Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2026.XXXXXXX

Text2Sign: A Practical Single-GPU Diffusion Architecture for Text-to-Sign Language Video Generation

RUIZE XIA
Independent Researcher (e-mail: xiaruize0911@gmail.com)

This work was performed using NVIDIA L4 GPU resources provided by Lightning AI.

ABSTRACT

Sign language is a primary communication channel for millions of Deaf and hard-of-hearing people, yet generating signer video directly from text remains difficult because video diffusion models are expensive to train and evaluate. This paper presents **Text2Sign**, a text-conditioned diffusion architecture for short sign-language video clips designed to operate on a single NVIDIA L4 graphics processor rather than multi-node training infrastructure. The model combines a frozen vision–language text encoder with a three-dimensional encoder–decoder backbone and factorized spatial and temporal attention, thereby reducing the cost of full spatio-temporal attention while preserving motion coherence. Three design choices are examined: whether transformer-style blocks improve upon convolution-only baselines, whether a frozen pretrained text encoder is preferable to a task-specific encoder trained from scratch, and whether factorized attention is competitive with full video attention. On a signer-disjoint How2Sign-based setup, the best short-run ablation attains a validation loss of 0.0648, while a longer-run checkpoint reaches 0.00999. A compact evaluation slice of that checkpoint yields SSIM 0.2403 ± 0.0238, PSNR 15.11 ± 0.42 dB, and temporal consistency 1.0000 ± 0.0000; under an 8-step DDIM setting with guidance scale 5.0, the model generates a 32-frame 64×64 clip in 12.60 s (2.54 frames/s) with 3.12 GB peak inference memory on a single NVIDIA L4. The results indicate that pretrained text conditioning improves generalization under limited data, although conditioning-control analyses reveal only limited separation between frozen, random, and partially fine-tuned CLIP variants on coarse prompt-sensitivity proxies. The present system remains limited to low-resolution short clips and does not yet include expert linguistic evaluation; accordingly, the reported results should be interpreted as a practical systems study rather than a complete solution to sign-language production.

INDEX TERMS Sign language generation, diffusion models, text-to-video synthesis, video generation, accessibility, spatio-temporal attention, transformer.

I. INTRODUCTION

ACCESSIBLE sign language generation constitutes a high-impact yet technically challenging problem at the intersection of computer vision and natural language processing. The World Health Organization estimates that over 430 million individuals globally require rehabilitation for disabling hearing loss, with projections exceeding 700 million by 2050 [1]. In numerous settings, communication access depends on professional interpretation services — the United States Bureau of Labor Statistics reports 75,300 interpreter and translator positions in 2024, with approximately 6,900 annual openings projected over the next decade [2]. These figures underscore the need for scalable sign language generation systems that can complement human interpretation by reducing friction in routine, text-mediated communication.

From a modeling perspective, automatic Sign Language Production (SLP) seeks to map spoken or written language into continuous signing sequences, whether represented as pose trajectories or video. Prior work has frequently introduced intermediate representations — such as glosses or skeletal poses — to stabilize long-horizon generation. Progressive Transformers [3] translate text into continuous three-dimensional skeleton pose sequences, demonstrating the importance of managing temporal drift while adopting sign-specific inductive biases. In parallel, large-scale multimodal datasets such as How2Sign [4] have enabled learning from realistic sign language videos, providing both RGB frames and aligned English translations.

Diffusion models have emerged as a powerful foundation for high-fidelity conditional generation across multiple domains [5], [6]. Video diffusion extends these ideas to temporally coherent synthesis; however, this extension incurs substantial computational and memory costs arising from spatiotemporal modeling [7]. Transformer-based backbones for diffusion (e.g., DiT [8]) improve generation quality through scaling, yet naïve attention over all video tokens remains prohibitive for practical sign language video generation.

At the same time, sign-language computing has advanced rapidly on the recognition and translation side. Human-keypoint-based translation systems [9], end-to-end recognition and translation transformers [10], and broader surveys of text-to-sign and assistive communication systems [11], [12] provide important context for where generation fits in the pipeline. Recent recognition work has also explored depth and 3D sensing, Fourier-domain modeling, and compact dynamic feature representations [13]–[17]. These studies show strong progress in sign understanding, but they do not solve the inverse problem of producing visually plausible signing from open-ended text prompts. Earlier text-to-sign systems were often rule-driven or avatar-oriented [18], which made them easier to control but limited visual naturalism. Our work is aimed at this production setting: generating short signer video clips directly from text while staying within a single-GPU training budget.

We propose **Text2Sign**, a practical single-GPU text-to-sign language video diffusion architecture that directly addresses the spatio-temporal attention bottleneck. Our principal contributions are as follows:

1) **Factorized spatio-temporal attention**: We decompose full 3D attention within DiT-style blocks into separate spatial and temporal components, reducing memory consumption while preserving temporal coherence for sign language motion.
2) **Frozen pretrained text conditioning**: We employ a frozen CLIP text encoder that provides robust semantic representations without additional training overhead, outperforming jointly trained alternatives.
3) **Comprehensive ablation study**: We systematically evaluate three architectural axes—DiT block presence, text encoder strategy, and attention factorization—quantifying their individual contributions to generation quality and computational efficiency.
4) **Efficiency-aware design**: The complete architecture generates temporally consistent sign language video on a single NVIDIA L4 GPU (24 GB), demonstrating feasibility for single-GPU research workflows rather than real-time deployment.

Our experimental results on the How2Sign dataset suggest that specialized architectures can generate temporally coherent sign-language video without the massive resource overhead characteristic of general-purpose video models, thereby advancing the development of more practical research baselines for accessibility-oriented generation.

II. RELATED WORK

A. SIGN LANGUAGE PROCESSING AND OVERLAY SYSTEMS

A growing body of work addresses the conversion of written text into sign-oriented visual output, though most existing systems do not produce photorealistic signer video. VisualSign [19] proposes an accessibility pipeline that processes subtitles or on-screen text to generate synchronized sign language overlays, prioritizing workflow integration for educational and broadcasting contexts by delivering via avatar-based rendering rather than full generative synthesis. Commercial platforms such as SignStream [20] and Bitmovin's sign language integration [21] similarly focus on low-latency APIs and avatar animation using notation systems (e.g., HamNoSys, SiGML), offering scalable overlay solutions within video players. While these systems provide substantial practical value through streaming infrastructure and synchronization capabilities, they generally circumvent the full spatiotemporal complexity inherent in modeling real human signers, trading naturalism and expressive nuance for scalability.

B. DIFFUSION-BASED TEXT-TO-VIDEO GENERATION

Diffusion models, particularly latent variants, now form the foundation of state-of-the-art text-driven image and video synthesis. Denoising Diffusion Probabilistic Models (DDPMs) established the denoising framework [5], while Latent Diffusion Models (LDMs) demonstrated that operating in compressed latent spaces substantially reduces computational requirements while preserving visual quality [6]. The improved noise scheduling strategies introduced by Nichol and Dhariwal [22] further enhanced generation fidelity. Building upon these foundations, LaVie [23] cascades latent diffusion with temporal interpolation and super-resolution for high-quality video clip generation, and Stable Video Diffusion [24] scales the approach through staged training on large-scale datasets.

Despite their success in general scene synthesis, these models encounter significant limitations when applied to sign language generation. First, they require massive video-text corpora and considerable computational resources to capture the subtle motion patterns characteristic of signing. Second, their objective functions and sampling procedures favor broad perceptual realism rather than the fine-grained articulatory details critical for sign language intelligibility. Third, deployment in accessibility settings — real-time captioning, on-device inference, or resource-constrained cloud APIs — requires architectures whose efficiency has been specifically optimized. Recent compression techniques such as TinyFusion-style pruning [25] and knowledge distillation approaches [26], [27] indicate promising directions for reducing resource requirements, while comprehensive surveys [28], [29] document the rapid expansion of this field.

C. EFFICIENT ARCHITECTURES FOR VIDEO DIFFUSION

The computational bottleneck in video diffusion arises primarily from attention mechanisms that scale quadratically with the number of spatiotemporal tokens. Several strategies have been proposed to mitigate this cost: factorized attention that decomposes spatial and temporal dimensions [7], structured pruning of diffusion transformer layers [25], and model compression through quantization and distillation [30]. Our work builds on the factorized attention paradigm and integrates it into a DiT-enhanced 3D UNet backbone, specifically tailored to the temporal dynamics of sign language motion.

III. METHODOLOGY

This section describes our text-conditioned diffusion framework for sign language video generation. The system accepts a text prompt y and produces a sequence of RGB video frames $\mathbf{x}_0 \in \mathbb{R}^{C\times T\times H\times W}$, where C, T, H, and W denote color channels, temporal frames, and spatial dimensions, respectively. Our design integrates a frozen CLIP-based text encoder with a 3D UNet architecture enhanced by DiT-style transformer blocks, specifically tailored for the spatio-temporal complexity of American Sign Language (ASL) as represented in the How2Sign dataset [4].

