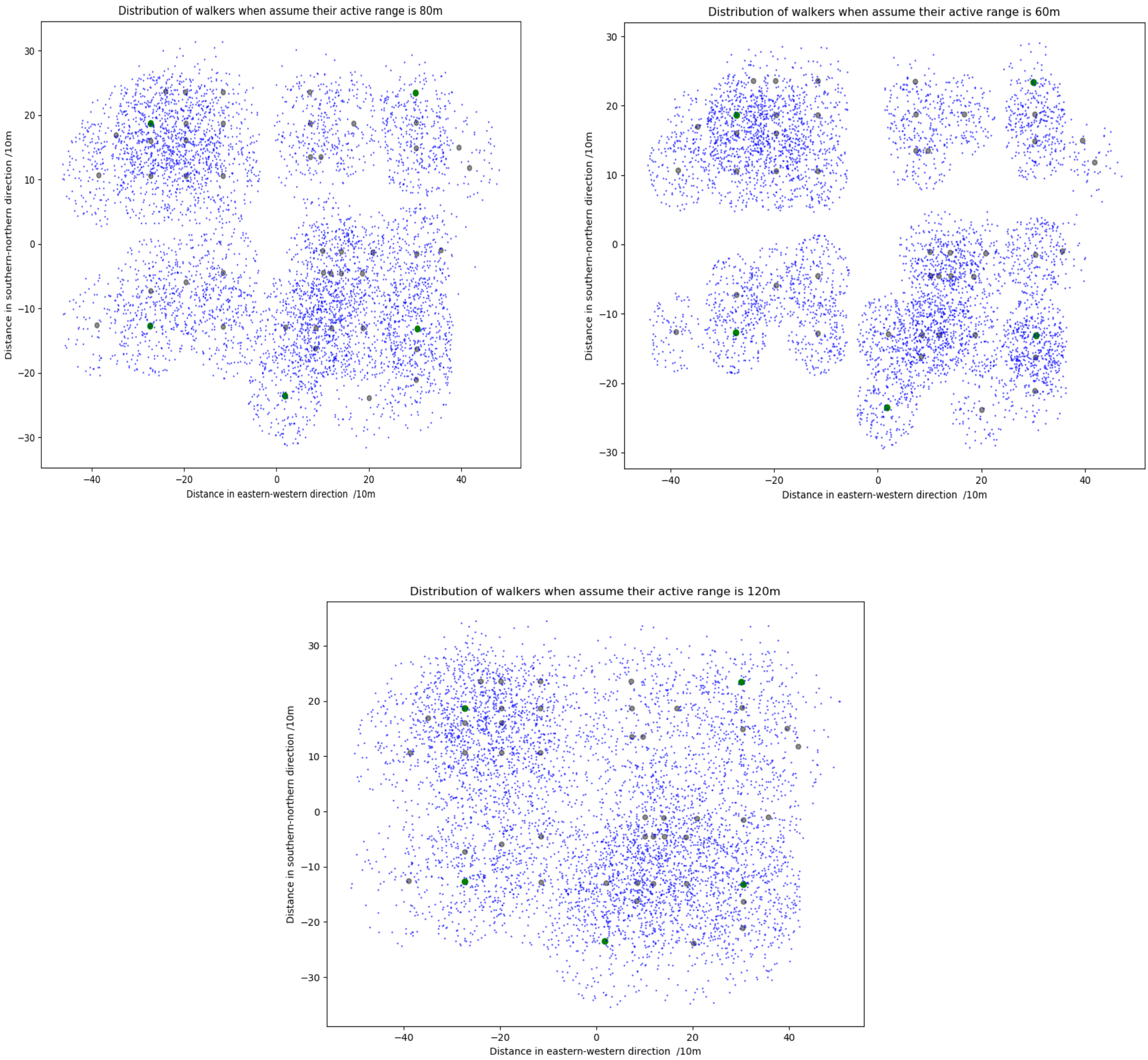
Here are the diagrams and results plotted by Python that I sent to the manager of the park in my community.



Figure(1)

All simulated points in graphs represent the possible positions of a walker around observational points. Assumed active range of walkers around observational points is 60m, 80m, and 120m, from the leftmost diagram to the rightmost diagram, respectively.

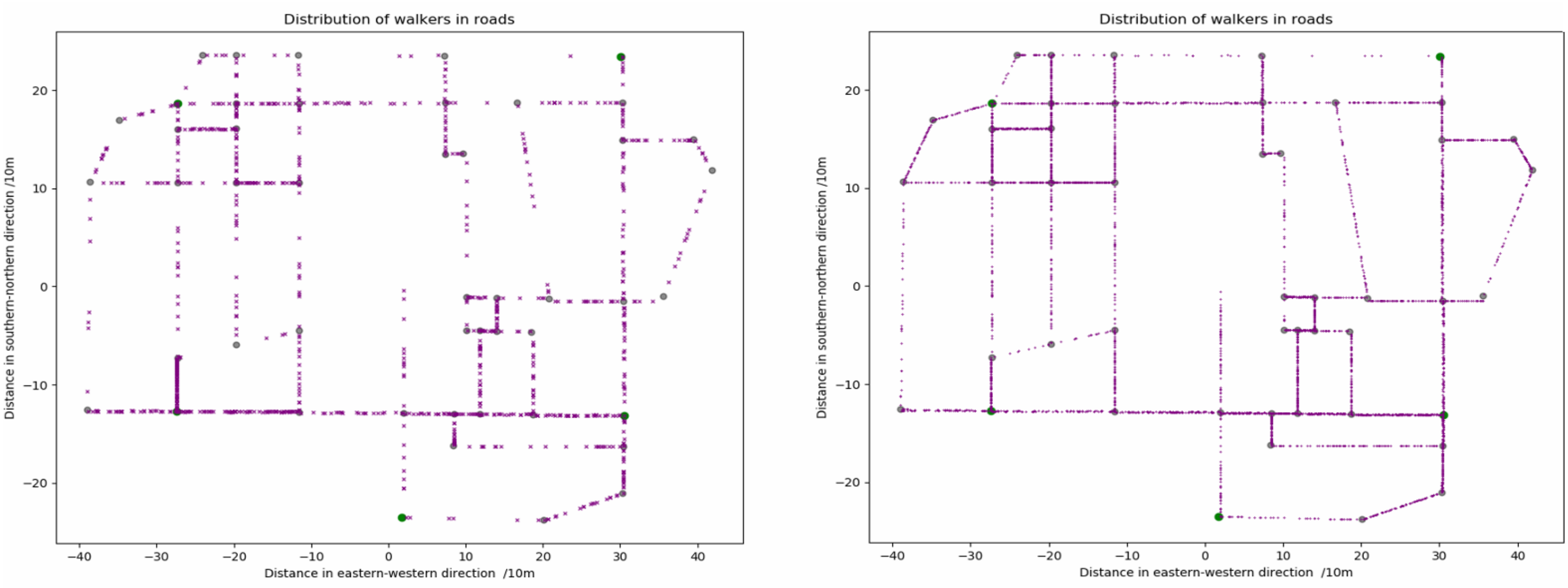


Figure (2)

Each point in graphs represents the simulated possible positions of a walker in roads. In the right graph, the exact number of points corresponds to the number in original data, and size of points=1. In order to denote points more clearly, in the left graph, it only includes fifth of the number of original data with larger size of points, size of points=3.

Initially, with original data I gathered, I denoted the number of walkers who passed observation points within a fixed period on each intersection of roads (observation points) in a map of the park, aiming to intuitively find out which parts of park are crowded. However, it didn't offer me strong conclusions. In this case, I had to utilize a stronger way to fully take advantage of my data. The scientific visualization of Python l learned in CIS program came into my mind. I had two ways to represent the pedestrian volume of observation points to test whether a region in the park was crowded or not.

In my first approach, a point in a circle whose center lies in observation points with a varied radius can represent the possible appearance of a walker. Then I generated the number of random points that equaled to the number of people passed observation points in my data set to plot in the circle. In this case, we know whether an area is crowded or not by visually judging how dense points are in an area. (shown in Figure1)

My second way is to plot the possible appearance of walkers directly on roads instead of in a circle. Suppose that there are five observation points named A, B, C, D, and E, and A, B, C are collinear in eastern-western direction while C, D, E are collinear in southern-northern direction. As all observation points are the intersection of roads in the park, most observation points, like point C described above, lie exactly in the intersection of two segments (AB and DE). Then I plotted and distributed the number of walkers passed the point C evenly in AB and DE. The number of random points on AB or DE, simulating the possible appearance of walkers on each road, would be half the number of walkers passed the point C because the volume needed to distribute evenly in the eastern-western direction and southern-northern direction Hence, after plotting all observation points by following the rules above, we know which roads are crowded and which roads are deserted.(shown in Figure 2)

The whole project and source code I used have been uploaded to Github in order to share and discuss with others.