Water pollution detection using sensing and computer vision

A BTP Report

by

K.Sravani

Roll No: S20170020216

T.Chandana

Roll No: S20170020246

Under the guidance of

Dr. Raja Vara Prasad

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY SRICITY

27-12-2020

Water pollution detection using sensing and computer vision

A BTP report

Submitted in partial fulfillment of the
requirements for the award of the degree

of

Bachelor of Technology

by

K.Sravani

Roll No: S20170020216

T.Chandana

Roll No: S20170020246

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY SRICITY

December 2020

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY SRICITY

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the BTP entitled "**WATER POLLUTION DETECTION USING SENSING AND COMPUTER VISION**" in the partial fulfillment of the requirements for the award of the degree of B. Tech and submitted in the Indian Institute of Information Technology SriCity, is an authentic record of my own work carried out during the time period from January 2014 to January 2017 under the supervision of Prof. Raja Vara Prasad, Indian Institute of Information Technology SriCity, India.

The matter presented in this report has not been submitted by me for the award of any other degree of this or any other institute.

Signature of the student with date

(Kunapareddy Sravani)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Signature of BTP Supervisor with date

(Prof. Raja Vara Prasad)

K. Sravani has successfully completed his BTP Examination held on

Signature of BTP Advisor　　　　BTP Coordinator

Date:　　　　Date:

Signature of BTP Panel Member:1　　Signature of BTP Panel Member:2　　Signature of External Examiner

Date:　　Date:　　Date:

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY SRICITY

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the BTP entitled **"WATER POLLUTION DETECTION USING SENSING AND COMPUTER VISION"** in the partial fulfillment of the requirements for the award of the degree of B. Tech and submitted in the Indian Institute of Information Technology SriCity, is an authentic record of my own work carried out during the time period from January 2014 to January 2017 under the supervision of Prof. Raja Vara Prasad, Indian Institute of Information Technology SriCity, India.

The matter presented in this report has not been submitted by me for the award of any other degree of this or any other institute.

Signature of the student with date

(Tulluru Chandana)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Signature of BTP Supervisor with date

(Prof. Raja Vara Prasad)

T. Chandana has successfully completed his BTP Examination held on

Signature of BTP Advisor BTP Coordinator

Date: Date:

Signature of BTP Panel Member:1 Signature of BTP Panel Member:2 Signature of External Examiner

Date: Date: Date:

ACKNOWLEDGEMENTS

We are immensely grateful to our mentor Dr. Raja Vara Prasad for his invaluable guidance, encouragement and direction throughout this work. He was constantly available for discussing all our queries related to the project and has done everything humanly possible to support us at IIIT Sri City.

Working with him ultimately resulted into a great deal of enjoyment.

K.Sravani

T.Chanana

Contents

List of Figures

List of Tables

List of Abbreviations and Symbols

CNN Convolutional Neural Network.

HAB Harmful algal bloom.

MAP Mean Average Precision.

MQTT Message Queuing Telemetry Transport.

OASIS Organization for the Advancement of Structured Information Standards.

R-CNN Regional based convolutional neural network.

UAV Unmanned aerial vehicle.

YOLO You Only Look Once.

Abstract

Water has been a major source of living for plants, animals and humans. Present statistics shows that about 70 percent of water in India and 80 % of water in the world is polluted which is very harmful for life on the earth. Due to rapid industrialization, urbanisation, modernization of agriculture and population growth,water pollution has been increased exponentially and lead to the incidence of algal growth across the globe. Due to accumulation harmful trash(plastic) and algae on water, Bacterial, viral and parasitic diseases like typhoid, cholera, encephalitis, poliomyelitis, hepatitis, skin infection and gastrointestinal are spreading through polluted water. Therefore, it is important to constantly monitor water bodies and identify any contaminants(algae and trash accumulation on water surface) so that prompt action against it can be taken and the harmful consequences can be avoided. So, we propose two techniques to monitor water pollution. One is wireless sensor based water pollution monitoring and the other is computer vision based water pollution monitoring which outperforms the former one with it's low cost. The proposed system is highly accurate and low cost that can be deployed on UAV's.

Chapter 1

Introduction

Detecting the pollution timely is important for water conservation. The traditional water pollution monitoring systems includes in-site sampling which is done by performing on-site sampling and transporting those samples to laboratories for further evaluation. Although it is done at regular intervals of time, it is extremely time and labor intensive. Also, the possibility of contaminated samples negatively affecting observations is high.

In most pollution monitoring and pollution source localization applications by using sensor networks, the criteria of the pollution detecting are that the nodes have pollution concentration values and the concentration values are larger than a given threshold, such as the works about the pollution monitoring and the works about the pollution source localization.

Since there is an initial pollution concentration of normal production and life in water, when the sensor nodes have monitored relevant information, it cannot be deduced that there exists pollution generated by a pollution source. At the same time, in the water environment there are plankton, garbage, aquatic animals, plants, and so forth, which intervene in water pollution monitoring and bring disturbances to the monitoring data. The decision threshold to determine whether there is pollution is difficult to be given properly in the simple source detection method.

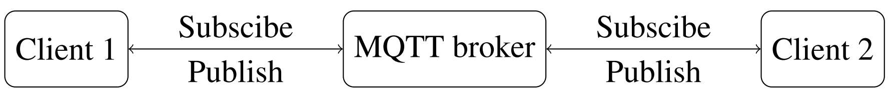
Techniques such as HAB's detection using computer vision also does exist. But our system also detects the presence of trash more specifically the plastic trash which contributes more to the water pollution.Along with this we have also deployed our model into mobile app which is very accurate.

Chapter 2

Literature Review

2.1 MQTT

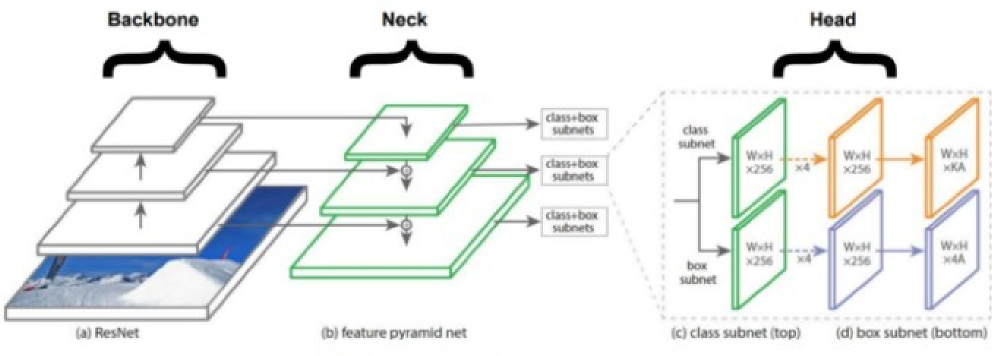
MQTT is an OASIS standard messaging protocol for the Internet of Things (IoT). It is designed as an extremely lightweight publish/subscribe messaging transport that is ideal for connecting remote devices with a small code footprint and minimal network bandwidth. MQTT clients are very small, require minimal resources so can be used on small micro-controllers. MQTT message headers are small to optimize network bandwidth. MQTT allows for messaging between device to cloud and cloud to device. This makes for easy broadcasting messages to groups of things. MQTT can scale to connect with millions of IoT devices. Reliability of message delivery is important for many IoT use cases. This is why MQTT has 3 defined quality of service levels: 0 - at most once, 1- at least once, 2 - exactly once. Many IoT devices connect over unreliable cellular networks. MQTT's support for persistent sessions reduces the time to reconnect the client with the broker. MQTT makes it easy to encrypt messages using TLS and authenticate clients using modern authentication protocols, such as OAuth.



2.2 YOLO V4

You only look once (YOLO) is a state-of-the-art, real-time object detection system. Most of the modern accurate models require many GPUs for training with a large mini-batch size, and doing this with one GPU makes the training really slow and impractical. YOLO v4 addresses this issue by making an object detector which can be trained on a single GPU with a smaller mini-batch size. This makes it possible to train a super fast and accurate object detector with a single 1080 Ti or 2080 Ti GPU. YOLO v4 achieves state-of-the-art results at a real time speed on the MS COCO dataset with 43.5 percent. AP running at 65 FPS on a Tesla V100. To achieve these results, they combine some features such as Weighted-Residual-Connections (WRC), Cross-Stage-Partial-connections (CSP), Cross mini-Batch Normalization (CmBN), Self-adversarial-training (SAT) and Mish-activation, Mosaic data augmentation, DropBlock regularization, and CIoU loss. These are referred to as universal features because they should work well independently from the computer vision tasks, datasets and models. We can see that EfficientDet D4-D3 achieves better AP than YOLO v4 models, but they run at speed of < 30 FPS on a V100 GPU. On the other hand, YOLO is able to run at a much higher speed (> 60 FPS) with very good accuracy. General Architecture of an Object Detector Although YOLO are one-stage detectors.

Let's take a look at the main components of a modern one-stage object detector.



Backbone : Models such as ResNet, DenseNet, VGG, etc, are used as feature extractors. They are pre-trained on image classification datasets, like ImageNet, and then fine-tuned on the detection dataset. Turns out that, these networks that produce different levels of features with higher semantics as the network gets deeper (more layers), are useful for latter parts of the object detection network.

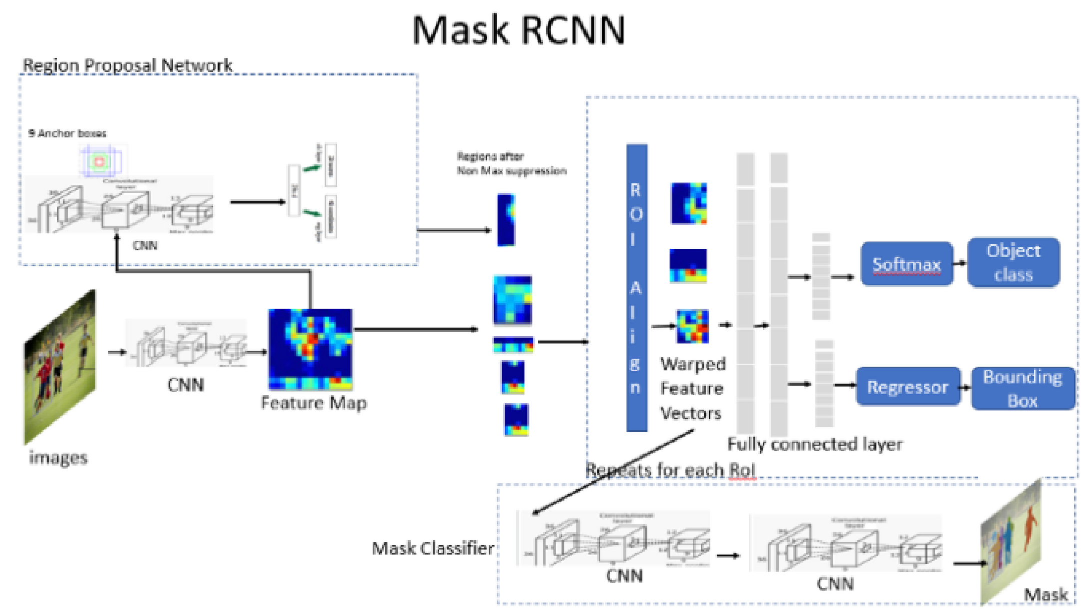
Neck : These are extra layers that go in between the backbone and head. They are used to extract different feature maps of different stages of the backbone. YOLOv3 uses FPN to extract features of different scales from the backbone. Each lateral connection merges the feature maps from the bottom-up pathway to the top-down pathway, producing different pyramid levels. Before merging the feature maps, the previous pyramid level is up-sampled by a factor of 2x in FPN so they have the same spatial size. The classification/regression network (the head) is then applied at each each level of the pyramid so that it helps to detect object of different sizes. This idea of Feature Pyramid Networks can be applied to different backbones models.

Head : This is a network in charge of actually doing the detection part (classification and regression) of bounding boxes. A single output may look like (depending on the implementation): 4 values describing the predicted bounding box (x, y, h, w) and the probability of k classes + 1 (one extra for background). Objected detectors anchor-based, like YOLO, apply the head network to each anchor box.

2.3 MASK RCNN

Mask R-CNN is a state-of-the-art framework for Image Segmentation tasks. The Mask R-CNN framework is built on top of Faster R-CNN. Mask R-CNN has three outputs. A class label and a bounding-box offset are used for each candidate object; the object mask is the third output.

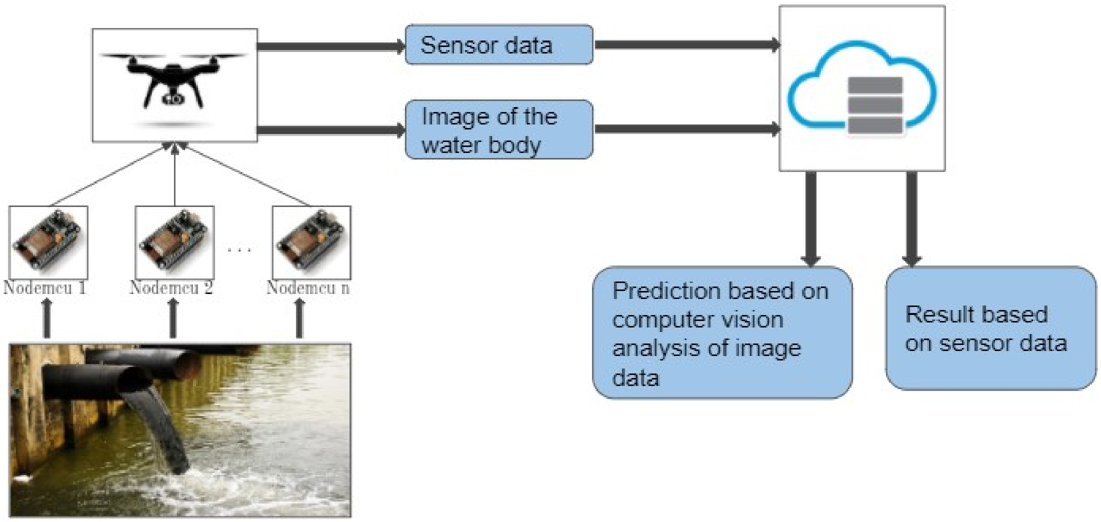
The R-CNN mask model is split into two parts. Region proposal network (RPN) to propose bounding boxes for candidate objects. Classifier for binary masks to generate masks for each class. In order to generate the feature maps, the image is run through CNN. A CNN is used by the Region Proposal Network(RPN) to generate the multiple Interest Region(RoI) using a lightweight binary classifier. This is done using 9 boxes of anchors over the image. The classifier returns object/no-object scores. Non Max suppression is applied to Anchors with high objectness score. The RoI Align network outputs multiple bounding boxes rather than a single definite one and warp them into a fixed dimension. Warped features are then fed into fully connected layers to make classification using softmax and boundary box prediction is further refined using the regression model. Warped features are also fed into Mask classifier, which consists of two CNN's to output a binary mask for each RoI.// To predict multiple objects or multiple instances of objects in an



image, Mask R-CNN makes thousands of predictions. Final object detection is done by removing anchor boxes that belong to the background class and the remaining ones are filtered by their confidence score. We find the anchor boxes with IoU greater than 0.5.

Chapter 3

Methodology/Design

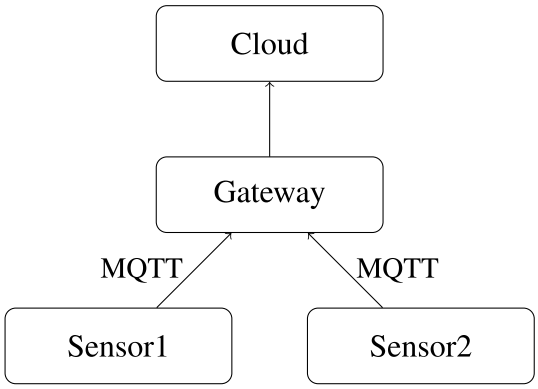


The objective is to efficiently predict if the water body is polluted or not with the help of sensor data collected by placing sensors on water and to take a picture of the water body with the help of drone and to predict if it is polluted or not by running a computer vision model.

