

🚀 Garbage Detection for Water Cleaning Robot using Tuned YOLOv8 Model

Momentum MU

Welcome to our project!

This work is part of **Momentum MU Team** at **Mansoura National University**, focusing on developing an intelligent **Water Cleaning Robot**. 🚶



In this notebook, we demonstrate how we trained and tuned a **YOLOv8** model for **garbage detection** to be deployed directly on our robot.

The robot will navigate and identify waste objects floating on water surfaces, enabling efficient and automated cleaning.

📸 Momentum MU Cleaning Robot



🛠 Project Highlights:

- **Model:** YOLOv8 (custom-tuned for garbage detection)
- **Platform:** ESP32 WROOM + Robotic Unit
- **Purpose:** Detect and localize garbage items to assist in water cleaning.
- **Team:** Momentum MU Team - Mansoura University
- **Robot Name:** Water Cleaning Robot

```
!ls /kaggle/input/garbage-data
```

```
→ data.yaml README.dataset.txt README.roboflow.txt test train valid
```

```
!pip install ultralytics
```



```

Downloading nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl (21.1 MB)
    21.1/21.1 MB 81.1 MB/s eta 0:00:00:00:0100:01
Downloading ultralytics_thop-2.0.14-py3-none-any.whl (26 kB)
Installing collected packages: nvidia-nvjitlink-cu12, nvidia-curand-cu12, nvidia-cufft-cu12, nvidia-cublas-cu12, nvidia-cusparse-cu12, nvidia-cudnn-cu12, nvidia-cusolver-cu12
  Attempting uninstall: nvidia-nvjitlink-cu12
    Found existing installation: nvidia-nvjitlink-cu12 12.8.93
    Uninstalling nvidia-nvjitlink-cu12-12.8.93:
      Successfully uninstalled nvidia-nvjitlink-cu12-12.8.93
  Attempting uninstall: nvidia-curand-cu12
    Found existing installation: nvidia-curand-cu12 10.3.9.90
    Uninstalling nvidia-curand-cu12-10.3.9.90:
      Successfully uninstalled nvidia-curand-cu12-10.3.9.90
  Attempting uninstall: nvidia-cufft-cu12
    Found existing installation: nvidia-cufft-cu12 11.3.3.83
    Uninstalling nvidia-cufft-cu12-11.3.3.83:
      Successfully uninstalled nvidia-cufft-cu12-11.3.3.83
  Attempting uninstall: nvidia-cublas-cu12
    Found existing installation: nvidia-cublas-cu12 12.8.4.1
    Uninstalling nvidia-cublas-cu12-12.8.4.1:
      Successfully uninstalled nvidia-cublas-cu12-12.8.4.1
  Attempting uninstall: nvidia-cusparse-cu12
    Found existing installation: nvidia-cusparse-cu12 12.5.8.93
    Uninstalling nvidia-cusparse-cu12-12.5.8.93:
      Successfully uninstalled nvidia-cusparse-cu12-12.5.8.93
  Attempting uninstall: nvidia-cudnn-cu12
    Found existing installation: nvidia-cudnn-cu12 9.3.0.75
    Uninstalling nvidia-cudnn-cu12-9.3.0.75:
      Successfully uninstalled nvidia-cudnn-cu12-9.3.0.75
  Attempting uninstall: nvidia-cusolver-cu12
    Found existing installation: nvidia-cusolver-cu12 11.7.3.90
    Uninstalling nvidia-cusolver-cu12-11.7.3.90:
      Successfully uninstalled nvidia-cusolver-cu12-11.7.3.90
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source of the following dependency conflicts
pylibcugraph-cu12 24.12.0 requires pylibraft-cu12==24.12.*, but you have pylibraft-cu12 25.2.0 which is incompatible.
pylibcugraph-cu12 24.12.0 requires rmm-cu12==24.12.*, but you have rmm-cu12 25.2.0 which is incompatible.
Successfully installed nvidia-cublas-cu12-12.4.5.8 nvidia-cudnn-cu12-9.1.0.70 nvidia-cufft-cu12-11.2.1.3 nvidia-curand-cu12-10.3.5.147 nvidia-cusolver-cu12-11.6.1.9 nvidia-c

```

```
!cp -r /kaggle/input/garbage-data /kaggle/working/garbage-data
```

⌄ Explor The Data

```

import matplotlib.pyplot as plt
import matplotlib.image as mpimg

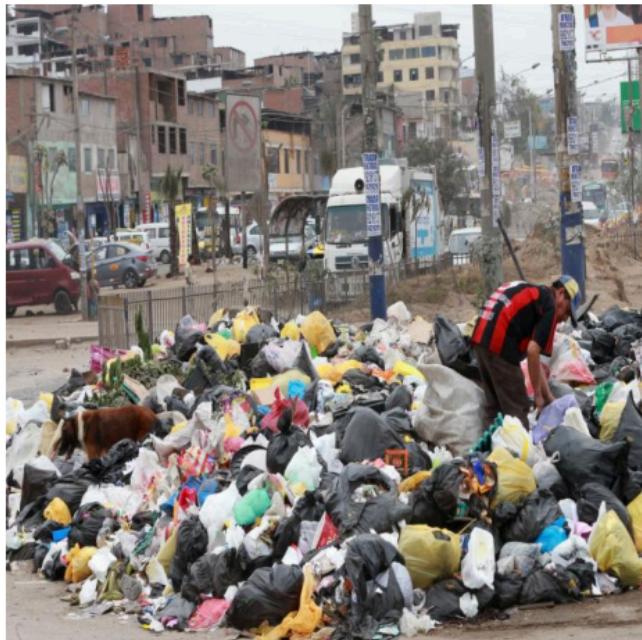
# Load your image
img = mpimg.imread('/kaggle/working/garbage-data/train/images/000545081W.jpg.rf.feca471c9b6a6e19bb78d8a8dd509298.jpg')

# Plot the image
plt.figure(figsize=(8, 6)) # you can change the size if you want
plt.imshow(img)
plt.axis('off') # Hide axis
plt.title('Sample Garbage Image', fontsize=16)
plt.show()

```



Sample Garbage Image



```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg

# Load your five images
img1 = mpimg.imread('/kaggle/working/garbage-data/train/images/000568727W.jpg.rf.10438490dc09a5e1da5dad923824a637.jpg')
img2 = mpimg.imread('/kaggle/working/garbage-data/valid/images/004633f2-679f-11e5-b0e3-40f2e96c8ad8.jpg.rf.c7fb6b39d01d029e82c697391873876d.jpg')
img3 = mpimg.imread('/kaggle/working/garbage-data/valid/images/000273202W.jpg.rf.2a0c7f191a80ad024b90ce66a2cb69c0.jpg')
img4 = mpimg.imread('/kaggle/working/garbage-data/train/images/05-_RESAH.jpg.rf.88fa789ca32b7e2debd198713f1debc3.jpg')
img5 = mpimg.imread('/kaggle/working/garbage-data/train/images/-de-s-enfoncer-tres-loin-pour-tomber-sur-des-immondices-photos-rl-15403880372.jpg.rf.7d0befa323c2f0ff9680687398f6f68

# Create a figure
fig, axes = plt.subplots(1, 5, figsize=(20, 5)) # 1 row, 5 columns

# List of images
images = [img1, img2, img3, img4, img5]
titles = ['Detection 1', 'Detection 2', 'Detection 3', 'Detection 4', 'Detection 5']

# Plot each image
for i in range(5):
    axes[i].imshow(images[i])
    axes[i].axis('off')
    axes[i].set_title(titles[i], fontsize=12)

plt.tight_layout()
plt.show()
```



▼ Update the ymal file

```
import os
dataset_path = '/kaggle/working/garbage-data'
```

```
with open(os.path.join(dataset_path, 'data.yaml'), 'r') as f:
    yaml_content = f.read()
yaml_content = yaml_content.replace('train: train', f'train: {dataset_path}/train')
yaml_content = yaml_content.replace('val: valid', f'val: {dataset_path}/valid')
yaml_content = yaml_content.replace('test: test', f'test: {dataset_path}/test')
with open(os.path.join(dataset_path, 'data.yaml'), 'w') as f:
    f.write(yaml_content)
```

▼ Train the model using the updated data.yaml

```
!yolo task=detect mode=train model=yolov8n.pt data=/kaggle/working/garbage-data/data.yaml epochs=50 imgsz=640 batch=16 name=yolov8n_garbage
```

→ Creating new Ultralytics Settings v0.0.6 file ✓
 View Ultralytics Settings with 'yolo settings' or at '/root/.config/Ultralytics/settings.json'
 Update Settings with 'yolo settings key=value', i.e. 'yolo settings runs_dir=path/to/dir'. For help see <https://docs.ultralytics.com/quickstart/#ultralytics-settings>.
 Downloading <https://github.com/ultralytics/assets/releases/download/v8.3.0/yolov8n.pt> to 'yolov8n.pt'...
 100%|██████████| 6.25M/6.25M [00:00<00:00, 107MB/s]
 Ultralytics 8.3.117 ↗ Python-3.11.11 torch-2.5.1+cu124 CUDA:0 (Tesla P100-PCIE-16GB, 16269MiB)
engine/trainer: task=detect, mode=train, model=yolov8n.pt, data=/kaggle/working/garbage-data/data.yaml, epochs=50, time=None, patience=100, batch=16, imgsz=640, save=True, s
 Downloading <https://ultralytics.com/assets/Arial.ttf> to '/root/.config/Ultralytics/Arial.ttf'...
 100%|██████████| 755k/755k [00:00<00:00, 24.5MB/s]
 Overriding model.yaml nc=80 with nc=1

	from	n	params	module	arguments
0	-1	1	464	ultralytics.nn.modules.conv.Conv	[3, 16, 3, 2]
1	-1	1	4672	ultralytics.nn.modules.conv.Conv	[16, 32, 3, 2]
2	-1	1	7360	ultralytics.nn.modules.block.C2f	[32, 32, 1, True]
3	-1	1	18560	ultralytics.nn.modules.conv.Conv	[32, 64, 3, 2]
4	-1	2	49664	ultralytics.nn.modules.block.C2f	[64, 64, 2, True]

```

5      -1 1    73984 ultralytics.nn.modules.conv.Conv      [64, 128, 3, 2]
6      -1 2    197632 ultralytics.nn.modules.block.C2f   [128, 128, 2, True]
7      -1 1    295424 ultralytics.nn.modules.conv.Conv   [128, 256, 3, 2]
8      -1 1    460288 ultralytics.nn.modules.block.C2f   [256, 256, 1, True]
9      -1 1    164608 ultralytics.nn.modules.block.SPPF  [256, 256, 5]
10     -1 1      0 torch.nn.modules.upsampling.Upsample [None, 2, 'nearest']
11     [-1, 6] 1      0 ultralytics.nn.modules.conv.Concat [1]
12     -1 1    148224 ultralytics.nn.modules.block.C2f   [384, 128, 1]
13     -1 1      0 torch.nn.modules.upsampling.Upsample [None, 2, 'nearest']
14     [-1, 4] 1      0 ultralytics.nn.modules.conv.Concat [1]
15     -1 1    37248 ultralytics.nn.modules.block.C2f   [192, 64, 1]
16     -1 1    36992 ultralytics.nn.modules.conv.Conv   [64, 64, 3, 2]
17     [-1, 12] 1      0 ultralytics.nn.modules.conv.Concat [1]
18     -1 1    123648 ultralytics.nn.modules.block.C2f   [192, 128, 1]
19     -1 1    147712 ultralytics.nn.modules.conv.Conv   [128, 128, 3, 2]
20     [-1, 9] 1      0 ultralytics.nn.modules.conv.Concat [1]
21     -1 1    493056 ultralytics.nn.modules.block.C2f   [384, 256, 1]
22     [15, 18, 21] 1    751507 ultralytics.nn.modules.head.Detect [1, [64, 128, 256]]
Model summary: 129 layers, 3,011,043 parameters, 3,011,027 gradients, 8.2 GFLOPs

```

Transferred 319/355 items from pretrained weights

Freezing layer 'model.22.dfl.conv.weight'

AMP: running Automatic Mixed Precision (AMP) checks...

Downloading <https://github.com/ultralytics/assets/releases/download/v8.3.0/yolo11n.pt> to 'yolo11n.pt'...

100% |██████████| 5.35M/5.35M [00:00<00:00, 117MB/s]

AMP: checks passed ✓

train: Fast image access ✓ (ping: 0.0±0.0 ms, read: 1707.4±449.5 MB/s, size: 91.4 KB)

train: Scanning /kaggle/working/garbage-data/train/labels... 3669 images, 210 ba

train: New cache created: /kaggle/working/garbage-data/train/labels.cache

WARNING ⚠ Box and segment counts should be equal, but got len(segments) = 6, len(boxes) = 3933. To resolve this only boxes will be used and all segments will be removed. To

albumentations: Blur(p=0.01, blur_limit=(3, 7)), MedianBlur(p=0.01, blur_limit=(3, 7)), ToGray(p=0.01, num_output_channels=3, method='weighted_average'), CLAHE(p=0.01, clip_val...

val: Fast image access ✓ (ping: 0.0±0.0 ms, read: 874.5±570.1 MB/s, size: 60.0 KB)

val: Scanning /kaggle/working/garbage-data/valid/labels... 260 images, 8 backgro

val: New cache created: /kaggle/working/garbage-data/valid/labels.cache

WARNING ⚠ Box and segment counts should be equal, but got len(segments) = 2, len(boxes) = 271. To resolve this only boxes will be used and all segments will be removed. To

Plotting labels to runs/detect/yolov8n_garbage/labels.jpg...

optimizer: 'optimizer=auto' found, ignoring 'lr0=0.01' and 'momentum=0.937' and determining best 'optimizer', 'lr0' and 'momentum' automatically...

optimizer: AdamW(lr=0.002, momentum=0.9) with parameter groups 57 weight(decay=0.0), 64 weight(decay=0.0005), 63 bias(decay=0.0)

Image sizes 640 train, 640 val

Using 4 dataloader workers

!ls //kaggle/working/runs/detect/yolov8n_garbage/weights

→ best.pt last.pt

!yolo task=detect mode=val model=/kaggle/working/runs/detect/yolov8n_garbage/weights/best.pt data=/kaggle/working/garbage-data/data.yaml

→ Ultralytics 8.3.117 🦄 Python-3.11.11 torch-2.5.1+cu124 CUDA:0 (Tesla P100-PCIE-16GB, 16269MiB)

Model summary (fused): 72 layers, 3,005,843 parameters, 0 gradients, 8.1 GFLOPs

val: Fast image access ✓ (ping: 0.0±0.0 ms, read: 1567.5±919.9 MB/s, size: 80.2 KB)

val: Scanning /kaggle/working/garbage-data/valid/labels.cache... 260 images, 8 b

WARNING ⚠ Box and segment counts should be equal, but got len(segments) = 2, len(boxes) = 271. To resolve this only boxes will be used and all segments will be removed. To av

Class	Images	Instances	Box(P	R	mAP50	m
all	260	271	0.894	0.842	0.862	0.567

/usr/local/lib/python3.11/dist-packages/matplotlib/colors.py:721: RuntimeWarning: invalid value encountered in less

xa[xa < 0] = -1

/usr/local/lib/python3.11/dist-packages/matplotlib/colors.py:721: RuntimeWarning: invalid value encountered in less

xa[xa < 0] = -1

Speed: 1.2ms preprocess, 3.0ms inference, 0.0ms loss, 1.8ms postprocess per image

Results saved to runs/detect/val
 Learn more at <https://docs.ultralytics.com/modes/val>

```
!yolo task=detect mode=predict model=/kaggle/working/runs/detect/yolov8n_garbage/weights/best.pt source=/kaggle/working/garbage-data/test/images conf=0.25 save=True
```

→ Ultralytics 8.3.117 🚀 Python-3.11.11 torch-2.5.1+cu124 CUDA:0 (Tesla P100-PCIE-16GB, 16269MiB)
 Model summary (fused): 72 layers, 3,005,843 parameters, 0 gradients, 8.1 GFLOPs

```
image 1/258 /kaggle/working/garbage-data/test/images/000144244W.jpg.rf.d1dc6442492ffbc7bcc5f0412bfea66.jpg: 640x640 3 garbages, 6.2ms
image 2/258 /kaggle/working/garbage-data/test/images/000839979_896x598_c.jpg.rf.04eba3aa5dbc551810d5b28ac9c38c53.jpg: 640x640 1 garbage, 6.0ms
image 3/258 /kaggle/working/garbage-data/test/images/02f330f2-9439-11e5-982d-40f2e96c8ad8.jpg.rf.af9f304e5a78fc6f84d268edef376fe3.jpg: 640x640 1 garbage, 5.4ms
image 4/258 /kaggle/working/garbage-data/test/images/07ff75e6-6799-11e5-8dd0-40f2e96c8ad8.jpg.rf.6e2fc221cd02c7e37fec6ca9af25d9f5.jpg: 640x640 1 garbage, 5.3ms
image 5/258 /kaggle/working/garbage-data/test/images/0_The-rubbish-left-behind-in-Sefton-ParkPic-Andrew-Teebay.jpg.rf.e04ea6eb8005b7c768ef3abda9389b8.jpg: 640x640 1 garbage
image 6/258 /kaggle/working/garbage-data/test/images/0ed6fe62-9439-11e5-982d-40f2e96c8ad8.jpg.rf.632814891b404ce212650bdc302e8852.jpg: 640x640 (no detections), 5.7ms
image 7/258 /kaggle/working/garbage-data/test/images/1076103483_015339992667_1920x0_80_0_f714d219f1a26124d2452fce46923cc2.jpg.rf.bf57dc5ef46e5c203dbad32368654149.jpg: 640x
image 8/258 /kaggle/working/garbage-data/test/images/1200x680_whatsapp_image_2021-09-26_at_10-12-35.jpg.rf.2ee83e27a1153d1e24d46b88f37923f8.jpg: 640x640 1 garbage, 5.6ms
image 9/258 /kaggle/working/garbage-data/test/images/12311590_679b-11e5-9696-40f2e96c8ad8.jpg.rf.555d843b794918eb6086292dd083538e.jpg: 640x640 1 garbage, 5.5ms
image 10/258 /kaggle/working/garbage-data/test/images/1280x680_075_acar-notitle200325_npwwq.jpg.rf.556cb5f262517b5b56e3242a5816651a.jpg: 640x640 (no detections), 5.4ms
image 11/258 /kaggle/working/garbage-data/test/images/162645714460f1c438246b2_1626457144_3x2_md.jpg.rf.018f84b9098c66ebb17dd57c32cbf76f.jpg: 640x640 1 garbage, 6.0ms
image 12/258 /kaggle/working/garbage-data/test/images/19aba664-679b-11e5-af8c-40f2e96c8ad8.jpg.rf.e849d6f7fb06fe9c940a99ce24305c46.jpg: 640x640 1 garbage, 5.9ms
image 13/258 /kaggle/working/garbage-data/test/images/1_jpg.rf.98fc8cb6cf5ee88791a083680b55d75.jpg: 640x640 2 garbages, 5.5ms
image 14/258 /kaggle/working/garbage-data/test/images/1b7ab13a-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.9618d46b947be76a0f9de5bf3e391621.jpg: 640x640 (no detections), 5.4ms
image 15/258 /kaggle/working/garbage-data/test/images/1c15c44a-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.9481aa1e29241cd649157ebcd339090d.jpg: 640x640 (no detections), 5.3ms
image 16/258 /kaggle/working/garbage-data/test/images/1c68ac06-6798-11e5-8c9e-40f2e96c8ad8.jpg.rf.ee444cd59bfe8d301877ff4e4c74ea7c.jpg: 640x640 1 garbage, 6.0ms
image 17/258 /kaggle/working/garbage-data/test/images/2-2-1024x576.jpg.rf.032040d8463e60c94e9e76efca766ee8.jpg: 640x640 (no detections), 5.4ms
image 18/258 /kaggle/working/garbage-data/test/images/20200924_161845-scaled.jpg.rf.46465de0a1977e4c6c6ca4fc9dffc29.jpg: 640x640 1 garbage, 5.4ms
image 19/258 /kaggle/working/garbage-data/test/images/21172353_hK9YZ.jpeg.rf.a67dfb37d620f6f00d94d69430d92d07.jpg: 640x640 1 garbage, 5.8ms
image 20/258 /kaggle/working/garbage-data/test/images/2251501384.jpg.rf.da6182839bfca14ca480c51f6b4fb1e.jpg: 640x640 1 garbage, 5.6ms
image 21/258 /kaggle/working/garbage-data/test/images/23-de-febrero-bolsa-basura.jpg.rf.6c444ea7a8e3cb96675093b0c1c3262e.jpg: 640x640 1 garbage, 5.5ms
image 22/258 /kaggle/working/garbage-data/test/images/2655dada-6799-11e5-8dd0-40f2e96c8ad8.jpg.rf.45733545ef906057526e10216135882a2.jpg: 640x640 1 garbage, 5.4ms
image 23/258 /kaggle/working/garbage-data/test/images/27de75ae-679b-11e5-af8c-40f2e96c8ad8.jpg.rf.9f61052dcf5f3291fa8b8326c813c9ed.jpg: 640x640 1 garbage, 5.8ms
image 24/258 /kaggle/working/garbage-data/test/images/297c00ea-6799-11e5-8dd0-40f2e96c8ad8.jpg.rf.92504429248f4fce756ffe761d057b4e.jpg: 640x640 1 garbage, 5.6ms
image 25/258 /kaggle/working/garbage-data/test/images/29dfdf7e-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.89c6a9279e3cbe9da34ba13b66876501.jpg: 640x640 (no detections), 5.5ms
image 26/258 /kaggle/working/garbage-data/test/images/2FKFBJ2.jpg.rf.1f10cd50aa393e384a9484f513fc6c0a.jpg: 640x640 1 garbage, 5.5ms
image 27/258 /kaggle/working/garbage-data/test/images/2b144732-6799-11e5-8dd0-40f2e96c8ad8.jpg.rf.573d8e2a2403f3b02e29ba4796e4c5a8.jpg: 640x640 1 garbage, 5.6ms
image 28/258 /kaggle/working/garbage-data/test/images/2b2d41a2-6798-11e5-8c9e-40f2e96c8ad8.jpg.rf.bbfd5a4efb4bfaf15c5f0e69d4e576ba.jpg: 640x640 1 garbage, 5.3ms
image 29/258 /kaggle/working/garbage-data/test/images/2bf41454-679c-11e5-a533-40f2e96c8ad8.jpg.rf.a38fe1a3c7fdd8fc827b7de80ffdaf14.jpg: 640x640 1 garbage, 5.4ms
image 30/258 /kaggle/working/garbage-data/test/images/2bf9a41c-6799-11e5-8dd0-40f2e96c8ad8.jpg.rf.c6728d16b6c5a88d62a60031a7ac7de.jpg: 640x640 1 garbage, 5.6ms
image 31/258 /kaggle/working/garbage-data/test/images/2fe04bda-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.88346b202644339c480aaef2ecfe553c8.jpg: 640x640 (no detections), 6.5ms
image 32/258 /kaggle/working/garbage-data/test/images/3-studio-de-design-donttakethisthewrongway.jpg.rf.b91c620c37eb5fb01f1292874614d16a.jpg: 640x640 2 garbages, 5.6ms
image 33/258 /kaggle/working/garbage-data/test/images/31450_medium.jpg.rf.e61e0d734a152d0098d185e428b0e1.jpg: 640x640 1 garbage, 5.5ms
image 34/258 /kaggle/working/garbage-data/test/images/37214739.jpg.rf.22eefc08c524f30bd4defb160b616f3b.jpg: 640x640 2 garbages, 5.4ms
image 35/258 /kaggle/working/garbage-data/test/images/3adb2c8-6799-11e5-8dc4-40f2e96c8ad8.jpg.rf.47a0ae8b1277b3f2424793ac07bf702b.jpg: 640x640 2 garbages, 6.0ms
image 36/258 /kaggle/working/garbage-data/test/images/3b500736-6798-11e5-8dd0-40f2e96c8ad8.jpg.rf.eda992661700d02d274f36285a41dcc2.jpg: 640x640 1 garbage, 5.9ms
image 37/258 /kaggle/working/garbage-data/test/images/41f3a844-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.16fc49a3e9d38c03163e7097d5233c9b.jpg: 640x640 (no detections), 5.5ms
image 38/258 /kaggle/working/garbage-data/test/images/441624891_orig.jpg.rf.32343939838fa8a82e9d1cbc969c8.jpg: 640x640 2 garbages, 5.6ms
image 39/258 /kaggle/working/garbage-data/test/images/44ab3b5e-679b-11e5-af8c-40f2e96c8ad8.jpg.rf.c4283862088d67891b5ab5f040f5bf1d.jpg: 640x640 1 garbage, 5.4ms
image 40/258 /kaggle/working/garbage-data/test/images/4fea3f94-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.d651fdc29e2899e10a5304204dd578a0.jpg: 640x640 (no detections), 5.4ms
image 41/258 /kaggle/working/garbage-data/test/images/5265963.jpg.rf.f16fd4d2f7ec54ad458cd5cd1825025.jpg: 640x640 1 garbage, 5.8ms
image 42/258 /kaggle/working/garbage-data/test/images/58c56134-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.96ea76168deb7619f6a454f05a1d340b.jpg: 640x640 (no detections), 5.8ms
image 43/258 /kaggle/working/garbage-data/test/images/6067a5de-9447-11e5-9ae8-40f2e96c8ad8.jpg.rf.2542c8183316a8c9f89eb0236844ed2.jpg: 640x640 (no detections), 5.6ms
image 44/258 /kaggle/working/garbage-data/test/images/61534d74-943b-11e5-9fb2-40f2e96c8ad8.jpg.rf.49d1def65e24945b5ff57d7b3da1d904.jpg: 640x640 (no detections), 5.5ms
image 45/258 /kaggle/working/garbage-data/test/images/641038096.jpeg.rf.850d5b046af4d027c761b95c1275cf4.jpg: 640x640 2 garbages, 5.4ms
image 46/258 /kaggle/working/garbage-data/test/images/65516d32-679c-11e5-aa4a-40f2e96c8ad8.jpg.rf.7eff047814b32ef51237983618c96dd1.jpg: 640x640 1 garbage, 6.5ms
image 47/258 /kaggle/working/garbage-data/test/images/67a30e40-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.63a6aebf1b7df995184856e1a1ce739d.jpg: 640x640 (no detections), 5.7ms
image 48/258 /kaggle/working/garbage-data/test/images/68baab0-9447-11e5-9ae8-40f2e96c8ad8.jpg.JPG.rf.49b397daea861793df078606a03faca09.jpg: 640x640 (no detections), 5.4ms
image 49/258 /kaggle/working/garbage-data/test/images/6b2a5106-9447-11e5-9ae8-40f2e96c8ad8.jpg.rf.188800ec704521e2eb7556aa4a500f9.jpg: 640x640 (no detections), 5.5ms
image 50/258 /kaggle/working/garbage-data/test/images/83067fa2-9447-11e5-9ae8-40f2e96c8ad8.jpg.rf.e04da2a56546857f88bd2d9b81d643d1.jpg: 640x640 (no detections), 5.8ms
image 51/258 /kaggle/working/garbage-data/test/images/85c7d32e-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.76c13ffa9876ccab9cfcab56e64fd5b.jpg: 640x640 (no detections), 5.3ms
```

```
image 52/258 /kaggle/working/garbage-data/test/images/86645da6-679a-11e5-9696-40f2e96c8ad8.jpg.rf.529af3fa0bbfe767121f3fb5641fbc7.jpg: 640x640 1 garbage, 5.5ms
image 53/258 /kaggle/working/garbage-data/test/images/870x489_20210105_113943.jpg.rf.fe3bc74efdc5516b46646e17722335af.jpg: 640x640 1 garbage, 5.7ms
image 54/258 /kaggle/working/garbage-data/test/images/870x489_ratecc_ina_rf_c79075eccccc70856rad699986a6hd25_ina_610v610_1_garbage_6_7mc
```

▼ Predicted Images

```
!ls /kaggle/working/runs/detect/predict
```

```
→ 000144244W.jpg.rf.d1dc6442492ffbc7bcc5f0412bfea66.jpg
000839979_896x598_c.jpg.rf.04eba3aa5dbc551810d5b28ac9c38c53.jpg
02f330f2-9439-11e5-982d-40f2e96c8ad8.jpg.rf.af9f304e5a78fc6f84d268edef376fe3.jpg
07ff75e6-6799-11e5-8dd0-40f2e96c8ad8.jpg.rf.6e2fc221cd02c7e37fec6ca9af25d9f5.jpg
0ed6fe62-9439-11e5-982d-40f2e96c8ad8.jpg.rf.632814891b404ce212650bdc302e8852.jpg
0_The-rubbish-left-behind-in-Sefton-ParkPic-Andrew-Teebay.jpg.rf.e04ea6eb8b005b7c768ef3abda9389b8.jpg
1076103483_015339992667_1920x0_80_0_0.f714d219f1a26124d2452fce46923cc2.jpg.rf.bf57dc5ef46e5c203dbad32368654149.jpg
_117865773_1a9c9b4a-d985-4f48-82be-0beeaaeaecec8.jpg.rf.5d2ac3aa62369648d48c1f07a83aedb7.jpg
1200x680_whatsapp_image_2021-09-26_at_10-12-35.jpg.rf.2ee83e27a1153d1e24d46b88f37923f8.jpg
12311590-679b-11e5-9696-40f2e96c8ad8.jpg.rf.555d843b794918eb6086292dd083538e.jpg
1280x680_075_acar-notitle200325_npwwq.jpg.rf.556cb5f262517b5b56e3242a5816651a.jpg
162645714460f1c438246b2_1626457144_3x2_md.jpg.rf.018f84b9098c66ebb17dd57c32cbf76f.jpg
19aba664-679b-11e5-af8c-40f2e96c8ad8.jpg.rf.e849df7fb06fe9c940a99ce24305c46.jpg
1b7ab13a-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.9618d46b947be76a0f9de5bffe391621.jpg
1c15c44a-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.9481aa1e29241cd640157ebed339090d.jpg
1c68ac06-6798-11e5-8c9e-40f2e96c8ad8.jpg.rf.ee444cd59bfe8d301877ff4e4c74ea7c.jpg
1.jpg.rf.98fc8ccb6cf5ee88791a083680b55d75.jpg
20200924_161845-scaled.jpg.rf.46465de0a1977e4c6c6ca4fc99dfffc29.jpg
21172353_hK9YZ.jpeg.rf.a67dfb37d620f6f00d94d69430d92d07.jpg
2-2-1024x576.jpg.rf.032040d8463e60c94e9e76efca766ee8.jpg
2251501384.jpg.rf.da6182839bfca14c480c51f6b4afb1e.jpg
23-de-febrero-bolsa-basura.jpg.rf.6c4a4ea7a8e3cb96675093b0c1c3262e.jpg
2655dada-6799-11e5-8dd0-40f2e96c8ad8.jpg.rf.4573354ef906057526e10216135882a2.jpg
27de75ae-679b-11e5-af8c-40f2e96c8ad8.jpg.rf.9f61052dcf5f3291fa8b8326c813c9ed.jpg
297c00ea-6799-11e5-8dd0-40f2e96c8ad8.jpg.rf.92504429248f4cfe756fe761d057b4e.jpg
29d6fd7e-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.89c6a9279e3cbe9da34ba13b66876501.jpg
2b144732-6799-11e5-8dd0-40f2e96c8ad8.jpg.rf.573de82a2403f3b02e29ba4796e4c5a8.jpg
2b2d41a2-6798-11e5-8c9e-40f2e96c8ad8.jpg.rf.bbffd54afeb4bff15c5f0e69d4e576ba.jpg
2bf4154-679c-11e5-a533-40f2e96c8ad8.jpg.rf.a38fe1a3c7fdd8fc827b7de80ffdaf14.jpg
2bf9a41c-6799-11e5-8dd0-40f2e96c8ad8.jpg.rf.c6728dd16b6c5a88d62a60031a7ac7de.jpg
2fe04bda-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.88346b202644339c480aab2ecfe553c8.jpg
2FKFBJ2.jpg.rf.1f10cd50aa393e384a9484f513fc6c0a.jpg
31450_medium.jpg.rf.e61e0d734a7152d0098d185e428b80e1.jpg
37214739.jpg.rf.22eefc08c524f30bd4defb160b616f3b.jpg
3adbac28-6799-11e5-8dc4-40f2e96c8ad8.jpg.rf.47a0ae81277b3f2424793ac07bf702b.jpg
3b500736-6798-11e5-8dd0-40f2e96c8ad8.jpg.rf.eda992661700d02d274f36285a41dcc2.jpg
3-studio-de-design-donttakehisthewrongway.jpg.rf.b91c620c37eb5fb01f1292874614d16a.jpg
41f3a844-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.16fc49a3e9d38c03163e7097d5233c9b.jpg
441624891_orig.jpg.rf.32343939838fa8a82e9d1cbc96c8.jpg
44ab3b5e-679b-11e5-af8c-40f2e96c8ad8.jpg.rf.c4283862088d6f891b5ab5f040f5bf1d.jpg
4fea3f94-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.d651fdc29e2899e10a5304204dd578a0.jpg
572659653.jpg.rf.f16fdd4d2f7ec54ad458cd5cd1825025.jpg
58c56134-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.96ea76168deb7619f6a454f05a1d340b.jpg
6067a5de-9447-11e5-9ae8-40f2e96c8ad8.jpg.rf.2542c8183316a8c9f89eb0c236844ed2.jpg
61534d74-943b-11e5-9fb2-40f2e96c8ad8.jpg.rf.49d1def65e24945b5ff57d7b3da1d904.jpg
641038096.jpeg.rf.850d5b046afdf40d27c761b95c1275cf4.jpg
65516d32-679c-11e5-aa4a-40f2e96c8ad8.jpg.rf.7eff047814b32ef51237983618c96dd1.jpg
67a30e40-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.63a6aebf1b7df995184856e1a1ce739d.jpg
68bab0a-9447-11e5-9ae8-40f2e96c8ad8.JPG.rf.49b397daa681793df078606a03fac09.jpg
```

```
6b2a5106-9447-11e5-9ae8-40f2e96c8ad8.jpg.rf.188800ec704521e2eab7556aa4a500f9.jpg
83067fa2-9447-11e5-9ae8-40f2e96c8ad8.jpg.rf.e04da2a56546857f88bd2d9b81d643d1.jpg
85c7d32e-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.76c13ffa9876ccab9cfcab56e64fd5b.jpg
86645da6-679a-11e5-9696-40f2e96c8ad8.jpg.rf.529af3fa0bbfe767121f3fb5641fbc7.jpg
870x489_20210105_113943.jpg.rf.fe3bc74efdc5516b46646e17722335af.jpg
870x489_ratss.jpg.rf.c799275eccccc70856c9d699986a6bda5.jpg
8b245de2-679e-11e5-b0d3-40f2e96c8ad8.jpg.rf.67050293f3f671f87ba7c9d9fcfd9d24a.jpg
8bce1fca-679a-11e5-9696-40f2e96c8ad8.jpg.rf.bad573755a958dafb4d3f498bd28b268.jpg
```

