

XR Dance Creativity with AI

Kenny Mitchell
kenny@creativecridhe.co.uk
Creative Cridhe
Edinburgh, UK

Alexandra Schmuklermann
alexandra.schmuklermann@gmail.com
Independent
Cologne, Germany

Noshaba Cheema
ncheema@mpi-inf.mpg.de
Max Planck Institute for Informatics
Saarbrücken, Germany

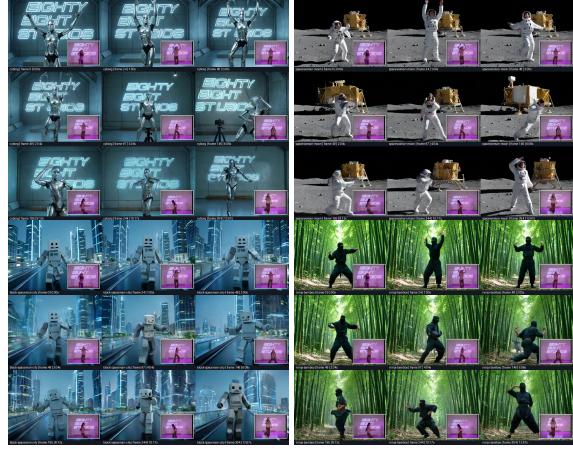


Figure 1: Live projected performance depiction through augmented video mirror display (left) and four recorded-pose grids (right: cyborg, moon, city, ninja). Each grid includes source-camera feed insets.

Abstract

We present a live augmented reality performance system in which human dancers and audience members co-create choreography with generative AI performers in real time. Live dancer motion is captured with a depth camera and conditions a streaming generative video model, transforming performers into stylized characters whose costume, embodiment and surrounding environment can be changed on the fly. Prompts are spoken on stage and transcribed through real-time speech recognition, so dancers can steer the system without leaving the performance. To keep generated output synchronous with the dancer, the system predicts motion and control signals ahead of time rather than reacting after the fact. High-fidelity fantasy key frames, produced through prompted image-to-image generation, anchor the streaming video. This combination pairs the responsiveness of video generation with the visual quality of image generation, enabling demanding live cases such as fantasy costumes, full environment replacement, and transformation of the whole audience. The complete pipeline runs as a self-contained application with a single local GPU, with output projected live into a shared stage space. Because the system is non-deterministic and conditioned on motion captured in the moment, every performance is unrehearsable. The dancers must respond to whatever the AI produces, which makes the piece a genuine exchange between human movement and machine interpretation.

CCS Concepts

• **Human-centered computing** → **Mixed / augmented reality**; • **Applied computing** → **Performing arts**; • **Computing methodologies** → **Motion capture**.

Keywords

dance, generative video, extended reality, motion prediction, GPU interoperability, audience interaction

1 Live Augmented Mirror Video Display

Captured motion conditions on stage image synthesis where dancers respond to the non-deterministic interpretation. Continuously streamed capture, synthesis, and presentation let bodily motion and appearance change at will according to spoken prompts.

We apply a LTX-2.5 based distilled video model with *Union* control [Lightricks 2026b,c] for multi-modal informed live generations. Default audience mode uses RGB camera appearance and a speech to text live prompted scenario guide. Additional tracked motion deriving one dancer’s skeleton from the Kinect depth sensor steers every frame generation with predictive weighted guidance to maintain approximate synchronization of the live dancer with the rendered video frames under marginal latency conditions with the pose signal injected directly into the model’s latent space. Native FLUX.2 Klein 4B provides enhanced generation of intermittent recent camera frames per 49-frame chunk for LTX first-frame conditioning¹. See this linked project page for more results and information.

2 Related Motion Latency Countermeasures

DanceGraph combines low-latency networked pose transport with rhythmic motion correctives for synchronized online dancing [Sinclair et al. 2023]. NeFT-Net learns rhythmic motion prediction using DCT-encoded histories and multi-window attention, and visualizes predicted poses through depth-conditioned diffusion [Ademola

¹FLUX.2 Klein 4B model card; split weights at revision 5f526678002e.

et al. 2025], whilst here we apply a bounded kinematic extrapolation approach to tackle local video generation latency.

