

```
In [1]: import zipfile
import os

# Path to the uploaded archive.zip
zip_file_path = "C:/Users/SEC/Downloads/ibm/archive.zip"

# Extract the zip file to the current working directory
with zipfile.ZipFile(zip_file_path, 'r') as zip_ref:
    zip_ref.extractall()

# Check the contents of the extracted folder
os.listdir('debris-detection')
```

```
Out[1]: ['sample_submission.csv', 'test', 'train', 'train.csv', 'val', 'val.csv']
```

```
In [2]: import pandas as pd
import cv2
import os

# Load the train CSV
train_df = pd.read_csv('debris-detection/train.csv')

# Function to convert bounding box to YOLO format
def convert_bbox_to_yolo(img_width, img_height, bbox):
    x_min, y_min, x_max, y_max = bbox
    x_center = (x_min + x_max) / 2 / img_width
    y_center = (y_min + y_max) / 2 / img_height
    width = (x_max - x_min) / img_width
    height = (y_max - y_min) / img_height
    return [x_center, y_center, width, height]

# Convert each row in the CSV
for _, row in train_df.iterrows():
    img_id = row['ImageID']
    bboxes = eval(row['bboxes']) # Convert the string to list format

    # Load the corresponding image to get its dimensions
    img_path = f"debris-detection/train/{img_id}.jpg"
    if os.path.exists(img_path):
        img = cv2.imread(img_path)
        img_height, img_width, _ = img.shape

    # Create the corresponding .txt file in YOLO format
    with open(f"debris-detection/train/{img_id}.txt", 'w') as f:
        for bbox in bboxes:
            x_min, y_min, x_max, y_max = bbox
            yolo_bbox = convert_bbox_to_yolo(img_width, img_height, [x_min, y_min, x_max,
            f.write(f"0 {' '.join(map(str, yolo_bbox))}\n") # Class ID is 0 for debris
```

```
In [6]: !git clone --depth 1 https://github.com/ultralytics/yolov5.git
```

```
Cloning into 'yolov5'...
```

```
In [7]: %cd yolov5
!pip install -r requirements.txt
```

```
C:\Users\SEC\yolov5
```

```
Collecting gitpython>=3.1.30 (from -r requirements.txt (line 5))
```

```
WARNING: Retrying (Retry(total=4, connect=None, read=None, redirect=None, status=None)) after
connection broken by 'ReadTimeoutError("HTTPSConnectionPool(host='pypi.org', port=443): Read t
imed out. (read timeout=15)")': /simple/gitpython/
WARNING: Retrying (Retry(total=3, connect=None, read=None, redirect=None, status=None)) after
connection broken by 'NewConnectionError('<pip._vendor.urllib3.connection.HTTPSConnection obje
ct at 0x0000021F9ED30A00>: Failed to establish a new connection: [Errno 11001] getaddrinfo fai
led')': /simple/gitpython/
WARNING: Retrying (Retry(total=2, connect=None, read=None, redirect=None, status=None)) after
connection broken by 'NewConnectionError('<pip._vendor.urllib3.connection.HTTPSConnection obje
ct at 0x0000021F9ED30D00>: Failed to establish a new connection: [Errno 11001] getaddrinfo fai
led')': /simple/gitpython/
WARNING: Retrying (Retry(total=1, connect=None, read=None, redirect=None, status=None)) after
connection broken by 'NewConnectionError('<pip._vendor.urllib3.connection.HTTPSConnection obje
ct at 0x0000021F9ED30EE0>: Failed to establish a new connection: [Errno 11001] getaddrinfo fai
led')': /simple/gitpython/
WARNING: Retrying (Retry(total=0, connect=None, read=None, redirect=None, status=None)) after
connection broken by 'NewConnectionError('<pip._vendor.urllib3.connection.HTTPSConnection obje
ct at 0x0000021F9ED30FA0>: Failed to establish a new connection: [Errno 11001] getaddrinfo fai
led')': /simple/gitpython/
WARNING: Error parsing dependencies of colab: Expected matching RIGHT_PARENTHESIS for LEFT_PAR
ENTHESIS, after version specifier
    pytz (>=2011n)
        ~~~~~~^
```

```
Requirement already satisfied: sympy in c:\users\sec\anaconda3\lib\site-packages (from torch>=1.8.0->-r requirements.txt (line 15)) (1.13.2)
Requirement already satisfied: networkx in c:\users\sec\anaconda3\lib\site-packages (from torch>=1.8.0->-r requirements.txt (line 15)) (3.2.1)
Requirement already satisfied: Jinja2 in c:\users\sec\anaconda3\lib\site-packages (from torch>=1.8.0->-r requirements.txt (line 15)) (3.1.4)
Requirement already satisfied: fsspec in c:\users\sec\anaconda3\lib\site-packages (from torch>=1.8.0->-r requirements.txt (line 15)) (2024.6.1)
Requirement already satisfied: colorama in c:\users\sec\anaconda3\lib\site-packages (from tqdm>=4.66.3->-r requirements.txt (line 17)) (0.4.6)
Requirement already satisfied: py-cpuinfo in c:\users\sec\anaconda3\lib\site-packages (from ultralytics>=8.2.34->-r requirements.txt (line 18)) (9.0.0)
Collecting ultralytics-thop>=2.0.0 (from ultralytics>=8.2.34->-r requirements.txt (line 18))
  Downloading ultralytics_thop-2.0.9-py3-none-any.whl.metadata (9.3 kB)
Requirement already satisfied: pytz>=2020.1 in c:\users\sec\anaconda3\lib\site-packages (from pandas>=1.1.4->-r requirements.txt (line 27)) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\sec\anaconda3\lib\site-packages (from pandas>=1.1.4->-r requirements.txt (line 27)) (2023.3)
