Spatial Processing of Global Heat Maps with Apache Spark

Arvind Rao, HERE Technologies
@cwcomplex

#EUds13

Agenda

- Definition of the heat map
  - How its built from access logs
- Contrast enhancement
  - Description of histogram equalization
- Kernel based image processing
  - Gaussian blur. Motivated by noise removal

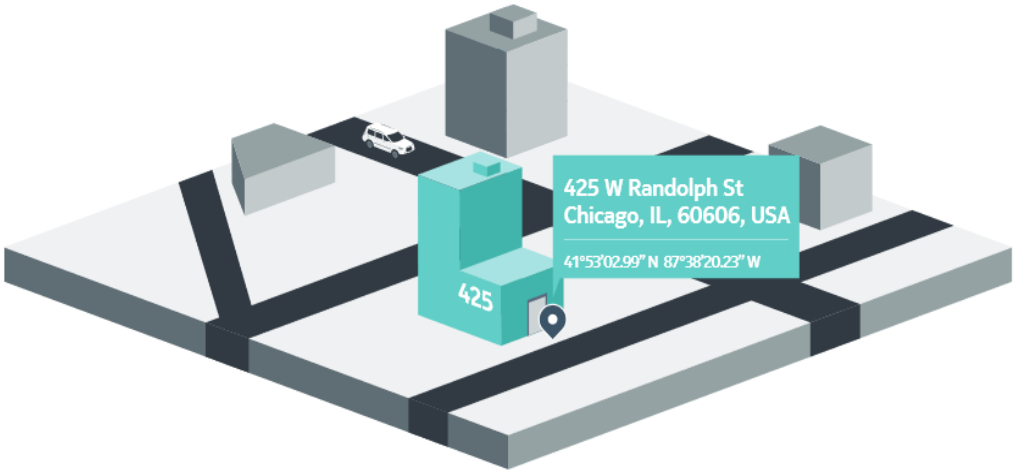
Maps for developers

APIs and SDKs to build location-aware web and mobile apps

Door to door address accuracy

Global coverage with fresh address

High volume and performance multi-reverse geocoding



Request URL

```
https://reverse.geocoder.cit.api.here.com
/6.2/reversegeocode.json
```

Parameters

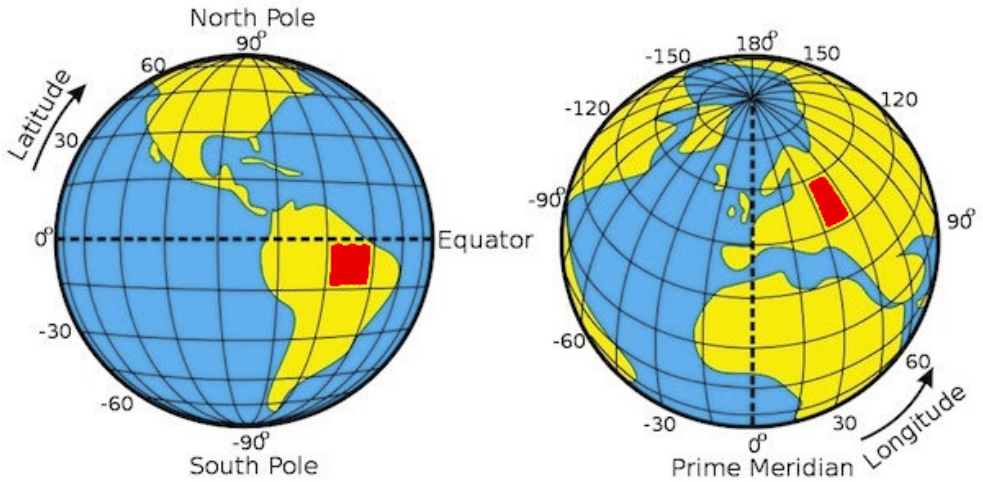
prox	☑	52.5309,13.3847,150
mode	☑	retrieveAreas
gen	☑	8
app_id	☑	DemoAppId01082C
app_code	☑	AJKnXv84fjrb0KIHa

Response

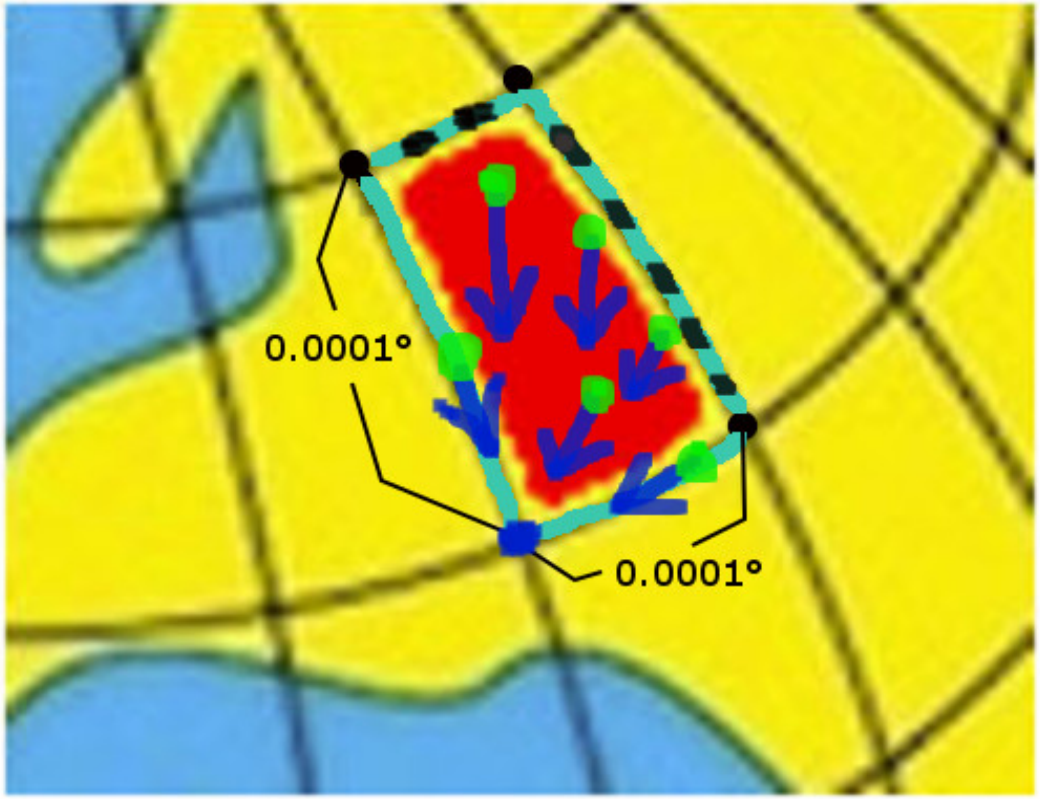
```
                "BottomRight" : {
                  "Latitude": 52.50407,
                  "Longitude": 13.42964
                }
              },
              "Address": {
                "Label": "Mitte, Berlin, Deutschl
                "Country": "DEU",
                "State": "Berlin",
                "County": "Berlin",
                "City": "Berlin",
                "District": "Mitte",
                "PostalCode": "10178",
                "AdditionalData": [
                  {
                    "value": "Deutschland",
                    "key": "CountryName"
                  },
                  {
                    "value": "Berlin",
                    "key": "StateName"
                  },
```

