Detecting Changes and Finding Collisions in 3D Point Clouds

Data Structures and Algorithms for Post-Processing Large Datasets

Johannes Schauer

Structure of this talk

1) Initial Problem Statement

2) Three Demos

3) Eight publications: from collision detection to change detection

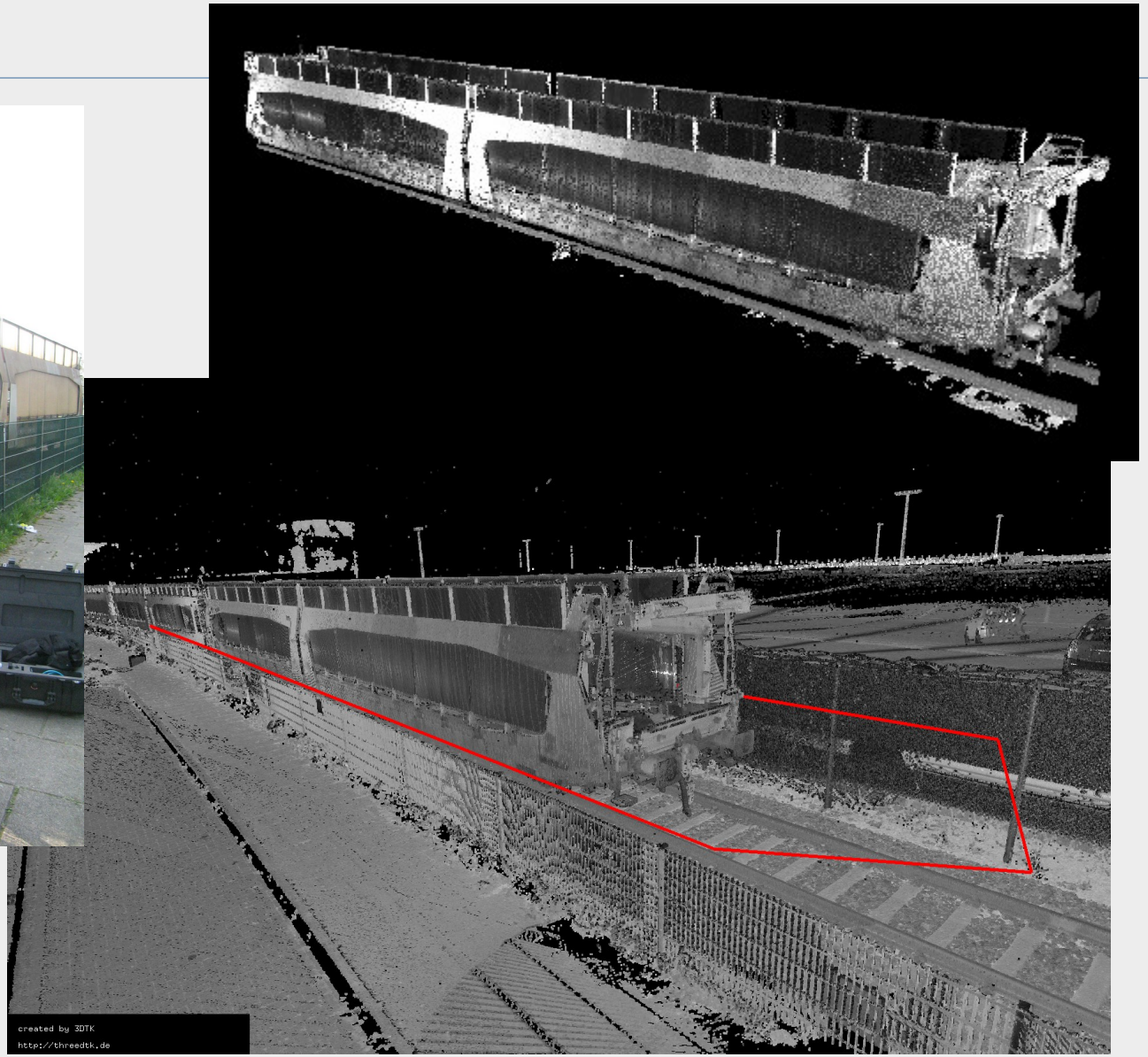
4) Future Work

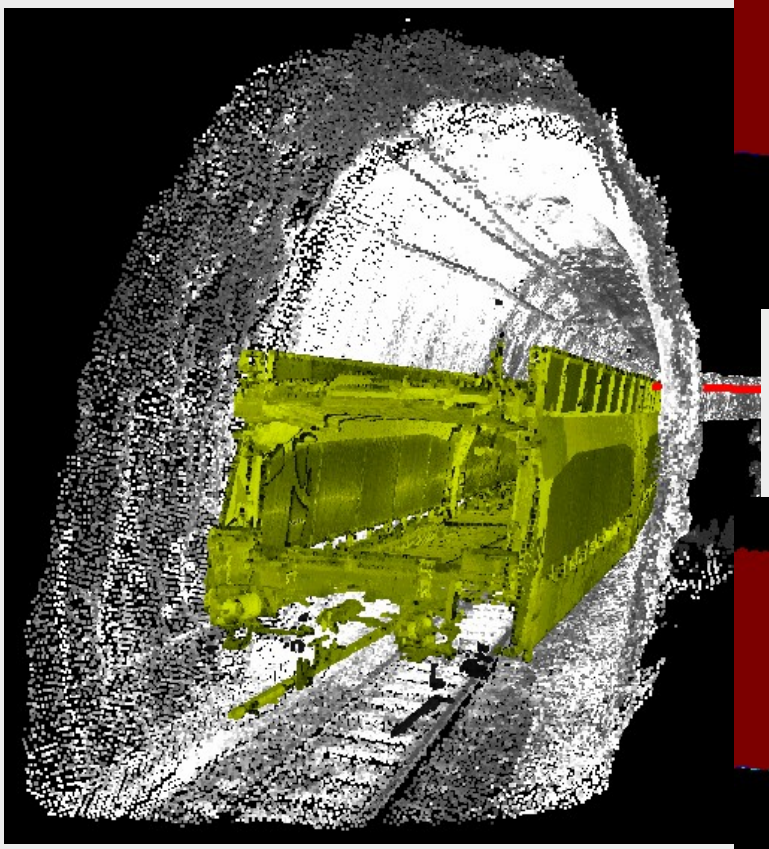
Initial Problem Statement

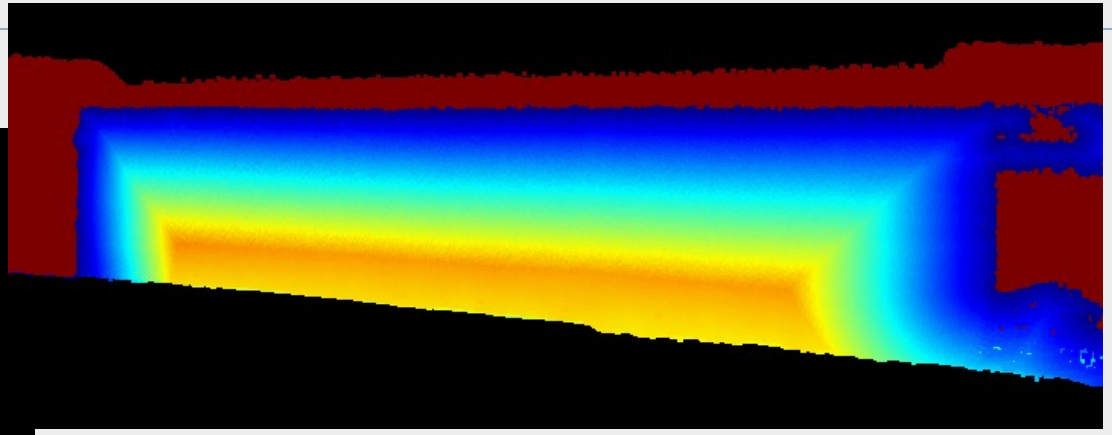
- Does the car body fit through the factory?