Collecting smmap<6,>=3.0.1 (from gitdb<5,>=4.0.1->gitpython>=3.1.30->-r requirements.txt (line 5))
  Downloading smmap-5.0.1-py3-none-any.whl.metadata (4.3 kB)
Requirement already satisfied: zipp>=3.1.0 in c:\users\sec\anaconda3\lib\site-packages (from importlib-resources>=3.2.0->matplotlib>=3.3->-r requirements.txt (line 6)) (3.20.2)
Requirement already satisfied: six>=1.5 in c:\users\sec\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib>=3.3->-r requirements.txt (line 6)) (1.16.0)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\sec\anaconda3\lib\site-packages (from Jinja2->torch>=1.8.0->-r requirements.txt (line 15)) (2.1.3)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in c:\users\sec\anaconda3\lib\site-packages (from sympy->torch>=1.8.0->-r requirements.txt (line 15)) (1.3.0)
Downloading GitPython-3.1.43-py3-none-any.whl (207 kB)
Downloading ultralytics-8.3.17-py3-none-any.whl (876 kB)
----- 0.0/876.6 kB ? eta -:--:--
----- 0.0/876.6 kB ? eta -:--:--
----- 0.0/876.6 kB ? eta -:--:--
----- 0.0/876.6 kB ? eta -:--:--
----- 0.0/876.6 kB ? eta -:--:--
----- 0.0/876.6 kB ? eta -:--:--
----- 0.0/876.6 kB ? eta -:--:--
----- 0.0/876.6 kB ? eta -:--:--
----- 0.0/876.6 kB ? eta -:--:--
----- 0.0/876.6 kB ? eta -:--:--
----- 262.1/876.6 kB ? eta -:--:--
----- 262.1/876.6 kB ? eta -:--:--
----- 262.1/876.6 kB ? eta -:--:--
----- 262.1/876.6 kB ? eta -:--:--
----- 262.1/876.6 kB ? eta -:--:--
----- 262.1/876.6 kB ? eta -:--:--
----- 262.1/876.6 kB ? eta -:--:--
----- 262.1/876.6 kB ? eta -:--:--
----- 524.3/876.6 kB 153.9 kB/s eta 0:00:03
----- 524.3/876.6 kB 153.9 kB/s eta 0:00:03
----- 524.3/876.6 kB 153.9 kB/s eta 0:00:03
----- 524.3/876.6 kB 153.9 kB/s eta 0:00:03
----- 524.3/876.6 kB 153.9 kB/s eta 0:00:03
----- 524.3/876.6 kB 153.9 kB/s eta 0:00:03
----- 786.4/876.6 kB 172.1 kB/s eta 0:00:01
----- 786.4/876.6 kB 172.1 kB/s eta 0:00:01
----- 786.4/876.6 kB 172.1 kB/s eta 0:00:01
----- 786.4/876.6 kB 172.1 kB/s eta 0:00:01
----- 876.6/876.6 kB 165.9 kB/s eta 0:00:00
Downloading gitdb-4.0.11-py3-none-any.whl (62 kB)
Downloading ultralytics_thop-2.0.9-py3-none-any.whl (26 kB)
Downloading smmap-5.0.1-py3-none-any.whl (24 kB)
Installing collected packages: smmap, gitdb, ultralytics-thop, gitpython, ultralytics
```

```
Attempting uninstall: ultralytics
Found existing installation: ultralytics 8.1.27
Uninstalling ultralytics-8.1.27:
  Successfully uninstalled ultralytics-8.1.27
Successfully installed gitdb-4.0.11 gitpython-3.1.43 smmap-5.0.1 ultralytics-8.3.17 ultralytic
s-thop-2.0.9
```

```
In [14]: import pandas as pd
import cv2
import os

# Define paths for train and validation datasets
train_csv_path = 'C:/Users/SEC/Downloads/ibm/debris-detection/train.csv'
train_img_dir = 'C:/Users/SEC/Downloads/ibm/debris-detection/train'
val_csv_path = 'C:/Users/SEC/Downloads/ibm/debris-detection/val.csv'
val_img_dir = 'C:/Users/SEC/Downloads/ibm/debris-detection/val'

# Create directories for the preprocessed dataset (if they don't already exist)
preprocessed_base_dir = 'C:/Users/SEC/Downloads/ibm/preprocessed'
os.makedirs(f'{preprocessed_base_dir}/train/images', exist_ok=True)
os.makedirs(f'{preprocessed_base_dir}/train/labels', exist_ok=True)
os.makedirs(f'{preprocessed_base_dir}/val/images', exist_ok=True)
os.makedirs(f'{preprocessed_base_dir}/val/labels', exist_ok=True)

# Function to convert bounding box to YOLO format
def convert_bbox_to_yolo(img_width, img_height, bbox):
    x_min, y_min, x_max, y_max = bbox
    x_center = (x_min + x_max) / 2 / img_width
    y_center = (y_min + y_max) / 2 / img_height
    width = (x_max - x_min) / img_width
    height = (y_max - y_min) / img_height
    return [x_center, y_center, width, height]

# Preprocess dataset (generalized for both train and val sets)
def preprocess_dataset(csv_path, img_dir, save_dir):
    df = pd.read_csv(csv_path)
    for _, row in df.iterrows():
        img_id = row['ImageID']
        bboxes = eval(row['bboxes']) # Convert the string to List format

        # Load the corresponding image
        img_path = f"{img_dir}/{img_id}.jpg"
        if os.path.exists(img_path):
            img = cv2.imread(img_path)
            img_height, img_width, _ = img.shape

            # Save the image to the preprocessed dataset directory
            new_img_path = f"{save_dir}/images/{img_id}.jpg"
            cv2.imwrite(new_img_path, img)

            # Create the corresponding .txt file in YOLO format
            label_path = f"{save_dir}/labels/{img_id}.txt"
            with open(label_path, 'w') as f:
                for bbox in bboxes:
                    yolo_bbox = convert_bbox_to_yolo(img_width, img_height, bbox)
                    f.write(f"0 {' '.join(map(str, yolo_bbox))}\n") # Class ID is 0 for debris

# Preprocess both train and validation datasets
preprocess_dataset(train_csv_path, train_img_dir, f'{preprocessed_base_dir}/train')
preprocess_dataset(val_csv_path, val_img_dir, f'{preprocessed_base_dir}/val')

print("Preprocessing completed!")
```

