WashU

Vision Language Models for Urban Health Assessment

2025 Digital Transformation Summer Corps

Collaborators

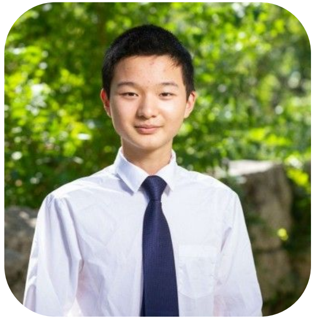
PIs: Tammy English, Rodrigo Reis, Nathan Jacobs

DI2 Engineer: Adith Boloor

Faculty Mentor: Doug Shook

Graduate Student: Eric Xing

Student Engineers: Dev Gupta, Ahmad Hamzeh, Sophia Raudez

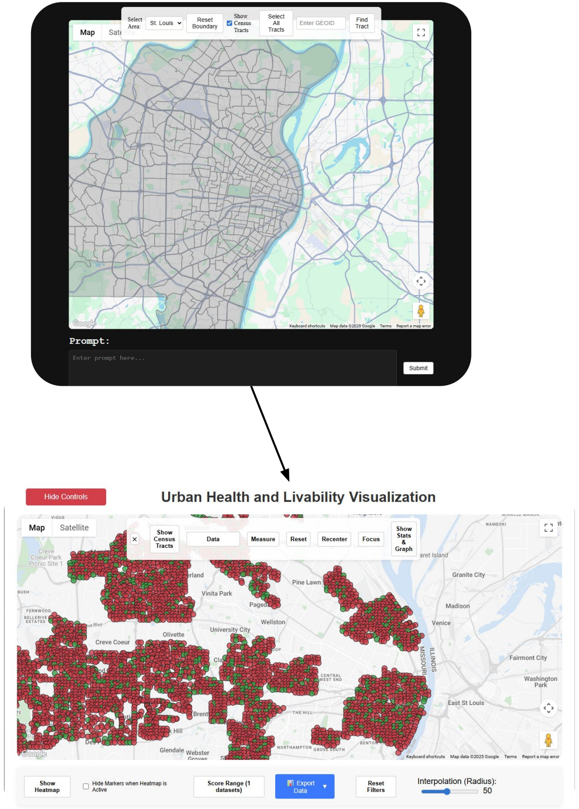


Problem Statement

- Public health assessment has entry barriers for researchers
    - Street view imagery (SVI) gives access to locations at our fingertips
    - Deep learning gives us feature extraction at greater efficiency (compared to human labor)
    - Inference jobs can be automated with little effort beyond setup

Overview

- Automate traditionally manual and resources-intensive process - sending out public health officials to evaluate environments
- Create an interface that will help public health researchers ask questions about given areas