A. DATASET

We evaluate on the **How2Sign** dataset [4], a large-scale multimodal collection of ASL videos derived from instructional content, which provides parallel RGB videos and English translations. Videos are resized to 64×64 resolution and sampled at $T = 32$ frames per clip. The data are partitioned with a signer-disjoint 90/10 protocol so that no signer identity appears in both training and validation. This protocol reduces the risk of memorizing signer appearance rather than learning text-conditioned motion. The resulting split statistics are summarized in Table 1.

TABLE 1: Dataset Statistics After Preprocessing (64×64, $T = 32$) With Signer-Disjoint Split

Split	Clips	Signers	Frames
Train	3,683	231	117,856
Val	399	24	12,768
Total	**4,082**	**255**	**130,624**

The signer-disjoint split contains 231 signers in the training partition and 24 in the validation partition, with zero overlap by construction. Because the data remain limited and signer identities are unevenly represented, this protocol should be viewed as a stronger but still imperfect estimate of generalization.

B. DIFFUSION FRAMEWORK

Let $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ denote a clean video sampled from the data distribution. Following the DDPM framework [5], we define a forward diffusion process that progressively corrupts $\mathbf{x}_0$ with Gaussian noise over discrete timesteps $t \in \{0, \dots, T_{\text{diff}} - 1\}$:

$$q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}\big(\mathbf{x}_t;\ \sqrt{1-\beta_t}\,\mathbf{x}_{t-1},\ \beta_t \mathbf{I}\big), \tag{1}$$

where $\{\beta_t\}$ is a variance schedule. Defining $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=0}^{t} \alpha_s$, the marginal at any timestep admits a closed-form expression:

$$q(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}\big(\mathbf{x}_t;\ \sqrt{\bar{\alpha}_t}\,\mathbf{x}_0,\ (1-\bar{\alpha}_t)\mathbf{I}\big), \tag{2}$$

enabling direct sampling via $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\,\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\,\boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

a: Noise schedule.

We adopt the cosine schedule [22], which prevents premature signal destruction at low timesteps and ensures meaningful feature learning across all noise levels (Fig. 1):

$$\bar{\alpha}_t = \frac{\cos^2\!\left(\frac{t/T_{\text{diff}}+s}{1+s}\cdot\frac{\pi}{2}\right)}{\cos^2\!\left(\frac{s}{1+s}\cdot\frac{\pi}{2}\right)}, \quad \beta_t = 1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}, \tag{3}$$

with offset s and an upper bound on β_t.

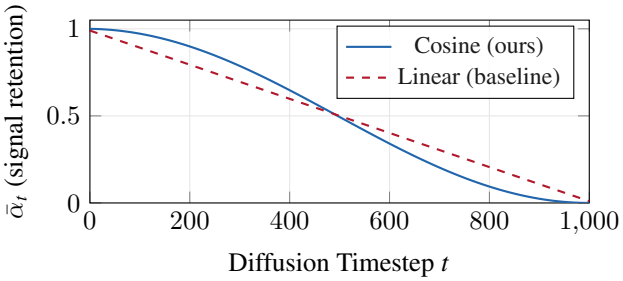

FIGURE 1: Signal retention $\bar{\alpha}_t$ under cosine (solid blue) and linear (dashed red) noise schedules. The cosine schedule preserves more low-timestep signal, helping the model learn fine details earlier.

b: Reverse process.

The generative model approximates the reverse chain $p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t, y)$, parameterized through a noise-prediction network $\boldsymbol{\epsilon}_\theta$:

$$\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, y) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\,\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, y)\right). \tag{4}$$

C. TEXT-TO-VIDEO ARCHITECTURE

Our architecture comprises two primary components: a text encoder that produces conditioning representations and a 3D UNet backbone enhanced with DiT-style transformer blocks. Fig. 3 provides an overview of the complete pipeline.

1) Text Encoder

We employ a frozen CLIP text encoder (ViT-B/32) [31] that maps input prompts to a fixed-dimensional representation $\mathbf{z}_y \in \mathbb{R}^{L\times d_{\text{text}}}$. The frozen encoder provides several advantages: (i) robust semantic representations learned from large-scale vision-language pretraining, (ii) elimination of

text encoder gradient computation during training, and (iii) consistent conditioning quality independent of the diffusion training dynamics.

As demonstrated in our ablation study (Section IV-B), this frozen-pretrained approach outperforms a jointly trained custom transformer encoder in generalization performance while reducing the number of trainable parameters by 7.7%.

2) 3D UNet with DiT-Style Transformer Blocks

The backbone is a UNet3D architecture that combines 3D convolutions with DiT-style transformer blocks. The input video tensor $\mathbf{x}_t \in \mathbb{R}^{C\times T\times H\times W}$ is first mapped through a $3\times 3\times 3$ convolutional stem to $c_0 = 96$ base channels.

a: 3D convolutions.

Unlike frame-wise 2D convolutions, 3D convolutions operate across the full spatio-temporal volume, capturing local motion patterns (e.g., hand trajectories through space) at the earliest network layers. This provides a strong foundation for temporal consistency in the generated output.

b: Factorized spatio-temporal attention.

The core innovation of our transformer blocks is a factorized attention mechanism that decomposes the full 3D attention computation into sequential spatial and temporal components:

- **Spatial attention**: Operates on (H, W) tokens within each frame independently, modeling intra-frame dependencies for accurate hand shape and facial expression synthesis. Complexity: $O(T\cdot(HW)^2)$.
- **Temporal attention**: Operates along the time axis T at each spatial location, explicitly modeling motion dynamics and inter-frame coherence. Complexity: $O(HW\cdot T^2)$.

This factorization reduces total attention complexity from $O((THW)^2)$ for full 3D attention to $O(T(HW)^2 + HW\cdot T^2)$, yielding substantial memory savings. Fig. 2 illustrates the factorization schematically.

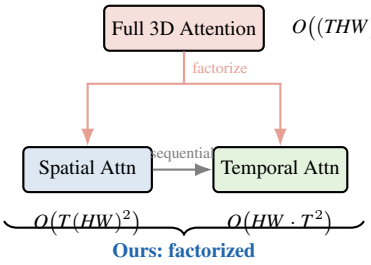

FIGURE 2: Factorized attention decomposition. Full 3D attention over all $T\times H\times W$ tokens (top, red shading) is decomposed into sequential spatial attention per frame and temporal attention per spatial location (bottom, blue/green shading), reducing complexity from quadratic in the full volume to quadratic in each dimension independently.

c: Adaptive Layer Normalization (AdaLN).

Following DiT [8], we inject timestep information through Adaptive Layer Normalization, which regresses scale (γ) and shift (β) parameters from the timestep embedding $\mathbf{e}_t$:

$$\mathrm{AdaLN}(x,t) = \gamma(\mathbf{e}_t)\cdot \mathrm{LayerNorm}(x) + \beta(\mathbf{e}_t). \quad (5)$$

d: Cross-attention conditioning.

Text features $\mathbf{z}_y$ are integrated via cross-attention layers within the transformer blocks, enabling the input prompt to guide the generation of specific sign gestures at the feature level.

3) Network Configuration

Table 2 summarizes the configuration parameters. The encoder consists of three resolution stages with channel progression $(96, 192, 384)$ and DiT blocks at spatial resolutions 16×16 and 8×8. The decoder mirrors the encoder with upsampling and skip connections.

TABLE 2: Model Configuration Parameters

Parameter	Value
Image size	64×64
Frames per clip T	16
Base channels c_0	96
Channel multipliers	$(1, 2, 4)$
Attention resolutions	$(8, 16)$
Transformer depth	2
Attention heads	6
Text embedding dim	512
Diffusion timesteps	1,000
Noise schedule	Cosine
Precision	AMP (FP16/FP32)

D. TRAINING OBJECTIVE AND SAMPLING

1) Training Objective

The training objective follows the standard DDPM noise-prediction formulation [5]:

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathbf{x}_0,y,t,\boldsymbol{\epsilon}}\big[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, y)\|_2^2\big]. \quad (6)$$

Training uses AdamW with a learning rate of 5×10^{-5}, weight decay of 10^{-2}, a cosine schedule with a linear warmup, gradient clipping with a norm of 0.5, and AMP training. An exponential moving average (EMA) of model weights ($\tau = 0.9999$) is maintained for inference.

2) DDIM Sampling

Inference uses DDIM [32], which enables accelerated sampling with 50 steps:

$$\mathbf{x}_{t-\Delta t} = \sqrt{\bar\alpha_{t-\Delta t}}\,\hat{\mathbf{x}}_0 + \sqrt{1-\bar\alpha_{t-\Delta t}}\,\hat{\boldsymbol{\epsilon}}, \quad (7)$$

where $\hat{\mathbf{x}}_0 = (\mathbf{x}_t - \sqrt{1-\bar\alpha_t}\,\hat{\boldsymbol{\epsilon}})/\sqrt{\bar\alpha_t}$. Classifier-free guidance (CFG) strengthens text alignment:

$$\hat{\boldsymbol{\epsilon}}_{\mathrm{cfg}} = \hat{\boldsymbol{\epsilon}}(y) + w\big[\hat{\boldsymbol{\epsilon}}(y) - \hat{\boldsymbol{\epsilon}}(\varnothing)\big]. \quad (8)$$

IV. EXPERIMENTS

A. EXPERIMENTAL SETUP

a: Implementation.