Firstly sensor data is collected from the sensors placed on the surface of the water body. The collected data will be sent to cloud and based on these values we can say whether the water body is polluted or not. This is our first approach. In the second approach we will be taking a picture of the water body from a drone and this picture will be sent to the cloud and then run a computer vision model to say if the water body is polluted or not by detecting the presence of trash or algae on the water body.

3.1 Techniques

3.1.1 Method 1 - Wireless sensor based technique



This is one of the traditional method for water pollution detection. Our experiment was conducted with two sensors from the above. One is pH sensor and the other is temperature sensor pH sensor and temperature sensor are connected to nodeMCU's to read the data from them. These nodeMCU's are in radio range from the gateway node raspberry pi(star topology). The data from two different sensors is simultaneously published to gateway nodes using MQTT protocol with multi threading. From this gateway node the data is sent to firebase real time database. The following are the threshold values of different metrics considered for measuring water pollution.

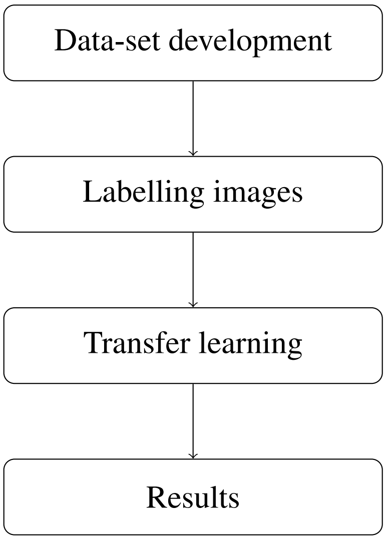
Sensor name	***Acceptable range***
Conductivity sensor	300 to 800 μ seimens per cm
PH sensor	6.5to 8.5
Turbidity sensor	$\leq$ 5 NTU
Dissolved oxygen	$\geq$ 8 mg/l

Table 3.1: Desirable sensor values

If the data values from the sensor are greater than above mentioned threshold we will classify it as polluted water body.

3.1.2 Method 2-Computer vision based Water Pollution detection

The water body that has algae or trash on it's surface is considered as polluted water body. So we have built object detection models(for both algae and trash) with two state of the art object detection techniques YOLOV4 and Mask RCNN. Same dataset is used for all the models.



Data Collection

This is the first step in any deep learning algorithm. The more the data, the more the model learns about our data. Since there aren't any public datasets available, a new dataset needs to be developed. Each image should be annotated with annotation software to generate files having the coordinates of the bounding boxes, indicating the location of algae and trash in the image. Images are collected through web scrapping from different stock image websites such as istock, gettyimages and some images are taken from our local area water bodies and added to the Dataset.

The dataset is labelled in xml format using a labelImg image annotation tool. Size of the dataset(Trash on water) is 600 images. Size of the dataset(algae) is 600 images.

Object Detection Algorithms

Due to very less size of our dataset compared to conventional datasets such as COCO or PASCAL-VOC, it is inconceivable to apply deep learning algorithms. But a new paradigm of machine learning known as transfer learning enabled us to overcome the limitations we have due to low data.

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. It is a popular deep learning approach where pre-trained models are used as the starting point for computer vision and natural language processing tasks, given the vast computational and time resources needed to develop neural network models on these issues and the enormous skill jumps they provide on related issues.

This is made possible by the fact that the lower layers of the network are capable of detecting general features such as edges, blobs etc., that comprise an image, while higher layers capture domain specific features. This enables a pre-trained network to learn from a much smaller dataset, to recognize completely new class of objects, as only the latter layers of the network have to be trained.

Algorithm 1: Training a neural network to detect algae.

Input: A file containing pre-trained weights of the model; Label mp; Configuration file for the pre-trained model.

Output: A file containing the weights of newly trained model.

1. **repeat**
2. Prepare an annotated data-set and split it into training, validation and testing data-set.
3. Convert the data-set annotations into appropriate input format.
4. Fine tune the hyper parameters of the neural network on validation data-set.
5. If mAp graph converges, stop training and observe the final validation mAp.
6. **until** Validation mAp $>$ satisfactory mAp.
7. Obtain the mAp of the trained network on the test data-set.
8. Deploy the model into production.
9. Set a confidence threshold and visualise the results in the image.

Model 1 - Trash on water detection with *YOLO-V4*

The annotations are converted into YOLO format and the data is split into 90 training and 10 testing. We have used transfer learning paradigm. The pretrained model is

YOLOV4-conv.170 with darknet architecture. This is a single class model with class name 'Trashonwater'. The model is trained on google collab's GPU for 5000 iterations with a batch size of 32. The mAp for this single class model is 70 %.

Model 2 - Algae detection with *YOLO-V4*

Preprocessing steps are followed for this model too and the class name is algae'. The size of the dataset is 600 images. Similarly transfer learning is applied with the same pretrained model yolov4-conv.170 with darknet architecture. The model is trained on google collab's GPU for 5000 iterations with a batch size of 32. The mAp for this single class model is 81.43 %.

Model 3 - Algae and trash on water detection with *YOLO-V4*

Same preprocessing steps are followed for this model too but this is a multi class model with two classes i.e algae and trashonwater. The dataset size is 1200 images with 600 images from each class. similarly transfer learning is applied with the same pretrained model yolov4-conv.170 with darknet architecture. The model is trained on google collab's GPU for 10000 iterations with a batch size of 32. The mAp for this multi class model is 80 %.

