Started 4 pm

Commenting things out to find out which operation is causing the draw update to take ~100ms. Removing the stroke seems to reduce each draw update by ~90%. However, the result is not desirable

Finished 5:30 pm

Started 6 pm

Still looking for the bug.

Forgot that in processing (java) objects are always passed by reference. Previous/current vis was not independent of next vis, so fading was broken.

Everything’s fixed. Experimented with dark blue/ yellow colors, but they didn’t look great.

Final code:

1. PShape country;
2. Table table;
3. FloatDict cur\_vis = **new** FloatDict();
4. FloatDict next\_vis = **new** FloatDict();
5. HashMap<String, PShape> district\_map = **new** HashMap<String, PShape>();
6. TableRow new\_row;
7. **int** table\_row = 0;
8. **int** table\_size;
9. **int** year = 1993;
10. **int** month = 1;
11. color dark = color(0, 0, 0);
12. color light = color(255, 255, 255);
13. **int** time\_update = 0;
14. **int** draw\_update = 0;
15. **static** **final** **float** update\_interval = 300;
17. **void** setup(){
18. //Set size
19. size(950, 1000);
20. //Load country
21. country = loadShape ("Data/India.svg");
22. country.disableStyle();
23. println("India svg successfully loaded");
24. println(country.getChildCount() + " regions found");
25. //Load data
26. table = loadTable("query\_sat-stdist-month.csv", "header");
27. println("Vis data successfully loaded");
28. table\_size = table.getRowCount();
29. println(table\_size + " total rows in  data");
30. //Initialize new row to be checked
31. new\_row = table.getRow(table\_row);
32. println("Data starts at " + new\_row.getInt("month") + "/" + new\_row.getInt("year"));
34. //Drawing variables
35. frameRate(60);
36. fill(0);
37. stroke(120);
38. strokeWeight(1);
39. shape(country, -50, -200);
40. }
42. Float curve\_vis(Float vis){
43. **return** sqrt(vis/63.0)\*63.0;
44. }
46. **void** drawDistrict(String loc, color c){
47. //If this district hasn't been loaded before
48. **if**(!district\_map.containsKey(loc)){
49. println("Loading new district shape for " + loc);
50. String[] location = splitTokens(loc, ":");
51. PShape region = country.getChild(location[1]);
52. **if**(region==**null**){
53. println("location " + location[0] + ":" + location[1] + " not found.");
54. district\_map.put(loc, **null**);
55. **return**;
56. }
57. PShape district = region.getChild(location[0]);
58. **if**(district==**null**){
59. println("location " + location[0] + ":" + location[1] + " not found.");
60. district\_map.put(loc, **null**);
61. **return**;
62. }
63. district\_map.put(loc, district);
64. district\_map.get(loc).disableStyle();
65. }
66. //Now we draw it
67. //println(d + " " + r + " " + c);
68. **if**(district\_map.get(loc)!=**null**)
69. {
70. pushStyle();
71. fill(c);
72. shape(district\_map.get(loc), -50, -200);
73. popStyle();
74. }
75. }

78. **void** draw(){
79. //Check if it's time to update the data (once every update\_interval)
80. **float** t\_delta = millis() - time\_update;
81. **if**(t\_delta > update\_interval){
82. time\_update = millis();
83. cur\_vis = next\_vis.copy();
84. println("Updating new data for " + month + "/" + year + " framerate: " + frameRate);
85. **while**(new\_row.getInt("year") == year && new\_row.getInt("month") == month )
86. {
87. String r = new\_row.getString("region");
88. String d = new\_row.getString("district");
89. **float** vis = new\_row.getFloat("vis");
91. //println("loaded district " + d);
93. next\_vis.set(d + ":" + r, curve\_vis(vis));
95. //Update new\_row with next row
96. ++table\_row;
97. **if**(table\_row < table\_size)
98. {
99. new\_row = table.getRow(table\_row);
100. }
102. }
103. println("Update finished in " + (millis() - time\_update));
104. //Move the date
105. **if**(!(year == 2013 && month == 12)){
106. ++month;
107. **if**(month>12){
108. month = 1;
109. ++year;
110. }
111. }
112. }
114. //Always update the map
115. draw\_update = millis();
117. **for**(String loc : cur\_vis.keys()){
118. color from = lerpColor(dark, light, cur\_vis.get(loc)/63.0);
119. color to = lerpColor(dark, light, next\_vis.get(loc)/63.0);
120. color res = lerpColor(from, to, (millis() - time\_update)/update\_interval);
121. //println(t\_delta/update\_interval);
122. **if**(cur\_vis.get(loc)!=next\_vis.get(loc))
123. drawDistrict(loc, res);
124. }
125. println("Draw finished in " + (millis() - draw\_update));
126. }

Finished at 7:30 pm

Total: 3 hours