YOLOv8 by Steven Indriano


```
!pip install ultralytics

Collecting ultralytics
  Downloading ultralytics-8.2.2-py3-none-any.whl (750 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 750.8/750.8 kB 4.6 MB/s eta
0:00:00
ent already satisfied: matplotlib>=3.3.0 in
/usr/local/lib/python3.10/dist-packages (from ultralytics) (3.7.1)
Requirement already satisfied: opencv-python>=4.6.0 in
/usr/local/lib/python3.10/dist-packages (from ultralytics) (4.8.0.76)
Requirement already satisfied: pillow>=7.1.2 in
/usr/local/lib/python3.10/dist-packages (from ultralytics) (9.4.0)
Requirement already satisfied: pyyaml>=5.3.1 in
/usr/local/lib/python3.10/dist-packages (from ultralytics) (6.0.1)
Requirement already satisfied: requests>=2.23.0 in
/usr/local/lib/python3.10/dist-packages (from ultralytics) (2.31.0)
Requirement already satisfied: scipy>=1.4.1 in
/usr/local/lib/python3.10/dist-packages (from ultralytics) (1.11.4)
Requirement already satisfied: torch>=1.8.0 in
/usr/local/lib/python3.10/dist-packages (from ultralytics)
(2.2.1+cu121)
Requirement already satisfied: torchvision>=0.9.0 in
/usr/local/lib/python3.10/dist-packages (from ultralytics)
(0.17.1+cu121)
Requirement already satisfied: tqdm>=4.64.0 in
/usr/local/lib/python3.10/dist-packages (from ultralytics) (4.66.2)
Requirement already satisfied: psutil in
/usr/local/lib/python3.10/dist-packages (from ultralytics) (5.9.5)
Requirement already satisfied: py-cpuinfo in
/usr/local/lib/python3.10/dist-packages (from ultralytics) (9.0.0)
Collecting thop>=0.1.1 (from ultralytics)
  Downloading thop-0.1.1.post2209072238-py3-none-any.whl (15 kB)
Requirement already satisfied: pandas>=1.1.4 in
/usr/local/lib/python3.10/dist-packages (from ultralytics) (2.0.3)
```

```
Requirement already satisfied: seaborn>=0.11.0 in
/usr/local/lib/python3.10/dist-packages (from ultralytics) (0.13.1)
Requirement already satisfied: contourpy>=1.0.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=3.3.0-
>ultralytics) (1.2.1)
Requirement already satisfied: cycler>=0.10 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=3.3.0-
>ultralytics) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=3.3.0-
>ultralytics) (4.51.0)
Requirement already satisfied: kiwisolver>=1.0.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=3.3.0-
>ultralytics) (1.4.5)
Requirement already satisfied: numpy>=1.20 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=3.3.0-
>ultralytics) (1.25.2)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=3.3.0-
>ultralytics) (24.0)
Requirement already satisfied: pyparsing>=2.3.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=3.3.0-
>ultralytics) (3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=3.3.0-
>ultralytics) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in
/usr/local/lib/python3.10/dist-packages (from pandas>=1.1.4-
>ultralytics) (2023.4)
Requirement already satisfied: tzdata>=2022.1 in
/usr/local/lib/python3.10/dist-packages (from pandas>=1.1.4-
>ultralytics) (2024.1)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from requests>=2.23.0-
>ultralytics) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in
/usr/local/lib/python3.10/dist-packages (from requests>=2.23.0-
>ultralytics) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from requests>=2.23.0-
>ultralytics) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from requests>=2.23.0-
>ultralytics) (2024.2.2)
Requirement already satisfied: filelock in
/usr/local/lib/python3.10/dist-packages (from torch>=1.8.0-
>ultralytics) (3.13.4)
Requirement already satisfied: typing-extensions>=4.8.0 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.8.0-
```

```
>ultralytics) (4.11.0)
Requirement already satisfied: sympy in
/usr/local/lib/python3.10/dist-packages (from torch>=1.8.0-
>ultralytics) (1.12)
Requirement already satisfied: networkx in
/usr/local/lib/python3.10/dist-packages (from torch>=1.8.0-
>ultralytics) (3.3)
Requirement already satisfied: jinja2 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.8.0-
>ultralytics) (3.1.3)
Requirement already satisfied: fsspec in
/usr/local/lib/python3.10/dist-packages (from torch>=1.8.0-
>ultralytics) (2023.6.0)
Collecting nvidia-cuda-nvrtc-cu12==12.1.105 (from torch>=1.8.0-
>ultralytics)
  Using cached nvidia_cuda_nvrtc_cu12-12.1.105-py3-none-
manylinux1_x86_64.whl (23.7 MB)
Collecting nvidia-cuda-runtime-cu12==12.1.105 (from torch>=1.8.0-
>ultralytics)
  Using cached nvidia_cuda_runtime_cu12-12.1.105-py3-none-
manylinux1_x86_64.whl (823 kB)
Collecting nvidia-cuda-cupti-cu12==12.1.105 (from torch>=1.8.0-
>ultralytics)
  Using cached nvidia_cuda_cupti_cu12-12.1.105-py3-none-
manylinux1_x86_64.whl (14.1 MB)
Collecting nvidia-cudnn-cu12==8.9.2.26 (from torch>=1.8.0-
>ultralytics)
  Using cached nvidia_cudnn_cu12-8.9.2.26-py3-none-
manylinux1_x86_64.whl (731.7 MB)
Collecting nvidia-cublas-cu12==12.1.3.1 (from torch>=1.8.0-
>ultralytics)
  Using cached nvidia_cublas_cu12-12.1.3.1-py3-none-
manylinux1_x86_64.whl (410.6 MB)
Collecting nvidia-cufft-cu12==11.0.2.54 (from torch>=1.8.0-
>ultralytics)
  Using cached nvidia_cufft_cu12-11.0.2.54-py3-none-
manylinux1_x86_64.whl (121.6 MB)
Collecting nvidia-curand-cu12==10.3.2.106 (from torch>=1.8.0-
>ultralytics)
  Using cached nvidia_curand_cu12-10.3.2.106-py3-none-
manylinux1_x86_64.whl (56.5 MB)
Collecting nvidia-cusolver-cu12==11.4.5.107 (from torch>=1.8.0-
>ultralytics)
  Using cached nvidia_cusolver_cu12-11.4.5.107-py3-none-
manylinux1_x86_64.whl (124.2 MB)
Collecting nvidia-cusparse-cu12==12.1.0.106 (from torch>=1.8.0-
>ultralytics)
  Using cached nvidia_cusparse_cu12-12.1.0.106-py3-none-
manylinux1_x86_64.whl (196.0 MB)
```

```
Collecting nvidia-nccl-cu12==2.19.3 (from torch>=1.8.0->ultralytics)
  Using cached nvidia_nccl_cu12-2.19.3-py3-none-manylinux1_x86_64.whl
(166.0 MB)
Collecting nvidia-nvtx-cu12==12.1.105 (from torch>=1.8.0->ultralytics)
  Using cached nvidia_nvtx_cu12-12.1.105-py3-none-
manylinux1_x86_64.whl (99 kB)
Requirement already satisfied: triton==2.2.0 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.8.0-
>ultralytics) (2.2.0)
Collecting nvidia-nvjitlink-cu12 (from nvidia-cusolver-
cu12==11.4.5.107->torch>=1.8.0->ultralytics)
  Using cached nvidia_nvjitlink_cu12-12.4.127-py3-none-
manylinux2014_x86_64.whl (21.1 MB)
Requirement already satisfied: six>=1.5 in
/usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7-
>matplotlib>=3.3.0->ultralytics) (1.16.0)
Requirement already satisfied: MarkupSafe>=2.0 in
/usr/local/lib/python3.10/dist-packages (from jinja2->torch>=1.8.0-
>ultralytics) (2.1.5)
Requirement already satisfied: mpmath>=0.19 in
/usr/local/lib/python3.10/dist-packages (from sympy->torch>=1.8.0-
>ultralytics) (1.3.0)
Installing collected packages: nvidia-nvtx-cu12, nvidia-nvjitlink-
cu12, nvidia-nccl-cu12, nvidia-curand-cu12, nvidia-cufft-cu12, nvidia-
cuda-runtime-cu12, nvidia-cuda-nvrtc-cu12, nvidia-cuda-cupti-cu12,
nvidia-cublas-cu12, nvidia-cusparse-cu12, nvidia-cudnn-cu12, nvidia-
cusolver-cu12, thop, ultralytics
Successfully installed nvidia-cublas-cu12-12.1.3.1 nvidia-cuda-cupti-
cu12-12.1.105 nvidia-cuda-nvrtc-cu12-12.1.105 nvidia-cuda-runtime-
cu12-12.1.105 nvidia-cudnn-cu12-8.9.2.26 nvidia-cufft-cu12-11.0.2.54
nvidia-curand-cu12-10.3.2.106 nvidia-cusolver-cu12-11.4.5.107 nvidia-
cusparse-cu12-12.1.0.106 nvidia-nccl-cu12-2.19.3 nvidia-nvjitlink-
cu12-12.4.127 nvidia-nvtx-cu12-12.1.105 thop-0.1.1.post2209072238
ultralytics-8.2.2
```

```python
import ultralytics
from ultralytics import YOLO
import cv2
from matplotlib import pyplot as plt
import urllib.request
import numpy as np

def load_model(model_name='yolov8n'):
    """Load the specified YOLOv8 model."""
    model = YOLO(model_name)
    return model

def download_image(image_url, image_path):
    """Download an image from the specified URL and save it to the
given path."""
```

```
    urllib.request.urlretrieve(image_url, image_path)
    return image_path

def load_image(image_path):
    """Load an image from the specified path."""
    image = cv2.imread(image_path)
    image_rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
    return image_rgb

def perform_inference(model, image_rgb):
    """Perform inference using the YOLOv8 model on the provided RGB
image."""
    results = model(image_rgb)
    return results

def visualize_results(results):
    """Visualize the results and count the number of persons
detected."""
    person_count = 0
    plot_image = None
    for result in results:
        for det in result.boxes:
            if det.cls == 0:
                person_count += 1
        plot_image = result.plot()

    plot_rgb = cv2.cvtColor(plot_image, cv2.COLOR_BGR2RGB)

    plt.figure(figsize=(10, 10))
    plt.imshow(plot_rgb)
    plt.title(f'Detected Persons: {person_count}')
    plt.axis('off')
    plt.show()

    return person_count

def main():
    model = load_model('yolov8n')

    image_url =
'https://www.ocf.berkeley.edu/~bluegold/static/img/portraits_icons/
portraits.png'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    results = perform_inference(model, image_rgb)

    person_count = visualize_results(results)
```

```
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
Downloading
https://github.com/ultralytics/assets/releases/download/v8.2.0/yolov8n
.pt to 'yolov8n.pt'...

100%|██████████| 6.23M/6.23M [00:00<00:00, 81.0MB/s]


0: 480x640 28 persons, 6 ties, 386.8ms
Speed: 18.0ms preprocess, 386.8ms inference, 3285.0ms postprocess per
image at shape (1, 3, 480, 640)
```

