

Long-Term Item Resolution in AR Scanning

Background

One of the biggest technical obstacles Linkt has tackled, is implementing a long-term item resolution system, to augment a YoloV8 object detection model. To provide some useful context, a current project we are working on involves deploying YoloV8 to a web browser environment where we have implemented a custom object tracking algorithm. As an abstract description, the application let's users "Add" items to an inventory by tapping on the bounding boxes drawn by the YoloV8 model.

However, the project requirements stipulate that the application needs to have long term memory. That is to say that if YoloV8 detects an object, and the user adds it to their inventory, then the application is expected to recognize that specific item and identify it as "ADDED" every time that it reappears on the video stream for the remainder of the session. Therefore the application must have some sort of "memory" which requires it to be able to resolve the inventory status of an item, even if the item hasn't been seen by the application in several minutes.

Assignment

Your task is to design the general architecture of a long-term item resolution component to the scanning application. You will be expected to do some initial research on appropriate technologies, models, or frameworks that you believe may be helpful. In addition to research, you should provide a notebook(s) with simple demonstrations of any important components of your proposed approach (e.g. if you are proposing we implement another model, write a notebook explaining the general outline of the model's training process and implementation guidelines). Your code need not be complete, it can use mock values and simply demonstrate various concepts you are proposing. However, it is important that your work is clearly explained and readable.

Additional Context

Here is some additional context on the larger application as a whole. You should use this information to inspire and guide your design decisions:

- The client side app is built with Next JS.
- The YOLOv8 Object Detection model is deployed in an [ONNX runtime environment](#) for the browser.
 - You may assume that the object detection model is running inference N frames per second of the video stream. In your design evaluation, you should

analyze how a change in N will affect the feasibility and performance of your proposed solution.

- For each frame the model processes, you are able to retrieve the cropped pixels of each bounding box that is predicted by the model
- You have the freedom to implement your solution entirely on the client side or propose a remote server to help with the task. If you suggest a remote server (or any component that will require utilization of a network protocol) you should be sure to outline the communication protocol between client and server.

Deliverables

Your response to this assignment should consist of two main components.

Research Report

First, you should write a brief research report (approx 2 pages). Your report should include any relevant notes from your research process, relevant links to the sources, articles, papers, and documentation you referred to throughout your research phase, and an outline of your proposed solution.

Colab Notebook

Once you have designed your solution, you should develop a Google Colab Notebook, which highlights any important technical aspects of your solution. I am leaving the structure of the notebook to be up to you. My general approach would be to have a few main sections that show some mock code of any important components of the solution. If you want to organize your code snippets into multiple notebooks for organizational purposes, that is perfectly fine as well. Most importantly is that your notebooks should be able to be used to communicate your thought process. You should include markdown blocks which evaluate the pros and cons of each component you are demonstrating. I should be able to follow your thought process and understand the justification for your design choices.