Crowd-Sourced Bathymetry Cell Aggregation

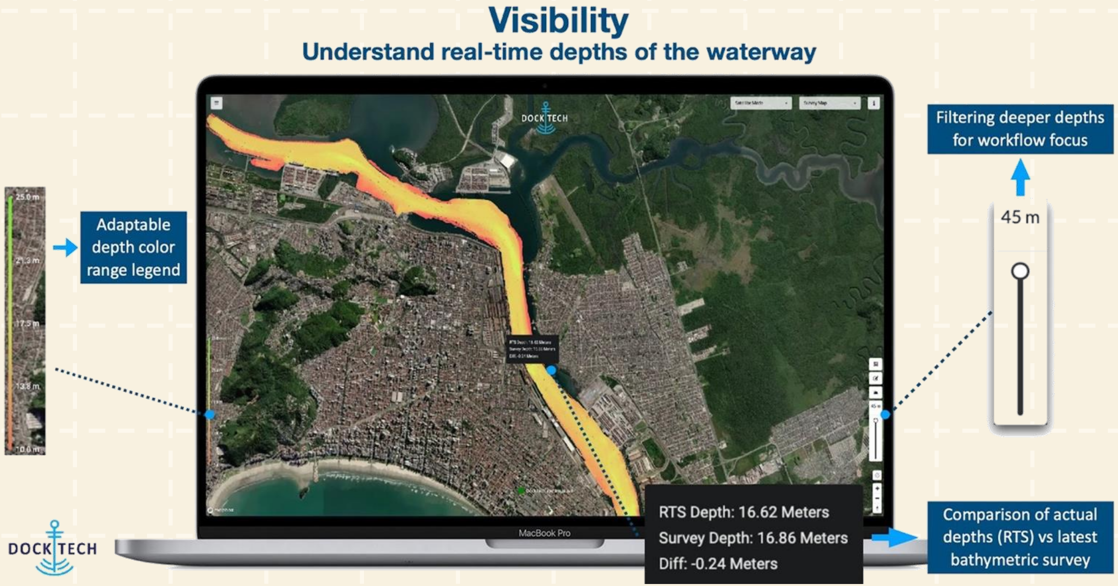
An Overview

DOCKTECH
THE FUTURE OF DEPTH

Quick Look at DockTech

DockTech offers a real-time status of the waterways conditions (digital twin) to facilitate maritime operations and prediction-based activity, leveraging AI and crowd-sourced dynamic data.

Implementing DockTech's solution in ports ensures safer navigation and better resource allocation, improving the overall maritime supply chain efficiency.



Quick Look at DockTech

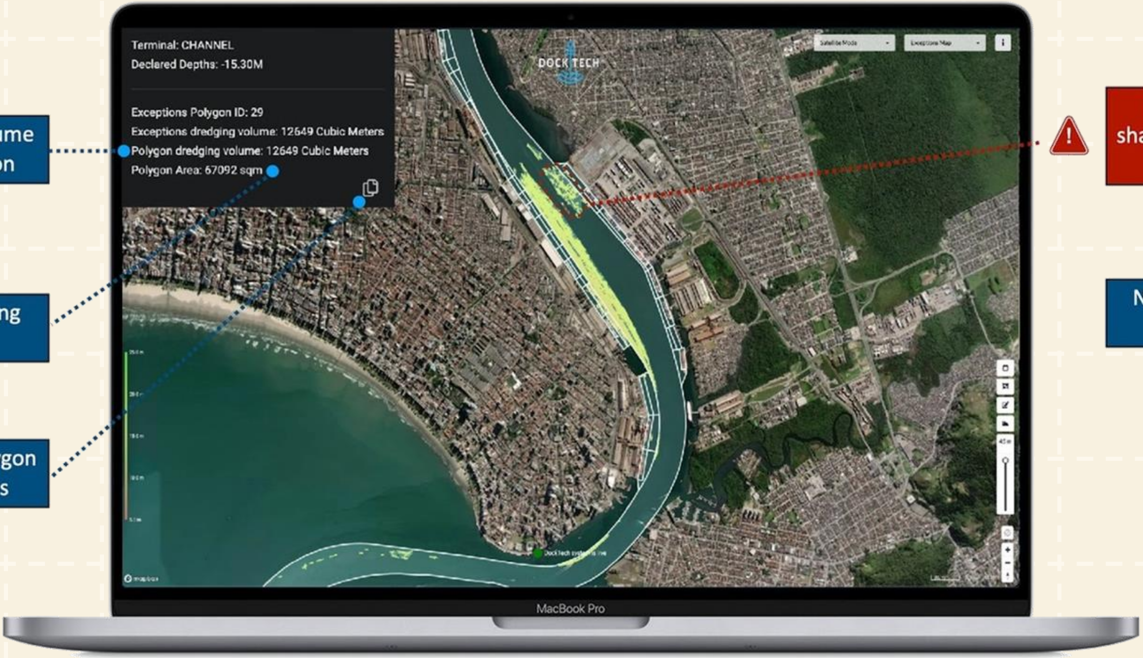
Efficiency

Dredging on demand - know where and when to dredge

Dredging volume interpolation

Total dredging area

Dredging polygon coordinates



Exception - shallow threshold exceeded

Notification to users

DOCK TECH

Quick Look at DockTech

Predictability

Forecast sedimentation and erosion

Erosion \ dredging Identification

Sedimentation detection

DockTech - Progress in 2022

- Just published paper in the IHR: Evaluation of a Crowd Sourced Bathymetric Approach
- DockTech's Data Loggers (DT99) onboard around
  100 service vessels in over 30 ports, across 4 countries.
  - Expanded to Colombia, and the States.
- Started to collect Wind Data from Tugboats
- Integrated AIS Data within our App
- Migrated to new Data Infrastructure to 10+x decrease query speed.
- Integrated GPS Noise Filtering
- Started VAX (Vessel Activity Explorer) for partnered tugboats.
- Generated new dashboards for data partners.
- Created new compression integration to handle data workloads.
- Implement support for new GPS NMEA sentence and contributed to open source

DockTech - What’s on Deck

- Working on new papers with Hydrographers in Brazil.
- Post-Processed filtering development.
- Research on new methods of Tide Enrichment.
- Add Precision Interval to our models (precursor to our own form of CATZOC).
- Allow users to look back at previous Depth maps,
  and provide tools to compare Depth layers.
- Automatic monitoring/alerts for tugboat partners.
- Increase precision from GPS/GNSS
- Explore Dynamic Draft

Overview

1

The Problem

Discussing the unique situation CSB is with aggregation.

2

Considerations

Discussing the various distinctions between options.

3

The Options

Discuss a few of the main options currently available.