3 End-to-End Sub-Second Local Synthesis

Dual-rate conditioning. The performance loop separates frame-synchronous movement from lower frequency appearance steering. Each admitted chunk uses the latest camera image and every frame is supplied with the predicted single-dancer poses. Audience mode treats the crowd as one scene without the single body predicted pose guide. Spoken prompts change the appearance as often as each chunk boundary.

Each chunk undergoes coarse denoising, spatial latent upscaling, full-resolution refinement, and decoding.

Staged synthesis and continuity. The final frame of the prior chunk provides a weighted contribution of the start frame of the next chunk generation. Generation runs continuously concurrently with payout, while pose extrapolation remains bounded by available observations. The text connector depends on the accepted prompt, so its output is reused across denoising steps; time- and frame-dependent work still runs. With steering enabled, the previous generated endpoint and newest edited image condition the next start. The last 2 denoised latent slices overlap to assist carry over of the previous chunk’s motion into the next, giving generation a shared latent-space starting point across chunk boundaries.

Adaptive-guided refinement. For the tracked dancer, segmentation, camera change, and swept pose capsules form an activity map with prior occupancy and a halo. Work is saved by sparsifying later full-resolution refinement passes; low-resolution synthesis and the first full-resolution refinement pass always processes every token. Later evaluations update active queries while combining current active rows with cached inactive context for full-scene attention K/Vs. Periodic dense refresh and missing-data fallbacks limit staleness.

4 Improved Generative Video Synchrony with Pose Predictions

Kinect supplies the real-time skeleton. Tracked mode maps calibrated Kinect joints to an 18-point body guide, following the DW-Pose body rendering convention [Yang et al. 2023]. A small raster pose plus confidence history trained encoder injects into the VAE guide latents, which in turn encodes clean Union conditioning tokens. One surmountable limitation is that one body identity and a bounded pose history deters arbitrary continuous switching between people within the camera view.

Let $\mathbf{x}_t$ be the latest root position and $q_{j,t}$ a joint’s local orientation, separated from the previous observation by Δt . The predictor estimates translation and angular velocity:

$$\mathbf{v}_t = \frac{\mathbf{x}_t - \mathbf{x}_{t-1}}{\Delta t}, \quad \omega_{j,t} = \frac{\log(q_{j,t-1}^{-1} q_{j,t})}{\Delta t}. \quad (1)$$

It then extrapolates using bounded velocities and joint-specific horizons:

$$\hat{\mathbf{x}} = \mathbf{x}_t + \bar{\mathbf{v}}_t h, \quad \hat{q}_j = q_{j,t} \text{Exp}(\bar{\omega}_{j,t} h_j). \quad (2)$$

The root horizon is at most 400 ms, scaled by core-joint confidence; joint horizons also use consecutive joint confidence. Velocity disagreement shortens prediction near reversals, and observations or

Matched adaptive LTX + FLUX initialization: 71.5 GiB

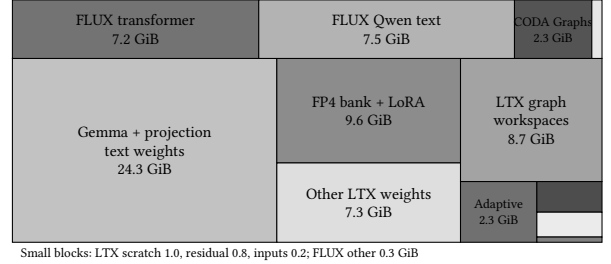


Figure 2: Area-proportional treemap of same-process adaptive LTX initialization followed by FLUX.2 Klein creation. Measured allocations are 54.262 and 17.275 GiB, totaling 71.537 GiB (71.5 rounded). Both Gemma and Qwen use is a limitation of convenience without training either to condition LTX and Flux simultaneously and as such incurs a small redundant time penalty disproportionate to the VRAM overhead.

gaps older than 100 ms disable extrapolation which is rare under stable tracking conditions. Fixed bone offsets preserve the articulated structure. Within observed history, translation and rotation are interpolated for continuous poses between samples, but this is not a smoothing filter: the motion is piecewise linear for translation and piecewise rotational between observations, so velocity can still change abruptly at sample boundaries.

Admission-time trajectories use only observations already available when generation starts. Generation-time p95 informs presentation targets, but extrapolation is clamped to the valid horizon; it cannot recover an unseen second of choreography. The temporal subsystem also supports camera-exposure or physical-presentation targets and a late body warp during the NoGraphicsApi Vulkan presentation stage.

5 Audience Appearance and Spoken Direction

Audience mode conditions a fixed-resolution scene rather than tracking each person. Model invocation resource usage therefore stays constant as the crowd grows.

Local Whisper tiny.en [Radford et al. 2022] transcribes four-second WASAPI audio chunks. Every eight seconds the viewer queues the newest nonempty composed transcript; the worker refreshes both stages’ text conditioning at a chunk boundary without re-initialization or reloading weights.

Flux Klein lower frequency higher quality image generations stay GPU-resident and are also fed by spoken prompts. Warm generations took 402–407 ms; later combined chunks took 1.92–1.93 s versus 2.04 s payout in short live-camera runs. These are throughput figures, not physical response latency.

6 Experiment Hardware and Software

Experiments use one 96 GB RTX PRO 6000 Blackwell GPU. A C++20/CUDA/Vulkan runtime shares the NoGraphicsAPI device and queue with inference through external memory and timeline semaphores. Camera bytes enter pinned host memory; weights, latents, and images stay GPU-resident, while pose metadata stays

Table 1: Same-process initialization allocation mapped in Figure 2.

Resident allocation group	Basis	GiB
Gemma + text projection weights	GGML buffers	24.3
FP4 bank + Union LoRA	native counter	9.6
Other model weights	GGML buffers	7.3
LTX graph workspaces	GGML allocators	8.7
Adaptive context + buckets	native counter	2.3
Native block scratch	native counters	1.0
Inputs + image buffers	buffers + sizes	0.2
LTX unassigned difference	measured residual	0.8
FLUX transformer weights	GGML buffer	7.2
FLUX Qwen text weights	GGML buffer	7.5
FLUX graph workspaces	GGML allocators	2.3
FLUX VAE, inputs, residual	buffers + difference	0.3
Measured combined total	cudaMemGetInfo	71.5

CPU-side. No runtime frames or prompts go to a cloud service. The native engine extends stable-diffusion.cpp and its GGML fork; SDL 3, ImGui, Slang, and cuDNN Frontend 1.30.0 support warp, presentation and inference.² Azure Kinect Sensor SDK 1.4.2 supplies capture, Body Tracking 1.1.2 supplies tracked poses, and whisper.cpp v1.8.3 with ggml-tiny.en.bin supplies local CPU ASR [Radford et al. 2022]; it has no allocation in the GPU VRAM map. CUDA, cuBLAS, cuDNN, and the tracking model are separately distributed vendor components.

Using the GGML fork’s CUDA backend, our tailored inference graphs replays apply fixed-shape work [Gray 2019] persistently resident, and in addition we compute fixed text conditioning once per stage in a separate caption graph. In matched 49-frame tests, this caption-graph optimization reduced chunk time by 12.2%.

The 48-block video transformer uses packed E2M1 FP4 weights with cuBLASLt, BF16 projections with cuBLAS, Union updates $Y = WX + B(AX)$, cuDNN attention, and lifetime-shared scratch. The measured schedule is eight half-resolution steps, $2\times$ latent up-scaling, three refinement steps, and Conv-VAE decoding. A caption graph computes fixed text-connector output once per stage. Tracked refinement uses a 32×24 activity map and captured 25/50/75% query buckets, refreshing densely every fourth chunk; audience camera snapshots are capped at 1 Hz.

