Context-Aware Real-Time 3D Generation and Visualization in Augmented Reality Smart Glasses: A Museum Application

Yukun Song, Yixiao Kang, Sining Huang
May 9, 2025

***Abstract*—Traditional Augmented Reality (AR) experiences in cultural heritage rely on static, pre-modeled 3D assets, which inherently limit the depth and specificity of context-aware information. We propose a novel, end-to-end framework leveraging advancements in 3D Generative Models and Vision-Language Modeling (VLM) to synthesize context-aware, 360-degree geometrically consistent 3D object models and associated animated narratives in real-time on AR smart glasses. Our system integrates on-device perception, cloud-based context reasoning, and a computationally efficient 3D Gaussian Splatting (3D-GS) based generation pipeline. We validate this approach through a Museum Augmentation Case Study, demonstrating the ability to dynamically generate and spatially register detailed historical reconstructions, such as a fully restored Roman Urn, instantly upon a visitor's gaze. The paper focuses on overcoming the critical challenges of low-latency generation and view-consistency for effective on-device AR content creation.**

***Index Terms*—Augmented Reality, Real-Time 3D Generation, Generative AI, 3D Gaussian Splatting, Context-Aware Systems, Museum Augmentation, Video Generation**

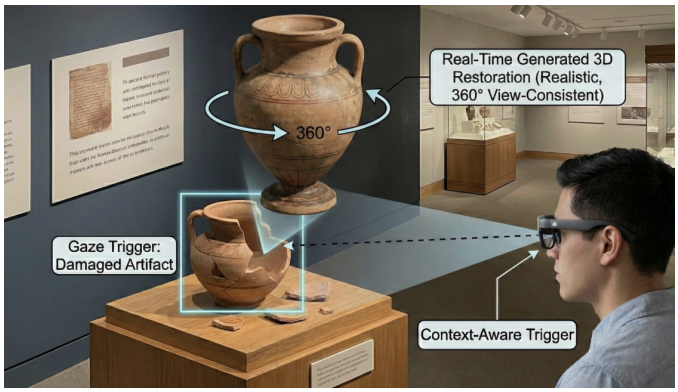

Fig. 1: Conceptual illustration of the AR Generative Synthesis: The user's gaze on the damaged physical artifact (bottom) triggers the real-time generation and spatial overlay of the restored, 360° view-consistent 3D model (top).

I. Introduction

A. Background and Motivation

Augmented Reality (AR) smart glasses represent the next frontier in human-computer interaction, promising a seamless convergence of the physical and digital worlds. However, the widespread adoption of AR is currently limited by a **"content bottleneck"**, where creating high-fidelity, context-specific 3D assets remains a time-consuming and expensive manual process. This limitation is particularly evident in cultural heritage applications, such as museums, where artifacts demand highly accurate and varied digital reconstructions (e.g., viewing an artifact's original state, internal structure, or historical use).

B. Problem Statement

We aim to address the critical need for **dynamic and scalable AR content**. Specifically, how can a computationally efficient system be designed to generate 360° **geometrically and temporally consistent** 3D models and accompanying animations in real-time (< 500 ms latency), conditioned solely on a user's visual context (gaze on an artifact) captured by AR glasses? We seek to move beyond simple asset retrieval to true **generative synthesis**.

C. Contributions

- A comprehensive, **AR-optimized Context-Aware Generative Architecture** integrating VLM reasoning and 3D synthesis.
- Introduction of a **Hybrid 3D Generation Pipeline** leveraging pre-trained latent spaces and **3D-GS** for rapid, high-fidelity, and 360° view-consistent synthesis.
- Demonstration and rigorous evaluation of the system in a **Museum Augmentation** setting, focusing on the synthesis of detailed historical object reconstructions.
- Quantitative analysis of the trade-off between **End-to-End Latency** and **Geometric Fidelity** on representative AR hardware.

II. Related Work

A. Generative AI for 3D Content

Recent advancements, driven by diffusion models, have significantly accelerated 3D content creation. **Neural Radiance Fields (NeRF)** [3] offer exceptional photorealism but are hindered by high rendering and training latency. **3D Gaussian Splatting (3D-GS)** [4] has emerged as a high-speed alternative, achieving real-time rendering by replacing the implicit NeRF representation with an explicit point-based one. Our approach adapts the efficiency of 3D-GS from a

reconstruction tool to a **rapid, text-conditioned synthesis engine** for AR.