4

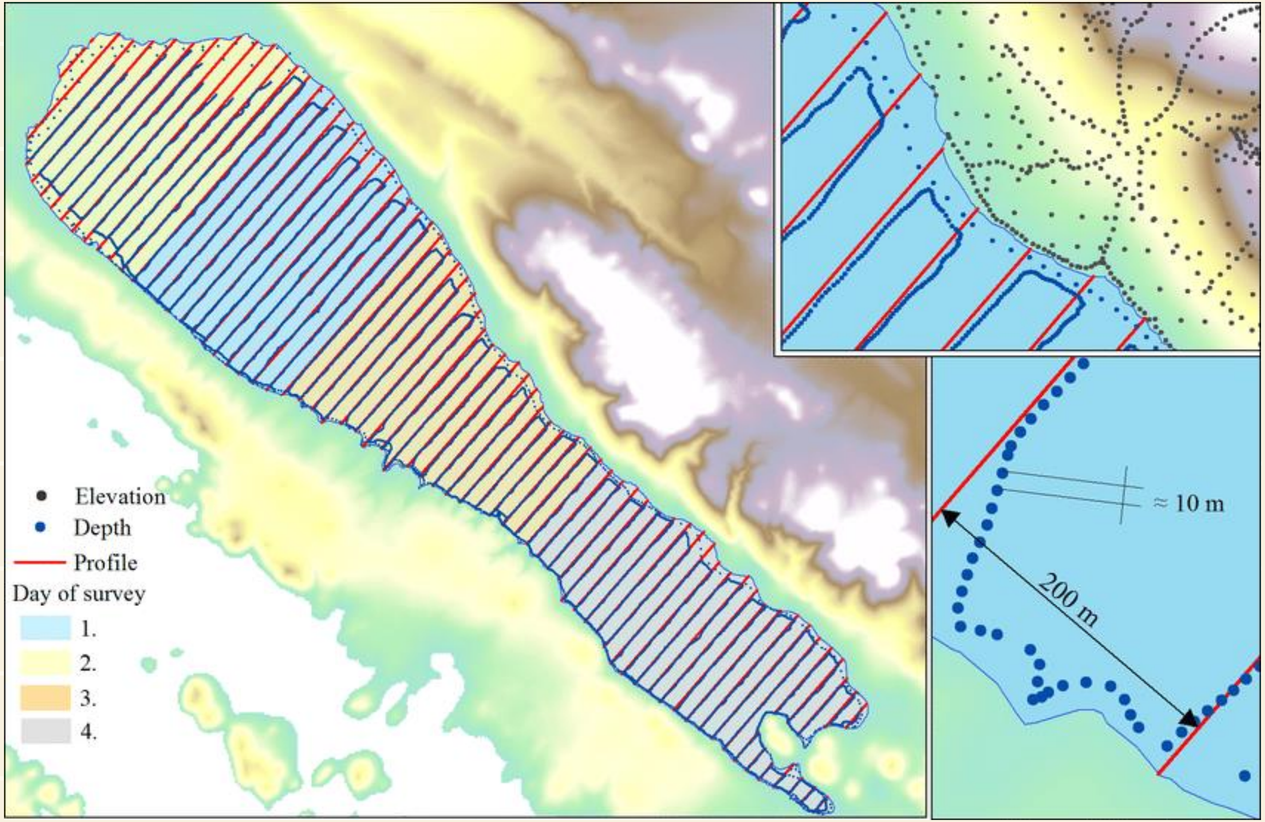
DockTech’s Choice

Discussing DockTech’s concerns and their choice.

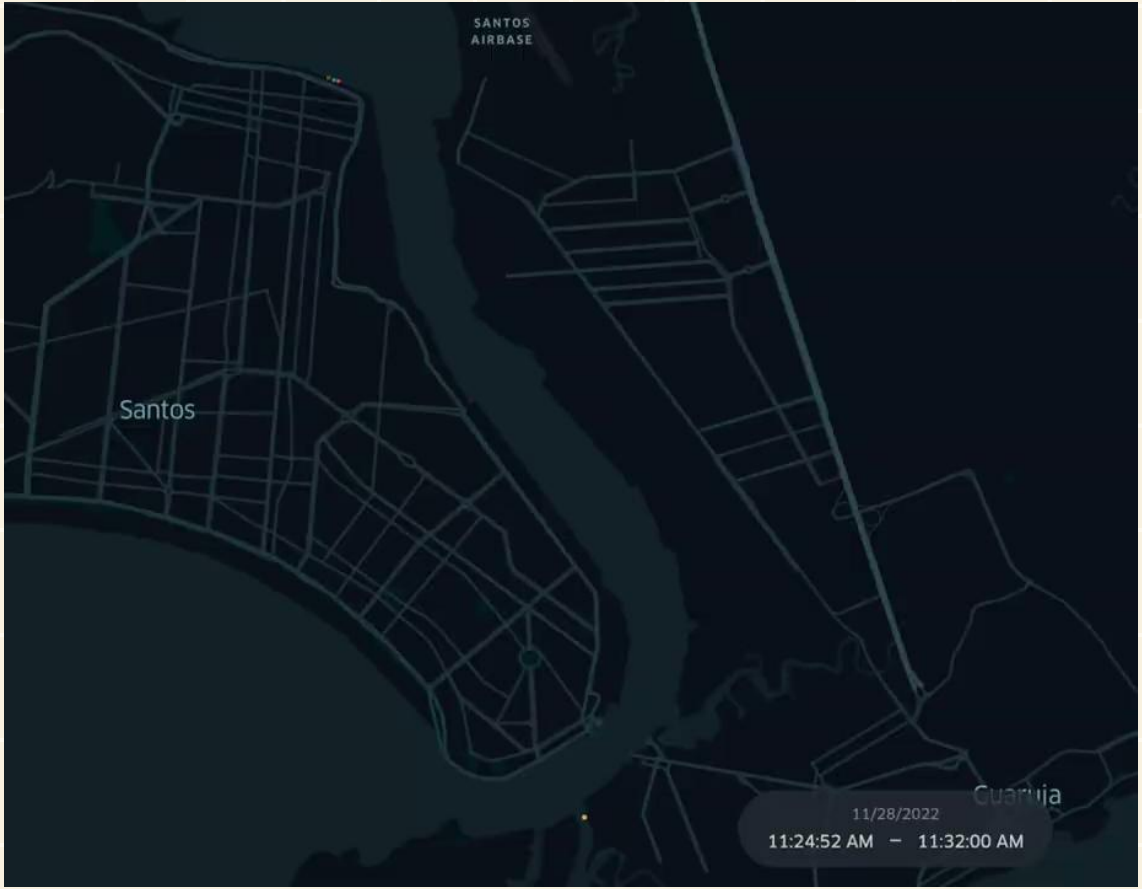
1

The Problem

From continuous to discrete space



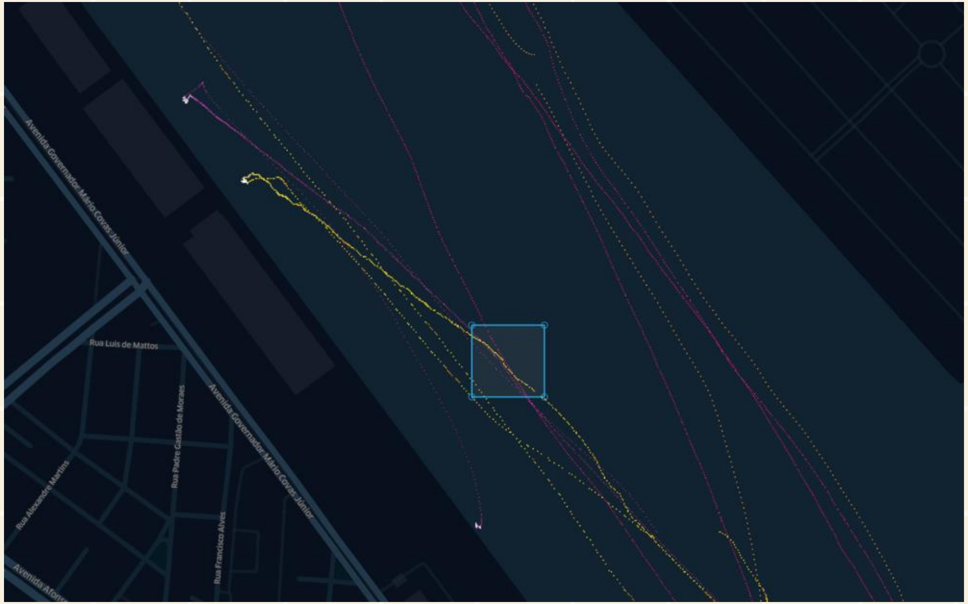
Sailing route for MBES Bathymetric Survey



Example of observations from Passage Soundings

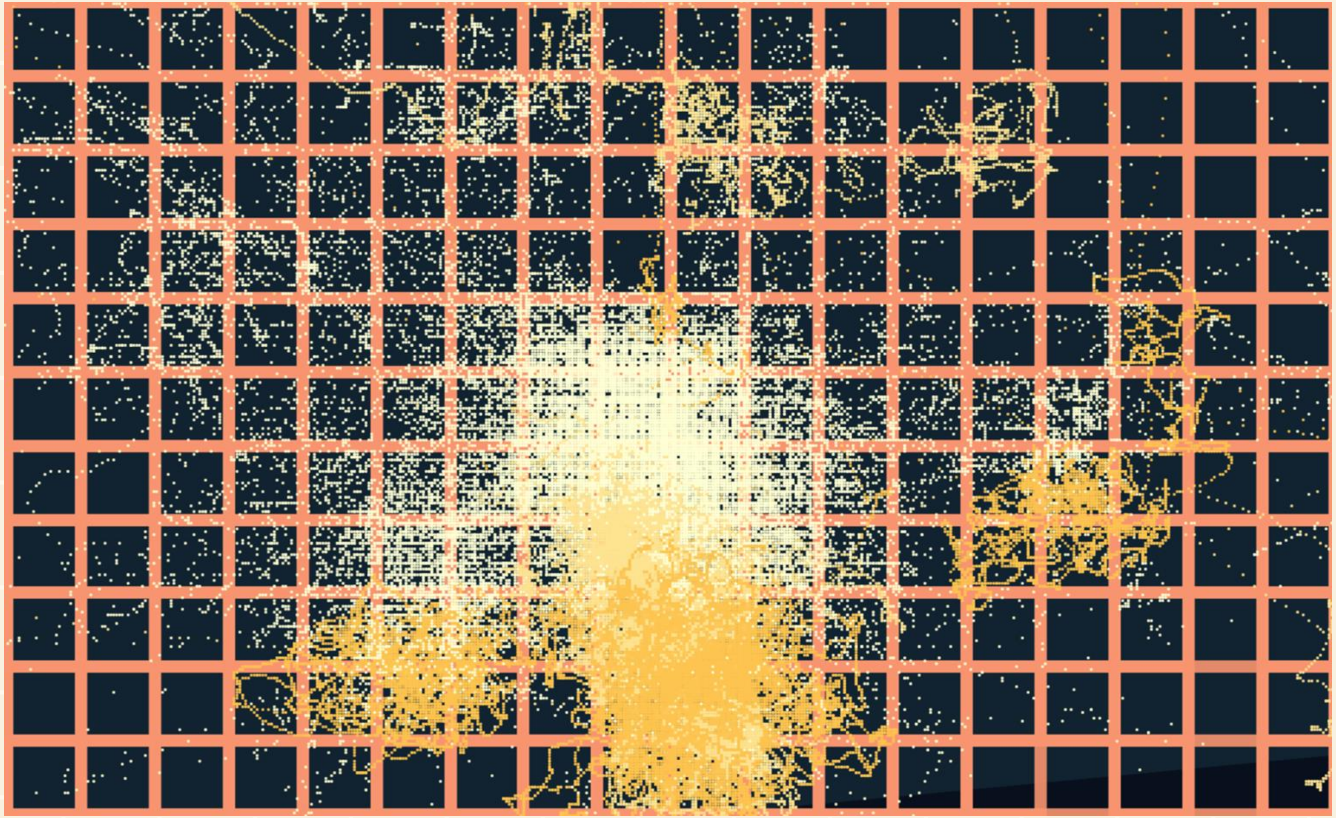
decimal places	degrees	distance
0	1	111 km
1	0.1	11.1 km
2	0.01	1.11 km
3	0.001	111 m
4	0.0001	11.1 m
5	0.00001	1.11 m
6	0.000001	11.1 cm
7	0.0000001	1.11 cm
8	0.00000001	1.11 mm
9	0.000000001	111 µm
10	0.0000000001	11.1 µm
11	0.00000000001	1.11 µm
12	0.000000000001	111 nm
13	0.0000000000001	11.1 nm

Lat/Lon are continuous



Take advantage of:

1. Multiple obs over same space
2. Obs from several Vessels



2

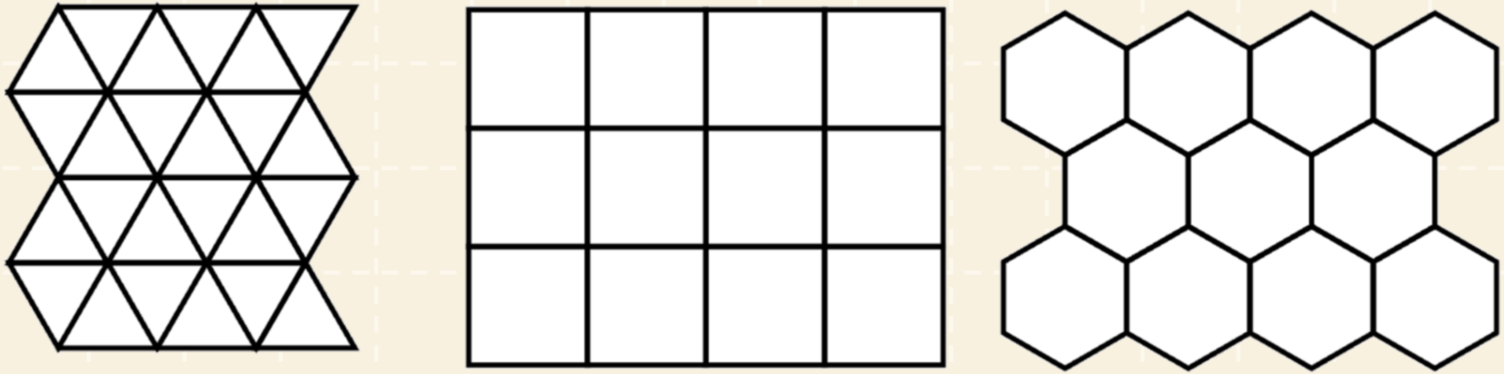
Considerations

One Shape to rule them all

Tessellation

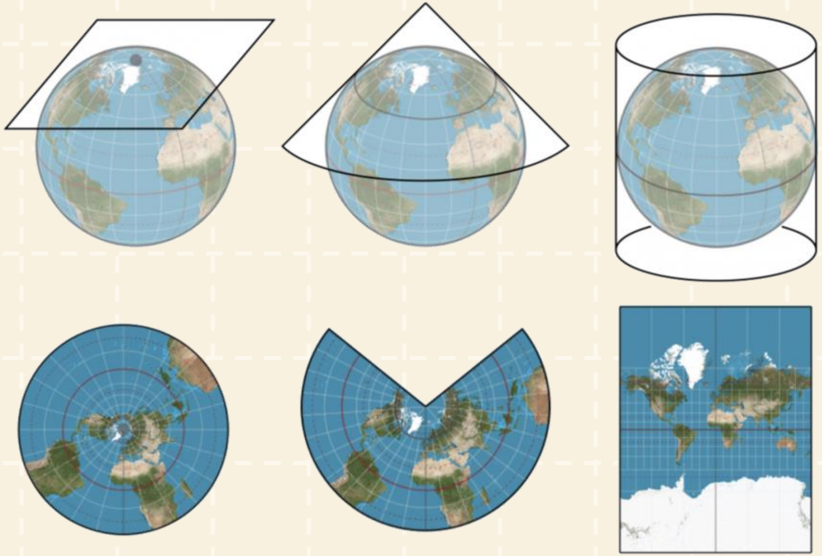
Famous M.C. Escher drawing

Tessellation

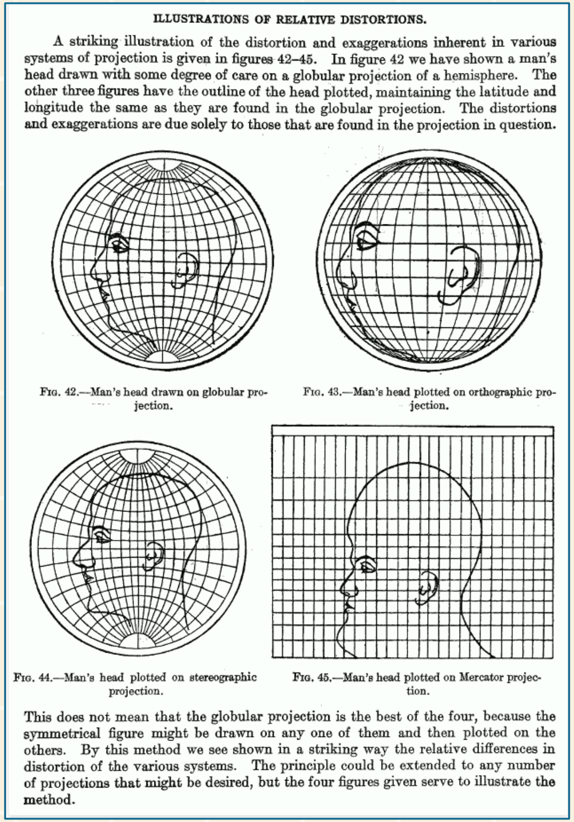


Only a triangle, square, and hexagon are the only simple polygons that tessellate.

Distortion



Converting from 3d to 2d will always lead to some form of distortion.



ILLUSTRATIONS OF RELATIVE DISTORTIONS.

A striking illustration of the distortion and exaggerations inherent in various systems of projection is given in figures 42–45. In figure 42 we have shown a man's head drawn with some degree of care on a globular projection of a hemisphere. The other three figures have the outline of the head plotted, maintaining the latitude and longitude the same as they are found in the globular projection. The distortions and exaggerations are due solely to those that are found in the projection in question.

FIG. 42.—Man's head drawn on globular projection.

FIG. 43.—Man's head plotted on orthographic projection.

FIG. 44.—Man's head plotted on stereographic projection.

FIG. 45.—Man's head plotted on Mercator projection.

This does not mean that the globular projection is the best of the four, because the symmetrical figure might be drawn on any one of them and then plotted on the others. By this method we see shown in a striking way the relative differences in distortion of the various systems. The principle could be extended to any number of projections that might be desired, but the four figures given serve to illustrate the method.

