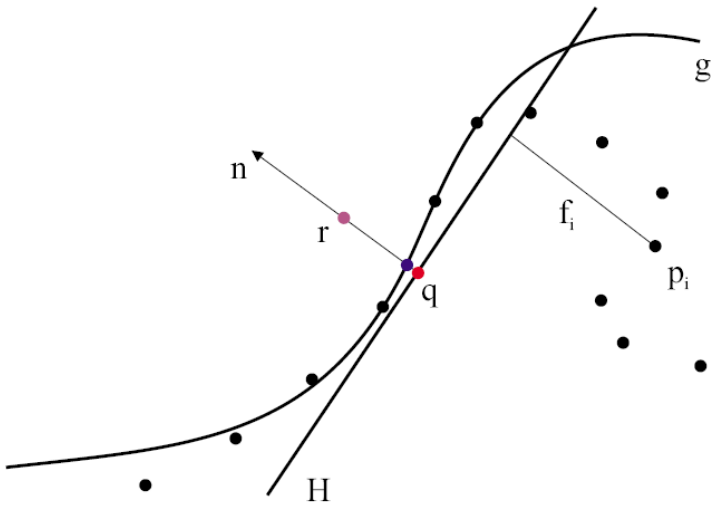
‘Point Set Surfaces’ implementation 2

Yi Yong-il

Projection procedure

- Reference domain
  - Find a local reference plane ‘H’ for ‘r’
  - Minimize a local weighted sum of square distances of ‘p’ to the plane
- Local map
  - Compute a local bivariate polynomial approximation to the surface