Heat Map is a Spatial Histogram

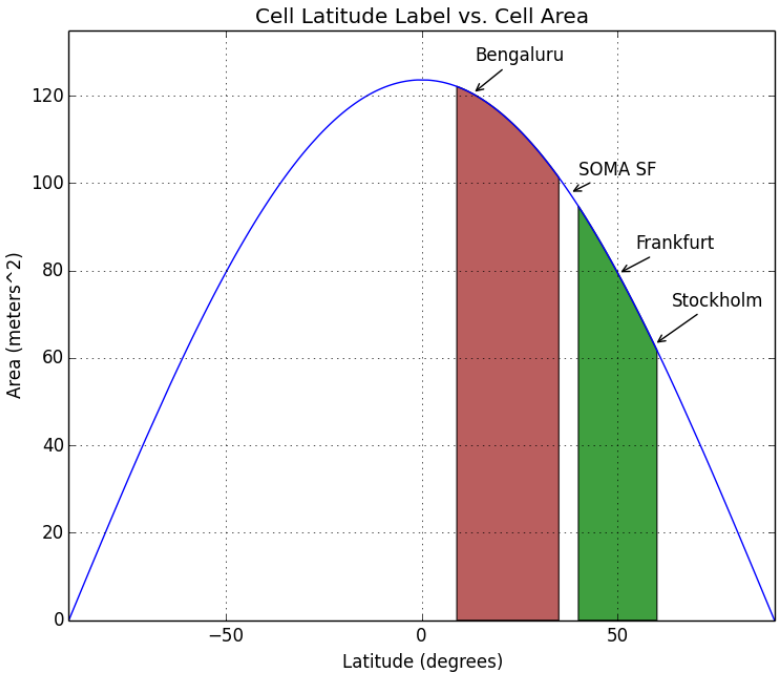
- The earth is partitioned into cells
  - Defined by fixed latitude & longitude increments
- Reverse geocodes within each cell are summed



Aggregation of Request Coordinates



Caveat: Cell Areas Vary by Latitude



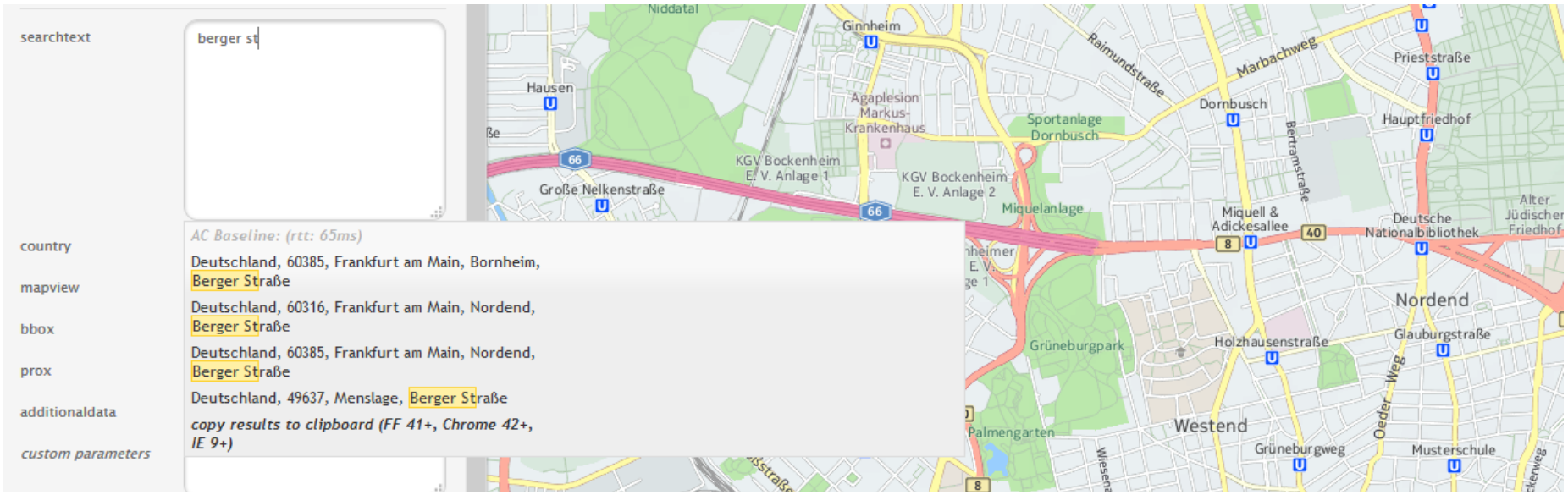
India

Europe

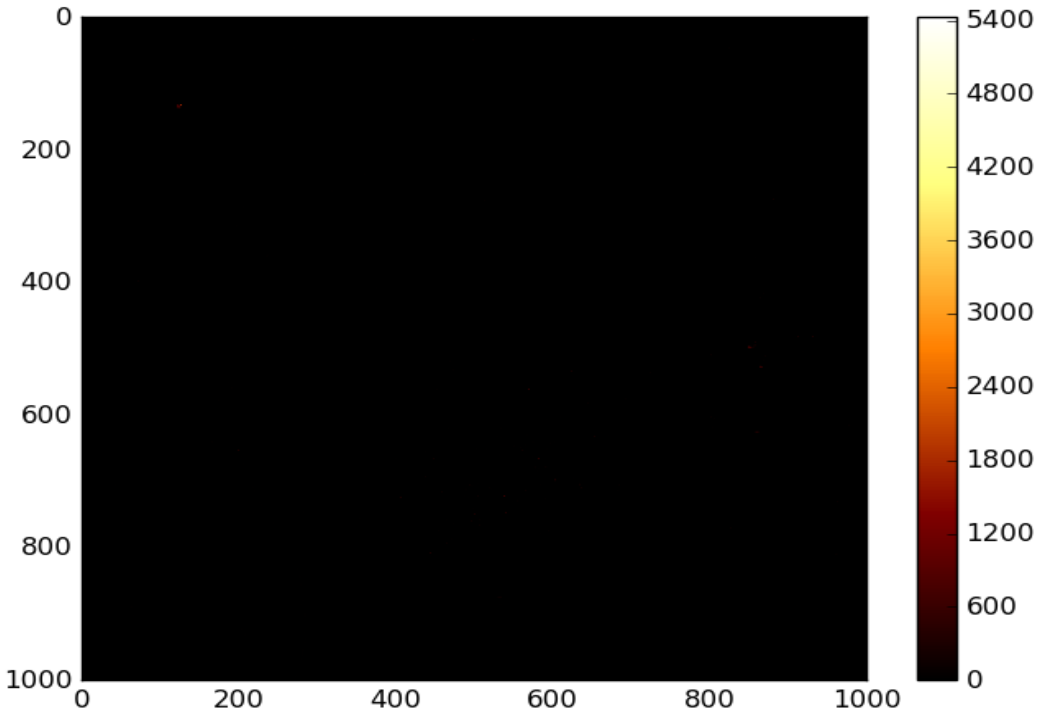
Definition & Data

- Not a conventional image
  - Not acquired by a camera or sensor
  - ~ 6.5 trillion tiles
- High Resolution
  - For perspective a 4k display has ~ 8 million pixels
  - By comparison the heatmap is ~ 1 million times more detailed

