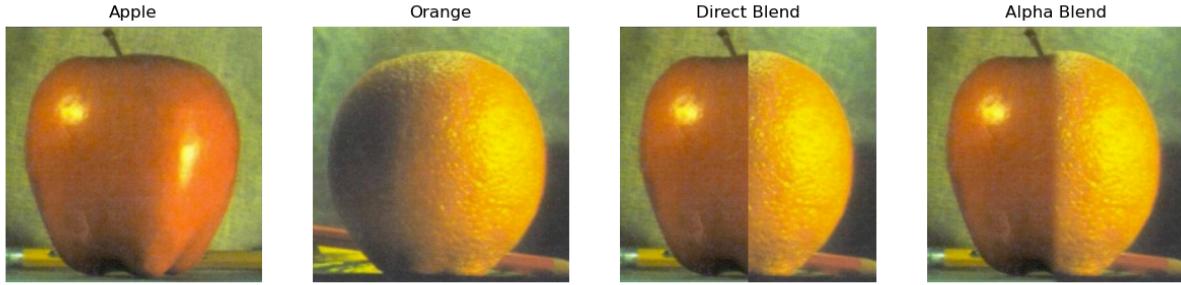


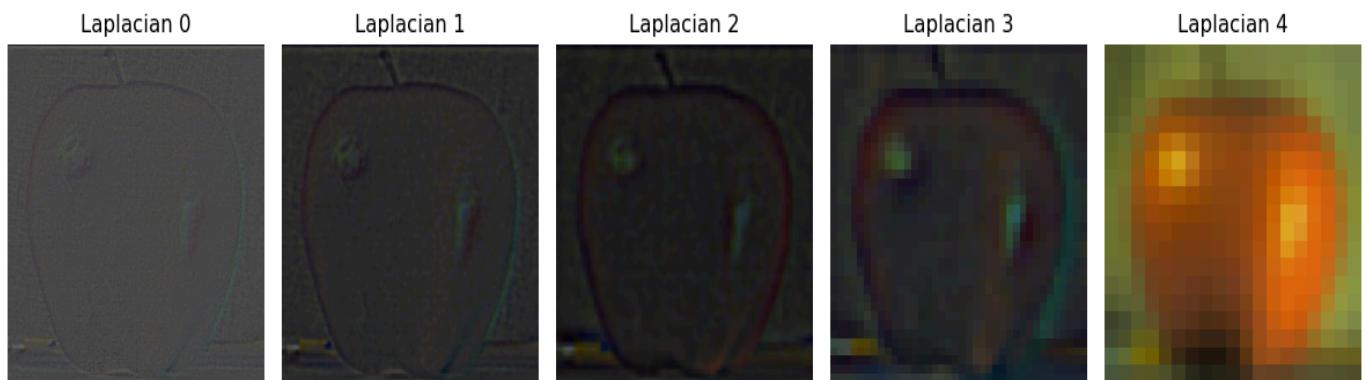
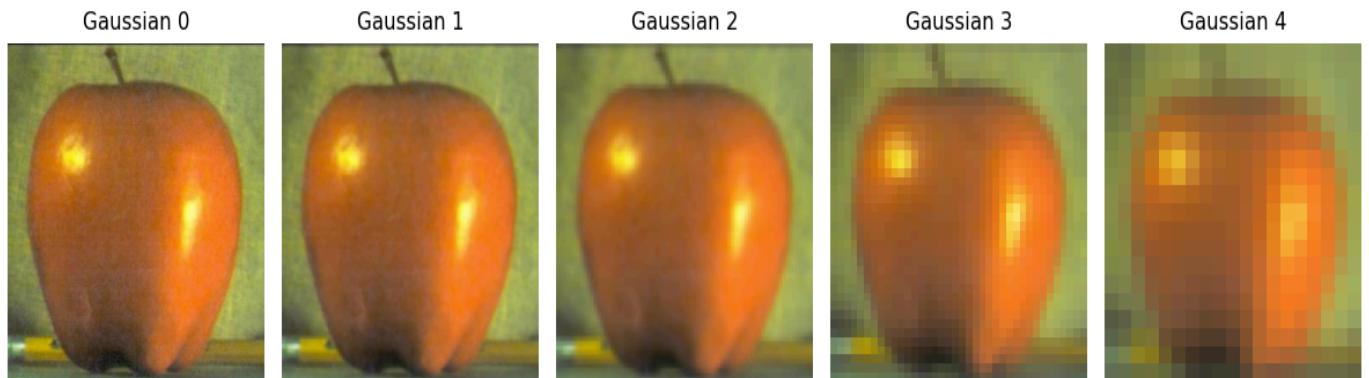
CMSC 472/672 Computer Vision
Homework 2
Submitted By: Garvita Jain (TR41373)

1. Multiresolution Image Blending

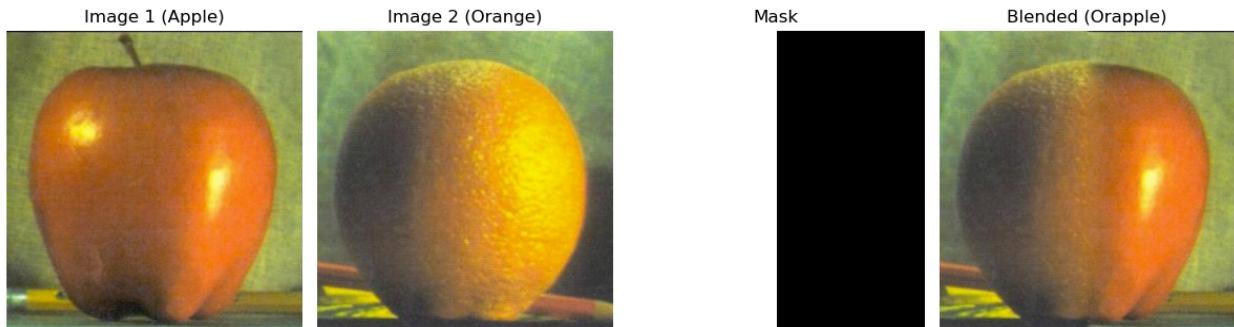
Task - 1: Code in the zip, file “question1_task1_AlphaBlend.py”



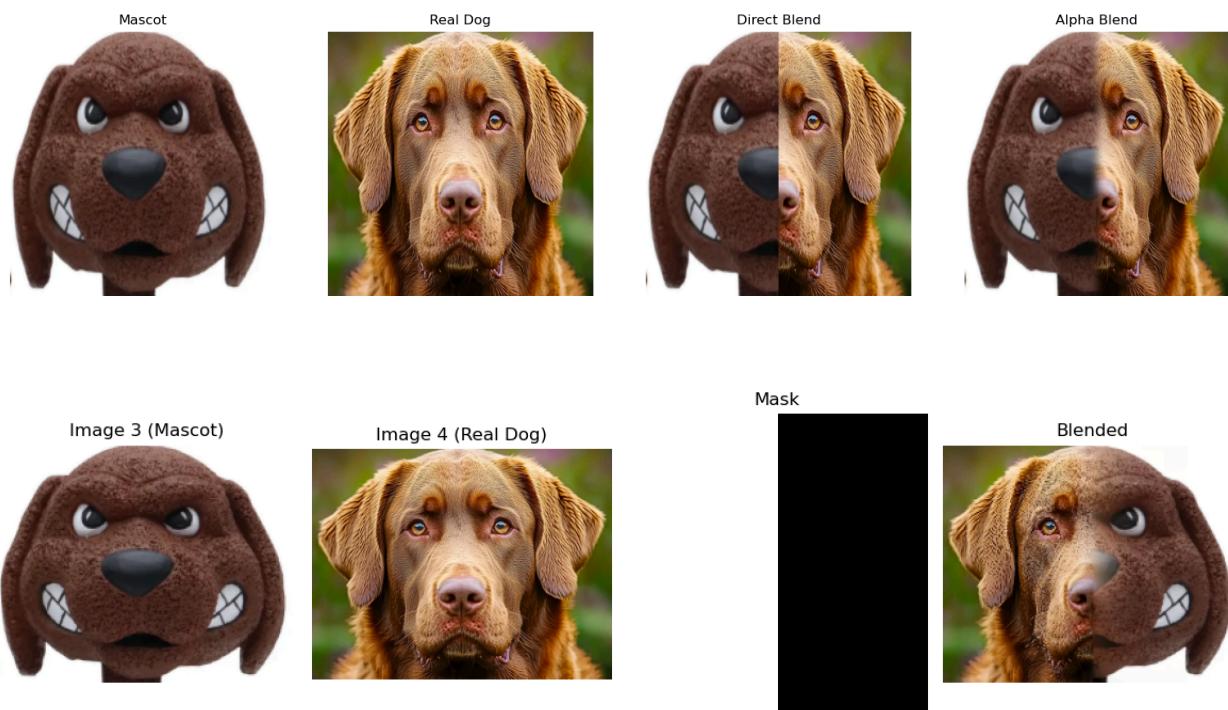
Task - 2: Code in the zip, file “question1_task2_GaussianLaplacianPyramids.py”



Task - 3: Code in the zip, file “question1_task3_MultiresolutionBlending.py”



Task - 4: Code in the zip, file “question1_task4_truegrit.py” and MultiBlend in “question1_task3_MultiresolutionBlending.py”



Explanation for results in Task 4:

Realistic blending is hard to achieve when combining two very different images, like a real Chesapeake Bay Retriever and the cartoon mascot True Grit, because their textures, lighting, colors, and structural details don't match naturally. The boundaries between the two are more noticeable, which makes the blend appear artificial. But for the “orapple” blend works better because an orange and an apple have similar shapes, colors, and surface textures, allowing their features to merge smoothly and produce a more visually convincing result.

2. Harris Corner Detection: Code for all the tasks is in the zip, file “question2.py”

```
● (cvenv) gravity-Aspire-A515-58MT% python question2.py
qt.qpa.plugin: Could not find the Qt platform plugin "wayland" in ""
Car: 101 keypoints detected
Elephant: 5192 keypoints detected
Sunflower: 4692 keypoints detected
Car - Harris: 101 keypoints, Alt: 61 keypoints
Elephant - Harris: 5192 keypoints, Alt: 2864 keypoints
Sunflower - Harris: 4692 keypoints, Alt: 3908 keypoints
Car - Provided: 101 keypoints, Custom: 375 keypoints
Elephant - Provided: 5192 keypoints, Custom: 307 keypoints
Sunflower - Provided: 4692 keypoints, Custom: 1162 keypoints
```

Car (101 keypoints)



Elephant (5192 keypoints)



Sunflower (4692 keypoints)



Task 6: Car

Harris: $\det(H) - 0.1 \cdot \text{tr}(H)^2$ (101 keypoints)



Nobel (1998): $\det(H) / (\text{tr}(H) + \epsilon)$ (61 keypoints)

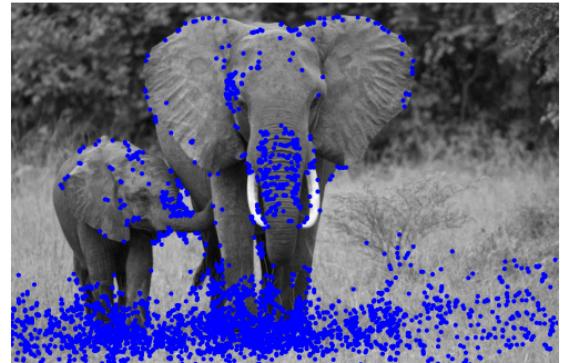


Task 6: Elephant

Harris: $\det(H) - 0.1 \cdot \text{tr}(H)^2$ (5192 keypoints)



Nobel (1998): $\det(H) / (\text{tr}(H) + \epsilon)$ (2864 keypoints)



Task 6: Sunflower

Harris: $\det(H) - 0.1 \cdot \text{tr}(H)^2$ (4692 keypoints)



Nobel (1998): $\det(H) / (\text{tr}(H) + \epsilon)$ (3908 keypoints)

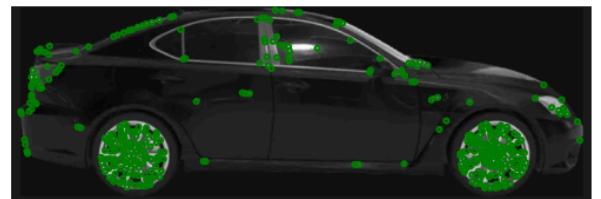


Task 7: Car

Provided Car (101 kp)



Custom Car (375 kp)

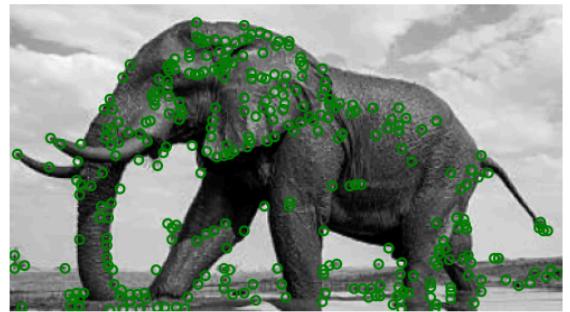


Task 7: Elephant

Provided Elephant (5192 kp)



Custom Elephant (307 kp)

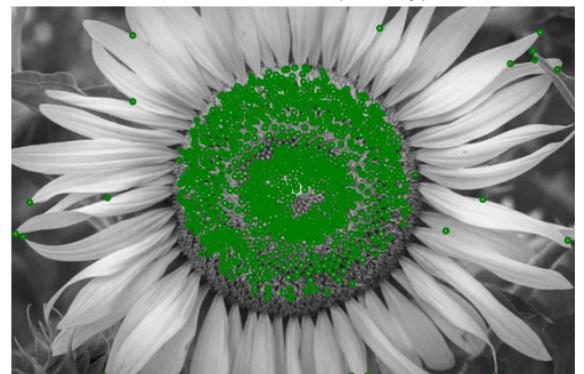


Task 7: Sunflower

Provided Sunflower (4692 kp)



Custom Sunflower (1162 kp)



Explanation for results in Task 6

The Nobel (1998) method consistently detects less number of keypoints than the Harris detector across all images. This shows that Nobel's formula $R = \det(M)/(\text{trace}(M)+\epsilon)$ is more selective in identifying corners. The ratio-based approach naturally filters out weaker features and edges better than Harris's $\det(H) - k \times \text{trace}(H)^2$ formula, which is more sensitive to the k parameter.

The largest difference is in the Elephant image (45% less), which may be because of it's complex textures and patterns. This shows that Nobel's method is better at rejecting textured regions that aren't true corners, while Harris is more liberal in detecting features. Both methods give similar results on the Sunflower (17% difference), suggesting they perform similarly on images with clear, well-defined corner structures. Overall, Nobel provides a cleaner, more focused set of strong corners, while Harris captures more features including weaker ones.