1921 Paper by Deets, Adams discussing map projections.

Distortion

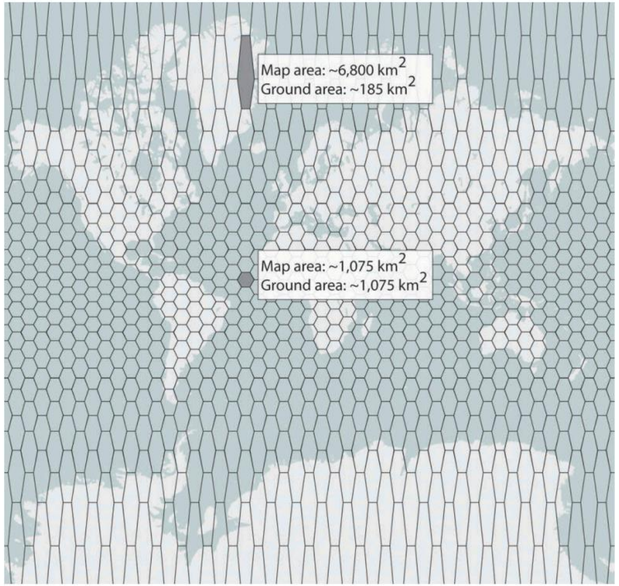
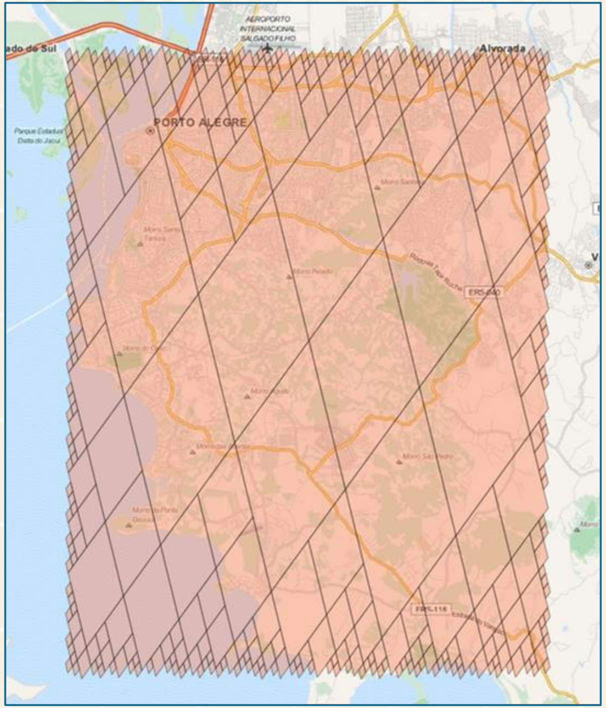


Figure 6. Hexagonal bins defined as 10° side on a plate carrée projection, and projected onto a Web Mercator base map. Measurements listed provide relative map area and ground area.

Shapes on a plane: evaluating the impact of projection distortion on spatial binning



Example of shape distortion on Google's S2

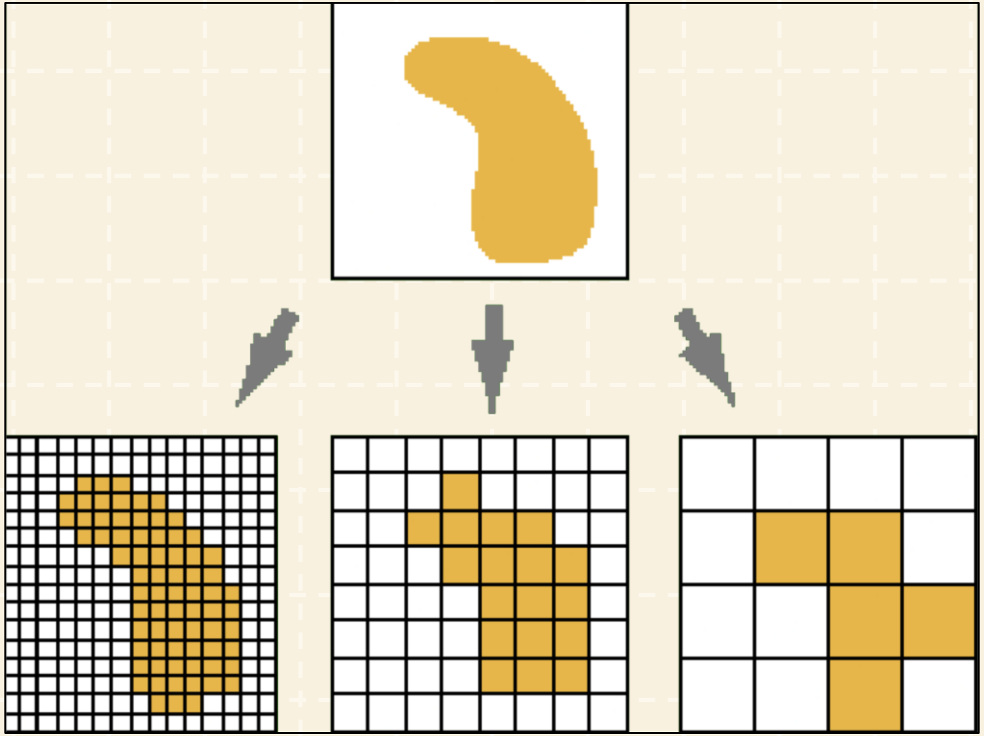
Subdivision

Example of subdivision for geohash

Spatial Hierarchy

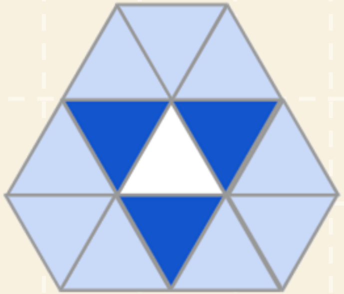
H3 cells covering California in dense (left) and sparse (right) cells.

Flexible Precision



Example of representing raster data with various cell sizes.

Neighbor Traversal

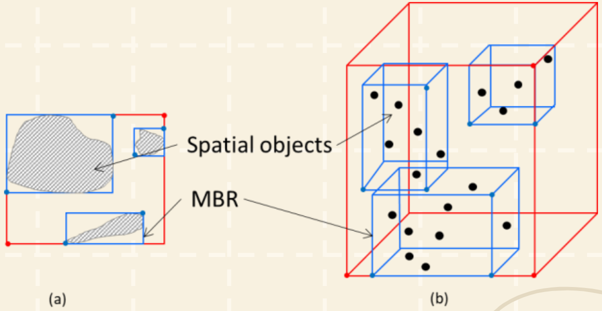


Triangle - 3 types of neighbors
Square - 2 types of neighbors
Hexagon - 1 type of neighbor (all equidistance)

Spatial Indexing

esri

PostGIS



Commonly used method for spatial indexing, Minimum Bounded Rectangle (MBR)

