FLUID-OMNI: PARALLELIZING STREAMING OMNI GENERATION WITH COMMIT-REPAIR-WRITE DIFFUSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Autoregressive omni-modal models naturally support streaming generation because each newly produced token extends a stable causal prefix that can be immediately consumed. However, their sequential decoding process limits generation efficiency. Discrete diffusion decoding provides a promising alternative through parallel generation, yet directly applying it to omni-modal generation introduces a fundamental challenge: future predictions may become available before they are reliable enough for irreversible commitment. We identify this prediction–commitment mismatch as a key barrier to efficient parallel streaming generation. We introduce FLUID-Omni, a causal diffusion framework that bridges parallel generation and streaming reliability through a Commit–Repair–Write paradigm. Instead of treating all predicted future tokens as equally reliable, FLUID-Omni selectively commits stable predictions, locally repairs inconsistent regions while preserving reusable context, and continuously writes new predictions in parallel. To align training with streaming inference, we develop a multi-stage adaptation strategy that equips the model with causal parallel prediction and pipeline-consistent refinement capabilities. Furthermore, we introduce prior-calibrated codec decoding to mitigate biased confidence estimation in audio generation. Experiments across semantic, acoustic, and streaming evaluations demonstrate that FLUID-Omni substantially improves generation efficiency while preserving omni-modal generation quality. These results show that parallel diffusion can achieve reliable streaming generation when prediction and commitment are explicitly decoupled.

1 INTRODUCTION

Recent advances in large omni-modal models have enabled unified perception and generation across diverse modalities, including text, vision, and speech (Xu et al., 2025). Beyond offline generation quality, real-world interactive applications increasingly demand models that can produce responses with low latency and continuously deliver outputs during generation (Xu et al., 2025; Défossez et al., 2024). Autoregressive (AR) architectures naturally support such streaming scenarios because each newly generated token extends a stable causal prefix that can be immediately consumed by downstream modules. However, their token-by-token dependency imposes an inherently sequential decoding path, limiting generation efficiency, especially for long-form responses with dense audio-token streams (Arriola et al., 2025; Ruan et al., 2026).

Discrete diffusion provides a promising alternative by replacing sequential next-token prediction with iterative denoising over discrete tokens (Austin et al., 2021a; Sahoo et al., 2024a). By predicting and refining multiple positions simultaneously, diffusion decoding can substantially reduce sequential generation steps (Chang et al., 2022; Ruan et al., 2026). Recent works have further explored causal, block-wise, and concurrent diffusion decoding (Chen et al., 2024a; Jin et al., 2026), while diffusion-based multimodal models have demonstrated the feasibility of extending this paradigm to vision-language understanding (Yu et al., 2025; You et al., 2026; Cheng et al., 2026). These advances motivate discrete diffusion as a new generation paradigm for omni-modal models. However, streaming omni-modal interaction imposes requirements beyond parallel token prediction.

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.

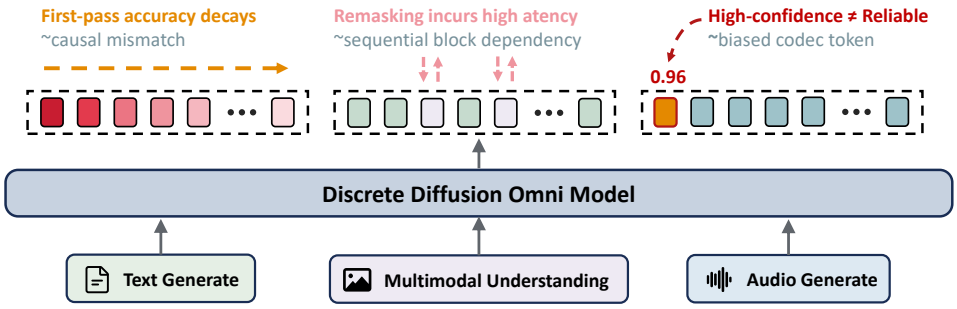

Figure 1: **Challenges of parallel diffusion for streaming omni-modal generation.** Direct parallel prediction introduces causal degradation over future positions, global remasking creates sequential refinement overhead, and raw codec confidence can be biased for audio commitment.

Unlike conventional diffusion generation, where predictions can undergo iterative refinement before the final output is exposed (Chang et al., 2022; Sahoo et al., 2024a), streaming generation must continuously decide which predictions are sufficiently reliable for irreversible commitment. As illustrated in Figure 1, we identify three challenges arising from this discrepancy. First, predictions farther from the current causal frontier become increasingly difficult to resolve in a single parallel pass. Second, uncertainty is often localized, whereas aggressive remasking may discard reusable future predictions and introduce unnecessary decoding steps. Third, audio generation introduces an additional reliability issue. Confidence-guided parallel decoding has been successfully applied to neural codec tokens (Borsos et al., 2023b), yet we observe that codec-token priors can systematically bias model confidence, such that high confidence does not necessarily imply reliable commitment.

These observations reveal a central challenge for parallel omni-modal diffusion generation: how can a model exploit the efficiency of parallel prediction while preserving the causal commitment behavior required for streaming interaction? Addressing this challenge requires moving beyond conventional diffusion decoding strategies that couple prediction with commitment.

In this work, we introduce FLUID-Omni, a causal parallel diffusion framework for streaming omni-modal generation. FLUID-Omni bridges parallel generation and reliable streaming through a unified Commit–Repair–Write paradigm. Instead of directly committing all predicted future tokens, FLUID-Omni selectively commits stable predictions, locally repairs uncertain regions while preserving reusable future context, and continuously writes new predictions in parallel. This design enables the model to maintain streaming progress while avoiding unnecessary recomputation caused by global remasking.

To support this inference paradigm, we further develop a training framework that aligns optimization with the streaming generation process. Our training procedure first equips the model with causal parallel prediction ability and subsequently exposes future writing to inference-induced intermediate states through pipeline-aligned training. In addition, we address the modality-specific reliability of audio codec generation through prior-calibrated commitment, which corrects biased codec confidence before predictions are committed.

Extensive experiments demonstrate that FLUID-Omni preserves strong multimodal understanding after diffusion adaptation and achieves competitive end-to-end speech interaction quality while substantially improving streaming efficiency. These results establish FLUID-Omni as an effective approach for building efficient and reliable omni-modal generators based on parallel diffusion.

2 Related Work

2.1 Discrete Diffusion for Efficient Sequence Generation

Autoregressive (AR) models achieve strong sequence generation quality through causal factorization, but suffer from inherently sequential decoding. Discrete diffusion provides an alternative by enabling iterative parallel refinement. D3PM (Austin et al., 2021b) formulates diffusion over discrete state spaces, while MaskGIT (Chang et al., 2022) demonstrates confidence-guided parallel

prediction with iterative masking and refinement. Diffusion-LM (Li et al., 2022) and MDLM (Sahoo et al., 2024b) further explore masked discrete diffusion for text generation. Similar principles have been extended to audio generation, where SoundStorm (Borsos et al., 2023b) achieves efficient codec-token generation through non-autoregressive confidence-based decoding.

Recent studies combine diffusion parallelism with causal structures to improve sampling efficiency while preserving sequential dependencies. Diffusion Forcing (Chen et al., 2024a) introduces heterogeneous noise levels within causal sequences, Block Diffusion (Arriola et al., 2025) performs diffusion inside blocks with autoregressive dependencies across blocks, and CARD (Ruan et al., 2026) explores causal attention with adaptive parallel decoding. Multi-Block Diffusion (Jin et al., 2026) further reveals the mismatch between training and inference states in block-wise generation. However, existing approaches mainly focus on improving parallel prediction or refinement efficiency, while streaming generation requires an additional mechanism to determine when predictions can be safely committed. FLUID-Omni addresses this challenge by explicitly separating prediction, repair, and commitment during streaming diffusion decoding.

2.2 Streaming and Incremental Decoding in Omni-Modal Systems

Real-time omni-modal systems commonly rely on causal autoregressive generation to support incremental output. Qwen2.5-Omni (Xu et al., 2025) enables streaming text–speech interaction through autoregressive audio token generation, while Moshi (Défossez et al., 2024) models real-time dialogue with codec-based speech streams. PSLM (Mitsui et al., 2024) reduces response latency by jointly generating spoken and textual responses, and GPT-4o (OpenAI, 2024) demonstrates end-to-end real-time multimodal interaction.

Neural audio codecs, including SoundStream (Zeghidour et al., 2021) and EnCodec (Défossez et al., 2022), represent speech as compact discrete tokens and enable language-model-based audio generation such as AudioLM (Borsos et al., 2023a). SoundStorm (Borsos et al., 2023b) further shows that codec tokens can be generated through parallel refinement. However, existing streaming omni-modal systems primarily rely on autoregressive decoding, while diffusion-based approaches lack an explicit mechanism for maintaining a reliable streaming frontier. FLUID-Omni bridges these directions by introducing causal parallel diffusion with Commit–Repair–Write decoding and confidence calibration for reliable audio commitment.

3 Preliminaries

3.1 Discrete Diffusion Omni-Modal Modeling

Consider an omni-modal generation dataset $\mathcal{D}$ of triples (m, c, y), where m denotes multimodal inputs, c represents the encoded multimodal context, and $y = (y_1, \ldots, y_L) \in \mathcal{V}^L$ is the output token sequence, including text tokens and audio codec tokens. The goal is to learn $p_\theta(y|c)$, where the conditioning context c remains clean and only response-side tokens are generated or corrupted.

Discrete diffusion models replace autoregressive factorization with a masked corruption-denoising process. Let z_s denote the corrupted response at noise level s, and $\mu = [\mathrm{MASK}]$. The absorbing corruption process is formulated as:

$$q(z_{s,i} = a|y_i) = \alpha_s \mathbf{1}_{a=y_i} + (1 - \alpha_s)\mathbf{1}_{a=\mu}, \tag{1}$$

where α_s controls the corruption schedule. The denoising model predicts the original tokens from the corrupted sequence by minimizing:

$$\mathcal{L}_{\text{diff}} = -\mathbb{E}\left[\lambda(s) \sum_{i \in \Omega_s} \log p_\theta(y_i|c, z_s)\right], \tag{2}$$

where $\Omega_s = \{i : z_{s,i} = \mu\}$ denotes the masked positions. During inference, diffusion models iteratively recover multiple masked positions in parallel.