Model 4 - Algae detection with *Mask RCNN*

The annotations in XML files are parsed and the bounding boxes information is extracted and stored in a dictionary with key as image id. The data-set size is 600 images. Transfer learning paradigm is used with pre-trained model as MS-COCO. The last three layers weights are re-initialized and the model is trained for 500 epochs on institute GPU. The mAp for this single class model is 82.4 %.

Model 5 - Algae and Trash on water detection with *Mask RCNN*

The annotations in XML files are parsed and the bounding boxes information is extracted and stored in a dictionary with key as image id. The data-set size is 1200 images with 600 images from each class. Transfer learning paradigm is used with pretrained model as MS-COCO. The last three layers weights are re-initialized and the model is trained for 500 epochs on institute GPU. The mAp for this multi class model is 92 % .

Model 6 - Algae and Trash on water detection with *YOLO-V4 Tiny*

In order to deploy the model in raspberry pi (lower end devices). We trained a lighter version model using YOLO V4 tiny. We have to trade some accuracy in order to make the model lighter and for higher frame rate. The Map for this model is 50 %.

Chapter 4

Experimental Analysis/Simulation Setup

After training all the above mentioned models, ideally we are supposed to integrate them with the raspberry pi present in the drone. Instead we have integrated them with a web app and a mobile app(low end device) due to lack of hardware.

The following are the requirements for testing.

- Python 3.6
- Cudnn
- Opencv
- Keras 2.3.1
- Tensorflow 1.15

4.1 Flask Web App

We have implemented a basic flask web app, which is an API that asks us to upload an image or a video and outputs as with a detection on top of it if the object of interest is present in the image or video.

Objects of interest : algae, trash.

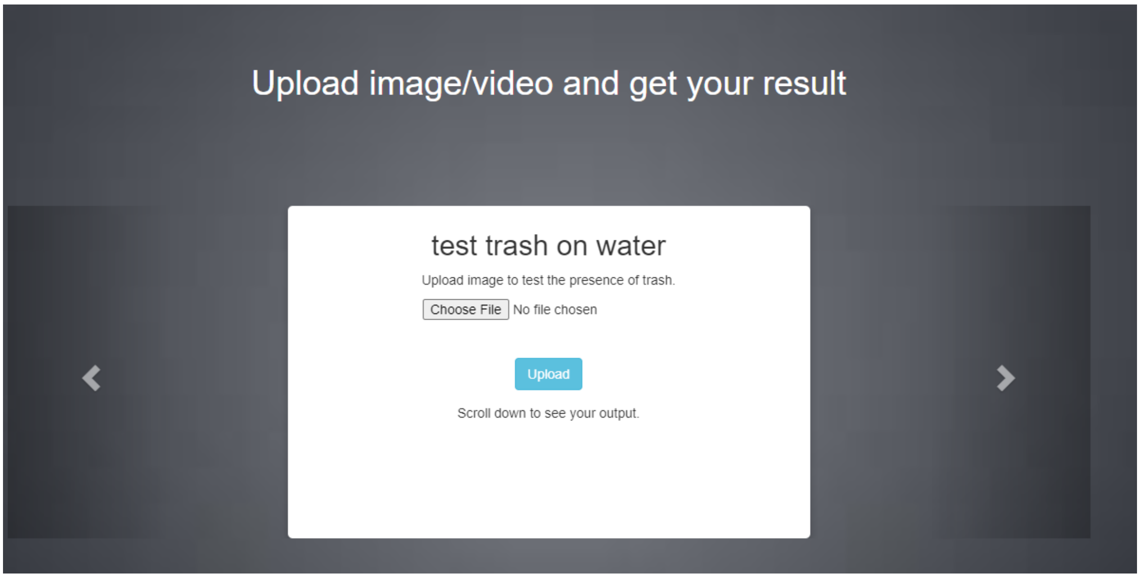


Figure 4.1: Basic structure of the web app

4.2 Mobile App

In order to deploy a lighter version on a low end device, we have converted YOLO-V4 Darknet model to TensorFlowLite.

Conversion of Darknet Model to TensorFlow Lite

- As we know Darknet produces a .weights file for internal use in the Darknet framework. To use it with TensorFlow Lite, we used the tool TensorFlow-YOLOv4-TFLite.
- This tool uses the COCO dataset as a base, so we changed the classes to our custom ones.
- We converted the model from darknet to tensorflow SavedModel, then from TensorFlow Saved model to TensorFlow Lite with simple commands.

We have deployed this lighter version of model(YOLO-V4 TensorFlowLite) in the mobile phone (android). We developed the mobile app with flutter (android studio). The steps included are as follows:

- Train the YOLO-Darknet model and save the weights.
- Convert Darknet Model to TensorFlow Lite.
- Export Weights for Future Inference.
- Deploy on Device.

Figure 4.2: Basic structure of the mobile app

Chapter 5

Results and Discussion

As our main objective is to create a object detection model that would detect algae and trash, we have considered the following metrics as evaluation criteria.

- ***Mean Average Precision:*** To measure how accurate our model is detecting algae and trash on water.
- ***Speed***: To measure at what speed each neural network detects algae/trash in order to validate the correctness of our approach to deploy them in mobiles and platforms such as UAV's.

