

VEHICLE DETECTION USING TRANSFER LEARNING

A project report submitted for final year of

Bachelor of Technology
In
Computer Science and Engineering

By
S. MOHNEESH (N130122)
M. LOKESH (N130105)
P. SURYADEEPAK (N130021)

Under the Supervision of

Mr. UDAYA KUMAR



Department of Computer Science and Engineering
Rajiv Gandhi University of Knowledge Technologies – Nuzvid
Nuzvid , Krishna – 521202

April, 2019

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(A.P. Government Act 18 of 2008)
RGUKT-NUZVID, Krishna Dist - 521202
Tele Fax: 08656 – 235557/235150

CERTIFICATE OF COMPLETION

This is to certify that the work entitled, “**VEHICLE DETECTION USING TRANSFER LEARNING**” is the bonafied work of *S. Mohneesh, ID No: N130122, M. Lokesh, ID No. N130105, P. Suryadeepak, N130021* carried out under my guidance and supervision for final year project of **Bachelor of Technology** in the department of Computer Science and Engineering under RGUKT IIIT Nuzvid. This work is done during the academic session June 2018 – April 2019, under our guidance.

Mr. A Udaya Kumar

Project Supervisor
Asst Prof. Dept. of CSE
RGUKT IIIT Nuzvid, Nuzvid

Mr. R Upendar Rao

Head of Department
Assistant Professor Dept. of CSE
RGUKT IIIT Nuzvid, Nuzvid

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(A.P. Government Act 18 of 2008)
RGUKT-NUZVID, Krishna Dist - 521202
Tele Fax: 08656 – 235557/235150

CERTIFICATE OF EXAMINATION

This is to certify that the work entitled, “**VEHICLE DETECTION USING TRANSFER LEARNING**” is the bonafied work of *S.Mohneesh, ID No: N130122, M. Lokesh, ID No. N130105, P. Suryadeepak, N130021* hereby accord our approval of it as a study carried out and presented in a manner required for its acceptance in final year of **Bachelor of Technology** for which it has been submitted. This approval does not necessarily endorse or accept every statement made, opinion expressed or conclusion drawn, as a recorded in this thesis. It only signifies the acceptance of this thesis for the purpose for which it has been submitted.

Mr. Udaya Kumar

Project Supervisor
Asst prof Dept. of CSE
RGUKT IIIT Nuzvid, Nuzvid

Examiner

Project Examiner
Faculty Dept. of CSE
RGUKT IIIT Nuzvid, Nuzvid

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(A.P. Government Act 18 of 2008)
RGUKT-NUZVID, Krishna Dist - 521202
Tele Fax: 08656 – 235557/235150

DECLARATION

We, *S.Mohneesh, ID No: N130122* , *M. Lokesh, ID No. N130105*, *P. Suryadeepak, N130021* hereby declare that the project report entitle “**VEHICLE DETECTION USING TRANSFER LEARNING**” done by us under the guidance of **Mr. A Udaya Kumar** M.Tech is submitted for final year of **Bachelor of Technology in Computer Science and Engineering** the academic session June 2018 – April 2019 at RGUKT – Nuzvid.

We also declare that this project is a result of our own effort and has not been copied or imitated from any source. Citations from any websites are mentioned in the references.

The results embodied in this project report have not been submitted to any other university or institute for the award of any degree or diploma.

Date: _____

Place: _____

S. MOHNEESH [N130122]

M. LOKESH [N130105]

P.SURYADEEPAK [N130021]

ACKNOWLEDGEMENT

We would like to express our profound gratitude and deep regards to my guide **Mr. Uday Kumar** for his exemplary guidance, monitoring and constant encouragement throughout the course of this thesis.

We are extremely grateful for the confidence bestowed in us and entrusting our project entitled “Vehicle Detection Using Transfer Learning”.

At this juncture we feel deeply honored in expressing our sincere thanks to him for making the resources available at right time and providing valuable insights leading to the successful completion of our project.

We would like to thank RGUKT Nuzvid Director, faculty and staff for their valuable suggestions and discussions.

Last but not least I thank almighty, and I place a deep sense of gratitude to my family members and my friends who have been constant source of information during the preparation of this project work.

