates this interference and provides the best overall balance across the four metrics. The qualitative results in Fig. 5 further demonstrate the benefit of Edit-NFT. Compared with the SFT baseline, Edit-NFT suppresses residual artifacts in subject removal and successfully replaces a rifle with a bow together with the corresponding sound. Notably, the rifle-to-bow edit is absent from the supervised training data, suggesting that reward-based post-training improves generalization beyond the fixed editing pairs observed during SFT.

6 Conclusion

In this work, we address two key bottlenecks in instruction-guided audio-video editing: limited editing diversity in existing datasets and poor open-domain generalization of supervised editing models. We construct OmniInstruct-500K, a large-scale dataset containing 500K audio-video editing triplets spanning nine diverse local, global, and speech editing types, built via an automatic pipeline with multi-stage quality verification. We further present OmniInstruct, an end-to-end editing framework equipped with a task-aware Mixture-of-Experts LoRA module and Edit-NFT, a modality-routed reinforcement learning post-training strategy. Our experiments validate that our dataset and method improve instruction adherence, source preservation, perceptual quality, and cross-modal synchronization, achieving strong generalization across open-domain editing tasks.

Limitations. Despite the promising performance of our method, several limitations remain. (1) The model currently supports editing clips of only about 5 seconds. (2) Our approach is limited to English instructions and speech. We leave long-duration audio-video editing and multilingual editing as future work.

AI USE STATEMENT

In this work, we use generative AI tools for dataset construction. Specifically, during the construction of the OmniInstruct-500K dataset (Section 3): (i) we use an LLM to rewrite dense captions of low-quality source videos and a state-of-the-art audio-visual generation model to regenerate the corresponding videos; (ii) we use Qwen3-Omni to synthesize dense multimodal captions and to verify the logical consistency between editing targets and generated instructions, and use Qwen3-235B to select editing types, formulate audio-visual editing instructions, and generate fine-grained meta-information; (iii) we use expert audio and video generation models to produce the edited content; (iv) we use Qwen3-Omni as a multimodal evaluator to judge instruction following and the consistency of the editing triplets. The annotations of OmniInstruct-Bench are written by humans. We do not use generative AI tools for developing theoretical models or conceptual frameworks, formulating mathematical claims, assisting in proofs, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, translation, reformatting datasets, qualitative or thematic data analysis, or interpreting results, and the remaining required disclosure tasks are not applicable to this work. Additionally, we use generative AI tools for improving paper readability and language fluency. We review all AI-assisted work: all generated captions, instructions, regenerated videos, and edited pairs are passed through the multi-stage quality-filtering pipeline described in Section 3 and are spot-checked by the authors, and all AI-polished text is carefully reviewed, fact-checked, and revised by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

The OmniInstruct-500K dataset is constructed from publicly available sources (SpeakerVid, VGGSound, and AudioSet), and we comply with the terms of use and licenses of these source datasets. The original audio and visual content remain the property of their respective owners. Since these sources contain in-the-wild videos of human speakers, we took care to minimize privacy risks: no personal information beyond what is already public in the source videos is collected, and we do not attempt to identify or profile any individual. The user study (Appendix C) involved 25 voluntary participants who gave informed consent, and no personally identifiable information was collected. We acknowledge that audio-visual editing techniques such as speech synthesis and lip-syncing carry a risk of misuse (e.g., producing deceptive content). To address this, we release the dataset for research purposes only and will include clear usage terms that prohibit deceptive or malicious applications. We take responsibility for the integrity of our research and the responsible release of our data and code.

REPRODUCIBILITY STATEMENT

We have made extensive efforts to ensure the reproducibility of our work. The data construction pipeline of OmniInstruct-500K, including data sources, processing steps, instruction generation, expert editing models, and the multi-stage quality-filtering mechanism, is described in Section 3 and the Appendix. The implementation details of our model, including network architecture, training objective, and hyperparameter settings, are provided in Section 4, Section 5, and Appendix A. We report evaluation protocols and metrics in Section 5 and Appendix A. We will release our source code, training scripts, the OmniInstruct-500K dataset, and model checkpoints.

REFERENCES

Runway AI. Stable Diffusion v1.5 Model Card. https://huggingface.co/runwayml/stablediffusion-v1-5, 2022a.

Stability AI. Stable Diffusion XL Model Card. https://huggingface.co/stabilityai/stablediffusion-xl-base-1.0, 2022b.

Qingyan Bai, Qiuyu Wang, Hao Ouyang, Yue Yu, Hanlin Wang, Wen Wang, Ka Leong Cheng, Shuailei Ma, Yanhong Zeng, Zichen Liu, et al. Scaling instruction-based video editing with a high-quality synthetic dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 37971–37981, 2026.

Roi Benita, Michael Finkelson, Tavi Halperin, Gleb Sterkin, and Yossi Adi. Cafa: a controllable automatic foley artist. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15917–15926. IEEE, 2025.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18392–18402. IEEE, 2023.

ByteDance. Seedance2.0. https://seed.bytedance.com/en/seedance2_0, 2026.

Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset, 2020. URL https://arxiv.org/abs/2004.14368.