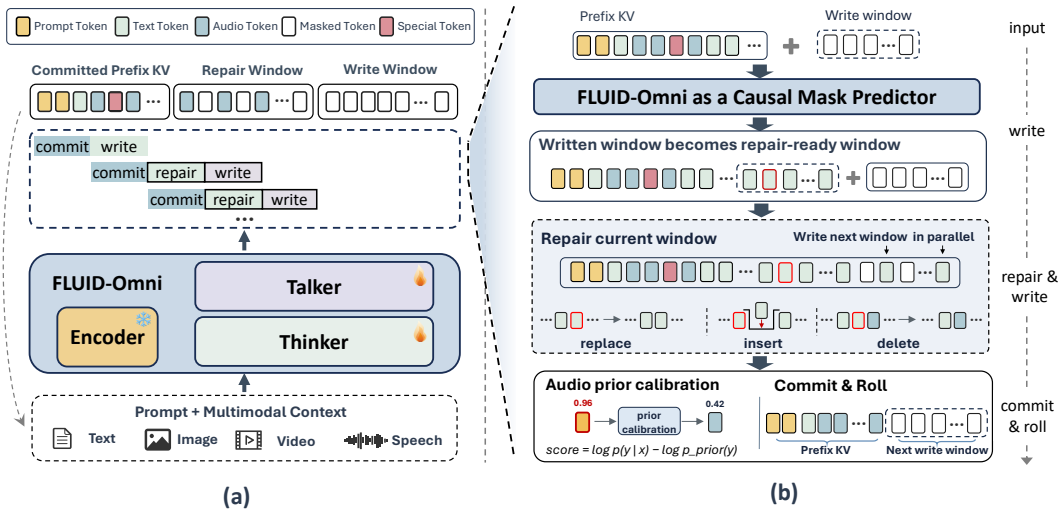

Figure 2: **Overview of FLUID-Omni and its streaming Commit–Repair–Write decoding. (a)** FLUID-Omni performs causal parallel generation over text and audio while maintaining a committed prefix, a repair-ready window, and a write window. **(b)** Each cycle overlaps local edit-based repair of the current window with parallel writing of the next, followed by prior-calibrated audio commitment and window rolling.

3.2 Block-Wise Parallel Omni Generation

To improve generation efficiency, recent diffusion-based methods perform block-wise parallel generation, where future tokens are predicted within fixed spans. Let Y_b denote the b-th generated block and $Y_{<b}$ the previous committed tokens. The generation process can be written as:

$$p(y|c) = \prod_{b=1}^{B} p_\theta(Y_b|c, Y_{<b}). \tag{3}$$

Within each block, diffusion models jointly predict multiple future tokens using bidirectional interactions. Although block-wise generation reduces sequential decoding steps, all tokens inside a predicted block are not necessarily equally reliable. In particular, token availability after parallel prediction does not imply that the corresponding positions are ready for irreversible commitment, which creates a fundamental challenge for streaming omni-modal generation.

4 FLUID-Omni

Given a multimodal context c, FLUID-Omni adapts the generation pathways of a pretrained omni-modal model into causal parallel diffusion predictors while preserving its multimodal understanding capability. As illustrated in Figure 2, generation maintains three logical regions: a *committed prefix*, a *repair-ready window* that remains refinable, and a *write window* for parallel future prediction. FLUID-Omni decouples prediction from commitment and continuously refines the current window while writing the subsequent one, forming a streaming Commit–Repair–Write process. The same generation principle is applied to both the Thinker and Talker, with the Talker additionally conditioned on aligned Thinker representations.

4.1 Causal Parallel Omni Generation

Let $C_t = (y_1, \ldots, y_t)$ denote the currently committed output prefix. FLUID-Omni appends a write window of K masked future positions,

$$W_t^{(0)} = (\mu_1, \ldots, \mu_K), \qquad \mu = [\text{MASK}], \tag{4}$$

and jointly predicts their clean tokens:

$$P_{t,j} = p_\theta \left(y_{t+j} \mid c, C_t, W_t^{(0)} \right), \qquad j = 1, \ldots, K. \tag{5}$$

To preserve the causal structure of the pretrained omni model, FLUID-Omni uses a strict causal block mask. Each write position attends to the committed prefix and only preceding positions within the same local block:

$$A_{ij} = \mathbf{1}[j \in C_t] \ \vee \ \mathbf{1}\left[i, j \in W_t^{(0)} \wedge r_j \leq r_i \right], \tag{6}$$

where r_i denotes the relative position inside the write window. This allows all K future positions to be denoised in parallel while preserving unidirectional dependencies within the block.

The resulting block $\hat{W}_t$ is not immediately appended to the committed prefix. Instead, it becomes a *repair-ready window* that remains subject to local refinement, separating parallel prediction from irreversible commitment.

4.2 Streaming Commit–Repair–Write Decoding

A completed parallel block is not necessarily ready for streaming commitment. FLUID-Omni therefore maintains three decoding states: the committed prefix C_t, a repair-ready window R_t, and the subsequent write window W_{t+1}. Once W_t is filled, it becomes R_t, while a new masked window W_{t+1} is immediately opened.

During steady-state decoding, FLUID-Omni jointly advances local refinement of R_t and parallel denoising of W_{t+1}:

$$R_t \ \parallel \ W_{t+1}. \tag{7}$$

Rather than remasking the entire block, refinement is performed through local discrete edits,

$$\mathcal{E} \in \{\textsc{Replace}, \textsc{Insert}, \textsc{Delete}\}, \qquad R_t^{(r+1)} = \mathcal{E}\left(R_t^{(r)} \right), \tag{8}$$

allowing erroneous, missing, or redundant tokens to be corrected while preserving already stable predictions.

When a prefix of the refined window becomes sufficiently stable, it is moved into the committed context. The concurrently generated W_{t+1} then becomes the next repair-ready window, and a new masked window is appended:

$$(C_t, R_t, W_{t+1}) \longrightarrow (C_{t+1}, R_{t+1}, W_{t+2}). \tag{9}$$

This rolling state transition continuously advances the commitment frontier without blocking future parallel generation.

4.3 Prior-Calibrated Commitment

Commitment requires distinguishing genuinely stable predictions from tokens that only exhibit high instantaneous confidence. FLUID-Omni therefore measures the stability of each position across local refinement. Let $\hat{y}_{t,j}^{(r)}$ and $\hat{y}_{t,j}^{(r+1)}$ denote the predictions for position j before and after a refinement update. A position is considered stable only when its prediction remains unchanged.

For audio codec tokens, raw model confidence is additionally biased by the non-uniform codec-token prior. We calibrate the refinement score as

$$s_{t,j} = \log p_\theta \left(\hat{y}_{t,j}^{(r)} \mid c, C_t, R_t^{(r)} \right) - \log \pi \left(\hat{y}_{t,j}^{(r)} \right), \tag{10}$$

where $\pi(\cdot)$ denotes the codec-token prior. A position becomes committable when its prediction remains stable under refinement and its calibrated score exceeds a threshold.

To preserve causal consistency, FLUID-Omni advances the commitment frontier only through the longest consecutive stable prefix:

$$a_t = \max \{ a : g_{t,1} = \cdots = g_{t,a} = 1 \}, \tag{11}$$

where $g_{t,j}$ indicates whether position j satisfies the commitment criterion. Only $R_{t,1:a_t}$ is transferred into C_{t+1}, while the remaining positions continue to participate in subsequent refinement.

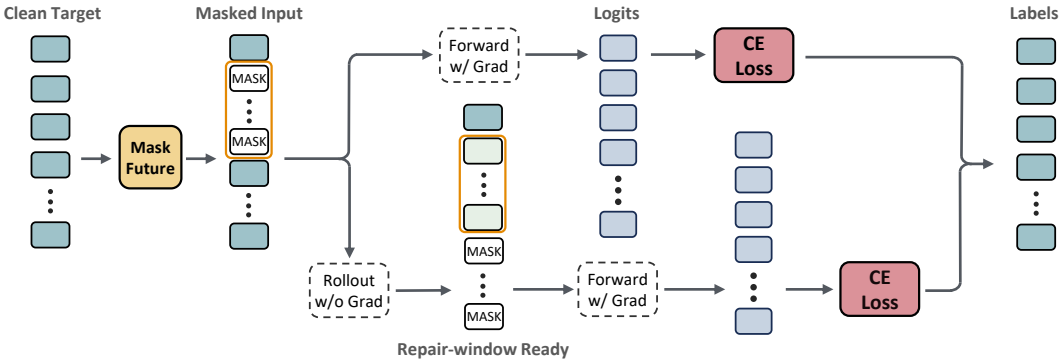

Figure 3: **Pipeline-aligned training of FLUID-Omni.** The upper branch performs direct causal parallel denoising from masked future tokens. In the lower branch, a no-gradient inference rollout first constructs a self-generated repair-ready state, after which a new masked write window is appended and optimized with gradient supervision. This exposes future writing to the imperfect intermediate context encountered during streaming inference.

4.4 Pipeline-Aligned Training

FLUID-Omni follows a two-stage training scheme: Stage 1 learns causal parallel writing, while Stage 2 aligns future writing with the intermediate states encountered during streaming inference.

Causal Parallel Adaptation. Given a clean target sequence, we mask a future span of K tokens and train the model to recover them under the causal block attention defined in Section 4.1:

$$\mathcal{L}_{\text{dir}} = -\sum_{j=1}^{K} \log p_\theta \left(y_{t+j} \mid c, C_t, W_t^{(0)} \right). \tag{12}$$

This adapts the pretrained generator to parallel future prediction while preserving its causal dependency structure.

Repair-Conditioned Future Writing. Direct masked training conditions on clean context, whereas streaming inference exposes future writing to self-generated intermediate states. As shown in Figure 3, we first obtain a repair-ready state through a no-gradient rollout,

$$\widetilde{R}_t = \text{Rollout}_\theta \left(c, C_t, W_t^{(0)} \right), \tag{13}$$

then append a new masked write window and optimize its future targets:

$$\mathcal{L}_{\text{rw}} = -\sum_{j=1}^{K} \log p_\theta \left(y_j^+ \mid c, C_t, \text{sg}[\widetilde{R}_t], W_{t+1}^{(0)} \right). \tag{14}$$

The repair-ready state is used only as self-generated context rather than a reconstruction target. During Stage 3, the two branches are jointly optimized:

$$\mathcal{L} = \tfrac{1}{2}\mathcal{L}_{\text{dir}} + \tfrac{1}{2}\mathcal{L}_{\text{rw}}. \tag{15}$$

5 Experimental Results

5.1 Experimental Setup

Training Data. We initialize FLUID-Omni from Qwen2.5-Omni-7B (Xu et al., 2025) and construct a multimodal training mixture through self-distillation from the same model. The source data are drawn from established multimodal instruction and understanding datasets: ShareGPT4V (Chen

Table 1: Main results across multimodal benchmarks. Data denotes the number of training samples. Bracketed labels indicate evaluation splits or settings. RWQA denotes RealWorldQA. Best results in diffusion methods are **bolded**, and our results are highlighted in .

Model	Size	Data	MMMU [val]	MMMU−Pro [vision]	AI2D [test]	MMBench [en-dev]	RWQA [test]	MMStar [test]
Autoregressive vision-language models								
Cambrian-1	8B	–	42.7	–	73.0	75.9	64.2	–
LLaVA-OV	7B	7.8M	48.8	–	81.4	80.8	66.3	61.7
LLaMA3-V	8B	16.5M	45.4	14.5	81.1	79.8	66.0	56.5
InternVL2	8B	–	47.9	–	83.7	82.7	64.7	61.2
InternVL2.5	8B	–	53.4	25.3	84.5	84.1	69.4	62.6
Qwen2.5-VL	7B	$>$ 9M	51.1	33.4	83.9	83.5	68.5	63.9
Diffusion vision-language models								
LaViDA-L	8B	1.6M	43.3	–	70.0	70.5	–	–
LaViDA-O	8B	–	45.1	–	76.7	76.4	–	–
Dimple	7B	1.3M	45.2	–	74.4	–	–	–
LLaDA-V	8B	16.5M	48.6	18.6	77.8	82.9	63.2	60.1
DiffusionVL	7B	738K	49.3	25.0	82.2	83.5	68.0	63.2
SDAR-VL	8B	$>$ 16M	53.0	–	79.9	82.2	66.5	59.9
FLUID-Omni	7B	253k	**53.1**	**46.5**	78.56	**89.0**	**71.2**	**65.2**

Model	Size	Data	ChartQA [test]	InfoVQA [val]	DocVQA [val]	OCRBench [test]	MathVista [testmini]	MathVerse [testmini]
Autoregressive vision-language models								
Cambrian-1	8B	–	73.3	–	77.8	62.4	49.0	–
LLaVA-OV	7B	7.8M	80.0	70.7	90.2	–	63.2	26.2
LLaMA3-V	8B	16.5M	77.8	58.9	86.2	–	62.1	29.0
InternVL2	8B	–	82.2	73.0	91.0	–	58.2	24.2
InternVL2.5	8B	–	82.8	75.5	92.0	82.2	63.2	27.2
Qwen2.5-VL	7B	$>$ 9M	60.3	79.3	91.6	80.2	67.3	–
Diffusion vision-language models								
LaViDA-L	8B	1.6M	64.6	34.2	59.0	–	44.8	27.2
LaViDA-O	8B	–	80.0	44.6	73.7	–	56.9	**36.9**
Dimple	7B	1.3M	63.4	–	–	–	42.3	–
LLaDA-V	8B	16.5M	78.3	66.3	83.9	57.3	59.7	28.5
DiffusionVL	7B	738K	84.2	69.9	90.8	76.2	61.5	–
SDAR-VL	8B	$>$ 16M	82.7	73.2	88.3	59.7	62.5	36.5
FLUID-Omni	7B	253k	**86.4**	**74.4**	**93.0**	**86.8**	**62.8**	**39.0**

et al., 2024b), LLaVA-Instruct (Liu et al., 2023), and LVIS-Instruct4V(Wang et al., 2023) for images; Speech-Instructions (Vikhrmodels, 2025) and multilingual FLEURS (Conneau et al., 2023) for speech; VideoChatGPT (Maaz et al., 2024), LLaVA-Video (Zhang et al., 2024), and OmniVideo (Cai et al., 2026) for videos; AVQA (Yang et al., 2022) and AVSD (Alamri et al., 2019) for audio-visual inputs; and DeepCtrl (DeepCtrl, 2025), Infinity-Instruct (Li et al., 2025), MOSS (Sun et al., 2024), and UltraChat (Ding et al., 2023) for text-only data. We use Qwen2.5-Omni-7B to generate teacher responses for these inputs, followed by quality filtering and unified-format conversion.

The resulting training set contains 252,791 instances, including 59,737 image, 54,407 audio, 75,867 video, 23,762 audio-visual, and 39,018 text-only instances. We filter refusals, empty or malformed responses, and highly repetitive outputs before training.

Implementation Details. We train FLUID-Omni for one epoch using two NVIDIA A40 GPUs with 48 GB memory each. The batch size is set to 1, with a learning rate of 2×10^{-4} and a warmup ratio of 0.1. Both the causal parallel adaptation stage and the pipeline-aligned training stage are initialized from Qwen2.5-Omni-7B.

Table 2: **End-to-end speech understanding and generation results.** Semantic quality is evaluated on WebQuestions, TriviaQA, CMMLU, and HSK, with Sem. Avg. reporting their average. Acoustic quality is measured by UTMOS, DNSMOS-P.835, and DNSMOS-P.808 (0–5, higher is better) on the generated speech responses. Type denotes the decoding paradigm. Our results are highlighted in .

