PR2: Object Detection

Published Date:
Feb. 1, 2024, 12:00 a.m.

Deadline Date:
Feb. 20, 2024, 11:59 p.m.

Description:

****
This is an individual assignment.
****

Overview and Assignment Goals:

The objective of this assignment are as follows:

- Train one or more detection models via transfer learning.
- Evaluate hyperparameter choices and how they affect the learning.
  - Use the validation set to validate parameter choices.
- Choose the best model, i.e., best classifier + best hyper-parameter choices.

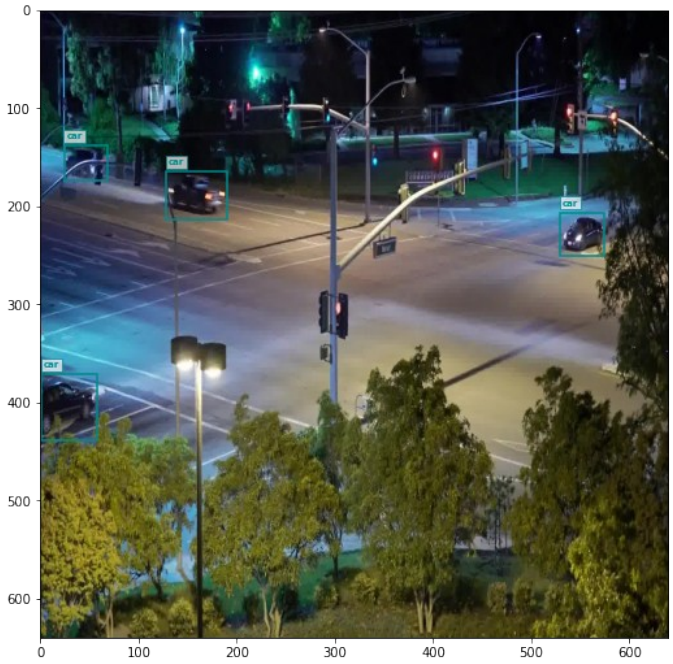
Detailed Description:
Develop predictive models that can determine, given an image, the class and location (bounding box) of objects of 3 types, according to the labels assigned in the training/validation set.

Learn to partially re-train a convolutional neural network adept at object detection (localization and classification). While existing detection networks (e.g., SSD, Yolo, etc.) are trained for many different classes, your model should be re-trained to only detect three kinds of objects: *car* (including SUVs, vans, pick-up trucks and other small trucks)*, medium truck* (e.g., Amazon of FedEx delivery trucks)*, large truck* (18-wheelers, buses, cargo trucks)*,* labeled as 1–3.

The program 2 dataset can be found on the HPC at /WAVE/projects/CSEN-342-Wi24/data/pr2. Please do not copy the dataset to any other location. It contains 14,000 training images and labels (*train* directory), 2,000 validation images and labels (*val* directory), and 2,000 test images (*test* directory). The train and val directories have a label text file that uses the following format: <image_id> <class> <cx> <cy> <w> <h>. The class is labeled 1 through 3. The remaining numbers are pixel values within the image canvas, where <cx, cy> are the horizontal and vertical coordinates of the center of the bounding box, and <w> and <h> are the width and

the height of the bounding box. The origin point of each bounding box is the top left corner. For example, the image below, which has been resized to 640x640 pixels, has the following 4 bounding box labels:

1 7 159 188 62 49
1 7 46 155 43 34
1 7 29 404 56 66
1 7 552 228 45 43



Some models require relative bounding box locations. These labels are easily translated to relative values by dividing cx/width, cy/height, w/width, and h/height, where *width* and *height* are the image width and height, giving the following labels:

1 7 0.2492187 0.2949074 0.0973958 0.0768519
1 7 0.0718750 0.2430556 0.0677083 0.0546296
1 7 0.0458333 0.6319444 0.0885417 0.1046296
1 7 0.8627604 0.3564815 0.0703125 0.0685185

The label in red belongs to the car that is in the lower left corner of the image. The bounding box has its center at coordinates <29, 404> and has a width of 56 pixels and a height of 66 pixels.

Note, however, that your dataset has images of size 960 x 540.

Your model should make predictions on each of the test images and store results in

a single text file, in the following format:

<image_id> <class> <cx> <cy> <w> <h> <conf>

The <image_id> is simply the file name minus the extension, i.e., the number of the image. The remaining values are the predicted class (1–3), the bounding box coordinates for the detection, as **relative values (not pixel coordinates)**, and the confidence of the prediction. Each new detection should be included in a new line in the file.

For evaluation purposes (Leaderboard Ranking) we will use the Mean Average Precision (mAP) metric comparing the predictions submitted by you on the test set with the ground truth. Some things to note:

- Some of your classmates may choose not to see the leaderboard status prior to the submission deadline. Please do not share leaderboard status information with others.
- The public leaderboard shows results for 50% of randomly chosen test instances only. This is a standard practice in data mining challenges to avoid gaming of the system. The private leaderboard will be released after the submission deadline, based on all the entries in the test set.
- In a given day (00:00:00 to 23:59:59), you are allowed to submit a prediction file only 5 times.
- The final ranking will always be based on the last submission unless you specify a certain submission to be used. Carefully decide what your final submission should be.

Rules:

- This is an individual assignment. Discussion of broad level strategies are allowed but any copying of prediction files and source codes will result in an honor code violation. This includes reusing code posted on the Web by others.

Deliverables:

- **Valid submissions to the Leader Board website:** https://clp.engr.scu.edu (username is your SCU username and your password is your SCU password). You must submit your code as a Python file along with your predicted labels. Your submitted code should produce the submitted labels.
- **Canvas submission of your report:**
  Write a 2–3-page, single-spaced report describing details regarding the steps you followed for implementing and testing your models. The report should be in PDF format and the file should be called **<SCU_ID>.pdf**. Be sure to include the following in the report:
  - Name and SCU ID.
  - Rank & score for your submission (at the time of writing the report). If

you chose not to see the leaderboard, state so.

- Your approach.
- Your methodology of choosing the approach and associated parameters.
- Graphs showing training/validation loss in different tested scenarios/models.
- A discussion on your bias/variance analysis for your training.
- Any special instructions for running your code.

Grading:

Grading for the Assignment will be split on your implementation (70%), and report (30%). Extra credit (1% of final grade) will be awarded to the top-3 performing algorithms. Note that extra credit throughout the quarter will be tallied outside of Canvas and will be added to the final grade at the end of the quarter.

Files you will find on WAVE:

- *Train directory:* 14000 images and labels
- *Val directory:* 2000 images and labels
- *Test directory:* 2000 images to find objects in