2:06 PM (2 hours)

Using processing to write an animation sketch

Processing CSV file reading, ordered chronologically

Table table;

int num\_villages = 80;

int num\_years = 21;

table = loadTable("Hardoi\_processed.csv", "header");

println(table.getRowCount() + " total rows in table");

for(int i = 0; i < num\_years\*12; i++){

for(int v = 0; v < num\_villages; v++){

TableRow printRow = table.getRow(v\*num\_years\*12 + i);

println("Village: " + printRow.getString("Village\_ID") + " year: " + printRow.getInt("Year") + " month: " + printRow.getInt("Month") + " vis: " + printRow.getFloat("Vis") );

}

}

3:13 PM

Added color interpolation to show varying brightnesses.

Seems very dark. I might add a curve to help bring up the brightnesses

Added a sqrt curve.

1. Table table;
2. **int** num\_villages = 80;
3. **int** num\_years = 21;
4. **int** v = 0;
5. **int** y = 1993;
6. **int** m = 1;
7. Float[] vis = **new** Float[num\_villages];
8. Float[] next\_vis = **new** Float[num\_villages];
9. **int** counter = 20;
10. color dark = color(0, 0, 0);
11. color light = color(255, 255, 255);

14. TableRow getEntry(**int** vil, **int** year, **int** month) {
15. **return** table.getRow(vil\*num\_years\*12 + (year-1993)\*12 + month -1);
16. }
18. Float curve\_vis(Float vis){
19. **return** sqrt(vis/63.0)\*63.0;
20. }
22. **void** setup() {
23. table = loadTable("Hardoi\_processed.csv", "header");
24. println(table.getRowCount() + " total rows in table");
25. frameRate(60);
26. size(520, 420);
27. }
29. **void** draw() {
30. println("Month: " + m + ", Year: " + y);
31. //If counter reaches 20, update.
32. **if** (counter>=20) {
33. counter = 0;
34. //Update all current vis
35. **if**(y < 2014){
36. **for** (**int** v = 0; v < num\_villages; v++) {
37. vis[v] = curve\_vis(getEntry(v, y, m).getFloat("Vis"));
38. }
39. **if** (m==12 && y<=2013) {
40. m = 1;
41. y++;
42. } **else** **if** (y<=2013) {
43. m++;
44. }
45. //Update all  next vis
46. **for** (**int** v = 0; v < num\_villages; v++) {
47. next\_vis[v] = curve\_vis(getEntry(v, y, m).getFloat("Vis"));
48. }
49. }
50. }
51. //Draw for each village
52. **for** (**int** v = num\_villages-1 ; v >= 0; v--) {
53. color interFrom = lerpColor(dark, light, vis[v]/63.0);
54. color interTo = lerpColor(dark, light, next\_vis[v]/63.0);
55. color interp = lerpColor(interFrom, interTo, counter/20.0);
56. fill(interp);
57. pushMatrix();
58. translate(v/8\*50, v%8\*50);
59. rect(10, 10, 50, 50);
60. popMatrix();
61. }
62. counter++;
63. }

Finished making a grid based visualization.

Next steps: Map it to a geographical representation of Hardoi

Finished recording

4:01 PM