# **INTER-CAMERA OBJECT TRACKING**

* **Problem Statement:** Develop an approach and write to code for tracking similar objects over multiple images. As an example, if the same car was found in images in two different traffic cameras, we want to be able to group those two images and state than those images contain the same car.
* **My approach to the problem:**

**Assumption:** Similar objects will give similar hash values if a hash function is applied over the area of the image that the object occupies. This would lead to a significant computational advantage as we would have smaller clusters of images to compare them against as opposed to the entire dataset that we have.

* + - * + Extract all bounding boxes for all vehicles from all the images over a 2 minute time period using an object detector. In our case, we used SSD (Liu, 2016), i.e., Single Shot Detector.
        + Apply a hash function over each bounding box so that instead of comparing the bounding box again each other, which would be extremely expensive computationally, we would first cluster the hash values and then compare bounding boxes to other bounding boxes *in the same cluster.*
        + Use Mean Shift clustering (Lars, 2013) with suitable initial values, to cluster nearby hash values together.
        + This is the step that the code has not been able to solve yet. We need to compare the bounding boxes in a given cluster against other bounding boxes in the same cluster. The approach that we initially took was to work on trying to get direct pixel by pixel comparison giving leeway for a margin of error, which failed for two major reasons. Firstly, not all of our cameras provide images at the same angle. Secondly, the bounding boxes/cars are not necessarily of the same size.
        + This step is just supposed to return sets of images with similar objects after running the comparison in the previous step, which we haven’t been able to so far.
* **Installation instruction and code:** 
  + - The code can be found at this link: <https://github.com/PurdueCAM2Project/CAM2ImageTeam/tree/master/InterCamTracking>
    - This can be accessed via a simple pull command.
    - Before you run create class, you need to have a folder of images on which you run an object detection algorithm on. The approach I have used is SSD. What SSD does is that is labels images are provides them in a file containing all the objects labels. The name of that file in the repository is labels.txt.
    - To use this code, just run the command python createClass.py, before you do this however, you need to have a file of labels as mentioned in the previous step.
    - The output of the code is clusters of images which might have similar objects. That is all the progress that has been made on this problem that has been made so far.
* **Possible alternative approaches:**
  + - * + If we have geographical data and timing we could build a probabilistic model to tell whether two objects are similar. Say, a pink car was spotted in a cameras 3 miles apart with a time difference of 5 minutes, we could say it is likely that both the images contain the same car. But, beyond rare colors and models, even this approach would break down.