		Semantic Quality					Acoustic Quality		
Models	**Type**	**WebQ**	**TriviaQA**	**CMMLU**	**HSK**	**Avg.**	**UTMOS**	**P.835**	**P.808**
Qwen3-Omni-30B-A3B-Instruct	AR	51.5	55.3	47.8	40.3	48.7	4.4	3.5	4.1
MiniCPM-o 2.6-9B	AR	40.0	40.2	49.2	80.7	52.5	4.1	3.4	4.0
Kimi-Audio-7B-Instruct	AR	33.7	38.2	66.9	97.4	59.1	2.9	3.2	3.6
GLM-4-Voice-10B	AR	32.0	36.4	52.6	71.1	48.0	4.2	3.5	4.1
FLUID-Omni	Diff.	35.7	38.2	72.3	92.4	59.7	3.6	3.4	4.1

Evaluation Protocol. We evaluate FLUID-Omni from two complementary perspectives. First, we assess whether causal diffusion adaptation preserves general multimodal understanding across a broad set of vision-language benchmarks. Second, we evaluate end-to-end speech interaction, covering both semantic response quality and acoustic generation quality. Unless otherwise specified, all FLUID-Omni results are obtained from the same 7B model trained with the above configuration.

5.2 Multimodal Understanding

We first evaluate whether FLUID-Omni preserves strong multimodal understanding after causal diffusion adaptation. As shown in Table 1, we compare against both autoregressive and diffusion-based vision-language models across a broad suite of benchmarks covering general perception, visual reasoning, document understanding, OCR, and multimodal mathematics.

FLUID-Omni achieves strong performance across most benchmarks and consistently outperforms existing diffusion-based models on MMMU, MMMU-Pro, MMBench, RealWorldQA, MMStar, ChartQA, InfoVQA, DocVQA, OCRBench, and MathVista. In particular, it reaches 53.1 on MMMU, 89.0 on MMBench, 71.2 on RealWorldQA, and 86.8 on OCRBench, while remaining competitive with strong autoregressive vision-language models. These results indicate that converting the pretrained omni model into a causal parallel diffusion generator does not sacrifice its general multimodal understanding capability.

5.3 End-to-End Speech Understanding and Generation

Semantic Quality. As shown in Table 2, FLUID-Omni achieves strong semantic performance across WebQuestions, TriviaQA, CMMLU, and HSK, reaching an average score of 59.4. In particular, it obtains 72.3 on CMMLU and 92.4 on HSK, demonstrating that causal diffusion decoding effectively preserves the semantic reasoning and spoken interaction capabilities of the pretrained omni-modal model. Overall, FLUID-Omni achieves competitive end-to-end semantic quality while replacing sequential autoregressive generation with parallel diffusion decoding.

Acoustic Quality. We further evaluate speech generation quality using UTMOS, DNSMOS-P.835, and DNSMOS-P.808. FLUID-Omni achieves scores of 3.6, 3.4, and 4.1, respectively, showing that the proposed parallel diffusion framework maintains high-quality speech generation under end-to-end spoken interaction. Together with the semantic results, these findings demonstrate that FLUID-Omni can support parallel speech generation while preserving both response correctness and perceptual quality.

5.4 Effect of Prior-Calibrated Commitment

Raw codec-token confidence can be biased by modality-specific token priors, leading to unreliable commitment decisions during streaming audio generation. Figure 4(a) shows that prior calibration redistributes token selection patterns, correcting confidence misalignment among frequently



Figure 4: (a) Effectiveness of adaptive Commit–Repair–Write decoding. (b) Codec-token confidence redistribution after prior calibration.

occurring codec tokens. The impact of this calibration is reflected in Table 3. Compared with raw confidence-based commitment, prior calibration improves the HSK semantic score from 90.5 to 92.4 and enhances acoustic quality, improving DNSMOS-P.835 from 2.6 to 3.4 and DNSMOS-P.808 from 2.8 to 3.8. These results demonstrate that calibrated confidence provides a more reliable criterion for audio token commitment.

Table 3: **Ablation on codec prior calibration.** We compare raw model confidence with prior-calibrated confidence for audio commitment.

Commitment	HSK	UTMOS	DNSMOS-P835	DNSMOS-P808
Confidence	90.5	3.1	26	2.8
Prior Calibrate	92.4	3.0	3.4	3.8

5.5 Efficiency Analysis

As shown in Figure 4(b), the sliding Commit–Repair–Write strategy achieves the best overall performance across multimodal benchmarks, obtaining an aggregate score of 37.14 compared with 31.56 for fixed-length remasking and 32.31 for repair-only decoding. Fixed-length remasking suffers from a rigid generation schedule that cannot adapt to the evolving uncertainty of future predictions, while repair-only refinement improves local correction but lacks concurrent future writing. In contrast, FLUID-Omni overlaps local repair with parallel future generation, allowing the commitment frontier to advance without introducing additional sequential decoding stages. These results demonstrate that explicitly coordinating commitment, refinement, and future writing is critical for efficient streaming diffusion decoding.

