GENERALIZING THE PERCEPTION MODEL:

Submitted by Keerthana Kannan (kkannan@buffalo.edu)

Pre-process the given input depth array to reduce noise and smooth the array for later process.

A **2-stage head detection process** can be used to locate the people, that is, to first explore the boundary information embedded in the depth array to locate the candidate regions that may indicate the appearance of people. 2D chamfer distance matching algorithm is a good fit for this. It scans across the whole image and gives the possible regions that may contain people.

Canny edge detector can be considered to find all edges in the depth array. To reduce calculation and reduce the disturbance from the surrounding irregular objects, eliminate all the edges whose sizes are smaller than a certain threshold. A binary head template can be taken and match the template to the resulted edge image. If the distance value lies below a certain threshold, the target object is considered detected at this place, which means that a head like object is found here. To make the algorithm invariant to scale, generate an image pyramid with the original image at the bottom; each image is subsampled to generate the next image at the higher level.

Then, examine each of these regions using a **3D head model**, which utilizes the relational depth information of the array for verification. Further extract the parameters of the head from the depth array and use the parameter to build a 3D head model. Match the 3D model against all the detected regions to make a final estimation.

Region growing algorithm to extract the whole body contours from the processed depth array. It is assumed that the depth values on the surface of a human object are continuous and vary only within a specific range. The algorithm starts with a seed location, which is the centroid of the region detected by 3-D model fitting. The rule for growing a region is based on the similarity between the region and its neighboring pixels.

A **segmentation scheme** can be used to segment the human from his/her surroundings and extract the whole contours of the figure based on the detection point.

This way the position of human can be determined in any image and the distance of the human from the sides of the images can be calculated to determine the direction of motion for the robot.

Also, the **sample output of the find_clearance.py**: