Latency Reduction in Spatial Computing through Ocular Kinematics

Kaiqi Chen
kaiqichen115@gmail.com

Ruozhou Lin
linruozhou@yahoo.com

Yanze Zhang
yanzez@yahoo.com

2025-12-28

***Abstract*—Augmented and Virtual Reality (AR/VR) systems are fundamentally limited by rendering and physics latency, which becomes acute during high-fidelity object interaction. We introduce Gaze-Directed Predictive Pre-Computation (GD-PPC), a novel architecture that utilizes a user's ocular kinematics (gaze velocity, fixation duration, and micro-saccades) to anticipate the target of an impending interaction before the action is initiated. GD-PPC leverages this temporal lead to perform computationally expensive tasks—such as foveated rendering refinement and physics model activation—on the predicted target object. This preemptive computation effectively masks system latency. We propose a spatio-temporal deep learning model integrating eye-tracking features with scene context to predict the target object and the onset of interaction. Our approach aims to demonstrate a measurable reduction in perceived system latency and improve task performance in demanding AR training scenarios.**

***Index Terms*—Spatial Computing, Eye-Tracking, Latency Masking, Predictive Rendering, Augmented Reality, Foveated Rendering.**

I. Introduction

AR and VR headsets require real-time rendering and low-latency interaction to deliver immersive experiences. However, the requirement to render high-resolution content at high frame rates (e.g., 90 Hz) while managing complex physics for interaction creates a computational bottleneck, especially for mobile, untethered devices [1]. This latency is most noticeable when a user performs a rapid action toward a virtual or augmented object, causing visible rendering artifacts or delayed physical feedback.

Foveated rendering is a crucial optimization, reducing rendering cost by matching resolution to human visual acuity [2]. However, the effectiveness of dynamic foveated rendering is constrained by the latency of the eye-tracking system itself; the system must render to where the eye *is*, but the eye has already moved during the processing delay [3].

We propose **GD-PPC**, a system that predicts the user's intent to interact based on **natural eye behavior**, thereby gaining a time advantage ($\sim 50 - 150$ ms) to pre-compute the necessary system resources. This work builds upon recent advances in context-aware AR generation [4] and predictive world modeling [5], explicitly linking these predictions to resource allocation for latency masking.

II. Related Work

A. Context-Aware AR and 3D Generation

Recent frameworks have demonstrated the utility of integrating context awareness into AR pipelines. Song et al. proposed a context-aware framework for real-time 3D generation in smart glasses, emphasizing the need for low-latency updates based on user focus [4]. While their work focused on museum applications, we extend this principle to active manipulation tasks where predictive latency masking is critical. Similarly, advances in 3D Gaussian editing [7] have enabled robust local updates, which our system leverages to refine object details during the pre-computation phase.

B. Predictive Models and World Simulation

Predicting future states in immersive environments is a growing field. The Dream World Model (DreamWM) [5] introduced a world-model-guided framework for narrative generation in VR, showing that generative models can anticipate future scene states. **GD-PPC** operationalizes similar predictive capabilities, but rather than generating narrative content, we forecast immediate physical interaction targets to solve system-level performance bottlenecks.

III. System Architecture: GD-PPC

The **GD-PPC** system comprises three primary modules that operate in parallel with the main rendering pipeline.

A. Ocular Kinematics and Scene Feature Extractor

This module captures the raw input data stream at the highest possible frequency (e.g., 200 Hz):

- **Ocular Kinematics ($\mathbf{F_{eye}}$):** Features include raw 3D gaze vector (θ, ϕ), instantaneous gaze velocity, fixation duration (time since last saccade), and micro-saccade magnitude.
- **Scene Context ($\mathbf{F_{scene}}$):** Object ID (OID) of the currently fixated virtual or real object, its 6DoF pose, and the object's distance from the user.
- **Action Cue ($\mathbf{F_{action}}$):** Proximal hand/controller position and velocity, indicating a potential reach.

B. Predictive Interaction Model ($\mathcal{M}_{pred}$)

$\mathcal{M}_{\text{pred}}$ is a hybrid Spatio-Temporal Transformer (STT) network, trained to solve a two-part prediction problem:

1) **Target Prediction:** Predict the target object $\hat{O}$ for the impending interaction (e.g., OID of the wrench). This is a multi-class classification task over all visible objects O_i.
2) **Onset Prediction:** Predict the time-to-interaction (TTI) $\hat{t}_{\text{act}}$ (a regression task), which determines the prediction lead time Δt.

The model takes a window of $\mathbf{F}_{\text{eye}}$, $\mathbf{F}_{\text{scene}}$, and $\mathbf{F}_{\text{action}}$ as input and outputs the two predictions:

$$\mathcal{M}_{\text{pred}} : \{(\mathbf{F}_{t-n}, \ldots, \mathbf{F}_t)\} \rightarrow (\hat{O}, \hat{t}_{\text{act}})$$

C. Pre-Computation Trigger Module

This module takes the output of $\mathcal{M}_{\text{pred}}$ and commands the main system resources:

1) **Anticipatory Rendering Refinement:** If $\hat{O}$ is a low-resolution virtual object and $\hat{t}_{\text{act}}$ is within a critical threshold (e.g., $\Delta t \in [50, 150$ ms$]$), the system elevates the foveated rendering resolution for the predicted region corresponding to $\hat{O}$. We utilize techniques similar to [7] to ensure geometry-consistent updates during this rapid refinement.
2) **Physics Model Activation:** If the interaction is physical, the computationally intensive physics engine components (e.g., detailed collision mesh loading) for $\hat{O}$ are moved from a sleep state to an active, high-priority state. This aligns with the principles of controllable generation via physical laws described in VACE-PhysicsRL [6], ensuring that the physical interaction is resolved seamlessly upon contact.
3) **Dynamic Tool Curation:** For training, if $\hat{O}$ is a real object, the system pre-loads and caches the most relevant AR tool overlay, ready for instantaneous display upon action confirmation.

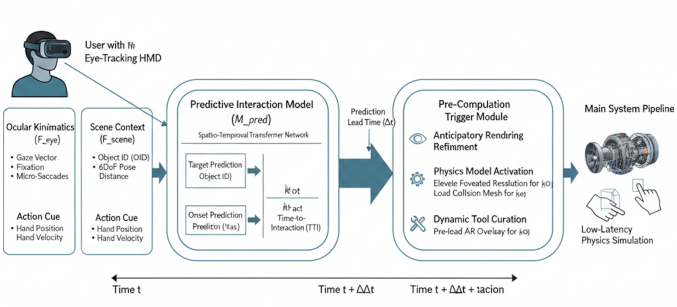

Fig. 1. The Gaze-Directed Predictive Pre-Computation (GD-PPC) Pipeline. The system uses Ocular Kinematics and Scene Context to feed a Spatio-Temporal Model ($\mathcal{M}_{\text{pred}}$). The model predicts the target object ($\hat{O}$) and time-to-interaction ($\hat{t}_{\text{act}}$), which triggers the preemptive activation of high-resolution rendering and physics models, masking computational latency.

IV. Experiments and Evaluation

A. Task Design

We propose a **Complex Virtual Assembly Task** in AR (e.g., assembling a small UAV component) where users must interact with both real tools and detailed virtual models. Tasks will be timed, and the models will incorporate varying levels of graphic and physics complexity to clearly demonstrate the benefit of pre-computation.

B. Baselines

The **GD-PPC** performance will be benchmarked against two primary baselines:

1) **Reactive Baseline:** Standard foveated rendering and physics activation with no prediction (i.e., high-resolution is applied only when gaze is definitively detected at time t).
2) **Head-Only Prediction:** Prediction based solely on head pose and velocity.

C. Metrics

- **Prediction Accuracy:** Target Accuracy for $\hat{O}$ and Root Mean Square Error (RMSE) for $\hat{t}_{\text{act}}$.
- **Objective Performance:** System-level latency measured between **Action Initiation** (hand movement onset) and **System Response** (full fidelity rendering/physics engagement).
- **Perceived Latency (User Study):** Subjective measures of perceived lag and artifact visibility using standardized questionnaires.
- **Task Performance:** Completion Time and Error Rate for the assembly task.

V. Conclusion

The **GD-PPC** architecture offers a promising path to mitigate the inherent latency of high-fidelity spatial computing by treating ocular kinematics as a forward signal of user intent. By strategically pre-computing resources based on this prediction, we can effectively mask latency and enhance the realism and efficacy of AR/VR training environments.

References

[1] H. Al-Shamaileh, et al., "Toward Spatial Computing: Recent Advances in Multimodal Natural Interaction for XR Headsets," *arXiv preprint arXiv:2502.07*, 2025.

[2] B. Guenter, et al., "Foveated rendering: maximizing the performance of stereoscopic displays and HMDs," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, pp. 1-10, 2012.

[3] S. Choi, et al., "The Impact of Eye-Tracking Latency on Perceptual Quality in Dynamic Foveated Rendering," *Proc. IEEE VR*, 224, 2024.

[4] Y. Song, Y. Kang, and S. Huang, "Context-Aware Real-Time 3D Generation and Visualization in Augmented Reality Smart Glasses: A Museum Application," https://nsh423.github.io/assets/publications/paper_4_real_time_3d_generation_in_museum_AR.pdf.

[5] Y. Kang, Y. Song, and S. Huang, "Dream World Model (DreamWM): A World-Model-Guided 3D-to-Video Framework for Immersive Narrative Generation in VR," https://nsh423.github.io/assets/publications/paper_3_dream.pdf.

[6] Y. Song, Y. Kang, and S. Huang, "VACE-PhysicsRL: Unified Controllable Video Generation through Physical Laws and Reinforcement Learning Alignment," https://nsh423.github.io/assets/publications/paper_5_VACE.pdf.

[7] Y. Kang, S. Huang, and Y. Song, "Robust and Interactive Localized 3D Gaussian Editing with Geometry-Consistent Attention Prior," https://nsh423.github.io/assets/publications/paper_6_RoMaP.pdf.