6 Conclusion

In this work, we introduce FLUID-Omni, a causal parallel diffusion framework for streaming omni-modal generation. By decoupling prediction from commitment through a Commit–Repair–Write decoding paradigm, FLUID-Omni enables parallel future generation while maintaining reliable streaming interaction. Extensive experiments demonstrate that FLUID-Omni achieves competitive multimodal understanding and end-to-end speech generation quality with improved streaming efficiency. Our results highlight the potential of parallel diffusion as a practical generation paradigm for building efficient and reliable omni-modal models.

REFERENCES

Huda Alamri, Vincent Cartillier, Abhishek Das, Jue Wang, Anoop Cherian, Irfan Essa, Dhruv Batra, Tim K Marks, Chiori Hori, Peter Anderson, et al. Audio visual scene-aware dialog. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7550–7559. IEEE, 2019.

Marianne Arriola, Aaron Gokaslan, Justin Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *International Conference on Learning Representations*, volume 2025, pp. 50726–50753, 2025.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021a.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021b.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, et al. Audiolm: a language modeling approach to audio generation. *IEEE/ACM transactions on audio, speech, and language processing*, 31:2523–2533, 2023a.

Zalán Borsos, Matt Sharifi, Damien Vincent, Eugene Kharitonov, Neil Zeghidour, and Marco Tagliasacchi. Soundstorm: Efficient parallel audio generation. *arXiv preprint arXiv:2305.09636*, 2023b.

Xinyue Cai, Chaoyou Fu, Yi-Fan Zhang, Ran He, and Caifeng Shan. Omnivideo-100k: A dataset for audio-visual reasoning through structured scripts and evidence chains. *arXiv preprint arXiv:2606.14702*, 2026.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11305–11315. IEEE, 2022.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024a.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. In *European Conference on Computer Vision*, pp. 370–387. Springer, 2024b.

Shuang Cheng, Yuhua Jiang, Zineng Zhou, Dawei Liu, Tao Wang, Linfeng Zhang, Biqing Qi, and Bowen Zhou. Sdar-vl: Stable and efficient block-wise diffusion for vision-language understanding. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 28882–28901, 2026.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. Fleurs: Few-shot learning evaluation of universal representations of speech. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pp. 798–805. IEEE, 2023.

DeepCtrl. Deepctrl-sft data. `https://www.modelscope.cn/datasets/deepctrl/deepctrl-sft-data`, 2025. ModelScope dataset.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*, 2022.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. Moshi: a speech-text foundation model for real-time dialogue. *arXiv preprint arXiv:2410.00037*, 2024.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 3029–3051, 2023.

Yijie Jin, Jiajun Xu, Yuxuan Liu, Chenkai Xu, Yi Tu, Jiajun Li, Dandan Tu, Xiaohui Yan, Kai Yu, Pengfei Liu, et al. Multi-block diffusion language models. *arXiv preprint arXiv:2606.29215*, 2026.

Jijie Li, Li Du, Hanyu Zhao, Bo-wen Zhang, Liangdong Wang, Boyan Gao, Guang Liu, and Yonghua Lin. Infinity instruct: Scaling instruction selection and synthesis to enhance language models. *arXiv preprint arXiv:2506.11116*, 2025.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in neural information processing systems*, 35: 4328–4343, 2022.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12585–12602, 2024.

Kentaro Mitsui, Koh Mitsuda, Toshiaki Wakatsuki, Yukiya Hono, and Kei Sawada. Pslm: Parallel generation of text and speech with llms for low-latency spoken dialogue systems. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 2692–2700, 2024.

OpenAI. Hello GPT-4o. `https://openai.com/index/hello-gpt-4o/`, 2024. Accessed as an official public system description.

Junhao Ruan, Bei Li, Yongjing Yin, Pengcheng Huang, Xin Chen, Jingang Wang, Xunliang Cai, Tong Xiao, and JingBo Zhu. Causal autoregressive diffusion language model. *arXiv preprint arXiv:2601.22031*, 2026.

Subham S Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184, 2024a.

Subham S Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184, 2024b.

Tianxiang Sun, Xiaotian Zhang, Zhengfu He, Peng Li, Qinyuan Cheng, Xiangyang Liu, Hang Yan, Yunfan Shao, Qiong Tang, Shiduo Zhang, et al. Moss: An open conversational large language model. *Machine Intelligence Research*, 21(5):888–905, 2024.

Vikhrmodels. Speech-instructions. `https://huggingface.co/datasets/Vikhrmodels/Speech-Instructions`, 2025. Hugging Face dataset.

Junke Wang, Lingchen Meng, Zejia Weng, Bo He, Zuxuan Wu, and Yu-Gang Jiang. To see is to believe: Prompting gpt-4v for better visual instruction tuning. *arXiv preprint arXiv:2311.07574*, 2023.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-Omni technical report. *arXiv preprint arXiv:2503.20215*, 2025. URL `https://arxiv.org/abs/2503.20215`.

Pinci Yang, Xin Wang, Xuguang Duan, Hong Chen, Runze Hou, Cong Jin, and Wenwu Zhu. Avqa: A dataset for audio-visual question answering on videos. In *Proceedings of the 30th ACM international conference on multimedia*, pp. 3480–3491, 2022.

Zebin You, Shen Nie, Xiaolu Zhang, JUN ZHOU, Zhiwu Lu, Ji-Rong Wen, and Chongxuan Li. Llada-v: Large language diffusion models with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10093–10105, 2026.

Runpeng Yu, Xinyin Ma, and Xinchao Wang. Dimple: Discrete diffusion multimodal large language model with parallel decoding. *arXiv preprint arXiv:2505.16990*, 2025.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507, 2021.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data, 2024. *URL https://arxiv. org/abs/2410.02713*, 17(2):6, 2024.



Figure 5: **Case study of FLUID-Omni streaming decoding.** The visualization illustrates the Commit–Repair–Write inference trajectory. FLUID-Omni first generates a future block through parallel writing, converts the generated block into a repair-ready window, performs local repair while writing the next future block in parallel, and finally commits the stabilized tokens before rolling forward.

A LLM USAGE

During the preparation of this manuscript, we used large language models, including ChatGPT (GPT-6, OpenAI) and Gemini (Google), in a limited and supportive capacity. Specifically, these models were used to assist with language refinement, including grammar correction, sentence restructuring, and improving the clarity of presentation. They were also occasionally consulted for exploring related literature and generating preliminary research ideas during the early stages of discussion. All suggested references were independently verified by the authors, and no technical content, experimental methodology, or scientific claims were generated or adopted without author review. All research ideas, algorithmic designs, implementations, experiments, analyses, and conclusions presented in this paper were conceived, conducted, and validated by the authors. The use of large language models was limited to auxiliary tasks, and all intellectual contributions of this work remain solely those of the authors.

B CASE STUDY OF STREAMING DECODING

To better illustrate the behavior of FLUID-Omni during streaming generation, we visualize a representative decoding trajectory in Figure 5. Unlike conventional diffusion decoding that repeatedly refines the entire future region, FLUID-Omni maintains a causal streaming interface through the Commit–Repair–Write pipeline.

The example shows that FLUID-Omni first generates a future block through parallel writing. Once the block is generated, it becomes a repair-ready window rather than being immediately committed. During the subsequent step, FLUID-Omni locally refines uncertain regions inside the current repair window while simultaneously writing the next future block in parallel. This allows the model to preserve useful future predictions without introducing the latency of sequential remasking.

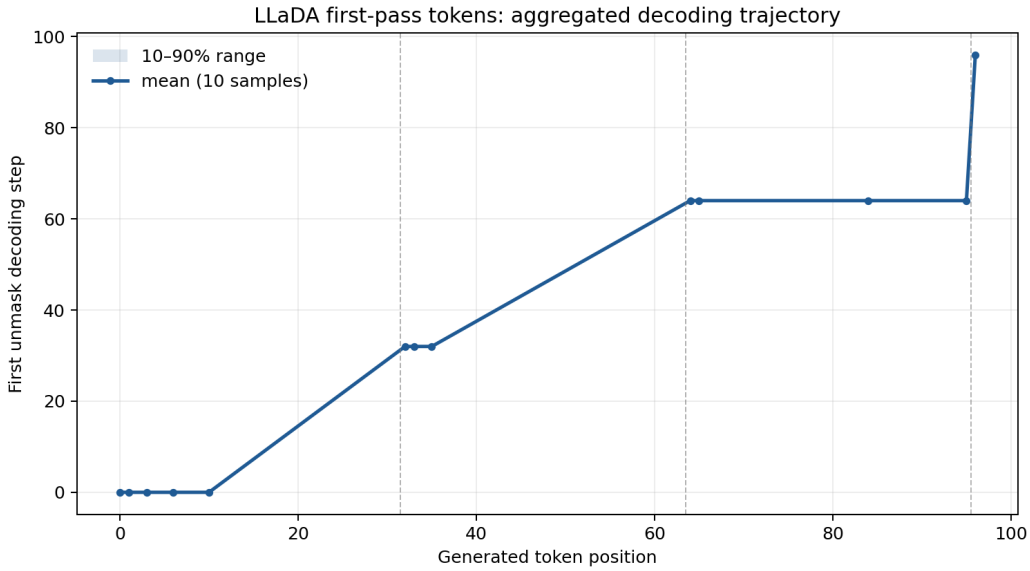

Figure 6: **First-pass denoising degradation along the generation trajectory.** We aggregate decoding traces from 20 randomly sampled GSM8K examples generated by LLaDA. Tokens farther from the causal frontier generally require more refinement steps, motivating adaptive commitment in streaming diffusion decoding.

After local refinement, stabilized tokens are committed into the prefix and the decoding process rolls forward with a newly created write window. This trajectory demonstrates how FLUID-Omni decouples prediction from commitment: future tokens can be generated aggressively in parallel, while only reliable regions are progressively exposed as stable context.

C Empirical Analysis of First-Pass Denoising

To investigate the reliability of parallel diffusion prediction during generation, we analyze the first-pass denoising trajectory of LLaDA on 20 randomly sampled GSM8K examples. Specifically, we record the decoding step at which each token position is first unmasked and aggregate the trajectories across samples.

As shown in Figure 6, tokens closer to the beginning of the generated sequence are typically resolved earlier, while tokens farther from the generation frontier require substantially more refinement steps before being exposed. This observation indicates that parallel diffusion predictions are not uniformly reliable across future positions: although multiple tokens can be predicted simultaneously, their stabilization process depends on their distance from the current decoding context.

This empirical finding motivates our Commit–Repair–Write decoding design, where future predictions are generated in parallel while commitment decisions are separated from initial prediction, allowing unstable regions to be refined before becoming persistent context.