Preprocessing completed!

In [20]: !python train.py --img 640 --batch 16 --epochs 50 --data C:/Users/SEC/yolov5/data.yaml --weig

```
In [21]: import torch
import cv2
import matplotlib.pyplot as plt

# Load your trained model
model_path = 'C:/Users/SEC/yolov5/runs/train/exp/weights/best.pt' # Adjust the path if needed
model = torch.hub.load('ultralytics/yolov5', 'custom', path=model_path, force_reload=True)

# Path to the test image
test_image_path = "C:/Users/SEC/Downloads/ibm/debris-detection/test/8.jpg"

# Perform inference
results = model(test_image_path)

# Results
results.print() # Print results to console
results.show()  # Display the image with bounding boxes
results.save()  # Save the results

# Optionally, display the image with matplotlib
img = cv2.imread(test_image_path)
plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
plt.axis('off') # Hide axis
plt.show()
```

C:\Users\SEC\anaconda3\lib\site-packages\torch\hub.py:294: UserWarning: You are about to download and run code from an untrusted repository. In a future release, this won't be allowed. To add the repository to your trusted list, change the command to {calling_fn}(..., trust_repo=False) and a command prompt will appear asking for an explicit confirmation of trust, or load(..., trust_repo=True), which will assume that the prompt is to be answered with 'yes'. You can also use load(..., trust_repo='check') which will only prompt for confirmation if the repo is not already trusted. This will eventually be the default behaviour

warnings.warn(
Downloading: "https://github.com/ultralytics/yolov5/zipball/master" to C:\Users\SEC\.cache\torch\hub\master.zip

WARNING Python>=3.10 is required, but Python==3.9.20 is currently installed

YOLOv5 2024-10-20 Python-3.9.20 torch-2.2.1+cpu CPU

```

109
(...)
133
134
--> 135     return _create(path, autoshape=autoshape, verbose=_verbose, device=device)

```

File ~\.cache\torch\hub\ultralytics_yolov5_master\hubconf.py:103, in _create(name, pretrained, channels, classes, autoshape, verbose, device)

```

101 help_url = "https://docs.ultralytics.com/yolov5/tutorials/pytorch_hub_model_loading"
102 s = f"{e}. Cache may be out of date, try `force_reload=True` or see {help_url} for help."

```

```
--> 103 raise Exception(s) from e
```

Exception: [Errno 2] No such file or directory: 'C:\\Users\\SEC\\yolov5\\runs\\train\\exp\\weights\\best.pt'. Cache may be out of date, try `force_reload=True` or see https://docs.ultralytics.com/yolov5/tutorials/pytorch_hub_model_loading for help.

```

In [24]: import torch
        from PIL import Image
        import cv2

        # Load the trained model
        model = torch.hub.load('ultralytics/yolov5', 'custom', path='C:/Users/SEC/runs/train/experiment1/weights/best.pt')

        # Load an image or video to test
        image_path = 'C:/Users/SEC/Downloads/ibm/debris-detection/test/8.jpg'
        img = Image.open(image_path)

        # Perform inference
        results = model(img)

        # Display results
        results.show()

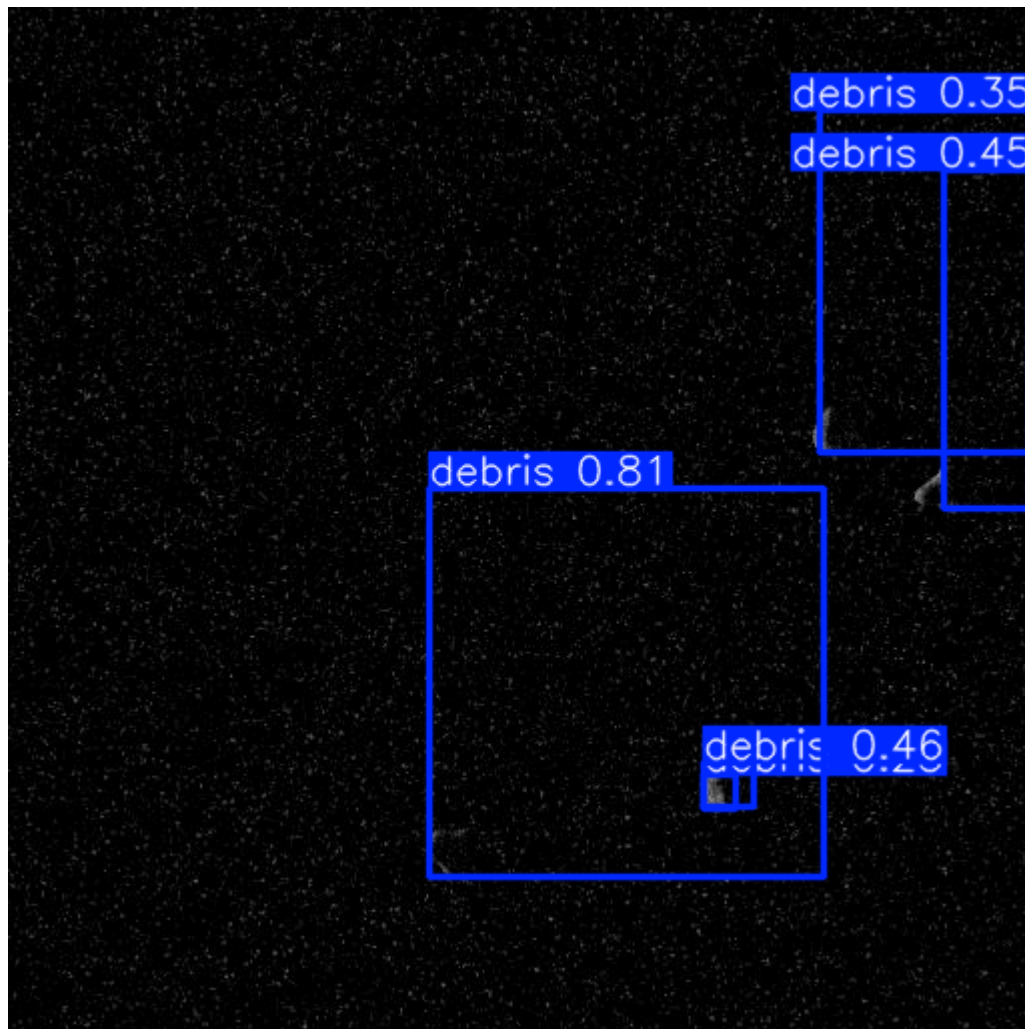
```

Using cache found in C:\Users\SEC\.cache\torch\hub\ultralytics_yolov5_master
YOLOv5 2024-10-20 Python-3.9.20 torch-2.2.1+cpu CPU

Fusing layers...

Model summary: 157 layers, 7012822 parameters, 0 gradients, 15.8 GFLOPs

Adding AutoShape...



In [31]: `!pip install filterpy`

Collecting filterpy

WARNING: Error parsing dependencies of colab: Expected matching RIGHT_PARENTHESIS for LEFT_PARENTHESIS, after version specifier

pytz (>=2011n)

~~~~~^

```

Downloading filterpy-1.4.5.zip (177 kB)
Preparing metadata (setup.py): started
Preparing metadata (setup.py): finished with status 'done'
Requirement already satisfied: numpy in c:\users\sec\anaconda3\lib\site-packages (from filterpy) (1.24.4)
Requirement already satisfied: scipy in c:\users\sec\anaconda3\lib\site-packages (from filterpy) (1.13.1)
Requirement already satisfied: matplotlib in c:\users\sec\anaconda3\lib\site-packages (from filterpy) (3.9.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\sec\anaconda3\lib\site-packages (from matplotlib->filterpy) (1.2.0)
Requirement already satisfied: cycler>=0.10 in c:\users\sec\anaconda3\lib\site-packages (from matplotlib->filterpy) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\sec\anaconda3\lib\site-packages (from matplotlib->filterpy) (4.51.0)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\sec\anaconda3\lib\site-packages (from matplotlib->filterpy) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\sec\anaconda3\lib\site-packages (from matplotlib->filterpy) (24.1)
Requirement already satisfied: pillow>=8 in c:\users\sec\anaconda3\lib\site-packages (from matplotlib->filterpy) (10.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\sec\anaconda3\lib\site-packages (from matplotlib->filterpy) (3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\sec\anaconda3\lib\site-packages (from matplotlib->filterpy) (2.9.0.post0)
Requirement already satisfied: importlib-resources>=3.2.0 in c:\users\sec\anaconda3\lib\site-packages (from matplotlib->filterpy) (6.4.0)
Requirement already satisfied: zipp>=3.1.0 in c:\users\sec\anaconda3\lib\site-packages (from importlib-resources>=3.2.0->matplotlib->filterpy) (3.20.2)
Requirement already satisfied: six>=1.5 in c:\users\sec\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib->filterpy) (1.16.0)
Building wheels for collected packages: filterpy
  Building wheel for filterpy (setup.py): started
  Building wheel for filterpy (setup.py): finished with status 'done'
  Created wheel for filterpy: filename=filterpy-1.4.5-py3-none-any.whl size=110541 sha256=bec6d779cd981082ce808aa034ce0afa554b029abceea563b36365944ac4820a
  Stored in directory: c:\users\sec\appdata\local\pip\cache\wheels\53\e6\de\ae09ea01e923aaf88b9f8c7c44329e857b2c1a31901167e55e6
Successfully built filterpy
Installing collected packages: filterpy
Successfully installed filterpy-1.4.5

```

```

In [1]: import cv2
import torch
import numpy as np

# Load YOLOv5 model (pre-trained or custom weights)
model = torch.hub.load('ultralytics/yolov5', 'custom', path='C:/Users/SEC/runs/train/experiment1/weights/best.pt')

# Define the Kalman filter for continuous tracking
class KalmanFilter:
    def __init__(self):
        # Define Kalman filter
        self.kf = cv2.KalmanFilter(4, 2)
        self.kf.measurementMatrix = np.array([[1, 0, 0, 0], [0, 1, 0, 0]], np.float32)
        self.kf.transitionMatrix = np.array([[1, 0, 1, 0], [0, 1, 0, 1], [0, 0, 1, 0], [0, 0, 0, 1]], np.float32)
        self.kf.processNoiseCov = np.eye(4, dtype=np.float32) * 0.03 # Lower noise for smoother tracking

    def predict(self, x, y):
        '''This function estimates the object's next position using the Kalman filter'''
        measurement = np.array([[np.float32(x)], [np.float32(y)]]).T
        self.kf.correct(measurement)
        prediction = self.kf.predict()

```



```

        return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates

# Instantiate Kalman Filter
kalman_filter = KalmanFilter()

# Define video input
video_path = "C:/Users/SEC/Downloads/ibm/sample video.mp4"
cap = cv2.VideoCapture(video_path)

# Video writer to save the output
output_video_path = "C:/Users/SEC/Downloads/ibm/output_with_kalman_continuous.mp4"
frame_width = int(cap.get(3))
frame_height = int(cap.get(4))
out = cv2.VideoWriter(output_video_path, cv2.VideoWriter_fourcc(*'mp4v'), 20, (frame_width, f

# Store previous points to draw the path
previous_points = []

while cap.isOpened():
    ret, frame = cap.read()
    if not ret:
        break

    # Run YOLO model on the frame
    results = model(frame)

    # Get detected debris bounding boxes
    for *xyxy, conf, cls in results.xyxy[0]:
        x_min, y_min, x_max, y_max = map(int, xyxy)

        # Calculate center of the detected debris
        center_x = (x_min + x_max) // 2
        center_y = (y_min + y_max) // 2

        # Predict the next position using Kalman Filter
        predicted_x, predicted_y = kalman_filter.predict(center_x, center_y)

        # Draw the detected debris bounding box
        cv2.rectangle(frame, (x_min, y_min), (x_max, y_max), (255, 0, 0), 2)

        # Store the current predicted point
        previous_points.append((predicted_x, predicted_y))

        # Draw the continuous predicted path with lines
        for i in range(1, len(previous_points)):
            if previous_points[i - 1] is None or previous_points[i] is None:
                continue
            cv2.line(frame, previous_points[i - 1], previous_points[i], (0, 255, 0), 2) # Dr

    # Save frame with predictions
    out.write(frame)

    # Display the frame (optional)
    cv2.imshow('Debris Tracking with Path Prediction', frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

# Release video capture and writer
cap.release()
out.release()
cv2.destroyAllWindows()

```

Fusing layers...

Model summary: 157 layers, 7012822 parameters, 0 gradients, 15.8 GFLOPs

Adding AutoShape...

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

C:\Users\SEC\AppData\Local\Temp\ipykernel\_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)

```
C:\Users\SEC\AppData\Local\Temp\ipykernel_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)
```

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

```
C:\Users\SEC\AppData\Local\Temp\ipykernel_22608\1182199763.py:22: DeprecationWarning: Conversion of an array with ndim > 0 to a scalar is deprecated, and will error in future. Ensure you extract a single element from your array before performing this operation. (Deprecated NumPy 1.25.)
```

