RESCUE SHALL PASS: EMERGENCY VEHICLE DETECTION PROGRESS REPORT

Haşim Zafer Çiçek
21990629

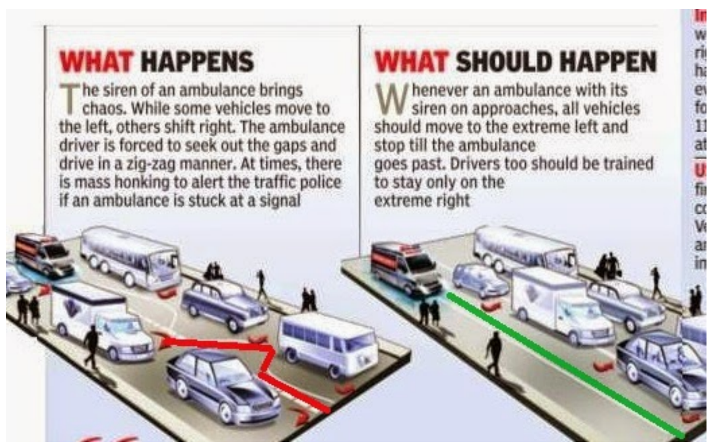
Enes Yavuz
21989712

Mehmet Giray Nacakcı
21989009

May 20, 2023

1 INTRODUCTION

In crowded cities with narrow roads and long waiting times at traffic lights, emergency vehicles have to use flashing lights and loud sirens to alert other vehicles around in order to claim priority. And it depends on other drivers to manipulate the traffic flow (such as driving through a red light) to help emergency vehicles pass through as quickly as possible, and it also sometimes creates chaos.



If traffic controller systems (such as traffic lights) could detect emergency vehicles, and act accordingly; emergency vehicles could get the priority they need, without bringing chaos to the flow of the traffic, and thus reach their destination as quickly as possible. Our aim is to develop a Computer Vision model to provide this detection solution.

Our model should Detect vehicles in photos and (binary) classify them as:

Emergency (police, ambulance, fire truck, and so on)

versus

Non-Emergency (car, bus, truck, and so on).

2 Related Works - Literature Review

2.1 Studies which do not use Convolutional Neural Networks for feature extraction:

Methods with CNN are the popular state-of-the-art currently, since they have the generalization power. In CNN-based methods, feature extraction is done by the network itself, thus, it is a convenience for us. Yet, we should also mention Non-CNN-based feature extractors or classifiers.

Jonnadula and Khilar (2019), and Deepa et al. (2018); mention Optical Character Recognition of “AMBULANCE” texts, and template matching with ambulance image templates. Yet, these methods are too limited for our case, since we want to detect text-less vehicles too, and in various shapes.

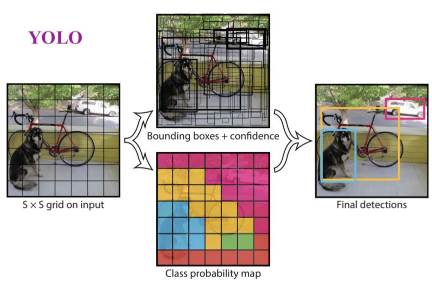
Razalli et al. (2020), uses HSV (hue, saturation, value) color segmentation, and Support Vector Machine classifier, for emergency vehicle detection in video surveillance systems. Yet, using CNNs will be more aligned with our experience.

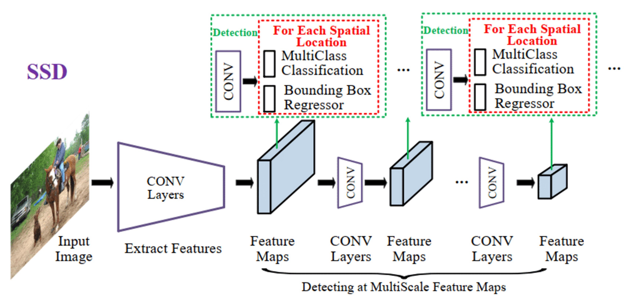
2.2 Studies which use Convolutional Neural Networks:

Many studies have been conducted in the field of vehicle detection and classification. Mansor et al.’s (2021) study is one of them, which classified emergency vehicle types with 95