Comparison for results in Task 7

Both provided and custom images detect keypoints at similar structural features of each object - edges, corners, and texture boundaries are identified in similar regions. For example, the Car images show keypoints on windows, wheels, and body panel edges in both versions, while Elephant images detect features around ears, trunk, body contours and some in the grass. This demonstrates that the Harris detector consistently identifies the same types of geometric features regardless of the specific image.

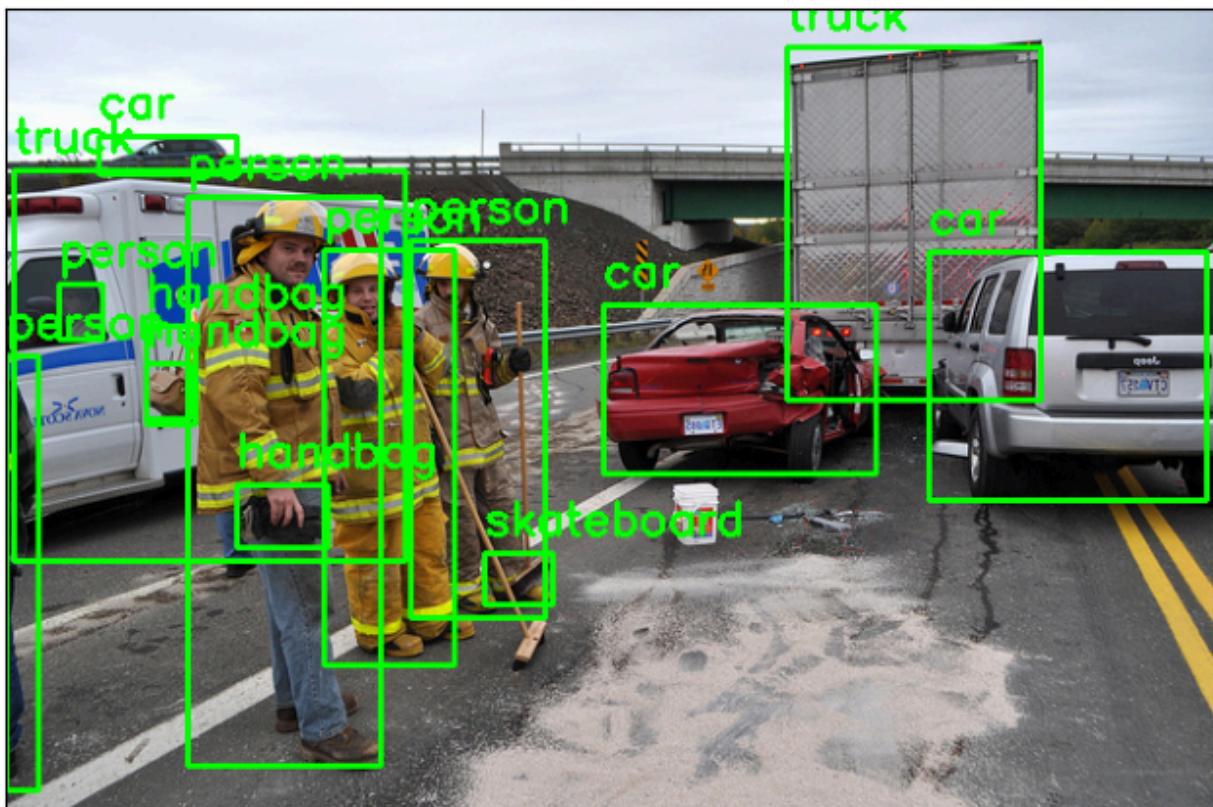
The custom images show significantly different keypoint counts compared to provided images - Car (375 vs 101), Elephant has similar counts (5104 vs 5192), and Sunflower has fewer keypoints (1162 vs 4692). These differences may be occurring due to differences in lighting conditions, viewing angles, image resolution and quality, and background complexity . Environmental factors like shadows, focus, and camera settings also changes local gradients, causing the same object to produce different numbers of detected corners even though the fundamental structural features remain similar.

3. Evaluation of Object Detection Models

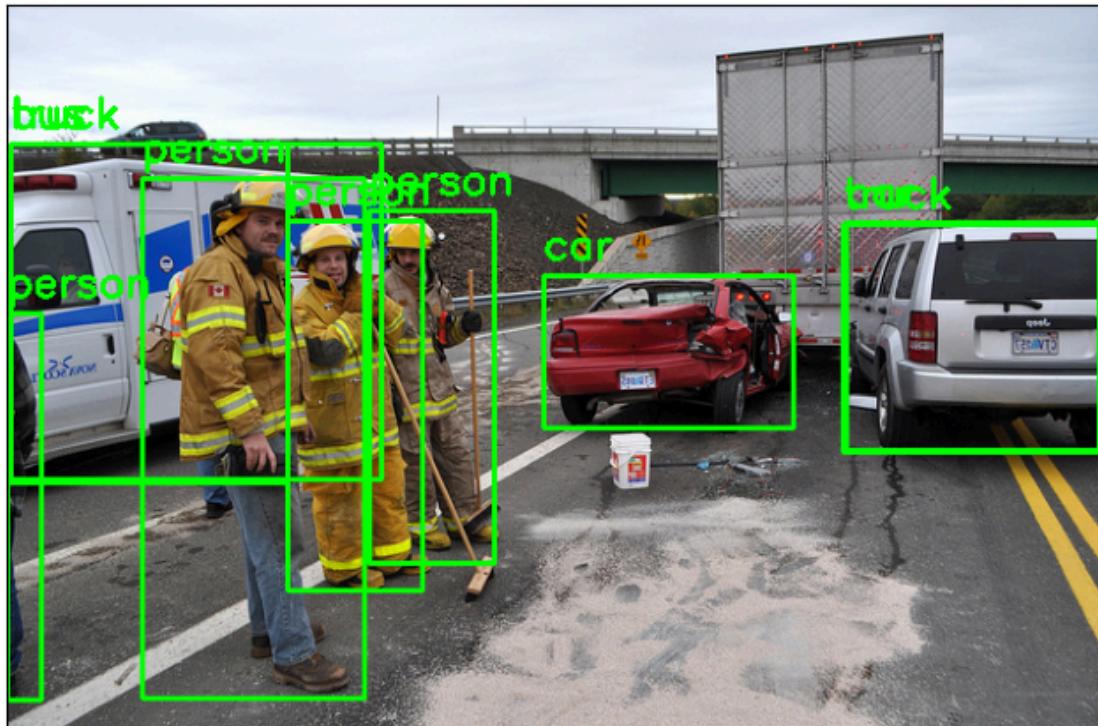
Code for all the tasks is in the .ipynb file

Results after completing Task 3

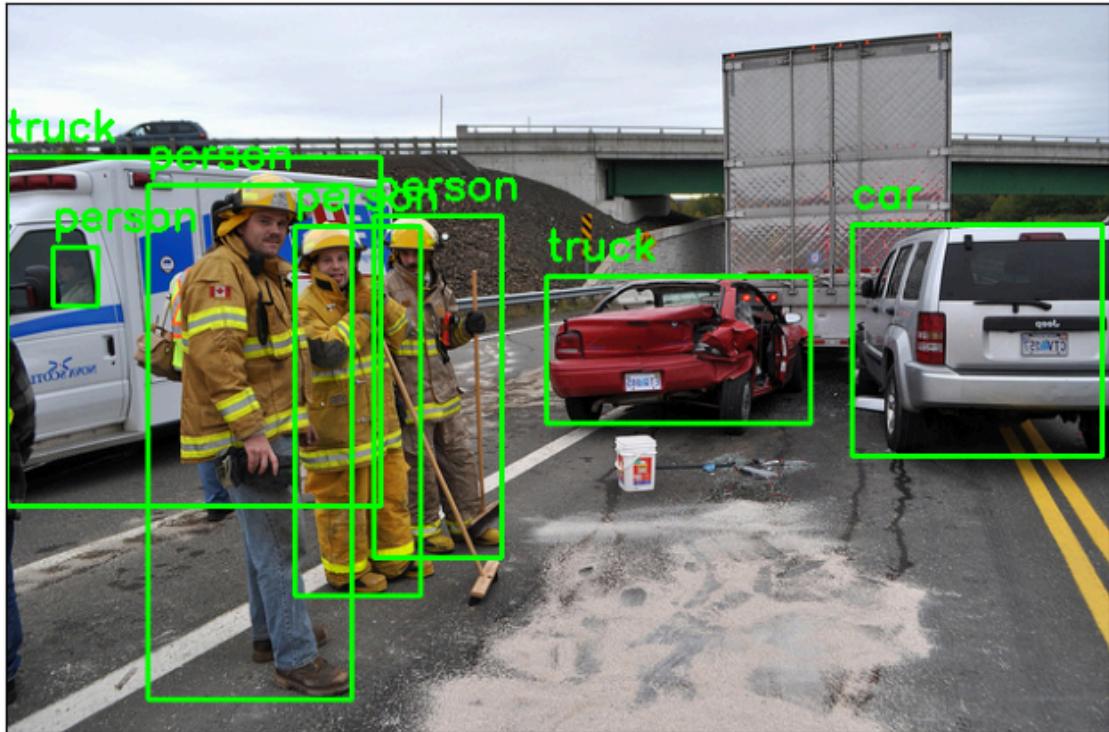
```
--2025-10-30 03:49:48-- https://www.tejasgokhale.com/images/vehicle.png
Resolving www.tejasgokhale.com (www.tejasgokhale.com)... 185.199.108.153,
185.199.109.153, 185.199.110.153, ...
Connecting to www.tejasgokhale.com (www.tejasgokhale.com)|185.199.108.153|:443...
HTTP request sent, awaiting response... 200 OK
Length: 581854 (568K) [image/png]
Saving to: 'vehicle.jpg'
vehicle.jpg    100%[=====] 568.22K --.-KB/s   in 0.007s
2025-10-30 03:49:48 (82.7 MB/s) - 'vehicle.jpg' saved [581854/581854]
Model 0 - FasterRCNN ResNet50 FPN:
```