Junyi Chen, Tong He, Zhoujie Fu, Pengfei Wan, Kun Gai, and Weicai Ye. Vino: A unified visual generator with interleaved omnimodal context, 2026a. URL https://arxiv.org/abs/2601.02358.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, Xiangzhan Yu, and Furu Wei. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518, October 2022. ISSN 1941-0484. doi: 10.1109/jstsp.2022.3188113. URL http://dx.doi.org/10.1109/JSTSP.2022.3188113.

Yinan Chen, Chuming Lin, Zhennan Chen, Yuxiang Zeng, Junwei Zhu, Yali Bi, Xijie Huang, Chengming Xu, Donghao Luo, Zhucun Xue, et al. Javedit: Joint audio-visual instruction-guided video editing with agentic data curation. *arXiv preprint arXiv:2606.03168*, 2026b.

Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, JianZhao JianZhao, Kai Yu, and Xie Chen. F5-tts: A fairytaler that fakes fluent and faithful speech with flow matching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6255–6271, 2025.

Ho Kei Cheng, Masato Ishii, Akio Hayakawa, Takashi Shibuya, Alexander Schwing, and Yuki Mitsufuji. Mmaudio: Taming multimodal joint training for high-quality video-to-audio synthesis. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28901–28911. IEEE, 2025.

Jiaxin Cheng, Tianjun Xiao, and Tong He. Consistent video-to-video transfer using synthetic dataset. In *International Conference on Learning Representations*, volume 2024, pp. 16867–16879, 2024.

Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *Asian conference on computer vision*, pp. 251–263. Springer, 2016.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation, 2024. URL https://arxiv.org/abs/2306.05284.

Google DeepMind. Veo 3. https://deepmind.google/models/veo, 2025.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

FLUX. FLUX. `https://blackforestlabs.ai/`, 2024.

Xin Gao, Li Hu, Siqi Hu, Mingyang Huang, Chaonan Ji, Dechao Meng, Jinwei Qi, Penchong Qiao, Zhen Shen, Yafei Song, et al. Wan-s2v: Audio-driven cinematic video generation. *arXiv preprint arXiv:2508.18621*, 2025.

Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *ICASSP*, New Orleans, LA, 2017.

Michal Geyer, Omer Bar Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. In *International Conference on Learning Representations*, volume 2024, pp. 1608–1620, 2024.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all, 2023. URL `https://arxiv.org/abs/2305.05665`.

Google. Gemma 3. `https://huggingface.co/google/gemma-3-12b-it-qat-q4_0-unquantized`, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Xu Guo, Fulong Ye, Qichao Sun, Liyang Chen, Bingchuan Li, Pengze Zhang, Jiawei Liu, Songtao Zhao, Qian He, and Xiangwang Hou. Dreamid-omni: Unified framework for controllable human-centric audio-video generation. *arXiv preprint arXiv:2602.12160*, 2026.

Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, et al. Ltx-2: Efficient joint audio-visual foundation model. *arXiv preprint arXiv:2601.03233*, 2026.

Haoyang He, Jie Wang, Jiangning Zhang, Zhucun Xue, Xingyuan Bu, Qiangpeng Yang, Shilei Wen, and Lei Xie. Openve-3m: A large-scale high-quality dataset for instruction-guided video editing. *arXiv preprint arXiv:2512.07826*, 2025.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

Vladimir Iashin and Esa Rahtu. Taming visually guided sound generation. *arXiv preprint arXiv:2110.08791*, 2021.

Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Synchformer: Efficient synchronization from sparse cues. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5325–5329. IEEE, 2024.

Masato Ishii, Akio Hayakawa, Takashi Shibuya, and Yuki Mitsufuji. Coherent audio-visual editing via conditional audio generation following video edits. *arXiv preprint arXiv:2512.07209*, 2025.

Dongzhi Jiang, Ziyu Guo, Renrui Zhang, Zhuofan Zong, Hao Li, Le Zhuo, Shilin Yan, Pheng-Ann Heng, and Hongsheng Li. T2i-r1: Reinforcing image generation with collaborative semantic-level and token-level cot. *Advances in Neural Information Processing Systems*, 38:39856–39890, 2026.

Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17191–17202. IEEE, 2025.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. Fr\'echet audio distance: A metric for evaluating music enhancement algorithms. *arXiv preprint arXiv:1812.08466*, 2018.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

Chunyu Li, Chao Zhang, Weikai Xu, Jingyu Lin, Jinghui Xie, Weiguo Feng, Bingyue Peng, Cunjian Chen, and Weiwei Xing. Latentsync: Taming audio-conditioned latent diffusion models for lip sync with syncnet supervision. *arXiv preprint arXiv:2412.09262*, 2024.

Sen Liang, Cong Wang, Zhentao Yu, Fengbin Guan, Zhengguang Zhou, Teng Hu, Youliang Zhang, Yuan Zhou, Xin Li, Qinglin Lu, et al. Goku: A million-scale universal dataset and benchmark for instruction-based video editing. *arXiv preprint arXiv:2606.30599*, 2026.

Yan-Bo Lin, Kevin Lin, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Chung-Ching Lin, Xiaofei Wang, Gedas Bertasius, and Lijuan Wang. Zero-shot audio-visual editing via cross-modal delta denoising. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 7344–7354. IEEE, 2026.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *ICLR*, 2023.

Jie Liu, Gongye Liu, Jiajun Liang, Ziyang Yuan, Xiaokun Liu, Mingwu Zheng, Xiele Wu, Qiulin Wang, Menghan Xia, Xintao Wang, Xiaohong Liu, Fei Yang, Pengfei Wan, Di Zhang, Kun Gai, Yujiu Yang, and Wanli Ouyang. Improving video generation with human feedback, 2025. URL https://arxiv.org/abs/2501.13918.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *Advances in neural information processing systems*, 38:40783–40818, 2026a.

Kai Liu, Wei Li, Lai Chen, Shengqiong Wu, Yanhao Zheng, et al. Javisdit: Joint audio-video diffusion transformer with hierarchical spatio-temporal prior synchronization. 2026b.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *ICLR*, 2023.

Fuchen Long, Cong Wang, Zitao Gao, Wenhao Zhong, Yu Cheng, Xiaolu Hou, Yan Li, Xiao Cao, Xinlong Sun, Xi Chen, et al. Coinve-200k: A large-scale high-quality dataset for compositional instruction-guided video editing. *arXiv preprint arXiv:2608.17566*, 2026.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Chetwin Low, Weimin Wang, and Calder Katyal. Ovi: Twin backbone cross-modal fusion for audio-video generation. *arXiv preprint arXiv:2510.01284*, 2025.

Simian Luo, Chuanhao Yan, Chenxu Hu, and Hang Zhao. Diff-foley: Synchronized video-to-audio synthesis with latent diffusion models. *Advances in Neural Information Processing Systems*, 36: 48855–48876, 2023.

Yifu Luo, Xinhao Hu, Keyu Fan, Haoyuan Sun, Zeyu Chen, Bo Xia, Tiantian Zhang, Yongzhe Chang, and Xueqian Wang. Reinforcement learning meets masked generative models: Mask-grpo for text-to-image generation. *Advances in Neural Information Processing Systems*, 38:108460–108485, 2026.

MiniMaxAI. MiniMax-H3. https://github.com/MiniMax-AI/MiniMax-H3, 2026.

OpenAI. Sora2. https://openai.com/index/sora-2/, 2025.

OpenAI. ChatGPT Images 2.0. https://deploymentsafety.openai.com/chatgpt-images-2-0/chatgpt-images-2-0.pdf, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, pp. 4195–4205, 2023.

KR Prajwal, Rudrabha Mukhopadhyay, Vinay P Namboodiri, and CV Jawahar. A lip sync expert is all you need for speech to lip generation in the wild. In *Proceedings of the 28th ACM international conference on multimedia*, pp. 484–492, 2020.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pp. 28492–28518. PMLR, 2023.

Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.

Sizhe Shan, Qiulin Li, Yutao Cui, Miles Yang, Yuehai Wang, Qun Yang, Jin Zhou, and Zhao Zhong. Hunyuanvideo-foley: Multimodal diffusion with representation alignment for high-fidelity foley audio generation. *arXiv preprint arXiv:2508.16930*, 2025.

Bowen Shi, Andros Tjandra, John Hoffman, Helin Wang, Yi-Chiao Wu, Luya Gao, Julius Richter, Matt Le, Apoorv Vyas, Sanyuan Chen, et al. Sam audio: Segment anything in audio. *arXiv preprint arXiv:2512.18099*, 2025.

Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matt Le, Nick Zacharov, Carleigh Wood, Ann Lee, and Wei-Ning Hsu. Meta audiobox aesthetics: Unified automatic quality assessment for speech, music, and sound, 2025. URL https://arxiv.org/abs/2502.05139.

Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Qiuheng Wang, Yukai Shi, Jiarong Ou, Rui Chen, Ke Lin, Jiahao Wang, Boyuan Jiang, Haotian Yang, Mingwu Zheng, Xin Tao, Fei Yang, Pengfei Wan, and Di Zhang. Koala-36m: A large-scale video dataset improving consistency between fine-grained conditions and video content, 2024. URL https://arxiv.org/abs/2410.08260.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 13(4):600–612, 2004.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025a.

Yuhui Wu, Liyi Chen, Ruibin Li, Shihao Wang, Chenxi Xie, and Lei Zhang. Insvie-1m: Effective instruction-based video editing with elaborate dataset construction. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16692–16701. IEEE, 2025b.

Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP*, pp. 1–5. IEEE, 2023.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, Yuanjun Lv, Yongqi Wang, Dake Guo, He Wang, Linhan Ma, Pei Zhang, Xinyu Zhang, Hongkun Hao, Zishan Guo, Baosong Yang, Bin Zhang, Ziyang Ma, Xipin Wei, Shuai Bai, Keqin Chen, Xuejing Liu, Peng Wang, Mingkun Yang, Dayiheng Liu, Xingzhang Ren, Bo Zheng, Rui Men, Fan Zhou, Bowen Yu, Jianxin Yang, Le Yu, Jingren Zhou, and Junyang Lin. Qwen3-omni technical report, 2025. URL https://arxiv.org/abs/2509.17765.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, et al. Dancegrpo: Unleashing grpo on visual generation. *arXiv preprint arXiv:2505.07818*, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025a. URL https://arxiv.org/abs/2505.09388.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *ICLR*, 2025b.