Interoperability

```
>>> import h3
>>> lat, lng = 37.769377, -122.388903
>>> resolution = 9
>>> h3.latlng_to_cell(lat, lng, resolution)
'89283082e73ffff'
```

```
>>> import libgeohash as gh

>>> gh.encode(57.64911, 10.40744, precision = 10)
'u4pruydqqv'

>>> gh.decode('u4pruydqqv', errors = True)
(57.64911264181137, 10.407437682151794, 2.682209014892578e-06, 5.364418029785156e-06)

>>> gh.neighbors('u4pruydqqv')
{'ne': 'u4pruydqrn', 'e': 'u4pruydqrj', 'n': 'u4pruydqqy', 'se': 'u4pruydqrh',
'w': 'u4pruydqqt', 'sw': 'u4pruydqqs', 'nw': 'u4pruydqqw', 's': 'u4pruydqqu'}

# Returns dimensions of the bounding box referred by the geohash in meters. (width, height)
>>> gh.dimensions('u4pruyd')
(152.9, 152.4)

# Returns the great circle distance (Haversine) between two geohashes or coordinates.
>>> gh.distance('u4pruyd', 'u4pruyg')
173.19066702376304
```

databricks

CARTO

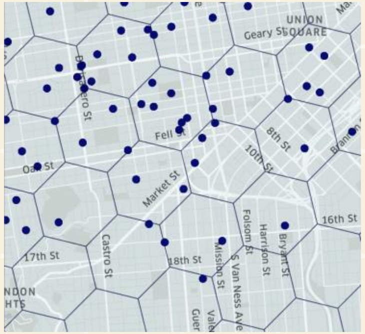
Examples of h3 and geohash (left) being implemented easily within Python.
Examples of platforms that help integrate geospatial data (right).

3

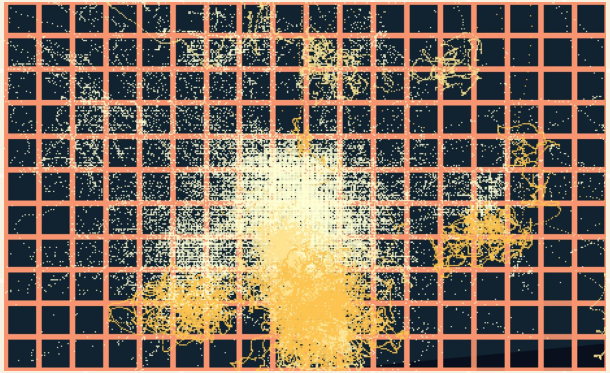
The Options

Well most of the most popular.

The Options

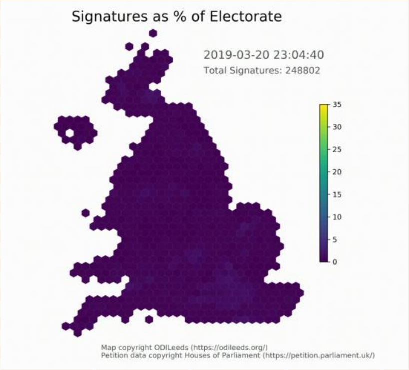


Uber's H3

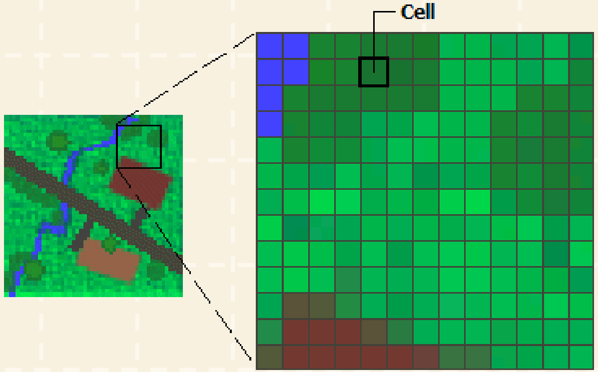


Geohash

Google's S2



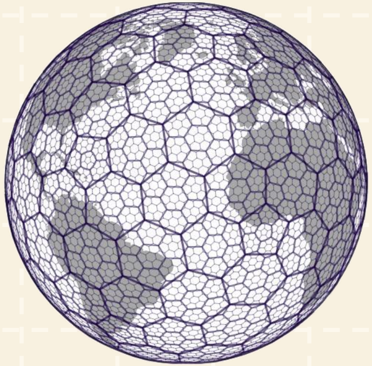
Hexbin

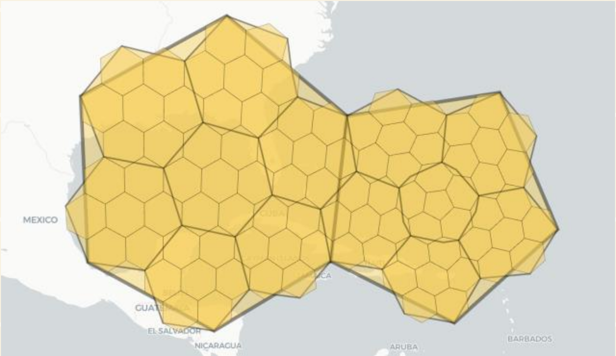


Raster

The Options

H3





Built initially for Uber, H3 combines a hexagonal grid system (icosahedron projection) with a hierarchical indexing system.

Utilizes hexagons for their unique mathematical properties.

The Options

H3

Distortion - No Shape Distortion, but slight area distortion.

Subdivision - Yes, but with rotation. Not complete containment.

Spatial Hierarchy - Yes, unique hierarchical index

Flexible Precision - 16 levels of precision.

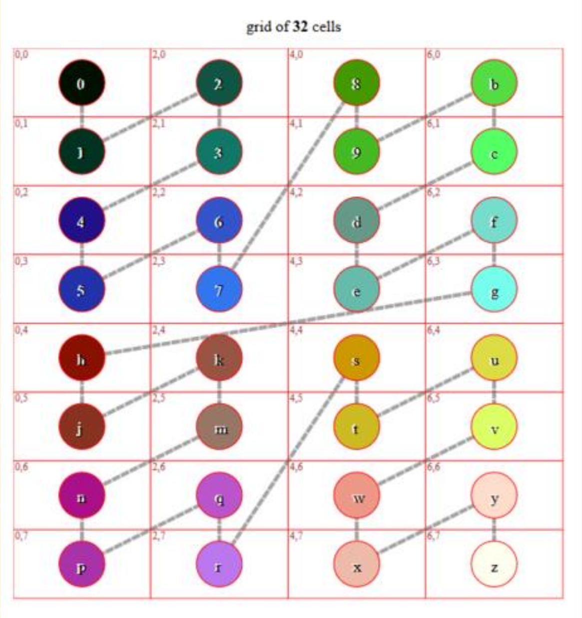
Neighbor Traversal - Best option. Many features for modeling flow.

Spatial Indexing - H3 cell indexes are designed to be 64 bit integers, which can be rendered and transmitted as strings if needed.

Interoperability - Well documented, integration supported (directly or via 3rd party products).

The Options

Geohash



Created in 2008 by Gustavo Niemeyer, Geohashing is a hierarchical data structure subdividing space into 32 identical rectangles.

It preserves spatial hierarchy in the code prefixes.

The Options

Geohash

Distortion - No significant shape distortion, but has area distortion.

Subdivision - Yes, but alternates between rectangle and square.

Spatial Hierarchy - Short alphanumeric string, with greater precision in longer strings.

Flexible Precision - 12 levels of precision.

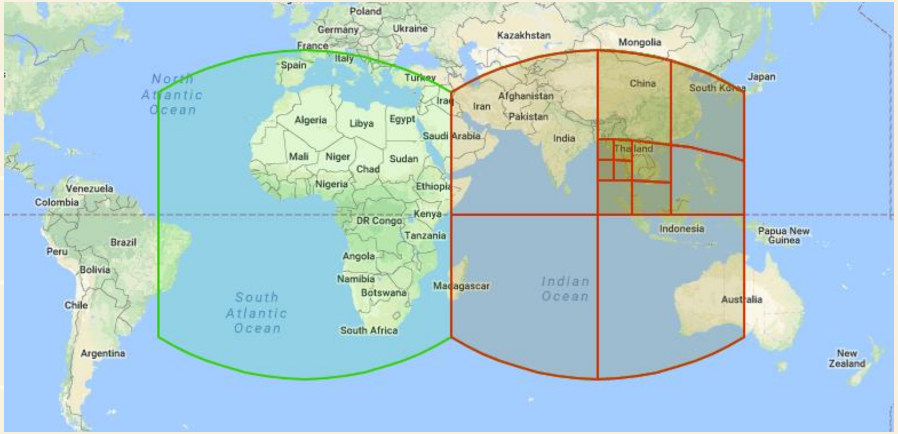
Neighbor Traversal - Indexing can be done in either Morton or Hilbert curve.

Spatial Indexing - Can be treated as a string, very fast operations.

Interoperability - Libraries available. Not the most ubiquitous.

The Options

S2



Created by Google, for defining a discrete global grid scheme.
S2 represents all the data on a three dimensional spherical projection.

The cell edges appear to be curved, because they are spherical geodesics.

The Options

S2

Distortion - Visually distorted and area distortion.

Subdivision - Each cell is subdivided into four.

Spatial Hierarchy - Intended to maximize locality of reference.

Flexible Precision - 30 levels of precision.

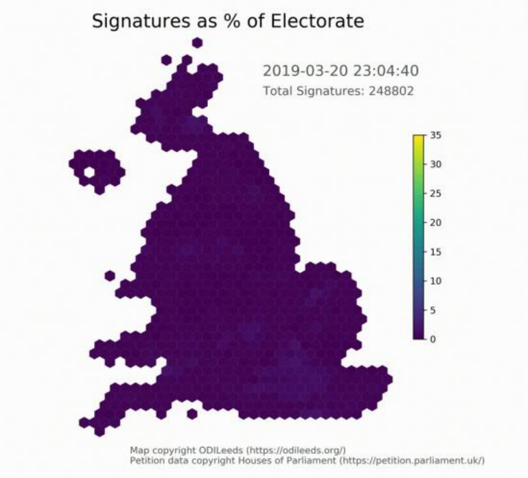
Neighbor Traversal - Different types of neighbors.

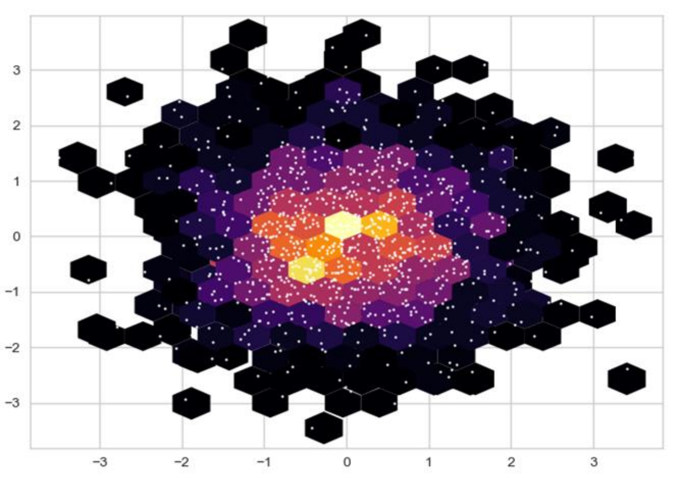
Spatial Indexing - Fast.

Interoperability - Well developed libraries.

The Options

Hexbin





Hexbins are mostly used for aggregation visualisation.
User defines the cell radius.

The Options

Hexbin

Distortion - User defines CRS. Generated per use-case.

Subdivision - No subdivision.

Spatial Hierarchy - No Hierarchies.

Flexible Precision - Flexible size.

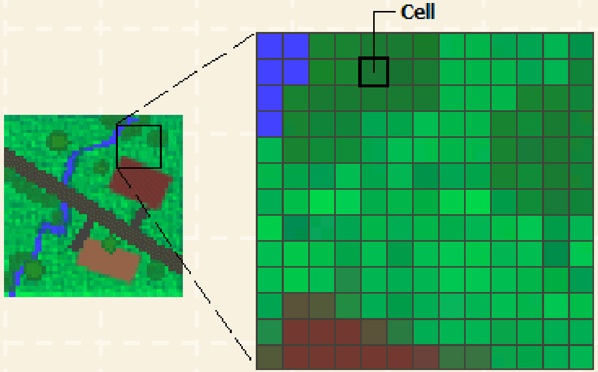
Neighbor Traversal - Hexagons (equidistant).

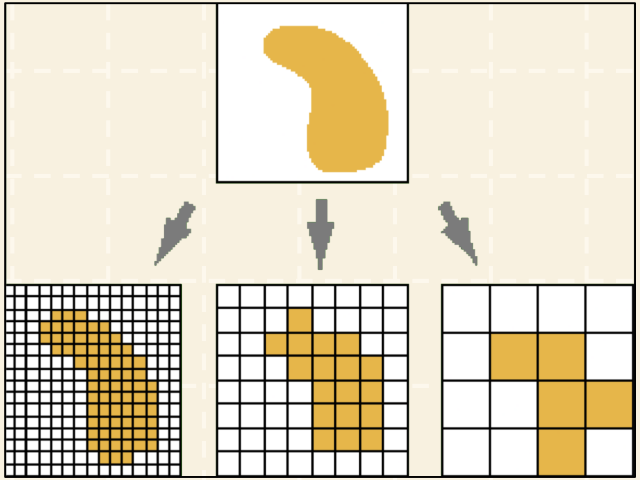
Spatial Indexing - Good locally. Complex Globally.

Interoperability - Classic method but not used in geospatial databases.

The Options

Raster





Raster data consists of a matrix of cells into a grid.
Used within many classic GIS software.

The Options

Raster

Distortion - User defines CRS. Generated per use-case.

Subdivision - No subdivision.

Spatial Hierarchy - No Hierarchies.

Flexible Precision - Flexible size.

Neighbor Traversal - Not the best, squares.

Spatial Indexing - Good locally. Complex Globally.

Interoperability - Used in a lot of classic software.

4

DockTech's Choice

Our considerations.

Cell Aggregation in Ports

- Taking 30 days of data in an active port, we could extract over 20 million observations.

  Querying speed and storage costs are important.

- We want to perform statistical analysis on all points within a cell (create a predicted depth on a cell level).
- Various stakeholders to interact with the data.
- Needs sub-division for parallelization.
- Different parts of the port “require” different levels of precision.

Cell Aggregation in Ports

Distortion - Ports are small enough, that slight distortion is ok. Don't want visual distortion.

Subdivision - Very important for spatial hierarchy and for visualisation.

Spatial Hierarchy - Very valuable for partitioning.

Flexible Precision - Preferred. A defined precision that approximates needs of ports + GPS accuracy works as a good proxy.

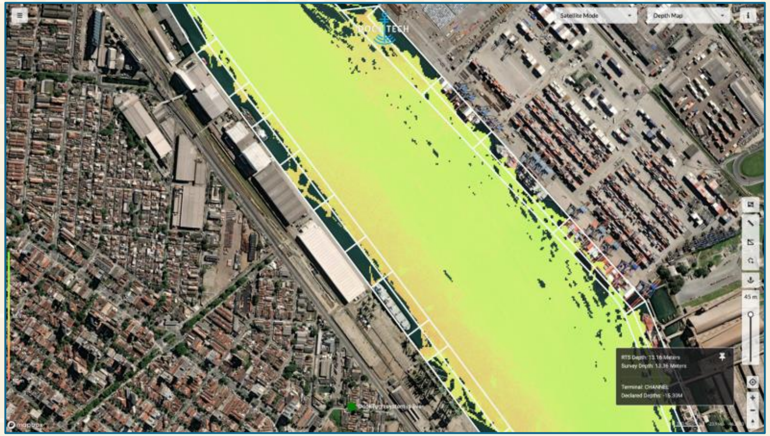
Neighbor Traversal - Neighbour traversal isn't so relevant, but exploring neighboring cells is (geospatial features).

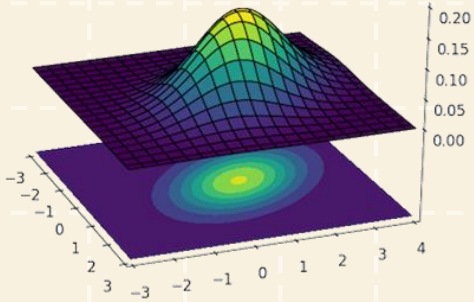
Spatial Indexing - Important. Querying needs to be fast.

Interoperability - Don't want to re-invent the wheel.

Geohash

Geohash length	Cell width	Cell height
1	≤ 5,000km ×	5,000km
2	≤ 1,250km ×	625km
3	≤ 156km ×	156km
4	≤ 39.1km ×	19.5km
5	≤ 4.89km ×	4.89km
6	≤ 1.22km ×	0.61km
7	≤ 153m ×	153m
8	≤ 38.2m ×	19.1m
9	≤ 4.77m ×	4.77m
10	≤ 1.19m ×	0.596m
11	≤ 149mm ×	149mm
12	≤ 37.2mm ×	18.6mm





At precision 9, we get an approximately 5mx5m cell. For ports, at that level of detail the accuracy is good enough.

To detail with the equidistance problem, we can weigh each observation by their Inverse Distance.

Being able to treat the feature as a string, querying time is extremely fast.

Future Work

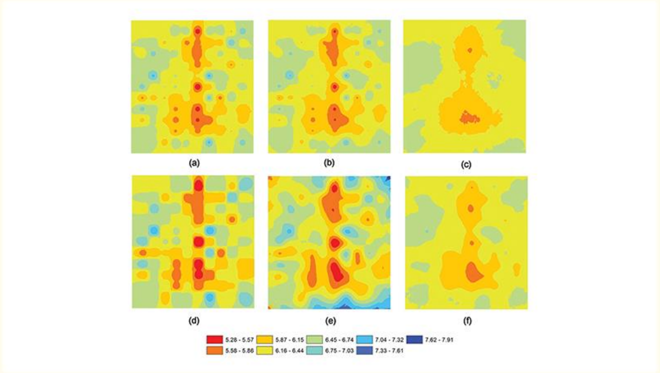
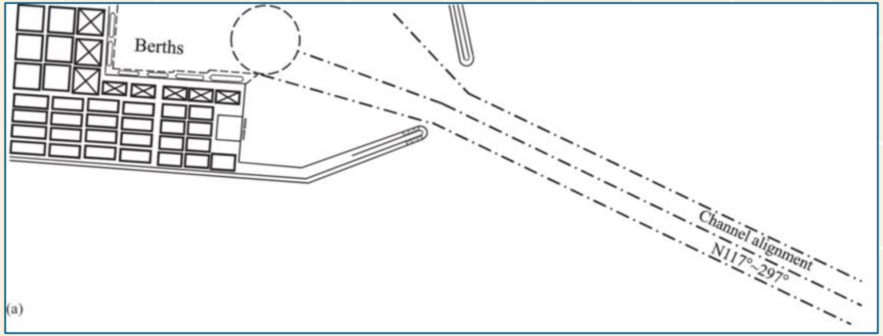


Figure 2.
Six interpolated surfaces with their parameters. N: neighbor. (a) IDW, power 2, N 10; (b) spline, tension, N 10; (c) kriging, circular, N 10; (d) IDW, power 4, N 20; (e) spline, thin plate, N 20; (f) kriging, exponential, N 20.



- Combine Geohash and Raster, generating user-defined cell sizes.
- Using geohashes to define cell centers, but search radius is circular.
- Combine various aggregation methods at different stages.

Created by Shaul Solomon
Shauls@docktech.net

For more info: https://www.docktech.net/

Resources

- https://h3geo.org/docs/
- https://desktop.arcgis.com/en/arcmap/10.3/manage-data/raster-and-images/what-is-raster-data.htm
- http://postgis.net/workshops/postgis-intro/indexing.html
- https://gistbok.ucgis.org/bok-topics/spatial-indexing
- https://think.design/services/data-visualization-data-design/hexbin/
- https://spatialthoughts.com/2020/07/01/point-in-polygon-h3-geopandas/
- https://s2geometry.io/
- https://www.intechopen.com/chapters/52704