Deep Learning’s Application in Driverless Cars

Gan Luan, Wenlong Feng

1. Introduction

A driverless car, also known as a robot car, autonomous car, or self-driving car, is a vehicle that is capable of sensing its environment and moving with little or no human input [1]. This is one of the technology that might change everyone’s life dramatically in the future. Both traditional automobile industry and high tech companies have put a lot effort in developing driverless car, such as Audi, Ford, Waymo, Tesla, Uber and so on.

According to SAE International (Society of Automotive Engineers) there are six levels of automation for driverless car, from no driving automation (Level 0) to full driving automation (Level 5). One criterion in determining the level of automation is how much time does the system allow for driver to react and to take over [1, 2, 3].

- **Level 0** (“no automation”): This automation accounts for the majority of the vehicles on road today, where automated systems have no control of the vehicle.
- **Level 1** (“hands on”): Both the driver and automated system share control of the vehicle. One example is the Adaptive Cruise Control, where the driver controlling the steering and break, and the systems controls speeds. And drive must be ready to take over at any time.
- **Level 2** (“hands off”): The automated systems fully control the vehicle (speed, brake, steering). However, the driver still needs to ready to fully take control immediately at any time. One example is the Autopilot features of Tesla.
- **Level 3** (“eyes off”): Drivers can turn their attention away from the road. The system can handle situation that needs immediate action, such as emergency braking. In case of the situation that the system can not handle, driver will have sometime to react to take control. One example is the traffic jam pilot on 2019 Audi A8. Audi claims: if the customers turns the traffic jam pilot on the uses it as intended, and the car was in control at the time of the accident, the driver goes to his insurance company and the insurance company will compensate the victims of the accident and in the aftermath they come to us and we have to pay them. (For some reasons, the function is not included for 2019 Audi A8 released in US.)
- **Level 4** (“minds off”) No human attention is required for safety, e.g. the driver may safely go to sleep or leave the driver seat. However, this self driving is only supported in a certain circumstance, such as traffic jam, or highway. In the

circumstance that the system can not handle, it can safely abort the trip, such as park the car.

- **Level 5** ("steering wheel optional") No human interaction is required at all.

There are both advantages and disadvantages for technologies [1,4]. Some of the advantages are:

- **Safety**: When the self-driving technology is fully developed, traffic accidents due to human errors, such as drunk drive, reaction time, road rage or other form of distracted or aggressive driving.
- **Welfare**: Autonomous car can free people from driving, so people can have more time for whatever they enjoy more. Also this provides more mobility for elder people, children, and people with disability.
- **Reduced Traffic Congestions**: Some of the congestions are caused by human driving behavior. With carefully planning, auto driving can reduce the traffic congestions. Also decreasing traffic accidents can reduce some traffic congestions. Several potential benefits for less traffic congestions include: less commute time, less CO_2 emission.
- **Parking Space**: Manually driving vehicles are been used on road only for 4-5% of the time, and being parked for the rest of time. When auto driving technology is fully developed, vehicles can be used continuously after it has reached it destination. Thus, auto driving will greatly reduce the need for parking space.

There are also some potential disadvantages associated with driverless car. Some of them are:

- **Unemployment**: Truck drives, taxi drivers, and also public transit drivers will lose jobs, since auto driving will takeover. Development of auto driving may receive resistance from unions of these workers. All the auto repair related professionals will suffer, since there will be less car accidents.
- **Cost**: Since driverless car requires the most updated technology and several different kinds of sensors, the price will be high. It may not be affordable for most people for a long time.
- **Safety Concern**: One mistake in the system, even very tiny one, could lead to huge damage that human error can cause. Also there are will be a disaster once the system is attacked by hacker. There is also a potential risk of terrorist attacks. Self-driving cars can be loaded with explosives and be used as bombs.
- **Privacy Concern**: The vehicle's location and position will be integrated to an interface in which other people have access to. Also for better planning,

information will be shared between cars on the same road. These cause privacy concerns.

Several Milestones have been achieved for developing of autonomous driving. Waymo (a company by Google) now provide auto driving rides within Phoenix, AZ and outside Phoenix (though safety drivers still needed to sit behind the wheel). By July, 2018, Waymo has driven more than 8 million miles autonomously, and Uber has driven more than 3 million miles autonomously [5,6]. Tesla has already drive more than 1 billion miles with Autopilot been activated [1]. And as mentioned above, Audi has released the first L3 auto driving car, with the traffic jam pilot.