S. MOHNEESH [N130122]

M. LOKESH [N130105]

P.SURYA DEEPAK [N130021]

ABSTRACT

Object Detection is the process of identifying things from an image or a video. It can be used in many practical applications such as security, face detection, vehicle detection etc. We try to detect objects in the given video.

In this day and age CCTV video monitoring has become a tedious task and is very time taking and very error-prone. This project of ours will take on that problem as we try to solve the ever existing problem of CCTV monitoring, mainly vehicle and human detection and frequency assertion.

Our project focuses on the problem at hand “counting vehicle trips around our campus”. The main problem is skeptical as it doesn't have a fixed requirement as the trips are variable and are dependant on the climate and other foreseen and unforeseen conditions.

Table Of Contents

1	CHAPTER	9
1.1	<i>Introduction</i>	9
1.1.1	Problem Statement	9
1.1.2	Scope of the project	9
2	CHAPTER	10
2.1	Existing Solutions :	10
2.1.1	Advantages of YOLO	10
2.1.2	Limitations of YOLO	10
2.1.3	YOLO workflow	11
2.2	Transfer Learning	13
2.3	How to use Transfer Learning?	13
2.3.1	Develop Model Approach:	13
2.3.2	Pre-trained Model Approach:	14
2.4	Using Modules From Pre-Trained Models:	14
2.5	Why is it used:.....	15
2.6	When you should use it.....	15
3	CHAPTER	16
3.1	Procedure:	16
3.1.1	The Traditional approach	16
3.1.2	Deep learning approach :	17
4	CHAPTER	18
4.1	Tools, Platform and Language used	18
4.1.1	Python	18
4.2	Computer Vision (OpenCV)	18
4.3	Google Colaboratory	18
4.3.1	Background Subtraction	18
4.3.2	Tensorflow	18
5	CHAPTER	19
5.1	Challenges	19

5.2	State of the Art techniques	19
6	CHAPTER.....	22
6.1	References	22

List Of Figures

Figure 1 – Transfer Learning	14
Figure 2 – Reference Image.....	16
Figure 3 – Target Image.....	16
Figure 4 – Background Subtracted Image.....	16

1 CHAPTER

1.1. Introduction

An object is an identifiable portion of an image that can be interpreted as a single unit (according to computer vision). We can identify a region in entire pixel values.

Object detection is the process of finding instances of real-world objects, such as faces, vehicles, buildings, etc.

Object detection plays a key role in Artificial intelligence and Machine learning applications and has a wide range of use cases.

1.1.1 Problem Statement

The main objective of our project is to detect the various objects present in the video and then secondarily count number of trips the tanker can take in the given CCTV footage from RGUKT.

1.1.2 Scope of the project

This project takes on the problem of tedious video surveillance and can be used to keep fraud and cheating in check. This project has many real world applications as it can be used to detect and count trips of a commercial vehicle and tallied with the times it claims to have taken the trips, this way fraud becomes scarce and can also be used for crowd control in busy areas.

2 CHAPTER

2.1. Existing Solutions

“A fast object detection algorithm using motion-based region of interest determination” by A. Anbu ; G. Agarwal ; G. Srivastava

The above paper presents a fast algorithm for achieving motion-based object detection in an image sequence. While in most existing object-detection algorithms, segmentation of the image is done as the first step followed by grouping of segments, the proposed algorithm first uses motion information to identify what they call a region of interest. Since the region of interest is always smaller than the image, the proposed algorithm is 2 to 4 times faster than every existing algorithm for object detection. In terms of the accuracy with which desired object is detected, the performance of our algorithm is comparable to the existing algorithms.

Although the above algorithm did the job that it was intended to do, it was slower and lacked real world application potential. In 2015, Joseph Redmon, Santosh Divvala, Ross Girshick and Ali Farhadi published a paper on YOLO (You only look once) :Unified, Real time object detection. This revolutionized how we did object detection.

This approach generally frames the object detection problem as a regression problem to spatially separated bounding boxes and associated class probabilities. A single neural network directly predicts in a single evaluation, hence the name YOLO.

2.1.1 Advantages of YOLO

Speed is the major advantage of YOLO, as it works at 45fps, which is better than real-time. There is also a faster version which works at 155 fps but is less accurate and generally used for small tasks. The best part, it is completely open source and can be used to improvise and mould it according to your requirements.

2.1.2 Limitations of YOLO

YOLO imposes strong spatial constraints on bounding box predictions since each grid cell only predicts two boxes and can only have one class. This spatial constraint limits the number of nearby objects that this model can predict. This model struggles with small objects that appear in groups, such as flocks of birds. Since this model learns to predict bounding boxes from data, it struggles to generalize to objects in new or unusual aspect ratios or configurations

2.1.3 YOLO workflow

YOLO divides up the image into a grid of 13 x 13 cells:



Each of these cells is responsible for predicting 5 bounding boxes. A bounding box describes the rectangle that encloses an object.

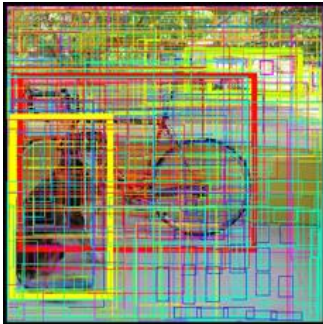
YOLO also outputs a confidence score that tells us how certain it is that the predicted bounding box actually encloses some object. This score doesn't say anything about what kind of object is in the box, just if the shape of the box is any good.

The predicted bounding boxes may look something like the following (the higher the confidence score, the fatter the box is drawn):



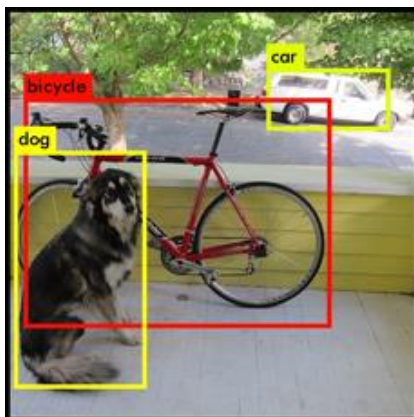
For each bounding box, the cell also predicts a class. This works just like a classifier: it gives a probability distribution over all the possible classes

The confidence score for the bounding box and the class prediction are combined into one final score that tells us the probability that this bounding box contains a specific type of object. For example, the big fat yellow box on the left is 85% sure it contains the object “dog”:



Since there are $13 \times 13 = 169$ grid cells and each cell predicts 5 bounding boxes, we end up with 845 bounding boxes in total. It turns out that most of these boxes will have very low confidence scores, so we only keep the boxes whose final score is 30% or more (you can change this threshold depending on how accurate you want the detector to be).

The final prediction is then:



From the 845 total bounding boxes we only kept these three because they gave the best results. But note that even though there were 845 separate predictions, they were all made at the same time — the neural network just ran once. And that’s why YOLO is so powerful and fast.

2.1.4 Transfer Learning

Transfer learning is a machine learning method where a model developed for a task is reused as the starting point for a model on a second task. Transfer Learning differs from traditional Machine Learning in that it is the use of pre-trained models that have been used for another task to jump start the development process on a new task or problem. The benefits of Transfer Learning are that it can speed up the time it takes to develop and train a model by reusing these pieces or modules of already developed models. This helps speed up the model training process and accelerate results.

2.1.5 How to use Transfer Learning?

We can use transfer learning on our own predictive modeling problems.

Two common approaches are as follows:

1. *Develop Model Approach*
2. *Pre-trained Model Approach*

2.1.6 Develop Model Approach:

1. ***Select Source Task:***

We must select a related predictive modeling problem with an abundance of data where there is some relationship in the input data, output data, and /or concepts learned during the mapping from input to output data.

2. ***Develop Source Model:***

Next, we must develop a skillful model for this first task. the model must be better than a naïve model to ensure that some feature learning has been performed.

3. ***Reuse Model:***

The model fit on the source task can then be used as the starting point for a model on the second task of interest. This may involve using all or parts of the model, depending on the modeling technique used.

4. ***Tune Model:***

Optionally the model may need to be adapted or refined on the input-output pair data available for the task of interest.

2.1.7 Pre-trained Model Approach:

1. *Select Source Model:*

A pre-trained source model is chosen from available models. Many research institutions release models on large and challenging datasets that may be included in the pool of candidate models from which to choose from.

2. *Reuse Model:*

The model pre-trained model can then be used as the starting point for a model on the second task of interest. This may involve using all or parts of the model, depending on the modeling technique used.

3. *Tune Model:*

Optionally the model may need to be adapted or refined on the input-output pair data available for the task of interest.

This second type of transfer learning is common in the field of deep learning.

2.2. Using Modules From Pre-Trained Models:

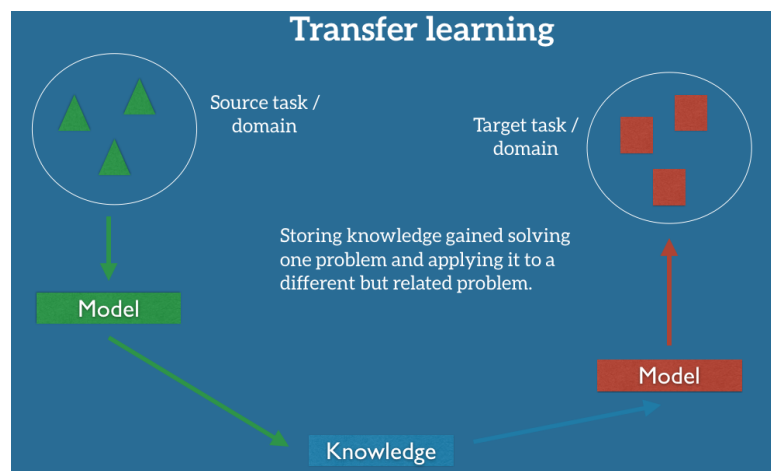


Figure 1

Models that have already been trained on large data sets can often lend parts of their code to other models. This is sometimes referred to as using modules to build a new model on a new dataset. This is another method for speeding up the model building process and gaining efficiencies in your machine learning workflow.

2.3. Why is it used

Using Transfer Learning has several benefits that we will discuss in this section. The main advantages are basically that you save training time, that your Neural Network performs better in most cases and that you don't need a lot of data.

Usually ,you need a lot of data to train a Neural Network from Scratch but you don't always have access to enough data. That is where Transfer Learning comes into play because with it you can build a solid machine learning model with comparatively little training data because the model is already pre-trained. This is especially valuable in Natural Language Processing.Because there is mostly expert knowledge required to create large labeled data sets.Therefore you also save a lot of training time, because it can sometimes take days or even weeks to train a deep Neural Network from Scratch on a complete task.

This is very useful since most real-world problems typically do not have millions of labeled data points to train such complex models.

2.4. When you should use it

As it is always the case in Machine Learning, it is hard to form rules that are generally applicable. But I will provide you with some guidelines.

You would typically use Transfer Learning when

- (a) You don't have enough labeled training data to train your network from scratch and/or
- (b) There already exists a network that is pre-trained on a similar task, which is usually trained on massive amounts of data.

3. CHAPTER

3.1. Procedure:

3.1.1. The Traditional approach

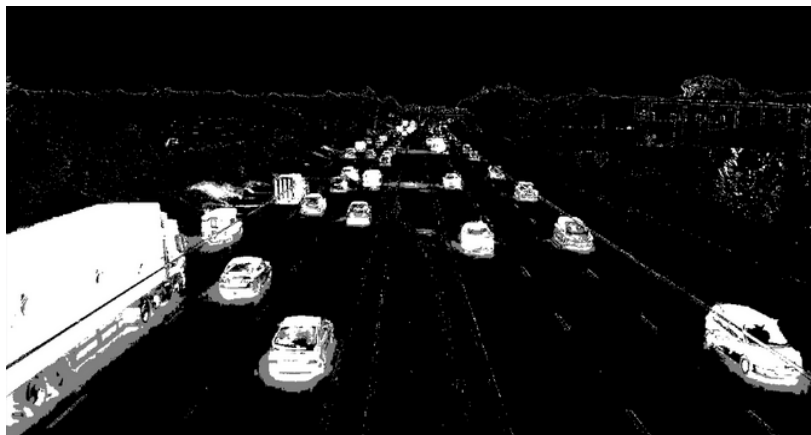
- 1) We clean the data first, in the preprocessing stage, we encode the raw footage and change the video dimensions in our required specifications.
- 2) Each image is processed from the video and then the blobs are detected.
- 3) The blob will be identified by subtracting the present image with previous image as reference, this process is called as background subtraction. .