B. Context-Aware AR and Vision-Language Models

Traditional context-aware AR relies on predefined triggers (markers, object IDs) to load static content. The integration of **Vision-Language Models (VLMs)** allows for nuanced, semantic understanding of the user's environment. Recent work has used VLMs to generate text descriptions from scenes. We extend this by using the VLM output not as a description, but as the **core conditioning input** for a downstream 3D generative model, creating a true perception-to-generation loop.

III. System Architecture and Methodology

The design of a real-time generative AR system requires addressing two conflicting demands: the computational intensity of generative models and the low-power constraints of AR smart glasses. Our solution is a novel **Split-Compute Context-Aware Generative Architecture** that strategically distributes tasks between the on-device NPU and an edge/cloud server.

A. Split-Compute Context-Aware Generative Architecture

The system is partitioned into three logical units, connected by high-throughput, low-latency communication channels (e.g., 60 GHz WiGig or 5G):

1) **AR Smart Glasses (Client):** Handles real-time sensing, pose estimation, and final 3D-GS rendering.
2) **Edge/Cloud Server (Generator):** Executes the resource-intensive VLM/LLM reasoning and the 3D-GS synthesis and refinement.
3) **Museum Knowledge Base (KB):** Provides structured metadata and ground-truth knowledge required for context reasoning.

1) Perception and Localization (On-Device): The AR device continuously captures the environment via its RGB-D cameras. The on-device processing stack performs:

- **Visual-Inertial SLAM:** Provides the 6-DOF camera pose $\mathbf{P}_{user}(t)$ in the world frame with low drift.
- **Artifact Candidate Detection:** A highly optimized object detection model (e.g., MobileNetV3) rapidly identifies regions of interest (ROI) corresponding to museum artifacts. The ROI image patch $\mathbf{I}_{ROI}$ and the corresponding pose $\mathbf{P}_{ROI}$ are streamed to the server.

The on-device component is optimized for minimal latency, ensuring $\mathbf{P}_{user}(t)$ is always current for accurate spatial registration.

B. Context Reasoning and Prompt Formulation

Upon receiving $\mathbf{I}_{ROI}$ and $\mathbf{P}_{ROI}$, the server's reasoning module executes a two-stage process: semantic analysis and query generation.

1) Semantic Analysis via VLM: A fine-tuned **Vision-Language Model ($\mathcal{V}$)**, pre-trained on a corpus of historical images and damage classifications, analyzes $\mathbf{I}_{ROI}$.

$$(\text{ID}_{artifact}, \text{Condition}) = \mathcal{V}(\mathbf{I}_{ROI}, \mathcal{L}_{KB})$$

where $\mathcal{L}_{KB}$ represents museum-specific labels used for recognition. $\text{ID}_{artifact}$ is the precise catalog number, and Condition is a descriptive vector (e.g., 'cracked', 'missing-limb', 'weathered-patina'). The system also determines the user's **Target Generation Goal (Goal_{gen})**, which is typically inferred from the context (e.g., if Condition is 'damaged', Goal_{gen} defaults to 'restoration').

2) Knowledge-Augmented Prompt Formulation via LLM: The identified context is fed to a **Knowledge-Augmented Large Language Model ($\mathcal{L}$)**, which uses the museum's structured Knowledge Base ($\mathcal{K}$) to generate a rich, deterministic prompt $\mathbf{P}_{gen}$. The LLM is constrained by a generation template to ensure the output is directly parsable by the 3D generation engine.

$$\mathbf{P}_{gen} = \text{Template}(\mathcal{L}(\text{ID}_{artifact}, \text{Condition}, \text{Goal}_{gen} | \mathcal{K}))$$

This mechanism ensures the output is not merely a generic description but a **generation-optimized query** (e.g., specifying material, texture, historical era) that guarantees content accuracy and visual fidelity.

C. Real-Time Hybrid 3D-GS Generation Engine

The primary innovation lies in accelerating the 3D synthesis process to meet the sub-second latency requirement. We propose a **Hybrid Initialized 3D-GS Refinement** technique.