2. Sensors and Autonomous Driving Tasks

2.1 Sensors mounted on autonomous cars

A typical autonomous car system equips multiple sensors [14], including on-board and in-car sensors, radar, LIght Detection and Ranging (LIDAR) tracking, cameras, and communication devices, to acquire a lot of real-time data. The acquired data are processed by the autonomous car’s central computer system and subsequently used by the decision-support system. The decision-support system actuates the autonomous car.

Different ranges of situational awareness apply to different applications, and they are achieved through different components. For instance, front and rear bumper collision are avoided through infrared devices; lane-change warning, short-range object detection, and traffic view construction are provided by short ranges; surrounding views are captured by a series of cameras; LIDAR is used for collision avoidance and emergency brakes; the cooperative cruise control and long-range traffic view construction and achieved by long-range radars.

2.2 Tasks in the autonomous driving

Autonomous cars are intelligent agents that need to perceive, predict, decide, plan, and execute their decisions in the real world, often in uncontrolled or complex environments. For a driverless car moving from the starting point to the destination, the car needs to perceive the surrounding environment, plan the trip, navigate, and make controlled movements on the road. These tasks can be summarized to three primary steps [14]: 1) situational and environmental awareness; 2) navigation and path planning; and 3) maneuver control.

The first important step for autonomous cars is neighborhood awareness including subject tracking, self-positioning, and lane sporting. The car must perceive what are surrounding it especially in the front direction. Cameras are frequently used for environmental and neighborhood awareness. However, the volume and speed of real-time data required for neighborhood awareness are too compute-intensive for vehicular computing. Moreover, the resolution of data obtained from cameras is inversely proportional to the speed and performance of the decision support system. LIDAR tracking is often used to get a full view of the surrounding environment since it can provide 360-degree visualization and object tracking with a relative long range. For intensive object detection such as collision resistance while parking, collision avoidance, and bumper protection, optimized radars are installed at the front, rear and sides of the car.

The primary function of the navigation system on an autonomous car is to enable the car to travel on the desired path. After the autonomous car is aware of its environment, it needs to plan a path based on the destination. The global positioning system (GPS) is the primary sources of navigation for the car. The navigation system must be robust to handle sudden and subsequent changes in the path by adjusting the already pre-compute route. Inertial navigation systems are applied when GPS signal is blocked or deteriorated by natural or artificial phenomena, such as underground roads and tunnels. The Inertial navigation system can use gyroscopes and accelerometers to capture the moving trajectory of the car. Combining with the GPS data of certain points on the trajectory, the navigation system can detect whether the car is moving on the desired path.

The maneuver control refers to the specific movement control of the autonomous car on its journey based on the perception of its surroundings and the navigation to the destination. It likes the executive branch of autonomous driving. During the journey, the autonomous car must maintain various maneuvers including lane keeping, bumper-to-bumper distance, sudden brakes, overtaking, and stopping at traffic lights. Most of the car components are electronically controlled by the decision-support system through the controller area network and electronic control units inside the vehicle.

3. Deep learning in autonomous car perception

One of the crucial aspects of autonomous cars is perception, and it is a good candidate for applying deep learning models. Autonomous cars mimic human-like perception capability. Most of computer vision related algorithms and mechanisms, such as object detection, scene identification, reconstruction and estimation, use both machine learning and deep learning mechanisms. Object detection and semantic segmentation are two fundamental and challenging problems in the perception. We elaborate the deep learning techniques in addressing these two problems in the following sections.

3.1 Object detection

The goal of object detection is to determine whether or not there are any instances of objects from the given categories (such as humans, cars, bicycles) in a scene and to return the spatial location and extent of each object instance [15]. Objects are usually recognized by estimation classification probability and localized with bounding boxes. As the cornerstone of computer vision, object detection forms the basis for solving more complex vision tasks, such as segmentation, scene understanding, and object tracking.

