Eddie Christopher Fox III

Student ID: 1155160788

April 3, 2022

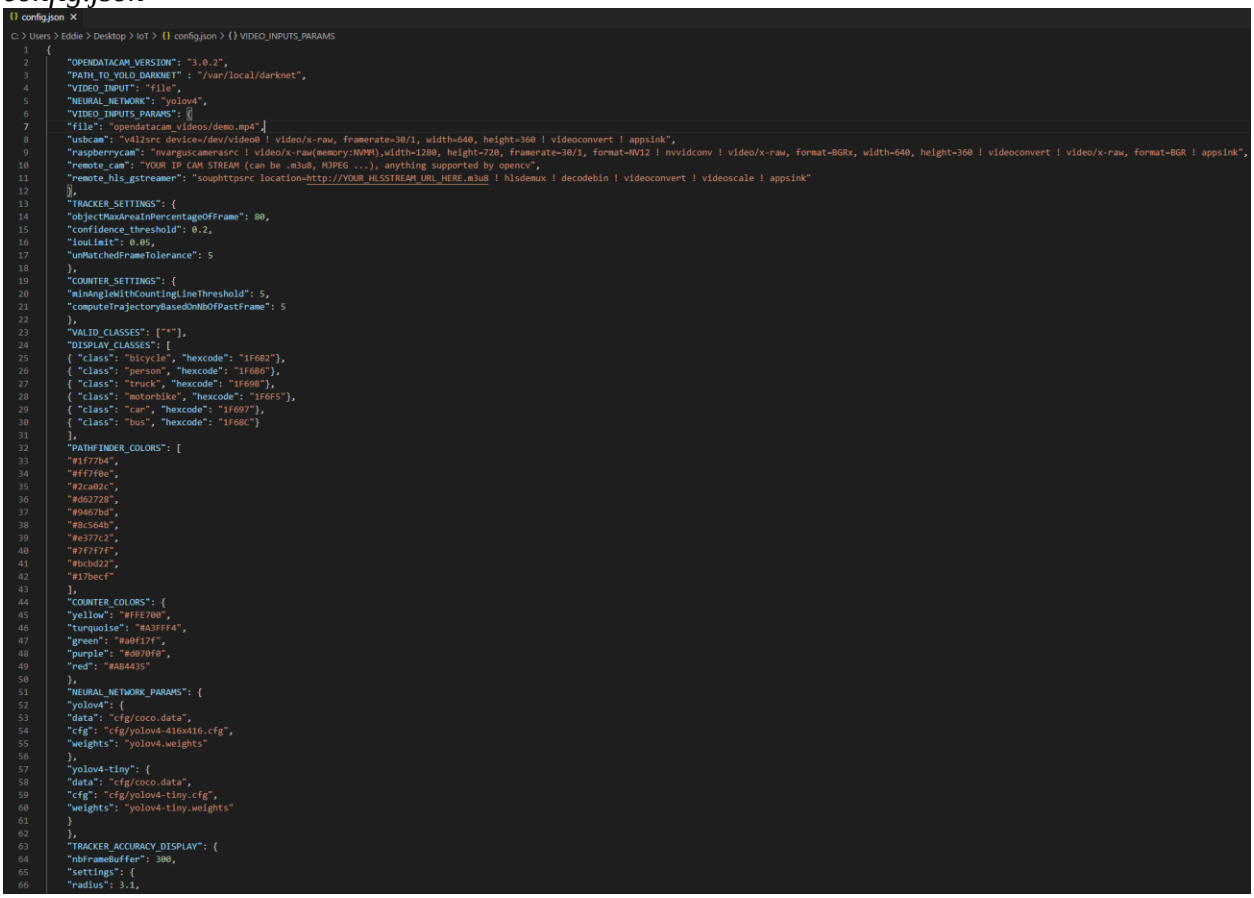
IEMS 5709 Lab 4 Report

Deliverables:

K8s YAML files for Opendatacam and Opendatacam-mongodb deployment and service.

I will post a screenshot of every file, and also upload them on blackboard.

config.json



```
config.json
C: > Users > Eddie > Desktop > IoT > {} config.json > {} VIDEO_INPUTS_PARAMS
{
    "OPENDATACAM_VERSION": "3.0.2",
    "PATH_TO_YOLO_DARKNET" : "/var/local/darknet",
    "VIDEO_INPUT": "file",
    "NEURAL_NETWORK": "yolov4",
    "VIDEO_INPUTS_PARAMS": {
    "file": "opendatacam_videos/demo.mp4",
    "usbcam": "v4l2src device=/dev/video0 ! video/x-raw, framerate=30/1, width=640, height=360 ! videoconvert ! appsink",
    "raspberrycam": "nvarguscamerasrc ! video/x-raw(memory:NVMM),width=1280, height=720, framerate=30/1, format=NV12 ! nvvidconv ! video/x-raw, format=BGRx, width=640, height=360 ! videoconvert ! video/x-raw, format=BGR ! appsink",
    "remote_cam": "YOUR IP CAM STREAM (can be .m3u8, MJPEG ...), anything supported by opencv",
    "remote_hls_gstreamer": "souphttpsrc location=http://YOUR_HLSSTREAM_URL_HERE.m3u8 ! hlsdemux ! decodebin ! videoconvert ! videoscale ! appsink"
    },
    "TRACKER_SETTINGS": {
    "objectMaxAreaInPercentageOfFrame": 80,
    "confidence_threshold": 0.2,
    "iouLimit": 0.05,
    "unMatchedFrameTolerance": 5
    },
    "COUNTER_SETTINGS": {
    "minAngleWithCountingLineThreshold": 5,
    "computeTrajectoryBasedOnNbOfPastFrame": 5
    },
    "VALID_CLASSES": ["*"],
    "DISPLAY_CLASSES": [
    { "class": "bicycle", "hexcode": "1F6B2"},
    { "class": "person", "hexcode": "1F6B6"},
    { "class": "truck", "hexcode": "1F69B"},
    { "class": "motorbike", "hexcode": "1F6F5"},
    { "class": "car", "hexcode": "1F697"},
    { "class": "bus", "hexcode": "1F68C"}
    ],
    "PATHFINDER_COLORS": [
    "#1f77b4",
    "#ff7f0e",
    "#2ca02c",
    "#d62728",
    "#9467bd",
    "#8c564b",
    "#e377c2",
    "#7f7f7f",
    "#bcbd22",
    "#17becf"
    ],
    "COUNTER_COLORS": {
    "yellow": "#FFE700",
    "turquoise": "#A3FFF4",
    "green": "#a0f17f",
    "purple": "#d070f0",
    "red": "#AB4435"
    },
    "NEURAL_NETWORK_PARAMS": {
    "yolov4": {
    "data": "cfg/coco.data",
    "cfg": "cfg/yolov4-416x416.cfg",
    "weights": "yolov4.weights"
    },
    "yolov4-tiny": {
    "data": "cfg/coco.data",
    "cfg": "cfg/yolov4-tiny.cfg",
    "weights": "yolov4-tiny.weights"
    }
    },
    "TRACKER_ACCURACY_DISPLAY": {
    "nbFrameBuffer": 300,
    "settings": {
    "radius": 3.1,
```

Opendatacam-deployment.yaml

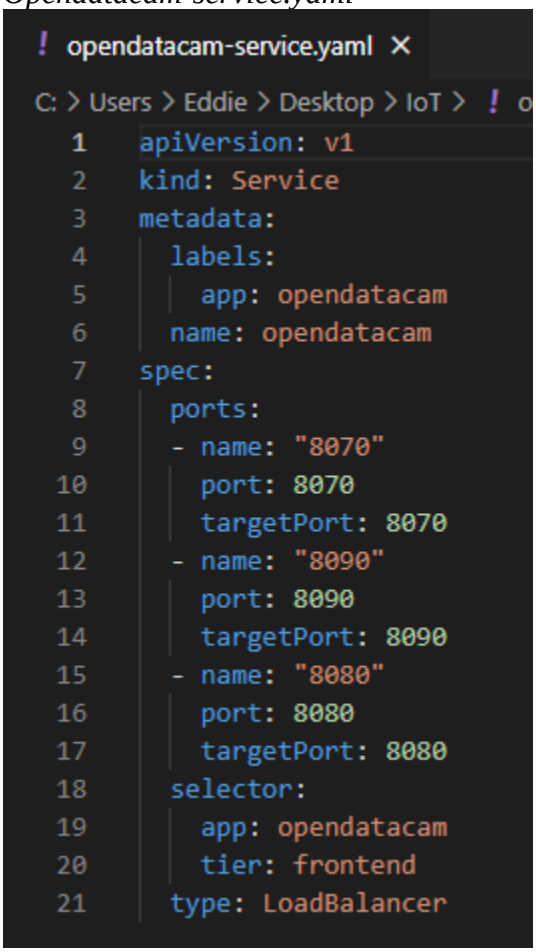
Screenshot on next page.

! opendatacam-deployment.yaml ×

C: > Users > Eddie > Desktop > IoT > ! opendatacam-deployment.yaml

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: opendatacam
  name: opendatacam
spec:
  replicas: 1
  selector:
    matchLabels:
      app: opendatacam
      tier: frontend
  template:
    metadata:
      labels:
        app: opendatacam
        tier: frontend
    spec:
      containers:
      - image: opendatacam/opendatacam:v3.0.2-xavier
        command: ["/bin/bash"]
        args: ["-c", "/var/local/opendatacam/launch.sh"]
        name: opendatacam
        ports:
        - containerPort: 8080
        - containerPort: 8070
        - containerPort: 8090
        resources: {}
        securityContext:
          privileged: true
        volumeMounts:
        - mountPath: /var/local/opendatacam/config.json
          name: opendatacam-config
          subPath: "config.json"
      restartPolicy: Always
      volumes:
      - name: opendatacam-config
        configMap:
          name: opendatacam
          items:
            - key: config.json
              path: config.json
```

Opendatacam-service.yaml



Opendatacam-mongo-deployment.yaml

Screenshot on next page.

! opendatacam-mongo-deployment.yaml ×

C: > Users > Eddie > Desktop > IoT > ! opendatacam-mongo-de

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: opendatacam
  name: opendatacam-mongo
spec:
  replicas: 1
  selector:
    matchLabels:
      app: opendatacam
      tier: mongo
  template:
    metadata:
      labels:
        app: opendatacam
        tier: mongo
    spec:
      containers:
      - image: mongo
        name: mongo
        ports:
        - containerPort: 27017
        resources: {}
        volumeMounts:
        - mountPath: /data/db
          name: mongodb-persistent-storage
      restartPolicy: Always
      volumes:
      - name: mongodb-persistent-storage
        persistentVolumeClaim:
          claimName: mongodb-pv-claim
```

Opendatacam-mongo-service.yaml
Screenshot on next page.

opendatacam-mongo-service.yaml

```yaml
apiVersion: v1
kind: Service
metadata:
  labels:
    app: opendatacam
  name: opendatacam-mongo
spec:
  ports:
  - name: "27017"
    port: 27017
    targetPort: 27017
  selector:
    app: opendatacam
    tier: mongo
  type: ClusterIP
```

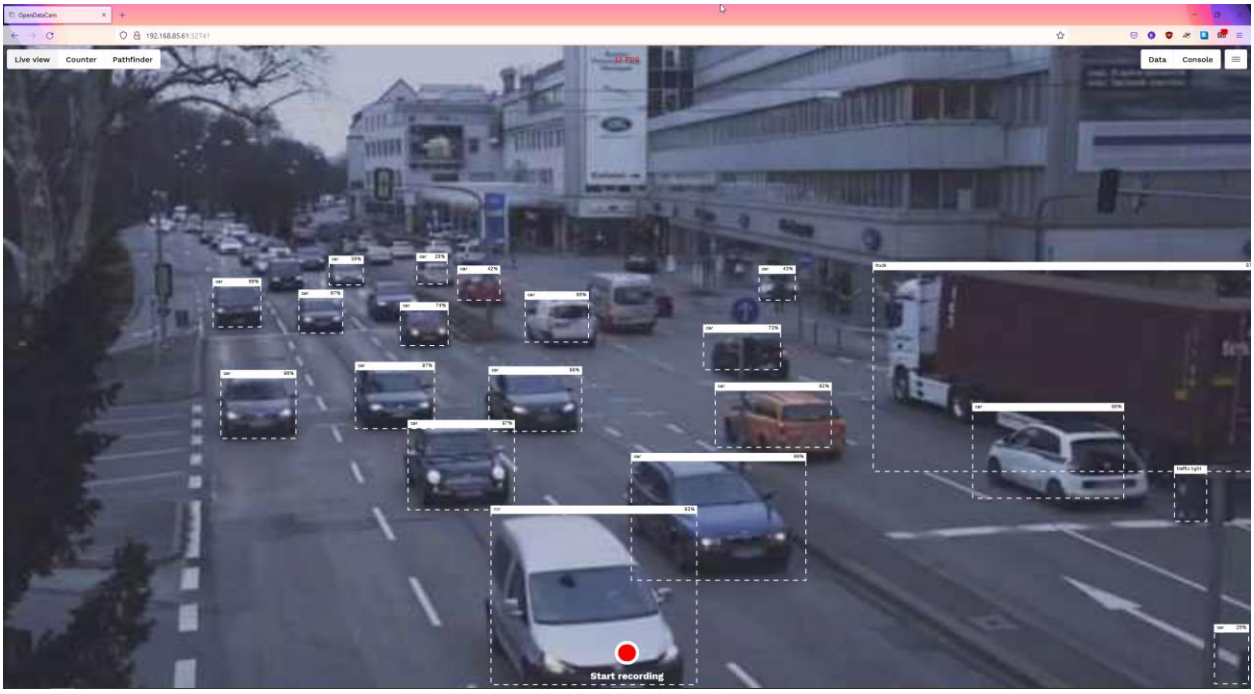
Opendatacam-mongo-pvc.yaml

opendatacam-mongo-pvc.yaml

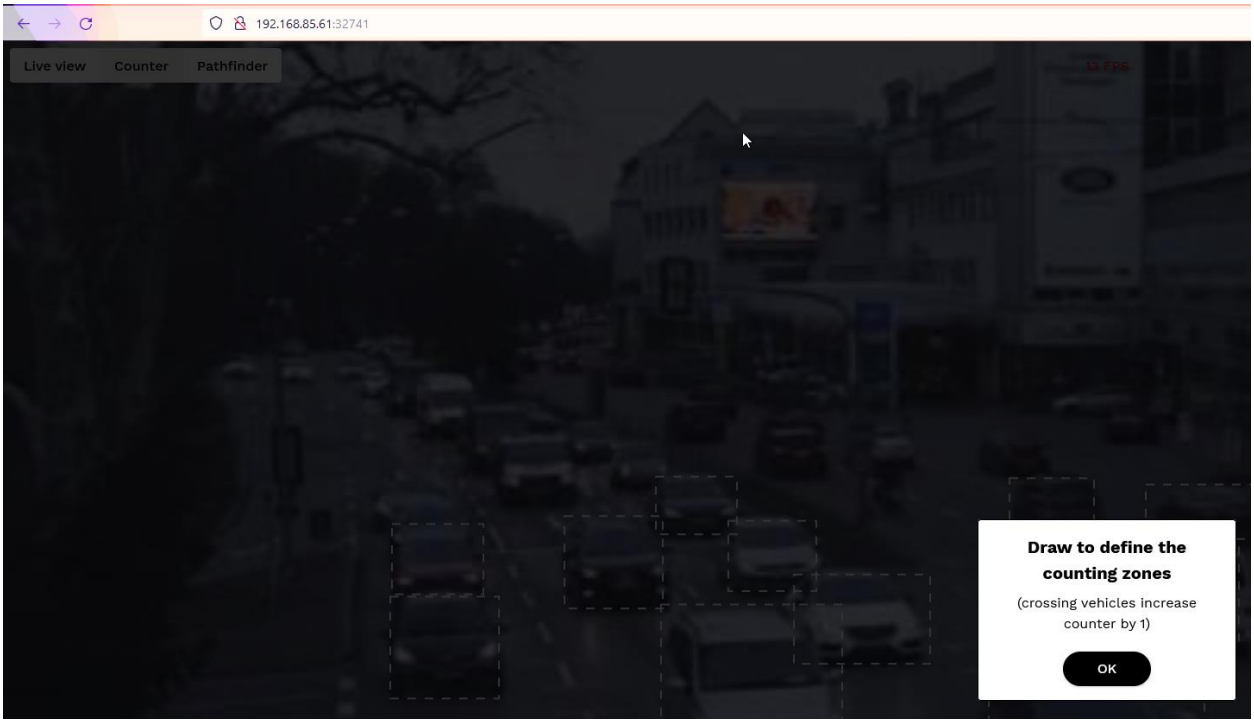
```yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: mongodb-pv-claim
  labels:
    app: opendatacam
spec:
  accessModes:
  - ReadWriteOnce
  resources:
    requests:
      storage: 10Gi
```

