CSC 7991: Intro to Deep Learning

Group 2
Jamal Warida
Mustafa Ahmed
Kyle W. Brown
Sujeet Shrestha

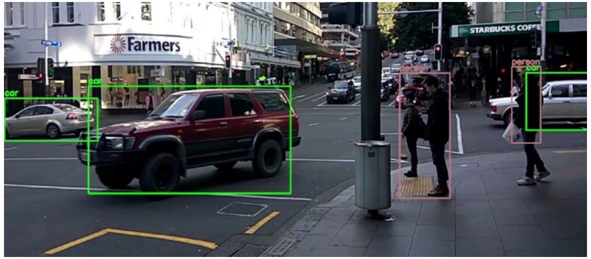
Vehicle Recognition using Single-Shot Detection for Autonomous Implementation.

Wayne State University

May 15, 2019

Overview

- Single-Shot Detection (SSD)
- Model
- Training
- Experiments
- Related Work
- Developing Applications
- Conclusion



YouTube

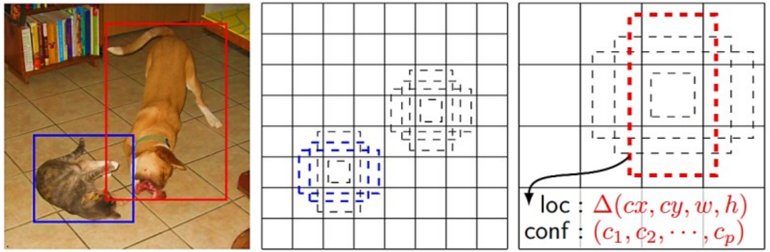
Objective: Identify vehicles in nighttime, snowy, and drone YouTube videos using SSD's bounding boxes for vehicle recognition.

Introduction

- Current state-of-the-art object detection systems are variants of the following approach:
  - Hypothesize bounding boxes
  - Resample pixels or features for each box
  - Applying a high-quality classifier
- Approaches are computationally intensive for embedded systems.
- Too slow for real-time applications.

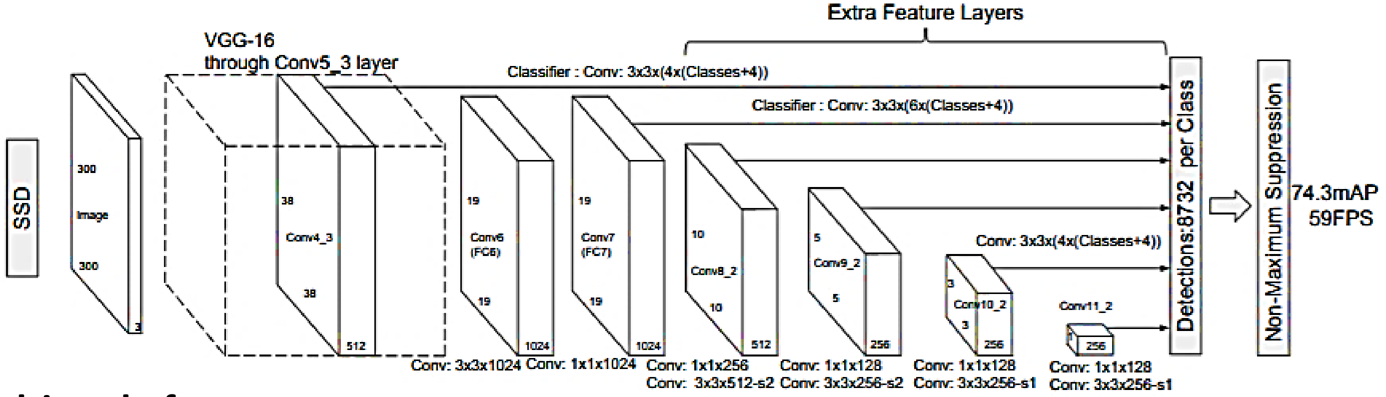
Single Shot Detection (SSD)

- **SSD**, is a single-shot detector for multiple categories that is faster than the previous state-of-the-art for single shot detectors (YOLO).
  - The core of SSD is predicting category scores and box offsets for a fixed set of default bounding boxes using small convolutional filters applied to feature maps.
  - To achieve high detection accuracy, predictions of different scales are produced from feature maps of different scales, then predictions are separated by aspect ratio.



(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

Model



- **Multi-scale feature maps**
  - Convolutional feature layers are added to the end of the network which decrease in size and allow predictions of detections at multiple scales
- **Convolutional predictors**
  - Bounding box offset output values are measured relative to a default box position relative to each feature map location.
- **Default Boxes and aspect Ratios**

Training

The overall objective loss function is a weighted sum of the localization loss (loc) and the confidence loss (conf):

$$L(x,c,l,g) = \frac{1}{N}(L_{conf}(x,c) + \alpha L_{loc}(x,l,g)) \tag{1}$$

The overall objective loss function is a weighted sum of the localization loss (loc) and the confidence loss (conf):

$$\begin{aligned} L_{loc}(x,l,g) &= \sum_{i \in Pos}^{N} \sum_{m \in \{cx,cy,w,h\}} x_{ij}^{k} \mathrm{smooth_{L1}}(l_i^m - \hat{g}_j^m) \\ \hat{g}_j^{cx} = (g_j^{cx} - d_i^{cx})/d_i^w \qquad & \hat{g}_j^{cy} = (g_j^{cy} - d_i^{cy})/d_i^h \\ \hat{g}_j^{w} = \log\left(\frac{g_j^w}{d_i^w}\right) \qquad & \hat{g}_j^{h} = \log\left(\frac{g_j^h}{d_i^h}\right) \end{aligned} \tag{2}$$

where N is the number of matched default boxes. If N = 0, wet set the loss to 0. The localization loss is a Smooth L1 loss between the predicted box (*l*) and the ground truth box (*g*) parameters.

Training

The confidence loss is the softmax loss over multiple classes confidences (c).

$$L_{conf}(x,c) = -\sum_{i\in Pos}^{N} x_{ij}^{p} log(\hat{c}_i^p) - \sum_{i\in Neg} log(\hat{c}_i^0) \quad \text{where} \quad \hat{c}_i^p = \frac{\exp(c_i^p)}{\sum_p \exp(c_i^p)} \quad (3)$$

Suppose we want to use *m* feature maps for prediction. The scale of the default boxes for each feature map is computed as:

$$s_k = s_{\text{min}} + \frac{s_{\text{max}} - s_{\text{min}}}{m-1}(k-1), \quad k \in [1, m] \qquad (4)$$

where S_{min} is 0.2 and S_{max} is 0.9, meaning the lowest layer has a scale of 0.2 and the highest layer has a scale of 0.9, and all layers in between are regularly spaced.

Experiments

- Bounding boxes to identify vehicles from YouTube videos.
- Experiments designed to test vehicle recognition using SSD to identify vehicles.
- Experiments include:
  - Identifying moving and parked vehicles in snowy conditions.
  - Using bounding boxes to identify vehicles in nighttime video.
  - Drone videos of vehicles moving and parked.

Snowy Experiments

Parked Vehicles

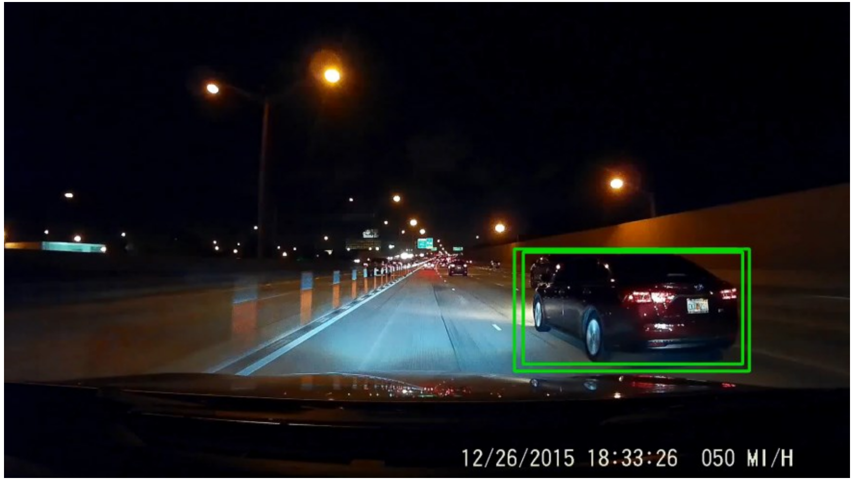
Distance	Total Vehicles	Identified Vehicles	Accuracy
< 10 ft	25	4	16%

