main.py

- main()
    - Entry point of program
    - Starts opencv, the main loop, then stops opencv

image_functions.py

- start_human_detection_loop()
    - Input: height, angle, fov, needAlert, needScreen
    - Output: none
    - Description: Analyzes the video feed one frame at a time, detect all people in each frame and display all relevant information. Also plays a sound and saves a screenshot on every frame where people violate social distancing. Needs the camera info to calculate distances correctly
- Display functions
    - display_blob()
        - Input: a list of objects
        - Output: none
        - Description: displays all objects provided
    - display_number_of_people()
        - Input: number of people, frame
        - Output: none
        - Description: prints the number of people onto the frame
    - draw_labels()
        - Input: boxes, confs, colors, class_ids, classes, img
        - Output: counter
        - Description: counts the number of people and draws a box around each one
    - draw_line()
        - Input: frame, x1, y1, x2, y2, color
        - Output: none
        - Description: draw a line of the provided color from point 1 to point 2 on the frame
    - draw_text()
        - Input: frame, text, x, y, color
        - Output: none
        - Description: prints text of the chosen color with the provided x and y coordinates as the bottom left on the frame
    - print_on_feet()
        - Input: list of boxes representing people, all box data, classes, img, height, angle, vertical FOV
        - Output: a list of distances
        - Description: using the provided boxes and data, get the distance between the camera and each person and return the list
- Detect people and certain parts of people
    - detect_objects()

        - Input: img, net, outputLayers
        - Output: blobs, output
        - Description: Uses the yolo software/algorithm the generate a list of objects
    - get_box_dimensions()
        - Input: outputs, height, width
        - Output: boxes, confs, class_ids
        - Description: based on the size of the frame and the list of objects, generate a list of boxes
    - get_feet_pos()
        - Input: a list of boxes representing people detected
        - Output: a list of feet positions
        - Descriptions: produces a list of feet positions, assuming each box in the list is a person
- Initialization
    - # opencv setup and end
    - init_opencv()
        - Input: none
        - Output: none
        - Description: start up the window thread
    - stop_opencv()
        - Input: none
        - Output: none
        - Description: closes all window threads and ends
    - load_yolo()
        - Input: none
        - Output: none
        - Description: Loads up the yolo software/algorithm
        - Loads in the yolo weights, cfg, and all classes from coco.names files
        - Returns all data from yolo and coco files
    - set_cap_height_and_width()
        - Input: cap, height, width
        - Output: none
        - Description: sets up the capture height and width
- Utility
    - resize_frame()
        - Input: frame, width, height
        - Output: none
        - Description: resizes the frame to the new width and height
    - start_videocapture()
        - Input: source, video file name
        - Output: cap
        - Description: creates a video capture object that allows for reading from a source. Currently supported video types are webcams and video files. If

source is video file, the video capture object reads from the video file provided

  - create_dir()
    - Input: directory name
    - Output: none
    - Description: creates a directory with the name provided
- dist_on_foot()
  - Input: distance, frame, coordinate
  - Output: none
  - Description: prints the distance from the camera to a detected foot. Uses the provided coordinate to as the bottom left corner of the text representative of the distance
- too_close_handler()
  - Input: violation bool, audio alert bool, screenshot bool, screenshot directory, frame, screenshot output video writer, screenshot number, screenshot filename, video writer fourcc (data)
  - Output: screenshot output video writer, screenshot number
  - Description: if there is a violation, optionally play an alert based on audio alert, optionally save the frame to a video, creating a new video writer to a new file as needed. If there are no violations, end the video writer and increment the screenshot number.

distance_functions.py

- adjust_angle()
    - Input: angle, fov, vertical_position
    - Output: final_angle
    - Description: calculates an adjusted angle based on the provided angle, FOV, and vertical position of the camera
- find_distance()
    - Input: height, angle, fov, vertical_position_of_persons_feet
    - Output: distance
    - Description: uses an adjusted angle from adjust_angle and the vertical position of a person’s feet to calculate a person’s distance from the camera
- find_distances_between_positions()
    - Input: list vert_positions
    - Output: list distances, list lines
    - Description: given a list of vertical positions, calculate a list of distances and lines between them