City of Light (COL): A City-Scale, Geo-Anchored Urban Simulator with High-Throughput Multi-Sensor Streams

Ilias Sarbout[1,2], **Mehdi Ounissi**[1], **Théo Cazenave-Coupet**[1], **Dan Milea**[1], **Daniel Racoceanu**[1, 2]

[1]Rothschild Foundation Hospital, Paris, France
[2]Sorbonne University, Paris Brain Institute, Paris, France
ilias.sarbout@gmail.com

Abstract

We present **City of Light (COL)**, a Unity-based, city-scale ($\sim 116\,\mathrm{km}^2$) simulator of Paris for high-throughput embodied AI research. COL fuses open geographic information system (GIS) sources into geo-anchored, per-tile meshes and provides a configurable, stochastic runtime with controllable traffic and pedestrians. Agents receive frame-synchronized multi-sensor observations (RGB, depth, normals, semantics) and execute step-synchronized actions to navigate the environment. To support high-rate vision pipelines, we introduce TURBO, a Unity-Python bridge that streams multi-camera observations and allows control at up to $\sim 1300\,\mathrm{FPS}$, achieving higher throughput than ML-Agents in our benchmark. We also provide a Street View Digital Twin that aligns simulator viewpoints with corresponding real-world panoramas for frame-accurate visual comparison and quantitative matching. COL enables fast scripting, large-scale data collection, and reinforcement learning (RL) in geo-anchored urban settings.

Code — https://github.com/iliassarbout/CityOfLight

Motivation and Contribution

Embodied AI in cities requires (i) geo-anchored, city-scale geometry, (ii) synchronized multi-sensor observations, and (iii) high-speed, developer-friendly tooling for AI (data collection, visualization, control, and training). Current urban virtual environments (e.g., CARLA (Dosovitskiy et al. 2017), MetaUrban (Wu et al. 2025)) address parts of this but typically lack city-scale geometric fidelity. In response, we introduce **COL**, which provides:

- *A geo-faithful urban replica* (inner Paris, $\sim 116\,\mathrm{km}^2$): built from public GIS with Python.
- *Stochastic, dynamic scenarios*: configurable traffic and pedestrian flows.
- *Synchronized sensing*: four image modalities per frame (RGB, depth, normals, semantics).
- *High-throughput AI integration*: up to 1300 FPS rendering (RTX 4090) with Python-side real-time control.
- *Multi-platform and VR support* for human experiments.nu

System Overview

COL is built from geospatial data to support high-throughput, Python-driven interaction in geo-anchored stochastic urban scenes set in Paris. As outlined in Fig. 1, public GIS layers are harmonized and fused into per-tile meshes with rich metadata, yielding a city-scale 3D model suitable for fast rendering. The runtime includes lightweight systems for pedestrians and vehicles, and a compact software stack with Python APIs for control and sensing.

Data Fusion

Buildings, environmental layers, the road network, and street furniture are sourced from OpenStreetMap and French institutions including the Institut national de l'information géographique et forestière and Paris Data. Processing and fusion are performed with a Python workflow that can be refreshed as upstream data are updated. This pipeline emits JSON packages for tiles of $\sim 100{,}000\,\mathrm{m}^2$, with meshes and metadata, ready for direct ingestion in Unity.

Rendering & Tile Baking

Unity readers reconstruct meshes and assign tags/layers from JSON files to build 3D tiles of Paris (objects, meshes, colliders, tags, etc.). We pre-bake tiles into compact prefabs via a custom *editor-time export–restore* pipeline, enabling instantiation of tiles already equipped with the appropriate components (scripts, physics, etc.).

Runtime

A controllable pedestrian agent is equipped with four visual sensors (RGB, depth, normals, semantics) implemented using lightweight replacement shaders. At runtime, we instantiate a grid of 3×3 tiles around the agent. Vehicles follow trajectories sampled on the road network and use raycasts for interaction handling (traffic and crosswalk occupancy). Pedestrians use Unity's built-in system to walk over navigable areas.

Python Interfacing

We introduce TURBO, a lightweight inter-process communication backend that enables high-speed COL control and data collection from Python (Fig. 2). A fixed shared-memory segment contains blocks for initialization, control, and observation retrieval. Image buffers are filled via