The object detector can be broadly grouped into two categories [15]: Two-stage detection framework, which includes a pre-processing step for region proposal, making the overall pipeline two-stage; One stage detection framework, or region proposal free framework, which is a single proposed method which does not separate detection proposal, making the overall pipeline single-stage. Several frequently referred objected detector are briefly introduced in the subsequent sections.

1. Region-based Convolutional Neural Networks (R-CNN)
   CNNs were too slow and computationally very expensive. It was impossible to run CNNs on so many patches generated by sliding window detector. Girshick et al. proposed R-CNN to solve the problem in 2013 [16]. R-CNN solves this problem by using an object proposal algorithm called Selective Search which reduces the number of bounding boxes that are fed to the classifier to close to 2000 region proposals. The framework of R-CNN is illustrated in figure 1. Selective Search scans the input images and uses local cues, such as texture, intensity, and colors, to generate all the possible regions of the object. A CNN model is subsequently run on top of each region proposal. The output of each

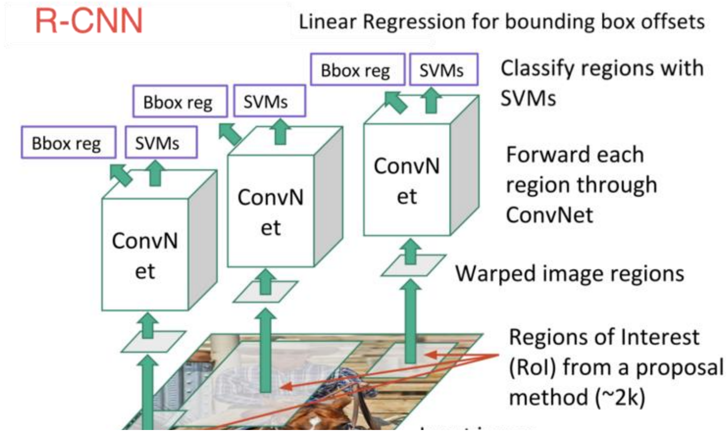
R-CNN
Linear Regression for bounding box offsets
Bbox reg
SVMs
Classify regions with SVMs
Bbox reg
SVMs
Bbox reg
SVMs
ConvNet
ConvNet
ConvNet
Forward each region through ConvNet
Warped image regions
Regions of Interest (RoI) from a proposal method (~2k)

Figure 1The framework of region-based convolutional neural network.

CNN is fed into an SVM to classify the region and a linear regressor to tighten the bounding box of the object if the object exists. Notes that the fully connected part of CNN only takes a fixed sized input.

2. Fast R-CNN

   Fast R-CNN is the immediate descendant of R-CNN. Girshick et al. [17] proposed it to address some disadvantages of R-CNN. It improved on its detection speed through two improvements:

   1) Performing feature extraction over the image before proposing regions, therefore only running one CNN over the entire image
   2) Replacing the SVM with a softmax layer, thus extending the neural network for predictions instead of creating a new model

   The framework of Fast R-CNN is illustrated in Figure 2. As it presents, the generation of region proposals is based on the last feature map of the convolutional part, not from the original image input. Therefore, only one CNN is trained for the entire image. Moreover, a single softmax layer is used to directly output the class probabilities instead of training multiple SVMs to classify object classes. These modifications assign Fast R-CNN a better performance in terms of speed comparing to R-CNN. However, the selective search algorithm for generating region proposals still constrains the speed of Fast R-CNN.

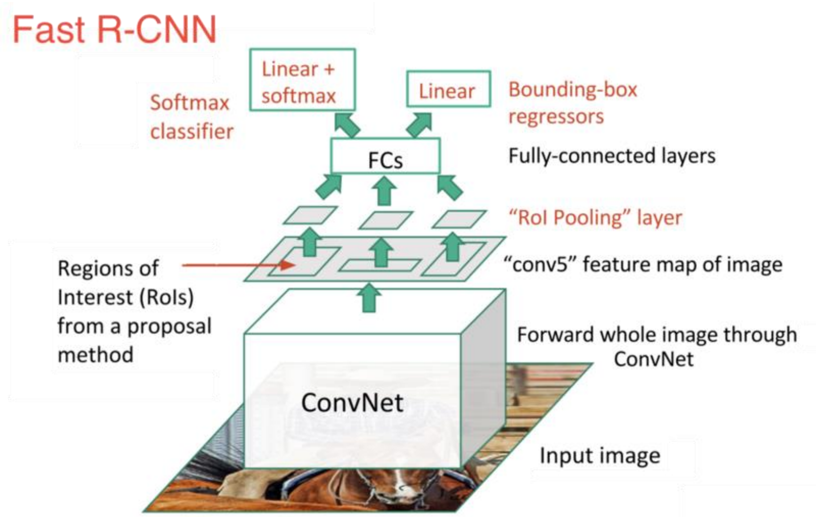

Figure 2 The framework of Fast R-CNN

3. Faster R-CNN

   The Faster RCNN framework (Figure 3a [18]) proposed by Ren et al. in 2015. The main improvement of Faster R-CNN comparing to Fast R-CNN is replacing the selective search algorithm with a very small convolutional network called Region Proposal Network (RPN) to generate regional proposals. A brief workflow of RPN is illustrated in Figure 3b. At the last layer of an initial CNN, a 3*3 sliding window moves across the feature map and maps it to a lower dimension. For each sliding-window location, it generates multiple possible regions based on K fixed-ratio anchor boxes (default bounding boxes). Each region proposal consists of an objectness score for that region and 4 coordinates representing the bounding box of the region. In other words, the algorithm looks at every location in the last feature map and consider K different boxes (tall box, wide box, large box, etc.) centered around it. For each of those boxes, the algorithm output whether it thinks the box contains an object and the coordinates of the box. The 2k scores stand for the softmax probability of each of the k bounding boxes being on “object”. Although the RPN outputs bounding boxes, it doesn’t classify any potential objects. If an anchor box has an “objectness” score above a certain threshold, the coordinates of this box will be passed forward as a regional proposal. The region proposals are further fed to a pooling layer, somefully connected layers and finally a softmax classification layer and bounding box regressor.

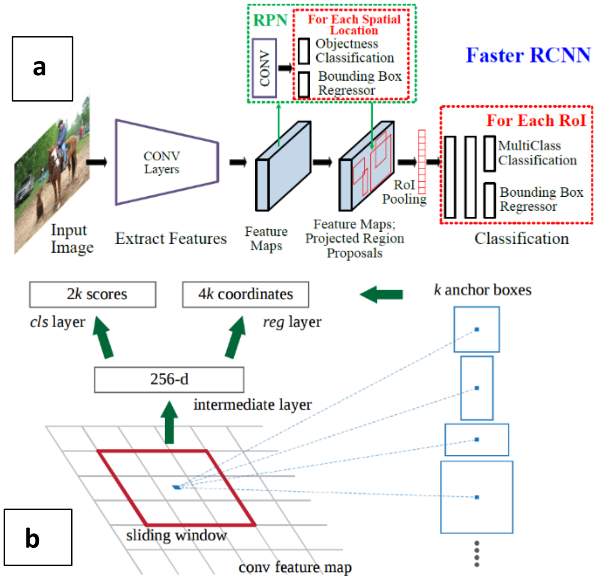

Figure 3 (a) Framework of Faster R-CNN and (b) Diagram of Region Proposal Network.

3.2 Semantic Segmentation

Segmentation means that to partition the image into several coherent parts without any attempt to understand what these parts meaning. On the other hand, semantic segmentation attempts to partition the image into semantic meaningful parts and label each part with prespecified classes (see Fig.4 [7,8]). For pixel-wise semantic segmentation, each pixel needs to be labelled with a class. Basically the semantic segmentation is try to semantically understand the role of each pixel in the image.

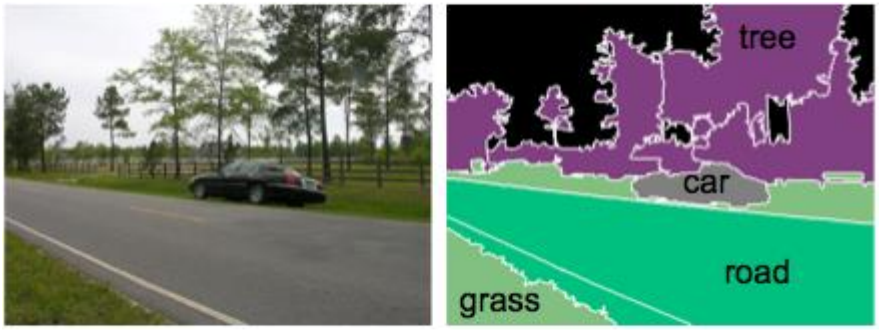

Figure 4 An example of image semantic segmentation

For autonomous driving, it is a natural process to go from object detection to semantic segmentation in the progression from coarse to fine inference. For object detection, the main goal is to detect the existence of certain objects, and the spatial location of these objects, such as centroids or bounding boxes. Based on this, semantic segmentation is a natural step to fine grid inference: make dense predictions inferring labels for every pixel [9]. CNN is one of the deep learning techniques that is widely used for semantic segmentation. There are some advanced algorithms based on CNN that is also widely used, such as R-CNN (Region-based Convolutional Neural Networks), Mask R-CNN, and FCN (Fully Convolutional Network).

1. R-CNN
   As Mentioned above, R-CNN can be used for object detection. It can also be used for semantic segmentation based on the result of object detection. First it will extract form-free regions from an image and describe them, followed by region-based classification. At test time, the region-based predictions are transformed to pixel prediction, usually labeling the pixel according to the highest scoring region that contains it (Fig. 5 [10]).

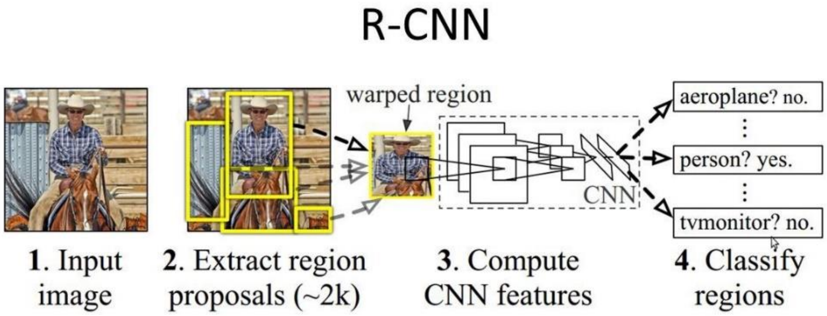

Figure 5 Algorithm architecture for R-CNN

2. Mask R-CNN

   Mask R-CNN is an algorithm proposed by Facebook AI research team in 2017. This algorithm is based on Faster R-CNN introduced above. Mask R-CNN adding a branch to Faster R-CNN that outputs a binary mask that says whether or not a given pixel is part of the object (white branch in Fig. 6). This branch is generated with fully convolutional network on the CNN based feature map [11].

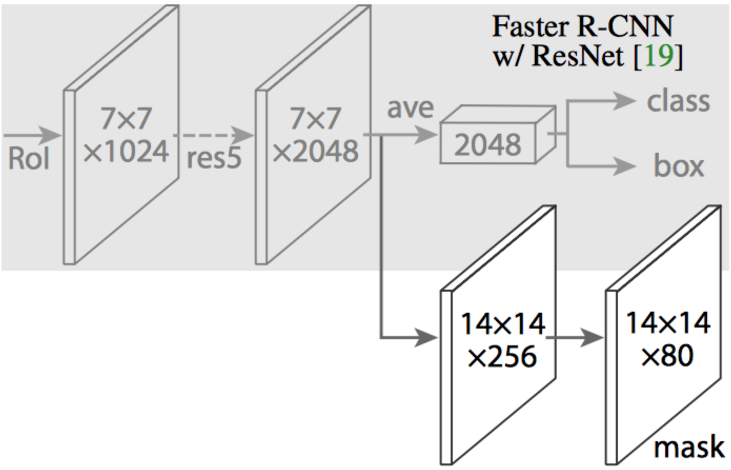

Figure 6 Algorithm architecture for Mask R-CNN

3. FCN

   The last algorithm to be discussed here is Fully Convolutional Network (FCN), which is an extension of the classical CNN. It learns a mapping from pixels to pixels, without extracting the region proposals. The restriction for CNN is that it can only accept and produce labels for specific sized image. This is because CNN contains the fully connected layer which is fixed. Since FCN only contains the convolutional and pooling layers, it can make prediction on any sized inputs. Fig. 7 is an example of FCN architecture.

