UrbanEye: A New Lens on Street-Level Urban Data

Mustahsin-Ul Aziz
West Virginia University
mustahsinaziz@gmail.com

May 14, 2025

1 Introduction

As cities expand at an unprecedented pace, the challenge of designing healthier, more equitable, and environmentally sustainable urban spaces has never been more pressing. Effective planning depends on accurate, detailed, and up-to-date data. Yet one of the most persistent gaps in urban analytics lies at ground level: we lack consistent tools to measure how streetscapes actually appear and function in the eyes of everyday residents. While remote sensing has transformed our understanding of urban form from above, it offers little insight into how greenery, infrastructure, or public life is experienced from the sidewalk.

UrbanEye is an alpha-stage, open-source Python toolkit developed to fill this gap. Using a combination of computer vision, machine learning, and AI, the toolkit analyzes Google Street View imagery to extract high-resolution, structured data about the visible urban environment and infrastructure. Its flexible, low-cost architecture enables researchers and policymakers to capture what people truly see as they navigate city streets, bringing new visibility to the lived experience of urban spaces.

2 How UrbanEye Works

UrbanEye functions through a three-stage pipeline designed for flexibility and replication across geographic contexts. The process begins with a single required input: a city-level shapefile containing the street network. From this, the algorithm generates sample points at regular intervals along the streets, distances that users can customize depending on the density of observation desired. In a demonstration using Pittsburgh, Pennsylvania, points were generated every 500 meters across the street network, which are visualized in Figure 1.

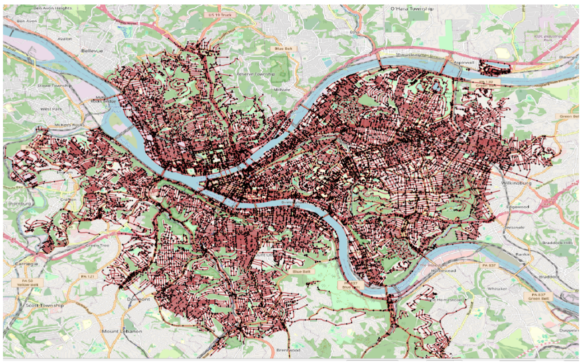

Figure 1: Points Mapping PIT

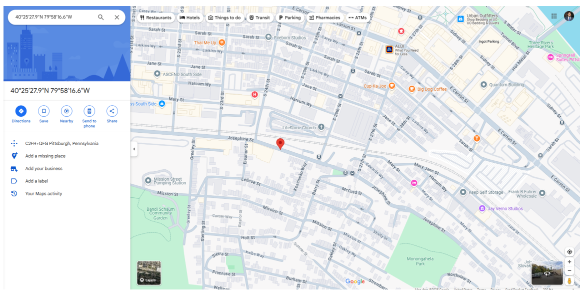

Figure 2: A Specific point in PIT

Once these points are established, the toolkit connects to the Google Street View API to retrieve metadata for each location, including panoramic IDs, timestamps, and location coordinates. For each point, the algorithm collects six directional images: at 0, 60, 120, 180, 240, and 300 degrees, capturing a panoramic snapshot of the street environment. One such location is highlighted in Figure 2 to illustrate this process.

The core of UrbanEye lies in its image analysis capabilities, which constitute the third stage of the workflow. The system employs a hybrid approach combining classical image segmentation (such as Otsu thresholding) with deep learning–based object detection using the YOLOv11 architecture. This AI-powered engine extracts structured data from the visual inputs, identifying green pixels to quantify street-level vegetation and detecting urban features such as pedestrians, bicycles, dogs, and vehicles. Figure 3 presents what a pedestrian would visually perceive, a standard panoramic image. While Figure 4 illustrates what UrbanEye interprets algorithmically: green pixels segmented out and object tags overlaid.

Users can choose how they want to aggregate this data, whether by calculating total greenspace exposure in a city or counting specific objects like cars across street segments. The resulting outputs can be saved in spatially referenced databases or as flat CSV files, ready for visualization or analysis in any GIS or data analysis platform.

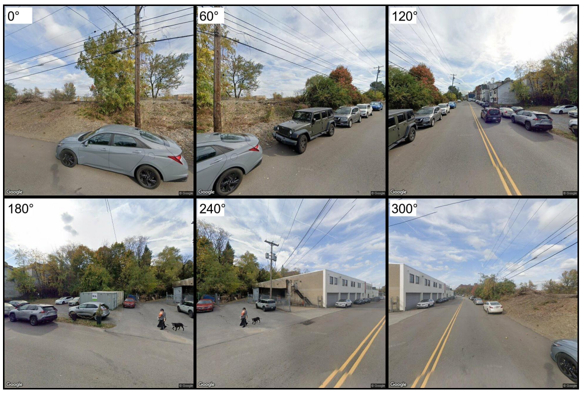

Figure 3: Natural GSV image

Figure 4: UrbanEye detection overlay

Importantly, this is an alpha version of the algorithm. While functional, it remains in development, with ongoing refinements aimed at enhancing precision, broadening detection capabilities, and minimizing classification errors. As it stands, UrbanEye represents a promising proof of concept that already enables a wide range of urban analyses, even as it continues to evolve.

3 Implications

UrbanEye represents a significant methodological advance in urban analytics. It enables fine-grained urban observation of previously unmeasurable features, such as the visibility of street trees or the presence of public activity. Compared to field surveys or in-person audits, UrbanEye provides a scalable and cost-effective way to monitor changes in the natural and built environment. By using AI and machine learning to translate unstructured visual data into actionable indicators, the toolkit can be applied in green equity audits, walkability analysis, and public infrastructure monitoring, among other things.

UrbanEye could be particularly transformative for rapidly growing cities in the Global South, such as Dhaka, Bangladesh. These urban areas often face limited resources for data collection, inconsistent spatial coverage, and rapid informal expansion that escapes official mapping efforts. By using freely available imagery and automated machine learning methods, UrbanEye can provide local planners and researchers with a way to monitor environmental quality, identify underserved neighborhoods, and prioritize interventions based on real visual conditions. In cities like Dhaka, where the disparity in public amenities is stark and urban planning capacity is stretched thin, such a tool could offer vital, low-barrier insights for more equitable development.

4 Further Work

UrbanEye is just the beginning. Several key areas are already identified for future enhancement. Expanding the model to assess additional urban environmental indicators, such as water quality, surface flooding, or air pollution proxies, will enrich its utility. While the current version uses a pre-trained YOLOv11 model, future iterations aim to fine-tune and train models to detect hyper-local features like potholes, illegal dumping, graffiti, signage, or structural decay. These detections could help cities pinpoint zones requiring maintenance or increased investment.

Integrating time-series capabilities to detect change over time, whether before and after infrastructure projects, natural disasters, or regulatory interventions, is another direction for development. Additionally, creating a plugin-based architecture would allow users to integrate UrbanEye with other spatial data platforms, city dashboards, or mobile field apps.

5 Conclusion

UrbanEye demonstrates how AI and machine learning can democratize urban intelligence by bringing the richness of street-level visual information into analytical workflows. Its open-source nature and modular architecture make it accessible to urban researchers, city planners, and environmental analysts alike. While still an alpha version, it already enables new questions to be asked and new patterns to be seen, making visible the invisible aspects of urban life.

By better seeing the street, UrbanEye helps us better understand and ultimately, better shape the city.