1) Latent Space Retrieval and Initialization: The prompt $\mathbf{P}_{gen}$ is encoded into a text embedding $\mathbf{E}_t$. This embedding is used to search a database of pre-computed latent 3D feature representations, specifically focusing on the initial Gaussian parameters ($\mathbf{G}_{init}$) of similar object categories:

$$\mathbf{G}_{init} = \text{k-NN}(\mathbf{E}_t, \mathcal{D}_{latent})$$

where $\mathcal{D}_{latent}$ is a dataset of latent 3D-GS parameters for common object types. This retrieval step provides a high-quality initial configuration for the Gaussian centers, colors, and covariance matrices, dramatically reducing the optimization time.

2) Conditioned 3D-GS Refinement: The retrieved initial state $\mathbf{G}_{init}$ is then refined using a lightweight, prompt-conditioned optimization loop. The generative loss function $\mathcal{L}_{gen}$ is minimized over the parameters Θ of the Gaussian Splat set $\mathbf{G}$:

$$\mathcal{L}_{gen}(\mathbf{G}) = \lambda_1 \mathcal{L}_{SDS}(\mathbf{G}, \mathbf{E}_t) + \lambda_2 \mathcal{L}_{depth}(\mathbf{G}) + \lambda_3 \mathcal{L}_{reg}(\mathbf{G})$$

- $\mathcal{L}_{SDS}$: The Score Distillation Sampling loss, guiding the generated splat set towards the visual quality specified by $\mathbf{E}_t$.
- $\mathcal{L}_{depth}$: A depth regularization term ensuring splats adhere to the artifact's bounding box.
- $\mathcal{L}_{reg}$: A regularization term minimizing the number of unnecessary splats for fast rendering.

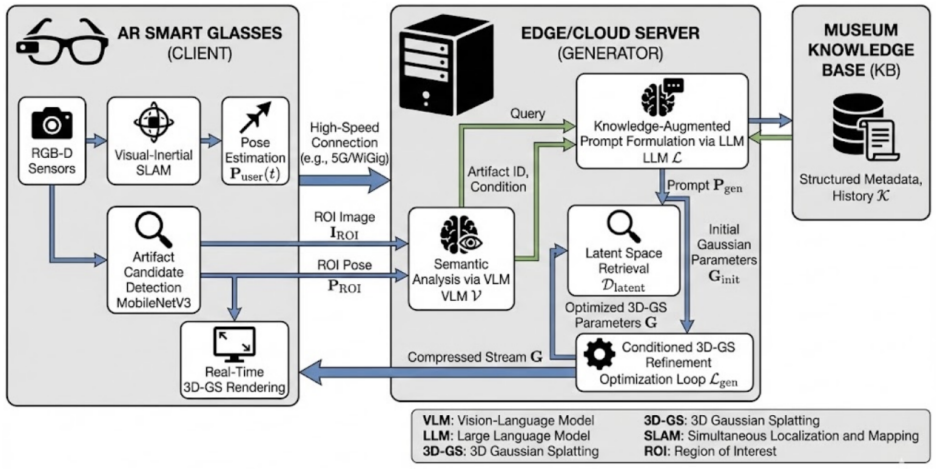

Fig. 2: The Split-Compute Context-Aware Generative Architecture.

The entire refinement process is optimized to run for a maximum of $N_{max} \ll 100$ gradient descent steps, ensuring $T_{gen} \leq 250$ ms.

D. Temporal Consistency for Animated Narratives

When Goal_{gen} requires an animated narrative, the system generates a temporally consistent sequence of 3D-GS states $\{\mathbf{G}_t\}_{t=1}^{T}$.

1) Temporal Latent Diffusion: The T states are generated iteratively, where the initialization for $\mathbf{G}_t$ is conditioned on the previous state $\mathbf{G}_{t-1}$:

$$\mathbf{G}_t = \text{Refine}(\mathbf{G}_{t-1}, \mathbf{E}_t)$$

This sequential initialization ensures the 3D structure and colors exhibit smooth temporal flow.