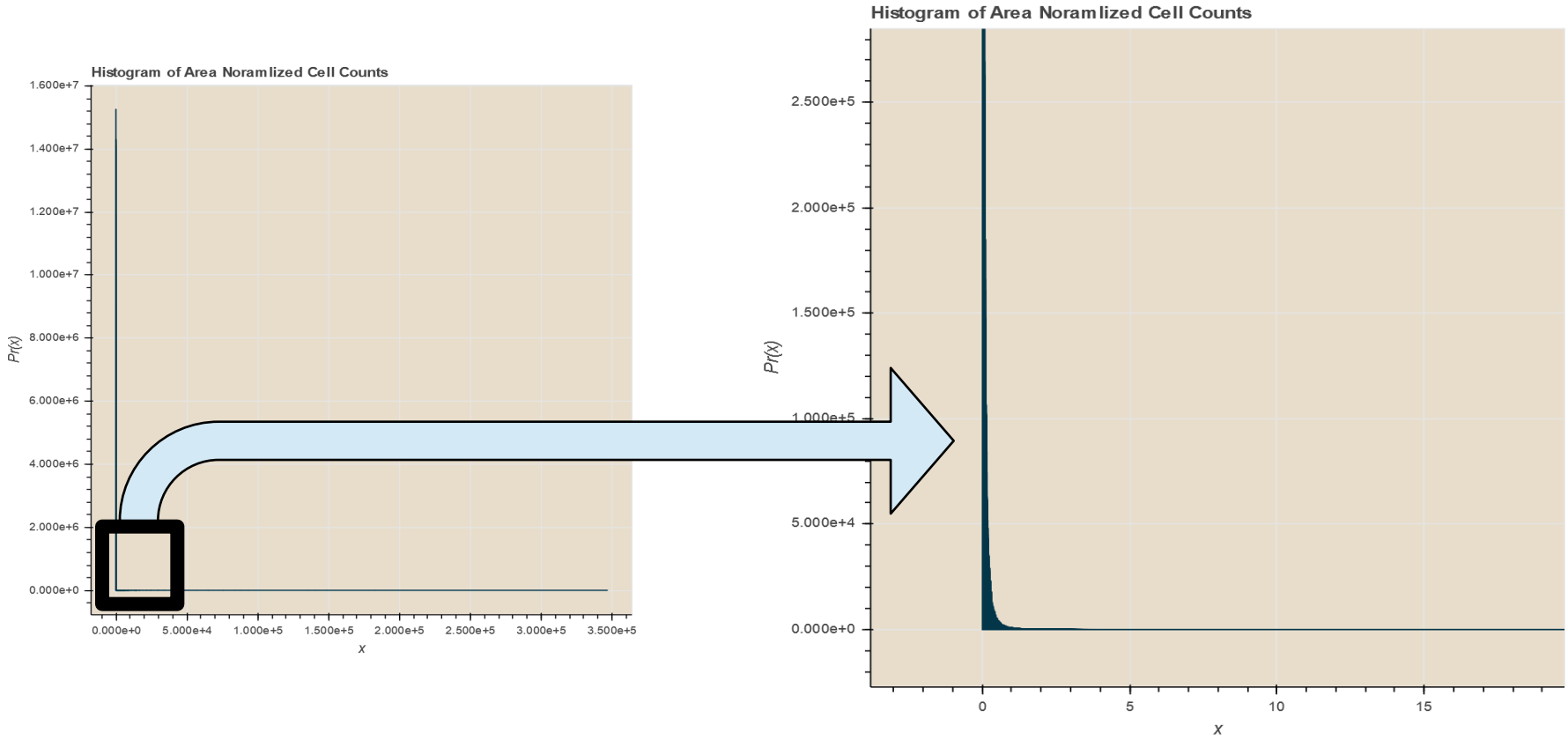
How is it used in HERE Search



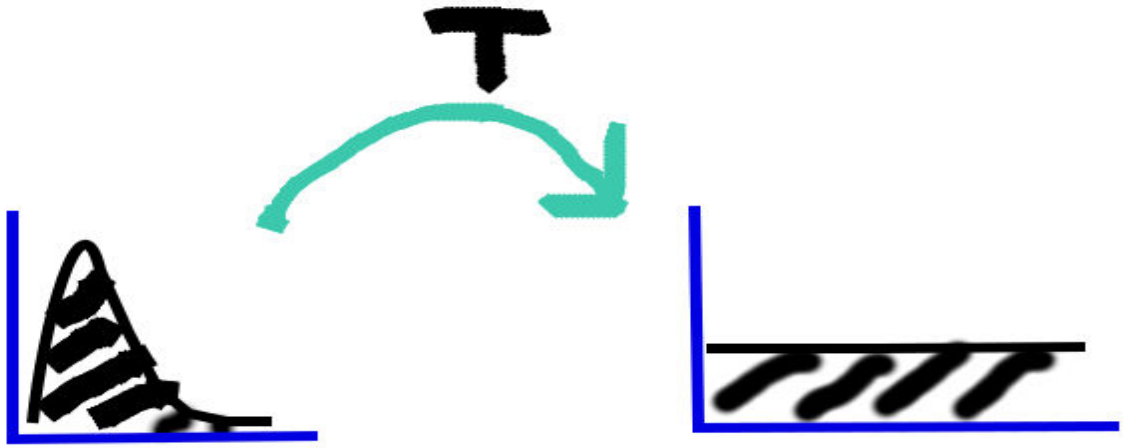
The Histogram is SUPER Skewed



The Histogram is SUPER Skewed



Loose Description of Histogram Equalization (HE)

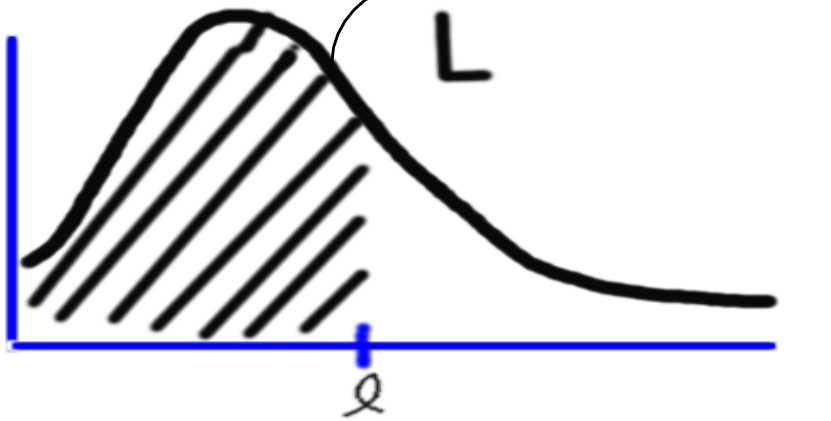


Loose Description of Histogram Equalization (HE)

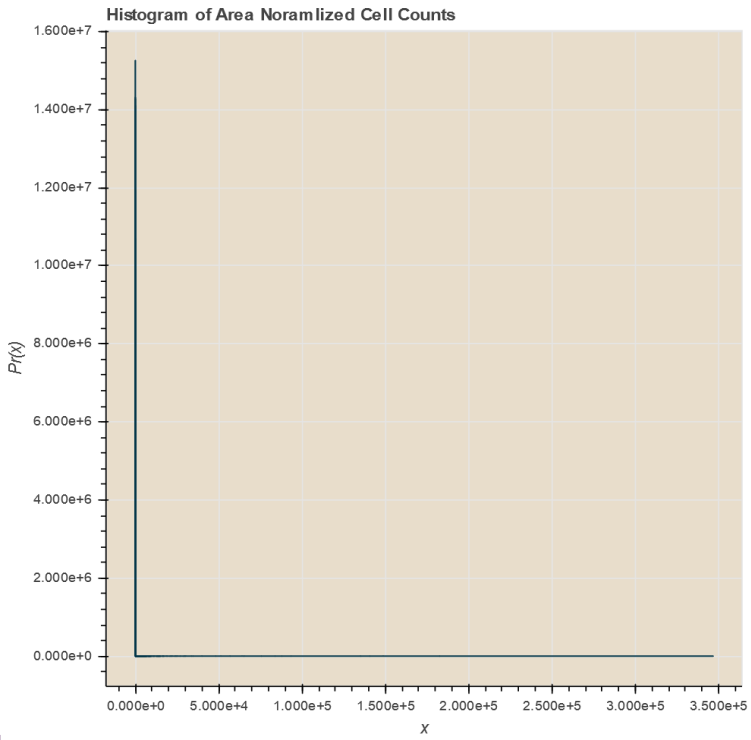
Literally mapping data to percentile rank.

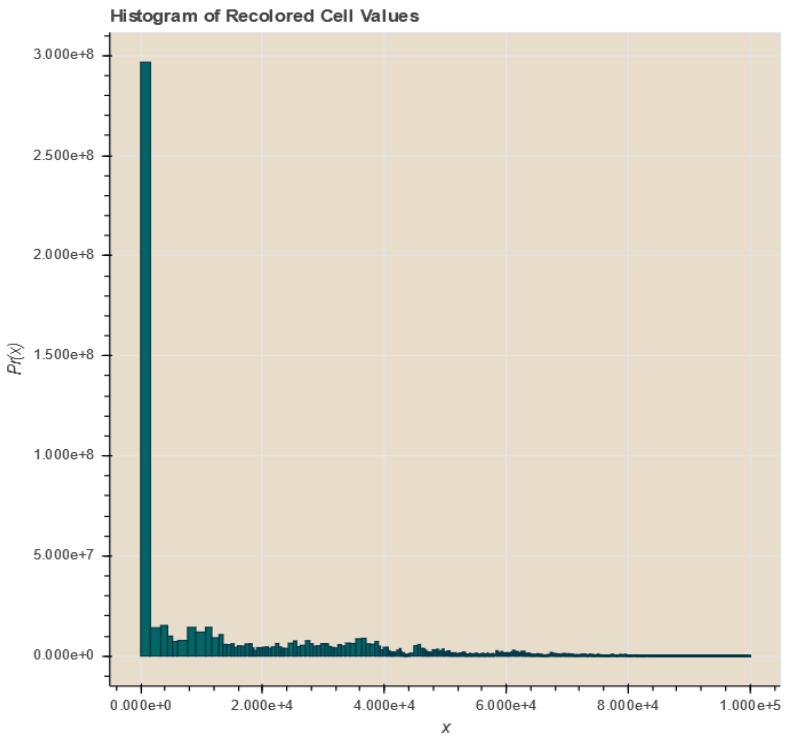
Loose Description of HE

$$l \mapsto \underbrace{P(L < l)}_{} \times \underbrace{N}_{\text{Max Pixel Value}}$$