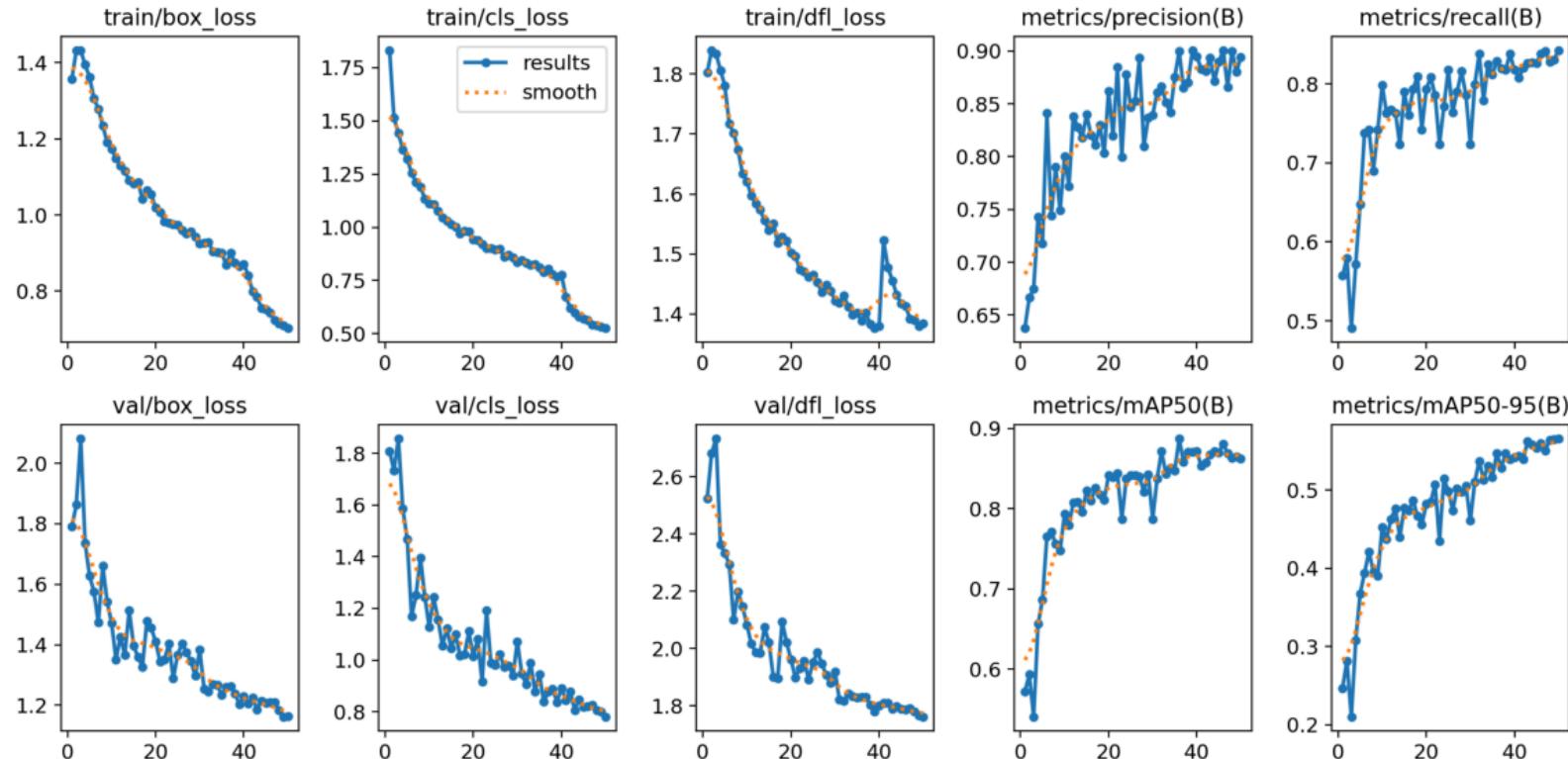
```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg

# Load your image
img = mpimg.imread('/kaggle/working/runs/detect/yolov8n_garbage/results.png')

# Plot the image
plt.figure(figsize=(15, 8)) # you can change the size if you want
plt.imshow(img)
plt.axis('off') # Hide axis
plt.title('Sample Garbage Detection Output', fontsize=16)
plt.show()
```



Sample Garbage Detection Output



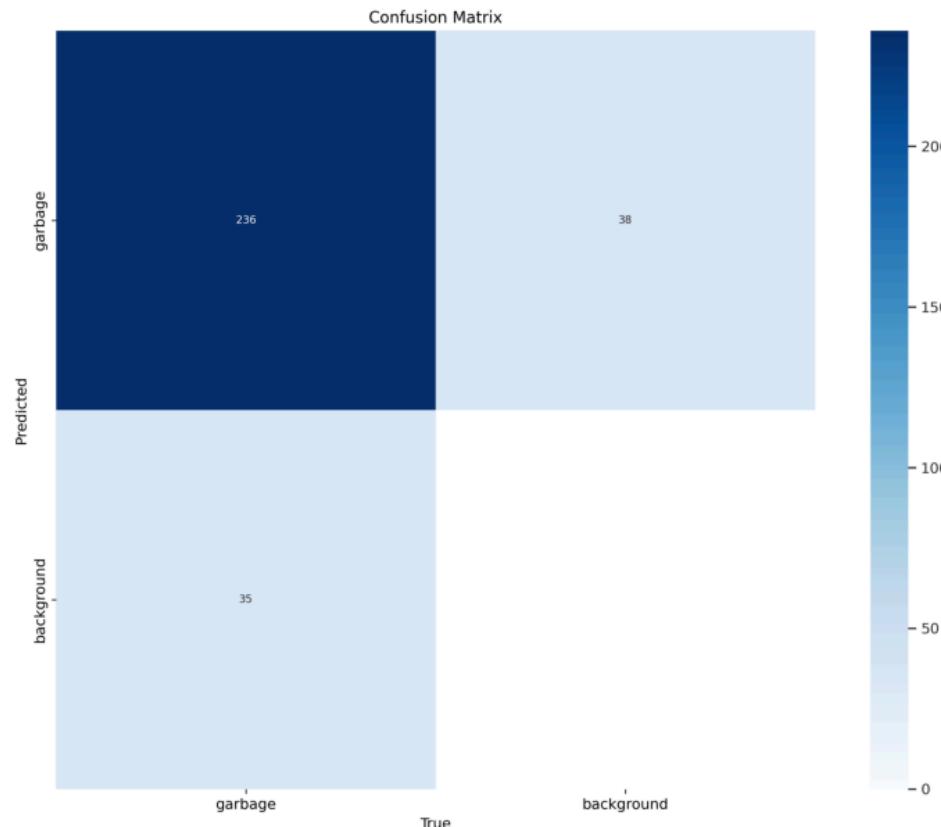
```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg

# Load your image
img = mpimg.imread('/kaggle/working/runs/detect/yolov8n_garbage/confusion_matrix.png')

# Plot the image
plt.figure(figsize=(15, 8)) # you can change the size if you want
plt.imshow(img)
plt.axis('off') # Hide axis
plt.title('Sample Garbage Detection Output', fontsize=16)
plt.show()
```



Sample Garbage Detection Output



```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg

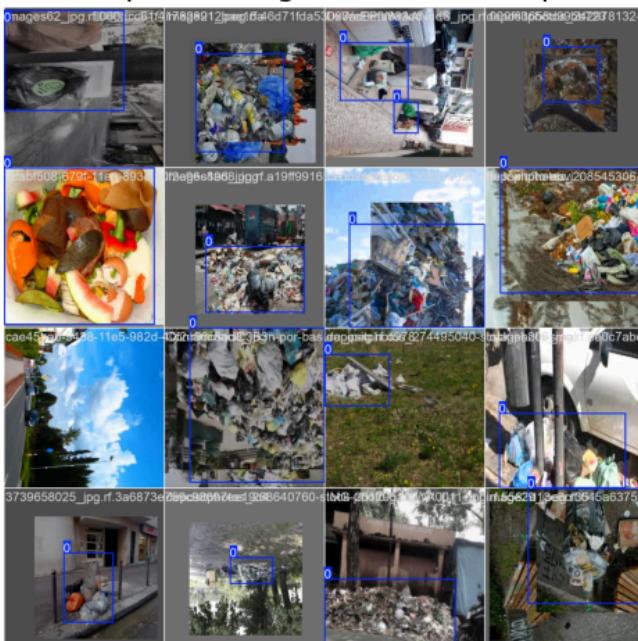
# Load your image
```

```
img = mpimg.imread('/kaggle/working/runs/detect/yolov8n_garbage/train_batch9202.jpg')
```

```
# Plot the image
plt.figure(figsize=(8, 6)) # you can change the size if you want
plt.imshow(img)
plt.axis('off') # Hide axis
plt.title('Sample Garbage Detection Output', fontsize=16)
plt.show()
```



Sample Garbage Detection Output



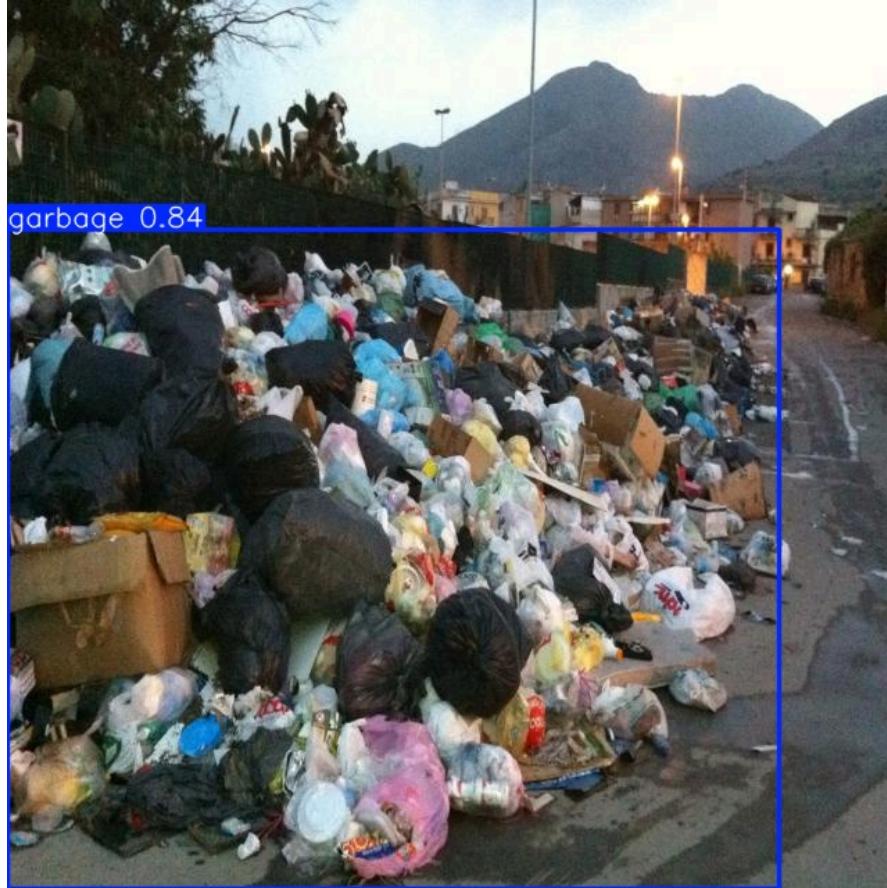
```
import os
from IPython.display import Image, display
```

```
# Get list of predicted images
pred_dir = r'/kaggle/working/runs/detect/predict'
image_files = [f for f in os.listdir(pred_dir) if f.endswith('.jpg')]

# Display up to 5 images (or adjust as needed)
for img_file in image_files[:30]:
    image_path = os.path.join(pred_dir, img_file)
    print(f"Displaying: {img_file}")
    display(Image(filename=image_path))
```



Displaying: segnalazione_jpg.rf.cbf5bcf1285ecff1f98fc1a96c2f39d5.jpg



Displaying: fad5823e-6798-11e5-8dd0-40f2e96c8ad8.jpg.rf.e14c9112533b52c2d6e2e08d427c38dd.jpg





Displaying: 9aba0088-9438-11e5-982d-40f2e96c8ad8.jpg_rf.ed7a070c8d591582b99fa10bb1675874.jpg



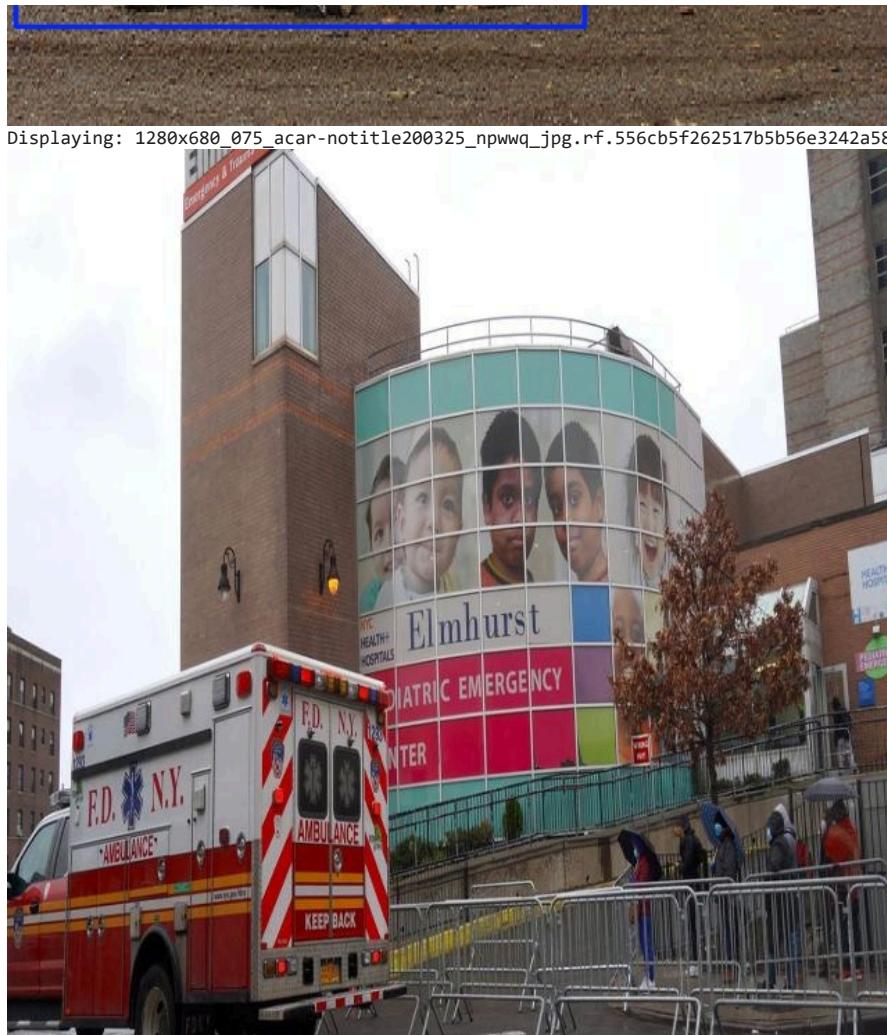
Displaying: images149.jpg_rf.52b868e9d27af4f1a67e060140482ca6.jpg





Displaying: c97d0eb4-6798-11e5-8dd0-40f2e96c8ad8.jpg.rf.f96f9b64fdcb5909d0f8ab0ff6914884.jpg







Displaying: 000144244W.jpg.rf.d1dc6442492ffbc7bcce5f0412bfea66.jpg



Displaying: Que-es-la-contaminacion-de-basura-en-las-calles.jpg.rf.4a76d1f8f82c03be66813741d3c0d379.jpg

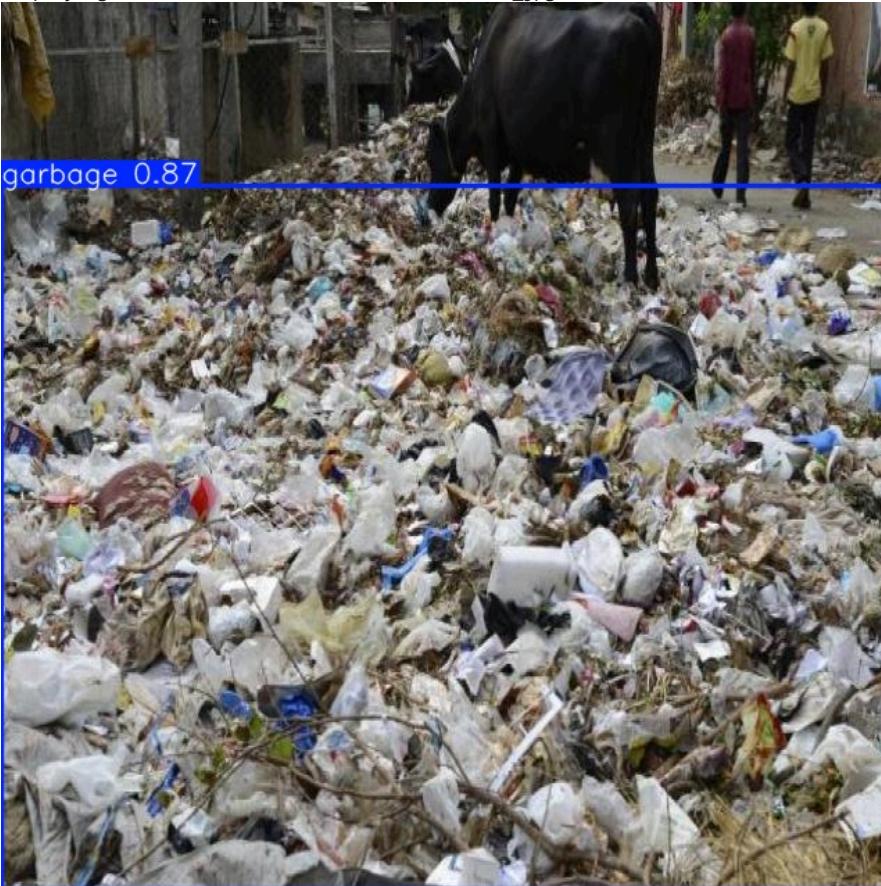


Displaying: image2.jpeg.rf.6cf31ea3d1d68049268f4a4ea8c2d229.jpg





Displaying: cbd479be-6797-11e5-8c9e-40f2e96c8ad8.jpg.rf.958bd8f6afeb72fa673d0d44c80f5ae3.jpg



Displaying: cb7f2eb8-6798-11e5-8dd0-40f2e96c8ad8.jpg.rf.99f009b478a751c7d83892e88932e415.jpg





Displaying: images74.jpg.rf.452fb1255d62fd113d9c4d023a5f590d.jpg



Displaying: garbage-in-trash-with-black-bag-at-park-picture-id1172153727.jpg.rf.c640fc219db02a2340905662b314965f.jpg



Displaying: mull-1_1558549054.jpg.rf.89fa1c5baa5f9df881a2ce3408a7dc88.jpg





Displaying: cb079cc2-6798-11e5-8dd0-40f2e96c8ad8.jpg.rf.2b0e37fe6c0a13b09cb5d09d2a23a6f0.jpg



Displaying: images-2.jpg.rf.a8f53a6e1d55567c2678fc10b387fbff9.jpg





Displaying: images251.jpg.rf.3ae8ba087dd9735400d28f2d445a61b0.jpg





Displaying: depositphotos_290716408-stock-photo-big-and-many-plastic-garbage3.jpg.rf.7fa5fe17819d550786a863e75ba97e07.jpg



garbage 0.40

garbage 0.56



Displaying: A-Rome-les-rues-jonchees-d-ordures-font-craindre-une-crise-sanitaire.jpg.rf.95198ff46488bf93cb9655306c1416e8.jpg



garbage 0.87



Displaying: ba076ca4-6798-11e5-8dd0-40f2e96c8ad8.jpg.rf.6dcd14d309f93cdd3c844056c8c5fe52.jpg

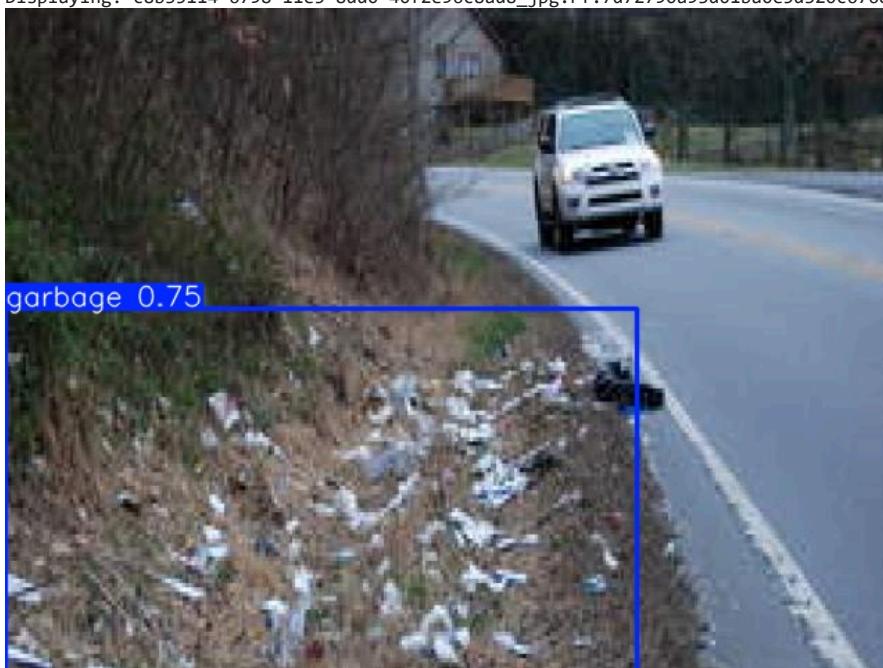


Displaying: images45.jpg.rf.9e0112dd87d2048983af07b2420c64bc.jpg





Displaying: c8b35114-6798-11e5-8dd0-40f2e96c8ad8.jpg.rf.7d72790a93d01ba0e5d320c6768e6ac3.jpg





Displaying: 44ab3b5e-679b-11e5-af8c-40f2e96c8ad8.jpg.rf.c4283862088d6f891b5ab5f040f5bf1d.jpg
garbage 0.86



Displaying: 95406aa8-679b-11e5-a533-40f2e96c8ad8.jpg.rf.86e25aaba765712e71567bd12c1bd08e.jpg



garbage 0.87



Displaying: collecte-dechets.jpg.rf.36986b4e692704604641c568aa116332.jpg



4/27/25, 5:12 AM

momentum-waste-detector (4).ipynb - Colab

Displaying: 11x0-da-pol15A/L3A30-tlorestai-deixado-na-tloresta-perto-estrada-1/6148111.jpg.rf.8c5b532/ad8a12895ff11924263589a3.jpg



Displaying: 98d30a06-943a-11e5-ba6d-40f2e96c8ad8.jpg.rf.68d52f06061175ef793ebb1b34c9b00a.jpg



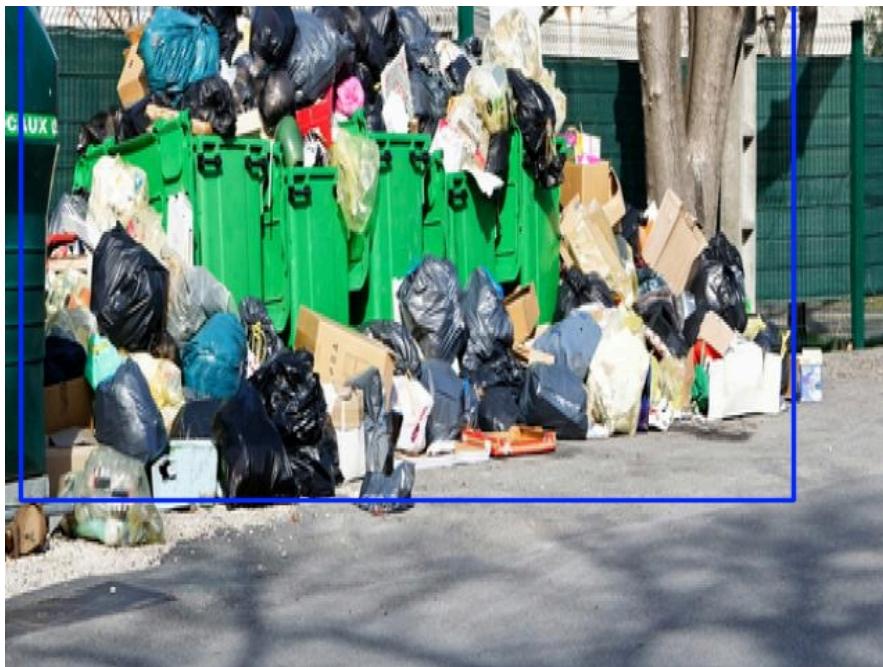


Displaying: depositphotos_227673224-stock-photo-junkyard-nature-blue-sky-clouds3.jpg.rf.f76fd7dc4b8d575503b840f7628c6c2c.jpg



Displaying: 31450_medium.jpg.rf.e61e0d734a7152d0098d185e428b80e1.jpg





Start coding or [generate](#) with AI.

▼ Egpt Waste

```
!yolo task=detect mode=predict model=/kaggle/working/runs/detect/yolov8n_garbage/weights/best.pt source=/kaggle/input/local-waste conf=0.25 save=True
```

```
→ Ultralytics 8.3.117 ✨ Python-3.11.11 torch-2.5.1+cu124 CUDA:0 (Tesla P100-PCIE-16GB, 16269MiB)
Model summary (fused): 72 layers, 3,005,843 parameters, 0 gradients, 8.1 GFLOPs
```

```
image 1/4 /kaggle/input/local-waste/IMG-20250426-WA0007.jpg: 640x384 2 garbages, 44.6ms
image 2/4 /kaggle/input/local-waste/IMG-20250426-WA0008.jpg: 640x384 1 garbage, 6.4ms
image 3/4 /kaggle/input/local-waste/IMG-20250426-WA0009.jpg: 640x640 4 garbages, 6.1ms
image 4/4 /kaggle/input/local-waste/IMG-20250426-WA0010.jpg: 640x480 2 garbages, 32.4ms
Speed: 3.0ms preprocess, 22.4ms inference, 46.7ms postprocess per image at shape (1, 3, 640, 480)
Results saved to runs/detect/predict2
💡 Learn more at https://docs.ultralytics.com/modes/predict
```

```
!ls /kaggle/working/runs/detect/predict2
```

```
→ IMG-20250426-WA0007.jpg  IMG-20250426-WA0009.jpg
IMG-20250426-WA0008.jpg  IMG-20250426-WA0010.jpg
```

```
import os
from IPython.display import Image, display

# Directory containing the predicted images
pred_dir = '/kaggle/working/runs/detect/predict2'
image_files = [f for f in os.listdir(pred_dir) if f.endswith('.jpg')]

# Display all images
for img_file in image_files:
    image_path = os.path.join(pred_dir, img_file)
    print(f"Displaying: {img_file}")
    display(Image(filename=image_path))
```

Displaying: IMG-20250426-WA0008.jpg





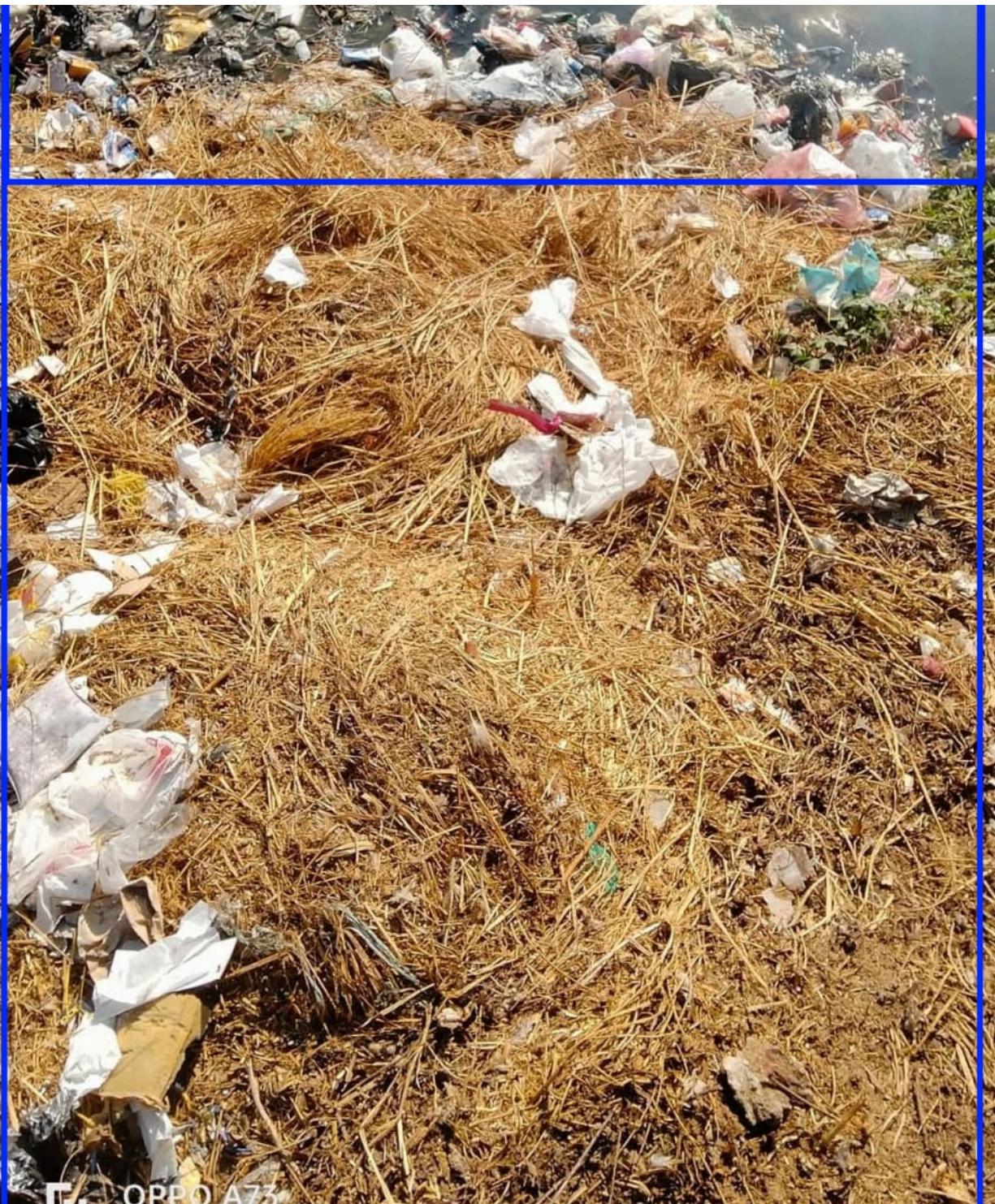
Displaying: IMG-20250426-WA0009.jpg





Displaying: IMG-20250426-WA0007.jpg









▼ Download Tuned Model

```
# Compress the runs directory
!zip -r /kaggle/working/runs.zip /kaggle/working/runs

→ adding: kaggle/working/runs/ (stored 0%)
adding: kaggle/working/runs/detect/ (stored 0%)
adding: kaggle/working/runs/detect/yolov8n_garbage/ (stored 0%)
adding: kaggle/working/runs/detect/yolov8n_garbage/train_batch1.jpg (deflated 1%)
adding: kaggle/working/runs/detect/yolov8n_garbage/train_batch2.jpg (deflated 1%)
adding: kaggle/working/runs/detect/yolov8n_garbage/PR_curve.png (deflated 22%)
adding: kaggle/working/runs/detect/yolov8n_garbage/val_batch1_labels.jpg (deflated 4%)
adding: kaggle/working/runs/detect/yolov8n_garbage/P_curve.png (deflated 15%)
adding: kaggle/working/runs/detect/yolov8n_garbage/R_curve.png (deflated 16%)
adding: kaggle/working/runs/detect/yolov8n_garbage/val_batch0_labels.jpg (deflated 5%)
adding: kaggle/working/runs/detect/yolov8n_garbage/val_batch1_pred.jpg (deflated 4%)
adding: kaggle/working/runs/detect/yolov8n_garbage/labels_correlogram.jpg (deflated 35%)
adding: kaggle/working/runs/detect/yolov8n_garbage/train_batch0.jpg (deflated 1%)
adding: kaggle/working/runs/detect/yolov8n_garbage/results.png (deflated 7%)
adding: kaggle/working/runs/detect/yolov8n_garbage/val_batch2_labels.jpg (deflated 5%)
adding: kaggle/working/runs/detect/yolov8n_garbage/train_batch9201.jpg (deflated 5%)
adding: kaggle/working/runs/detect/yolov8n_garbage/val_batch2_pred.jpg (deflated 4%)
adding: kaggle/working/runs/detect/yolov8n_garbage/weights/ (stored 0%)
adding: kaggle/working/runs/detect/yolov8n_garbage/weights/last.pt (deflated 9%)
adding: kaggle/working/runs/detect/yolov8n_garbage/weights/best.pt (deflated 9%)
```