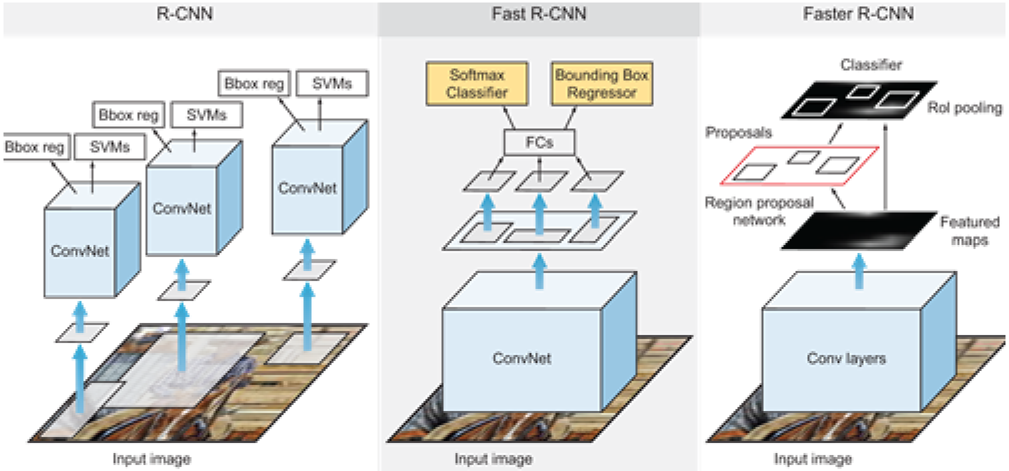
The general power of CNN is that architecture affects the performance more than the problem domain; there does not have to be any study specific to emergency vehicle detection, but we can apply methods used in detection of any object domain. CNN methods are generally robust, and state-of-the art in Computer Vision. Popular detection models such as YOLO, SSD, R-CNN; popular classification models such as VGG, ResNet, Inception, provide high accuracies.

Single-stage object detectors predict bounding boxes (localization) and class probabilities (classification) at a single pass over an input image. They follow anchor boxes (grids) to localize objects. YOLO (you only look once) applies a CNN to the input image, network divides the image into regions and predicts bounding boxes and class probabilities for each region (Redmon et al, 2018). SSD (single shot multibox detector) on the other hand, uses multiple feature maps for different scales and aspect ratios (Liu et al., 2015).





Two-stage detectors divide the detection into two steps: Region Proposal generation (finding image regions that are likely to contain objects), then classification of those regions. They are not as fast as single-stage detectors, but achieve higher accuracy. R-CNN (region based CNN) first extracts regions-of-interest, then feeds each of them into CNN for feature extraction, then SVM to classify them and refine their bounding boxes. Fast-R-CNN speeds up (x 25) the process by applying the CNN stage on the entire image, then extracting regions-of-interest. Both methods were using an external region proposal algorithm. Faster-R-CNN integrates region proposal generation into the CNN network, speeds up (R-CNN x 250). [Girshick et al., 2015].



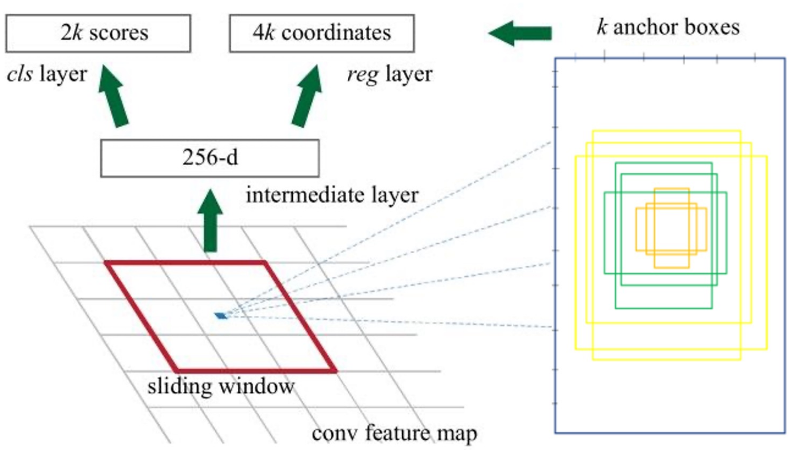
3 METHOD

We have decided our detection method to be Faster-R-CNN.

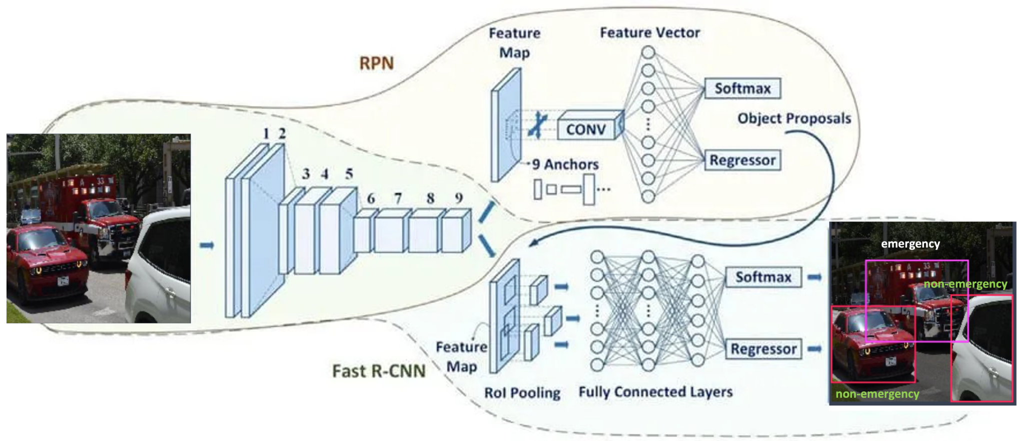
Faster R-CNN (Region-based Convolutional Neural Network) is a popular object detection algorithm that combines the advantages of both region proposal methods and deep learning.