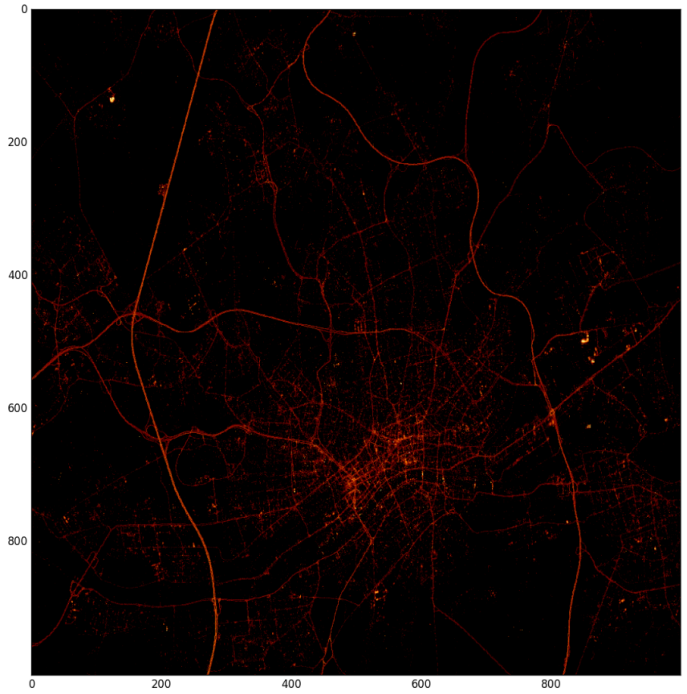


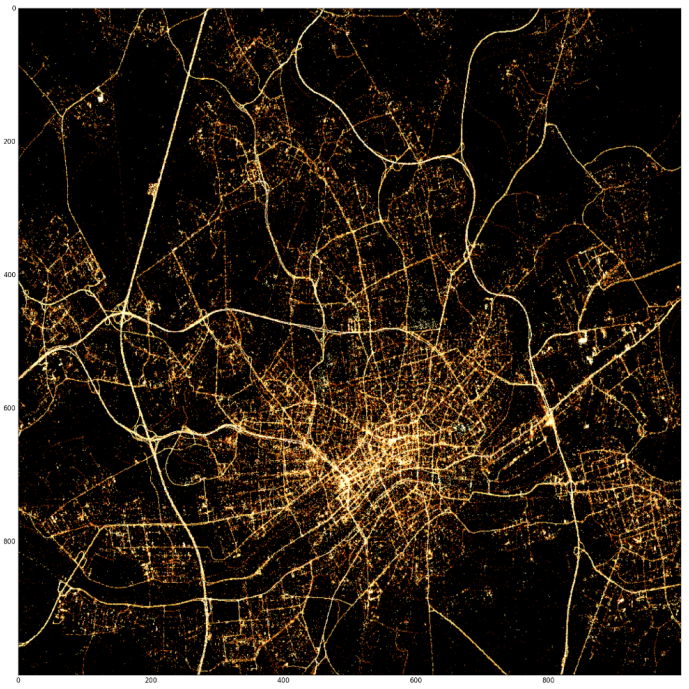
Loose Description of HE





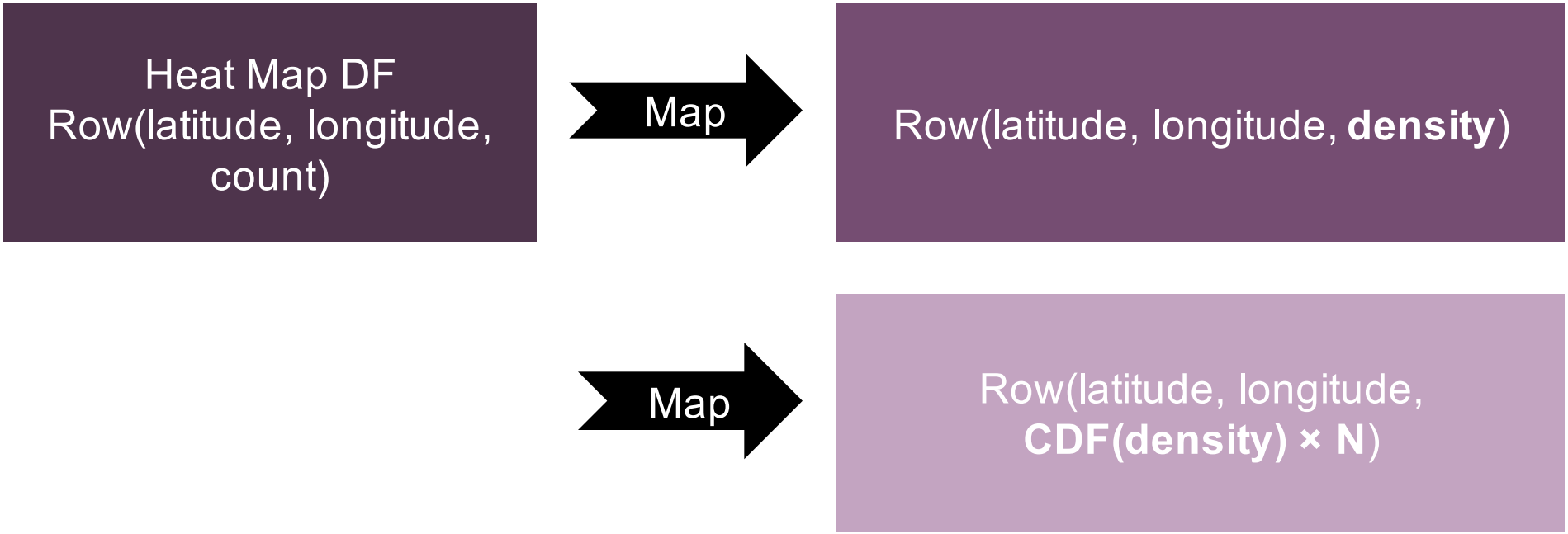
Log Scaled vs. Histogram Normalized



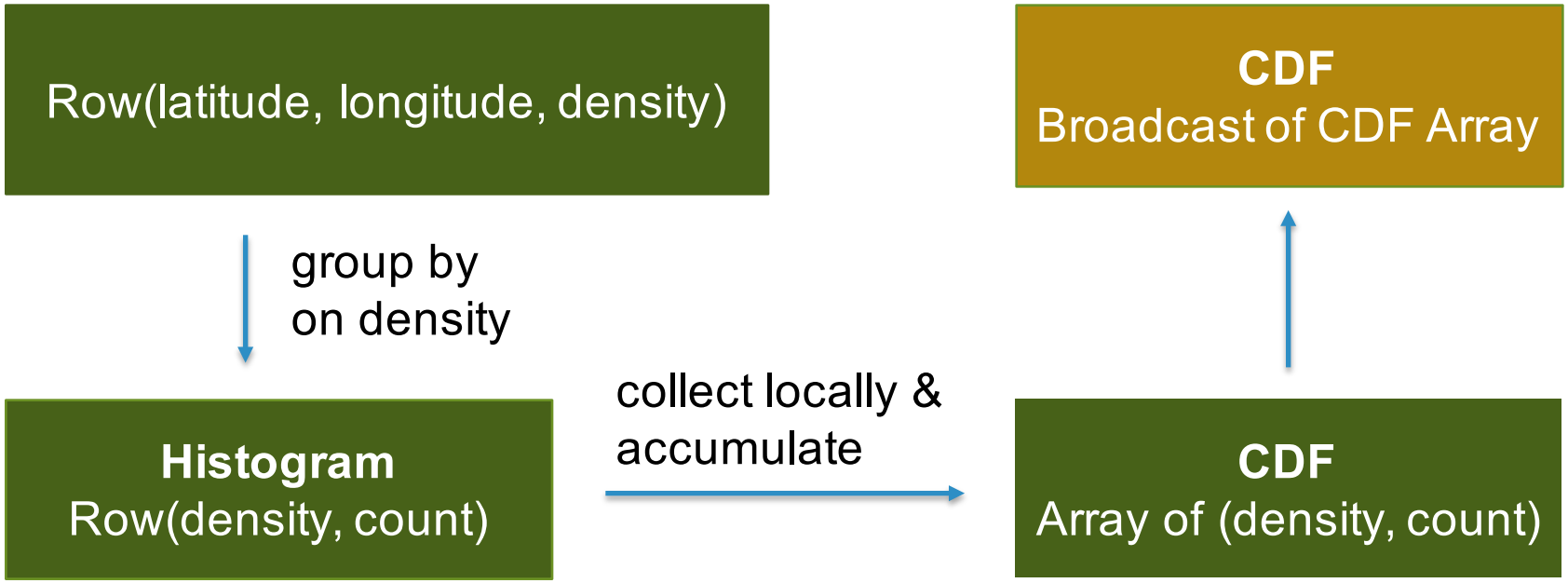


HE – Well- Defined Comparison

HE Implementation



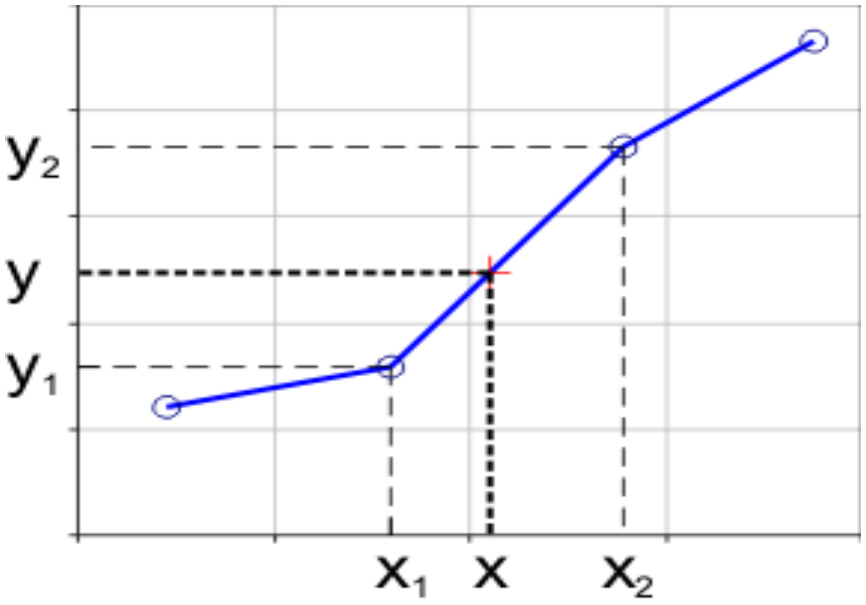
Cumulative Distribution Function



HE Implementation

Binary search finds closest left bin to ***given density*** in CDF.

Then linear interpolate between left and right bins



Some Things To Know

- May increase the contrast of background noise, while decreasing the usable signal