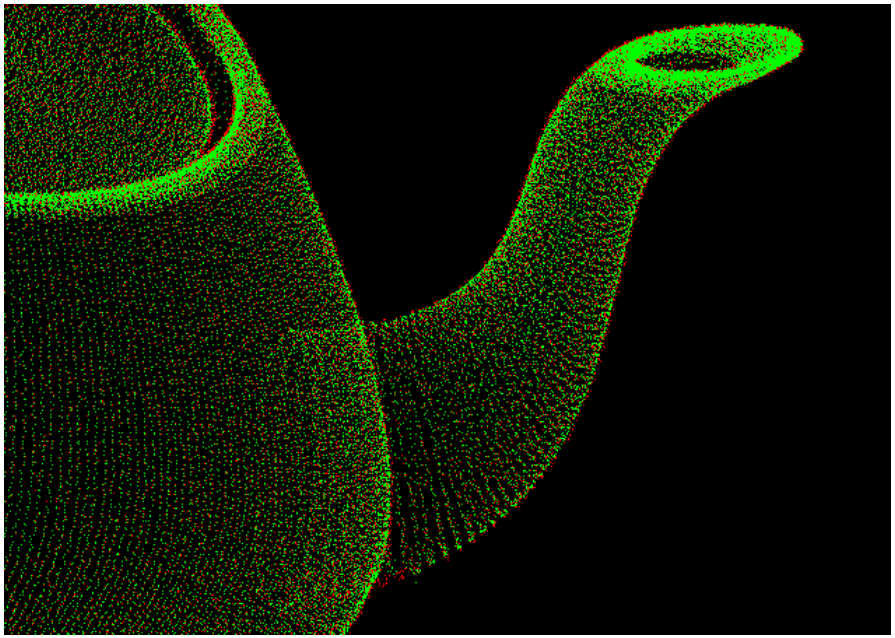


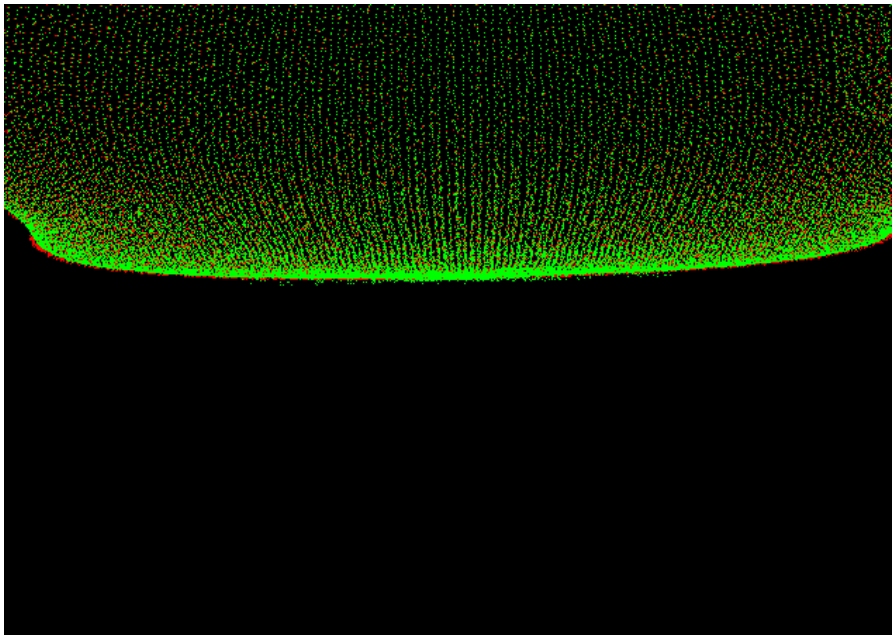
Revision – reference domain

- minimizatio $\sum_{i=1}^{N} \langle n, p_i - r - tn \rangle^2 \theta(\|p_i - r - tn\|)$ $\min_{\|n\|=1} n^T B n$
- At first, get the Eigenvector of $b_{jk} = \sum_i \theta_i (p_{i_j} - r_j)(p_{i_k} - r_k)$. that corresponds to the smallest Eigenvalue
  - We can get initial normal ‘n’
- And then get the ‘t’ for the initial normal ‘n’
  - ‘t’ is in –H/2 ~ H/2
- At second, execute powell iteration for ‘n’
  - ‘t’ is fixed
  - There are many local minimum. We have to find the one that has smallest ‘t’

Result

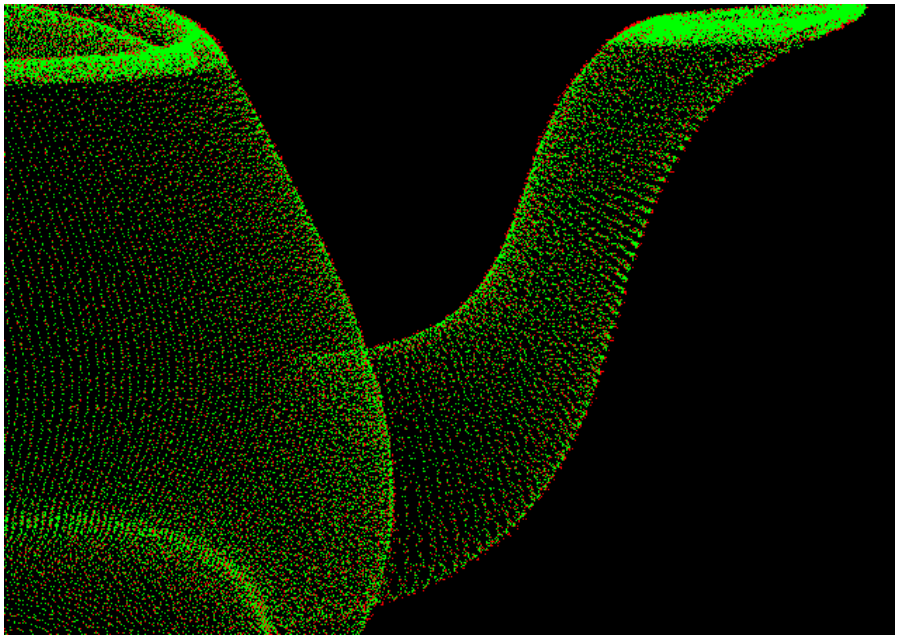
- Old version, teapot(131330 vertices), 80 nearest points
  - About 174 secs
  - There are some errors