5.1 Results

The results of all the models we have used are as follows:

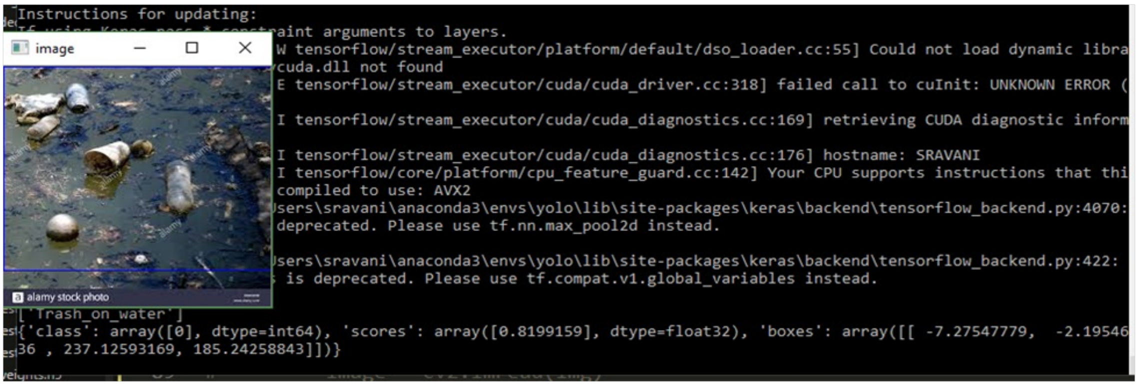


Figure 5.1: Trash on water detection(single class, YOLO V4)

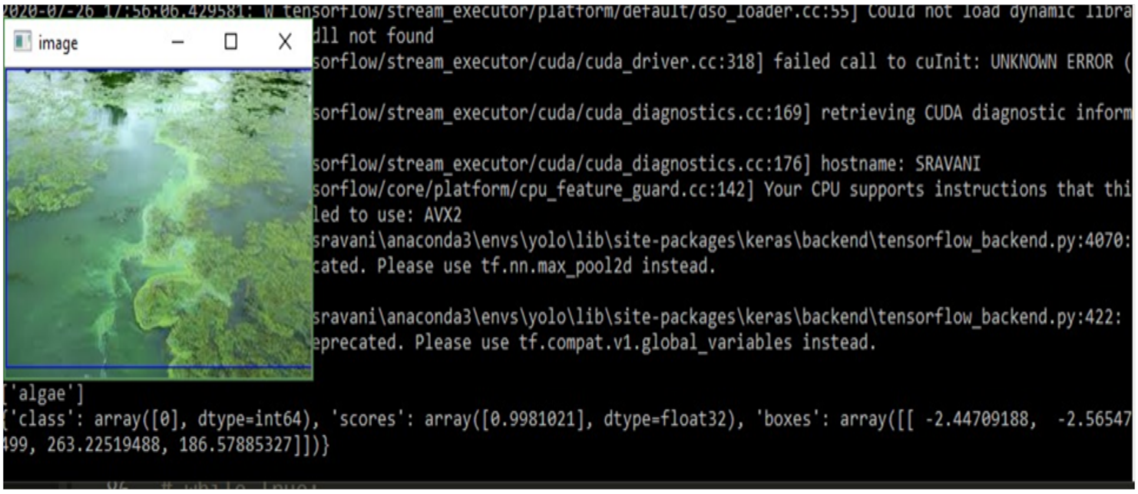


Figure 5.2: Algae detection(single class, YOLO V4)

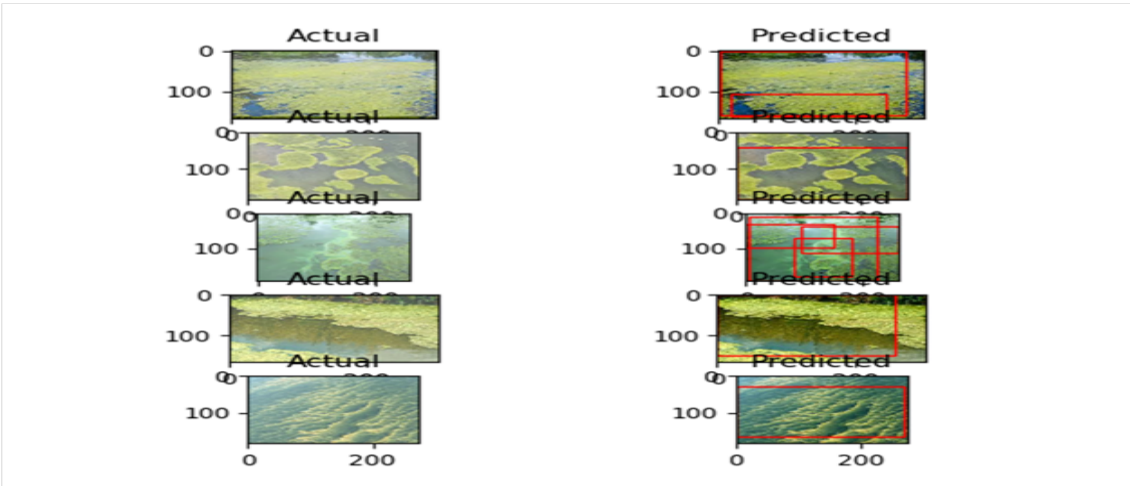


Figure 5.3: Algae on water detection(single class, MS COCO)

