Flutura's Model Card

MODEL SPECIFICATION

Model Type

Yolov5 large

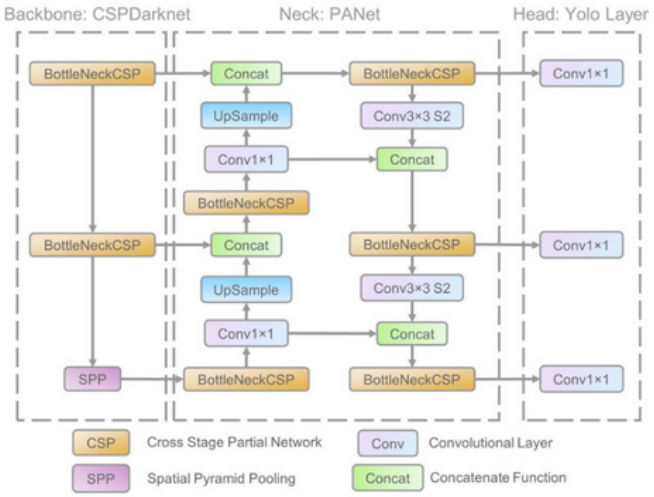
Model Size , Format & Framework

- 80MB size
- pytorch (.pth) format
- YoloV5 from Ultralytics

Model Architecture

Conv Model is Consist of Three Major Part:-

- DarkNet Layer
- PANet Layer
- Yolo Layer



Input

Regions in the video frames where a drone has been detected. Represented as a 640x480x3 array.

Output

Output array from model containing:-.

- Bounding Box co-ordinates of detected pallets
- Confidance Score
- predicted Class id

Basic Information

Trained By With Date

- Employee Name (ID)
- Nov-23 -2022

Training Source Code Licensed Under

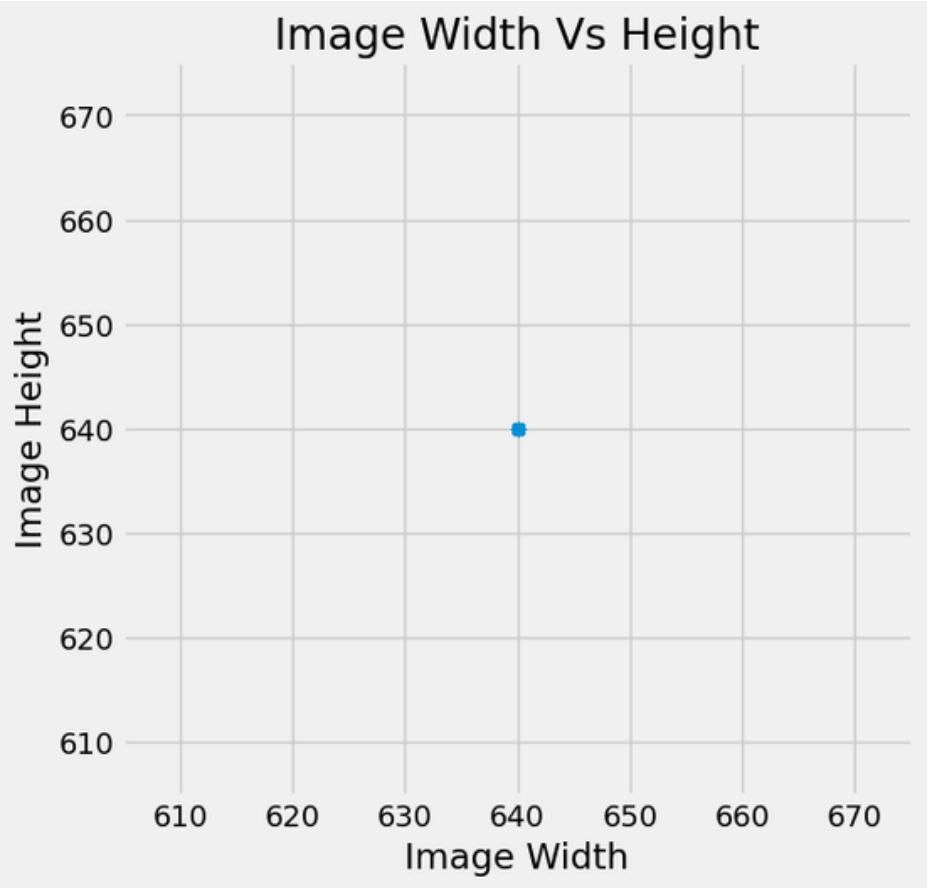
GPL-3.0 license

Data Information

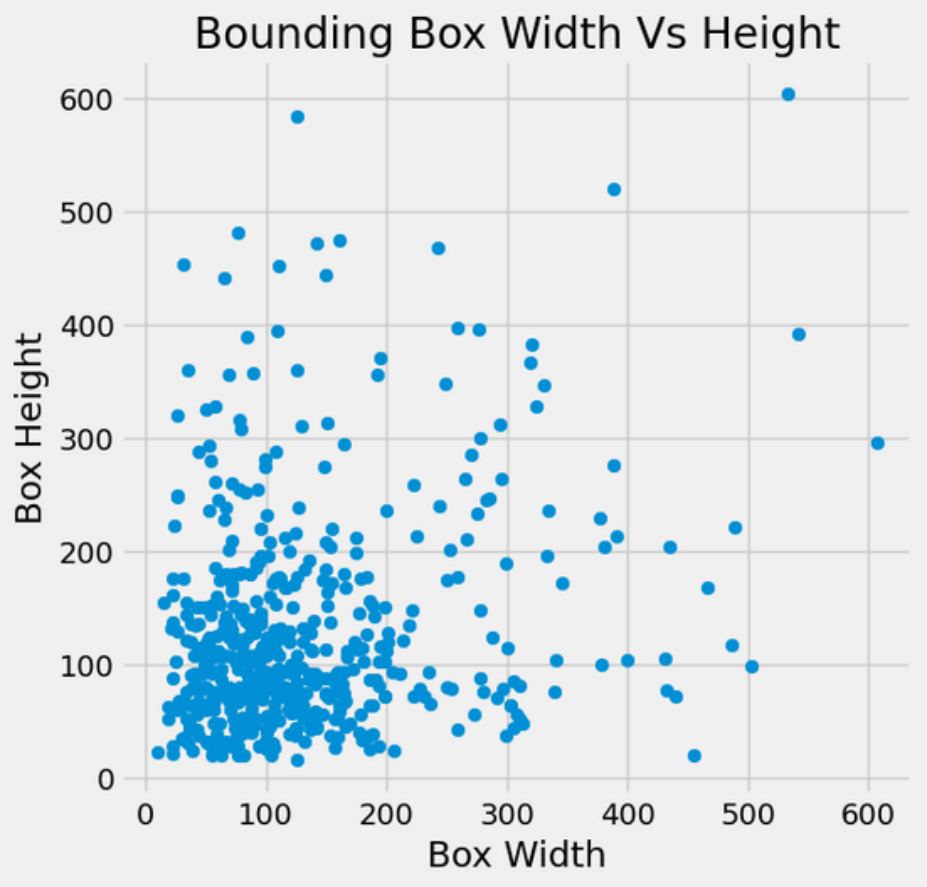
- **Annotation Format** :- YOLO Format
- **Machine Used**:- Colab/GDrive (Sarang Flutura Account)
- **Drive Link:-** Link
- **Data Path in Machine**:- /path/to/MyDrive/cardBox/data_dir/box_defect_yolo
- **Data Type:-** Without Augmentation
- **Data Version:-** 1.0
- **Number Of Images:- 687**
- **Data Split Ratio:-**
  - Train Ratio - 0.80
  - Val Ratio - 0.10
  - Test Ratio - 0.10
- **Number Of Classes & List:-**
  - 1
  - [defect]
- **Data Licensed Under :-** Data Collection from Roboflow & Random site manually Download
- **Inference Result:-**/path/to/MyDrive/cardBox/exp2/inference_new

Exploratory Data Analysis

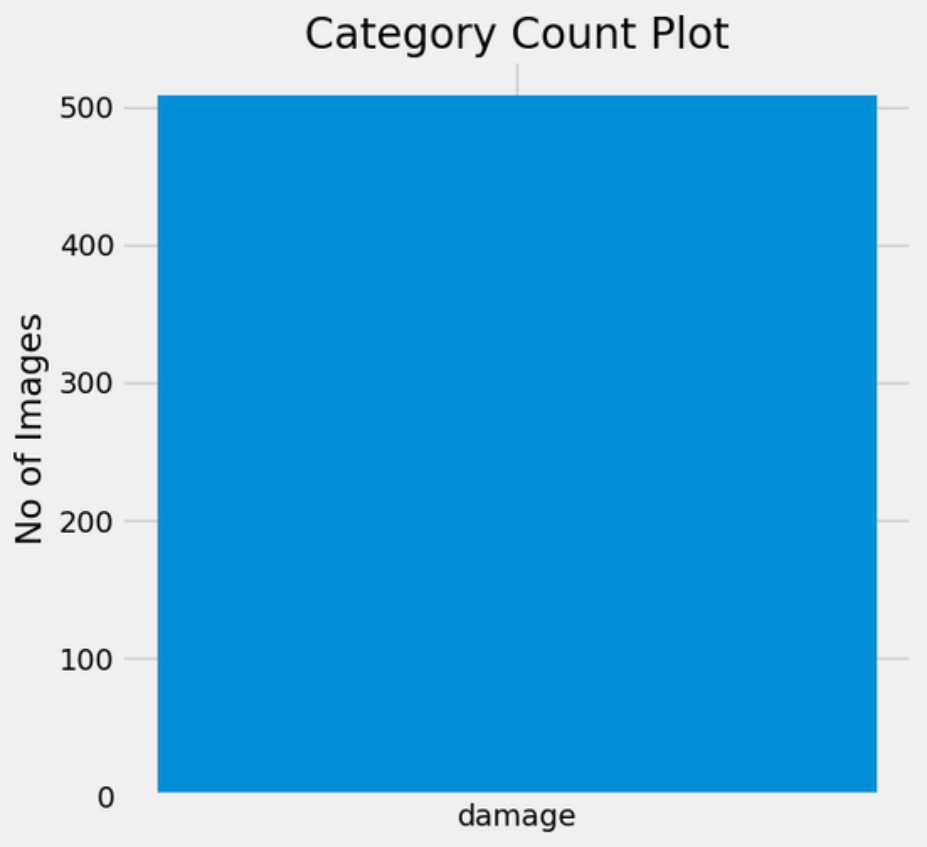
Train Images Height Vs Width



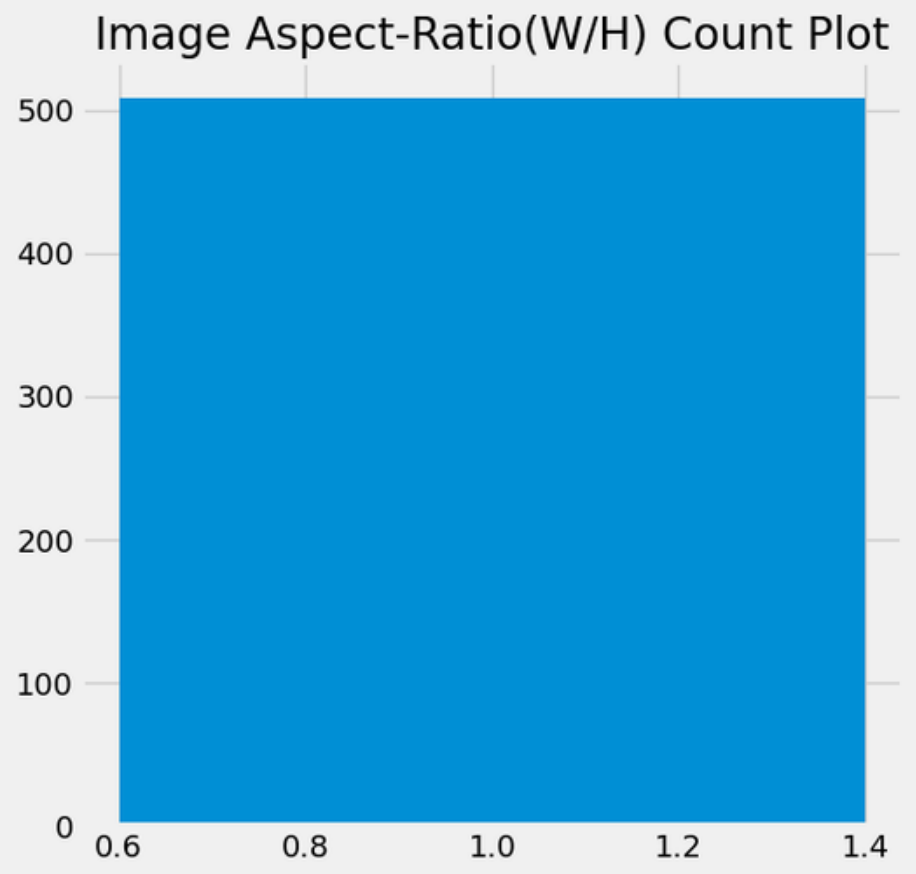
Train Bounding Box Height Vs Width



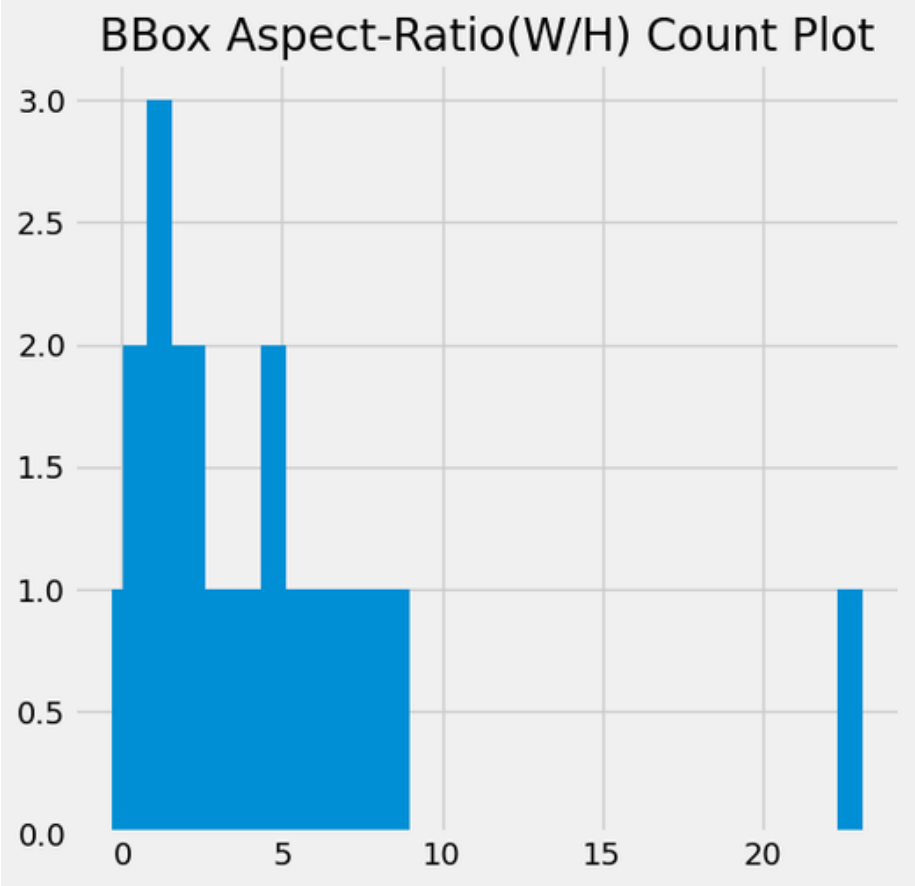
Train Class Label CountPlot



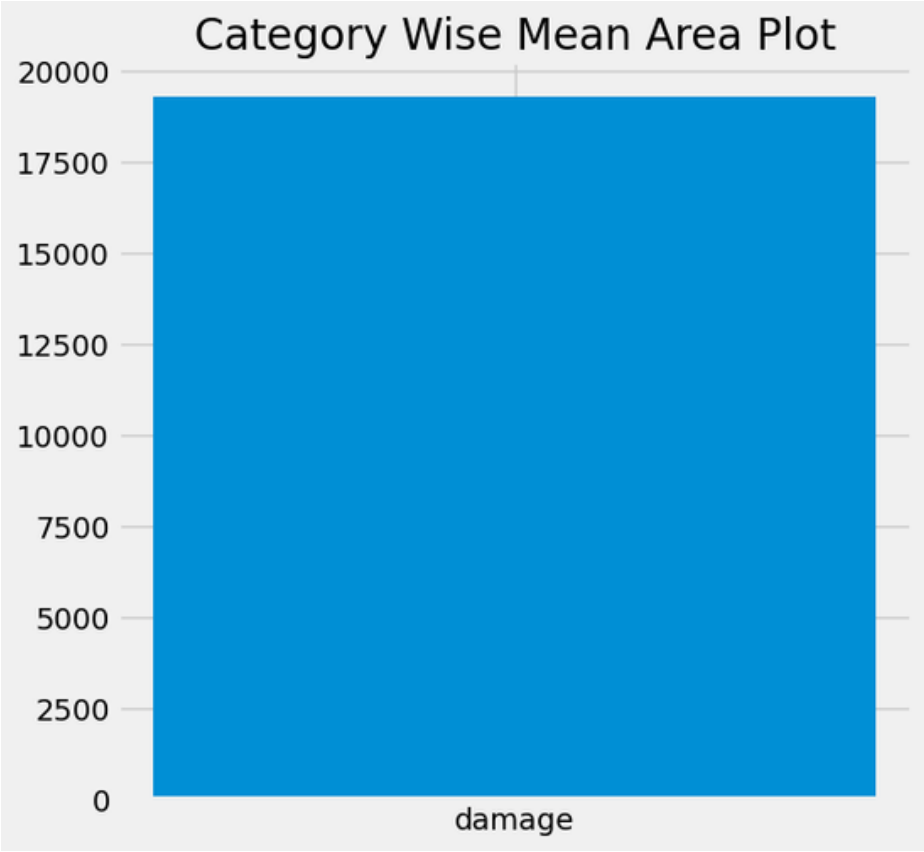
Train Image Aspect Ratio Count



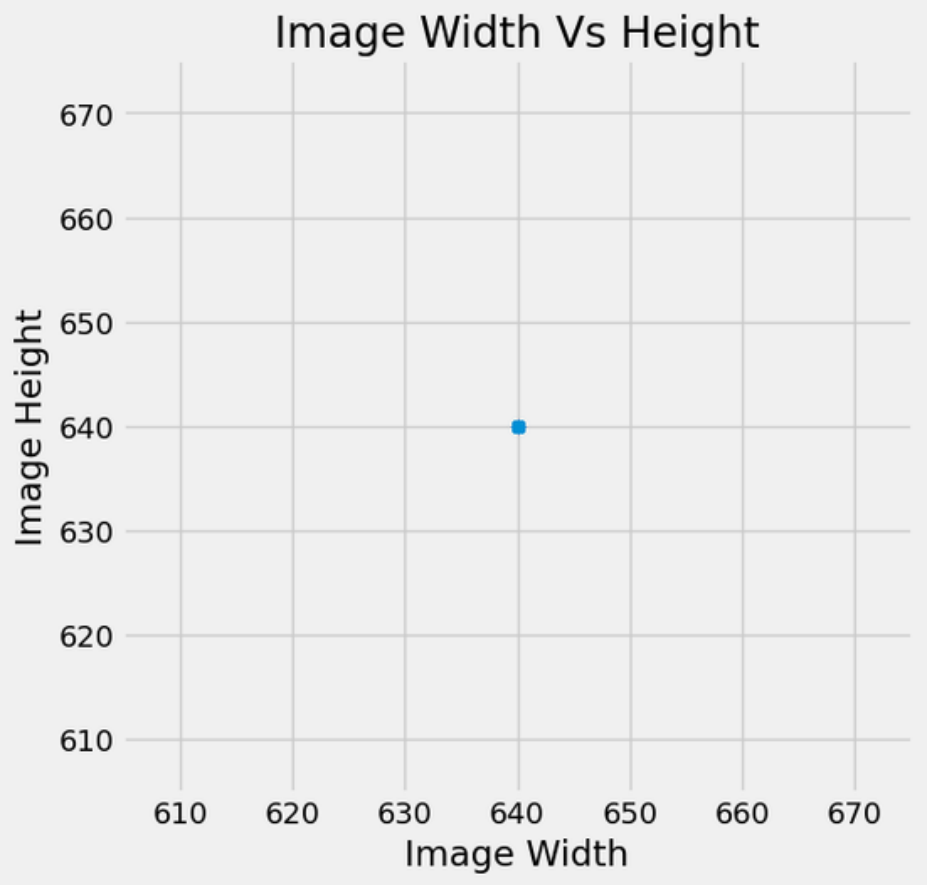
Train BBox Aspect Ratio Count



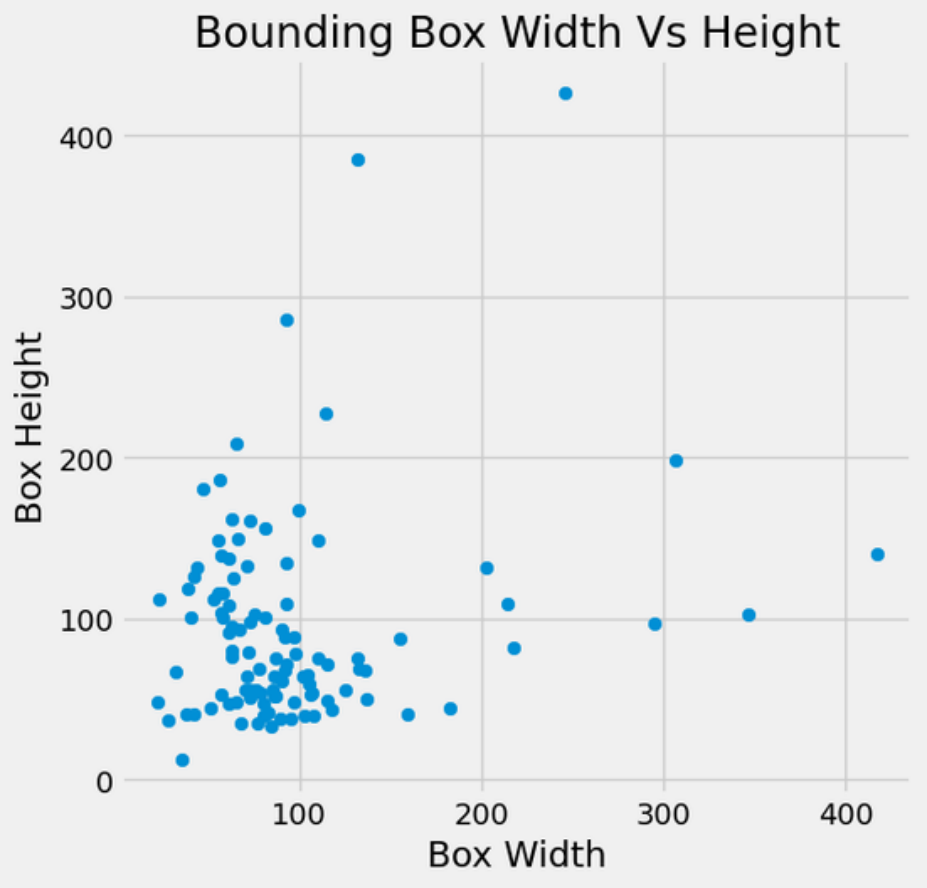