Guohui Zhang, XiaoXiao Ma, Jie Huang, Hang Xu, Hu Yu, Siming Fu, Yuming Li, Zeyue Xue, Lin Song, Haoyang Huang, et al. Omninft: Modality-wise omni diffusion reinforcement for joint audio-video generation. *arXiv preprint arXiv:2605.12480*, 2026a.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in neural information processing systems*, 36:31428–31449, 2023a.

Shiwei Zhang, Jiayu Wang, Yingya Zhang, Kang Zhao, Hangjie Yuan, Zhiwu Qin, Xiang Wang, Deli Zhao, and Jingren Zhou. I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models. *arXiv preprint arXiv:2311.04145*, 2023b.

Yiming Zhang, Yicheng Gu, Yanhong Zeng, Zhening Xing, Yuancheng Wang, Zhizheng Wu, Bin Liu, and Kai Chen. Foleycrafter: Bring silent videos to life with lifelike and synchronized sounds. *International Journal of Computer Vision*, 134(1):46, 2026b.

Youliang Zhang, Zhaoyang Li, Duomin Wang, Jiahe Zhang, Deyu Zhou, Zixin Yin, Xili Dai, Gang Yu, and Xiu Li. Speakervid-5m: A large-scale high-quality dataset for audio-visual dyadic interactive human generation. *arXiv preprint arXiv:2507.09862*, 2025.

Haojie Zheng, Shuchen Weng, Jingqi Liu, Siqi Yang, Boxin Shi, and Xinlong Wang. Audio-sync video instance editing with granularity-aware mask refiner. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23150–23160, 2026a.

Haojie Zheng, Yixin Yang, Siqi Yang, Shuchen Weng, and Boxin Shi. Instructav2av: Instruction-guided audio-video joint editing. *arXiv preprint arXiv:2605.18467*, 2026b.

Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. Diffusionnft: Online diffusion reinforcement with forward process. In *International Conference on Learning Representations*, volume 2026, pp. 134129–134150, 2026c.

Bojia Zi, Weixuan Peng, Xianbiao Qi, Jianan Wang, Shihao Zhao, Rong Xiao, and Kam-Fai Wong. Minimax-remover: Taming bad noise helps video object removal, 2025. URL https://arxiv.org/abs/2505.24873.

Bojia Zi, Penghui Ruan, Marco Chen, Xianbiao Qi, Shaozhe Hao, Shihao Zhao, Youze Huang, Bin Liang, Rong Xiao, and Kam-Fai Wong. Señorita-2m: A high-quality instruction-based dataset for general video editing by video specialists. *Advances in Neural Information Processing Systems*, 38, 2026.

APPENDIX

This Appendix is organized as follows. Appendix A elaborates on more implementation details. Appendix B provides more details of our collected dataset OmniInstruct-500K. Appendix C gives a user study as a complement to the quantitative comparison. Appendix D presents additional qualitative visualization results.

A More Implementation Details

A.1 Training Details

We adopt a two-stage training strategy to develop OmniInstruct on top of LTX-2.3 (HaCohen et al., 2026). We first establish instruction-guided editing capabilities through supervised fine-tuning (SFT), and then improve generalization through Edit-NFT, our reward-based reinforcement learning post-training method.

Stage 1: Supervised Fine-Tuning. We fine-tune the pretrained LTX-2.3 generation backbone into an audio-video editing model on our OmniInstruct-500K dataset. To retain the generative prior while adapting to the editing task, we freeze the base diffusion transformer and inject trainable low-rank adapters, mixing two adapter types across its 48 dual-stream (video/audio) blocks. Standard LoRA of rank 128 (scaling factor $\alpha = 128$) is attached to the query, key, value projections and the feed-forward layers of both the video and audio streams. The output projection of every attention block is instead replaced by a mixture-of-experts LoRA (MoE-LoRA): 4 experts of rank 32 (so that the aggregate rank $4 \times 32 = 128$ matches the standard branch) combined by a token-wise softmax router conditioned on both the latent token and the pooled instruction feature, i.e. $\mathrm{out}(x) = W_o x + \frac{\alpha}{r_e} \sum_{i=1}^{4} g_i\big(x, \bar{C}\big)\, B_i A_i x$, where $g(\cdot)$ is the learned gate, $\bar{C}$ is the pooled instruction feature, and $r_e = \alpha = 32$, giving the same unit scaling as the standard branch. The base weights remain frozen throughout. The model is optimized with AdamW (learning rate 1×10^{-4}, weight decay 1×10^{-2}) for one epoch on 64 H20 GPUs. This stage converts the T2AV generator into an instruction-guided editor.