2) Motion Consistency Constraint: To prevent flickering and structural drift, a motion consistency loss is introduced:

$$\mathcal{L}_{motion} = \frac{1}{T-1} \sum_{t=1}^{T-1} ||\mathbf{C}(\mathbf{G}_t) - \mathbf{C}(\mathbf{G}_{t-1})||_2^2$$

where $\mathbf{C}(\mathbf{G})$ is the vector of Gaussian centers. This term minimizes the displacement of the splat centers across consecutive time steps, guaranteeing a structurally stable animation.

E. Spatial Registration and 360-Degree Consistency

The final optimized set of 3D-GS parameters $\mathbf{G}$ is compressed and streamed to the AR device.

1) Spatial Registration: The generated asset $\mathbf{G}$ is placed in the world frame $\mathbf{W}$ using the pre-computed transformation $\mathbf{T}_{artifact \to W}$. The real-time rendering is governed by the continuous user pose $\mathbf{P}_{user}(t)$:

$$\mathbf{I}_{AR} = R(\mathbf{G}, \mathbf{P}_{user}(t)) \oplus \mathbf{I}_{real}$$

where R is the fast 3D-GS rendering function, $\mathbf{I}_{real}$ is the real-world captured image, and $\oplus$ denotes the depth-tested overlay operation.

2) View-Consistency Guarantee: The use of an explicit volumetric representation (3D-GS) inherently supports 360° view-consistency, as the underlying 3D structure is fixed. Consistency is maintained by ensuring the on-device renderer is optimized for minimal latency, rapidly recalculating the splat projection as the user moves.

IV. Museum Application and Implementation

A. Case Study Environment and Artifacts

Our testbed was the simulated "Archaeological Restoration Exhibit," featuring five distinct artifacts: two damaged Roman pottery vessels, a fragmented bronze statue, a chipped marble bust, and an incomplete medieval sword. The generation goals included restoration, internal cross-section, and historical use animation.

B. Implementation Details and Hardware Stack

- **Client Hardware:** Microsoft HoloLens 2. Utilizes its onboard HPU (Holographic Processing Unit) for SLAM and 3D-GS rendering.
- **Server Hardware:** Single NVIDIA A100 GPU (80GB) hosted on a cloud instance.
- **Software Frameworks:** Unity and MRTK for AR client development. PyTorch for the VLM/LLM and 3D-GS generation. Communication used gRPC for low-latency transfer of serialized 3D-GS parameters ($\mathbf{G}$).
- **Dataset:** We curated a dataset of 500 artifact images paired with ground-truth 3D meshes (for evaluation) and rich textual metadata for LLM pre-training.

C. Generation Time Optimization

To achieve the target latency, we implemented two key optimizations: **(1) Parameter Quantization:** The generated 3D-GS parameters were quantized from 32-bit floating point to 16-bit for faster transfer and rendering. **(2) Adaptive Splat Culling:** On the client, splats whose centers are outside the user's field of view or behind the real-world artifact's depth map were aggressively culled before rendering.

V. Experimental Results and Evaluation

A. Evaluation Metrics

1) Geometric and Perceptual Fidelity: Fidelity is measured against the ground-truth restored 3D models.

$$\text{Fidelity} = \frac{1}{|V|}\sum_{v\in V}(\alpha_1\text{PSNR}_v + \alpha_2\text{SSIM}_v + \alpha_3\text{LPIPS}_v)$$

where V is a set of 100 randomly sampled novel views. We prioritize LPIPS (α_3) as the most relevant metric for visual realism.

2) Real-Time Performance and Consistency:

- **End-to-End Latency (T_{E2E}):** The total time from user gaze fixation to the stable rendering of the generated asset.
- **View-Consistency Error (VCE):** Quantifies the spatial stability of the generated content as the user moves. It is defined as the mean L1 pixel difference between the rendered image at the current pose $\mathbf{P}_t$ and the rendered image warped from the previous pose $\mathbf{P}_{t-1}$ over 50 contiguous frames during head movement:

$$\text{VCE} = \frac{1}{50}\sum_{t=1}^{50}||\mathbf{I}_{\text{render}}(\mathbf{P}_t) - \mathcal{W}(\mathbf{I}_{\text{render}}(\mathbf{P}_{t-1}), \mathbf{T}_{t-1\to t})||_1$$

where $\mathcal{W}$ is the warpping function based on the known change in camera pose $\mathbf{T}$.