Train Category Wise mean Area Plot



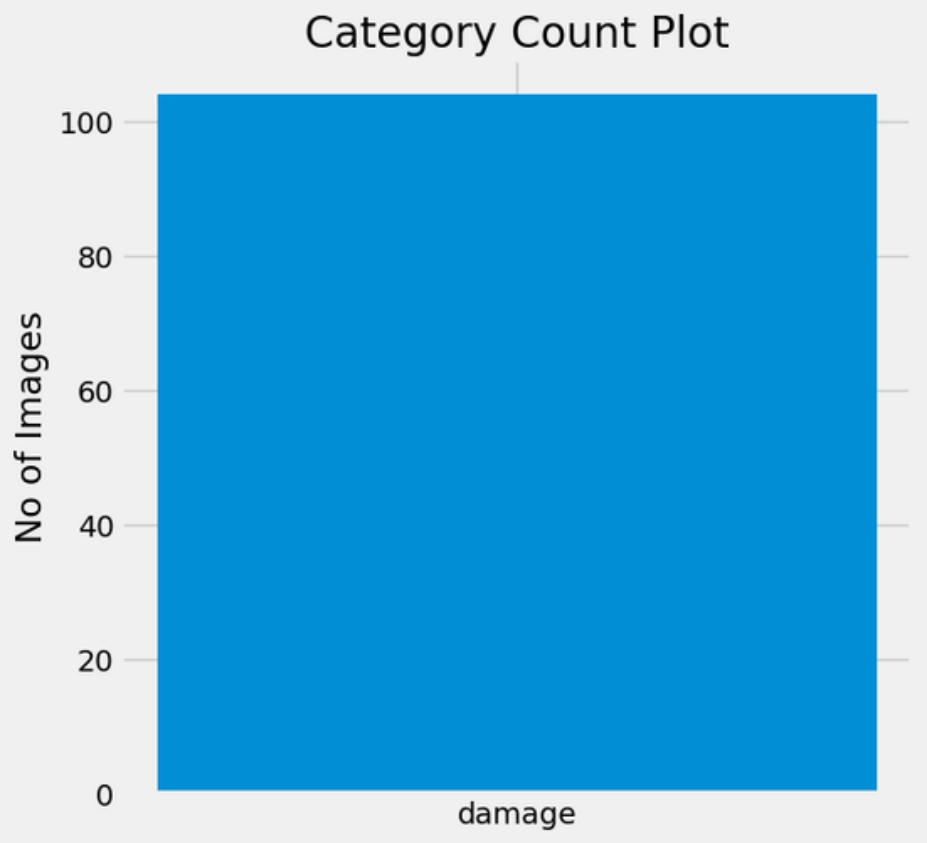
Validation Image Width Vs Height Plot



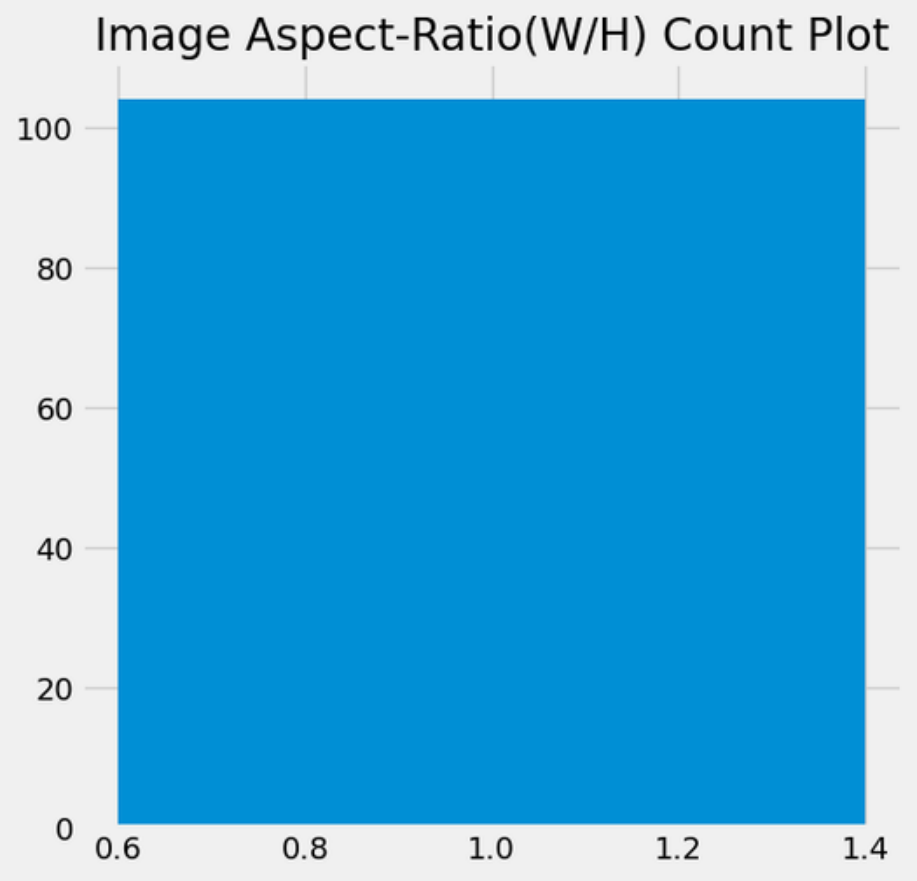
Validation Bounding Box Width Vs Height Plot



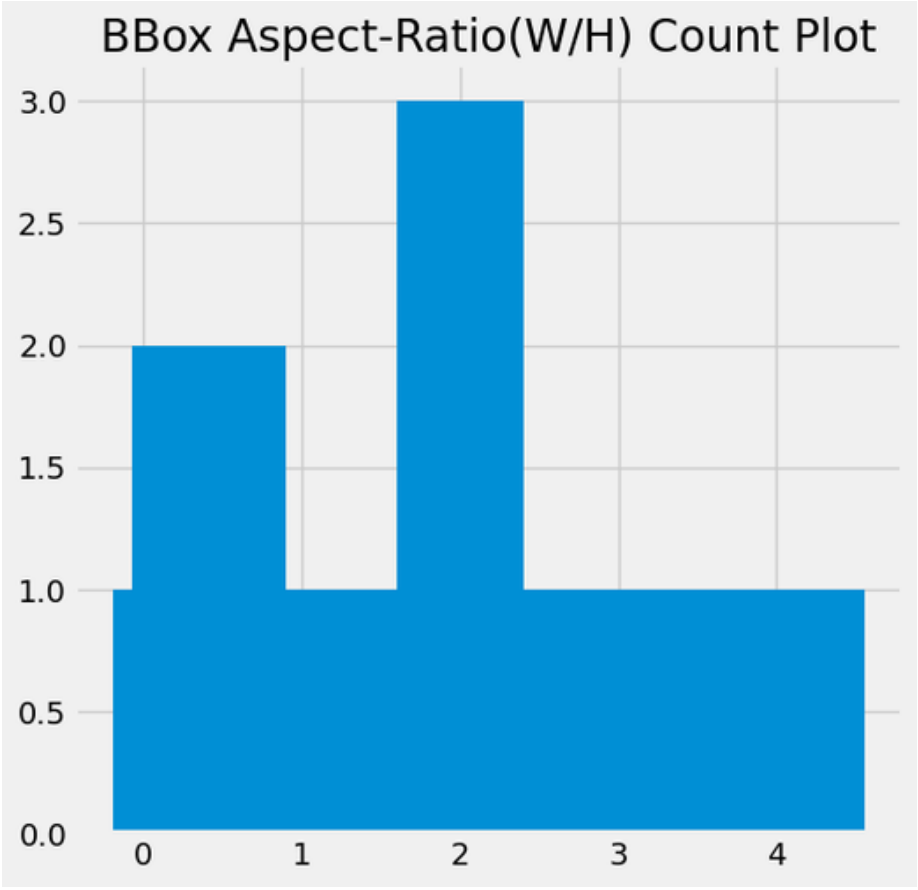
Validation Category Count Plot



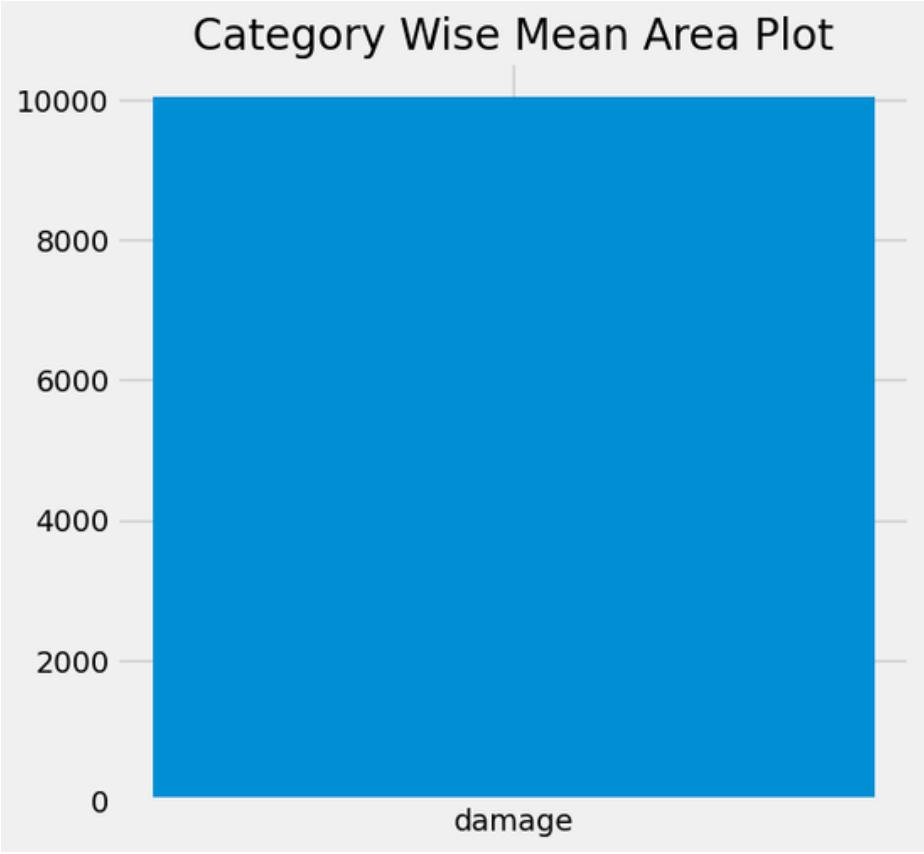
validation Image Aspect -Ratio Plot



Validation BBOX Aspect -Ratio Plot



Validation Category Wise Mean Area Plot



Model Training Information

Training factors

Instrumentation & Hardware

- All dataset images were collected from Roboflow.
- environment with different light, noise and motion conditions Application.
- Model Trained on Tesla T4 GPU 16GB RAM

Training Environment Requirements

- Cuda Version should be 11.3
- Pytorch Version Should be 1.12.0
- Anaconda should be use to create environment
- Python version should be 3.7.12

Model & Hyper Parameters Information