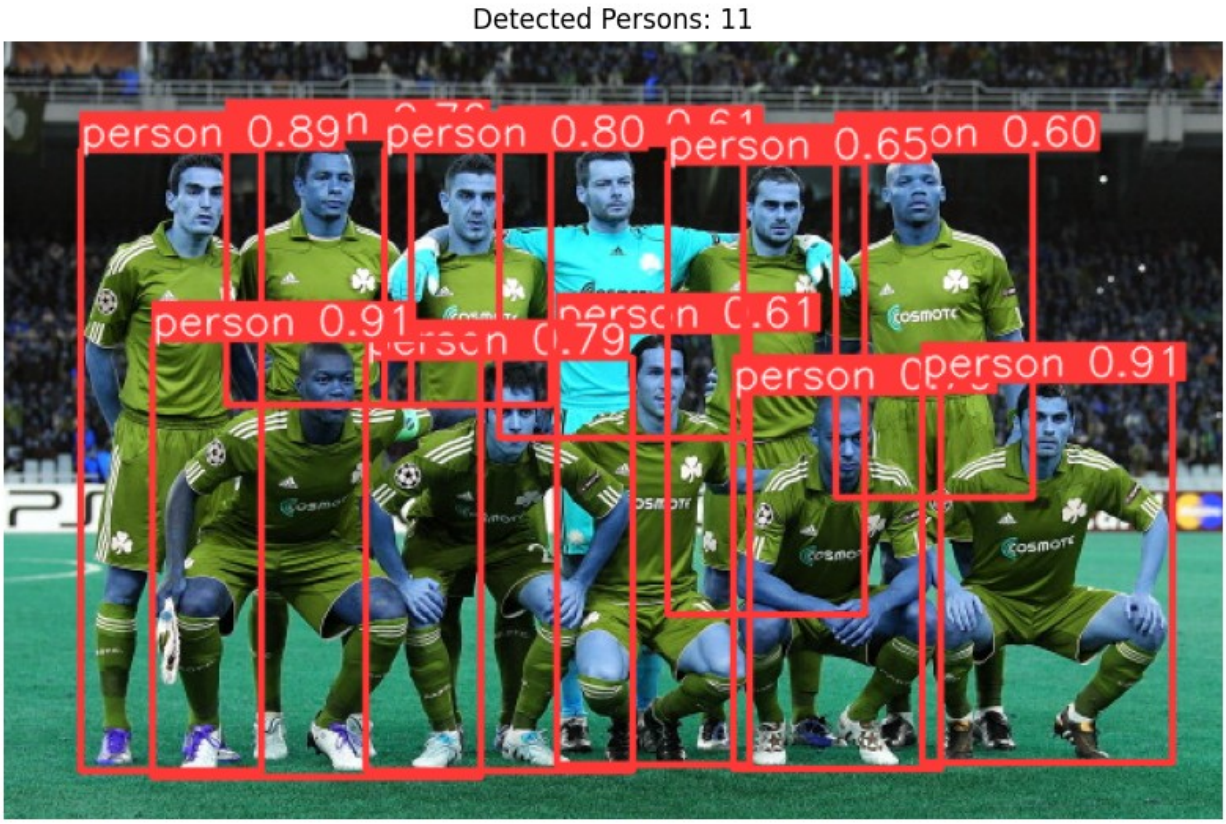

```
Number of people detected in the image: 28
```

Using YOLOv8m

```
def perform_inference(model, image_rgb, conf=0.5, iou=0.5):
    """Perform inference using the YOLOv8 model on the provided RGB
image.

    Args:
        model: The YOLOv8 model.
        image_rgb: The RGB image.
        conf: Confidence threshold for detections.
        iou: Intersection over Union threshold for non-maximum
suppression.

    Returns:
        Results from the model inference.
    """

    results = model(image_rgb, conf=conf, iou=iou)
    return results

def main():

    model = load_model('yolov8m')

    image_url =
'https://cdn.bleacherreport.net/images_root/slides/photos/000/852/862/
107111019_original.jpg?1302567411'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)


    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
Downloading
https://github.com/ultralytics/assets/releases/download/v8.2.0/yolov8m
.pt to 'yolov8m.pt'...

100%|██████████| 49.7M/49.7M [00:00<00:00, 192MB/s]
```

```
0: 416x640 11 persons, 1714.7ms
Speed: 3.6ms preprocess, 1714.7ms inference, 2.1ms postprocess per
image at shape (1, 3, 416, 640)
```

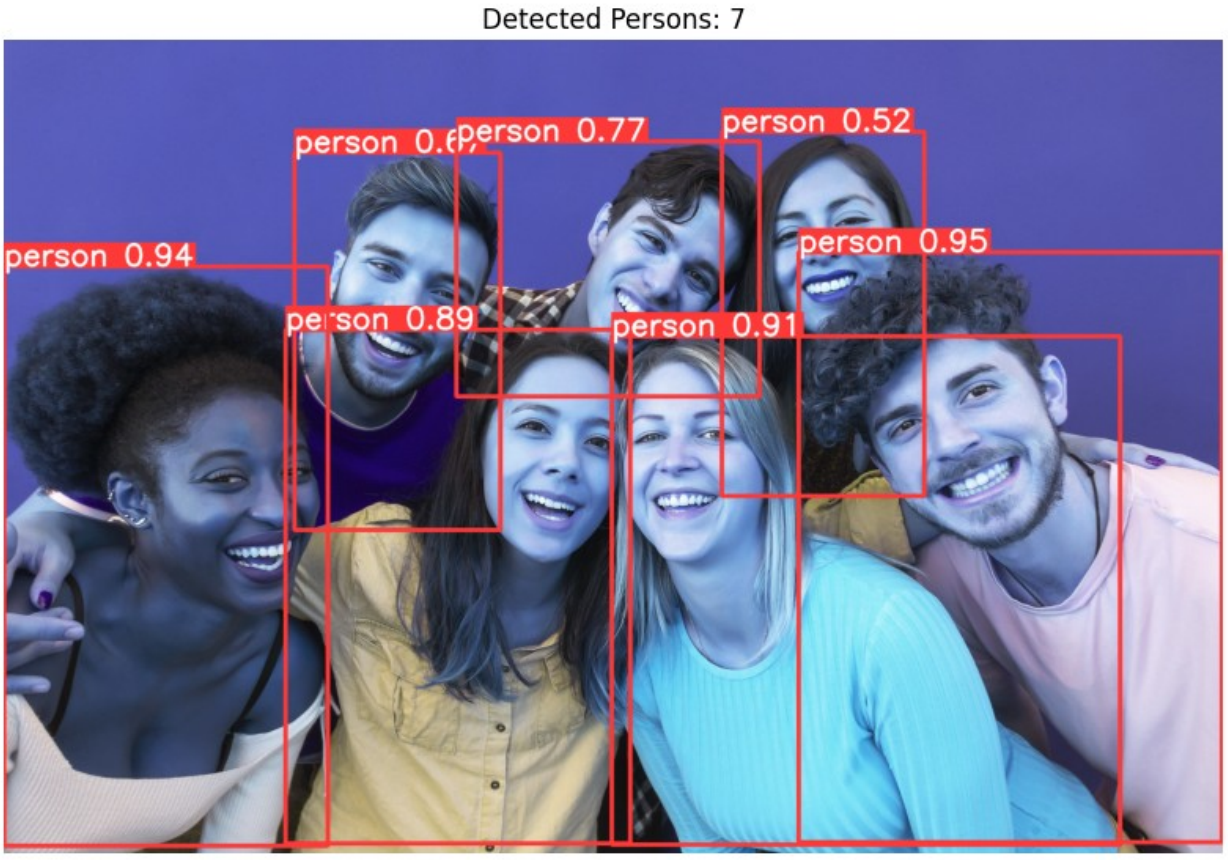



```
Number of people detected in the image: 11
```

```
def main():

    model = load_model('yolov8m')

    image_url =
'https://www.lukkap.com/wp-content/uploads/2023/08/sistemas-de-
voz.jpeg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)


    conf_threshold = 0.5
    iou_threshold = 0.5
```

```
    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

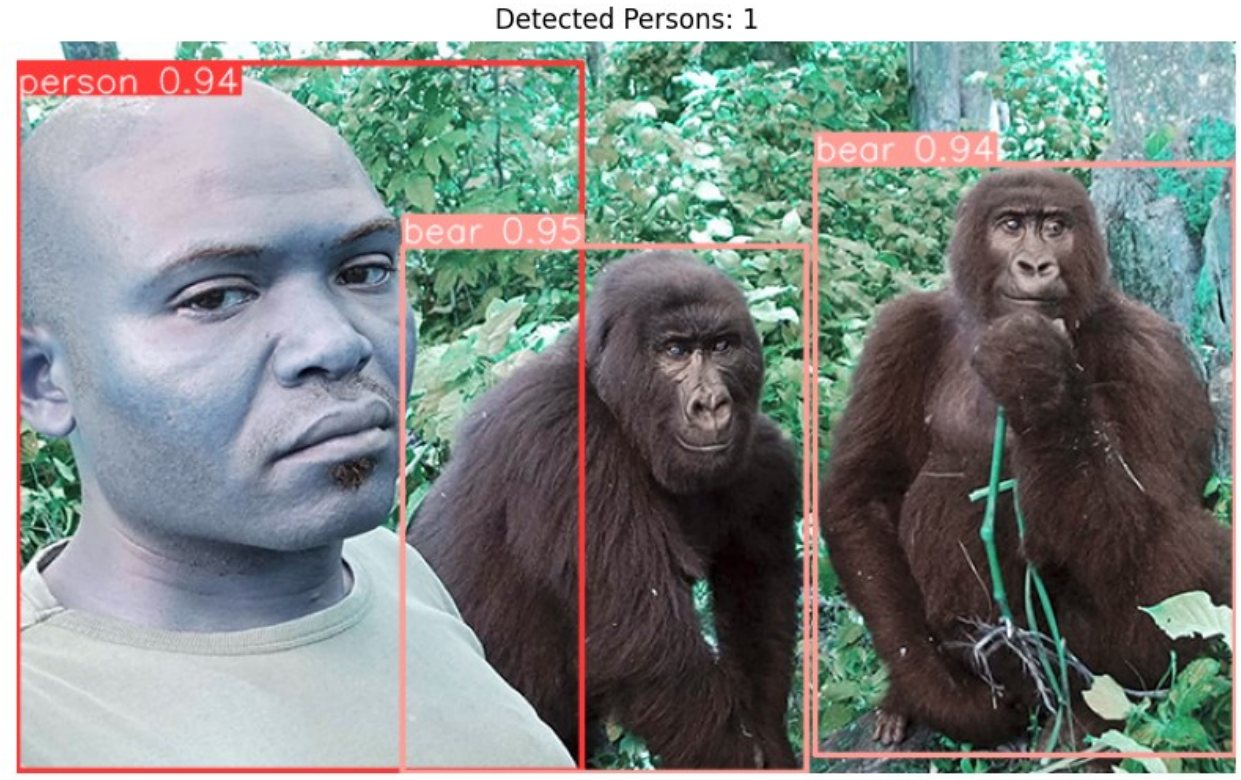
```
0: 448x640 7 persons, 1165.4ms
Speed: 5.9ms preprocess, 1165.4ms inference, 1.4ms postprocess per
image at shape (1, 3, 448, 640)
```




```
Number of people detected in the image: 7
```

```
def main():

    model = load_model('yolov8m')

    image_url =
'https://img.theepochtimes.com/assets/uploads/2020/06/03/ET-
GORILLA5784443620-700x420.jpg'
```

```
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()


0: 384x640 1 person, 2 bears, 988.2ms
Speed: 4.5ms preprocess, 988.2ms inference, 1.7ms postprocess per
image at shape (1, 3, 384, 640)
```

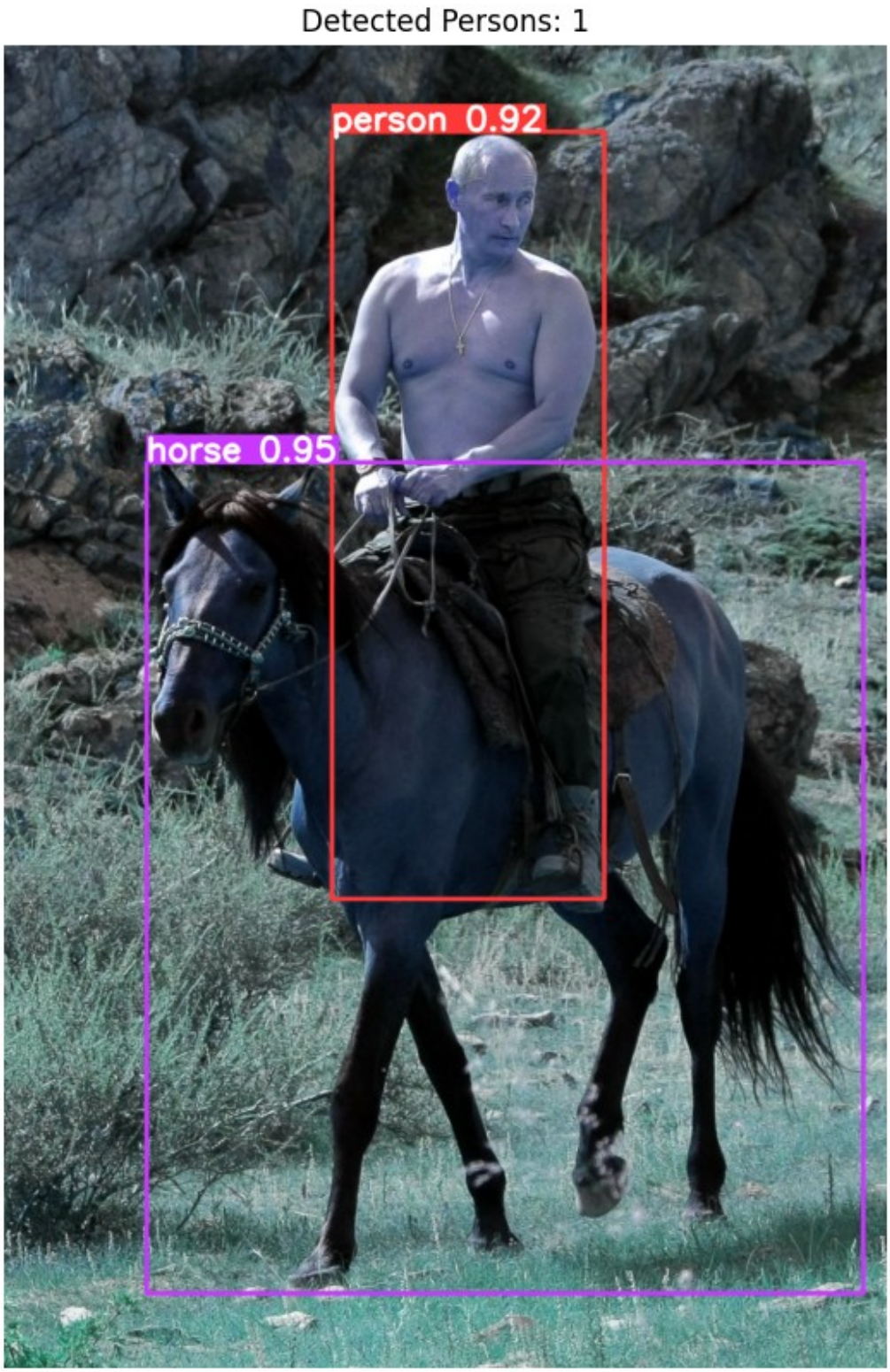



```
Number of people detected in the image: 1
```

Somehow it detects gorilla as a bear?? hmm

```python
def main():

    model = load_model('yolov8m')

    image_url = 'https://s.abcnews.com/images/International/gty-putin-horse-er-170510_2x3_1600.jpg?w=1600'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold, iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 640x448 1 person, 1 horse, 1289.6ms
Speed: 6.2ms preprocess, 1289.6ms inference, 1.4ms postprocess per image at shape (1, 3, 640, 448)
```




```
Number of people detected in the image: 1
```

```python
def main():

    model = load_model('yolov8m')

    image_url =
'https://upload.wikimedia.org/wikipedia/commons/thumb/a/a2/Transperth_
Volgren_CR228L_bodied_Volvo_B7RLE.jpg/1200px-
Transperth_Volgren_CR228L_bodied_Volvo_B7RLE.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 352x640 1 bus, 943.1ms
Speed: 5.4ms preprocess, 943.1ms inference, 1.6ms postprocess per
image at shape (1, 3, 352, 640)
```

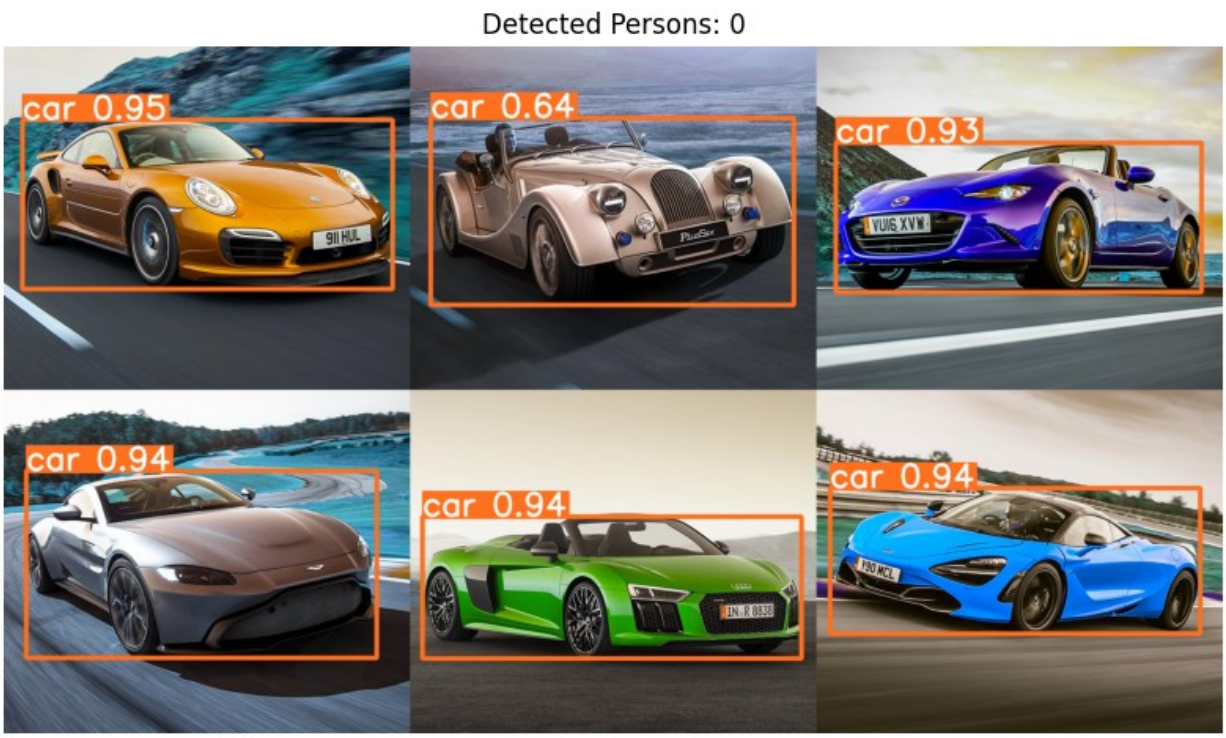
Detected Persons: 0


```
Number of people detected in the image: 0
```

```
def main():

    model = load_model('yolov8m')

    image_url =
'https://media.gq-magazine.co.uk/photos/5e29911eef785000088173ee/master/pass/20200123-Sports-Cars-09.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 384x640 6 cars, 987.8ms
```

```
Speed: 4.7ms preprocess, 987.8ms inference, 2.1ms postprocess per
image at shape (1, 3, 384, 640)
```




```
Number of people detected in the image: 0
```

```python
def main():

    model = load_model('yolov8m')

    image_url =
'https://upload.wikimedia.org/wikipedia/commons/thumb/f/fa/T-
72_Ajeya1.jpg/220px-T-72_Ajeya1.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')
```

```
if __name__ == '__main__':
    main()
```

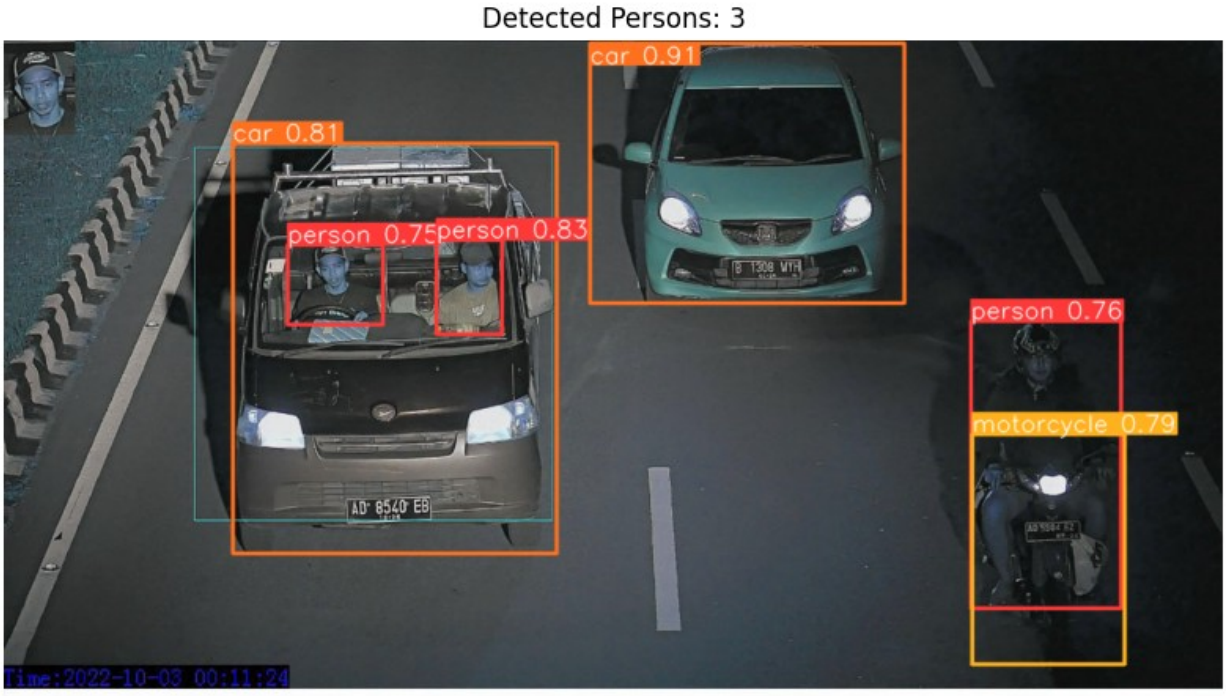
```
0: 608x640 1 person, 1786.1ms
Speed: 3.9ms preprocess, 1786.1ms inference, 1.5ms postprocess per
image at shape (1, 3, 608, 640)
```


```
Number of people detected in the image: 1
```

```
def main():

