Retraining YOLO

This step involves delving deep into the working of our base algorithm, Yolo. A brief introduction along with some of the goals we seek to accomplish are:

- **YOLO (You Only Look Once) is an image detection algorithm that uses the “darknet” framework.**

- Uses a neural network and machine learning to be able to detect numerous objects.


- We aim to build further on this algorithm to improve and enhance its capability.

- Can we train the neural network for better scalability in small, embedded machines?

- Can we improve its processing time, make it more reliable?

- Customize yolo with our own datasets and object detection definitions.

Object Counting and Tracking

An important step of building the “comprehensive information processing engine” is to ensure the vehicle is aware of the real-time conditions around it. Our algorithm strives to:

- Count the number of relevant objects in real-time (cars, buses, pedestrians, cyclists).

- Combine this data with GPS and infrastructure information to build traffic models.

- Enable object tracking. “Is this the same car from the intersection earlier?”

- Enable counting persistence. “Is the blue sedan from frame one, same as the blue sedan from frames two?”

Benchmarks

Below is a comparison of Yolo with other established image recognition algorithms. The table shows the algorithm along with the dataset used to train it (Pascal VOC 2007/2012), the mean average precision (mAP) of correctly identified objects and the frames (processed) per second (FPS).

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Dataset</th>
<th>Train</th>
<th>mAP</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real-Time Detectors</td>
<td>Train</td>
<td>mAP</td>
<td>FPS</td>
<td></td>
</tr>
<tr>
<td>Faster R-CNN (F)</td>
<td>Train</td>
<td>2007</td>
<td>30.4</td>
<td>15</td>
</tr>
<tr>
<td>R-CNN Mobile R [46]</td>
<td>Train</td>
<td>2007</td>
<td>53.5</td>
<td>6</td>
</tr>
<tr>
<td>YOLO VGG-16</td>
<td>Train</td>
<td>2007</td>
<td>68.4</td>
<td>33</td>
</tr>
</tbody>
</table>

Face Recognition

- License plate hashing. We blur out and anonymize license plates for information security, but also preserve their ability to be a unique identifier.

Privacy Concerns

The sensitive nature of the data our algorithm collects makes it imperative to have, out of the box, some information protection features. Two important features we have incorporated in this project are:

- Color out the faces of detected people.

Future Plans:

In the vision to build the “comprehensive information processing engine”, we have set out on an interesting mission as a possible “step 2” of our project:

- Bicyclist detection we think is an immediate and practical application of our algorithm.

- Is there a way to detect the bicyclist while also recognizing what kind of bike is it?

- Can we detect the presence of a helmet-wearing bicyclist?

- The ability can help us recognize the level of awareness (and hence risk) on the streets of our cities.

- It would help increase awareness and also make the decision maker (driver or car) more informed.

- Is it possible to sensitize the neural network framework to recognize helmet-wearing cyclists in various situations?

Accreditations:

This project is run with the support of: Illinois Geometry Lab, Engineering at Illinois, Illinois Applied Research Institute, Department of Computational Science and Engineering, Ernst Energy Laboratory, College of Liberal Arts and Sciences, Actuarial Science Program, National Centre for Supercomputing Applications.