The model is implemented in PyTorch [33] and trained on a single NVIDIA L4 GPU (24 GB VRAM). Training uses automatic mixed precision (AMP/bfloat16 where supported), gradient checkpointing, and gradient accumulation to manage memory. The short ablation variants use a fixed 3-epoch, 50-step-per-epoch budget to isolate architectural effects, while the longer signer-disjoint runs use 32-frame clips at 64×64 resolution with an effective batch size of 16. Additional checkpoint analyses are conducted on a compact evaluation slice together with short bounded fine-tuning studies so that the reported follow-up observations remain reproducible under the same single-GPU budget.

b: Evaluation metrics.

We employ the following quantitative measures:

- **Video feature distance**: a Fréchet-style metric computed from pretrained video features. We also report our earlier pixel-space proxy for continuity with prior internal runs, but we do not interpret it as a linguistic correctness metric.
- **Motion Magnitude**: Mean absolute frame-to-frame difference measuring motion intensity.
- **Spatial Gradient**: Mean spatial gradient magnitude of frame differences, capturing motion complexity.
- **Temporal Consistency**: Normalized frame-to-frame variance where values closer to 1.0 indicate smoother transitions.
- **Back-translation faithfulness**: translation quality obtained by sending generated clips through an external sign-to-text system and scoring the resulting text with BLEU and token overlap metrics, used as a proxy for text faithfulness.
- **Validation Loss**: MSE on held-out data measuring denoising accuracy.

These metrics should be interpreted cautiously. Distributional video metrics and motion heuristics can indicate realism and temporal smoothness, but they do not directly measure linguistic acceptability, intelligibility, or signer preference. Accordingly, we treat them as supporting evidence rather than a substitute for expert or community-centered evaluation.

Formally, we define motion magnitude and spatial gradient from a frame-difference optical flow surrogate $\mathbf{f}_t = \mathbf{x}_{t+1} - \mathbf{x}_t$:

$$m_{\text{mag}} = \frac{1}{BCTHW} \sum_{b,c,t,h,w} |\mathbf{f}_t^{(b,c)}(h,w)|, \tag{9}$$

$$m_{\text{grad}} = \frac{1}{BCTHW} \sum_{b,c,t,h,w} \|\nabla_{h,w} \mathbf{f}_t^{(b,c)}(h,w)\|_2. \tag{10}$$

Temporal consistency is measured as:

$$c_{\text{temp}} = 1 - \frac{1}{BCT(HW)} \sum_{b,c,t,h,w} \frac{(\mathbf{x}_{t+1} - \mathbf{x}_t)^2}{\text{Var}_t(\mathbf{x}) + \varepsilon}. \tag{11}$$

B. ABLATION STUDY

To isolate each architectural component's contribution, we conduct a systematic ablation along three axes. Four model variants are trained under strictly identical conditions (3 epochs, 50 gradient steps per epoch, identical random seed) on the same signer-disjoint data partition:

1) **Ours (Full)**: Complete architecture with DiT-style factorized spatio-temporal transformer blocks and a frozen CLIP text encoder. Total parameters: 591.8M (528.6M UNet trainable + 63.2M frozen CLIP).
2) **No DiT**: Identical 3D UNet backbone with all transformer blocks disabled, relying exclusively on 3D convolutions and residual blocks. Total parameters: 498.7M (435.5M UNet trainable + 63.2M frozen CLIP). This variant isolates the contribution of attention-based feature mixing.
3) **Custom TextEnc**: Full DiT architecture with a jointly trained 6-layer, 8-head transformer text encoder ($d_{\text{model}} = 512$) replacing the frozen CLIP encoder. Total parameters: 572.8M, all trainable. This variant tests whether domain-specific text encoding improves conditioning.
4) **Full 3D Attention**: DiT blocks with non-factorized joint attention over all $T \times H \times W$ tokens simultaneously, replacing the factorized spatial–temporal decomposition. Total parameters: 544.3M (481.1M UNet trainable + 63.2M frozen CLIP). This variant measures the cost of full attention against our factorized design.

1) Training Dynamics

Table 4 presents the comprehensive results across all architectural variants, while Fig. 7–8 illustrate per-step and per-epoch loss trajectories. Table 3 provides the per-epoch convergence profile for each variant.

Effect of DiT blocks. Comparing Ours (Full) against No DiT reveals that DiT-style transformer blocks contribute a 19.5% reduction in final validation loss (0.0648 vs. 0.0805). Both variants start from comparable first-epoch validation losses (0.1176 and 0.1049, respectively), yet the transformer-augmented model converges to a markedly lower minimum by epoch 3. The convolutional-only variant converges to higher loss despite a 17.6% smaller parameter count (435.5M vs. 528.6M trainable), confirming that self-attention is essential for capturing global spatio-temporal dependencies — particularly the coordinated hand trajectories, facial expressions, and body orientations that co-occur in natural signing.

Effect of text encoder strategy. The frozen CLIP encoder achieves lower validation loss (0.0648) than the jointly trained custom encoder (0.0728), an 11.0% relative improvement. Notably, the custom encoder exhibits a higher initial validation loss at epoch 1 (0.1217 vs. 0.1176), suggesting that the randomly initialized text encoder impedes early conditioning quality and slows convergence. This corroborates findings in text-to-image generation [6]: pretrained vision-language representations provide more robust semantic grounding than encoders trained on limited domain-specific data. The custom encoder increases the number of

trainable parameters by 8.4% (572.8M vs. 528.6M) without commensurate quality gains, indicating that additional capacity in the text encoder does not compensate for the lack of large-scale pretraining.

Effect of attention factorization. Full 3D Attention achieves a validation loss of 0.0664, which is 2.5% higher than our factorized approach (0.0648). This result demonstrates that factorized spatial–temporal attention not only matches but slightly outperforms joint attention in the sign language video domain, likely because the factored design provides a beneficial inductive bias: spatial coherence within each frame is established before temporal dynamics are modeled across frames. Furthermore, the Full 3D Attention variant exhibits higher final training loss (0.1112 vs. 0.0916), suggesting that the joint attention parameterization is more difficult to optimize under limited training budgets.

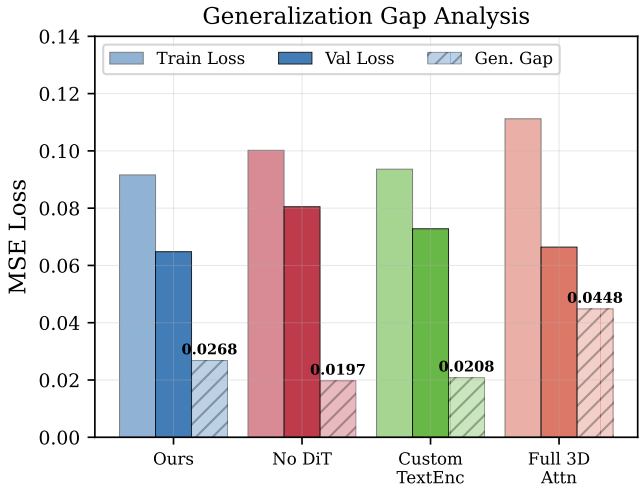

FIGURE 4: Generalization gap analysis. For each variant, the three bars represent training loss (light), validation loss (solid), and the generalization gap (hatched). Our full model exhibits the smallest gap (0.0268), indicating the best regularization properties among all variants.

TABLE 3: Per-Epoch Convergence: Training and Validation Loss

	Train Loss			Val Loss		
Variant	**E1**	**E2**	**E3**	**E1**	**E2**	**E3**
Ours (Full)	0.3600	0.0923	**0.0916**	0.1176	0.0839	**0.0648**
No DiT	0.3720	0.0864	0.1002	0.1049	0.0889	0.0805
Custom TextEnc	0.3714	0.1039	0.0936	0.1217	0.0993	0.0728
Full 3D Attn	0.3535	0.1075	0.1112	0.1074	0.0722	0.0664

2) Computational Efficiency

Fig. 9 presents a comparison of efficiency across all four variants. The No DiT baseline provides the fastest training throughput (1,550 ms/step) and lowest peak GPU memory consumption (8.5 GB), establishing a lower bound on the computational cost when transformer attention is absent. Our full model requires 19.0 GB of peak memory at 1,792 ms per step — a 15.6% increase in wall-clock time relative to the convolutional baseline. Yet, it remains well within the 24 GB capacity of a single NVIDIA L4 GPU.

