MAIS 202 - PROJECT DELIVERABLE 2

Home Invasion Detection

1. Problem Statement:

The Home Invasion Detection project aims to identify human intruders. The model will be trained to identify human subjects in home environments, to be able to subsequently notify house tenants in the event of a possible home invasion.

2. Data Preprocessing:

Although my initial dataset of choice was the COCO dataset, I have chosen to move forward using Google's Open Images Dataset V6 to train my model. As I will elaborate on later in this document, I have chosen to complete this project using the YOLOv3 model. Thus, my dataset and its corresponding information had to follow a format specific to the YOLOv3 training constraints. As such, Google's Open Images Dataset V6 was an extremely convenient dataset to use, considering all images in the dataset came with corresponding bounding box information for each class.

I focussed on gathering data from two classes, "person" and "furniture", as my project aims to identify intruders in a home environment. 400 instances of each class were used in the final training dataset.

To first download the data, I opted to use the OIDv4 toolkit, as it allowed me to download both the images for the dataset as well as annotation files for each. I then proceeded to use an additional python script written by theAIGuy to convert the .txt annotation files to the proper format for training.

3. Machine Learning Model:

Upon discussing with Zedi, my Technical Project Manager for this project, I have decided to use the YOLOv3 (You Only Look Once). Considering the fact that the aim of this project is to identify intruders quickly and notify house tenants immediately, choosing a model that focusses on speed of detection was important. YOLO uses a single neural network, which allows the model to achieve high detection speeds, which was optimal for my project.

a. To train my model, I followed steps specified in theAIGuy's guide to yolov3 custom object detection model training protocol.
b. As a preliminary test to see the performance of the model I have trained, I ran the trained model on an image containing two "person" objects. The bounding boxes that the YOLOv3 model produces assigns a certain confidence metric to every label it presents, which I have used to assess the performance of the model. I suspect my model to be overfitting to the dataset, as the confidence in prediction of the model when faced with a new picture was quite low.
c. The biggest challenge faced in this step of the process was understanding how to work with the YOLOv3 model, especially when dealing with preprocessing of data. Due to the strict constraints imposed on training data of the YOLOv3 model, it was necessary to conduct much research to find a proper pre-processing method. Furthermore, in the training process, I kept encountering a bug which was ultimately being caused by wrong formatting in my code. This led the model to be tained on the first 100 instances numerous times before I could know if the bug was fixed.

4. Preliminary results:

As mentioned above, my preliminary results are not as satisfactory as I had wished. As shown in the picture below, the confidence in the prediction that the model has made is 0.66. Furthermore, it failed to notice the people in the back, as well as separate the two people at the front of the picture.

5. Next steps:

Considering the prediction confidence of the model, I suspect there will be some fine-tuning that is needed. Instead of training the model on two classes, I may opt to train it on one class "person", and increase the data instances that it is trained on, since there is no particular need to identify furniture instances in my project. In terms of further work upon editing my model, I hope to implement a web page in which I present video surveillance footage, labeled with the YOLOv3 predictions to showcase how the model may work in practice.

Bibliography

@article{DBLP:journals/corr/RedmonDGF15,
  author   = {Joseph Redmon and
               Santosh Kumar Divvala and
               Ross B. Girshick and
               Ali Farhadi},
  title   = {You Only Look Once: Unified, Real-Time Object Detection},
  journal   = {CoRR},
  volume   = {abs/1506.02640},
  year   = {2015},
  url   = {http://arxiv.org/abs/1506.02640},
  archivePrefix = {arXiv},
  eprint   = {1506.02640},
  timestamp = {Mon, 13 Aug 2018 16:48:08 +0200},
  biburl   = {https://dblp.org/rec/journals/corr/RedmonDGF15.bib},
  bibsource = {dblp computer science bibliography, https://dblp.org}
}