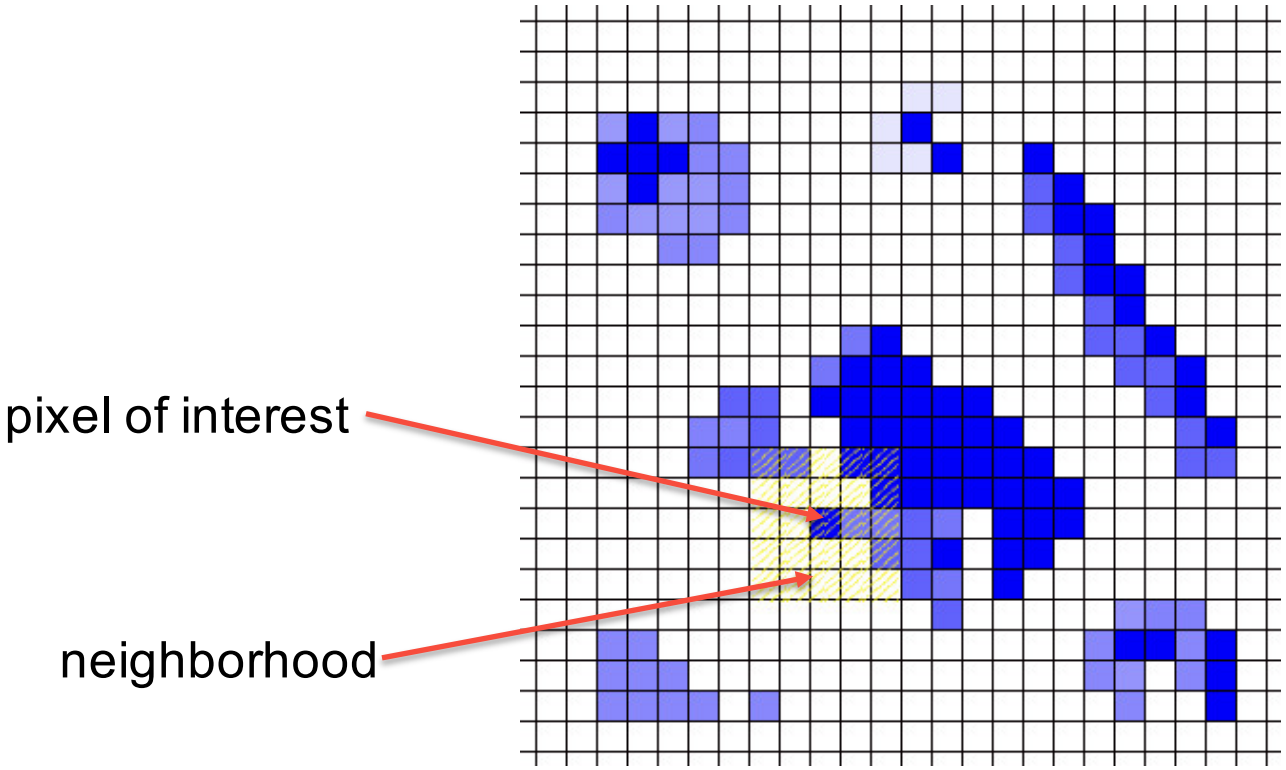
- HE often best in scientific images, like thermal, satellite, or X-rays

Spatial Processing

- Transform image pixel by a function of neighborhood pixel values

- Examples of kernel based methods:
  - Mean, median, Gaussian, etc.
  - Also useful in edge-detection & segmentation

Spatial Processing



Imperative solution:
Slide kernel window over image

- Iterate pixel by pixel
- Each pixel knows its neighborhood.

Spatial Processing – Kernel for Blur

Gaussian kernel with $\sigma = 1$,

$$H(x,y) = \frac{1}{273}\begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$

Transformed image,

$$\tilde{I}(x,y) = \sum_{\Delta x, \Delta y} H(\Delta x, \Delta y) I(x + \Delta x, y + \Delta y)$$

kernel application

Implemented with a GroupBy

Spatial Processing

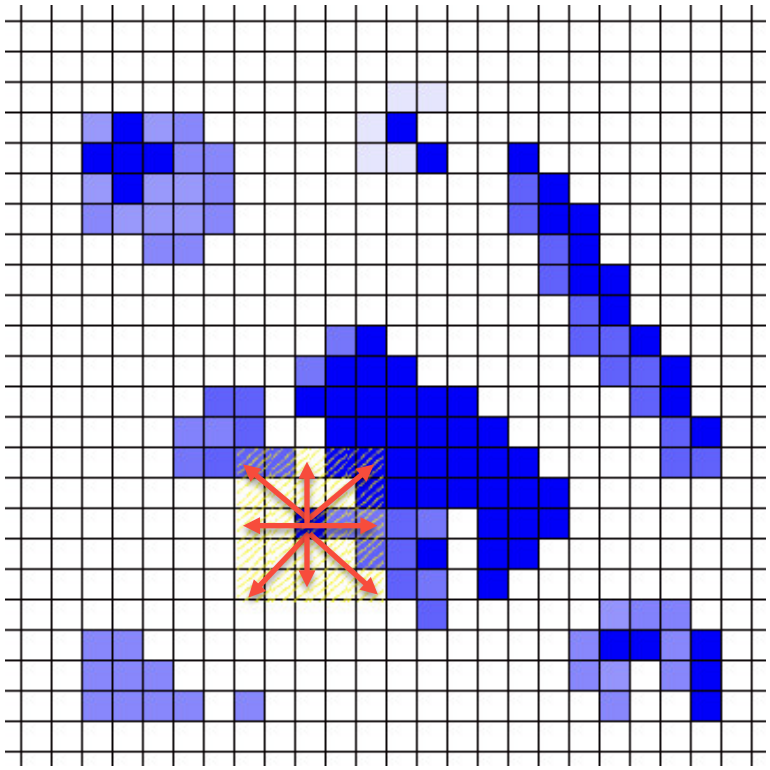
Functional approach: process each pixel independently.

Sketch of Algorithm:

1. Explode heat map to DataFrame of pointers from new pixel indices to all non-null neighbors in the original image
2. Map over exploded DataFrame to apply kernel
3. GroupBy on pixel indices (latitude & longitude)

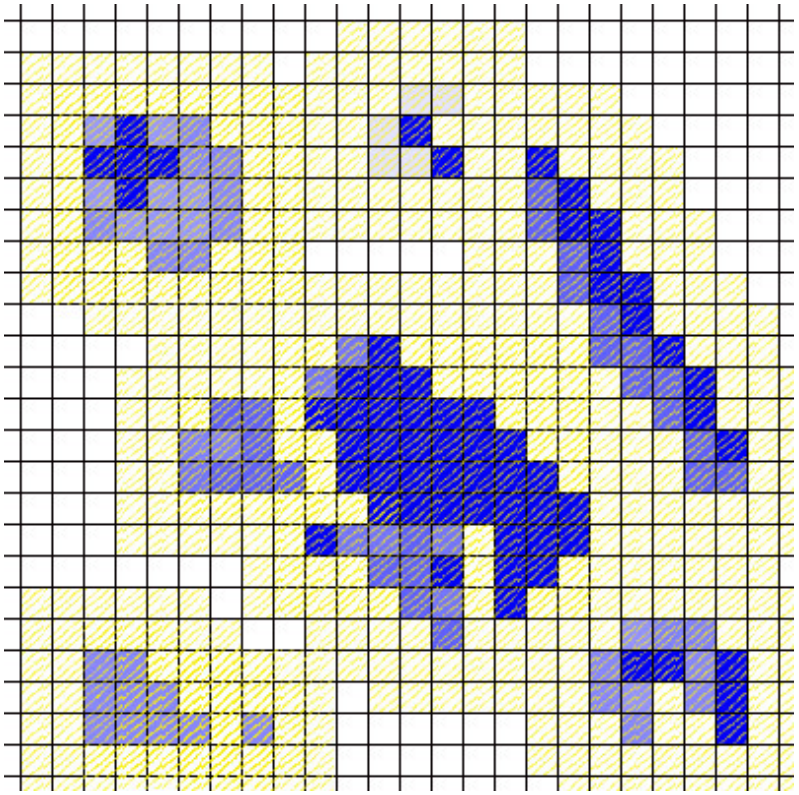
Spatial Processing

Exploding heat map DataFrame gives all indices of pixels in the transformed image.

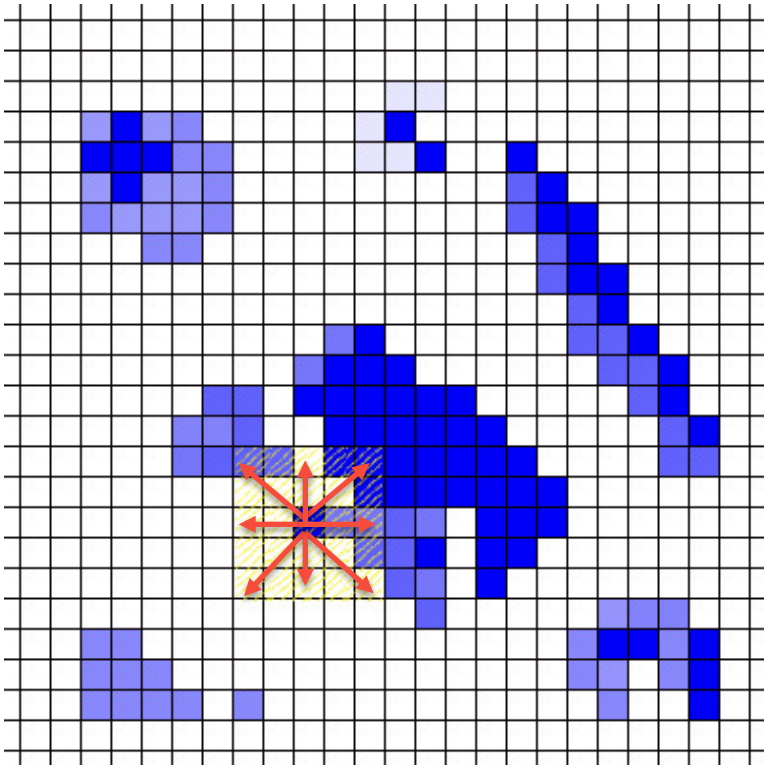


Spatial Processing

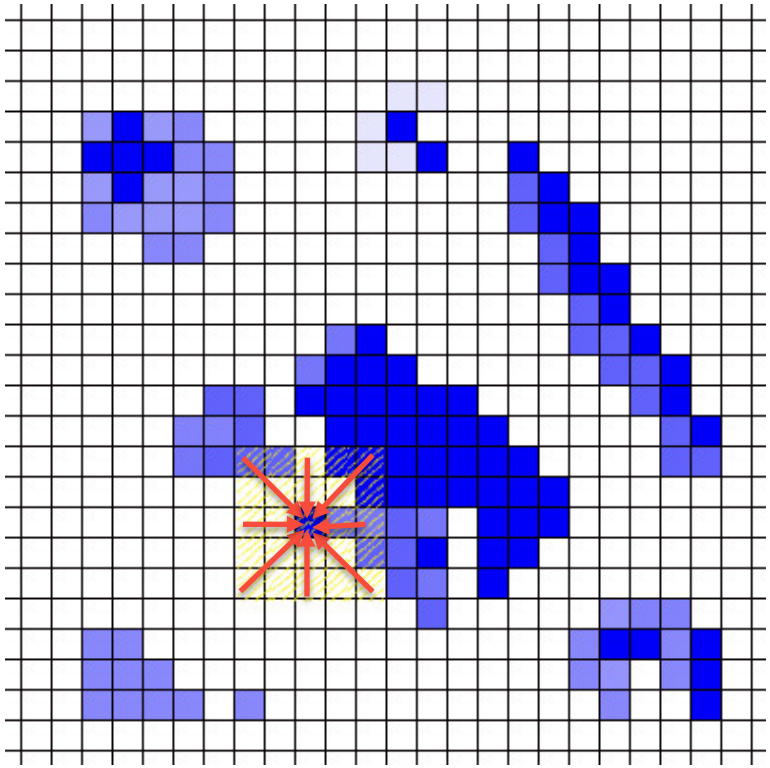
The smoothed image is larger and less sparse.



Spatial Processing



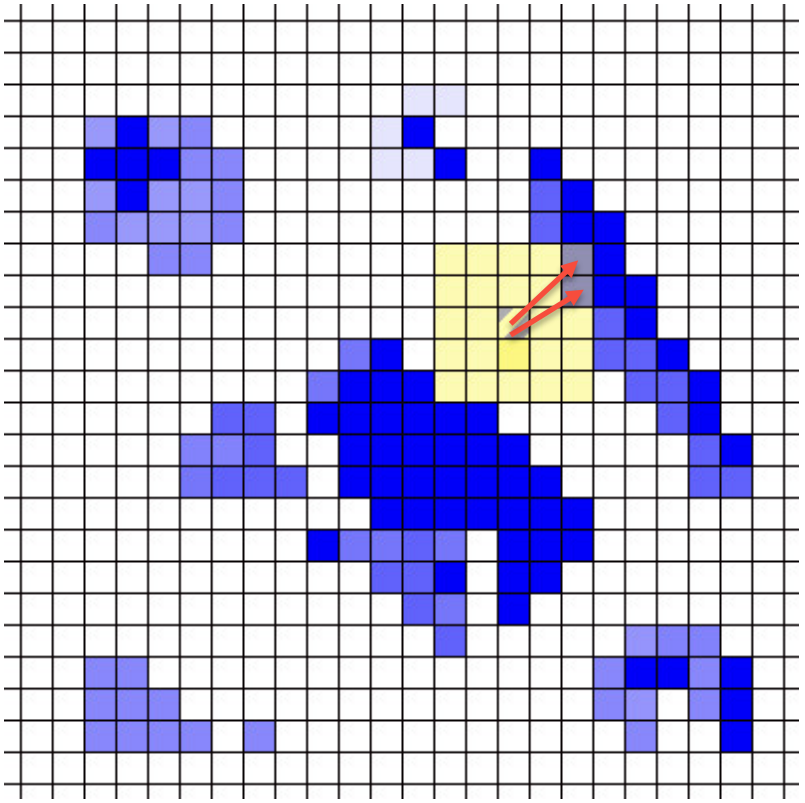
Being a neighbor is reciprocal.



Spatial Processing

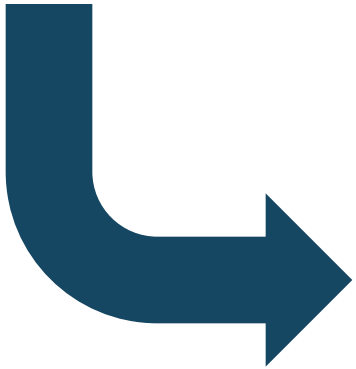
Should think of rows in the exploded DataFrame as a pointer from any pixel index in the image domain to its non-null neighbor from input image.

Sparsity guarantees that a pixel index exists in exploded DataFrame *iff* it has a non-null neighbor.



Spatial Processing

x	y	value
34.23	120.56	120
45.23	123.12	23



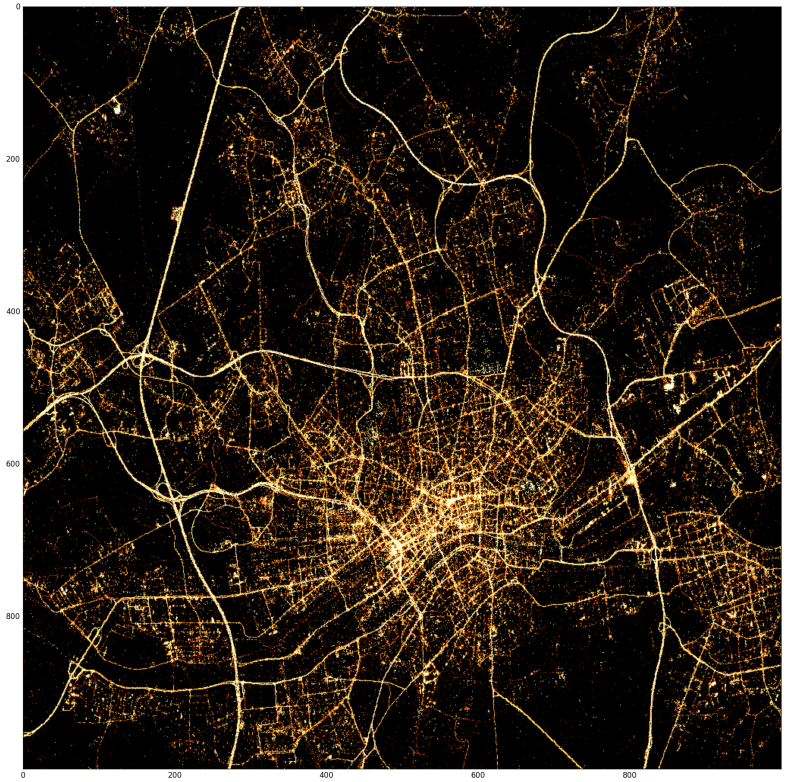
Δx	Δy	nb-x	nb-y	originating value
0	0	34.23	120.56	120
1	-1	34.24	120.55	120
2	-2	34.25	120.54	120
⋮	⋮	⋮	⋮	⋮