- **working Directory:-** /path/to/MyDrive/cardBox/pytorch_yolov5/
- **Config File Name:-** models/yolov5_l.yaml
- **Load From:-** AutoDownload / path_to_custom_model
- **Resume From:-** path_to_custom_model
- **Log_Interval:-** 1 epochs (Early Stopping at 10)
- **Optimizer_Type:-** SGD
- **Optimizer_Learning_Rate:-** 0.01
- **Momentum:-** 0.9
- **Weight_Decay:-** 0.0001

Training Parameters Information

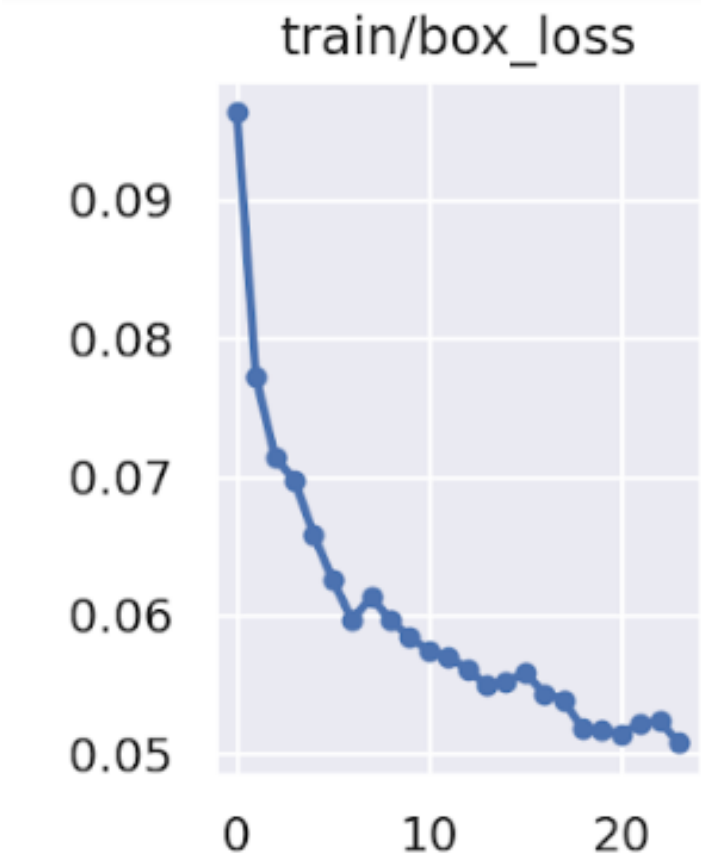
- **No Of Epochs :-** 300 (20 Early Stopping)
- **Work Flow:-** [('train',1),('val',1)]
- **Experiment Name:-** card_box
- **Device:-** CUDA
- **Log Checkpoint Interval:-** 10
- **Experiment Interval:-** 1
- **Gpu Ids:-** [0]
- **Evalution Metric:-** 'bbox'