Stage 2: Reinforcement Learning Post-Training. Starting from the SFT checkpoint, we further improve editing generalization using Edit-NFT, our reinforcement learning post-training method based on DiffusionNFT (Zheng et al., 2026c). We freeze the backbone and SFT adapters and optimize a separate LoRA adapter of rank 64 with a scaling factor of $\alpha = 128$, using AdamW with a learning rate of 3×10^{-5} and weight decay of 1×10^{-4}. For each source audio-video pair and editing instruction, we generate a group of eight candidates using different random seeds, each sampled with 20 denoising steps. We independently compute group-relative advantages for three video rewards, three audio rewards, and one shared audio-video synchronization reward, and route them to the corresponding modality branches. We set the NFT guidance strength to $\beta = 1.0$ in Eq. 4. In addition to the NFT objective in Eq. 5, we regularize the policy toward the frozen SFT model to prevent it from drifting away from the editing capability acquired during SFT, so that the overall objective is

$$\mathcal{L}_{\text{Edit-NFT}} = \sum_{m \in \{v,a\}} \mathcal{L}^{m}_{\text{NFT}} + \lambda_{\text{KL}}\, \mathcal{L}_{\text{KL}}, \tag{14}$$

where $\mathcal{L}^{m}_{\text{NFT}}$ is the NFT loss of branch m computed with its routed advantage $\tilde{A}_m$, and $\lambda_{\text{KL}} = 1 \times 10^{-4}$. Training runs for 400 iterations. This stage directly optimizes model-generated edits beyond the fixed target pairs used during SFT.

A.2 Baseline

We compare OmniInstruct with the following representative methods for audio-video editing. We also considered JAVEdit (Chen et al., 2026b), a recent instruction-guided joint audio-video editing method. However, to the best of our knowledge, its dataset and model checkpoints were not publicly available at the time of submission, preventing a reproducible comparison.

AvED (Lin et al., 2026). AvED is a training-free framework for zero-shot audio-video editing. It introduces cross-modal delta denoising, which identifies prompt-relevant audio and visual patches and applies a contrastive objective to improve cross-modal coherence and synchronization during the denoising process.

CoherentAVEdit (Ishii et al., 2025). CoherentAVEdit adopts a sequential audio-visual editing pipeline: it first applies an existing video editor and then generates an audio track conditioned on the source audio, edited video, and text prompt. It dynamically adjusts the contribution of the source audio according to the editing complexity, preserving its original structure whenever possible.

AVI-Edit (Zheng et al., 2026a). AVI-Edit is an end-to-end framework for audio-synchronized, instance-level video editing, taking a coarse instance mask and a text description as input. It introduces a granularity-aware mask refiner for precise spatial control and a self-feedback audio agent that produces curated audio guidance for fine-grained temporal control.

InstructAV2AV (Zheng et al., 2026b). InstructAV2AV is an end-to-end framework that jointly edits audio and video using only a source audio-video sample and a natural-language instruction. It is trained on the synthesized InsAVE-80K dataset and introduces Source-Instruction Gated Attention, together with a two-stage training strategy, to balance instruction following, source-content preservation, and audio-video synchronization.

MiniMax-H3 (MiniMaxAI, 2026). MiniMax-H3 is a joint audio-video generation model that supports reference-conditioned generation. As it does not natively support instruction-guided editing, we use it in a reference-based generation manner, conditioning on the source audio-video clip and the editing instruction to synthesize the edited result.

A.3 Evaluation

We assess edited outputs along three complementary dimensions— **Instruction Following**, **Source Consistency**, and **Editing Quality**—using ten metrics.

Instruction Following. This group measures whether the edit realizes what the instruction asks, across the visual, audio, and speech modalities. *CLIP* embeds every output frame and the target visual description with CLIP-B (Radford et al., 2021) and reports their mean cosine similarity, so a higher value means the visual content better matches the requested edit. *CLAP* is its audio counterpart: it embeds the edited audio track and the target audio description with CLAP and reports their similarity, measuring whether the produced sound matches the instruction. *WER* (Word Error Rate) targets speech: we transcribe the edited audio with Whisper (Radford et al., 2023) and compare it to the reference transcript, with a lower error indicating the intended spoken content is faithfully rendered.

Source Consistency. This group evaluates whether content not targeted by the instruction remains consistent with the source. *SSIM* measures structural similarity between the source and edited video frames, with a higher score indicating stronger visual preservation. For audio, following InstructAV2AV (Zheng et al., 2026b), we adopt Learned Perceptual Audio Patch Similarity (*LPAPS*) (Iashin & Rahtu, 2021), which extracts hidden states from all layers of the CLAP audio encoder for the source and edited audio, ℓ_2-normalizes them, and sums the spatially averaged distances across layers. A lower LPAPS indicates greater perceptual audio similarity.

Editing Quality. This group evaluates the perceptual quality and realism of the produced audio–visual result. *FVD* (Fréchet Video Distance) compares the distributions of edited and reference video features to assess visual quality and temporal realism, while *FAD* (Fréchet Audio Distance) does the same for audio naturalness; both are lower-is-better. *LSE-C* and *LSE-D* (Lip-Sync Error, confidence and distance) use a SyncNet-style model over detected face tracks to measure lip–speech alignment, with higher confidence and lower distance denoting better-synchronized talking-face content. Finally, *DeSync* generalizes synchronization beyond lips: a Synchformer model estimates the audio–visual temporal offset for general sound–motion correspondence, where a smaller value denotes tighter alignment.

