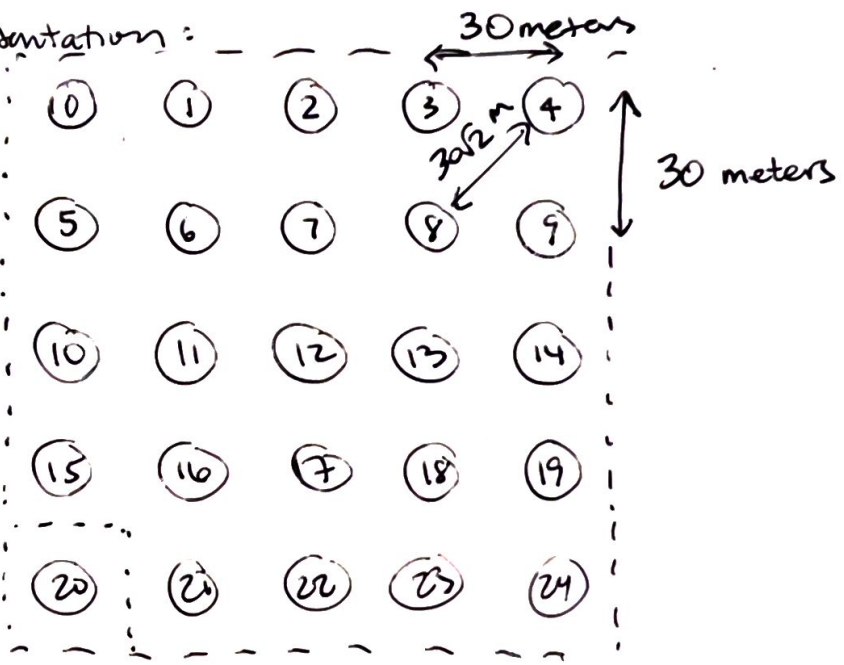
Cesium.: Mount St. Helen's Project 3/8/23

Input: • 512×512 pixel height data (as unsigned 8-bit value)
↳ 1 = 11 meters
before and after maps
• $(x_1, y_1) = A$
$(x_2, y_2) = B$ } locations in pixel coordinates

Output: • distance in meters along surface from A → B
• PRINT: print the difference in surface distance for the before + after eruption maps!

Data Representation:



Ideal: treat pixels as cell centers
→ Have 15360 m × 15360 m grid
with 30 m wide buckets

★ Read data into 1-D array of size $(512)^2 = 262144$
→ make mapping for row-major 2D → 1D

★ Then generate a series of points from A + B
using segments of length ...? $r_p = 30\sqrt{2}$!

Idea: Use spherically symmetric cubic kernel to obtain smooth height field! 3/8/23

→ this requires a neighbor search per particle that we query the height at (point)

Kernel: from Homel and Herbold 2016

$$\omega(\bar{r}) = \begin{cases} 1 - 3\bar{r}^2 + 2\bar{r}^3 & : 0 \le \bar{r} \le 1 \\ 0 & : \bar{r} > 1 \end{cases}$$

$$\bar{r} = \|x - x_p\| / r_p$$

$\bar{r}$ = normalized distance from query point

r_p = neighbor radius

In this grid, $\Delta x = 30$ m

↳ $r_p = \sqrt{30^2 + 30^2} = 30\sqrt{2}$ meters

Normalize this field by dividing kernel sum to remove edge effects

Normalized Height Field

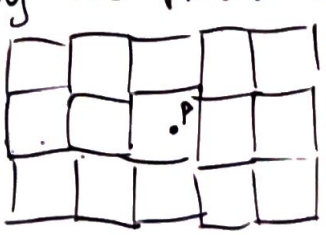
$$\bar{H}(x) = \frac{H(x)}{S(x)} = \frac{\sum_{p=1}^{n_p} h_p \, \omega(\bar{r})}{\sum_{p=1}^{n_p} \omega(\bar{r})}$$

(n_p = num neighbors; h_p = pixel height (pixels are our quadrature to compute height field))

How to simplify neighbor search?

$r_p = 30\sqrt{2}$, outside this range, no other particles (pixels) will have influence

For each point we query, put it in the pixel bucket, then know that any given point is only influenced by the pixels at i, $i+1$, $i+2$!



5x5 pixels stencil! (this is just from this specific kernel)

So, the pseudocode! 3/8/23

- Read input data for each map
  - ↳ 2 1-D arrays of pixel data
- Generate series of query points from A→B at least V_p apart ($30\sqrt{2}$ m).
- for each line segment defined by these points, Compute start and end height and Compute Euclidean distance between these points!