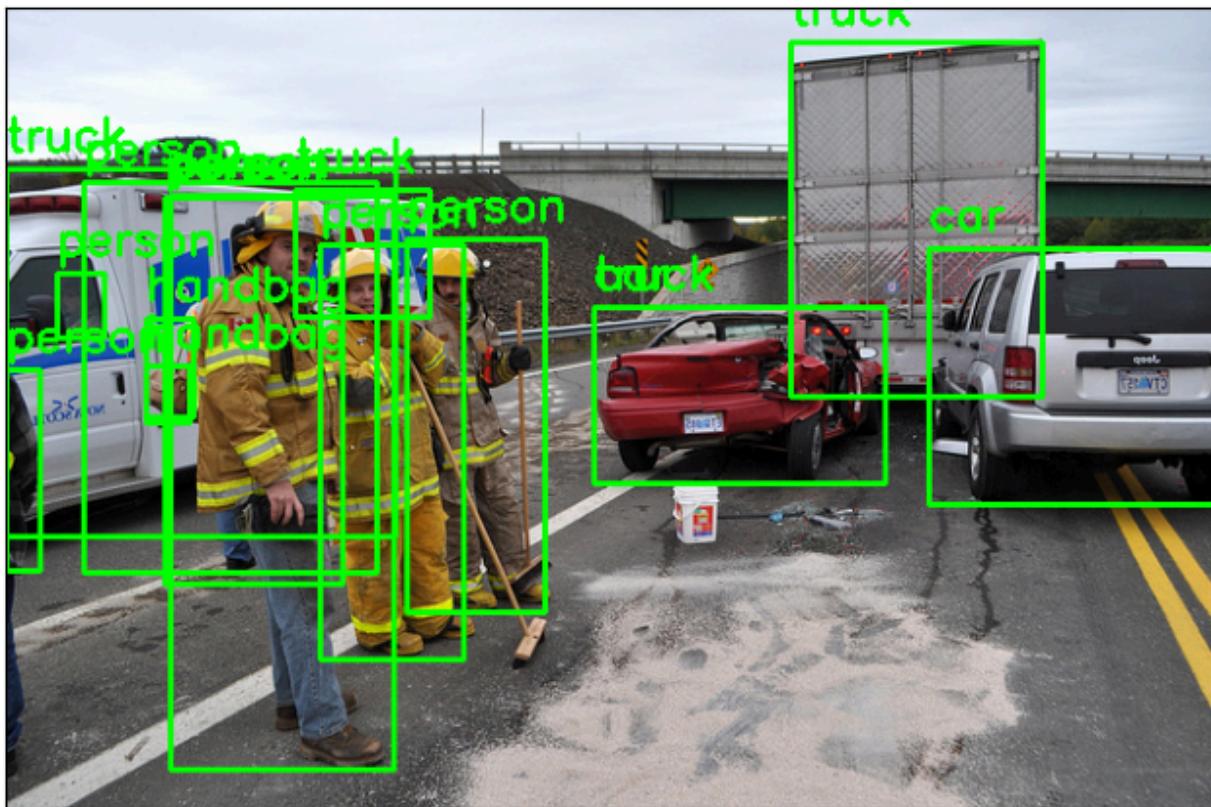
Model 1 - FasterRCNN MobileNet V3:



Model 2 - RetinaNet ResNet50 FPN:



Model 3 - FCOS ResNet50 FPN:



--2025-10-30 03:50:13--

https://pennaspillo.it/wp-content/uploads/2019/03/IMG_E4565.jpg

Resolving pennaspillo.it (pennaspillo.it)... 35.214.204.2

Connecting to pennaspillo.it (pennaspillo.it)|35.214.204.2|:443... connected.

HTTP request sent, awaiting response... 200 OK

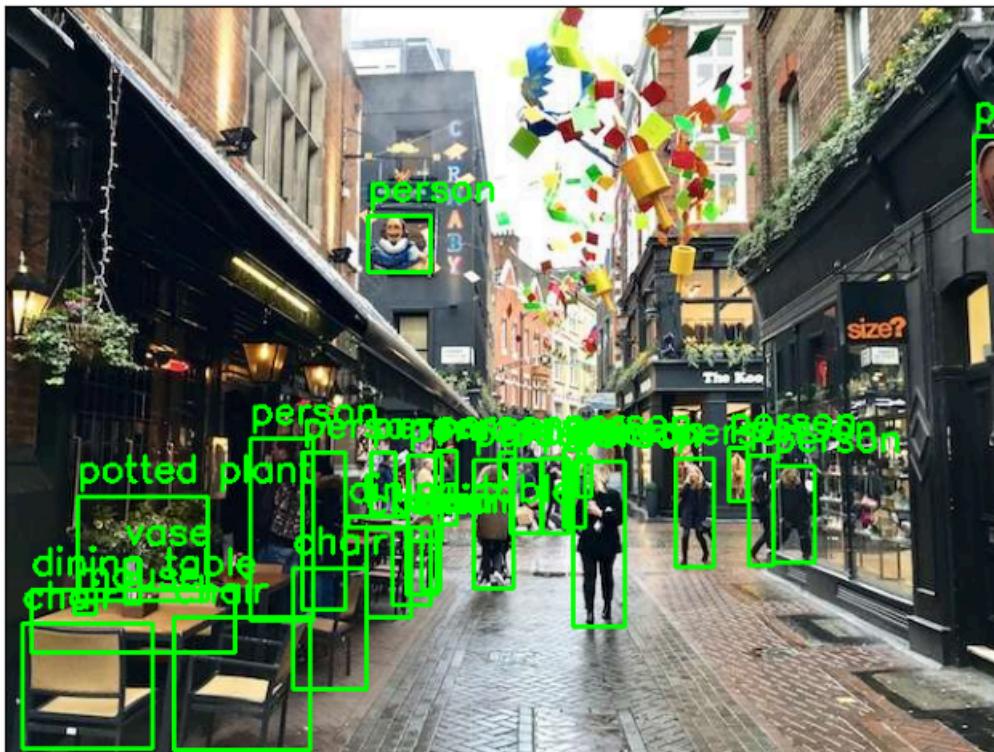
Length: 70260 (69K) [image/jpeg]

Saving to: 'IMG_E4565.jpg.1'

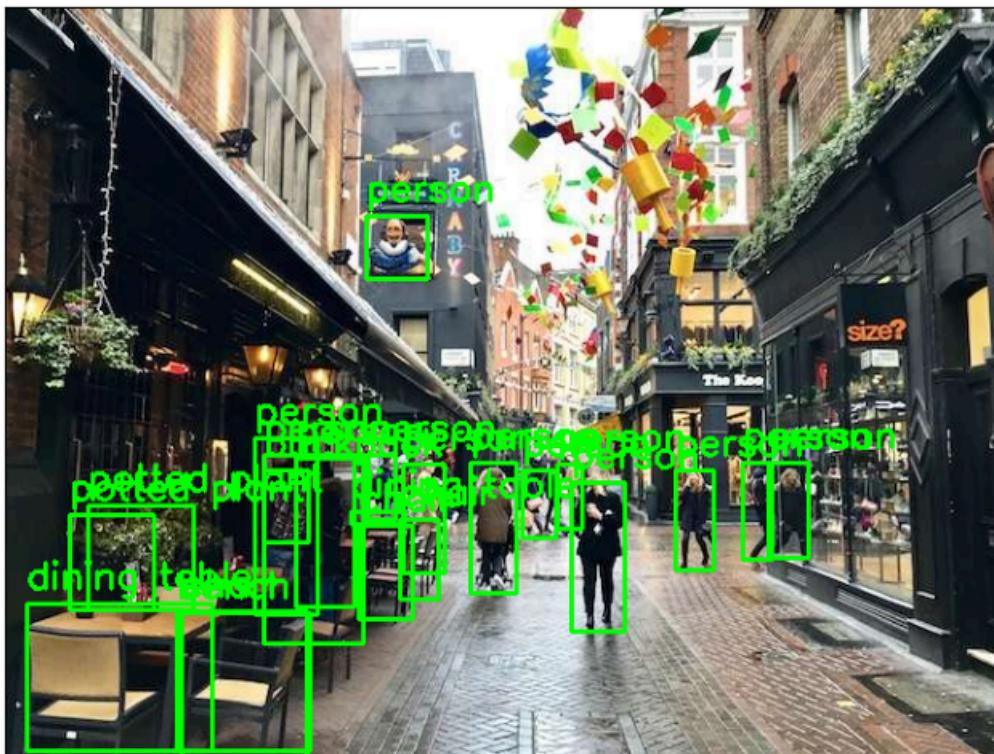
IMG_E4565.jpg.1 100%[=====>] 68.61K --.KB/s in 0.09s

2025-10-30 03:50:14 (786 KB/s) - 'IMG_E4565.jpg.1' saved [70260/70260]

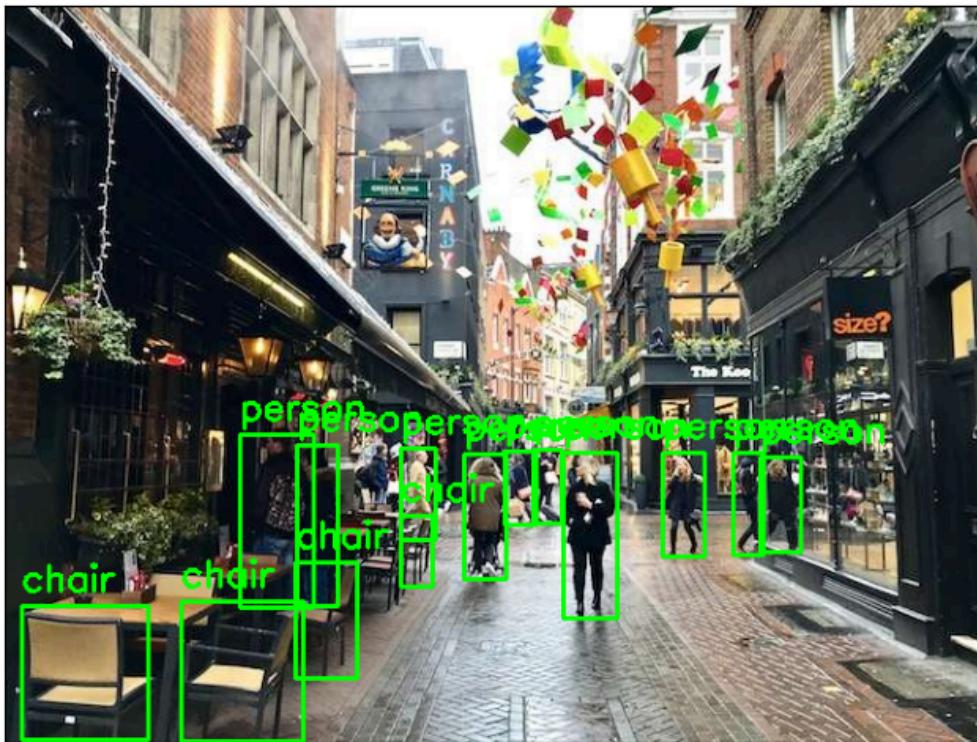
Model 0 - FasterRCNN ResNet50 FPN:



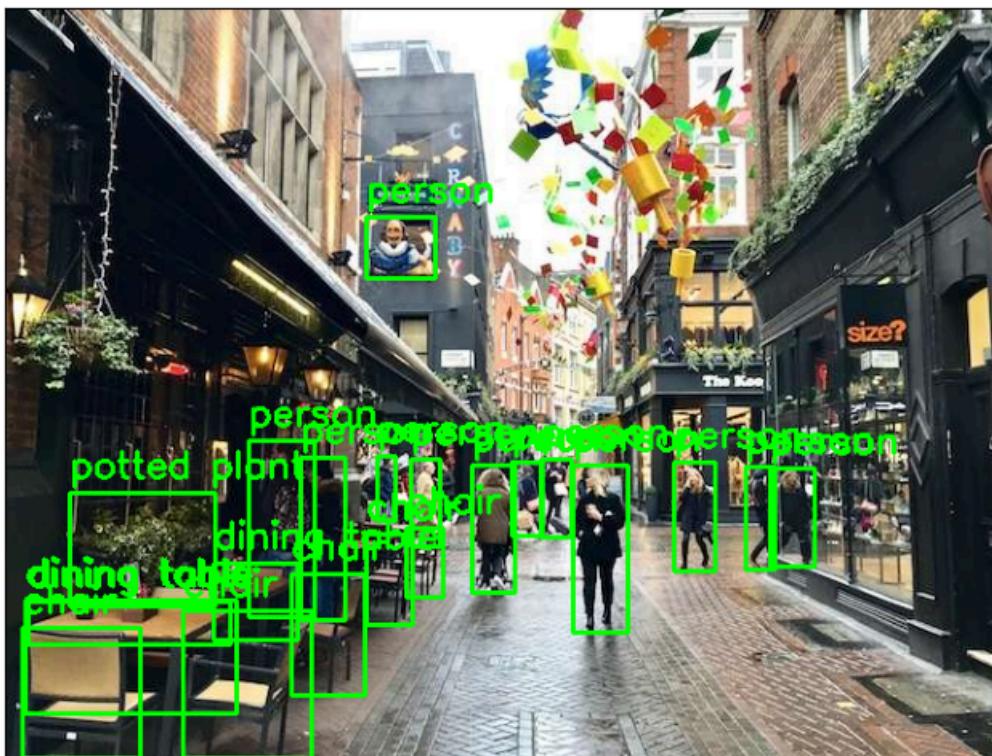
Model 1 - FasterRCNN MobileNet V3:



Model 2 - RetinaNet ResNet50 FPN:



Model 3 - FCOS ResNet50 FPN:



Task - 4: Evaluation using MS-COCO dataset

Evaluating FasterRCNN ResNet50...

FasterRCNN ResNet50 - meanIOU: 0.5221, Precision: 0.5285, Recall: 0.6701

Evaluating FasterRCNN MobileNet...

FasterRCNN MobileNet - meanIOU: 0.5777, Precision: 0.6163, Recall: 0.5464

Evaluating RetinaNet ResNet50...

RetinaNet ResNet50 - meanIOU: 0.7612, Precision: 0.8163, Recall: 0.4124

Evaluating FCOS ResNet50...

FCOS ResNet50 - meanIOU: 0.6226, Precision: 0.6707, Recall: 0.5670

Model	Mean IoU	Precision	Recall
FasterRCNN ResNet50	0.5221	0.5285	0.6701
FasterRCNN MobileNet	0.5777	0.6163	0.5464
RetinaNet ResNet50	0.7612	0.8163	0.4124
FCOS ResNet50	0.6226	0.6707	0.5670