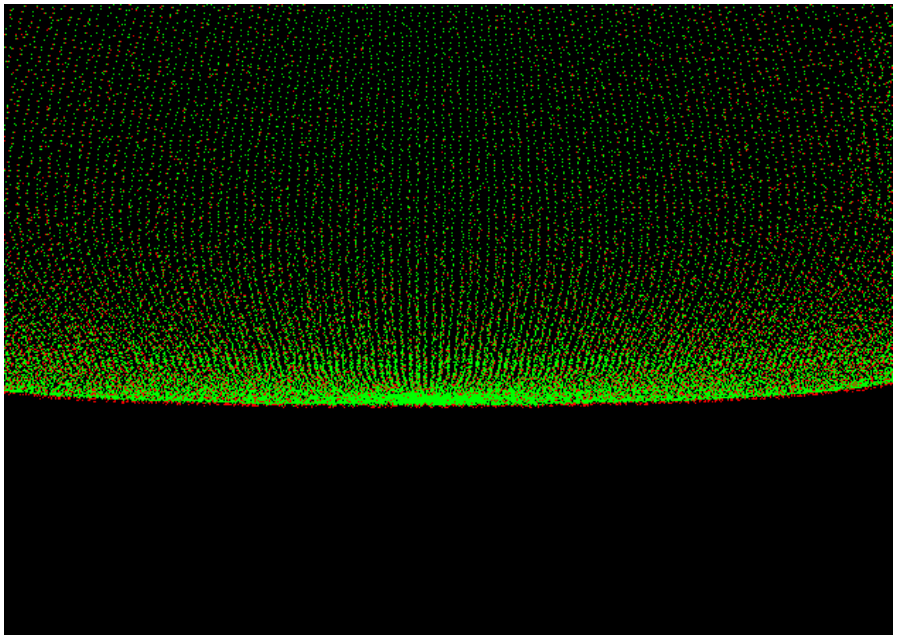




Result

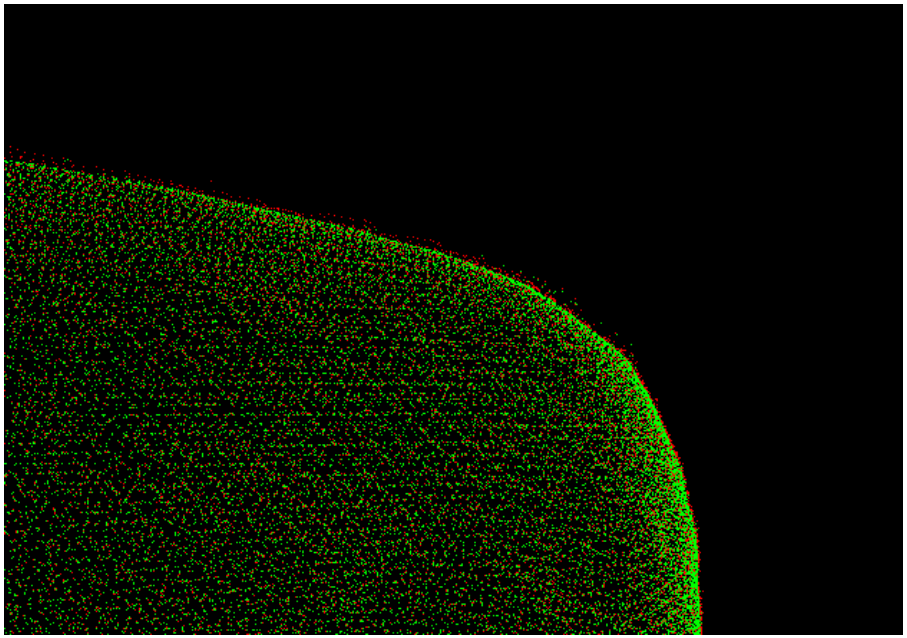
- New version, teapot(131330 vertices), 80 nearest points
  - About 99 secs => much faster
  - Better quality

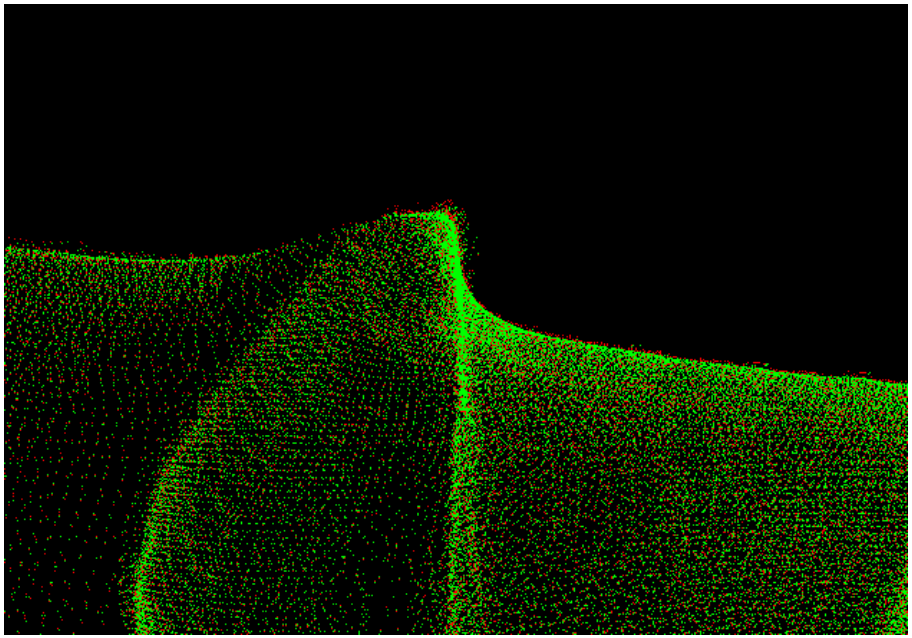




Result

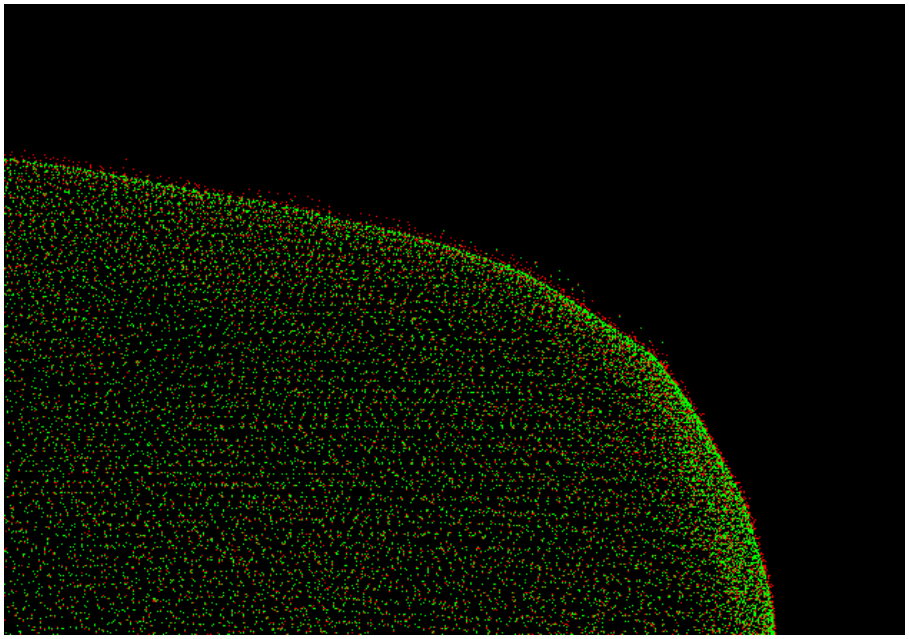
- Old version, too noisy model(167424 vertices), 80 nearest points
  - About 260 secs

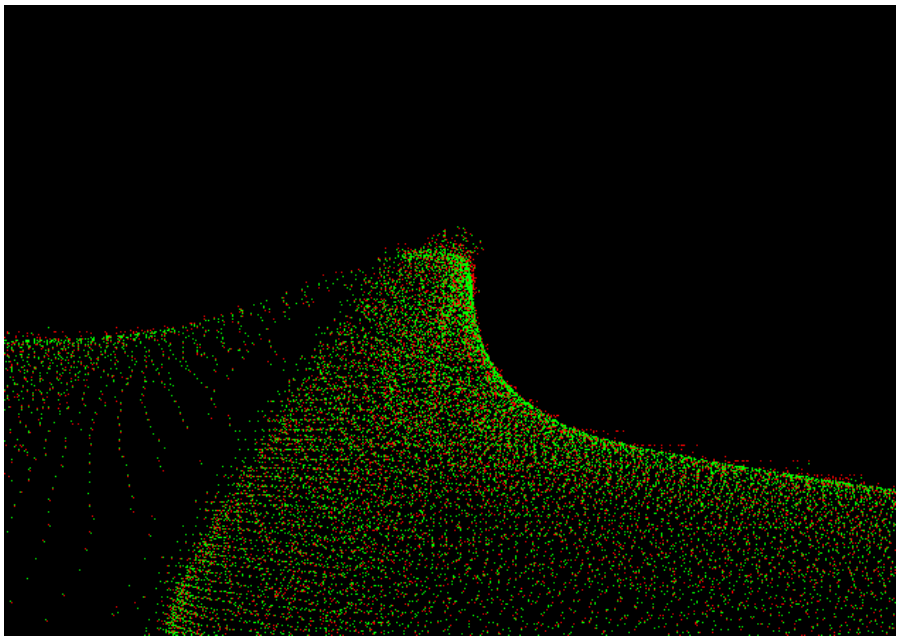




Result

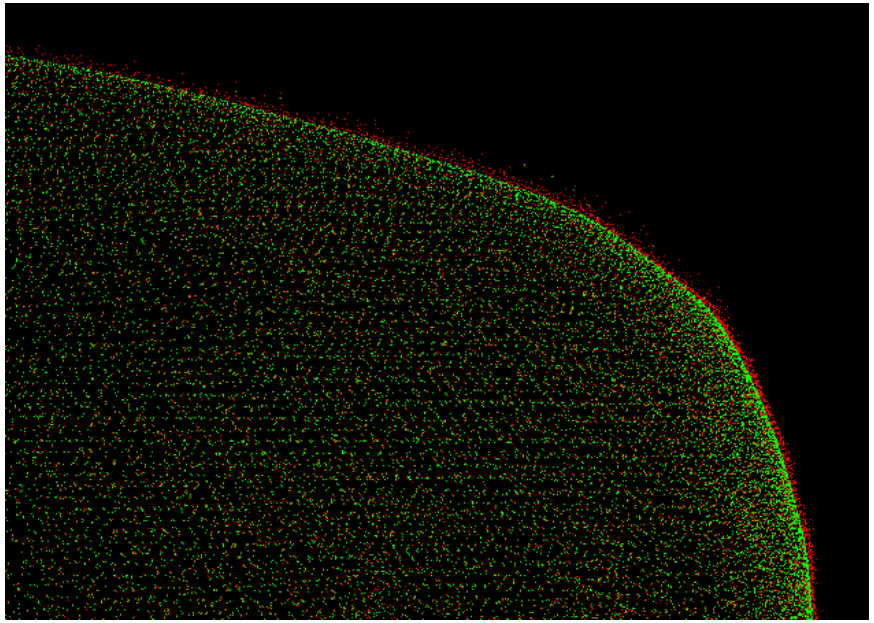
- New version, too noisy model(167424 vertices), 80 nearest points
  - About 119 secs, better quality

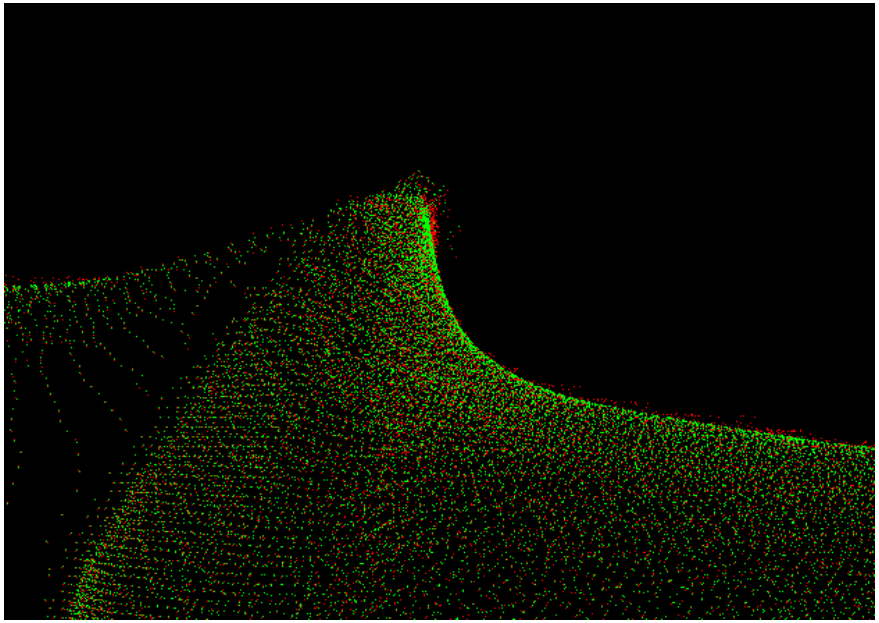




Result

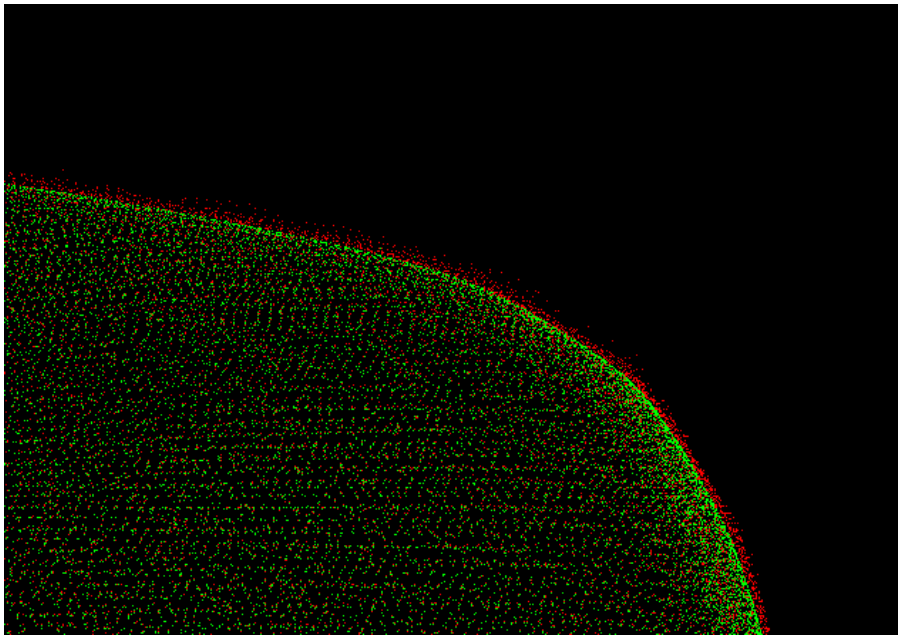
- New version, too noisy model(167424 vertices), 400 nearest points
  - For better quality, it needs more nearest points
  - about 645 secs

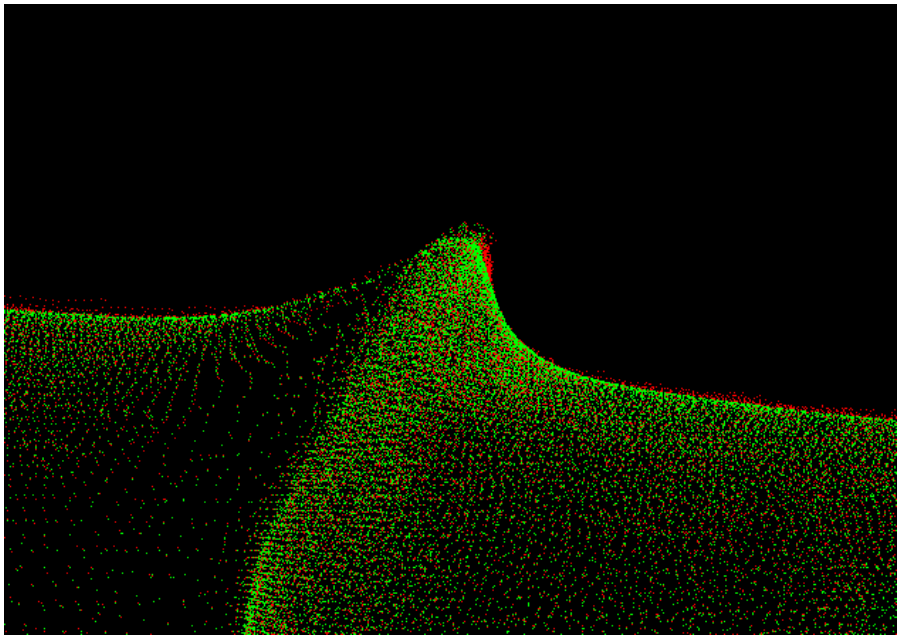




Result

- New version, too noisy model(167424 vertices), 400 nearest points, larger H value(1.0)
  - Needs to adjust several parameters
  - About 632 secs





Further work

- New data structure
  - Use grid
    - Neglect distance ‘d’ => distance that weight function becomes zero
    - Grid cell size is 2*d, so a maximum 8 cells is needed
  - Each cell is organized as an Octree
    - Leaf nodes contain nearest points ‘p’
    - Inner nodes contain some information
      - Number of points in the subtree
      - Centroid of the subtree
      - If the nodes’s dimension is much smaller than its distance to ‘r’, the centroid is used for computing the coefficients
      - Whole nodes can be neglected if the distance is larger than ‘d’
  - when you get the nearest points, include ‘r’ itself