Evaluation on InsAVE-80K. For the comparison on the public InsAVE-80K benchmark, we follow the evaluation protocol of InstructAV2AV (Zheng et al., 2026b). *TV-A* measures text-video alignment with VideoCLIP-XL, *TC* measures temporal consistency via consecutive-frame CLIP similarity, *TA-A* measures text-audio alignment with CLAP, *AV-A* measures audio-visual alignment with ImageBind, *PEAVS* assesses overall audio-visual synchronization quality, and *S-C*/*S-D* denote the Sync-C/Sync-D lip-synchronization scores. FVD, SSIM, FAD, and LPAPS are computed as described above.

B Dataset

B.1 Editing Types Definition

Audio-Visual Editing Taxonomy. We organize audio-visual editing instructions into three major categories: *Local Edit*, *Global Edit*, and *Speech Edit*. Each category contains three fine-grained editing types:

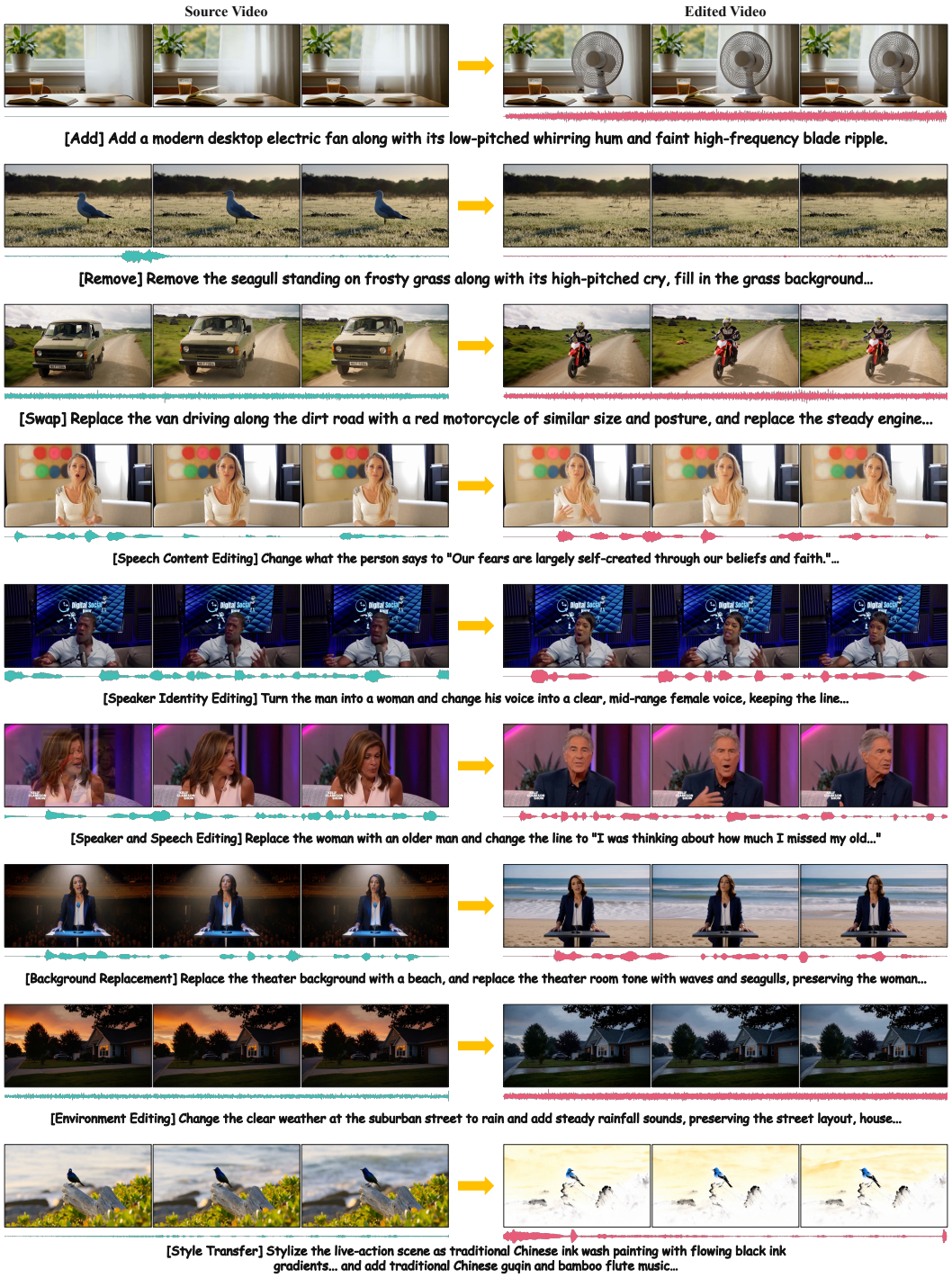
- **Local Edit.** This category targets individual entities and their associated sounds while leaving the remainder of the audio-visual scene unchanged.
  - **Add:** Introduces a new visible entity and adds the sound naturally associated with it. For example, "Add a fire truck to the scene and include a synchronized siren sound."
  - **Remove:** Eliminates a visible entity and its corresponding sound, while reconstructing the occluded visual background and preserving unrelated audio events. For example, "Remove the guitarist from the scene together with the guitar sound."
  - **Swap:** Replaces an existing entity with another entity and correspondingly substitutes its acoustic characteristics. For example, "Swap the dog for a cat and replace the barking with meowing."
- **Global Edit.** This category modifies scene-level visual and acoustic properties while preserving the principal foreground content.
  - **Background:** Replaces the overall setting and updates the associated ambient sounds to match the new location. For example, "Replace the office background with a beach and change the indoor ambience to ocean waves and sea breeze."
  - **Environment:** Alters environmental conditions, such as the time of day, weather, or season, and correspondingly adapts the soundscape. For example, "Transform the daytime forest into a nighttime scene and replace the birdsong with insect and frog sounds."
  - **Style:** Transforms the overall visual appearance and acoustic presentation according to a specified artistic or media style. For example, "Convert the realistic video into a Japanese anime style and adapt the Foley effects and background music accordingly."
- **Speech Edit.** This category modifies spoken content, speaker identity, or both, while maintaining consistency among speech, lip movements, facial appearance, and body motion.
  - **Speech Content:** Revises an utterance through replacement, insertion, or deletion, with the generated speech and lip movements synchronized to the edited content while the speaker's identity and voice timbre are preserved. For example, "Change the spoken phrase 'See you tomorrow' to 'See you tonight' and update the voice and lip movements accordingly."
  - **Speaker Identity:** Changes identity-related attributes of a speaker, such as personal identity, age, or gender, together with the corresponding vocal characteristics, while retaining the original speech content and actions. For example, "Replace host A with host B while preserving the original dialogue and body movements."
  - **Speech & Identity:** Jointly changes the speaker's identity-related attributes and the spoken content, with the new appearance, voice, and lip movements kept consistent with the edited line. For example, "Replace the man with a young boy and change the line to 'I don't wanna wait, I wanna go now!' "