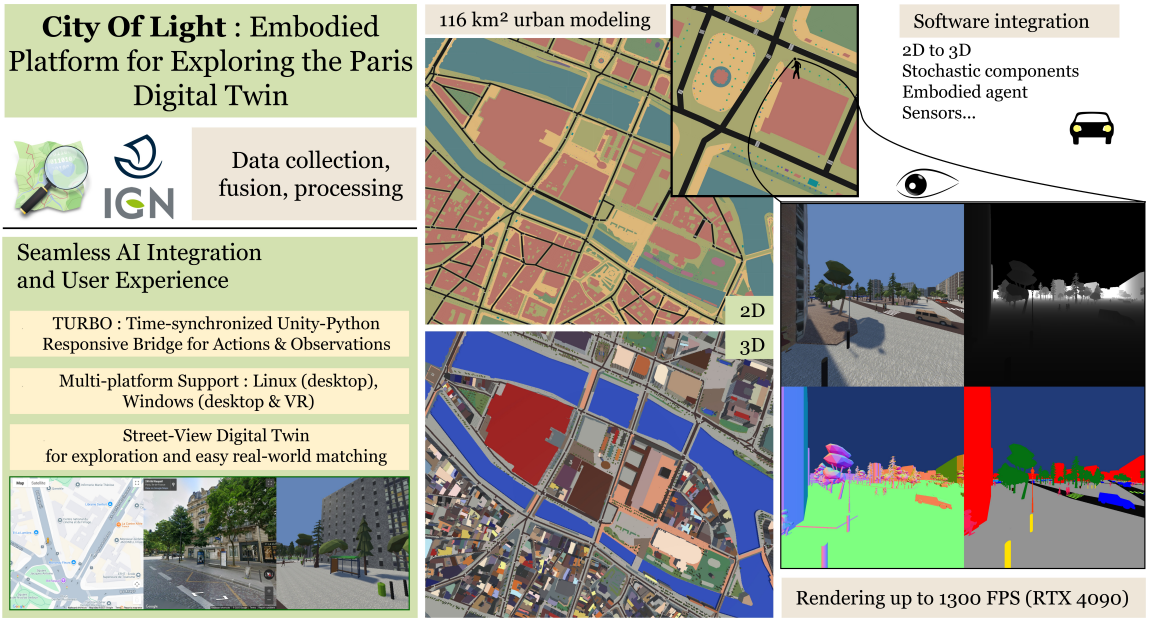

Figure 1: Overview of *City of Light (COL)*. COL builds on harmonized GIS data to generate geo-anchored, tile-aligned meshes of Paris. It provides a configurable runtime with stochastic traffic and pedestrians, streams synchronized multi-sensor views (RGB, depth, normals, semantics), and interfaces seamlessly with AI pipelines via TURBO and a Street View Digital Twin for large-scale embodied AI experiments.

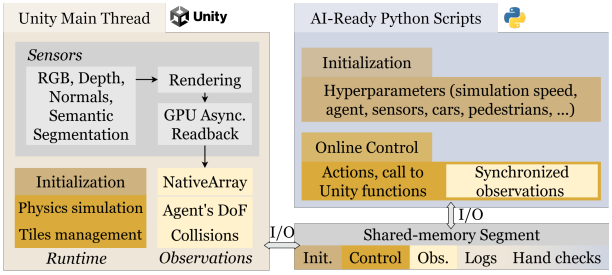

Figure 2: Overview of TURBO. TURBO leverages a shared-memory exchange buffer with blocks for handshake, control, and fixed-stride multi-camera images.

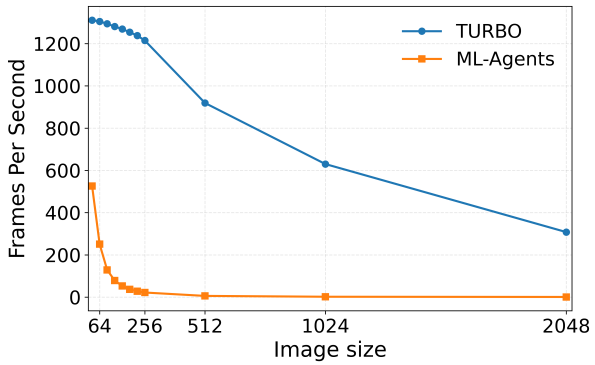

Figure 3: Throughput comparison of TURBO and ML-Agents when streaming RGB observations to Python at increasing image resolutions. Computed over 15,000 frames in a static scene composed of a single COL tile (RTX 4090, AMD Ryzen 7 7700).

asynchronous GPU readback directly into shared memory; Python maps camera blocks as strided NumPy views (no serialization, per-pixel CPU copies, or sockets), sustaining high-frequency closed-loop control with multi-camera observations. TURBO maintains high frame rates across a wide range of image resolutions (Fig. 3), whereas socket-based pipelines (e.g., ML-Agents (Juliani et al. 2018)) exhibit resolution-dependent degradation. This gap arises from additional CPU passes and gRPC protocol overhead in ML-Agents—namely per-pixel float materialization and message serialization/deserialization absent from TURBO. We also introduce a lightweight browser tool that aligns the current COL viewpoint with Google Street View, facilitating matching to real Paris scenes.

Availability

Project releases and documentation are maintained in the project's GitHub repository.

Conclusion

City of Light (COL) provides a geo-anchored, city-scale simulator of Paris with frame-synchronized multi-sensor streams. COL serves as a platform for embodied research, enabling rapid data collection for AI as well as human experimentation in geo-faithful urban environments.

Ethical Statement

COL uses public geospatial data from OpenStreetMap, the French Institut national de l'information géographique et forestière and Paris Data. COL does not contain personal data, and all third-party assets are used under their respective terms. The Street View Digital Twin is not affiliated with or endorsed by Google and requires a user-provided Google Maps Platform API key; usage adheres to Google's terms and quotas. Intended research applications of COL include perception, navigation, training and evaluation of embodied AI models and agents, and sim-to-real transfer.

References

Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; and Koltun, V. 2017. CARLA: An Open Urban Driving Simulator. In *Proceedings of the 1st Annual Conference on Robot Learning (CoRL)*, 1–16.

Juliani, A.; Berges, V.-P.; Vckay, E.; Gao, Y.; Henry, H.; Mattar, M.; and Lange, D. 2018. Unity: A General Platform for Intelligent Agents. *arXiv preprint arXiv:1809.02627*. Unity ML-Agents Toolkit.

Wu, Y.; Li, Z.; Chen, R.; Zhou, X.; Wang, Y.; and Sun, D. 2025. MetaUrban: Scalable Geo-Anchored Urban Environments for Embodied AI. In *International Conference on Learning Representations (ICLR)*. To appear.