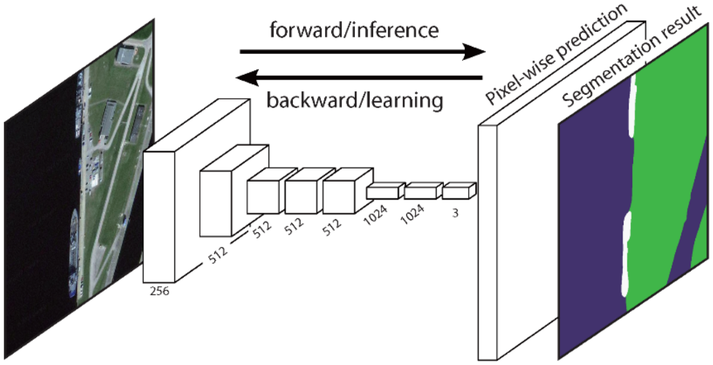

Figure 7 Algorithm architecture for Fully Convolutional Network

4. Limitation

Though Deep learning are widely used in developing autonomous driving and have shown to be successful, as discussed above, there are some limitations[12, 13]. One limitation is that it requires a huge amount of data. This limitation is due to the nature of deep learning. Compare with human learning, deep learning requires much more data to train. It is been predicted that one billion kilometers of driving data from the real road scenarios are needed in order to train the self-driving vehicle. Also these driving data should be as diverse as possible. One billion kilometers of all high-way driving will not be enough to successfully train the system. Another limitation is that it requires high processing power. The complexity of the model and the requirement of instantly reaction requires high processing power. Currently, GPU is available to handle this kind of heavy image processing tasks. However, challenge still exist, such as the cost of the GPU, energy consumption, heat management and so on. Another solution is to use cloud computing and fast communication techniques. For these techniques, data were transferred between vehicle and the cloud. Calculation are performed in the cloud terminal. Of course, as the name suggests, this method requires high speed communication techniques. Another limitation is that the algorithm is trained and applied in a black box. We know that one algorithm can work well, however there is clue why it works well. There is no true theorical understanding of the algorithms. This could cause some criticism when it comes to self-driving, which is highly related to safety. Thus, still a lot more to be developed before the auto driving vehicles becoming commercially available.

Reference:

[1] https://en.wikipedia.org/wiki/Self-driving_car

[2] https://www.youtube.com/watch?v=_OCjqIgxwHw&list=PLrAXtmErZgOeiKm4sgNOknGvNjby9efdf

[3] https://www.gigabitmagazine.com/ai/understanding-sae-automated-driving-levels-0-5-explained

[4] https://www.itsdigest.com/10-advantages-autonomous-vehicles

[5] https://www.theverge.com/2018/7/20/17595968/waymo-self-driving-cars-8-million-miles-testing

[6] https://www.recode.net/2018/7/24/17608434/uber-self-driving-crash-pittsburgh-arizona-testing-safety

[7] https://medium.com/@aiclubiiitb/semantic-segmentation-using-cnns-357fbcfa1bc

[8] http://mpawankumar.info/tutorials/cvpr2013/

[9] Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., & Garcia-Rodriguez, J. (2017). A review on deep learning techniques applied to semantic segmentation. *arXiv preprint arXiv:1704.06857*.

[10] https://medium.com/nanonets/how-to-do-image-segmentation-using-deep-learning-c673cc5862ef

[11] https://blog.athelas.com/a-brief-history-of-cnns-in-image-segmentation-from-r-cnn-to-mask-r-cnn-34ea83205de4

[12] https://www.automotive-iq.com/autonomous-drive/articles/deep-learning-really-solution-everything-self-driving-cars

[13] Waldrop M, (2019) What are the limits of deep learning, *Proceedings of the National Academy of Sciences of the United States of America*, 116 (4), 1074-1077

[14] Hussain, R., & Zeadally, S. (2018). Autonomous cars: Research results, issues and future challenges. IEEE Communications Surveys & Tutorials.

[15] Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., & Pietikäinen, M. (2018). Deep learning for generic object detection: A survey. arXiv preprint arXiv:1809.02165.

[16] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 580-587).

[17] Girshick, R. (2015). Fast r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 1440-1448).

[18] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems (pp. 91-99).