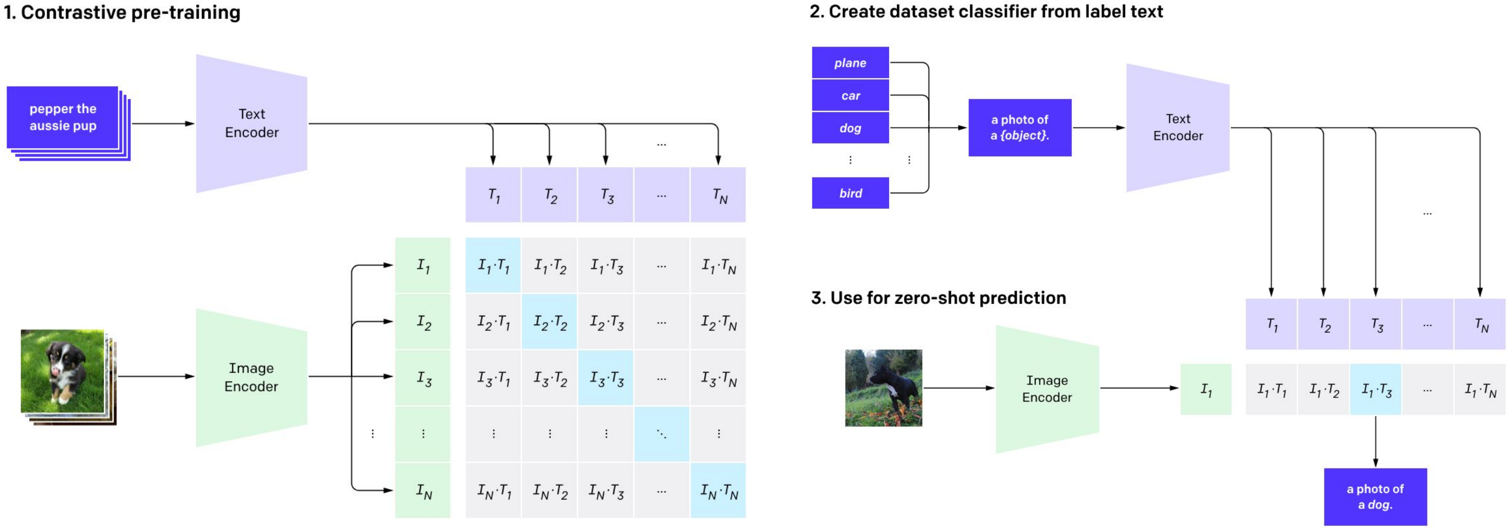


Broader Impact

- Global Data → Global Scaling:
  - Public health officers
  - Government officials
  - University researchers
- Indirect users:
  - Real estate
  - Local businesses
  - Urban planners
  - Safety officers

Background

- Vision Language Model (VLM)
  - Unifies vision and text domains



1. Contrastive pre-training

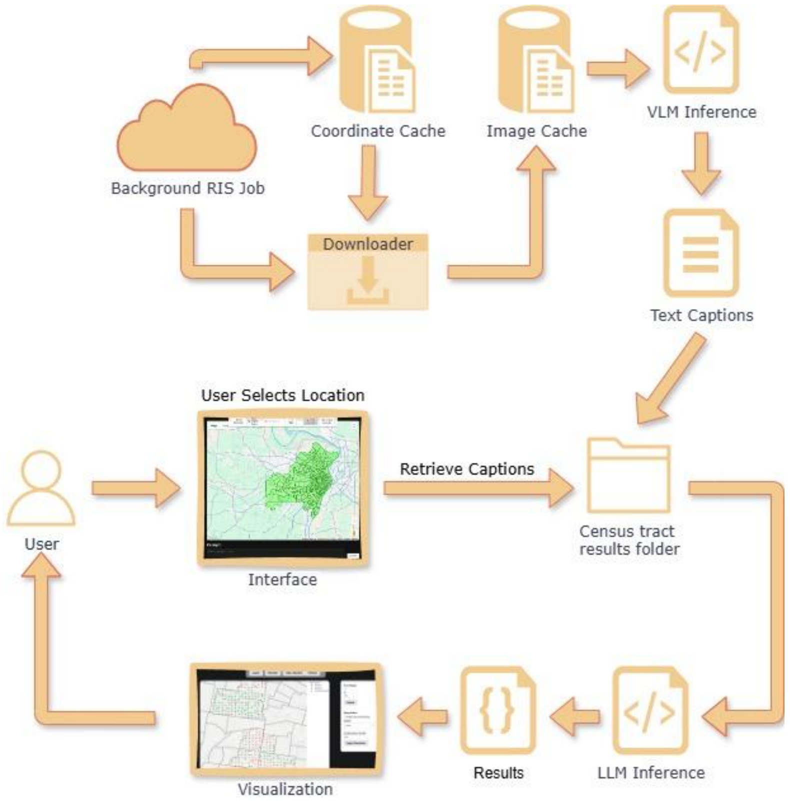
2. Create dataset classifier from label text

