River plastic and Debris Detection

-computer vision

TEAM MEMBERS:

1. Deepak T : AI&DS-A (21011101039)
2. Karthick N G : AI&DS-A (21011101058)

RESEARCH PAPER: LINK

ABSTRACT:

Plastic pollution is a critical global issue. Increases in plastic consumption have triggered increased production, which in turn has led to increased plastic disposal. In site observation of plastic litter is tedious and cumbersome, especially in rural areas and around transboundary rivers. We therefore propose automatic mapping of plastic in rivers using unmanned aerial vehicles (UAVs) (like DRONES) and deep learning (DL) models that require modest compute resources.

The model is evaluated at two different sites:

- The Houay Mak Hiao River, Laos
- Khlong Nueng canal in Talad Thai.

Various stakeholders in the effort to monitor and reduce plastic waste in our waterways, can **utilize the resulting** deep learning approach irrespective of location.

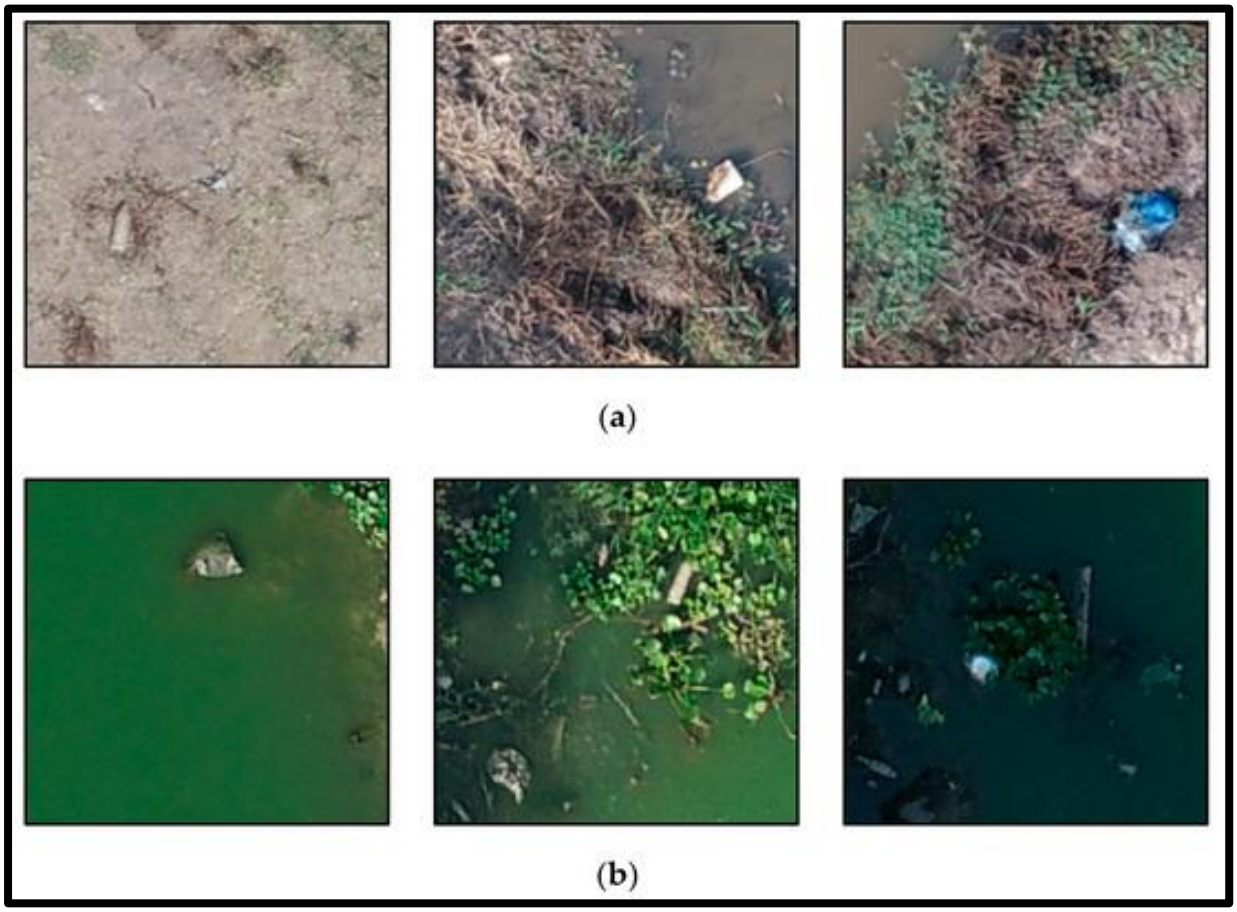
DATASET DESCRIPTION:

The image dataset comprised tiled ortho-images cropped to a size of 256 × 256 pixels corresponding to 2 m × 2 m patches of terrain.

DATASET LINK

DATASET 1	Image count: 500
DATASET 2	Image count: 500

Since, the image quality is low (i.e. bad quality) we planned to add some more drone images .

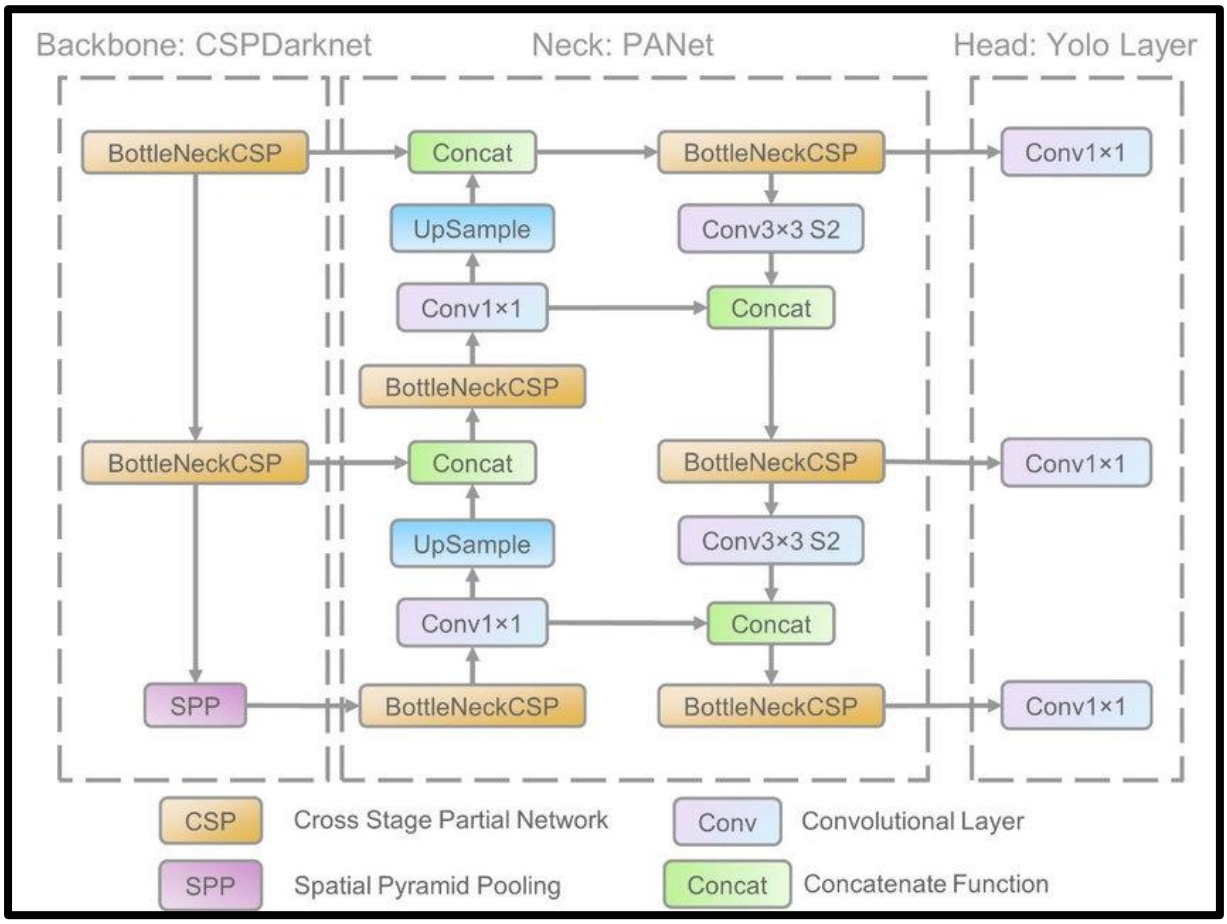


EXISTING ARCHITECTURE:

In the paper they have trained with YOLOv2, YOLOv3, YOLOv4 and YOLOv5 and concluded that **YOLOv5s** gives better Map.

So, we will see its **architecture** now:

The **YOLOv5** architecture is based on a single-stage object detection framework, which means it performs both bounding box localization and object classification in a single forward pass through the neural network. It uses a deep convolutional neural network to achieve this.



- **Backbone Network:** YOLOv5 uses a CSPDarknet53 as the backbone network. This is a modified version of the Darknet architecture, which employs a CSP (cross-stage partial connections) module to improve information flow and training efficiency.

- **Neck:** YOLOv5 uses PANet (**Path Aggregation Network**) as its neck, which is responsible for combining features of different scales to aid in detecting objects of various sizes.
- **Detection Head:** The detection head is composed of a series of convolutional layers responsible for predicting bounding boxes and class probabilities. The final output is a set of bounding box coordinates along with corresponding class probabilities for each detected object.
- **Loss Function:** YOLOv5 uses a combination of multiple loss functions, including objectness loss, box regression loss, and class prediction loss. These losses are used to train the model to accurately detect and classify objects.

The YOLOv5 family includes different variants with different model sizes, denoted by the suffixes (e.g., yolov5s, yolov5m, yolov5l, and yolov5x), where 's' stands for "**small**". These variants differ in the number of layers and parameters, allowing users to choose the appropriate model based on their hardware constraints and accuracy requirements.

PROOF OF SPACE FOR IMPORVEMENT:

Model : Used YOLOv5 , but latest available model-YOLOv8.

Accuracy : 78% which can be improved.

RESULT ANALYSIS

Model	Training Time (h)	Inference Time per Image (s)	Model Size (MB)	Computational Complexity (GFLOPs)	mAP @ 0.5 IoU for Validation Dataset	Map @ 0.5 IoU for Testing Dataset	Highest F1 Score	Computing Platform
Pre-trained YOLOv2	0.359	4.74	192.9	29.338	0.723	0.442	0.66	Google Colab
YOLOv2 scratch	0.367	4.84	192.9	29.338	0.581	0.259	0.6	
Pre-trained YOLOv2-tiny	0.166	3.53	42.1	5.344	0.467	0.293	0.38	
YOLOv2-tiny scratch	0.23	3.52	42.1	5.344	0.348	0.286	0.44	
Pre-trained YOLOv3 tiny	0.082	0.01	16.5	12.9	0.714	0.366	0.7	
YOLOv3-tiny scratch	0.082	0.004	16.5	12.9	0.555	0.336	0.58	
Pre-trained YOLOv3	0.259	0.018	117	154.9	0.735	0.396	0.72	Intel®Core™ i7-10750H CPU @2.60 GHz, 16 GB RAM, and GPU as NVIDIA GeForce RTX 2060
YOLOv3 scratch	0.258	0.017	117	154.9	0.479	0.311	0.54	
Pre-trained YOLOv3-spp	0.266	0.017	119	155.7	0.787	0.402	0.75	
YOLOv3-spp scratch	0.279	0.014	119	155.7	0.59	0.265	0.57	
Pre-trained YOLOv4	1.884	6.85	244.2	59.563	0.809	0.463	0.78	Google Colab
YOLOv4 scratch	1.961	5.54	244.2	59.563	0.766	0.373	0.74	
Pre-trained YOLOv4-tiny	0.899	2.92	22.4	6.787	0.758	0.418	0.76	
YOLOv4-tiny scratch	0.968	2.72	22.4	6.787	0.732	0.355	0.73	
Pre-trained YOLOv5s	0.146	0.019	13.6	16.3	0.810	0.424	0.78	
YOLOv5s scratch	0.149	0.017	13.6	16.3	0.740	0.272	0.67	
Pre-trained YOLOv5m	0.195	0.041	40.4	50.3	0.787	0.434	0.77	
YOLOv5m scratch	0.197	0.04	40.4	50.3	0.695	0.331	0.70	Intel®Core™ i7-10750H CPU @2.60 GHz, 16

PROPOSED ARCHITECTUE:

YOLOv8m (LEARNING IN PROCESS FOR US)