A-Fig. 1: More high-quality examples from OmniInstruct-500K Dataset.

B.2 Dataset Statistics

OmniInstruct-500K contains 500K audio-video editing triplets across nine fine-grained editing types. As summarized in A-Tab. 2, it provides substantially greater scale and editing diversity than existing audio-video editing datasets while covering diverse real-world source domains. The nine



A-Fig. 2: More high-quality examples from OmniInstruct-500K Dataset.

editing types are grouped into three categories: speech editing accounts for 41.6% of the dataset, local editing for 41.1%, and global editing for 17.2%, as illustrated in Fig. 1.

A-Tab. 1 further evaluates the quality of the constructed data. Our quality-verification process consistently improves all automatic and MLLM-based metrics over the unfiltered version. Compared with InsAVE-80K, OmniInstruct-500K achieves better audio quality, audio-visual synchro-

A-Tab. 1: Quantitative comparison of dataset quality. IF, SP, and EQ denote instruction following, source preservation, and editing quality, respectively, as rated by Qwen3-Omni on a five-point scale. The best and second-best results are highlighted in **bold** and underlined, respectively.

Dataset	ABox-PQ ↑	LSE-C ↑	LSE-D ↓	DeSync ↓	IF ↑	SP ↑	EQ ↑
InsAVE-80K	6.67	3.03	9.87	0.55	3.93	**4.27**	3.36
OmniInstruct-500K (*unfiltered*)	6.57	4.25	9.27	0.47	4.05	4.18	3.32
OmniInstruct-500K	**6.78**	**6.31**	**7.93**	**0.37**	**4.22**	4.26	**3.67**

nization, instruction following, and editing quality. InsAVE-80K obtains a marginally higher source-preservation score, as its mask-guided construction primarily focuses on localized editing and explicitly constrains non-target regions. In contrast, OmniInstruct-500K maintains comparable source preservation while covering a broader range of local, global, and speech editing tasks.

A-Tab. 2: Comparison with existing video editing datasets. Gray rows: visual-only datasets, for scale reference only. Within each block, datasets are ordered by first public release. ✓/×/"–": provided / not provided / not reported.

Dataset	**Scale**	**Audio**	**Edit Types**	**Domain**	**Resolution**	**Frame Count**
Señorita-2M (Zi et al., 2026)	∼2M	×	18	General	728×1288	33–64
InsViE-1M (Wu et al., 2025b)	∼1M	×	–	General	1024×576	25
Ditto-1M (Bai et al., 2026)	∼1M	×	3	General	1280×720	101
OpenVE-3M (He et al., 2025)	∼3M	×	8	General	1280×720	65–129
Goku-2M (Liang et al., 2026)	∼2M	×	10	General	720p	65–129
AVISet (Zheng et al., 2026a)	73k	✓	4	Local	720p	∼240
InsAVE-80K (Zheng et al., 2026b)	88k	✓	5	Local	720p	∼121
JAVEdit-100k (Chen et al., 2026b)	∼103k	✓	5	Human	1280×720	121
Ours	**∼500k**	✓	**9**	**General**	**1280×720**	**∼121**