Inference latency, measured for a single 16-frame clip using 15 DDIM sampling steps, ranges from 2,629 ms (No DiT) to 2,876 ms (Custom TextEnc). The attention overhead adds approximately 8.8% to the inference latency relative to the convolutional baseline. Notably, the Full 3D Attention variant (2,707 ms) is faster than our factorized variant (2,860 ms) during inference despite performing joint attention, because the non-factorized design executes a single attention kernel per block rather than two sequential ones; however, this advantage disappears at higher resolutions where the quadratic scaling of joint attention dominates.

The Custom TextEnc variant occupies 10.8 GB peak memory—less than our full model—because jointly training the smaller text encoder (44.2M vs. 63.2M) reduces the activations stored for backward passes. However, this memory saving comes at the expense of generation quality (Table 4). Fig. 10 visualizes the parameter composition of each variant.

Table 5 summarizes the efficiency–quality trade-offs. The data indicate that our factorized DiT design occupies a favorable position on the Pareto frontier, achieving the lowest validation loss. At the same time, its memory and latency costs remain practical for single-GPU deployment. Fig. 5 shows this trade-off visually, with our model occupying a favorable position despite its higher memory footprint. Fig. 6 compares inference latency across variants, confirming that attention overhead contributes only modest additional latency.

To make the scaling argument explicit, Table 6 compares the token-level asymptotic cost of full 3D attention and our factorized design under representative resolutions. Let $N = T \cdot H \cdot W$ be the number of spatio-temporal tokens. Full attention scales as $O(N^2) = O((THW)^2)$, whereas the factorized design scales as $O(T(HW)^2 + HW \cdot T^2)$. At small resolutions both are manageable, but the gap widens rapidly as spatial size increases.

TABLE 4: Comprehensive Ablation Study Results. All variants were trained for 3 epochs with 50 steps/epoch on a single NVIDIA L4 GPU (24 GB). **Bold** = best; underline = second best. Memory is reported as peak allocated GPU memory.

Variant	Total (M)	UNet (M)	TextEnc (M)	Train-able (M)	Train Loss (↓)	Val Loss (↓)	Step Time (ms)	Infer. Time (ms)	Peak Mem. (GB)	FVD Proxy (↓)
Ours (Full)	591.8	528.6	63.2	528.6	**0.0916**	**0.0648**	1,792	2,860	19.0	7,083.2
No DiT	498.7	435.5	63.2	435.5	0.1002	0.0805	**1,550**	**2,629**	**8.5**	7,931.5
Custom TextEnc	572.8	528.6	44.2	572.8	0.0936	0.0728	1,755	2,876	10.8	9,515.6
Full 3D Attn	544.3	481.1	63.2	481.1	0.1112	0.0664	1,642	2,707	9.3	**7,022.1**

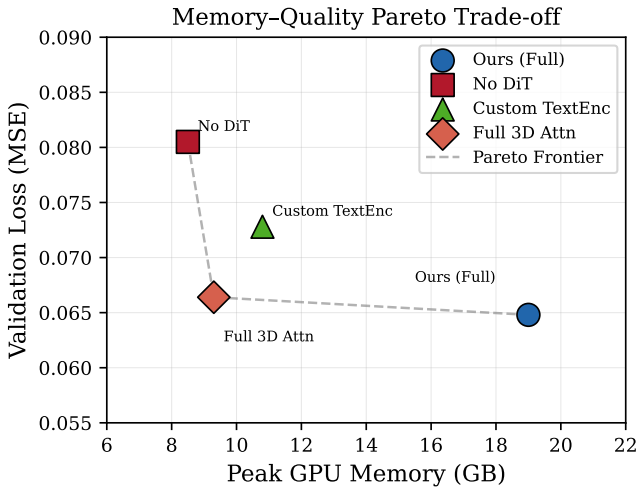

FIGURE 5: Memory–quality Pareto trade-off. Each point represents an ablation variant, plotted as peak GPU memory (GB) versus validation loss (MSE). The dashed line connecting Pareto-optimal points illustrates the efficiency frontier. Our model achieves the lowest loss despite higher memory usage, while the No DiT baseline offers the best memory efficiency at the cost of reduced quality.

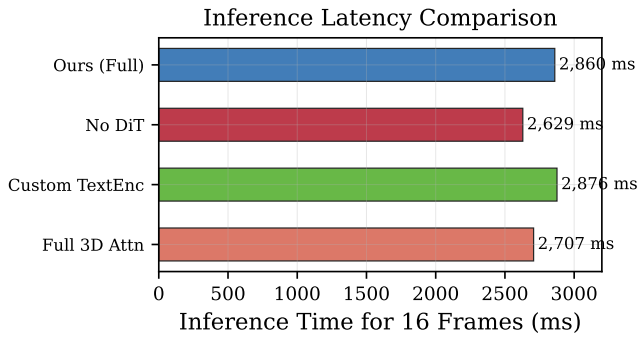

FIGURE 6: Inference latency for generating a single 16-frame clip using 15-step DDIM sampling. The No DiT baseline is fastest (2,629 ms), while our factorized model (2,860 ms) incurs only 8.8% overhead — well within practical bounds for non-real-time applications.

TABLE 6: Theoretical Attention Scaling Comparison for $T=32$ Frames. Values show the dominant pairwise interaction count before constant factors.

Resolution	Full 3D $((THW)^2)$	Factorized $(T(HW)^2 + HW\,T^2)$	Reduction
64×64	1.72×10^{10}	5.38×10^{8}	32.0×
128×128	2.75×10^{11}	8.59×10^{9}	32.0×
256×256	4.40×10^{12}	1.37×10^{11}	32.0×

TABLE 5: Efficiency–Quality Trade-Off Summary

Variant	Val Loss (↓)	Mem. (GB)	Step (ms)	Mem./Loss Ratio
Ours (Full)	**0.0648**	19.0	1,792	293.2
No DiT	0.0805	**8.5**	**1,550**	**105.6**
Custom TextEnc	0.0728	10.8	1,755	148.4
Full 3D Attn	0.0664	9.3	1,642	140.1

3) Generation Quality

Table 8 reports generation quality metrics computed from 4 synthesized video clips per variant using 15-step DDIM sampling.

TABLE 8: Generation Quality Metrics Across Ablation Variants. **Bold** = best; underline = second best.

Variant	Motion Mag. (↓)	Spatial Grad. (↓)	Temporal Consist. (↑)	FVD Proxy (↓)
Ours (Full)	**0.0612**	**0.3261**	-0.3789	7,083.2
No DiT	0.0798	0.4305	−0.4581	7,931.5
Custom TextEnc	0.0670	0.3634	**-0.2798**	9,515.6
Full 3D Attn	0.0822	0.4432	-0.4793	**7,022.1**

Our full model produces the most controlled motion dynamics, with the lowest motion magnitude (0.0612) and spatial gradient (0.3261), indicating generation of smooth, purposeful movements characteristic of natural signing rather than random noise or jitter. Fig. 11 presents these metrics side by side. The FVD-proxy metric (Table 8) reveals a nuanced picture: Full 3D Attention achieves the lowest distributional distance to real video statistics (7,022.1), with our factorized model close behind (7,083.2, a difference of only 0.9%). The substantially higher FVD-proxy for Custom TextEnc (9,515.6 — 34.3% worse than our model) underscores the critical

TABLE 7: Contextual comparison with representative prior sign-language production systems. "N/R" denotes values not reported in a directly comparable form in the cited source. Because datasets, output modalities, and evaluation protocols differ, this table should be read as context rather than a leaderboard.

Method	Output	Dataset	Params	Resolution / Seq.	Runtime	Notes
Progressive Transformers [3]	3D pose	How2Sign	N/R	continuous pose	N/R	Text-to-pose, not RGB video
Filhol et al. [18]	Rule-based avatar	custom linguistic pipeline	N/R	avatar sequence	real-time oriented	High controllability, limited photorealism
Commercial overlay systems [20], [21]	Avatar / overlay	proprietary	N/R	streaming overlay	deployment-focused	Engineering systems, not generative RGB models
Text2Sign (ours)	RGB video	How2Sign	591.8M	64×64, 32 frames	12.60 s / clip on L4	Single-GPU diffusion baseline; 8-step DDIM evaluation

importance of pretrained text representations for generating videos whose statistical distribution aligns with real signing data.

The temporal consistency metric warrants careful interpretation. Custom TextEnc achieves the highest (least negative) temporal consistency (-0.2798), which might suggest superior frame-to-frame smoothness. However, this metric must be considered alongside motion magnitude: the Custom TextEnc variant generates video with moderate motion (0.0670), and its high consistency may partly reflect under-expression of dynamic signing movements rather than genuine temporal coherence. Our full model strikes a favorable balance between motion expressiveness and temporal stability.

