1: Store n points and let lat_i, lon_i, and $date_i$ represent the latitude, longitude, and date of each point, respectively. Let g represent the grid size measured in degrees latitude, and r represent the radius measured in grid steps. For each lat_i and lon_i, round each to the nearest grid point, and store the rounded points as $tlat_i$ and $tlon_i$. Set $m = 0$.
2: **for** $i = 1$ to n **do**
3: **if** bin_density_hash$(tlat_i, tlon_i)$ exists **then**
4: bin_density_hash$(tlat_i, tlon_i) + +$
5: bin_temporal_hash$(tlat_i, tlon_i) += date_i$
6: **else**
7: bin_density_hash$(tlat_i, tlon_i) = 1$
8: bin_temporal_hash$(tlat_i, tlon_i) = date_i$
9: active_grid_hash$(m) = (tlat_i, tlon_i)$
10: $m + +$
11: **end if**
12: **end for**
13: **for** $j = 1$ to m **do**
14: retrieve $tlat_j$ and $tlon_j$ from active_grid_hash(j)
15: **for** $lat_t = tlat_j - rg$ to $tlat_j + rg$ **do**
16: $t = \arccos(\cos(rg)/\cos(lat_t - tlat_j))/g$
17: round t to the nearest integer
18: $t = t * g$
19: **for** $lon_t = tlon_j - t$ to $tlon_j + t$ **do**
20: density_hash $(lat_t, lon_t) + +$
21: temporal_hash$(lat_t, lon_t) +=$ temporal_hash$(tlat_j, tlon_j)$
22: $lon_t = lon_t + g$
23: **end for**
24: $lat_t = lat_t + g$
25: **end for**
26: **end for**
27: **for** $i = 1$ to n **do**
28: round $(lat_t = lat_i/g)$ to the nearest integer
29: $lat_t = lat_t * g$
30: round $(lon_t = lon_i/g)$ to the nearest integer
31: $lon_t = lon_t * g$
32: temporal_hash$(lat_t, lon_t) =$ temporal_hash(lat_t, lon_t)/density_hash(lat_t, lon_t)
33: print lat_i, lon_i, density_hash(lat_t, lon_t), temporal_hash(lat_t, lon_t)
34: **end for**