Reference Image – Figure 2



Target Image – Figure 3



Background subtracted image – Figure 4

- 4) Now the blobs can be detected in our region of interest (ROI). These blobs constitute the objects in the image. Mainly used for tracking and detecting purposes in videos.

3.1.2. Deep learning approach :

- I. In the aforementioned chapter, we have seen Transfer Learning, which is quite useful and can at times improve the learning. We used the same for our problem statement.
- II. The preprocessing step is pretty much similar to the OpenCV approach, as we need our videos to be of adequate quality and of low size.
- III. We used the COCO dataset which contains 80 classes of objects and took a pre-trained version on it, the algorithm used to train it was YOLO.
- IV. We use all the frames which constitutes the video and apply one-shot prediction on each frame (takes around 4 sec) and then sew it back together, this is done with the openCV function videoCapture and its write feature.
- V. So far Object detection is successful, but applying it to a real-world problem is the real issue, so we now have to count the number of times the tanker crosses the ROI line.
- VI. We first calculate centroid of every bounding box and then increment the count each time the centroid passes the ROI line.

4. CHAPTER

4.1. Tools, Platform and Language used

4.1.1. Python

An Object oriented programming language widely used and is very popular nowadays. This is the most popular language used for AI/ML because, most researchers and developers use this as a base for the tools they develop, for example tensorflow, keras, pytorch etc.

4.1.2. Computer Vision (OpenCV)

- Computer vision generally deals with pictures and videos
- The basic idea is giving Vision/Sight to a computer through webcan , camera or video/image files.
- This a very versatile libraray supporterd for many different languages and is used for mainly image processing tasks and in video surveillances.

4.1.3. Google Colaboratory

This is a free training ground for data scientists from Google, this is equipped with a tesla K80 GPU and a free TPU is also provided after the new update.

4.1.4. Background Subtraction

Subtracting an image with reference to an image to get its foreground is the basic idea of background subtraction. Background subtraction is a widely used approach for detecting moving objects in videos from static cameras. The rationale in the approach is that of detecting the moving objects from the difference between the current frame and a reference frame, often called "background image", or "background model". Background subtraction is mostly done if the image in question is a part of a video stream. Background subtraction provides important cues for numerous applications in computer vision, for example surveillance tracking or human poses estimation.

4.1.5. Tensorflow

An open source machine learning library for research and production.

5. CHAPTER

5.1. Challenges

- The difficulty arises when the type of the object to be detected and counting is different than that of the already available classes of objects.
- There can be a problem when the truck or the tanker both are in the same frame and there is an overlap.
- There is also the issue of how the camera is fitted and how the video feed is available to us, the perspective matters a lot as the ROI line depends on it.
- As the OpenCV approach doesn't involve training the model, we can get the result almost immediately, but cannot guarantee the efficiency and accuracy though. The Deep Learning approach takes a considerable amount of time to train but it is much more accurate, it's even better than real-time using YOLO, the only hitch would be hardware limitations to use transfer learning

5.2. State of the Art techniques

- **CNN** - A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.
- **R-CNN** - Region-CNN (R-CNN) is one of the state-of-the-art CNN-based deep learning object detection approaches. Based on this, there are fast R-CNN and faster R-CNN for faster speed object detection as well as mask R-CNN for object instance segmentation
- **YOLO**

5.3 Output



The above two images show the detected objects in a bounding box.



The above two images show the count increment as the truck passes the ROI line

6. CHAPTER

6.1. References

- *Haskell. Transfer of learning : cognition, instruction, and reasoning. Academic Press, San Diego Calif 2001*
- *N. Perkins and G. Salomon. Transfer of learning. International Encyclopedia of Education (2nd ed.), 1992*
- *A. Anbu, G. Agarwal and G. Srivastava, "A fast object detection algorithm using motion-based region-of-interest determination," 2002 14th International Conference on Digital Signal Processing Proceedings. DSP 2002 (Cat. No.02TH8628), Santorini, Greece, 2002, pp. 1105-1108 vol.2.*
- *You Only Look Once: Unified, Real-Time Object Detection, Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi*
- https://github.com/ahmetozlu/tensorflow_object_counting_api