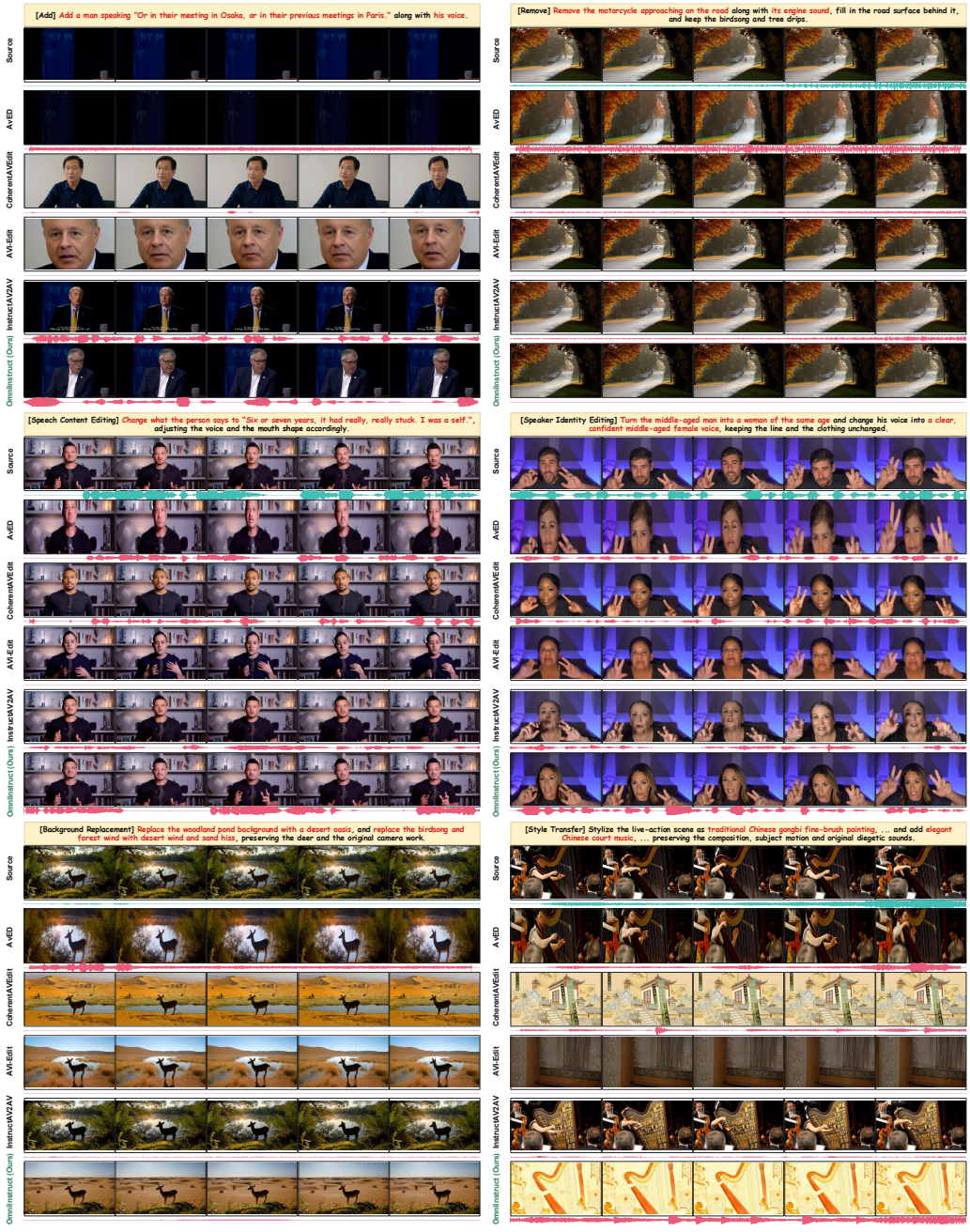
A-Fig. 3: User preference across four dimensions: instruction adherence, editing consistency, editing quality, and overall quality.

C User Study

We conducted a user study with 25 participants on 35 editing examples randomly sampled from OmniInstruct-Bench. For each example, participants were presented with the source video, the editing instruction, and six anonymized outputs generated by OmniInstruct and five competing methods. The display order of the methods was randomized to mitigate positional bias. For each of four perceptual dimensions, *instruction adherence*, *editing consistency*, *editing quality*, and *overall quality*, participants independently selected the single best output. As summarized in A-Fig. 3, OmniInstruct achieves the highest best-choice rate for instruction adherence (39.7%), editing quality (36.1%), and overall quality (43.8%). It also remains competitive in editing consistency (27.2%), ranking second only to AVI-Edit (29.4%). These results demonstrate a clear overall user preference for OmniInstruct, particularly in faithfully executing editing instructions and producing high-quality audio-visual edits.

D More Qualitative Comparison

In A-Fig. 4, we present additional qualitative comparisons with existing state-of-the-art methods: AvED (Lin et al., 2026), CoherentAVEdit (Ishii et al., 2025), AVI-Edit (Zheng et al., 2026a), and InstructAV2AV (Zheng et al., 2026b).



A-Fig. 4: More qualitative comparison results with state-of-the-art methods for instruction-guided audio-video editing.