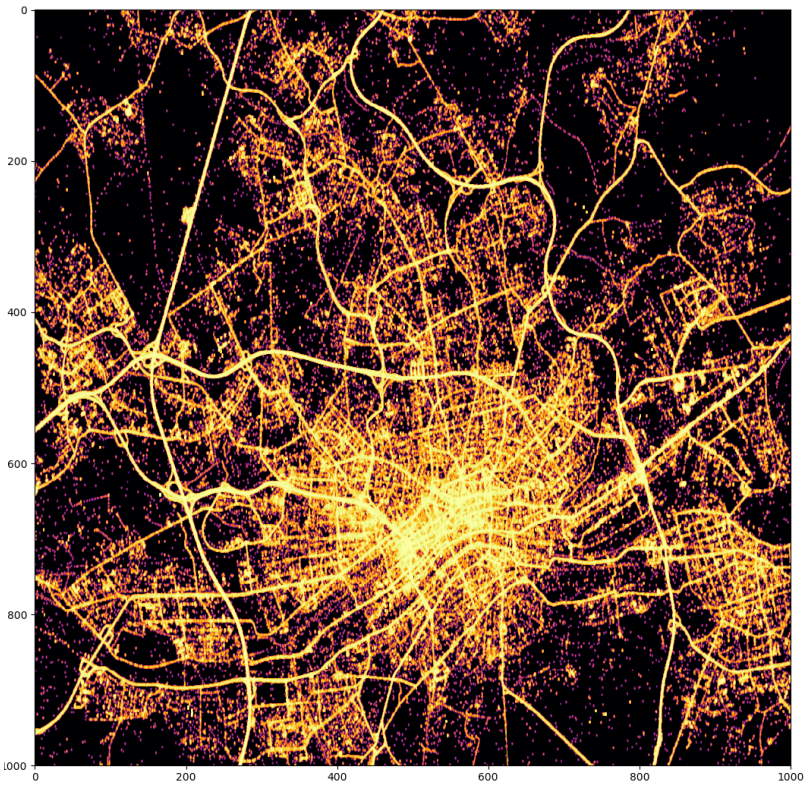
Spatial Processing

Sketch of Algorithm

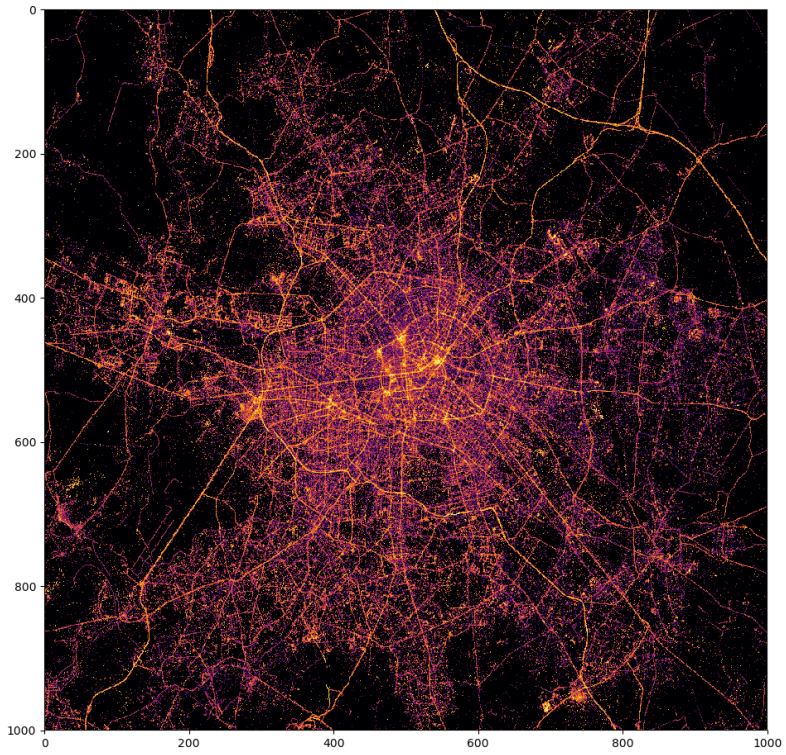
1. Explode heat map to DataFrame of pointers from new pixel indices to all non-null neighbors in the original image.
2. Map over exploded DataFrame to apply kernel.
3. GroupBy on pixel indices (latitude & longitude)

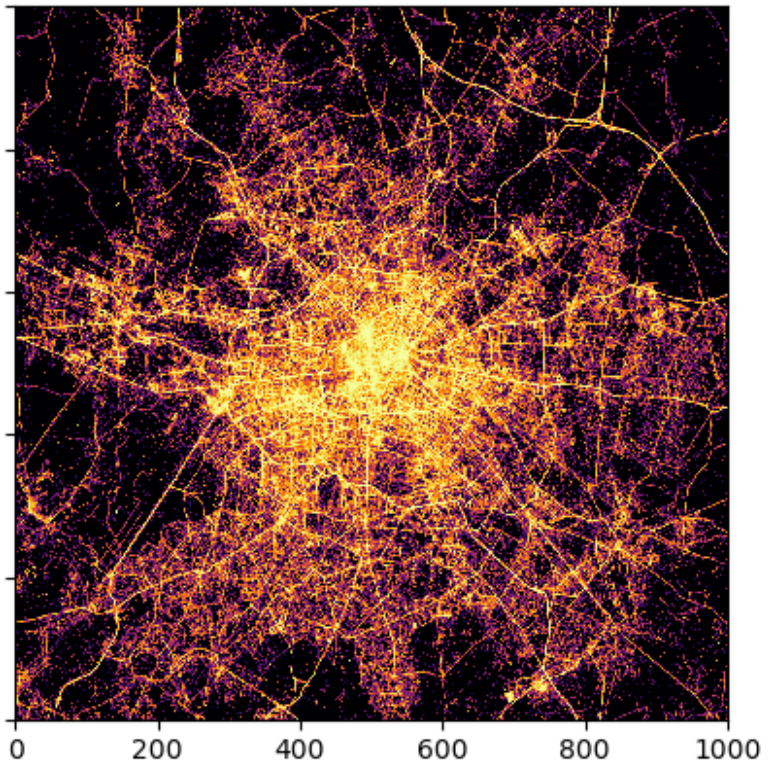
Frankfurt – Without & With Blurring





Berlin – Without & With Blurring





Take Care!

Arvind Rao, HERE Technologies
@cwcomplex

#EUds13