# HackLab Solution

# Report

We were given two tasks. In the first task, we were asked to implement person detection, and in the second task, we were asked to implement person detection using tiling with a size of 4x3, resulting in 12 tiles.

The following is an overview of the approach we used for these tasks. We used YOLOv5 as the model for person detection.

### Without Tiling:

* We load the pre-trained YOLOv5 model from a specified file and set it to evaluation mode.
* We then define the path to a directory containing images and loop over each image, using the model to detect objects in each image.
* We extract the bounding boxes and class labels from the model's results and filter the results to only include "person" class labels.
* Finally, we draw the bounding boxes on the image and save the image with the bounding boxes overlaid.

### With Tiling:

* We follow a similar process, but first compute the size of each tile of the image and loop over each tile to apply the person detection model.
* We then crop each tile from the image and use the model to detect objects in each tile.
* We extract the bounding boxes and class labels from the model's results and filter the results to only include "person" class labels.
* Finally, we draw the bounding boxes on each tile and save each tile with the bounding boxes overlaid.

Results:

### Without Tiling:



### With tiling: without tiling of the same figure:





As we can see in the above 2 phots , the person detection is comparatively better with tiling method than the non tiling method. But as far as the computation is concerned with tiling takes more time than non tiling method. Accuracy wise tiling method is more accurate.