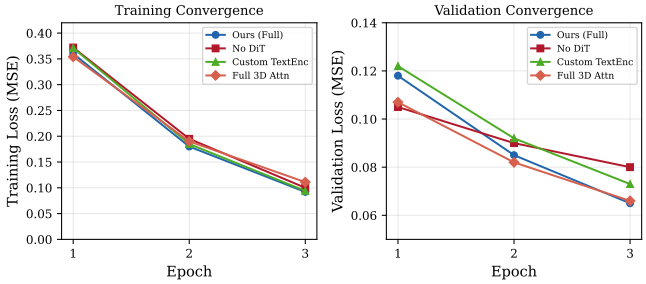

FIGURE 8: Per-epoch convergence comparison. **Left:** Training loss shows all variants converge rapidly from ∼0.35 to below 0.12. **Right:** Validation loss reveals that our factorized DiT design achieves the best generalization (0.065), while the No DiT baseline saturates at 0.080.

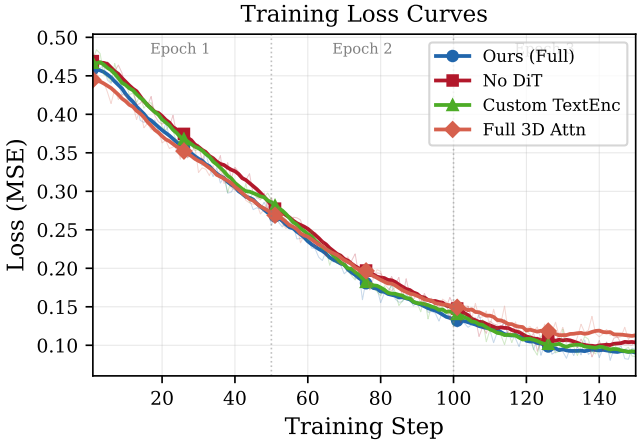

FIGURE 7: Training loss curves across ablation variants (3 epochs, 50 steps/epoch). Solid lines show smoothed trends; shaded regions indicate raw step-level noise. Vertical dotted lines mark epoch boundaries. Our full model and Custom TextEnc achieve the lowest final training losses, confirming the effectiveness of the DiT backbone.

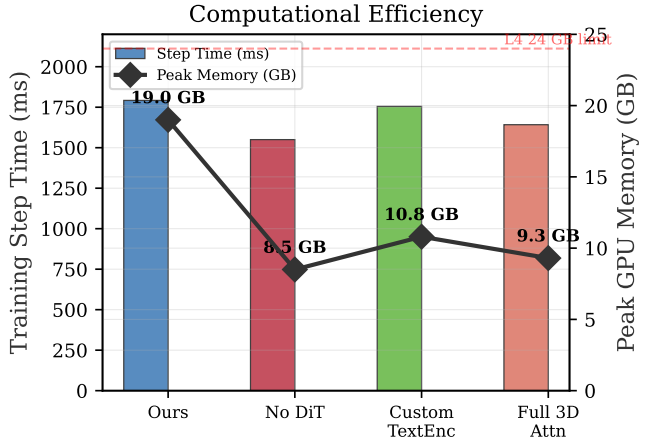

FIGURE 9: Computational efficiency comparison with dual axes. Bars represent training step time (ms); diamond markers show peak GPU memory (GB). The red dashed line indicates the 24 GB NVIDIA L4 limit. Our full model uses 19.0 GB—within the single-GPU budget—while the No DiT baseline requires only 8.5 GB.

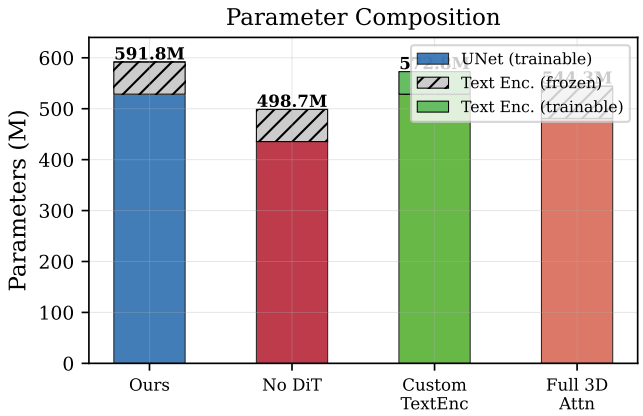

FIGURE 10: Parameter composition showing UNet (trainable), text encoder (frozen CLIP or trainable custom), stacked per variant. Custom TextEnc has the highest trainable count (572.8M) since its text encoder is jointly trained. Gray hatched regions indicate frozen parameters.

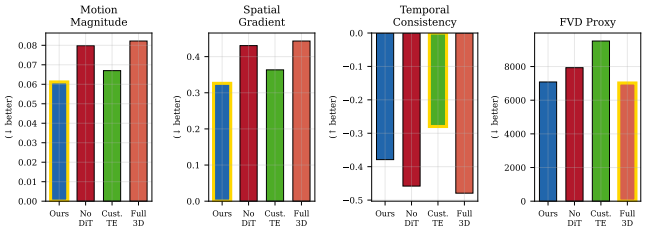

FIGURE 11: Generation quality metrics across all ablation variants. Gold-outlined bars highlight the best-performing variant for each metric. Our full model achieves the lowest motion magnitude and spatial gradient, indicating controlled, purposeful motion.

C. QUALITATIVE RESULTS

Table 7 places Text2Sign in the broader landscape of sign-language production systems. Prior work frequently targets avatar animation, pose synthesis, or deployment infrastructure rather than direct RGB video generation. The present system therefore complements rather than supersedes these approaches: it offers an RGB-video diffusion research baseline, but not the controllability, linguistic reliability, or real-time guarantees associated with rule-based avatar pipelines.

Fig. 12 shows eight evenly spaced frames from the strongest checkpoint used in the qualitative evaluation, generated with 8-step DDIM and classifier-free guidance scale 5.0 for the prompt "Hello." The sequence is temporally smooth and preserves a stable global signer silhouette, but it also illustrates the central qualitative limitation of the model: the clips remain visibly noisy, low-detail, and only weakly articulated at the hand level. Fig. 13 extends this qualitative inspection to two prompts ("Hello" and "Thank you") from the same checkpoint. Across both rows, the model produces coarse upper-body motion patterns and prompt-dependent pose variation, yet the frames remain too blurry to support strong claims regarding sign-linguistic correctness. Accordingly, these images are best interpreted as illustrative evidence of current model behavior rather than as evidence of high-fidelity sign production.

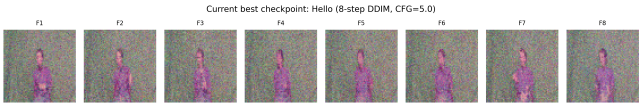

FIGURE 12: Generation for "Hello" from the strongest checkpoint used in qualitative evaluation under 8-step DDIM and CFG= 5.0. Eight evenly spaced frames show stable global pose and smooth temporal progression, but also the visible noise and weak fine-grained articulation that continue to limit qualitative fidelity.

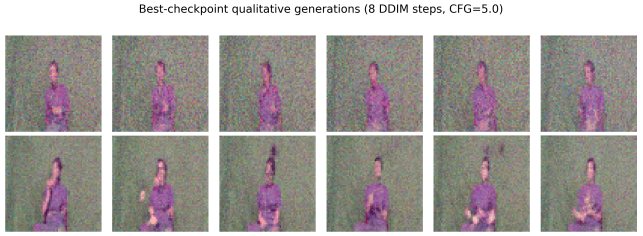

FIGURE 13: Qualitative generations for two prompts under the same 8-step DDIM, CFG= 5.0 setting. Top: "Hello." Bottom: "Thank you." The model changes coarse pose patterns across prompts, but both outputs remain noisy and low-detail.

V. DISCUSSION

A. KEY FINDINGS

The ablation study yields several insights with implications for designing efficient video diffusion architectures, particularly in resource-constrained settings. Fig. 14 provides a holistic multi-metric comparison across all variants, while Fig. 15 summarizes the three principal improvement margins.

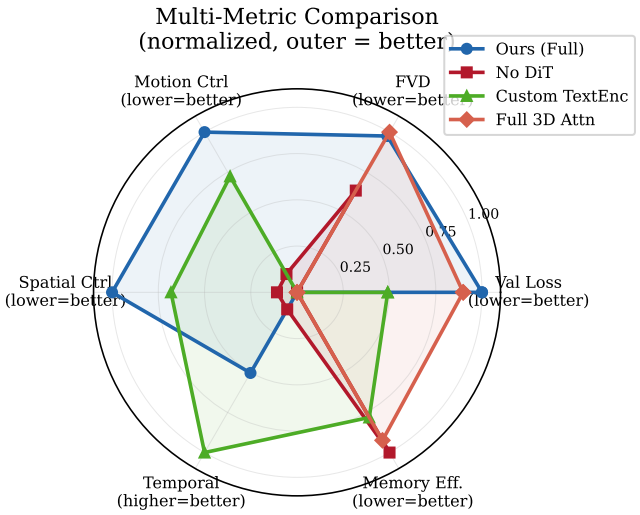

FIGURE 14: Normalized multi-metric radar comparison across all ablation variants. Each axis is independently normalized so that the outer boundary represents the best score. Our full model (blue) achieves the most balanced profile, performing best or near-best on five of six axes. The Custom TextEnc variant trades motion control for temporal consistency, while the No DiT baseline underperforms across all quality metrics.

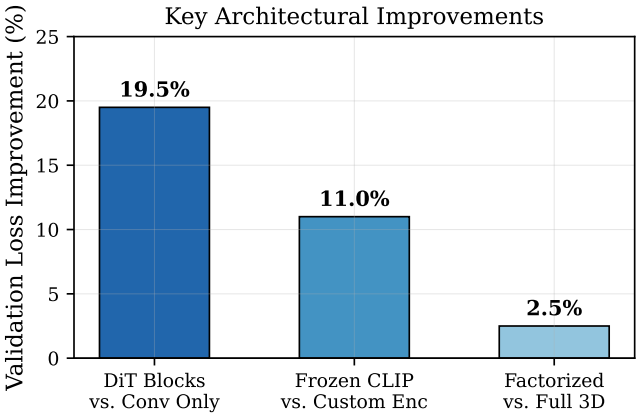

FIGURE 15: Summary of percentage improvements in validation loss achieved by each architectural component: DiT blocks provide the largest gain (19.5%), followed by pretrained text encoding (11.0%), and factorized attention (2.5%).

DiT blocks are essential for sign language fidelity. The 19.5% validation loss improvement (0.0648 vs. 0.0805) confirms that global attention mechanisms capture dependencies that 3D convolutions alone cannot model. Sign language communication involves coordinated, simultaneous movement across multiple articulators — hands, face, torso, and gaze — whose relationships extend beyond local receptive fields. The per-epoch convergence data (Table 3) further show that this advantage is not a transient effect: the No DiT variant exhibits consistently higher loss across all training epochs, indicating a fundamental representational limitation rather than a convergence speed issue.

Pretrained encoders provide superior generalization. Despite 7.7% fewer trainable parameters (528.6M vs. 572.8M), the frozen CLIP encoder achieves 11.0% lower validation loss (0.0648 vs. 0.0728) and 25.6% lower FVD-proxy (7,083.2 vs. 9,515.6) relative to the Custom TextEnc variant. This finding extends the well-established text-to-image observation [6] to the video domain. It suggests that the semantic structure captured by large-scale vision-language pretraining — even when the pretraining data contains no sign language — transfers effectively to the sign language conditioning task. The Custom TextEnc variant's higher first-epoch validation loss (0.1217 vs. 0.1176) also indicates that randomly initialized text encoders actively impede early training dynamics by providing noisy, uninformative conditioning signals.

Because the text encoder is frozen, the model also inherits some zero-shot semantic behavior from the pretrained language–vision space. In practice, this means the system can often respond more sensibly to prompts that are semantically related to, but not identical with, the small set of phrases seen during training. This does not guarantee sign-linguistic correctness, but it is a plausible mechanism for improved behavior on previously unseen prompts compared with a text encoder trained entirely from scratch on the limited sign-language corpus.

Factorized attention is both sufficient and beneficial. The marginal quality difference between factorized (0.0648) and full 3D attention (0.0664)—a 2.5% gap—contrasts with the substantial memory reduction (19.0 GB vs. 9.3 GB for the full 3D variant, though our model's higher memory usage also reflects the frozen CLIP encoder's activations). Crucially, our factorized model achieves *lower* validation loss than full 3D attention, suggesting that the factored inductive bias—first resolving spatial structure per frame, then propagating temporal dynamics—aligns well with the hierarchical nature of sign language, where hand shape (spatial) and motion trajectory (temporal) represent distinct linguistic primitives.

Efficiency–quality trade-offs. The efficiency–quality summary (Table 5) reveals that no single variant dominates all axes. The No DiT baseline is the most resource-efficient but suffers from substantially degraded quality. Full 3D Attention offers the best FVD-proxy at moderate cost, but cannot scale to longer sequences without quadratic memory explosion. Our factorized design strikes the most favorable balance for practical single-GPU experimentation: it achieves the best validation loss while its memory footprint remains within a 24 GB budget.

B. ANALYSIS OF TRAINING DYNAMICS

All four variants exhibit rapid initial convergence, with training loss decreasing from approximately 0.35–0.37 at epoch 1 to below 0.12 by the final epoch. However, the generalization gap—the difference between training and validation loss — varies across variants (Fig. 4). Our full model exhibits the

smallest generalization gap ($0.0916 - 0.0648 = 0.0268$), while the Full 3D Attention variant shows the largest ($0.1112 - 0.0664 = 0.0448$), suggesting that the joint attention parameterization is more prone to overfitting under limited data conditions. This observation has practical implications: factorized attention may be preferable not only for computational efficiency but also for regularization when training on small-to-medium sign language corpora.

C. EXTENDED 100-EPOCH RUN

To complement the short (3-epoch) ablation, we trained the full model for 100 epochs on the same data configuration (batch size 4, 32 frames, 64×64 resolution). The run lasted 76.7 hours (22,898 optimization steps). Table 9 summarizes the key outcomes; Fig. 16 shows the per-epoch loss curves, and Fig. 17 plots the train–validation gap. The best validation loss (0.00578) occurred at epoch 84, and the final validation loss at epoch 100 was 0.00768.

TABLE 9: Summary of the 100-epoch training run (text2sign_20260207_124013).

Metric	Value
Total epochs	100
Total steps	22,898
Wall-clock	76.7 hours
Batch size	4 (grad. accum. 4)
Frames per clip	32 at 64×64
Best val loss	0.00578 (epoch 84)
Final val loss	0.00768 (epoch 100)
Final train loss	0.00748 (epoch 100)

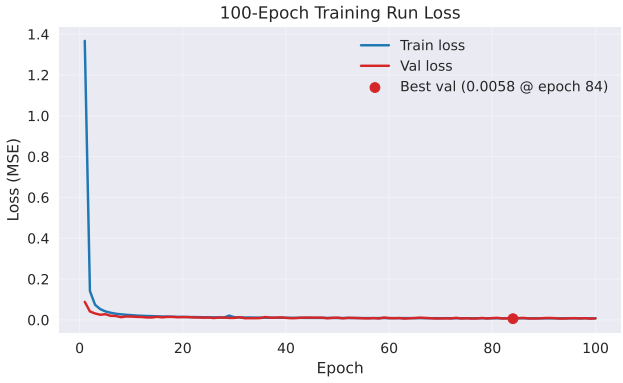

FIGURE 16: Per-epoch train and validation loss for the 100-epoch run. The red marker denotes the best validation epoch (84, with a loss of 0.00578).

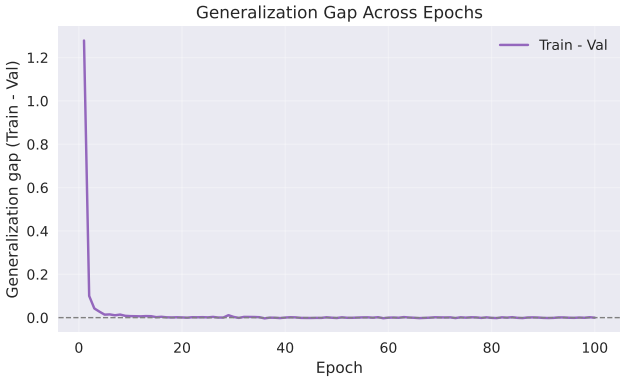

FIGURE 17: Train–validation gap (train−val) over 100 epochs. The gap narrows and stabilizes after roughly epoch 70.

Across the 100 epochs, the training loss fell from 0.141 (epoch 2) to 0.00748 (epoch 100), while the validation loss reached its minimum at epoch 84 and rose modestly to 0.00768 by epoch 100. The generalization gap tightened to within 2×10^{-4} in the final epochs, indicating little overfitting under this configuration. Overall, the long run shows that the architecture remains stable over extended training and that most quality gains accrue within the first 80 epochs, with diminishing returns thereafter.

D. CHECKPOINT EVALUATION AND FINE-TUNING ANALYSIS

To complement the short ablation and long-run training results, the best available signer-disjoint checkpoint was evaluated on a compact validation slice together with two short follow-up fine-tuning studies. The compact evaluation was intentionally limited to a small validation subset so that the reported metrics could be reproduced quickly on the same NVIDIA L4 used for training. Even under this reduced protocol, the checkpoint remained qualitatively weak despite exhibiting substantially lower validation loss than the short ablation models.

The compact evaluation slice produced a mean SSIM of 0.2403 ± 0.0238 and a mean PSNR of 15.11 ± 0.42 dB. These pixel-space values are included only as coarse reconstruction proxies; as discussed earlier, they should not be interpreted as sign-language correctness metrics because signer appearance differs between real and generated clips. More informative for motion stability, the same slice yielded a temporal-consistency score of 1.0000 ± 0.0000 with mean frame-to-frame motion magnitude 0.0987 ± 0.0072. In practice, these numbers indicate that the model is generating very smooth clips, but smoothness alone was insufficient to produce convincing hand articulation or linguistically reliable signing in visual inspection.

The same checkpoint was also benchmarked under an 8-step DDIM setting with classifier-free guidance scale 5.0, which provided the most satisfactory qualitative trade-off

among the sampled configurations. Averaged over five warm-start runs on one NVIDIA L4, this configuration required 12.60 s per 32-frame clip (2.54 frames/s) with 3.12 GB peak inference memory. Although this setting is materially lighter than high-step sampling, it remains far from interactive deployment speed and does not, by itself, resolve the qualitative fidelity limitations.

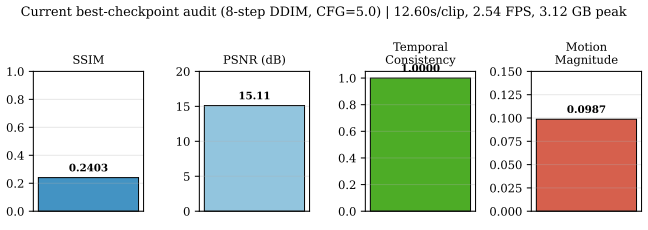

FIGURE 18: Compact evaluation summary for the strongest checkpoint under 8-step DDIM and CFG$= 5.0$. The checkpoint is extremely smooth temporally (temporal consistency $= 1.0000$) and relatively lightweight at inference time, but its low SSIM/PSNR values and the accompanying qualitative samples indicate that smoothness does not translate into convincing signing fidelity.

Two targeted fine-tuning studies were then conducted from the same checkpoint. In the first study, the last two CLIP layers were unfrozen while the learning rate was reduced to 10^{-5} and the training-time sampling frequency increased. This configuration proved unstable: the optimizer state could not be restored cleanly after the trainable parameter set changed, and the earliest generated samples degraded into near-random noise. In the second study, CLIP remained frozen, the restored optimizer state was preserved, the learning rate was reduced to 10^{-5}, and 8-step sample snapshots were recorded more frequently. This second study was numerically stable for the observed steps (e.g., training losses in the 0.0015–0.0092 range over steps 4392–4404), but its qualitative samples remained noisy and did not improve over the original checkpoint. These results suggest that neither modest learning-rate reduction nor partial CLIP unfreezing is sufficient, on its own, to recover substantially better signing quality.

Four prompt-conditioning variants were also evaluated on the same checkpoint: frozen CLIP, no text, random text, and CLIP fine-tuned in the last two layers. Table 10 reports coarse prompt-sensitivity and motion-stability proxies under the same 8-step, CFG$= 5.0$ setting. The no-text control produced the largest same-prompt drift and the highest motion magnitude, indicating noisier and less stable generations when text information was removed entirely. However, the random-text and partially fine-tuned CLIP variants were nearly indistinguishable from the frozen-CLIP baseline on these pixel-space proxies. This result suggests that, for the current checkpoint, text conditioning is not yet exploited strongly enough for simple prompt-shuffling controls to produce a large quantitative separation, a finding that is consistent with the weak qualitative signing fidelity observed in the generated videos.

E. LIMITATIONS AND FUTURE WORK

a: Resolution, computational budget, and visual fidelity.

The combination of 64×64 output resolution, single-GPU training budget, and limited data scale restricts the model's ability to learn the fine-grained articulatory detail required for reliable sign generation, including finger spelling and subtle facial grammar markers. Accordingly, the present study does not achieve consistently reliable high-fidelity signing under the computational regime considered here. Cascaded super-resolution networks, latent diffusion formulations operating on learned codebook representations, and larger-scale training budgets may help address this limitation without proportionally increasing the spatio-temporal attention cost.

b: Temporal length and compositional generation.

Fixed $T = 32$ clips (approximately 1.07 s at 30 fps) still constrain output to individual signs or short phrases. Generating sentence-level or discourse-level signing requires autoregressive decoding with overlap-based blending, sliding-window attention, or hierarchical latent structures that condition fine-grained motion on coarse-grained linguistic plans.

c: Semantic and linguistic evaluation.

Current metrics—video feature distance, motion magnitude, temporal consistency, and our auxiliary back-translation proxy—assess perceptual, distributional, and indirect semantic qualities, but they still do not measure linguistic correctness or intelligibility directly. Future work should incorporate signer-disjoint evaluation, stronger sign-language recognition or translation back-ends, and, where feasible, user studies with Deaf and hard-of-hearing participants or sign-language experts to assess communicative efficacy.

d: Deployment optimization.

The present system is not an edge or real-time method: even the 8-step inference setting still requires 12.60 s for a 32-frame clip on an NVIDIA L4. Achieving real-time inference (sub-100 ms per frame) for interactive applications will require quantization (INT8/FP8), structured pruning of redundant transformer heads, stronger latent compression, and compilation to optimized inference backends (ONNX Runtime, TensorRT). Our modular architecture—featuring a clearly separated text encoder, denoising backbone, and scheduler—is well-suited to such post-training optimizations.

e: Dataset diversity and generalization.

The How2Sign dataset features a limited number of signers performing instructional content. Signer-specific idiosyncrasies in handshape, signing speed, and spatial extent may limit cross-signer generalization. Evaluation on multi-signer corpora (e.g., PHOENIX-2014T, BSL Corpus) and data augmentation strategies (spatial jittering, temporal rate perturbation) would strengthen claims of generalizability.

TABLE 10: Conditioning-control analysis for the strongest checkpoint (8-step DDIM, CFG= 5.0). Cross-prompt distance measures mean MSE between generations from different prompts; same-prompt distance measures mean MSE across repeated generations of the same prompt. These are coarse pixel-space probes rather than linguistic faithfulness metrics.

Variant	Cross-prompt dist.	Same-prompt dist.	Temporal consistency	Motion magnitude
Frozen CLIP	0.01525	0.01560	0.999998	0.09445
No text	**0.02095**	**0.02116**	0.999958	**0.10800**
Random text	0.01526	0.01560	0.999998	0.09462
CLIP fine-tuned (last 2)	0.01527	0.01561	0.999998	0.09449

VI. CONCLUSION

This paper presents Text2Sign, a practical single-GPU diffusion architecture for text-to-sign-language video generation. The short (3-epoch) ablation on a signer-disjoint How2Sign split (4,082 clips, 255 signers) provides indicative, though not definitive, trends: (i) DiT-style transformer blocks reduce validation loss by roughly 19.5% relative to convolution-only baselines (0.0648 vs. 0.0805), suggesting the value of global attention for sign-language structure; (ii) frozen CLIP text encoders yield about 11.0% lower validation loss (0.0648 vs. 0.0728) with 7.7% fewer trainable parameters, indicating that pretrained vision-language priors transfer effectively; and (iii) factorized spatial–temporal attention is marginally better than full 3D attention (0.0648 vs. 0.0664) while offering substantially better theoretical scaling. A longer signer-disjoint checkpoint reaches validation loss 0.00999 and, under a compact validation slice, yields SSIM 0.2403 ± 0.0238, PSNR 15.11 ± 0.42 dB, and temporal consistency 1.0000 ± 0.0000. Under an 8-step DDIM setting, it generates a 32-frame clip in 12.60 s on a single NVIDIA L4. At the same time, the qualitative and conditioning-control analyses indicate that reliable high-fidelity sign generation remains out of reach under the present single-GPU computational budget, output resolution, and data regime. The contribution of this work should therefore be understood as an efficiency-oriented research baseline that clarifies architectural trade-offs under constrained resources rather than as a complete sign-language production system. Together with the fine-tuning analyses, these findings suggest that future progress will likely require architectural, representational, and scaling advances rather than simple continuation of the same training recipe.

ACKNOWLEDGMENT

The author thanks Dr. Khushi Desai for advising on this research. Lightning AI provided computing resources.

REFERENCES

[1] World Health Organization, "Deafness and hearing loss," https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss, 2025, accessed: 2026-01-08.

[2] Bureau of Labor Statistics, U.S. Department of Labor, "Occupational outlook handbook: Interpreters and translators," https://www.bls.gov/ooh/media-and-communication/interpreters-and-translators.htm, 2025, accessed: 2026-01-08.

[3] B. Saunders, N. C. Camgoz, and R. Bowden, "Progressive transformers for end-to-end sign language production," *arXiv preprint arXiv:2004.14874*, 2020, arXiv:2004.14874.

[4] A. Duarte, S. Palaskar, L. Ventura, D. Ghadiyaram, K. Dehghan, F. Metze, J. Torres, and X. Giro-i Nieto, "How2sign: A large-scale multimodal dataset for continuous american sign language," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 2735–2744.

[5] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 6840–6851.

[6] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[7] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, "Video diffusion models," *arXiv preprint arXiv:2204.03458*, 2022, arXiv:2204.03458.

[8] W. Peebles and S. Xie, "Scalable diffusion models with transformers," *arXiv preprint arXiv:2212.09748*, 2023, arXiv:2212.09748.

[9] S.-K. Ko, C. J. Kim, H. Jung, and C. Cho, "Neural sign language translation based on human keypoint estimation," *Applied Sciences*, vol. 9, no. 13, p. 2683, 2019.

[10] W. F. Maia, A. M. Lopes, and S. A. David, "Automatic sign language to text translation using mediapipe and transformer architectures," *Neurocomputing*, vol. 642, p. 130421, 2025.

[11] N. K. Kahlon and W. Singh, "Machine translation from text to sign language: A systematic review," *Universal Access in the Information Society*, vol. 22, pp. 1–35, 2023.

[12] P. A. Rodríguez-Correa, A. Valencia-Arias, O. N. Patiño-Toro, Y. Oblitas Díaz, and R. Teodori De la Puente, "Benefits and development of assistive technologies for deaf people's communication: A systematic review," *Frontiers in Education*, vol. 8, p. 1121597, 2023.

[13] S. B. Abdullahi and K. Chamnongthai, "Idf-sign: Addressing inconsistent depth features for dynamic sign word recognition," *IEEE Access*, vol. 11, pp. 88 511–88 526, 2023.

[14] S. B. Abdullahi, K. Chamnongthai, L. A. Gabralla, and H. Chiroma, "*F*sign-Net: Depth sensor aggregated frame-based fourier network for sign word recognition," *IEEE Sensors Journal*, vol. 24, no. 22, pp. 37 630–37 645, 2024.

[15] S. B. Abdullahi, K. Chamnongthai, V. Bolon-Canedo, and B. Cancela, "Spatial–temporal feature-based end-to-end fourier network for 3d sign language recognition," *Expert Systems with Applications*, vol. 240, p. 123258, 2024.

[16] S. B. Abdullahi and K. Chamnongthai, "American sign language words recognition using spatio-temporal prosodic and angle features: A sequential learning approach," *IEEE Access*, vol. 10, pp. 15 911–15 923, 2022.

[17] S. B. Abdullahi and V. Bolon-Canedo, "Minimizing redundancy in hand dynamic features for enhanced sign language recognition," *Intelligent Data Analysis*, p. 1088467X251367228, 2025.

[18] M. Filhol, M. N. Hadjadj, and B. Testu, "A rule triggering system for automatic text-to-sign translation," *Universal Access in the Information Society*, vol. 15, no. 4, pp. 487–498, 2016.

[19] S. Thangam, V. Muthuswamy, and P. Sarah, "Visualsign: Revolutionizing video accessibility through sign language translation," in *Accessibility and Assistive Technologies*. Springer, 2025, pp. 1–15, accessed: 2025-11-08.

[20] Signapse, "How we built signstream: Rapidly developing accessible video translation software for sign language in just 30 days," https://www.signapse.ai/post/how-we-built-signstream-rapidly-developing-accessible-video-translation-software-for-sign-language-in-just-30-days, 2025, accessed: 2025-11-08.
[21] Bitmovin, "Leveraging ai models to enhance accessibility with sign language in video streams," https://bitmovin.com/blog/ai-sign-language-video-streaming-accessibility/, 2025, accessed: 2025-11-08.
[22] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021.
[23] Y. Wang, X. Chen, X. Ma *et al.*, "Lavie: High-quality video generation with cascaded latent diffusion models," *arXiv preprint arXiv:2309.15103*, 2023, arXiv:2309.15103.
[24] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch *et al.*, "Stable video diffusion: Scaling latent video diffusion models to large datasets," *arXiv preprint arXiv:2311.15127*, 2023, arXiv:2311.15127.
[25] G. Fang, K. Li, X. Ma, and X. Wang, "Tinyfusion: Diffusion transformers learned shallow," *arXiv preprint arXiv:2412.01199*, 2024.
[26] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015, arXiv:1503.02531.
[27] V. Sanh, L. Debut, F. Dernoncourt, R. Louf *et al.*, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2020, arXiv:1910.01108.
[28] W. Ma, X. Yang, L. Jiao, L. Li, X. Liu, F. Liu, P. Chen *et al.*, "Video diffusion generation: comprehensive review and open problems," *Artificial Intelligence Review*, 2025, accessed: 2025-11-08.
[29] Y. Wang, X. Liu, W. Pang, L. Ma, S. Yuan, P. Debevec, and N. Yu, "Survey of video diffusion models: Foundations, implementations, and applications," *arXiv preprint arXiv:2504.16081*, 2025, arXiv:2504.16081.
[30] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," in *International Conference on Learning Representations (ICLR)*, 2016.
[31] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021, pp. 8748–8763.
[32] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *International Conference on Learning Representations (ICLR)*, 2021.
[33] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, vol. 32, pp. 8026–8037, 2019.

RUIZE XIA is an independent researcher with interests in generative modeling, computer vision, and accessible computing. His current work focuses on efficient diffusion architectures for sign language video synthesis.

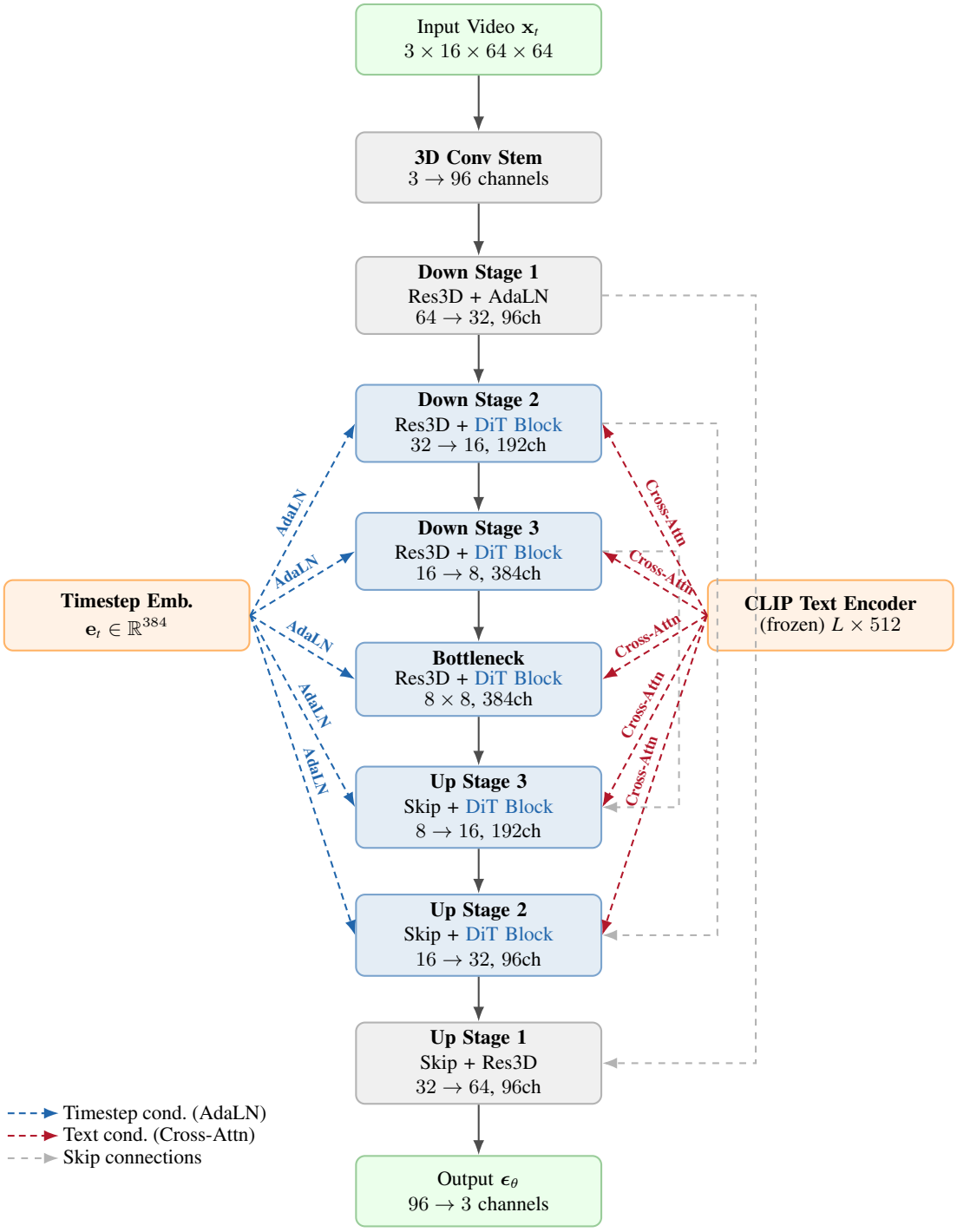

FIGURE 3: Architecture overview of the Text2Sign 3D UNet with DiT-style transformer blocks. The network operates on 3D video tensors through an encoder–decoder structure with skip connections. DiT blocks (blue shading) apply factorized spatial–temporal attention at resolutions 16×16 and 8×8. Blue dashed arrows indicate timestep conditioning via Adaptive Layer Normalization (AdaLN); red dashed arrows indicate text conditioning via cross-attention with frozen CLIP features.