```
    return int(prediction[0]), int(prediction[1]) # Return predicted X, Y coordinates
```

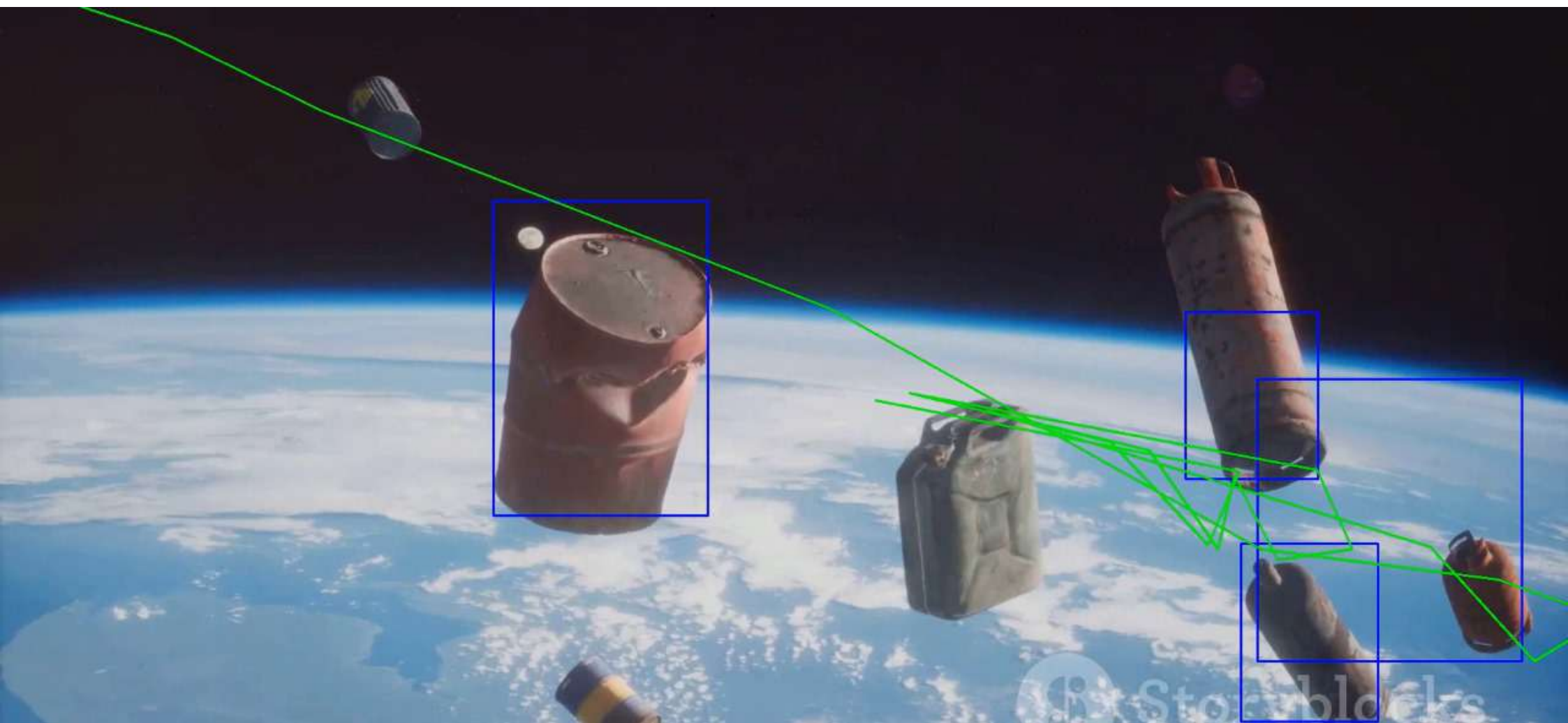
In [ ]:

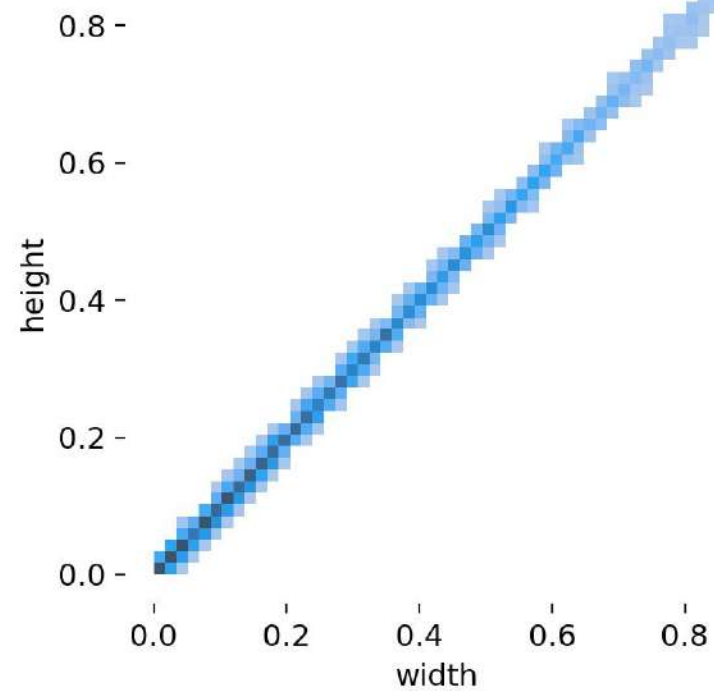
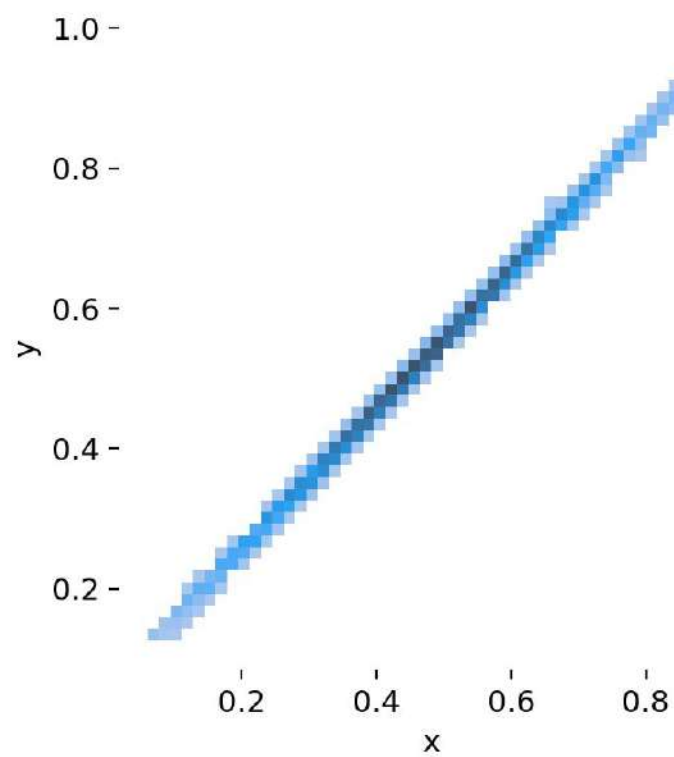
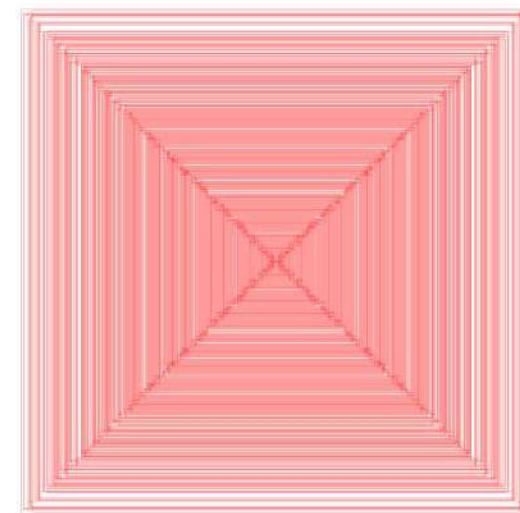
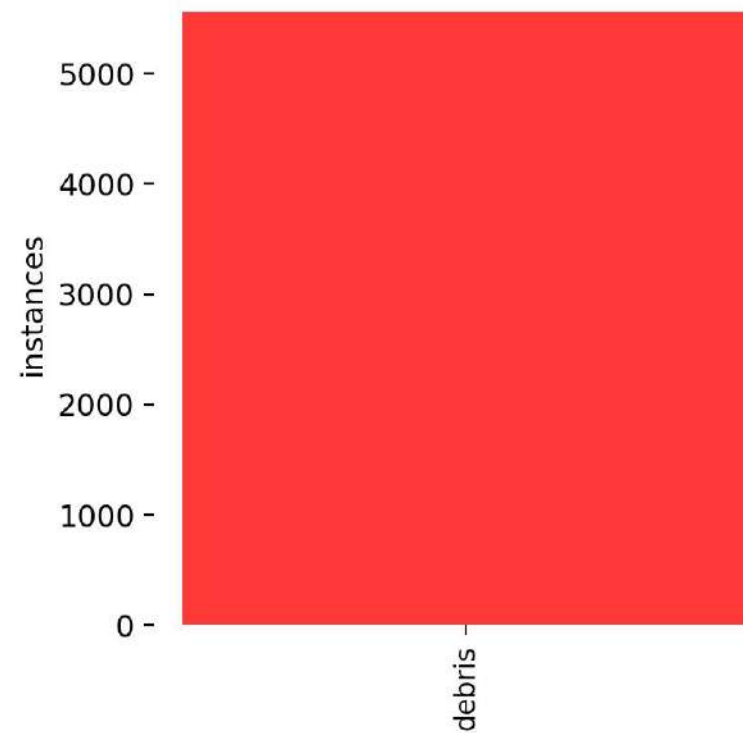
```
:\\Users\\SEC\\yolov5\\train.py:412: FutureWarning: `torch.cuda.amp.autocast(args...)` is deprecated. Please use `torch.
amp.autocast('cuda', args...)` instead.
  with torch.cuda.amp.autocast(amp):
    4/4      0G    0.04584    0.03194      0      105      640: 97%|██████████| 104/107 [15:27<00:23, C
:\\Users\\SEC\\yolov5\\train.py:412: FutureWarning: `torch.cuda.amp.autocast(args...)` is deprecated. Please use `torch.
amp.autocast('cuda', args...)` instead.
  with torch.cuda.amp.autocast(amp):
    4/4      0G    0.04581    0.03197      0      93      640: 98%|██████████| 105/107 [15:35<00:15, C
:\\Users\\SEC\\yolov5\\train.py:412: FutureWarning: `torch.cuda.amp.autocast(args...)` is deprecated. Please use `torch.
amp.autocast('cuda', args...)` instead.
  with torch.cuda.amp.autocast(amp):
    4/4      0G    0.04579    0.03197      0      89      640: 99%|██████████| 106/107 [15:43<00:07, C
:\\Users\\SEC\\yolov5\\train.py:412: FutureWarning: `torch.cuda.amp.autocast(args...)` is deprecated. Please use `torch.
amp.autocast('cuda', args...)` instead.
  with torch.cuda.amp.autocast(amp):
    4/4      0G    0.04574    0.03195      0      73      640: 100%|██████████| 107/107 [15:51<00:00,
      Class      Images  Instances      P      R      mAP50      mAP50-95: 100%|██████████| 5/5 [00:29<0
      all        158      522      0.928      0.739      0.871      0.546