LTX-2.5 weights [Lightricks 2026b].³ compose with the separate Union control LoRA inherited from LTX-2.3 [Lightricks 2026a]. Dancer tracking uses dnn_model_2_0_op11.onnx [Microsoft 2026]. The low frequency image generator uses FLUX.2 Klein 4B⁴+qwen_3_4b.safetensors, and flux2-vae.safetensors. Figure 2 and Table 1 report measured GPU initialization allocation with the

higher quality infrequent image generator persisting, excluding host CPU Whisper automated speech recognition (ASR) processing.

7 Recorded Results and Evaluation Boundary

First-frame-conditioned 25-frame warmed synthesis takes 897.078 ms at 512×384 and matched 49-frame caption-cache runs fall from 1527.92 to 1341.04 ms. In the adaptive guide benchmark, two later refinement passes updated 858 of 1,728 video tokens using a 896-query fixed CUDA graph bucket. These timings exclude minimal costs tracking, segmentation, compositing, and presentation.

Chunk time is $L_{\text{chunk}} = t_{\text{ready}} - t_{\text{admit}}$ and end-to-end also includes capture, tracking, admission, queuing, and display. GPU cache prewarming startup time is approximately 24 s.

Acknowledgments

The authors thank the support of Llogari Casas-Cambra, David Sinclair, and Ye Pan.

²Open-source code: stable-diffusion.cpp, leejet GGML, NoGraphicsAPI, SDL, ImGui, Slang, cuDNN Frontend, Kinect Sensor SDK, and whisper.cpp.

³Loaded LTX files (all .safetensors): ltx-2.5-22b-distilled-transformer-nvfp4, ltx-2.5-22b-distilled-transformer-bf16, gemma4-12b-with-proj-ltx-2.5-bf16, ltx-2.5-video-vae-bf16, ltx-2.5-video-vae-conv-bf16, and ltx-2.5-latent-spatial-upscaler-x2-bf16-1.0.

⁴FLUX.2 Klein model card; split weights, revision 5f526678002e.

References

- Adeyemi Ademola, David Sinclair, Babis Koniaris, Samantha Hannah, and Kenny Mitchell. 2025. NeFT-Net: N-window Extended Frequency Transformer for Rhythmic Motion Prediction. *Computers & Graphics* 129, Article 104244 (June 2025), 8 pages. doi:10.1016/j.cag.2025.104244
- Alan Gray. 2019. Getting Started with CUDA Graphs. NVIDIA Technical Blog. <https://developer.nvidia.com/blog/cuda-graphs/>
- Lightricks. 2026a. LTX-2.3-22b-IC-LoRA-Union-Control: Model Weights. Checkpoint ref0.5; revision b4d1c4d8c9e5. <https://huggingface.co/Lightricks/LTX-2.3-22b-IC-LoRA-Union-Control/tree/b4d1c4d8c9e544e9bbbd6811bb4363708b6093ff>
- Lightricks. 2026b. LTX-2.5: Model Card and Checkpoints. Weight release revision 5e6e71018ee1; filenames in the weight manifest. <https://huggingface.co/Lightricks/LTX-2.5/tree/5e6e71018ee1756ed329b697a7b4aedc934dfce9>
- Lightricks. 2026c. Union Control: Structural Control for LTX. <https://docs.ltx.io/open-source-model/feature-guides/structural-control/union-control>
- Microsoft. 2026. Azure Kinect Body Tracking SDK and Model. Package version 1.1.2; dnn_model_2_0_op11.onnx. <https://www.nuget.org/packages/Microsoft.Azure.Kinect.BodyTracking/1.1.2>
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust Speech Recognition via Large-Scale Weak Supervision. arXiv:2212.04356. doi:10.48550/arXiv.2212.04356
- David Sinclair, Adeyemi Ademola, Babis Koniaris, and Kenny Mitchell. 2023. Dance-Graph: A Complementary Architecture for Synchronous Dancing Online. In *36th International Conference on Computer Animation and Social Agents (CASA)*. Limassol, Cyprus, 8 pages. <https://napier-repository.worktribe.com/output/3089818>
- Zhendong Yang, Ailing Zeng, Chun Yuan, and Yu Li. 2023. Effective Whole-body Pose Estimation with Two-stages Distillation. arXiv:2307.15880. doi:10.48550/arXiv.2307.15880