MOM #5/Main Report3

Name: Priyankar Kumar
To: Mrs. Srishti

Focus of the meeting: The Cleaning Module

Present Approach (Removing duplicates):

- Extract information about all the lines (their start and end points and id) and find and store other information about the lines such as slopes and intercepts. Stored as a list of dictionaries.
- Now from all this information, we need to get a list of duplicate lines that are to be removed from the above list of dictionaries of line information.
- We know that any two line segments will lie along the same line if they have the same slope and intercepts.
- We group lines according to the same slopes and intercepts by storing their ids in a list of lists each smaller list containing ids of lines with same slopes and intercepts.
- Now among the line ids in these groups we find the largest line's id. We delete it from the smaller list of line ids.
- At the end,we have a list containing ids of lines **that have to be deleted from the main list of dictionaries containig actual line information.**
- So we delete the smaller lines using the ids from the list of line ids.

Issue : It's removing some non duplicate lines and is not very efficient.

Sir's Approach:

- First find all distances betweeen the walls and store in a list with ids of the wall (dxf) .
- Find smallest distance among all the wall distances
- To find the threshold for finding the overlapping duplicates, we use half the distance between the walls.
- First classify lines according to their slopes and intercepts.
- If two lines have different slopes and intercepts-> no issues
- same slope different intercepts-> parallel lines->no issues
- have same slopes ad intercepts-> lie on same line. So here we try using the distance between the lines as a metric to distinguish and remove the smaller line, retaining the longer line,

Approach for removing the dangles

- Find the threshold (maybe 1/10th of wall width)
- And remove the extension.

Speeding up the algo

- Divide the area into many regions and use the algorithm to reduce number of comparisons.

Other discussions

- **Computer Vision Approaches: Snake Algorithm (And why it can't be used), Image Segmentation**

Future Work

- Independently test the Polygonisation code on manually cleaned data to check if it works correctly.

- Implementing function to find distance between parallel lines
- Getting a DXF viewer
- Brushing up the GUI
- Integration work completion