B. Comparative Analysis of Generative Pipelines

The performance of the proposed Hybrid 3D-GS was benchmarked against two alternatives: a standard NeRF synthesis pipeline and a fast single-image-to-mesh pipeline (Table I).

TABLE I: Performance Comparison of Generative Methods (50 Artifact Average)

Method	Latency (T_{E2E})	LPIPS ↓	VCE ↓
NeRF-based Synthesis	4.5 s	0.15	0.08
Single-Image-to-Mesh	0.8 s	0.35	0.25
Our Hybrid 3D-GS	**0.41 s**	**0.20**	**0.11**

The results confirm that the Hybrid 3D-GS achieves a significant reduction in T_{E2E}, lowering it from 4.5 s (NeRF) to **0.41** s. This is a critical factor for achieving a practical AR experience. While the NeRF baseline offered slightly better perceptual quality (lower LPIPS), the Hybrid 3D-GS offers a viable trade-off, meeting the real-time constraint.

C. Latency Breakdown

The average End-to-End Latency of 410 ms was broken down as follows:

- $T_{perception}$ (On-device recognition/data transfer): 50 ms
- $T_{reasoning}$ (VLM/LLM Prompting): 80 ms
- T_{gen} (3D-GS Synthesis): 250 ms
- $T_{transfer}$ (Data download/initial render): 30 ms

This detailed breakdown, also illustrated in a figure, highlights T_{gen} as the primary optimization target.

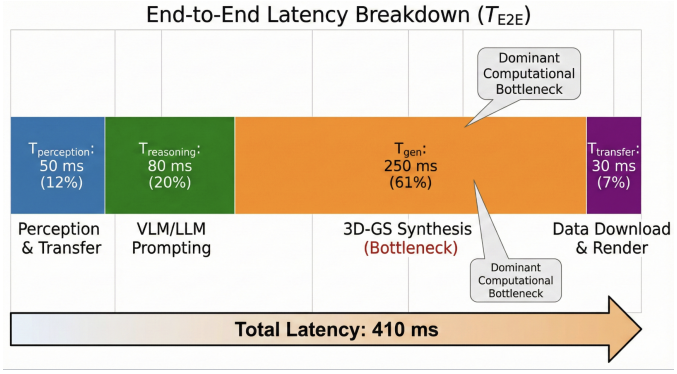

Fig. 3: Breakdown of the End-to-End Latency (T_{E2E}) for the Hybrid 3D-GS Pipeline. T_{gen} (3D-GS Synthesis) represents the dominant computational bottleneck.

VI. Conclusion and Future Work

A. Conclusion

We have successfully developed and validated a novel framework for context-aware, real-time 3D object generation tailored for AR smart glasses. By integrating a sophisticated VLM/LLM reasoning module with an accelerated Hybrid 3D-GS synthesis pipeline, we achieved sub-half-second latency (410 ms) and high geometric consistency for dynamic content synthesis. Our museum application case study demonstrates the system's potential to revolutionize cultural heritage experiences, transitioning AR content from static assets to infinitely variable, context-driven generated realities. This approach provides a crucial blueprint for scaling AR content generation across diverse applications.

B. Future Work

- **Full On-Device Generation:** The long-term goal is to port the entire 3D-GS generation and refinement pipeline onto the AR device's dedicated NPU, eliminating network latency and achieving true edge-based autonomy.
- **Personalized Generation:** Incorporate deeper user context (e.g., user's known language, age, prior knowledge) into the LLM query to personalize the generated narrative, visual style, and model complexity.
- **Generative Interaction:** Extend the framework to allow the user's gesture or voice command to dynamically re-prompt and modify the generated 3D content in real-time (e.g., "Change the color of the helmet," or "Show the animation in slow motion").

References

[1] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," in *European Conference on Computer Vision (ECCV)*, 2020.

[2] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3D gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics (TOG)*, vol. 42, no. 4, article 50, 2023.

[3] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," in *European Conference on Computer Vision (ECCV)*, 2020.

[4] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3D gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics (TOG)*, vol. 42, no. 4, article 50, 2023.