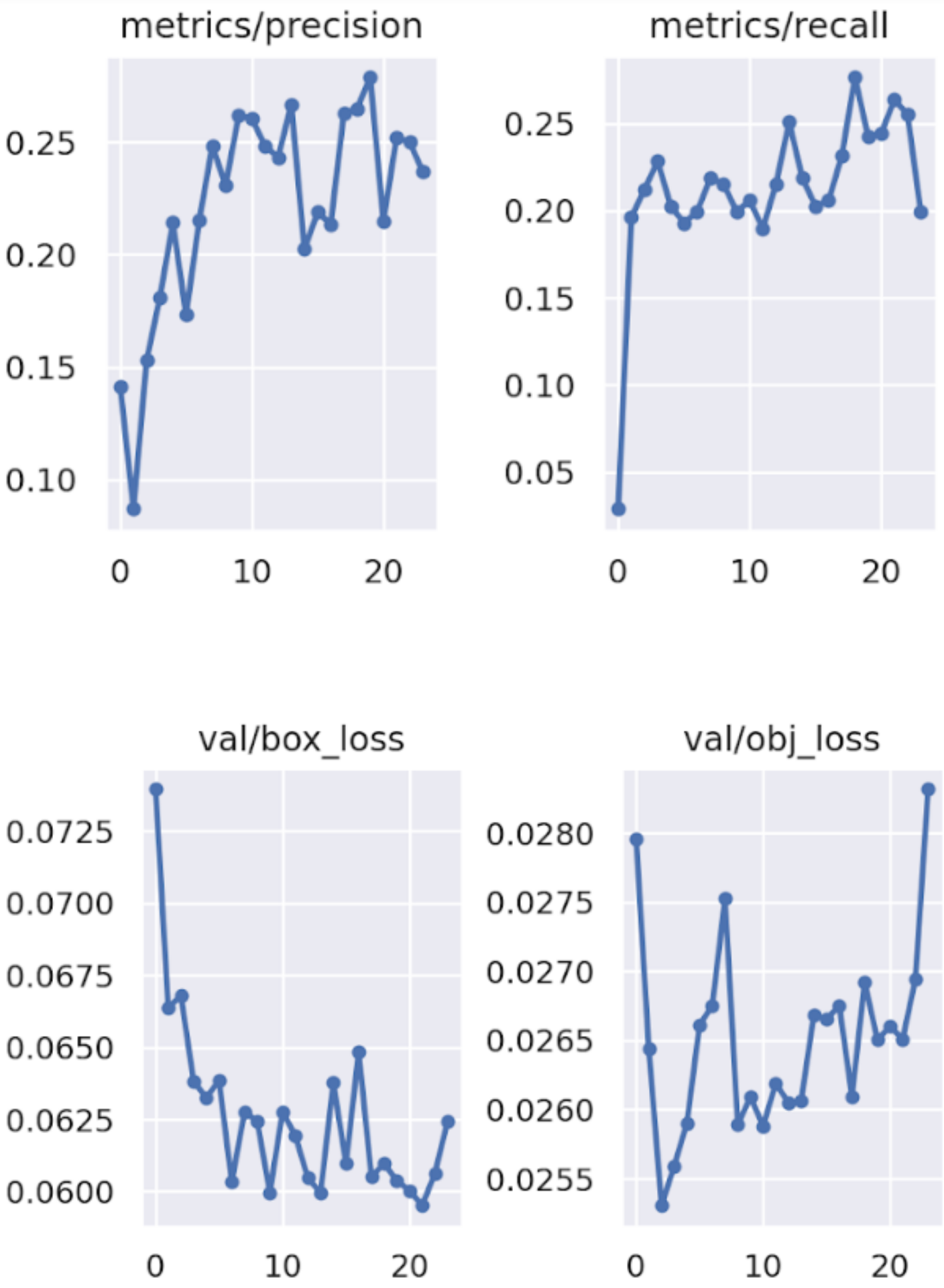
Post Training Evalution

- **Best Model Path With Metrics :-**

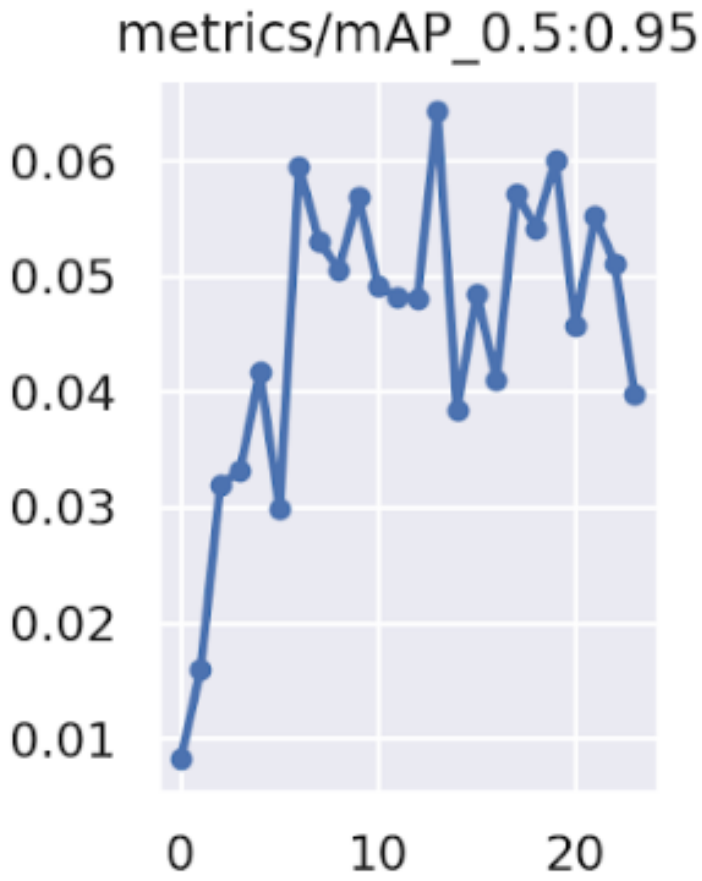
- Path:- /path/to/MyDrive/work_dir/card_box/experiment2

- Metrics:-
    - Coco mAP:50 0.16
    - Coco mAP_50:95 0.06

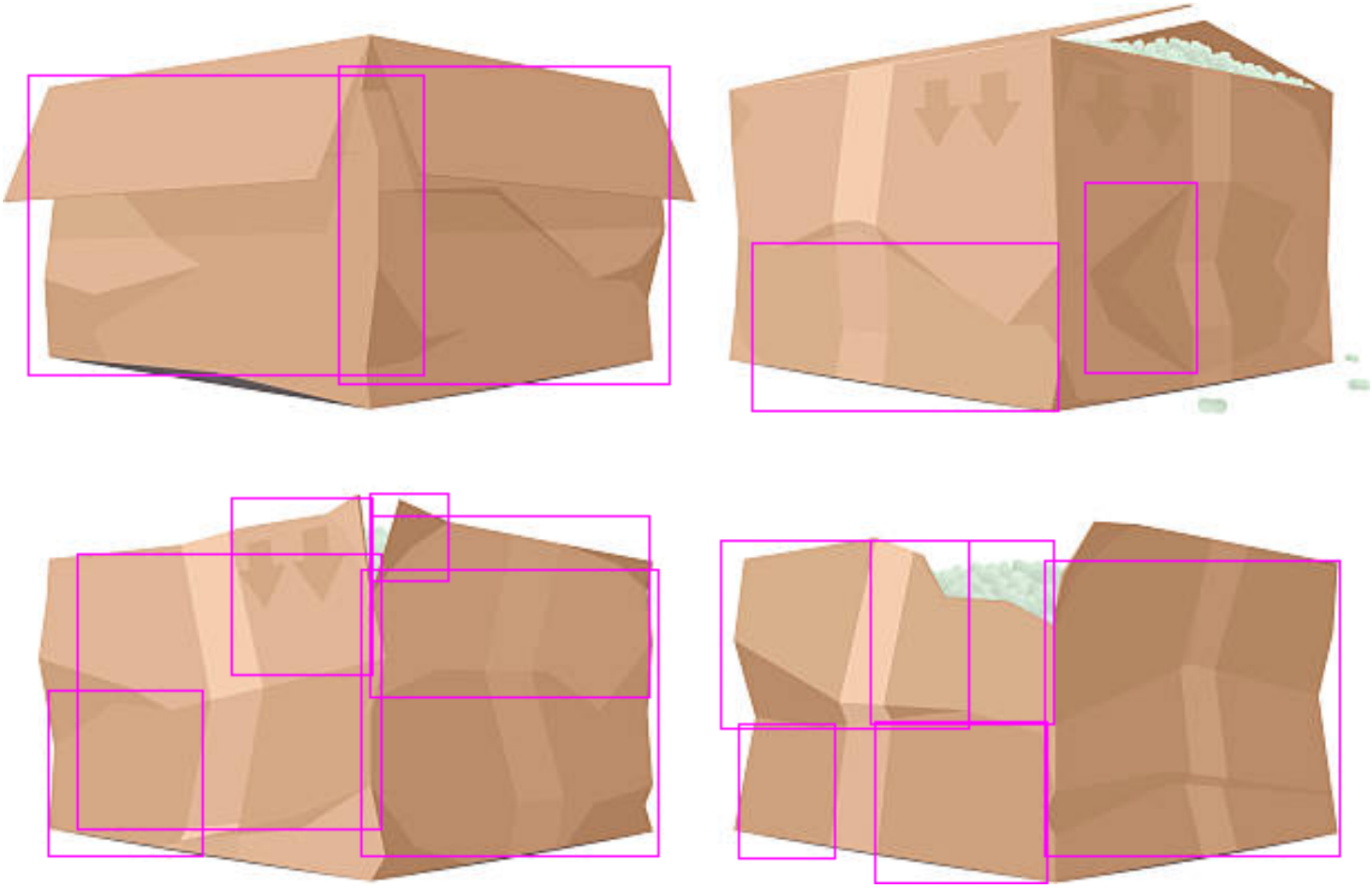


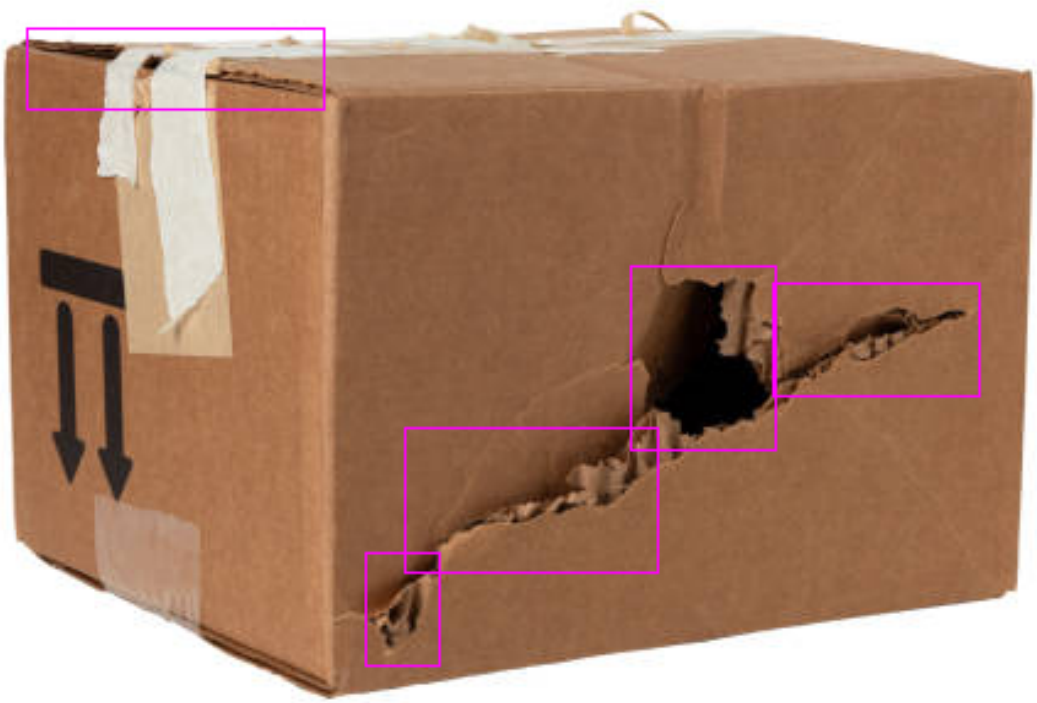

metrics/precision
0.25
0.20
0.15
0.10
0
10
20
metrics/recall
0.25
0.20
0.15
0.10
0.05
0
10
20
val/box_loss
0.0725
0.0700
0.0675
0.0650
0.0625
0.0600
0
10
20
val/obj_loss
0.0280
0.0275
0.0270
0.0265
0.0260
0.0255
0
10
20

metrics/mAP_0.5
0.15
0.10
0.05
0
10
20


metrics/mAP_0.5:0.95
0.06
0.05
0.04
0.03
0.02
0.01
0
10
20

Results





INTENDED USES

Application

- To Detect the defect in Card Box.

Domain And Uses

- Manufacturing Industry
- Material Defect Detection
- Real Time Health & Safety Applications
- To Detect the Gaseous & aqueous content

Out Of Scope Applications

- It is Trained only for one class (defect)
- We can't use this model for other Multiclass purpose other we have to take the change the head.

Limitations

Presence Of Attributes

- Detects only box card on 640 * 480 scaled Image.
- All defects types are classified as one.
- Able to get detection properly but having issue with less mAP but accuracy is good.

Trade-Offs

- The model is optimized for real-time performance on a wide variety of mobile devices, but is sensitive to drone position, Orientation.
- Environment during Inference should be same as Training.

Environment

- When changing the environment lightening, Image noise, motion , Contrast & Saturation.
- Changing the OS & GPU Card will change the Training Time & Inference Time of Model.
- May not perform well on a fast paced line.