3. Use for zero-shot prediction

pepper the aussie pup
Text Encoder
Image Encoder
T_1 T_2 T_3 ... T_N
I_1 I_2 I_3 ⋮ I_N
$I_1{\cdot}T_1$ $I_1{\cdot}T_2$ $I_1{\cdot}T_3$ ... $I_1{\cdot}T_N$
$I_2{\cdot}T_1$ $I_2{\cdot}T_2$ $I_2{\cdot}T_3$ ... $I_2{\cdot}T_N$
$I_3{\cdot}T_1$ $I_3{\cdot}T_2$ $I_3{\cdot}T_3$ ... $I_3{\cdot}T_N$
⋮ ⋮ ⋮ ⋱ ⋮
$I_N{\cdot}T_1$ $I_N{\cdot}T_2$ $I_N{\cdot}T_3$ ... $I_N{\cdot}T_N$
plane
car
dog
⋮
bird
a photo of a {object}.
Text Encoder
T_1 T_2 T_3 ... T_N
Image Encoder
I_1
$I_1{\cdot}T_1$ $I_1{\cdot}T_2$ $I_1{\cdot}T_3$ ... $I_1{\cdot}T_N$
a photo of a dog.

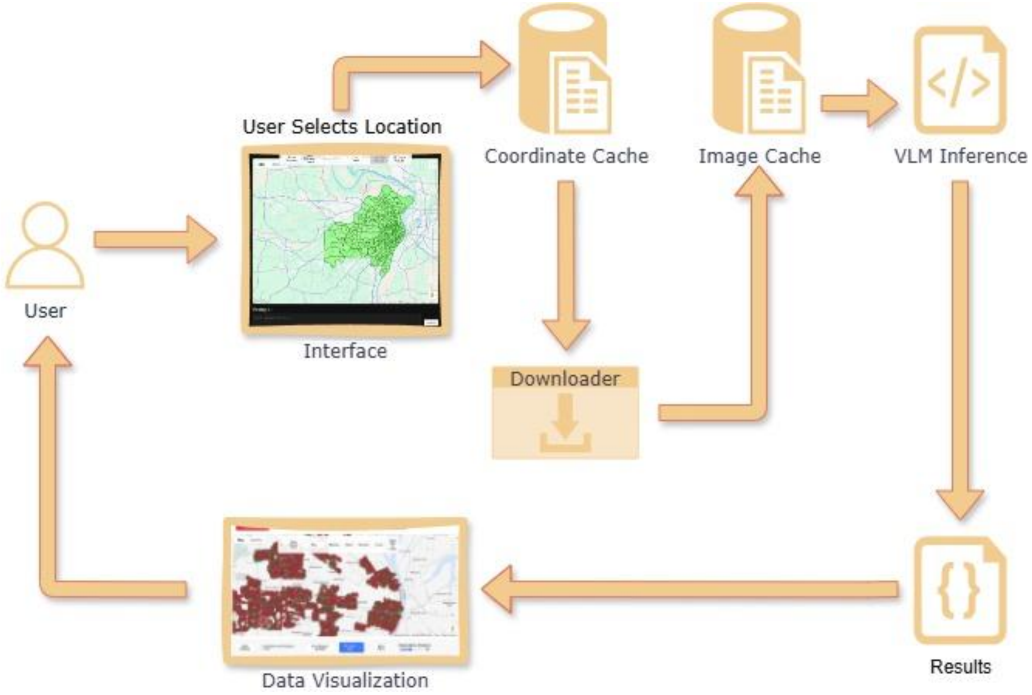
Initial Approach

- Generate large text captions using background job
- Use those at inference time for higher performance
- Useful in the future for features like keyword search and RAG
- Models: InternVL-38B (VLM), Phi4-mini (LLM)
- Plotly visualizing tool



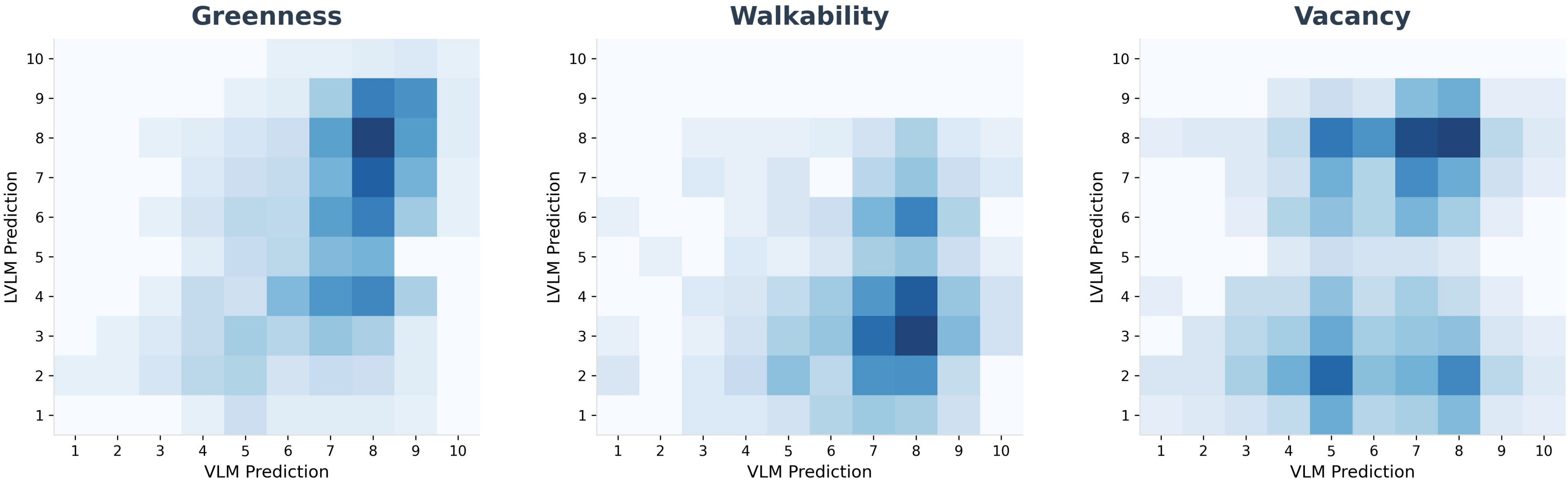
Current Approach

- Use smaller VLM for direct image input at runtime
- Models: InternVL-2B (VLM)
- Google maps API w/ React app

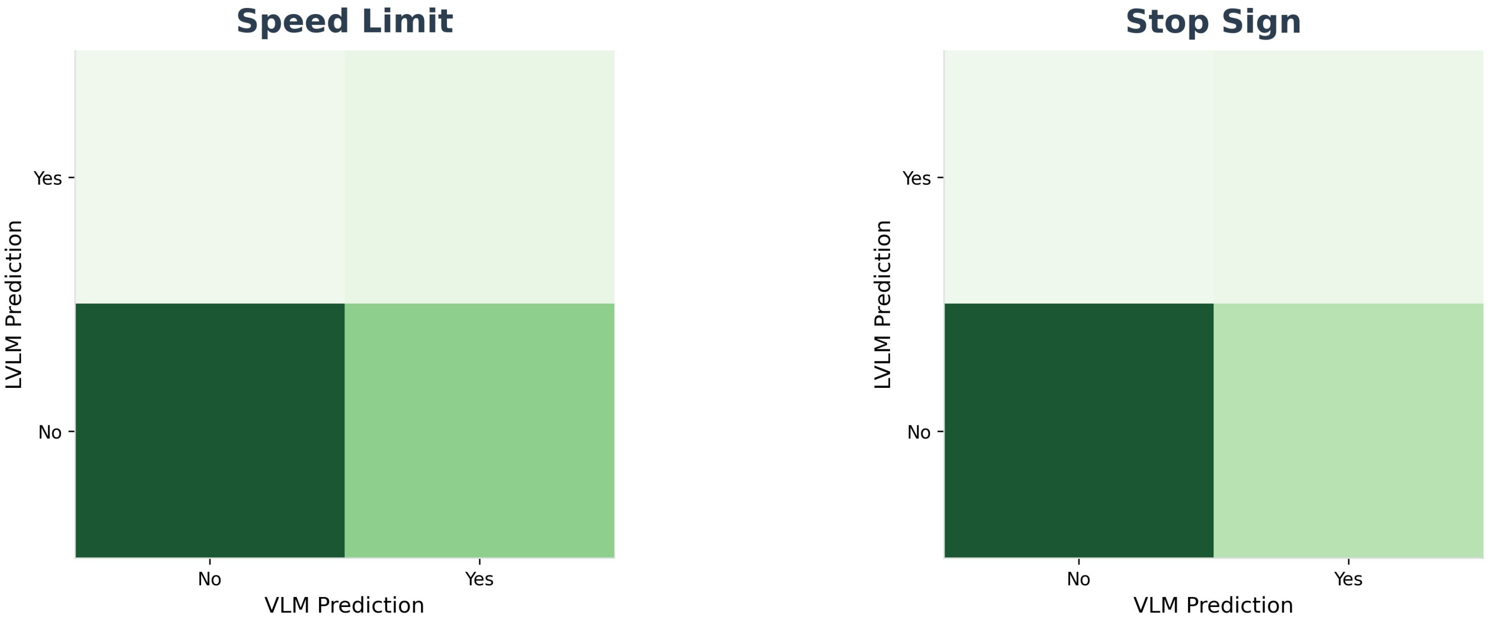


Live Demo

VLM Evaluation (LVLM-as-a-judge)

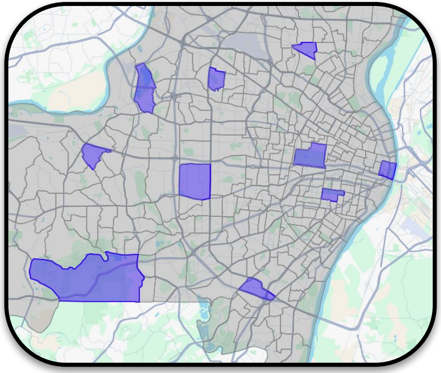


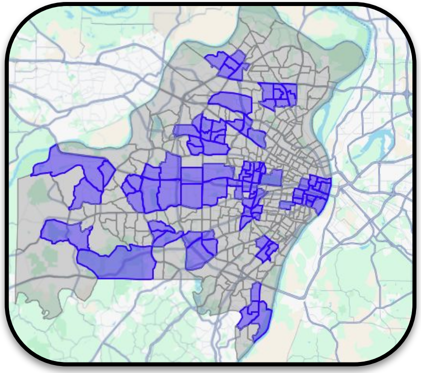
VLM Evaluation (LVLM-as-a-judge)



Performance on 10/100 Census Tracts

Benchmark	Time* (s)	Time** (s)
Green Score	104.0	802.38
Walkability	104.0	912.17
Inhabitance	108.7	1275.5
Speed Limit Signs	103.98	765.53
Stop Signs	100.27	799.77



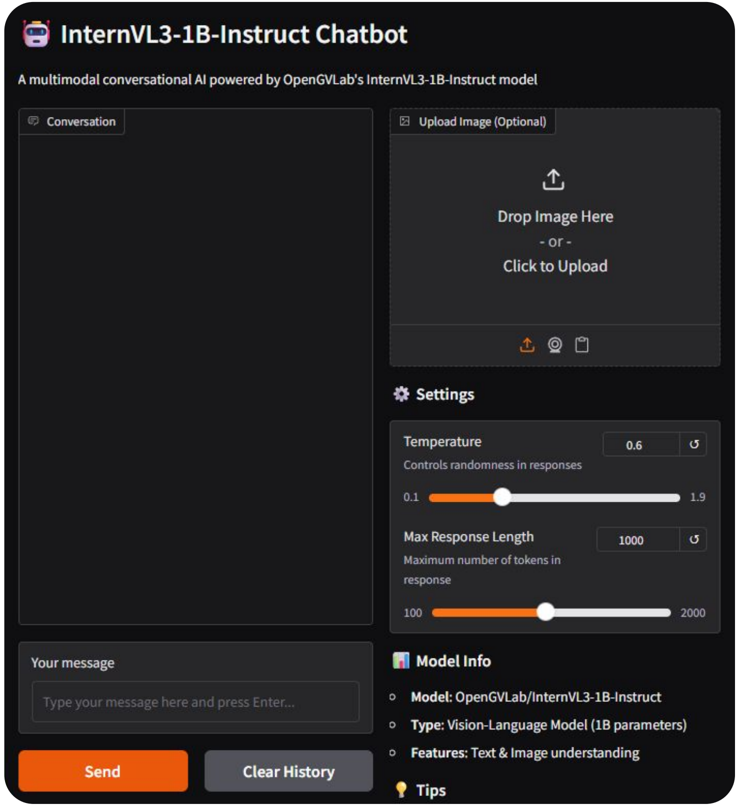


* Based on 1088 locations - Average: 0.120 seconds per location

** Based on 7906 locations - Average: 0.144 seconds per location

Next Steps

- Integrate RAG-based conversational agent which retrieves relevant visual information
- Features:
    - Saves compute time by precomputing visual embeddings
    - Saves inference time by retrieving only relevant information
    - Allows for conversational UI with context history for ease-of-use
- Apply street-level imagery to assess livability in LMICs (via Global Incubator Seed Grant)