Moving Vehicles

Distance	Total Vehicles	Identified Vehicles	Accuracy
< 5 ft	9	4	44 %

Night Time Experiment

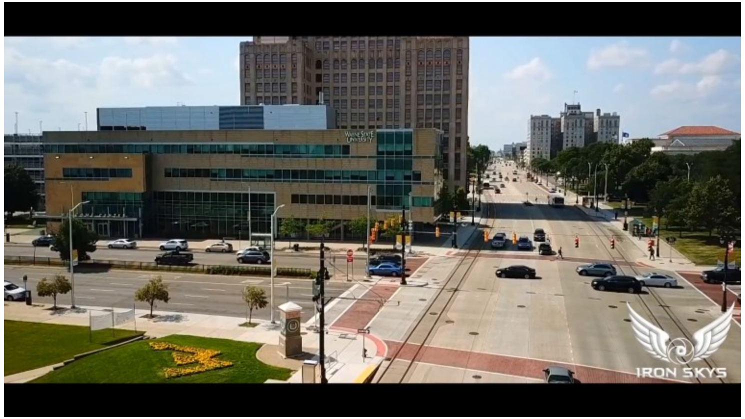
Moving Vehicles



Distance	Total Vehicles	Identified Vehicles	Accuracy
< 10 ft	14	6	43 %

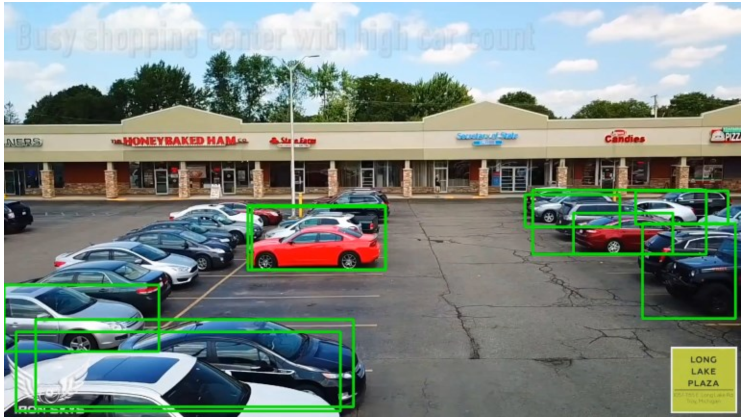
Drone Experiments

Moving Vehicles



Distance	Total Vehicles	Identified Vehicles	Accuracy
> 50 ft	17	3	18 %

Parked Vehicles



Distance	Total Vehicles	Identified Vehicles	Accuracy
< 15 ft	28	23	82 %

Related Work

Method	mAP	FPS	batch size	# Boxes	Input resolution
Faster R-CNN (VGG16)	73.2	7	1	~ 6000	$\sim 1000 \times 600$
Fast YOLO	52.7	155	1	98	448×448
YOLO (VGG16)	66.4	21	1	98	448×448
SSD300	74.3	46	1	8732	300×300
SSD512	76.8	19	1	24564	512×512
SSD300	74.3	59	8	8732	300×300
SSD512	76.8	22	8	24564	512×512

- SSD300 is the only real-time detection method that can achieve above 70% mAP.
- When using a larger input image, SSD512 outperforms all methods on accuracy while maintaining a near real-time speed.

Developing Applications

- Autonomous
  - ADAS
- Military
  - Transportation
  - Bomb disposal
- Drones
  - Emergency & Rescue
  - Object detection & tracking

Conclusion

- Discussed SSD, a fast-single shot detector for multiple categories.
- Introduced state-of-the-art related work using SSD for vehicle recognition using YouTube videos.
- Experiments
  - Snowy conditions:
  - Nighttime conditions:
  - Drone videos

Experiment	Accuracy
Snowy -Parked	16%
Snowy - Moving	44%
Nighttime	43%
Drone - Moving	18%
Drone - Parked	82%

- New and developing technology with many potential military, rescue, and autonomous applications.

Any Questions or Comments?