1. Environment Setup

```
from google.colab import files
files.upload()
!unzip /content/Backdoor.v2i.yolov8.zip
```

```
Choose Files  Backdoor.v2i.yolov8.zip
• Backdoor.v2i.yolov8.zip(application/x-zip-compressed) - 1309787 bytes, last modified: 4/15/2025 - 100% done
Saving Backdoor.v2i.yolov8.zip to Backdoor.v2i.yolov8.zip
Archive:  /content/Backdoor.v2i.yolov8.zip
  inflating: README.dataset.txt
  inflating: README.roboflow.txt
  inflating: data.yaml
   creating: train/
   creating: train/images/
 extracting: train/images/expected-output-of-task5_png.rf.0836c89ca57169198e58c31830ae64ce.jpg
 extracting: train/images/expected-output-of-task5_png.rf.580e558983f1dc91a4129ae082c8adcf.jpg
 extracting: train/images/expected-output-of-task5_png.rf.5db217d79fdff4cd1708d4c3219b7f04.jpg
 extracting: train/images/suggested-8QVd4a4j1JPhrrza2kNh_jpg.rf.690e6a10adb2c64a3a7d9d0f823b82d9.jpg
 extracting: train/images/suggested-8QVd4a4j1JPhrrza2kNh_jpg.rf.6ab2cc6eaff6bbffc7e114fbcd0bf3dd.jpg
 extracting: train/images/suggested-8QVd4a4j1JPhrrza2kNh_jpg.rf.b2292c338d6bf75da3976df6860462e3.jpg
 extracting: train/images/suggested-LsRLygFjdRNj9lH0KU7A_jpg.rf.551b489d3e8994025acee8d6da3e485f.jpg
 extracting: train/images/suggested-LsRLygFjdRNj9lH0KU7A_jpg.rf.8416a0a2c976393ecb1e5733e25906ce.jpg
 extracting: train/images/suggested-LsRLygFjdRNj9lH0KU7A_jpg.rf.850ec926673073b6fe638636e19c3799.jpg
 extracting: train/images/suggested-Orse5CnMV4NLsmN5ZHu5_jpg.rf.617be87100f2c8f08cc114fb61c73ea4.jpg
 extracting: train/images/suggested-Orse5CnMV4NLsmN5ZHu5_jpg.rf.8b2fcb6229cae07f459aee6d8472ae33.jpg
 extracting: train/images/suggested-Orse5CnMV4NLsmN5ZHu5_jpg.rf.d95015ddbd0207b308874916905bc9dd.jpg
 extracting: train/images/suggested-RO3r8ll5ycSBtmGYYjlV_jpg.rf.04f4ce14e36329c5b80ee80e4c998fe0.jpg
 extracting: train/images/suggested-RO3r8ll5ycSBtmGYYjlV_jpg.rf.b156dc4c636a66a7bf083559405fd9c1.jpg
 extracting: train/images/suggested-RO3r8ll5ycSBtmGYYjlV_jpg.rf.b76d98e3ee7d584d01c2c5176e3e58e9.jpg
 extracting: train/images/task5_img_png.rf.280d2aa52d143970e6b61d9de10b2310.jpg
 extracting: train/images/task5_img_png.rf.ba9f58498df3821a5fbdd12b63d1062c.jpg
 extracting: train/images/task5_img_png.rf.ed8c7a6fbbc7cdf1945ed8482f2d11f0.jpg
   creating: train/labels/
  inflating: train/labels/expected-output-of-task5_png.rf.0836c89ca57169198e58c31830ae64ce.txt
  inflating: train/labels/expected-output-of-task5_png.rf.580e558983f1dc91a4129ae082c8adcf.txt
  inflating: train/labels/expected-output-of-task5_png.rf.5db217d79fdff4cd1708d4c3219b7f04.txt
  inflating: train/labels/suggested-8QVd4a4j1JPhrrza2kNh_jpg.rf.690e6a10adb2c64a3a7d9d0f823b82d9.txt
  inflating: train/labels/suggested-8QVd4a4j1JPhrrza2kNh_jpg.rf.6ab2cc6eaff6bbffc7e114fbcd0bf3dd.txt
  inflating: train/labels/suggested-8QVd4a4j1JPhrrza2kNh_jpg.rf.b2292c338d6bf75da3976df6860462e3.txt
  inflating: train/labels/suggested-LsRLygFjdRNj9lH0KU7A_jpg.rf.551b489d3e8994025acee8d6da3e485f.txt
  inflating: train/labels/suggested-LsRLygFjdRNj9lH0KU7A_jpg.rf.8416a0a2c976393ecb1e5733e25906ce.txt
  inflating: train/labels/suggested-LsRLygFjdRNj9lH0KU7A_jpg.rf.850ec926673073b6fe638636e19c3799.txt
  inflating: train/labels/suggested-Orse5CnMV4NLsmN5ZHu5_jpg.rf.617be87100f2c8f08cc114fb61c73ea4.txt
  inflating: train/labels/suggested-Orse5CnMV4NLsmN5ZHu5_jpg.rf.8b2fcb6229cae07f459aee6d8472ae33.txt
  inflating: train/labels/suggested-Orse5CnMV4NLsmN5ZHu5_jpg.rf.d95015ddbd0207b308874916905bc9dd.txt
  inflating: train/labels/suggested-RO3r8ll5ycSBtmGYYjlV_jpg.rf.04f4ce14e36329c5b80ee80e4c998fe0.txt
  inflating: train/labels/suggested-RO3r8ll5ycSBtmGYYjlV_jpg.rf.b156dc4c636a66a7bf083559405fd9c1.txt
  inflating: train/labels/suggested-RO3r8ll5ycSBtmGYYjlV_jpg.rf.b76d98e3ee7d584d01c2c5176e3e58e9.txt
  inflating: train/labels/task5_img_png.rf.280d2aa52d143970e6b61d9de10b2310.txt
  inflating: train/labels/task5_img_png.rf.ba9f58498df3821a5fbdd12b63d1062c.txt
  inflating: train/labels/task5_img_png.rf.ed8c7a6fbbc7cdf1945ed8482f2d11f0.txt
```

```
!pip install ultralytics
```

```
Collecting ultralytics
  Downloading ultralytics-8.3.108-py3-none-any.whl.metadata (37 kB)
Requirement already satisfied: numpy<=2.1.1,>=1.23.0 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (2.0.2)
Requirement already satisfied: matplotlib>=3.3.0 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (3.10.0)
Requirement already satisfied: opencv-python>=4.6.0 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (4.11.0.86)
Requirement already satisfied: pillow>=7.1.2 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (11.1.0)
Requirement already satisfied: pyyaml>=5.3.1 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (6.0.2)
Requirement already satisfied: requests>=2.23.0 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (2.32.3)
Requirement already satisfied: scipy>=1.4.1 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (1.14.1)
Requirement already satisfied: torch>=1.8.0 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (2.6.0+cu124)
Requirement already satisfied: torchvision>=0.9.0 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (0.21.0+cu124)
Requirement already satisfied: tqdm>=4.64.0 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (4.67.1)
Requirement already satisfied: psutil in /usr/local/lib/python3.11/dist-packages (from ultralytics) (5.9.5)
Requirement already satisfied: py-cpuinfo in /usr/local/lib/python3.11/dist-packages (from ultralytics) (9.0.0)
Requirement already satisfied: pandas>=1.1.4 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (2.2.2)
Requirement already satisfied: seaborn>=0.11.0 in /usr/local/lib/python3.11/dist-packages (from ultralytics) (0.13.2)
Collecting ultralytics-thop>=2.0.0 (from ultralytics)
  Downloading ultralytics_thop-2.0.14-py3-none-any.whl.metadata (9.4 kB)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib>=3.3.0->ultralytics) (
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.11/dist-packages (from matplotlib>=3.3.0->ultralytics) (0.1
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib>=3.3.0->ultralytics)
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib>=3.3.0->ultralytics)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib>=3.3.0->ultralytics) (
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib>=3.3.0->ultralytics)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.11/dist-packages (from matplotlib>=3.3.0->ultralyti
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas>=1.1.4->ultralytics) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas>=1.1.4->ultralytics) (2025.
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests>=2.23.0->ultraly
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests>=2.23.0->ultralytics) (3.10)
```

```
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests>=2.23.0->ultralytics)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests>=2.23.0->ultralytics)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from torch>=1.8.0->ultralytics) (3.18.0)
Requirement already satisfied: typing-extensions>=4.10.0 in /usr/local/lib/python3.11/dist-packages (from torch>=1.8.0->ultralytic
Requirement already satisfied: networkx in /usr/local/lib/python3.11/dist-packages (from torch>=1.8.0->ultralytics) (3.4.2)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.11/dist-packages (from torch>=1.8.0->ultralytics) (3.1.6)
Requirement already satisfied: fsspec in /usr/local/lib/python3.11/dist-packages (from torch>=1.8.0->ultralytics) (2025.3.2)
Collecting nvidia-cuda-nvrtc-cu12==12.4.127 (from torch>=1.8.0->ultralytics)
  Downloading nvidia_cuda_nvrtc_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
Collecting nvidia-cuda-runtime-cu12==12.4.127 (from torch>=1.8.0->ultralytics)
  Downloading nvidia_cuda_runtime_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
Collecting nvidia-cuda-cupti-cu12==12.4.127 (from torch>=1.8.0->ultralytics)
  Downloading nvidia_cuda_cupti_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)
Collecting nvidia-cudnn-cu12==9.1.0.70 (from torch>=1.8.0->ultralytics)
  Downloading nvidia_cudnn_cu12-9.1.0.70-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)
Collecting nvidia-cublas-cu12==12.4.5.8 (from torch>=1.8.0->ultralytics)
  Downloading nvidia_cublas_cu12-12.4.5.8-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
Collecting nvidia-cufft-cu12==11.2.1.3 (from torch>=1.8.0->ultralytics)
  Downloading nvidia_cufft_cu12-11.2.1.3-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
Collecting nvidia-curand-cu12==10.3.5.147 (from torch>=1.8.0->ultralytics)
  Downloading nvidia_curand_cu12-10.3.5.147-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
Collecting nvidia-cusolver-cu12==11.6.1.9 (from torch>=1.8.0->ultralytics)
  Downloading nvidia_cusolver_cu12-11.6.1.9-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)
Collecting nvidia-cusparse-cu12==12.3.1.170 (from torch>=1.8.0->ultralytics)
  Downloading nvidia_cusparse_cu12-12.3.1.170-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)
Requirement already satisfied: nvidia-cusparselt-cu12==0.6.2 in /usr/local/lib/python3.11/dist-packages (from torch>=1.8.0->ultrali
Requirement already satisfied: nvidia-nccl-cu12==2.21.5 in /usr/local/lib/python3.11/dist-packages (from torch>=1.8.0->ultralytics
```

```
!echo -e "train: /content/train/images\nval: /content/train/images\nnc: 1\nnames: ['container_door']" > /content/data.yaml
```

Task 5: Object Extraction - Back Door of a Container

Objective: Extract only the back door from the container image(task5_img).

Yolo8-Seg With Cropped

```
from ultralytics import YOLO
model = YOLO('yolov8n-seg.pt')
model.train(data='/content/data.yaml', epochs=20, imgsz=640, patience=5)
```

```
0.4034,      0.4044,     0.40541,     0.40641,     0.40741,
```

```
            0.40841,     0.40941,     0.41041,     0.41141,     0.41241,     0.41341,     0.41441,     0.41542,     0.41642,
     0.41742,     0.41842,     0.41942,     0.42042,     0.42142,     0.42242,     0.42342,     0.42442,     0.42543,     0.42643,
     0.42743,     0.42843,     0.42943,     0.43043,     0.43143,
            0.43243,     0.43343,     0.43443,     0.43544,     0.43644,     0.43744,     0.43844,     0.43944,     0.44044,
     0.44144,     0.44244,     0.44344,     0.44444,     0.44545,     0.44645,     0.44745,     0.44845,     0.44945,     0.45045,
     0.45145,     0.45245,     0.45345,     0.45445,     0.45546,
            0.45646,     0.45746,     0.45846,     0.45946,     0.46046,     0.46146,     0.46246,     0.46346,     0.46446,
     0.46547,     0.46647,     0.46747,     0.46847,     0.46947,     0.47047,     0.47147,     0.47247,     0.47347,     0.47447,
     0.47548,     0.47648,     0.47748,     0.47848,     0.47948,
            0.48048,     0.48148,     0.48248,     0.48348,     0.48448,     0.48549,     0.48649,     0.48749,     0.48849,
     0.48949,     0.49049,     0.49149,     0.49249,     0.49349,     0.49449,      0.4955,      0.4965,      0.4975,      0.4985,
     0.4995,      0.5005,      0.5015,      0.5025,      0.5035,
             0.5045,     0.50551,     0.50651,     0.50751,     0.50851,     0.50951,     0.51051,     0.51151,     0.51251,
     0.51351,     0.51451,     0.51552,     0.51652,     0.51752,     0.51852,     0.51952,     0.52052,     0.52152,     0.52252,
     0.52352,     0.52452,     0.52553,     0.52653,     0.52753,
```

```
model = YOLO('/content/runs/segment/train/weights/best.pt')
results = model.predict(source='/content/task5_img.png', conf=0.015, save=True)
```

```
image 1/1 /content/task5_img.png: 352x640 1 container_door, 215.8ms
Speed: 2.8ms preprocess, 215.8ms inference, 18.5ms postprocess per image at shape (1, 3, 352, 640)
Results saved to runs/segment/predict
```

```
from IPython.display import Image
Image('/content/runs/segment/predict/task5_img.jpg')
```

Double-click (or enter) to edit

```
# 1. Import the necessary libraries
from ultralytics import YOLO
import cv2
import matplotlib.pyplot as plt

# 2. Load the trained model
model = YOLO('/content/runs/segment/train/weights/best.pt')

# 3. Perform prediction on the image
results = model.predict(source='/content/task5_img.png', conf=0.0148, save=True)

# 4. Read the original image using OpenCV
image = cv2.imread('/content/task5_img.png')

# 5. Extract the Bounding Box Coordinates
for result in results:
    if result.boxes:  # If there are detections
        for box in result.boxes:
            # Extract the coordinates
            x_min, y_min, x_max, y_max = box.xyxy[0]  # xyxy is the bounding box coordinates
            x_min, y_min, x_max, y_max = int(x_min), int(y_min), int(x_max), int(y_max)

            # 6. Crop the image using the coordinates
            cropped_image = image[y_min:y_max, x_min:x_max]

            # 7. Save the cropped image
            cv2.imwrite('/content/cropped_container_door.png', cropped_image)

            # 8. (Optional) Display the cropped image
            plt.imshow(cv2.cvtColor(cropped_image, cv2.COLOR_BGR2RGB))
            plt.axis('off')
            plt.show()
```

```
image 1/1 /content/task5_img.png: 352x640 1 container_door, 71.5ms
Speed: 14.8ms preprocess, 71.5ms inference, 12.2ms postprocess per image at shape (1, 3, 352, 640)
Results saved to runs/segment/predict2
```

Cropped_Coordinate_container_door

```
import cv2
import numpy as np
import matplotlib.pyplot as plt

# 1. Read the original image
image_path = '/content/task5_img.png'  # Replace with the path to your image
image = cv2.imread(image_path)
height, width = image.shape[:2]  # Dimensions of the image (height and width)

# 2. Polygon coordinates from the label file
label = "0 0.4854489796875 0.90787209375 0.7487017265625 0.699909884375 0.797004709375 0.04025 0.4866562015625 0.1654738375 0.48544897968
label = label.split()  # Split the line into a list

# Extract the coordinates (ignore the first value as it is the class ID)
coords = list(map(float, label[1:]))  # Convert the values to numeric data
points = [(coords[i], coords[i+1]) for i in range(0, len(coords), 2)]  # Convert the values into (x, y) pairs

# 3. Convert the coordinates from normalized to pixels
points_pixels = [(int(x * width), int(y * height)) for x, y in points]

# 4. Create an empty mask with the same size as the image
mask = np.zeros((height, width), dtype=np.uint8)

# 5. Draw the polygon on the mask
points_array = np.array(points_pixels, dtype=np.int32)
cv2.fillPoly(mask, [points_array], 255)  # Fill the polygon with white (255)

# 6. Apply the mask to the original image
masked_image = cv2.bitwise_and(image, image, mask=mask)

# 7. (Optional) Extract the bounding box for precise cropping around the polygon
x, y, w, h = cv2.boundingRect(points_array)  # Get the bounding rectangle around the polygon
cropped_image = masked_image[y:y+h, x:x+w]

# 8. Save the cropped image
cv2.imwrite('/content/segmented_container_door.png', cropped_image)

# 9. Display the cropped image
plt.imshow(cv2.cvtColor(cropped_image, cv2.COLOR_BGR2RGB))
plt.axis('off')
plt.show()
```


```
Start coding or generate with AI.
```