InternVL3-1B-Instruct Chatbot

A multimodal conversational AI powered by OpenGVLab's InternVL3-1B-Instruct model

Settings

Temperature: 0.6
Controls randomness in responses
0.1 – 1.9

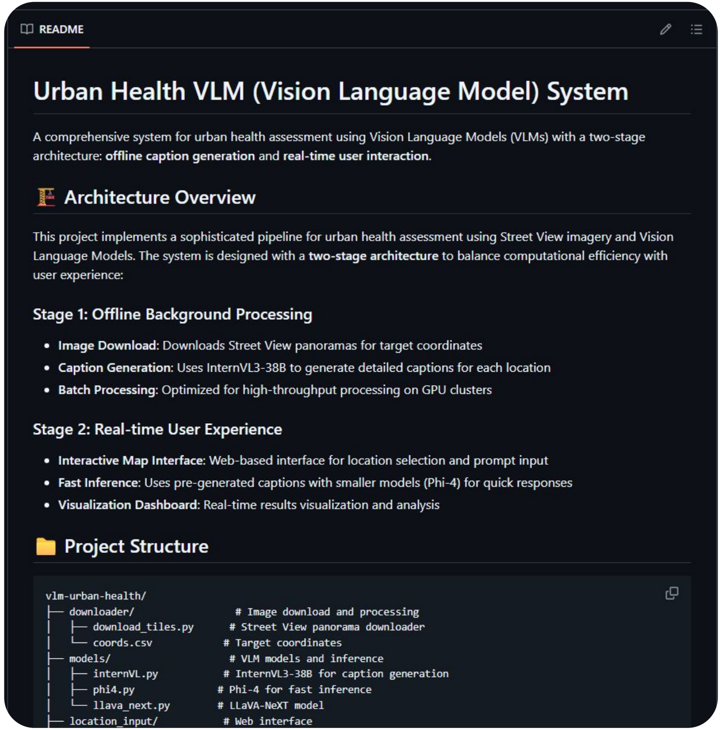
Max Response Length: 1000
Maximum number of tokens in response
100 – 2000

Model Info

- Model: OpenGVLab/InternVL3-1B-Instruct
- Type: Vision-Language Model (1B parameters)
- Features: Text & Image understanding

WashU

Repository



Urban Health VLM (Vision Language Model) System

A comprehensive system for urban health assessment using Vision Language Models (VLMs) with a two-stage architecture: **offline caption generation** and **real-time user interaction**.

🏗️ Architecture Overview

This project implements a sophisticated pipeline for urban health assessment using Street View imagery and Vision Language Models. The system is designed with a **two-stage architecture** to balance computational efficiency with user experience:

Stage 1: Offline Background Processing

- **Image Download**: Downloads Street View panoramas for target coordinates
- **Caption Generation**: Uses InternVL3-38B to generate detailed captions for each location
- **Batch Processing**: Optimized for high-throughput processing on GPU clusters

Stage 2: Real-time User Experience

- **Interactive Map Interface**: Web-based interface for location selection and prompt input
- **Fast Inference**: Uses pre-generated captions with smaller models (Phi-4) for quick responses
- **Visualization Dashboard**: Real-time results visualization and analysis

📁 Project Structure

```
vlm-urban-health/
├── downloader/                # Image download and processing
│   ├── download_tiles.py     # Street View panorama downloader
│   └── coords.csv           # Target coordinates
├── models/                   # VLM models and inference
│   ├── internVL.py          # InternVL3-38B for caption generation
│   ├── phi4.py             # Phi-4 for fast inference
│   └── llava_next.py       # LLaVA-NeXT model
├── location_input/          # Web interface
```

https://github.com/washu-dev/vlm-urban-health

Questions?