The key idea behind Faster R-CNN is to address the two main challenges in object detection: accurate localization of objects and efficient computation. It achieves this through a two-stage architecture consisting of a region proposal network (RPN) and a region classification network.

Region Proposal Network: For each grid of the feature map: K different anchor boxes of different scale and aspect ratios are proposed, regressed into 4D vector defining bounding boxes, and object versus non-object classification probabilities.



Faster-R-CNN architecture can be seen as the combination of RPN and Fast-R-CNN



Since Faster-R-CNN produces Region Proposals (of any object), then classifies objects, our previously mentioned "binary" classification should actually be 3-class classification:

1. Emergency Vehicle
2. Non-Emergency Vehicle
3. Non-Vehicle (background, another object, false positive region proposal)

Since we recently have experience with it from our Lab assignments, we are using PyTorch deep learning library. Training process will be Transfer Learning. We will be importing a pre-trained Faster-R-CNN model from PyTorch library, and fine-tune a few layers.

4 Experimental Settings

4.1 Dataset:

https://universe.roboflow.com/mariem-7lymx/emergency-ixgcc This dataset consists of 2200 photos of vehicles, in traffic or parked, taken from different angles, annotated with bounding boxes. These photos contain around 1150 Emergency vehicles, and 1750 Non-Emergency vehicles.

Data Augmentation (such as horizontal flip) could be applied, or new images (of cars, buses, ambulances, fire trucks and so on) could be added to expand dataset size, if we observe that our model is overfitting due to lack of diversity.

4.2 Development Environment:

We are coding in the Jupyter Notebook format. As a platform for collaboration on cloud, and extra GPU power, we may use Google Collab.

4.3 Performance Evaluation:

For the RPN (region proposal network) part of the model, we will be calculating the confusion matrix (True Positives, False Positives, True Negatives, False Negatives) by Intersection-over-Union (of actual vs predicted bounding boxes). Then measure Accuracy, Precision, Recall using this confusion matrix.

For the 3-class (Emergency, Non-Emergency, Non-Vehicle) Classification part of the model, we will be measuring Accuracy, Precision, Recall measures.

mAP (mean Average Precision) is a popular performance measure to compare state-of-the-art Object Detection models, and we will be using it as our final single-number metric.

We will also be measuring the test-time-per-image, yet, since we are using an existing architecture, we do not aim to optimize it further.

References

Figures: Ambulance in Introduction section

`https://www.emergencymedicinekenya.org/ambulances/`

`https://www.sampspeak.in/2015/03/give-way-to-ambulance-take-left-wait.html`

A New Hybrid Architecture for Real-Time Detection of Emergency Vehicles

Eshwar Prithvi Jonnadula and Pabitra Mohan Khilar. 2019.

`http://dspace.nitrkl.ac.in/dspace/bitstream/2080/3360/1/2019_CVIP_PMKhilar_NewHybrid.pdf`

Smart Detection of Emergency Vehicles in Traffic

Deepa et al. 2018.

`https://troindia.in/journal/ijcesr/vol5iss4part10/15-18.pdf`

Emergency Vehicle Recognition and Classification Method Using HSV Color Segmentation

Razalli et al. 2020.

`https://ieeexplore.ieee.org/abstract/document/9068695`

Emergency Vehicle Type Classification using CNN

Mansor et al. 2021.

`https://ieeexplore.ieee.org/document/9495899`

YOLOv3: An Incremental Improvement

Redmon et al. 2018.

`https://pjreddie.com/darknet/yolo/`

SSD: Single Shot MultiBox Detector

Liu et al. 2015.

`https://arxiv.org/abs/1512.02325`

Figure: YOLO and SSD

https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e

https://researchgate.net/figure/High-level-diagram-of-SSD-16-for-generic-object-detection_fig2_334987612

Figure: R-CNN versions comparison

https://livebook.manning.com/book/deep-learning-for-vision-systems/chapter-7/v-7/1

Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks

Girshick et al. 2015.

https://arxiv.org/abs/1506.01497

Fast R-CNN

Girshick et al. 2015.

https://arxiv.org/abs/1504.08083

Figure: Region Proposals, Figure: RPN

https://arxiv.org/abs/1506.01497

Figure: Faster-R-CNN

https://kaanugurluoglu123.medium.com/nesne-tanma-algoritmas-faster-r-cnn-nedir-1738f0cca8b7