A Comparison of Embedded Deep Learning Methods for Person Detection

https://www.scitepress.org/Papers/2019/73863/73863.pdf

summary:

general theme

In our project we have research the performance of different deep learning platforms which is provide an object detection ability.

Our purpose was to develop autonomous car model based on **RC car** and embedded linux machine as a microprocessor (raspberry pi and finally jetson nano).

For this task it is necessary to choose a very efficient object detector (for detection of cars, pedestrians and traffic signs), which is able to find these objects in real time.

RCNN Variants

Despite RCNN has
lots of advantages over the conventional DNN object detector, this technique is still quite slow for any real-time application.

YOLO Variants

You only look once (YOLO) is another state of the art object detection algorithm which mainly targets real time applications. it looks at the whole image at test time and its predictions are informed by global context in the image.

It also makes predictions with a single network evaluation unlike models such RCNN, which require thousands for a single image.
YOLO divides the input image into an SxS grid. If the center
of an object falls into a grid cell, that cell is responsible for detecting that object.

SSD Variants

Single shot multi-box detector (SSD) is one of the best object detector in terms of speed and accuracy.
The SSD object detector comprises two main steps including feature maps extraction, and convolution filters application to detect objects.

SqueezeDet

SqueezeDet is a real-time object detector used for autonomous driving systems.
This model claims high accuracy as well as reasonable response latency, which are crucial for autonomous driving systems.
Inspired by the YOLO, this model uses fully convolutional layers not only to extract feature maps, but also to compute the bounding boxes and predict the object classes.
The detection pipeline of SqueezeDet only contains a single forward pass over the network, making it extremely fast.

CONCLUSION

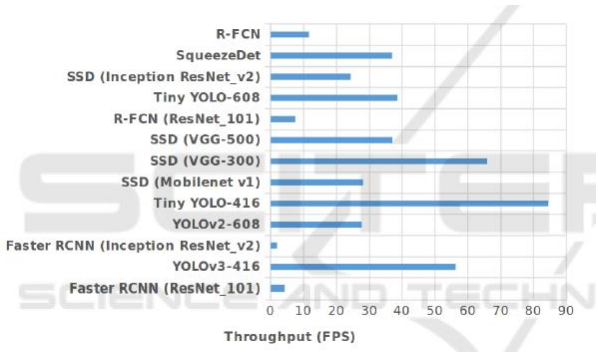
Experiments results shows that Tiny YOLO-416 and SSD (VGG-300) are among the fastest models and Faster RCNN (Inception ResNet-v2) and RFCN (ResNet-101) are the most accurate ones.
However, neither of these models nail the tradeoff between speed and accuracy.
Further analysis indicates that YOLO v3-416 delivers relatively accurate result in reasonable amount of time, which makes it a desirable model for person detection in embedded platforms.

Table 1: Average precision at IoU 0.95 and 0.50.

#	Model	Framework	AP [IoU=0.95]	AP [IoU=0.50]
1	Faster RCNN (ResNet-101)	Tensorflow	**0.245**	**0.476**
2	YOLOv3-416	Darknet	0.143	0.367
3	Faster RCNN (Inception ResNet-v2)	Tensorflow	**0.317**	**0.557**
4	YOLOv2-608	Darknet	0.198	**0.463**
5	Tiny YOLO-416	Darknet	0.035	0.116
6	SSD (Mobilenet v1)	Tensorflow	0.094	0.233
7	SSD (VGG-300)	Tensorflow	0.148	0.307
8	SSD (VGG-500)	Tensorflow	0.183	0.403
9	R-FCN (ResNet-101)	Tensorflow	**0.246**	**0.486**
10	Tiny YOLO-608	Darknet	0.06	0.185
11	SSD (Inception ResNet-v2)	Tensorflow	0.116	0.267
12	SqueezeDet	Tensorflow	0.003	0.012
13	R-FCN	Tensorflow	0.124	0.319

Table 2: Total latency of inference in both CPU and GPU modes.

#	Model	CPU Latency (S)	GPULatency (S)
1	Faster RCNN (ResNet-101)	3.271	0.232
2	YOLOv3-416	5.183	0.017
3	Faster RCNN (Inception ResNet-v2)	10.538	0.478
4	YOLOv2-608	11.303	0.035
5	Tiny YOLO-416	1.018	0.011
6	SSD (Mobilenet v1)	**0.081**	0.03
7	SSD (VGG-300)	0.361	**0.015**
8	SSD (VGG-500)	0.968	**0.026**
9	R-FCN (ResNet-101)	1.69	0.131
10	Tiny YOLO-608	2.144	**0.025**
11	SSD (Inception ResNet-v2)	**0.109**	0.04
12	SqueezeDet	**0.14**	**0.027**
13	R-FCN	3.034	0.084



Summarized by: Oren lacker