5 epochs completed in 1.476 hours.
Optimizer stripped from runs\\train\\experiment2\\weights\\last.pt, 14.4MB
Optimizer stripped from runs\\train\\experiment2\\weights\\best.pt, 14.4MB

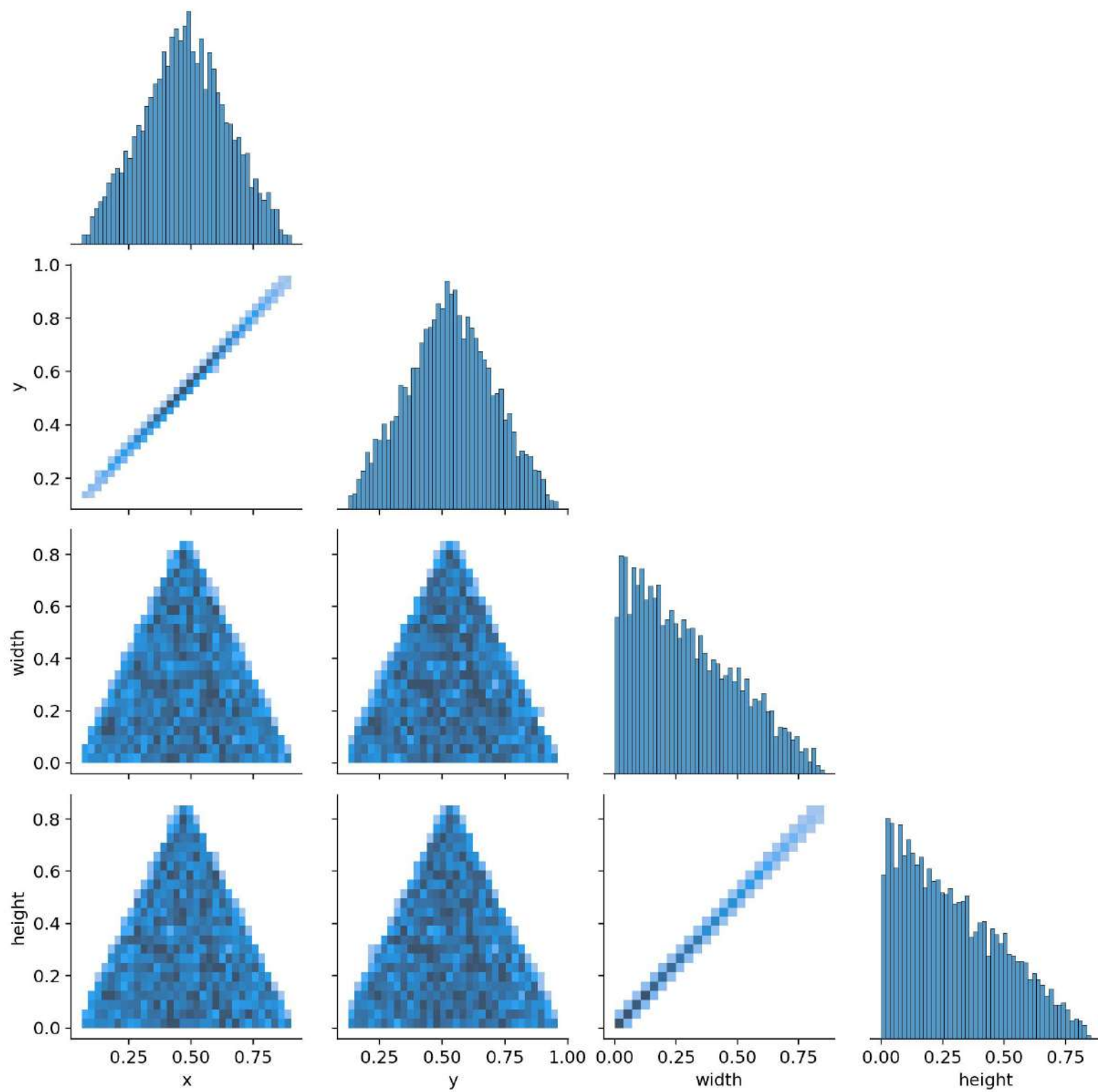
Validating runs\\train\\experiment2\\weights\\best.pt...
Fusing layers...
Model summary: 157 layers, 7012822 parameters, 0 gradients, 15.8 GFLOPs
      Class      Images  Instances      P      R      mAP50      mAP50-95: 100%|██████████| 5/5 [00:27<0
      all        158      522      0.928      0.739      0.871      0.546

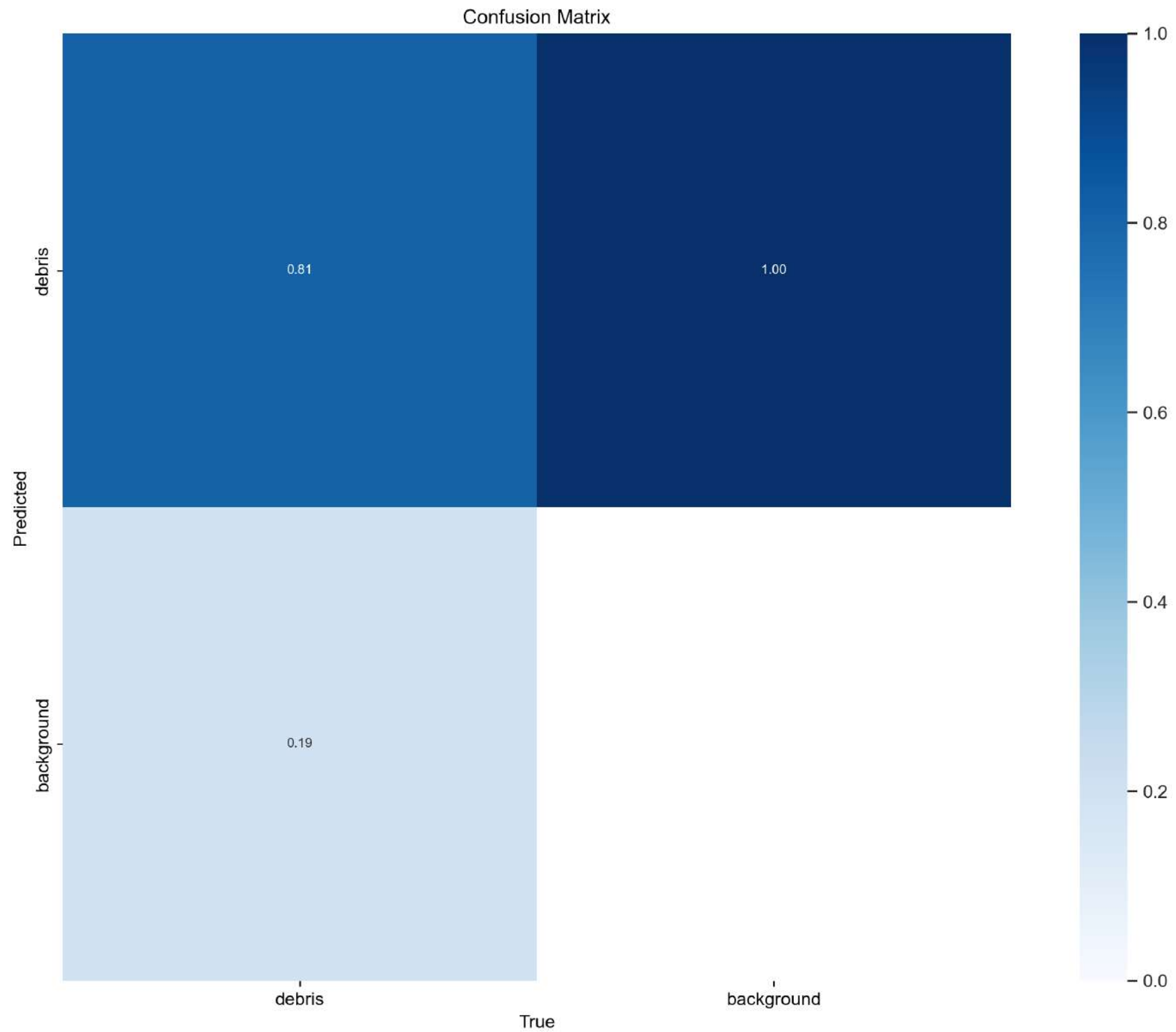
Results saved to runs\\train\\experiment2
PS C:\\Users\\SEC> |
```



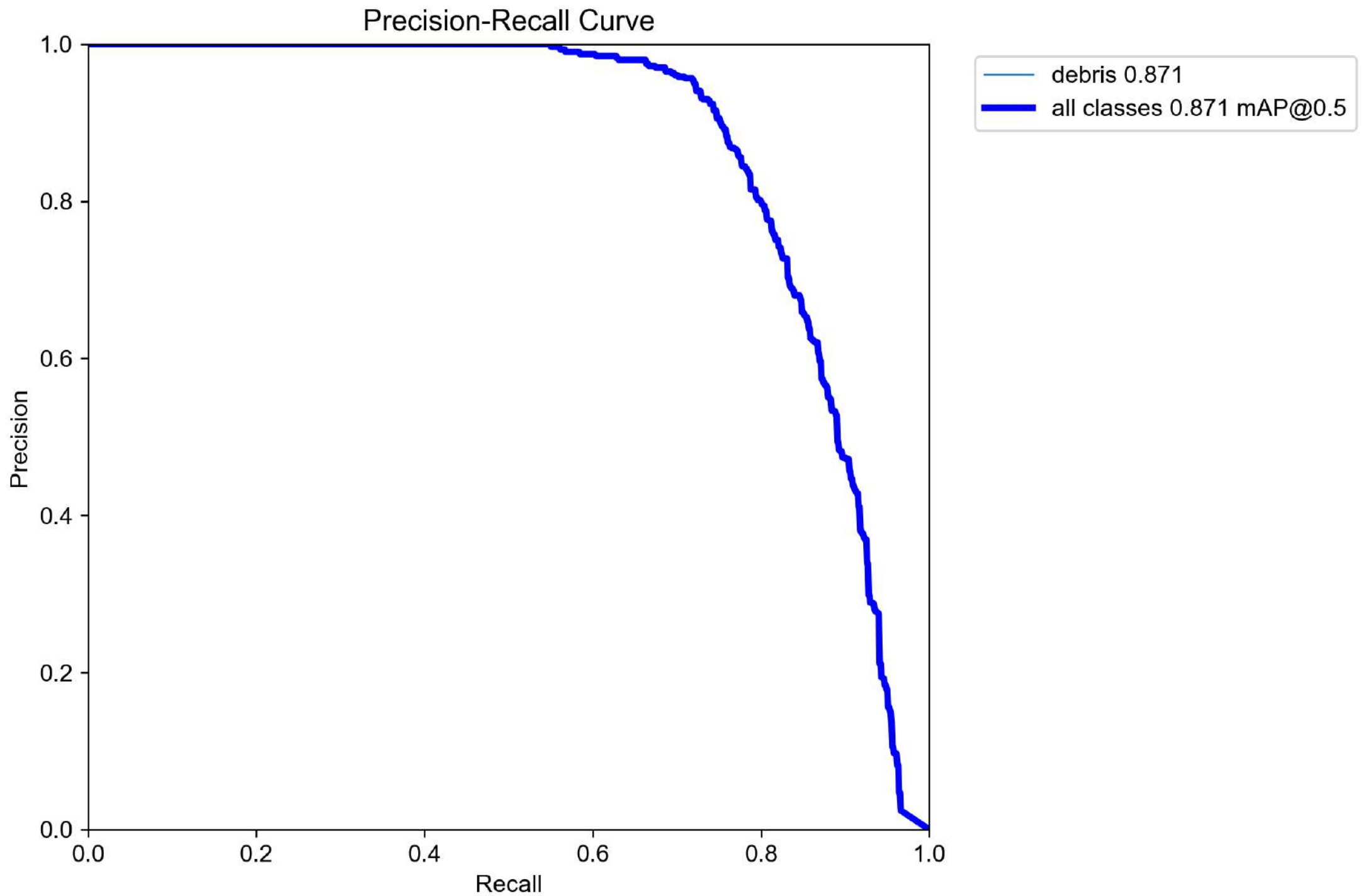




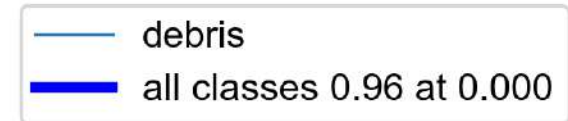
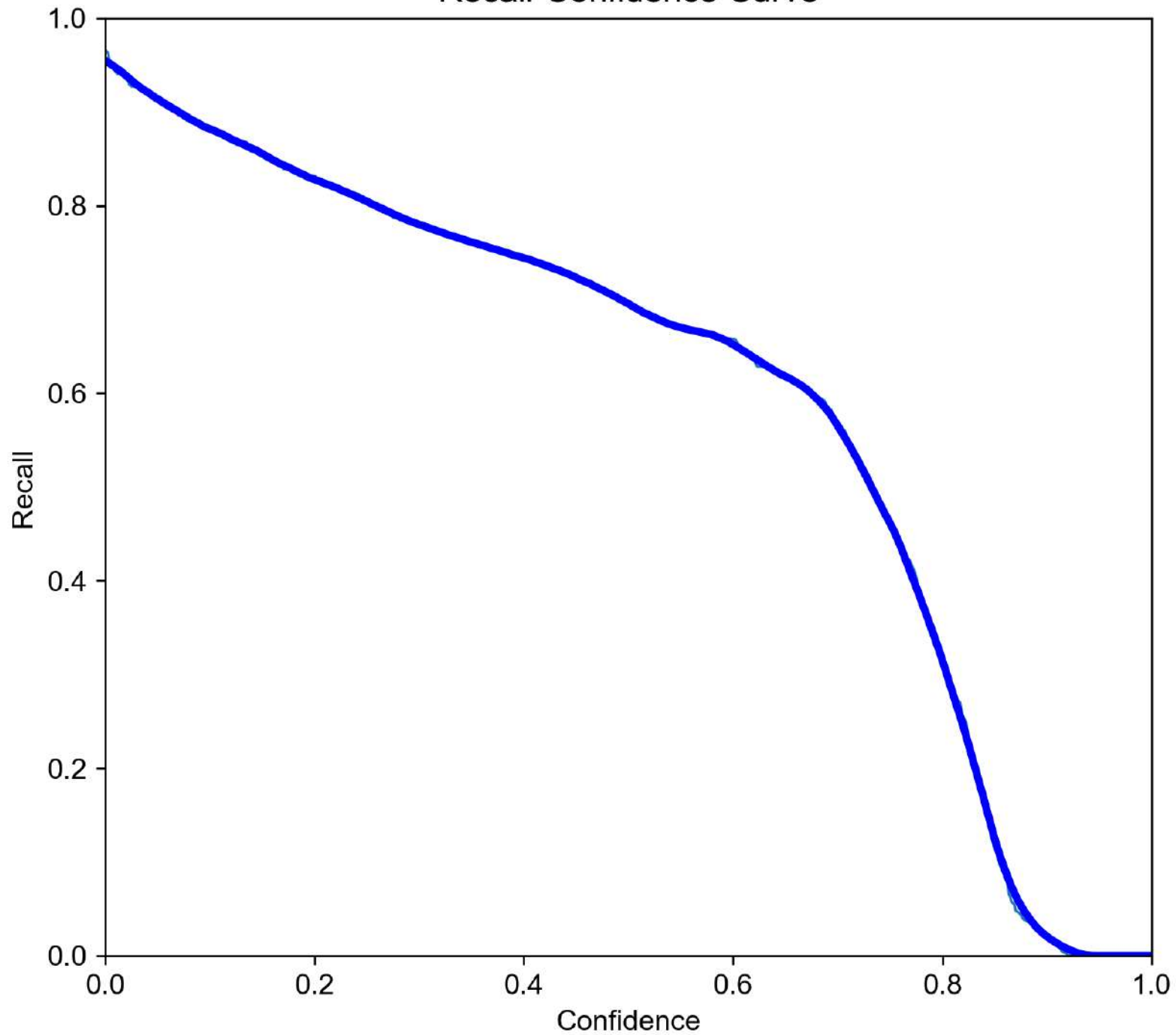








Recall-Confidence Curve



F1-Confidence Curve

