COMS 4735-Assignment 3

Stephen Pasch

Step 1

To represent the buildings, I first parsed the PMG file into a NumPy array of intensity values between 0-255. My buildingCoordinates() method took that intensity array and created a dictionary of {IntensityValue: [list of associated coordinates]}. This gave me a solid base to iterate over all the pixels of each building. The pixel area was calculated as the len(pixel list) and stored in a new area dictionary represented {IntensityValue: Area in px}. To get the center of mass of each building, I iterated over its pixels and averaged their x and y coordinates. This result was stored in a center of mass dictionary represented {IntensityValue : (x,y of the COM)}. The Minimum bounding rectangle for each building was calculated by iterating over the pixels and finding the leftmost x, rightmost x, uppermost y, and lowermost y pixel values and storing these in a dictionary that matched the intensity value to a list of the upper left coordinate and lower right coordinate of the MBR. MBR_Dictionary = {Intesity:[(leftmost, uppermost),(rightmost, lowermost)]}. The Diagonal of the minimum bounding rectangle was calculated from the Euclidean distance of the two bounding description coordinates. To get a list of what Bounding rectangles intersected with one another I iterated over all the buildings twice. The outer loop iterated over each building, while the inner loop iterated through all the other buildings. Then compared the corners of the other bounding box to see if it would fall into the volume of the original one. The iteration carried out until each building had a list of all the other buildings its MBR intersected with. This information was then stored in a dictionary described, {IntesityValue : [list of intensities that intersect]}

Step 2

“What” describing shape

- To describe smallest, I chose the building with the smallest area. This was alma mater.
- To describe largest, I chose the building with the largest area. But I did not count the hole in Mudd where the courtyard is. That does not contribute to its area. This the hamilton-john jay building group was considered my largest.
- To describe middle-range-sized buildings, I took an average of the smallest and largest size and set a mean value. Then I created an upper threshold value halfway from the mean to the max and a lower threshold halfway between the mean of the min. If the size was in between the two thresholds(closer in size to mean value than extremes), then it was considered middle-sized.
- The large items were any building above the large threshold previously calculated, excluding the max.
- The small items were any building below the smaller threshold, excluding the min.

- For the Aspect ratio, I defined anything slimmer than 2:3 to be narrow, anything wider than 3:2 to be wide, and in between to be a medium width. These values, I tested around until they captured just the buildings that I thought to be narrow verses just rectangles.
- In special medium-width circumstances, when an object was between 8.5:10 and 10:8.5 I considered this square. I opted not to proceed with a standard 1:1 because people I asked thought CEPSR was square. This gave square some wiggle room for imperfection that is not noticeable to the eye.
- For Geometry the square must maintain an aspect ratio of a square(defined above), but it does not have to maintain a full square volume. I gave the areathreshold 80% of the MBR because this captured buildings like low library that are "plus-shaped," which I thought should be represented as a square for simplicity.
- Rectangles had a similar approach but had to cover 85% of the area within the MBR. This threshold came from experimentation to capture buildings like Fayweather with the "plus shape" corners.
- For the I-Shaped buildings, I defined them as having two gaps on either side of their major axis. The gap size that captured all the I shape buildings that I discovered was ⅛ of the minor axis and ⅓ of the major axis. If a blank area of these dimensions was found centered along the edges of the major axis then it is considered I-shaped.
- For the C-Shaped buildings I followed the same suit but it could only contain one gap along an edge of a major axis while the other is solid. I ensured this by taking a boolean of each gap's existence and XOR'ing it.
- For the L-Shaped Buildings, I took an area of each corner of the MBR. If it were the case that only one of the four corners was empty, then it could be considered L-shaped. This method worked well. From an interview with friends, they considered Havermeyer and Schermerhorn to be L-shaped, so I tweaked the corner area side length to ⅛ of the building width dimension.
- To determine if a shape was asymmetric, I counted the pixels of the top of the image and compared them to the count of the bottom half of the image. If they fell within a 50 px range(any higher and buildings like Mudd became symmetric) I considered it to be relatively symmetric, I did the same with the left and right sides. If both Top/Bottom and Left/Right were not symmetrical, I deemed the building asymmetric.
- Confusions: I kept of the three "what" factors in a dictionary with the building number serving a the key. I compared each against the others for duplicates. This was stored in a similarity Dictionary which I output as a list of possible confusing building descriptions.
- Minimization: To minimize condition I simplified the description of the extremes "Largest" and "Smalllest" I did not implement any other short hands.

Step 3

"Where" describes absolute space

- For verticality, I used the Center of Mass(COM) method I created to keep track of building positions.(Upper is closer to 0 on the Y.)
- Uppermost was the building that had the lowest COM y-coordinate.
- Lowermost was the building with the highest COM y-coordinate

- Like Middle size, middle-hieght was determined through a mean between highest and lowest and the distance of other buildings to that average as opposed to the extrema.
- Upper fell above below the upper threshold and excluded the min.
- Lower fell above the lower threshold and excluded max.
- I based my orientation off of rectangular principles. If an object had an aspect ratio close to a square I considered it non-oriented.(L-shape, square, C-shape, some asymmetric) If its MBR was rectangular with a vertical major axis, I considered it vertically oriented. I did the same with the case of a horizontal major axis being considered horizontally orientated.
- Confusions: I kept of the three “where” factors in a dictionary, with the building number serving a the key. I compared each against the others for duplicates. This was stored in a similarity Dictionary which I output as a list of possible confusing building descriptions.
- Minimization: To minimize the descriptions of the extremes I reduced them to “uppermost” and “lowermost” I did not implement any other short hands.