    model = load_model('yolov8m')
```

```
    image_url =
'https://www.inspirasiline.com/wp-content/uploads/2022/10/tidak-
memakai-sabuk-pengaman.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 352x640 3 persons, 2 cars, 1 motorcycle, 930.1ms
Speed: 5.2ms preprocess, 930.1ms inference, 1.6ms postprocess per
image at shape (1, 3, 352, 640)
```

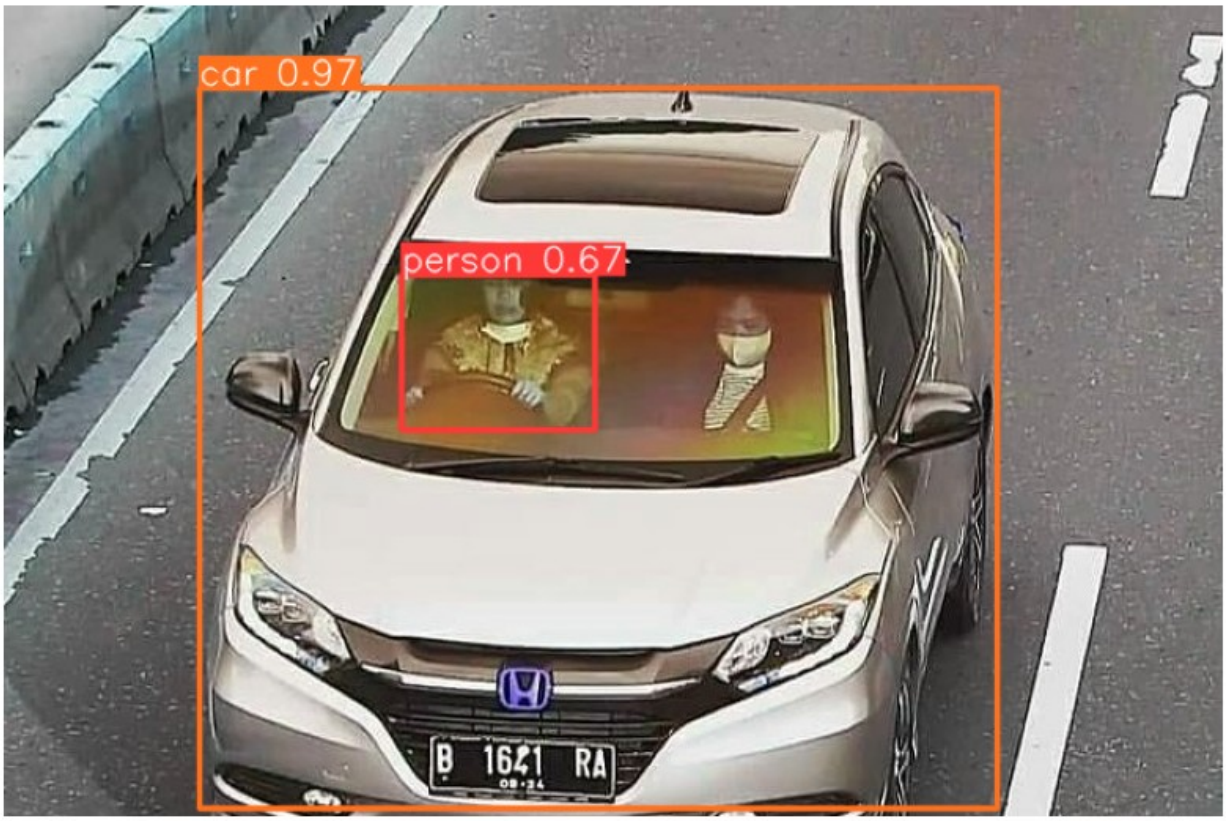



```
Number of people detected in the image: 3
```

```
def main():

    model = load_model('yolov8m')

    image_url =
'https://imgx.gridoto.com/crop/0x0:0x0/700x465/photo/2021/01/08/1258576482.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 448x640 1 person, 1 car, 1110.4ms
Speed: 4.5ms preprocess, 1110.4ms inference, 1.4ms postprocess per
image at shape (1, 3, 448, 640)
```

Detected Persons: 1



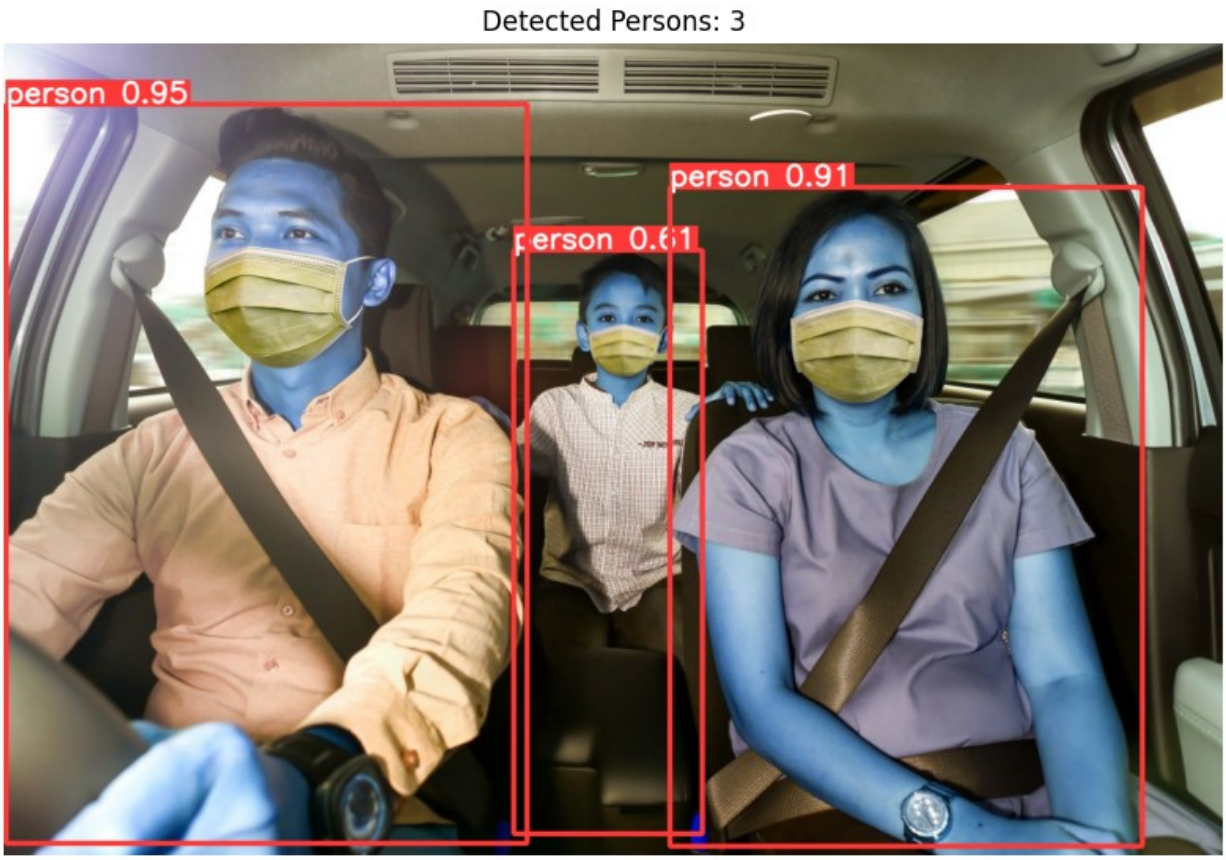

```
Number of people detected in the image: 1
```

Doesn't work on a person who wears mask

```
def main():

    model = load_model('yolov8m')

    image_url =
'https://hantaran.co/wp-content/uploads/2020/10/pengendara-mobil.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')
```

```
if __name__ == '__main__':
    main()
```

```
0: 448x640 3 persons, 1120.4ms
Speed: 4.6ms preprocess, 1120.4ms inference, 1.4ms postprocess per
image at shape (1, 3, 448, 640)
```




```
Number of people detected in the image: 3
```

```
def main():

    model = load_model('yolov8m')

    image_url = 'https://scontent-cgk1-2.xx.fbcdn.net/v/t1.6435-
9/106499612_912295805938016_3965479555057181490_n.jpg?
_nc_cat=105&ccb=1-
7&_nc_sid=5f2048&_nc_ohc=yhMUQ6l0nuEAb7jeLkS&_nc_ht=scontent-cgk1-
2.xx&oh=00_AfBj5GKJ3PiCfMqJzzOixsLpT3JNeZ_e9-l4YqcmEiolxA&oe=664596A5'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
```

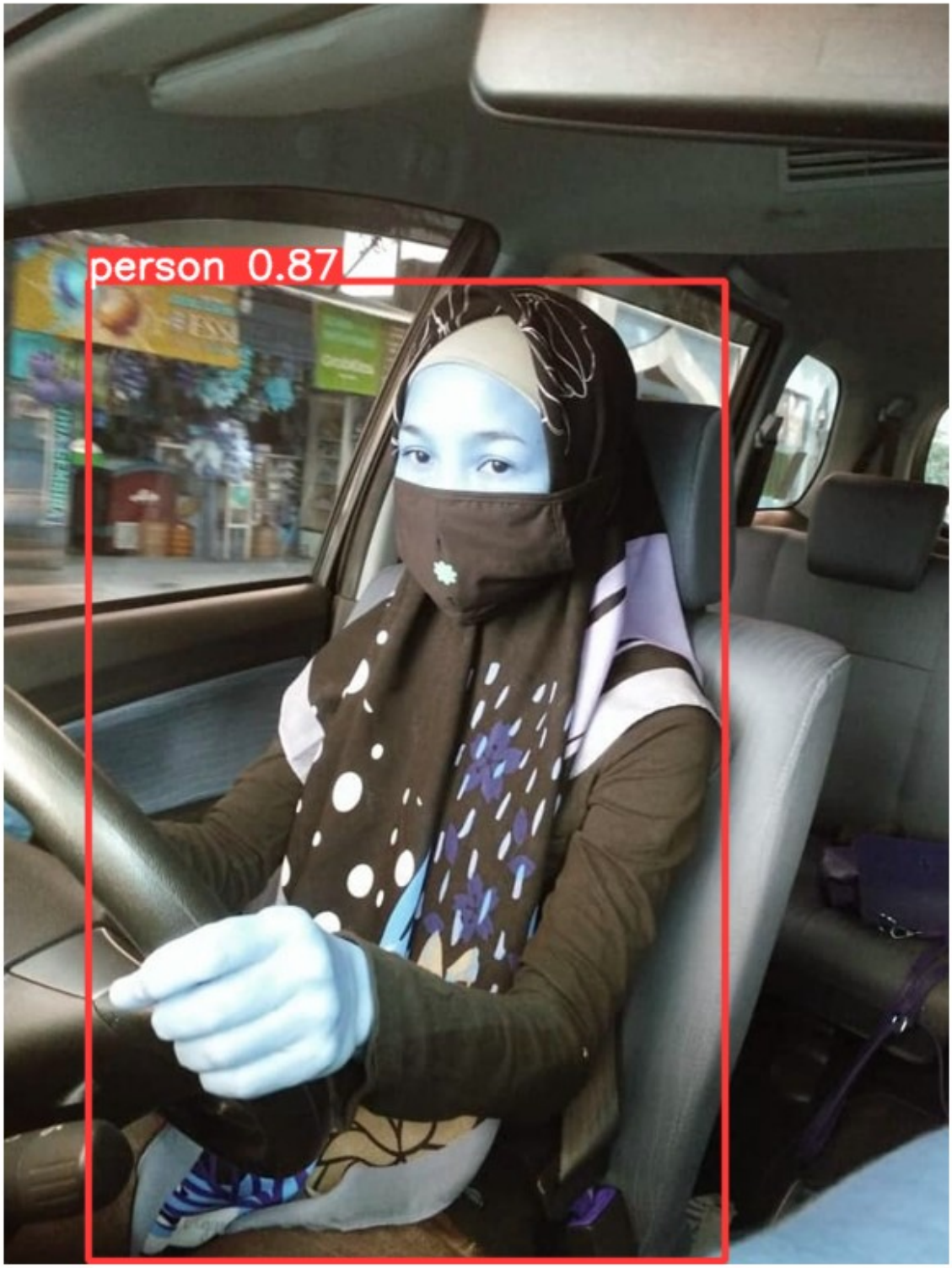
```
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 640x480 1 person, 1211.7ms
Speed: 6.5ms preprocess, 1211.7ms inference, 1.5ms postprocess per
image at shape (1, 3, 640, 480)
```


```
Number of people detected in the image: 1
```

```python
def main():

    model = load_model('yolov8m')

    image_url = 'https://scontent-cgk1-2.xx.fbcdn.net/v/t1.6435-9/78250763_912295632604700_8473819145037906527_n.jpg?_nc_cat=109&ccb=1-7&_nc_sid=5f2048&_nc_ohc=dCaRP-mpToUAb6zXZ2Z&_nc_ht=scontent-cgk1-2.xx&oh=00_AfCwHYBBTpKFs0wu38L52n25oDewNn86Sz0GBtzvnVBAkA&oe=66459A65'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold, iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 640x480 1 person, 1262.4ms
Speed: 6.1ms preprocess, 1262.4ms inference, 1.6ms postprocess per image at shape (1, 3, 640, 480)
```




```
Number of people detected in the image: 1
```

```python
def main():

    model = load_model('yolov8m')

    image_url =
'https://media.kompas.tv/library/image/content_article/article_img/
20200917090651.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

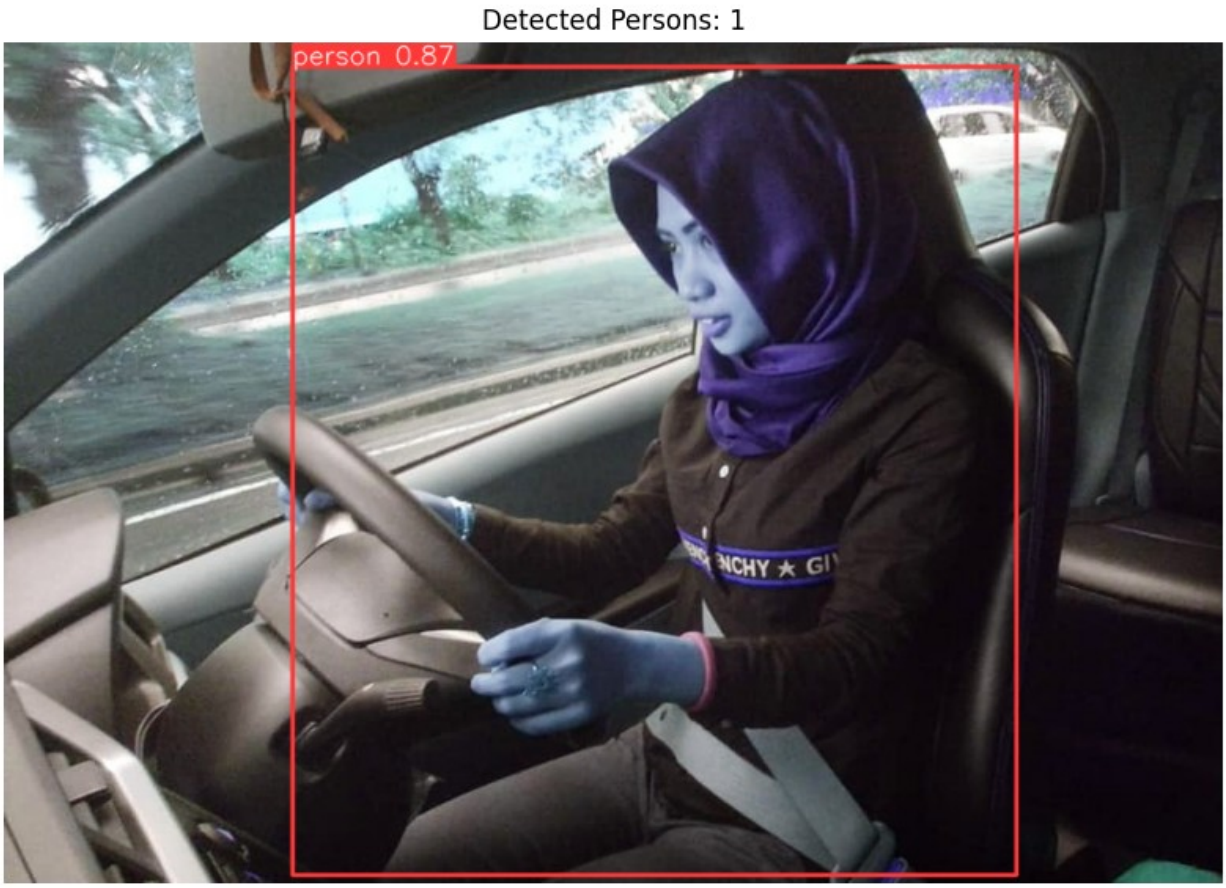
```
0: 384x640 1 person, 1 chair, 974.2ms
Speed: 3.1ms preprocess, 974.2ms inference, 1.4ms postprocess per
image at shape (1, 3, 384, 640)
```

Detected Persons: 1


```
Number of people detected in the image: 1
```

```
def main():

    model = load_model('yolov8m')

    image_url = 'https://scontent-cgk1-2.xx.fbcdn.net/v/t1.6435-
9/106729047_912295745938022_6327782745429973842_n.jpg?
_nc_cat=109&ccb=1-7&_nc_sid=5f2048&_nc_ohc=NcPKBj-GWDMAb7vX1J-
&_nc_ht=scontent-cgk1-2.xx&oh=00_AfDN-
HG7kBoKCRqhnk6OxffXr5DYkEqzAF1dqGL_KRvAYQ&oe=664591D5'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 448x640 1 person, 1525.7ms
Speed: 5.4ms preprocess, 1525.7ms inference, 1.7ms postprocess per
image at shape (1, 3, 448, 640)
```

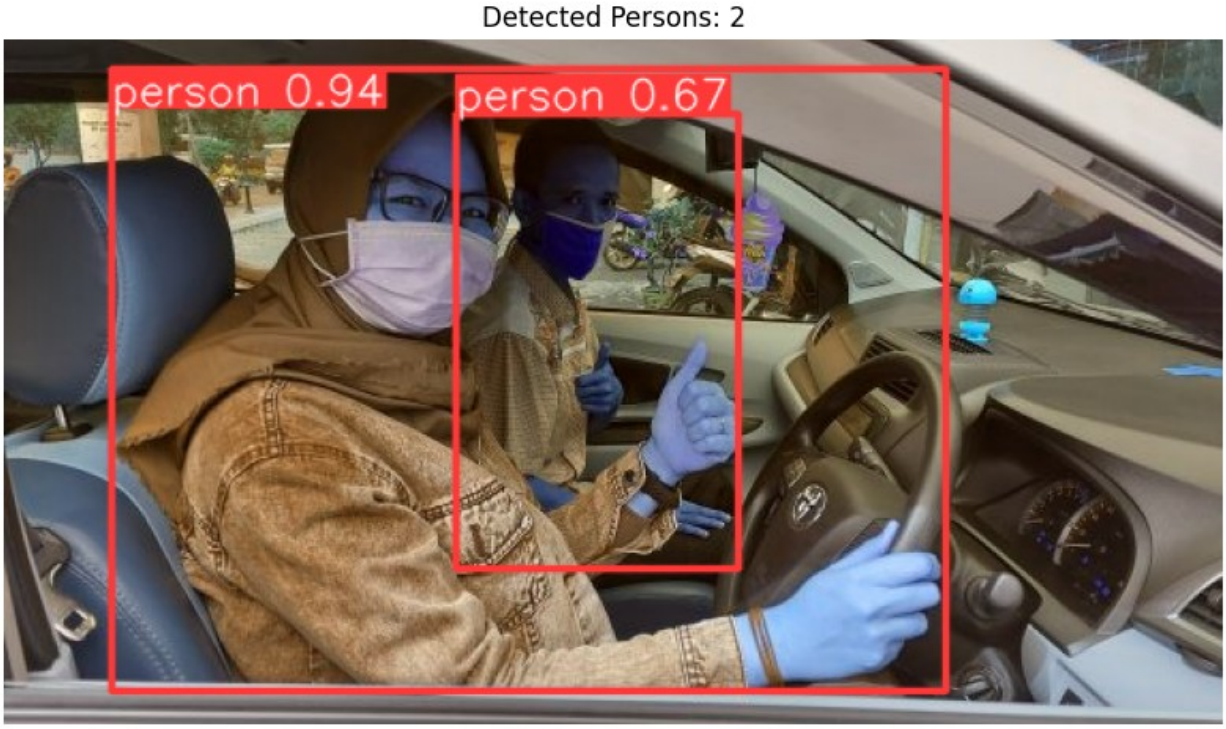



```
Number of people detected in the image: 1
```

```python
def main():

    model = load_model('yolov8m')

    image_url =
'https://satriajayanti.co.id/wp-content/uploads/2020/07/20200619_09500
6-570x320.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5
```

```
    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 384x640 2 persons, 933.9ms
Speed: 4.6ms preprocess, 933.9ms inference, 1.2ms postprocess per
image at shape (1, 3, 384, 640)
```




```
Number of people detected in the image: 2
```

```
def main():

    model = load_model('yolov8m')

    image_url =
'https://images.meesho.com/images/products/333767201/zevjc_512.webp'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)
```

```
    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()


0: 640x480 1 person, 1204.5ms
Speed: 5.0ms preprocess, 1204.5ms inference, 1.9ms postprocess per
image at shape (1, 3, 640, 480)
```


```
Number of people detected in the image: 1
```

```python
def main():

    model = load_model('yolov8m')

    image_url =
'https://media.wired.com/photos/62e1bc97d7368105da057d7f/master/w_320%
2Cc_limit/Goncalves-3313-Comic-Con-2022%2520-Culture.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 640x480 1 person, 1865.5ms
Speed: 3.7ms preprocess, 1865.5ms inference, 2.4ms postprocess per
image at shape (1, 3, 640, 480)
```

Detected Persons: 1


```
Number of people detected in the image: 1
```

```
def main():

    model = load_model('yolov8m')

    image_url =
'https://ih0.redbubble.net/image.3295242309.1693/raf,360x360,075,t,faf
afa:ca443f4786.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()


0: 640x640 1 person, 1653.4ms
Speed: 5.4ms preprocess, 1653.4ms inference, 1.7ms postprocess per
image at shape (1, 3, 640, 640)
```


```
Number of people detected in the image: 1
```

```
def main():

    model = load_model('yolov8m')

    image_url =
'https://awsimages.detik.net.id/community/media/visual/2022/10/22/
kanye-west-dan-balenciaga-1_34.jpeg?w=700&q=90'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
```

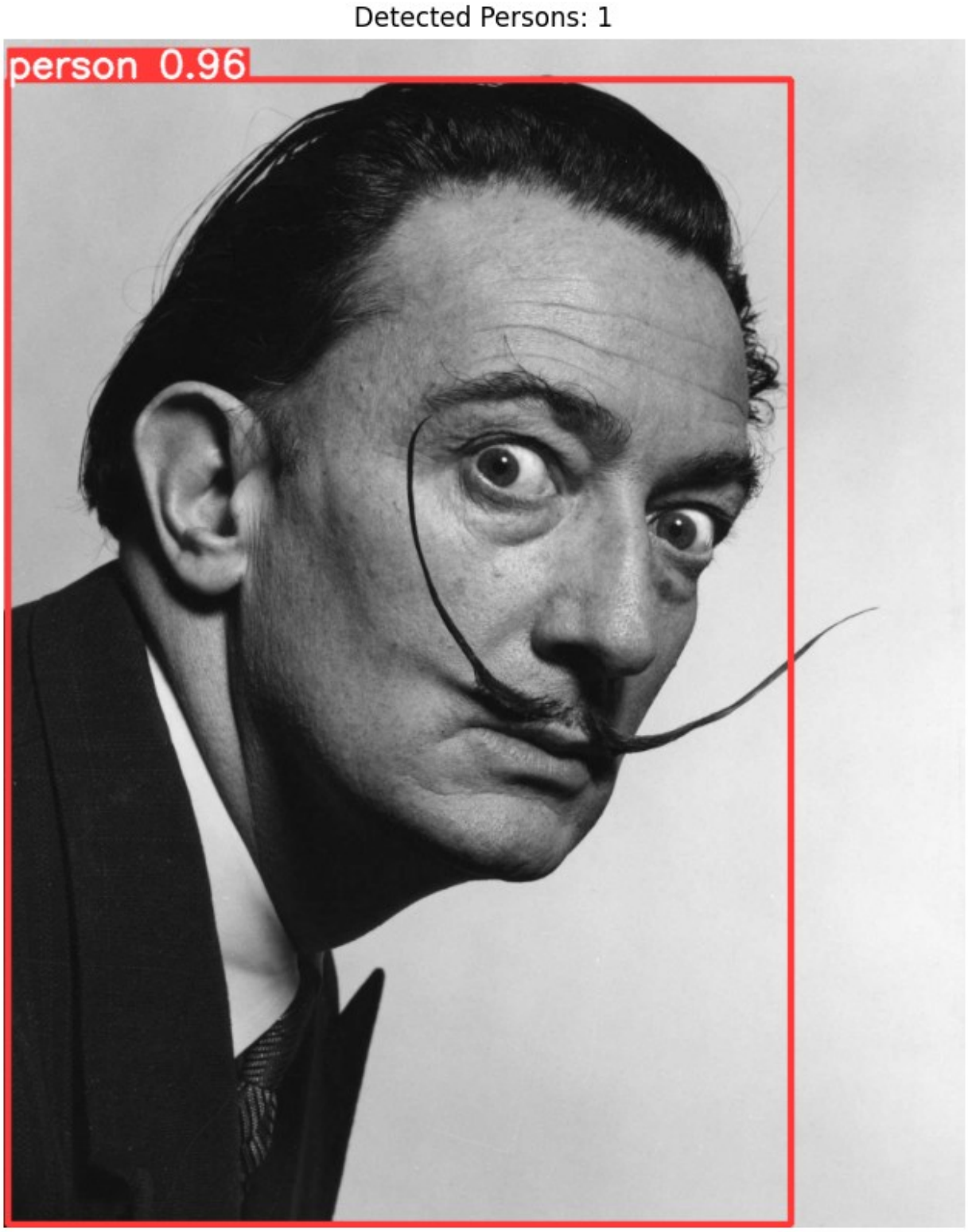
```
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 640x512 1 person, 1 umbrella, 1324.9ms
Speed: 6.2ms preprocess, 1324.9ms inference, 1.4ms postprocess per
image at shape (1, 3, 640, 512)
```


```
Number of people detected in the image: 1
```

```python
def main():

    model = load_model('yolov8m')

    image_url =
'https://assets-a1.kompasiana.com/items/album/2017/07/21/dali-
597159d24fc4aa25a3640862.jpg?t=o&v=770'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5


    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 640x544 1 person, 1427.9ms
Speed: 5.9ms preprocess, 1427.9ms inference, 2.0ms postprocess per
image at shape (1, 3, 640, 544)
```




```
Number of people detected in the image: 1
```

```python
def main():

    model = load_model('yolov8m')

    image_url =
'https://media.gqitalia.it/photos/65d4689834d744a8a641c65f/16:9/w_2560
%2Cc_limit/dune2.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 384x640 1 person, 980.6ms
Speed: 5.3ms preprocess, 980.6ms inference, 1.3ms postprocess per
image at shape (1, 3, 384, 640)
```


```
Number of people detected in the image: 1
```

YOLOv8x (Most Accurate but the Slowest)

```
def main():

    model = load_model('yolov8x')

    image_url = 'https://img.freepik.com/free-photo/front-view-smiley-
people-group-therapy-session_23-2148752041.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

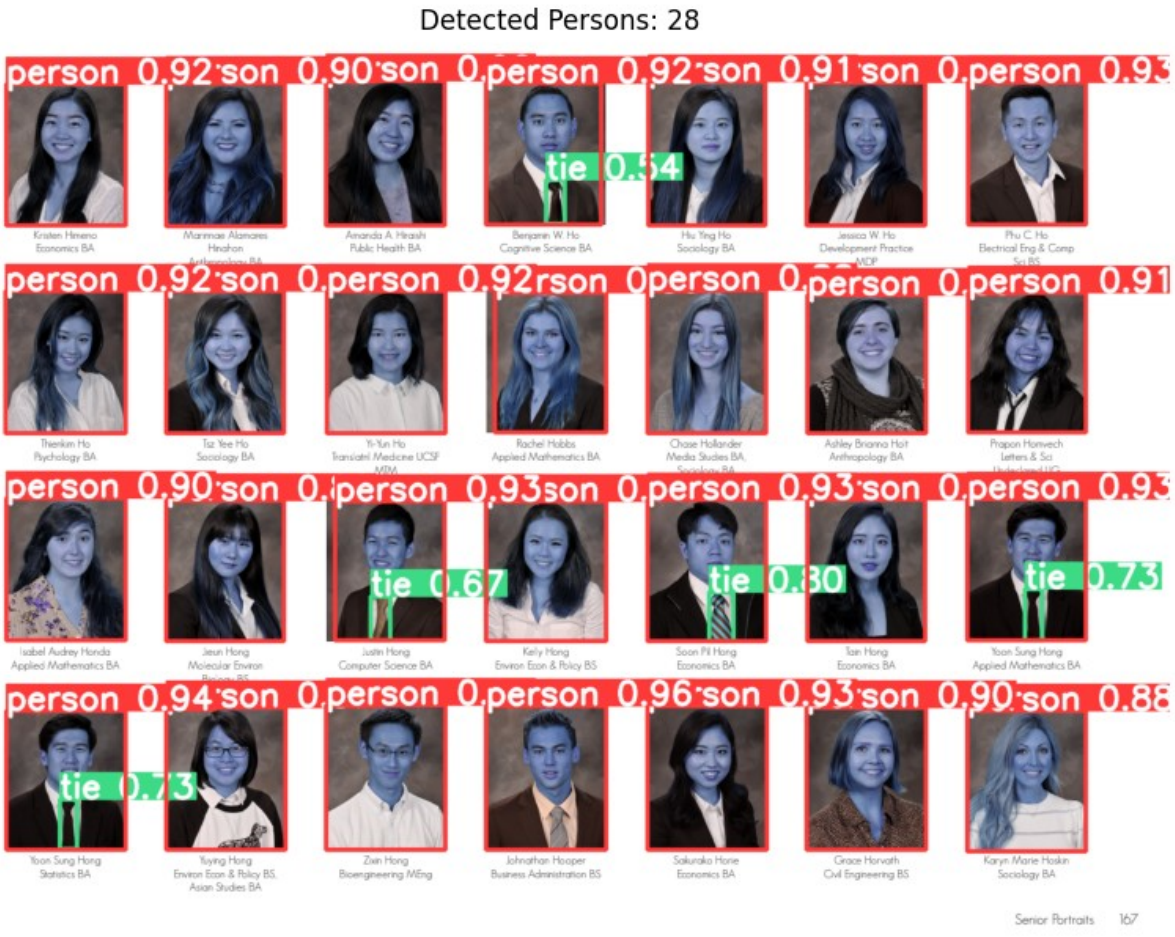
```
Downloading
https://github.com/ultralytics/assets/releases/download/v8.2.0/yolov8x
.pt to 'yolov8x.pt'...

100%|██████████| 131M/131M [00:00<00:00, 188MB/s]


0: 448x640 6 persons, 3198.2ms
Speed: 4.1ms preprocess, 3198.2ms inference, 1.5ms postprocess per
image at shape (1, 3, 448, 640)
```


```
Number of people detected in the image: 6
```

```python
def main():

    model = load_model('yolov8x')

    image_url =
'https://www.ocf.berkeley.edu/~bluegold/static/img/portraits_icons/
portraits.png'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
```

```
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 480x640 28 persons, 5 ties, 4338.0ms
Speed: 5.6ms preprocess, 4338.0ms inference, 5.3ms postprocess per
image at shape (1, 3, 480, 640)
```




```
Number of people detected in the image: 28
```

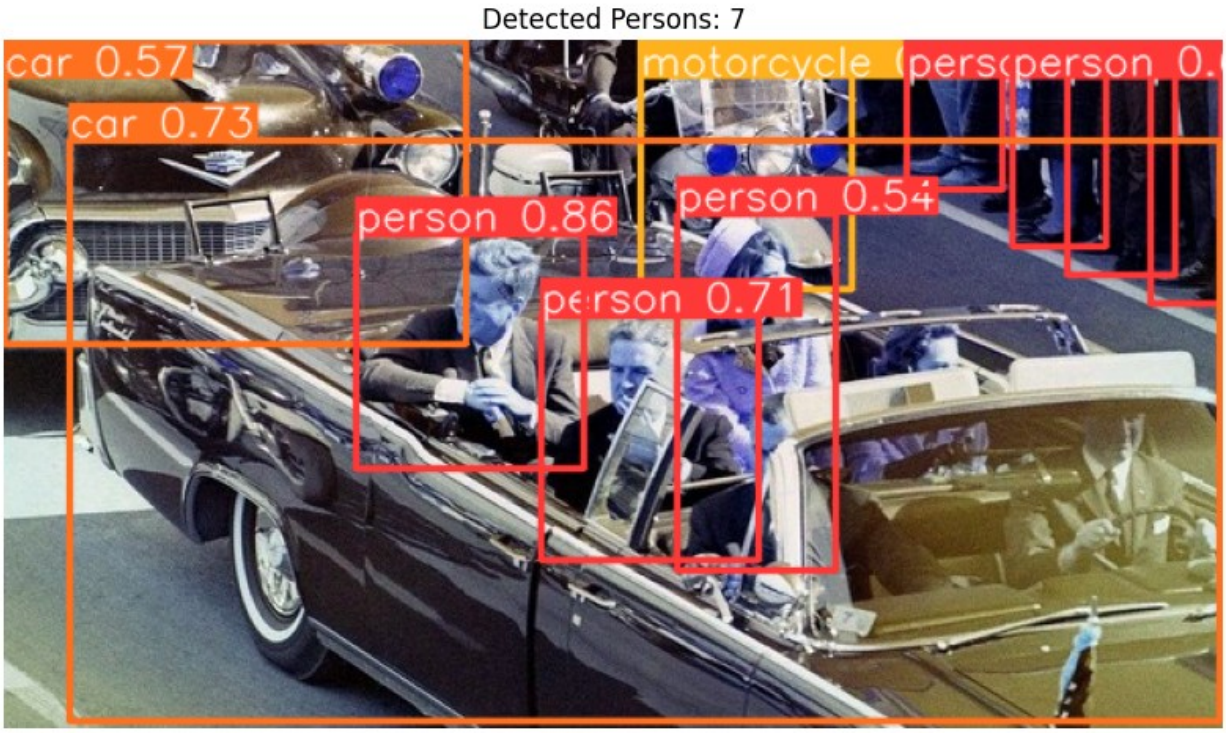
```python
def main():

    model = load_model('yolov8x')

    image_url =
'https://www.security-ligue.org/fileadmin/_processed_/c/c/csm_crowd_9a
2fce0777.jpg'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()
```

```
0: 384x640 5 persons, 2894.8ms
Speed: 6.2ms preprocess, 2894.8ms inference, 1.8ms postprocess per
image at shape (1, 3, 384, 640)
```


```
Number of people detected in the image: 5
```

```
def main():

    model = load_model('yolov8x')

    image_url =
'https://awsimages.detik.net.id/community/media/visual/2023/10/09/
misteri-abadi-yang-masih-meliputi-pembunuhan-jfk.jpeg?w=600&q=90'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')

if __name__ == '__main__':
    main()


0: 384x640 7 persons, 2 cars, 1 motorcycle, 2843.9ms
```

```
Speed: 3.9ms preprocess, 2843.9ms inference, 2.4ms postprocess per
image at shape (1, 3, 384, 640)
```




```
Number of people detected in the image: 7
```

```python
def main():

    model = load_model('yolov8x')

    image_url =
'https://media.cnn.com/api/v1/images/stellar/prod/gettyimages-
1254495486.jpg?c=original'
    image_path = 'test_image.jpg'

    download_image(image_url, image_path)
    image_rgb = load_image(image_path)

    conf_threshold = 0.5
    iou_threshold = 0.5

    results = perform_inference(model, image_rgb, conf=conf_threshold,
iou=iou_threshold)

    person_count = visualize_results(results)
    print(f'Number of people detected in the image: {person_count}')
```

```
if __name__ == '__main__':
    main()
```

```
0: 448x640 (no detections), 4006.8ms
Speed: 6.0ms preprocess, 4006.8ms inference, 2.2ms postprocess per
image at shape (1, 3, 448, 640)
```


```
Number of people detected in the image: 0
```