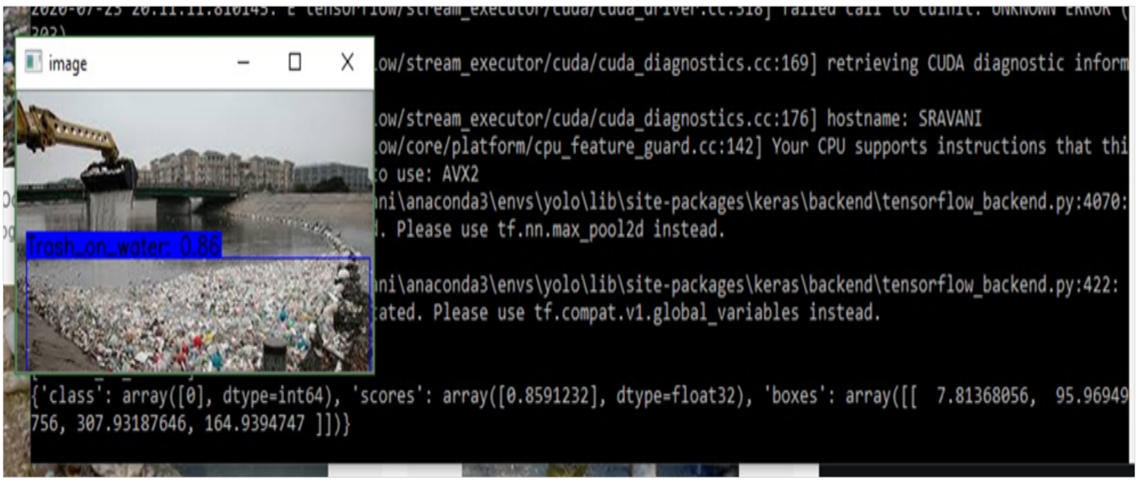


Figure 5.4: Trash on water detection(multi class, YOLO V4)

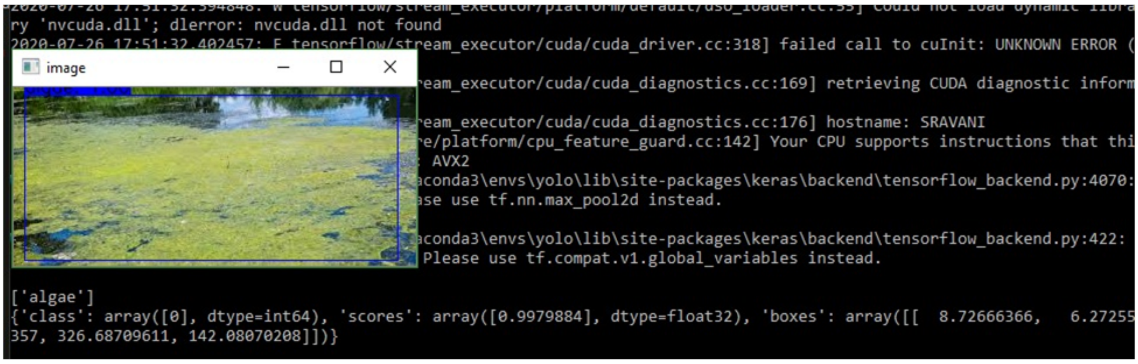


Figure 5.5: Algae on water detection(multi class, YOLO V4)

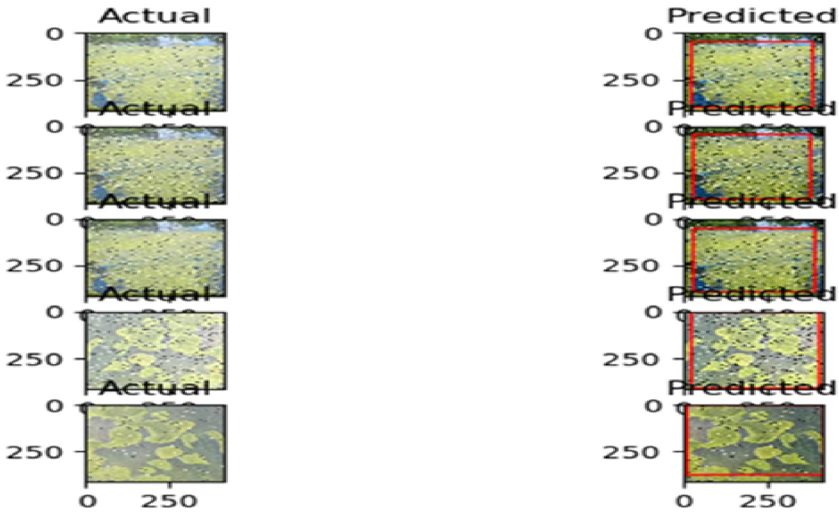


Figure 5.6: Algae on water detection(multi class, MS COCO)

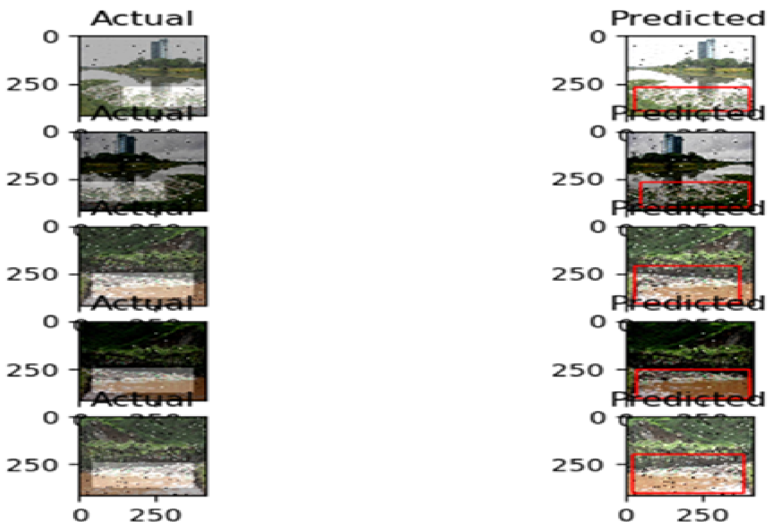


Figure 5.7: Trash on water detection(multi class, MS COCO)