Screenshot of [Counter] and [Data] from Opendatacam web UI.

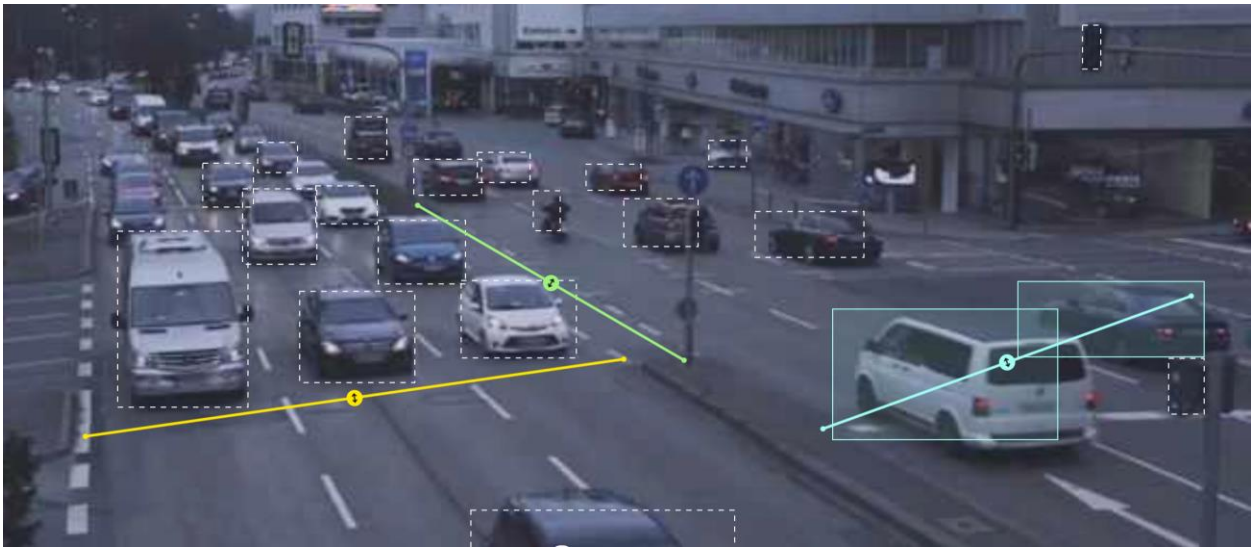
http://192.168.85.61:32741
Initial interface when loading web page:



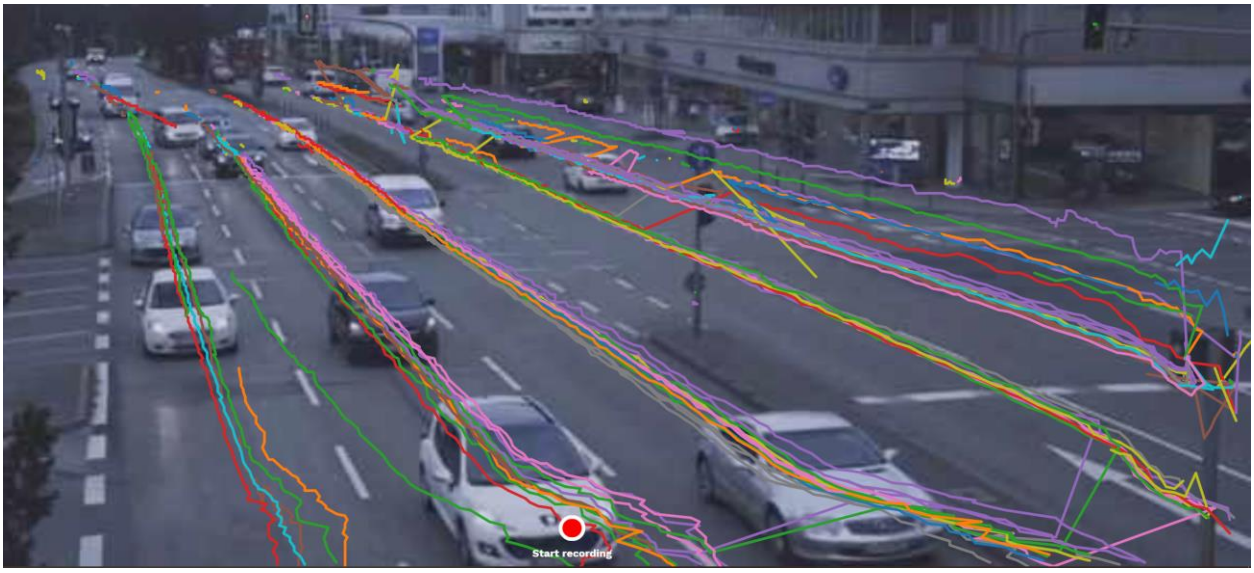
Clicking counter:



Lane-left, lane-right, and lane-middle



Pathfinder UI:

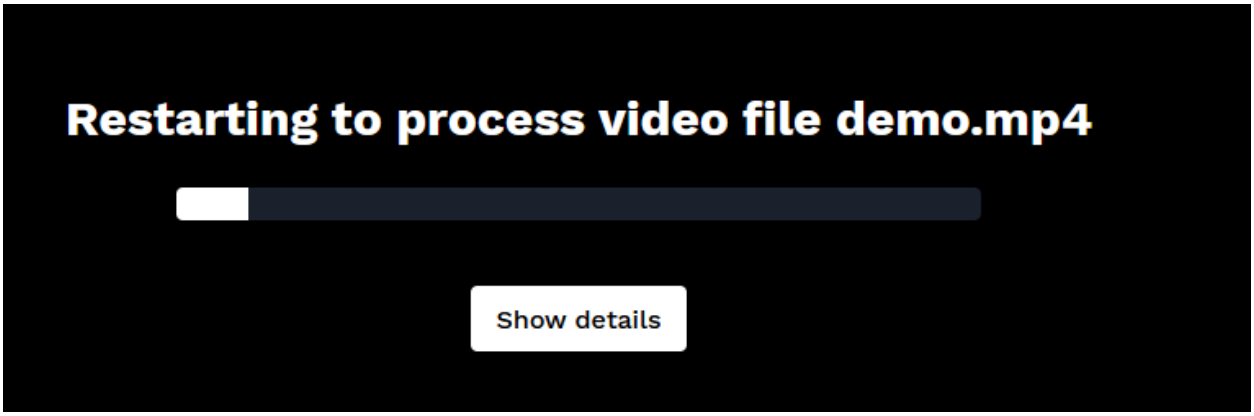


Console on next page:

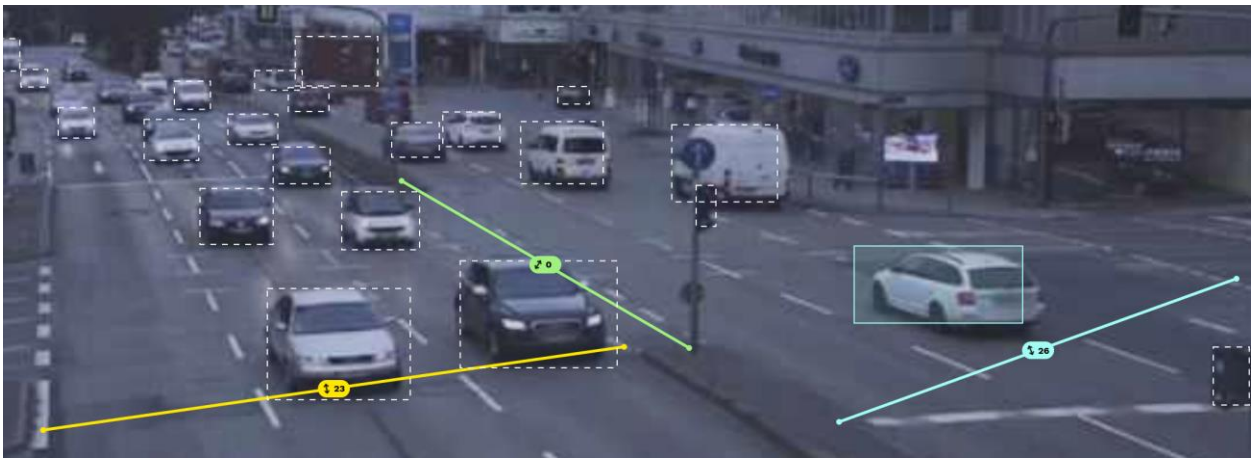
```
 124 conv    256       1 x 1/ 1     26 x  26 x 512 ->   26 x  26 x 256 0.177 BF
 125 conv    512       3 x 3/ 1     26 x  26 x 256 ->   26 x  26 x 512 1.595 BF
 126 conv    256       1 x 1/ 1     26 x  26 x 512 ->   26 x  26 x 256 0.177 BF
 127 conv    128       1 x 1/ 1     26 x  26 x 256 ->   26 x  26 x 128 0.044 BF
 128 upsample                2x    26 x  26 x 128 ->   52 x  52 x 128
 129 route  54                                       ->   52 x  52 x 256
 130 conv    128       1 x 1/ 1     52 x  52 x 256 ->   52 x  52 x 128 0.177 BF
 131 route  130 128                                  ->   52 x  52 x 256
 132 conv    128       1 x 1/ 1     52 x  52 x 256 ->   52 x  52 x 128 0.177 BF
 133 conv    256       3 x 3/ 1     52 x  52 x 128 ->   52 x  52 x 256 1.595 BF
 134 conv    128       1 x 1/ 1     52 x  52 x 256 ->   52 x  52 x 128 0.177 BF
 135 conv    256       3 x 3/ 1     52 x  52 x 128 ->   52 x  52 x 256 1.595 BF
 136 conv    128       1 x 1/ 1     52 x  52 x 256 ->   52 x  52 x 128 0.177 BF
 137 conv    256       3 x 3/ 1     52 x  52 x 128 ->   52 x  52 x 256 1.595 BF
 138 conv    255       1 x 1/ 1     52 x  52 x 256 ->   52 x  52 x 255 0.353 BF
 139 yolo
[yolo] params: iou loss: ciou (4), iou_norm: 0.07, cls_norm: 1.00, scale_x_y: 1.2
 140 route  136                                      ->   52 x  52 x 128
 141 conv    256       3 x 3/ 2     52 x  52 x 128 ->   26 x  26 x 256 0.399 BF
 142 route  141 126                                  ->   26 x  26 x 512
 143 conv    256       1 x 1/ 1     26 x  26 x 512 ->   26 x  26 x 256 0.177 BF
 144 conv    512       3 x 3/ 1     26 x  26 x 256 ->   26 x  26 x 512 1.595 BF
 145 conv    256       1 x 1/ 1     26 x  26 x 512 ->   26 x  26 x 256 0.177 BF
 146 conv    512       3 x 3/ 1     26 x  26 x 256 ->   26 x  26 x 512 1.595 BF
 147 conv    256       1 x 1/ 1     26 x  26 x 512 ->   26 x  26 x 256 0.177 BF
 148 conv    512       3 x 3/ 1     26 x  26 x 256 ->   26 x  26 x 512 1.595 BF
 149 conv    255       1 x 1/ 1     26 x  26 x 512 ->   26 x  26 x 255 0.177 BF
 150 yolo
[yolo] params: iou loss: ciou (4), iou_norm: 0.07, cls_norm: 1.00, scale_x_y: 1.1
 151 route  147                                      ->   26 x  26 x 256
 152 conv    512       3 x 3/ 2     26 x  26 x 256 ->   13 x  13 x 512 0.399 BF
 153 route  152 116                                  ->   13 x  13 x1024
 154 conv    512       1 x 1/ 1     13 x  13 x1024 ->   13 x  13 x 512 0.177 BF
 155 conv   1024       3 x 3/ 1     13 x  13 x 512 ->   13 x  13 x1024 1.595 BF
 156 conv    512       1 x 1/ 1     13 x  13 x1024 ->   13 x  13 x 512 0.177 BF
 157 conv   1024       3 x 3/ 1     13 x  13 x 512 ->   13 x  13 x1024 1.595 BF
 158 conv    512       1 x 1/ 1     13 x  13 x1024 ->   13 x  13 x 512 0.177 BF
 159 conv   1024       3 x 3/ 1     13 x  13 x 512 ->   13 x  13 x1024 1.595 BF
 160 conv    255       1 x 1/ 1     13 x  13 x1024 ->   13 x  13 x 255 0.088 BF
 161 yolo
[yolo] params: iou loss: ciou (4), iou_norm: 0.07, cls_norm: 1.00, scale_x_y: 1.0
Total BFLOPS 60.137
avg_outputs = 500162
 Allocate additional workspace_size = 52.43 MB
Loading weights from yolov4.weights...Demo
net.optimized_memory = 0
mini_batch = 1, batch = 8, time_steps = 1, train = 0
nms_kind: greedynms (1), beta = 0.600000
nms_kind: greedynms (1), beta = 0.600000
nms_kind: greedynms (1), beta = 0.600000
Done! Loaded 162 layers from weights-file
Got first message from JSONStream
setvideoresolution

 seen 64, trained: 32032 K-images (500 Kilo-batches_64)
video file: opendatacam_videos/demo.mp4
Video stream: 640 x 360
JSON_sender: new client 29
Restore counting areas
MJPG_sender: new client 31
MJPG_sender: new client 32
Error with message send by YOLO, not valid JSON
```

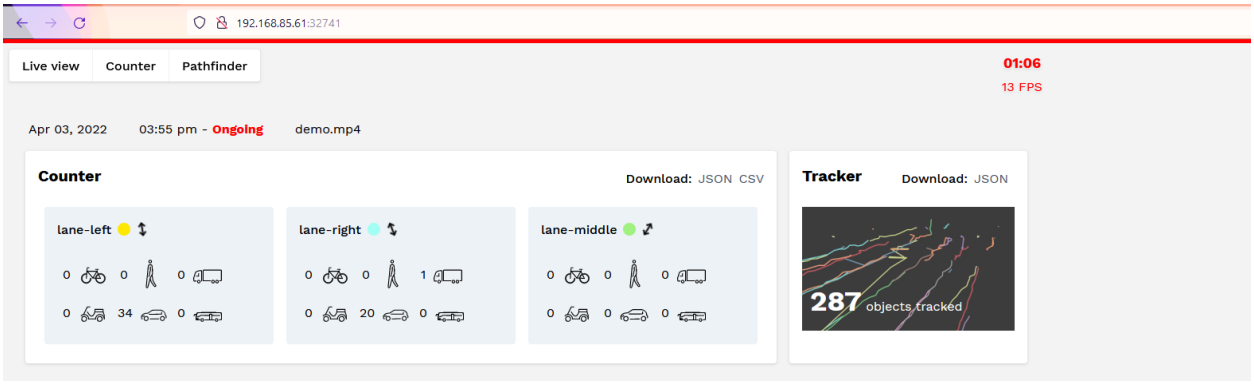
Upon hitting record:



Count:



Data UI:



Downloaded JSON of counter example: