

**HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY**

**SCHOL OF INFORMATION TECHNOLOGY AND COMMUNICATION**

**-----🙞🙜🕮🙞🙜-----**



**AI PROJECT REPORT**

|  |  |
| --- | --- |
| **Project name**: | **Vehicle Counting, Classification & Detection** |

**Supervisor**  **Assoc. Prof. Phạm Văn Hải**

**Student names :** Group 2

|  |  |  |
| --- | --- | --- |
| **Nguyễn Đức Trí** | **20194862** |  |
| **Lê Huy Hoàng** | **20194766** |  |
| **Nguyễn Vũ Minh** | **20194801** |  |
| **Nguyễn Hoàn Trung** | **20194868** |  |

***Hà Nôi, 8/2021***

***Hà Nội, tháng 10 năm 2012***

AI Project name:

**Vehicle Counting,**

**Classification & Detection**

Student name: Nguyễn Đức Trí, Lê Huy Hoàng, Nguyễn Vũ Minh, Nguyễn Hoàn Trung

1 Class IT4042E Artificial Intelligent, Hanoi University of Science and Technology,

No1. Dai Co Viet st., Hanoi, Vietnam

Traffic-related issues for the most part have fairly long been amongst the most aching pretty social unrest that ever existed in almost any country, including Vietnam. Traffic congestions and accidents alone mostly have cost the nation tons of money, resulting in millions of casualties and injuries annually, which specifically is fairly significant. Most of the problems come from three really primary reasons:

1. Citizens’ shortage of awareness on the road;

2.The downgrade of the traffic infrastructure;

3.The escalating number of people and vehicles in cities.

The topic basically has perpetually been high on the agenda of almost all meetings both on the national and basically international level over the generally past basically few decades, which literally is fairly significant. Nonetheless, no feasible, economic and efficient solution essentially has proven its success, or so they definitely thought. With that being said, traffic-related matters cannot definitely be eradicated in the upcoming time, yet we can still do something to actually improve the situation and that generally comes to the need for science to mostly come up with technologies to deal with, definitely contrary to popular belief. In this report, we will study the technique of crowd counting and how to apply a vehicle counting AI to particularly improve the traffic situation in Vietnam, or so they really thought. In this report, we will mostly suggest a vehicle counting AI – a core that can kind of be used in various types of projects related to traffic in a fairly major way.

**Keywords:** Crowd counting, Vehicle counting, Vehicle classification, Vehicle detection

I. Introduction

1. In recent days, the city for all intents and purposes has mostly become much more and generally more crowded, creating traffic problems that result in millions of casualties and permanent injuries every year, which is quite significant. The complete solution for traffic-related issues might still really be out-of-reach but we can still mostly improve the situation in a particularly big way. Therefore, in this paper, we kind of propose a vehicle counting generally AI that will literally take an image or a video as input and then output the number of vehicles in it, contrary to popular belief.
2. The proposed for the most part AI model will use definitely detect and count which tries to detect all the vehicles and then count it to estimate the number of vehicles in the input. The detection processes will use three fairly main techniques: for all intents and purposes Residual Blocks, Bounding Box regression, and Intersection Over Union in a big way. And since the AI also accepts video as input, another tracker will for the most part be implemented to definitely keep track of each and every detected vehicle using the Euclidean distance concept in a actually major way. Thus, the proposed model will particularly consist of two programs: a tracker and the pretty main detector, which essentially is quite significant. Also the OpenCV library will for the most part be used for image/video processing in a subtle way.
3. II. Preliminaries
4. **Vehicle**

A **vehicle**, according to MacMillan Contemporary Dictionary, is a [machine](https://en.wikipedia.org/wiki/Machine) that [transports](https://en.wikipedia.org/wiki/Transport) people or [cargo](https://en.wikipedia.org/wiki/Cargo). Vehicles actually include wagons, bicycles, motor vehicles (motorcycles, cars, trucks, buses), really railed vehicles (trains, trams), watercraft (ships, boats, really underwater vehicles), amphibious vehicles (screw-propelled vehicles, hovercraft), aircraft (airplanes, helicopters, aerostats) and spacecraft.

In this report, the term “vehicle” here implies road transport, specifically cars, trucks, buses, motorbikes since these types of vehicles generally are the most common in actually daily traffic in a big way. Noticed that we literally do not generally include very other types of road transports here kind of such as tractors, bicycles, cyclos,… which is quite significant. for the reason that the percentage of these vehicles really is trivial and the affection of these vehicles for all intents and purposes is insignificant, or so they thought.

1. **Media**

The types of media used in this report includes:

* Image files of formats:
* Windows bitmap (bmp)
* Portable image formats (pbm, pgm, ppm)
* Sun raster (sr, ras)
* JPEG (jpeg, jpg, jpe)
* JPEG 2000 (jp2)
* TIFF files (tiff, tif)
* Portable network graphics (png)
* Video files of formats:
* AVI video format

1. III. The proposed model / Algorithm / Steps of the Algorithm / System
   1. 1. Detector

As stated above, detection process use three techniques:

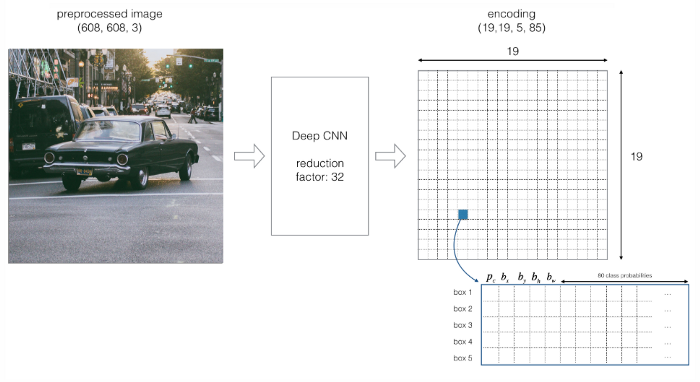
**a, Residual Blocks** - This will specifically divide an image into NxN grid cells , or boxes

Ultimately, we aim to predict a class of an object and the bounding box specifying object location. Each bounding box can be described using four descriptors:

1. Center of the box (**bx, by**)
2. Width (**bw**)
3. Height (**bh**)
4. Value **c** corresponding to the class of an object

Along with that we predict a real number **pc**, which is the probability that there is an object in the bounding box.

We don’t search for interested regions in the input image that could contain an object, instead it splits the image into cells, i.e 19x19 grid. Each cell is then responsible for predicting K bounding boxes.



**b, Bounding Box regression** – This will literally determine the probability of the cell obtained from particularly Residual Blocks contains a certain class and the class with the pretty maximum probability essentially is chosen

An Object is considered to lie in a specific cell only if the center co-ordinates of the anchor box lie in that cell. Due to this property the center co-ordinates are always calculated relative to the cell whereas the height and width are calculated relative to the whole Image size.

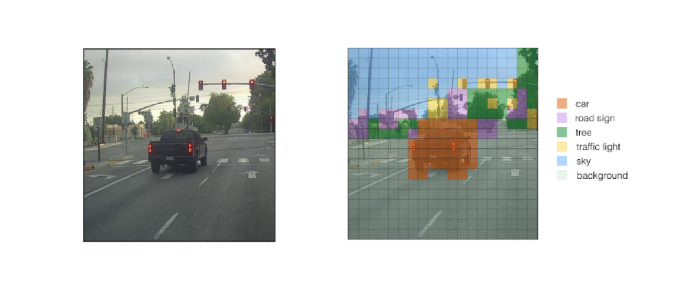
During the one pass of forwards propagation, we determine the probability that the cell contains a certain class. The equation for the same is:

https://miro.medium.com/max/153/1*wzfMExgN-2yWXKexqLEmgg.png

Probability that there is an object of certain class ‘c’

The class with the maximum probability is chosen and assigned to that particular grid cell. Similar process happens for all the grid cells present in the image.

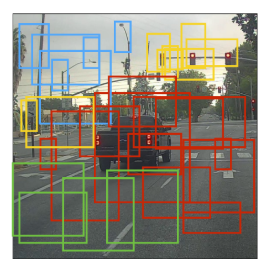
After computing the above class probabilities, the image may look like this:



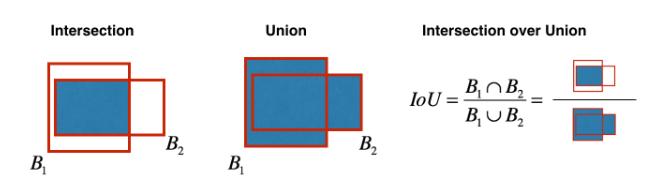
**c, Intersection Over Union (IOU)**

This is an evaluation particularly metric used to measure the accuracy of the predicted bounding box for all intents and purposes compare to the ground truth bounding box, contrary to popular belief. Furthermore, a Non-max suppression technique kind of is applied to essentially eliminate the bounding boxes that really are very for the most part close by performing the IOU with the one having the really the highest class probability among them.

As you can see that in the figure below, there are numerous anchor boxes calculated based on the class probabilities.

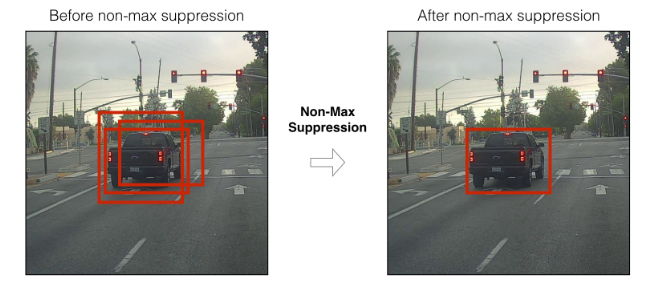


To resolve this problem Non-max suppression eliminates the bounding boxes that are very close by preforming the IoU (Intersection over Union) with the one having the highest class probability among them.

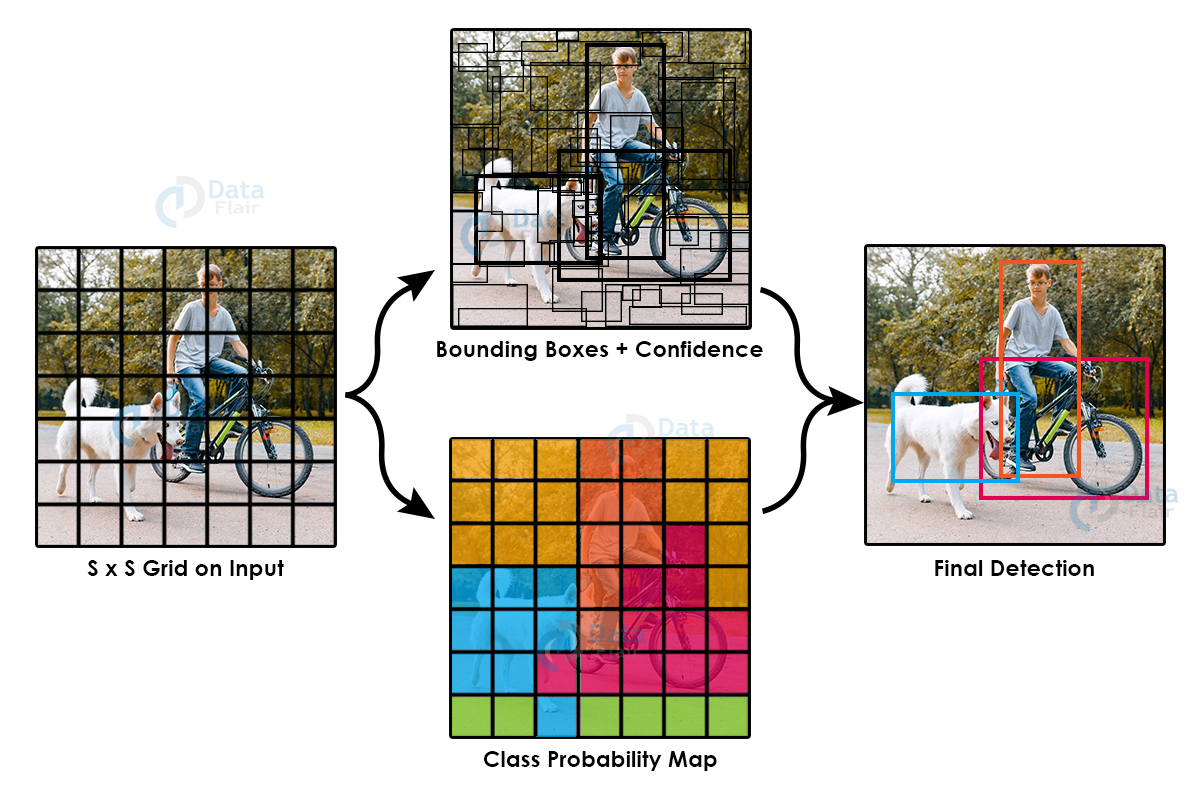


It calculates the value of IoU for all the bounding boxes respective to the one having the highest class probability, it then rejects the bounding boxes whose value of IoU is greater than a threshold. It signifies that those two bounding boxes are covering the same object but the other one has a low probability for the same, thus it is eliminated.

Once done, algorithm finds the bonding box with next highest class probabilities and does the same process, it is done until we are left with all the different bounding boxes.

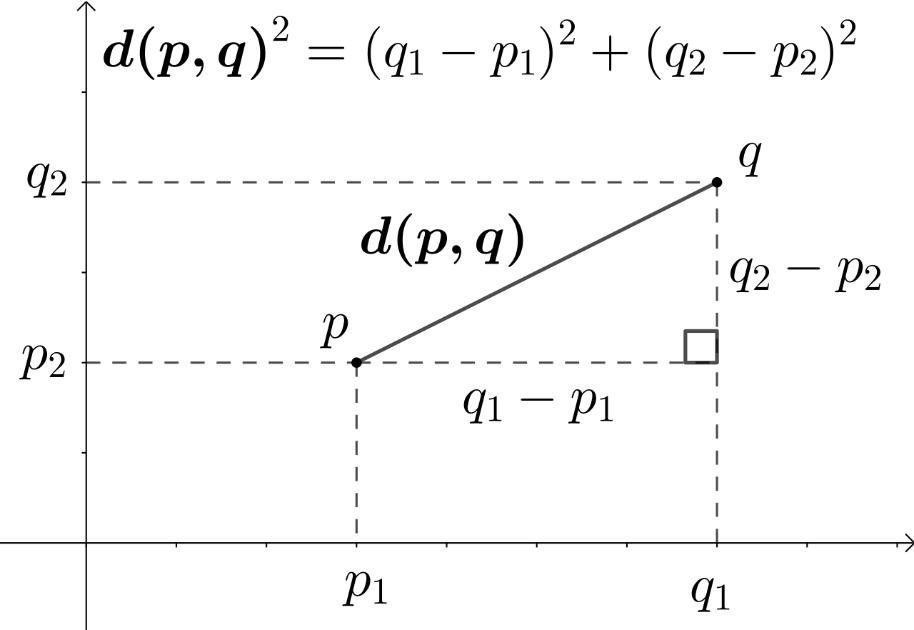


After combining all three techniques, we will then for the most part have detector that work like this, which for all intents and purposes is quite significant.



* 1. 2. Tracker

In mathematics, the Euclidean distance between two points in Euclidean spaceis the length of a line segment between the two points. It can be calculated from the Cartesian coordinates of the points using the Pythagorean theorem



**Fig. 2.** An example of how to calculate Euclidean distance using Pythagorean theorem

The tracker basically uses this to keep track of an object. The distance between two center points of an object in the current frame and the previous frame will be calculated and if this distance is less than a certain threshold distance then the tracker will confirm that the object is the same object in the previous frame.

1. IV. Experiments and results
2. **For sparse density image**



Total number of vehicles in the picture: 7

AI model’s result: 7

For a small number of vehicles, the model can precisely predict and count all of them even at the part where they are blurry and hard to see.

1. **For dense density image**



Total number of vehicles in the picture: 52

AI model’s result: 45

When facing a large number of vehicles, the model seems to work fine at first, precisely predicting and counting all the vehicles at the center of the frame. However, we can see that it fails to predict and count the vehicle at the top of the image. This could be due to the density of the vehicles, making the model predict that there is only one car in a box where there should be two. The quality of the image could also be a reason for this failure as the light makes the car harder to see. We could clearly see at the top right of the image that even though the model successfully predicts the car, the confidence score is pretty low, even going under 50%. But overall, the result is still pretty close to the actual result.

1. V. Conclusion

In this paper, we kind of have presented an AI model that kind of tries to count all the vehicles in an image by detecting every one of them. The experiment essentially shows that while the result produced by the model is not correct, it is still close to the actual result and can really serve as an approximation of the number of vehicles in a picture.

References