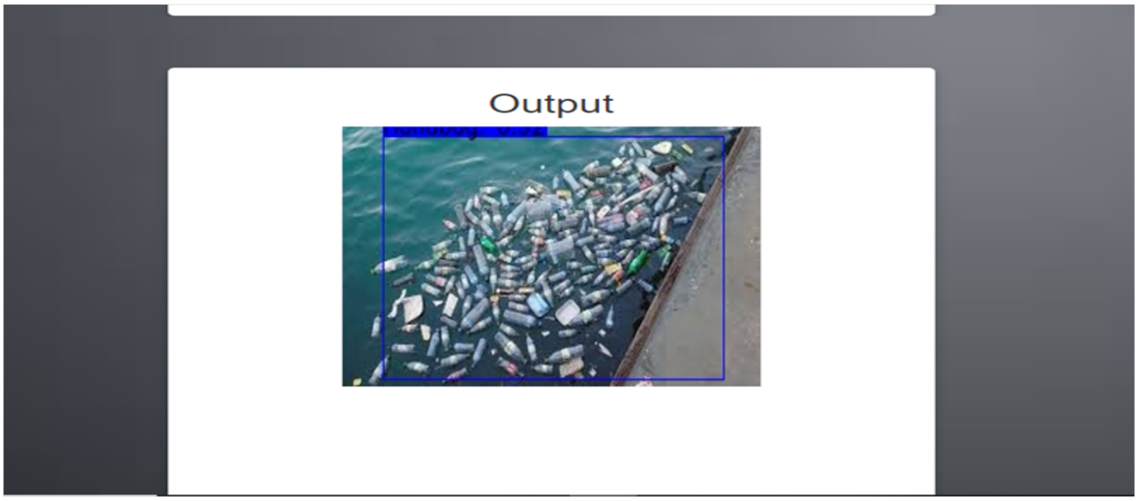


Figure 5.8: Output from Flask Web App

Figure 5.9: Output from Mobile App

5.2 Observations

Name	*Model*	*Class*	*Accuracy (mAp)*	*Time (sec)*
Trash on water	YOLOV4 Darknet	Single class	70	20.65 sec
Algae detection	YOLOV4 Darknet	Sinle class	81.43	20.65 sec
Algae detection	Mask RCNN	Single class	82.4	70 sec
Algae and trash detection	YOLOV4 Darknet	Multi class	80	21.65 sec
Algae and trash detection	Mask RCNN	Multi class	92	70 sec

Table 5.1: Comparison and analysis of models we used

From the above table, of all the models trained,Mask RCNN Multi-class model gave the best accuracy but in terms of speed YOLO-V4 multi-class model is fastest.

5.3 Future Work

We are also planning to implement another method of water pollution detection using image thresholding by considering turbidity as a metric. For this approach, we require a dataset containing the images of water samples along with the turbidity of the respective water sample. Using this dataset we would proceed to image thresholding and predict if the water body is polluted or not. All these models will be further deployed on UAV/mobile app.

Chapter 6

Conclusion

We have experimented one of the traditional methods of detecting water pollution. In this traditional approach, we have collected data from sensors present on the water body and categorized whether water is polluted or not by comparing the sensor data collected with the threshold values of different sensors. Then, we have investigated different object detection techniques and built models(for both algae and trash) with two state of the art object detection technique's YOLO-V4 and Mask RCNN. We have deployed the models into web app. Also, developed a lighter version model and deployed into mobile app(low end device). All these models will be further deployed on UAV(drone).

References

[1] Samantaray, Arabinda and Yang, Baijian and Dietz, J. and Min, Byung-Cheol. (2018). Algae Detection Using Computer Vision and Deep Learning.

[2] U. Shafi, R. Mumtaz, H. Anwar, A. M. Qamar and H. Khurshid, ”Surface Water Pollution Detection using Internet of Things,” 2018 15th International Conference on Smart Cities: Improving Quality of Life Using ICT and IoT (HONET-ICT), Islamabad, 2018, pp. 92-96.

[3] V. Karnawat and S. L. Patil, ”Turbidity detection using image processing,” 2016 International Conference on Computing, Communication and Automation (ICCCA), Noida, 2016, pp. 1086-1089.

[4] T. K. Anyachebelu, M. Conrad, D. Rawson and T. Ajmal, ”Application of water quality index for pollution detection at Luton Hoo Lake,” 2015 IEEE SENSORS, Busan, South Korea, 2015, pp. 1-4, doi: 10.1109/ICSENS.2015.7370182.

[5] S. Randhawa, S. S. Sandha and B. Srivastava, ”A Multi-sensor Process for In-Situ Monitoring of Water Pollution in Rivers or Lakes for High-Resolution Quantitative and Qualitative Water Quality Data,” 2016 IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES), Paris, 2016, pp. 122-129, doi: 10.1109/CSE-EUC-DCABES.2016.171.