

## Person Detection & Waypoint Algorithm

The person detection algorithm we implemented identifies people in the camera's FOV and tracks them (using a Euclidean distance heuristic) frame-to-frame. It considers targets "locked" if detected in the FOV for 30 consecutive frames. Upon identifying a set of locked targets, the world coordinates of the targets get turned into waypoints for the robot to travel to along a multi-segment trajectory.

Let tracked\_persons be an empty list.

Let targets\_locked equal false.

While targets\_locked is false:

- Get depth frame from the camera
- Get RGB frame from the camera
- Run MobileNet image classifier on RGB frame:
  - Filter to person objects with confidence > 0.7
  - Denote bounding boxes around all persons detected
- If persons detected in frame:
  - Merge depth frame and person bounding boxes
  - Compute  $R^3$  world coordinates of bounding boxes using depth + camera intrinsics
  - Check if detected persons are tracked persons:
    - For tracked\_person in tracked\_persons:
      - Is the Euclidean distance of tracked\_person within 0.2m of the detected person?
        - If yes, increment tracked\_person's match count
        - Recursively update tracked\_persons position in the world using moving average position
      - If no, add the person to tracked persons
- For tracked\_persons:
  - If all have > 30 match count: Targets Locked, set targets\_locked = True, \*break\*
  - If not all > 30 match count: Targets Not Locked, \*continue\*

## Motion Control Algorithm

We implemented a trapezoidal trajectory for motion control using maximum acceleration and velocity values to ensure smooth motion and to avoid spilling any payload contents.

```
float decel_disp = fabs(target_disp) - (prev_vel * prev_vel) / (2 * max_acc);
if (fabs(curr_disp) < decel_disp) {
    target_vel = prev_vel + dir * max_acc * dt_s;
    target_vel = Saturate(target_vel, max_vel);
} else if (fabs(prev_vel) > max_acc * dt_s) {
    target_vel = prev_vel - dir * max_acc * dt_s;
} else {
    target_vel = 0;
}
```

- P\_new =
- V\_max =
- A\_max =