- Does the trailer fit through the urban environment?
- Does the mining equipment fit through the tunnel?
- What changes are required?
- What about moving objects?

Three Demos

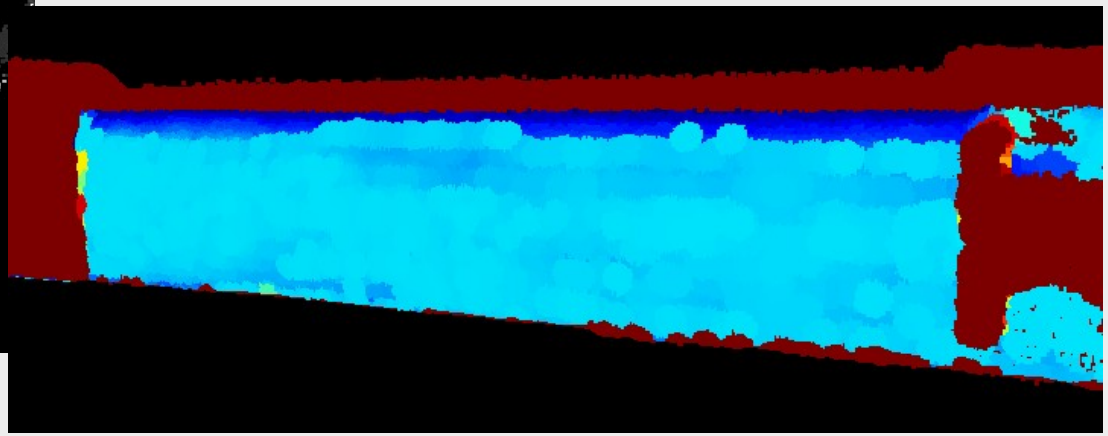
Schauer, J., Nüchter, A.: **Efficient Point Cloud Collision Detection and Analysis in a Tunnel Environment using Kinematic Laser Scanning and k-d Tree Search.** Proceedings of the Photogrammetric Computer Vision (PCV '14). p. 289--295. , Zürich, Switzerland (2014).


created by 3DTK
http://threedtk.de



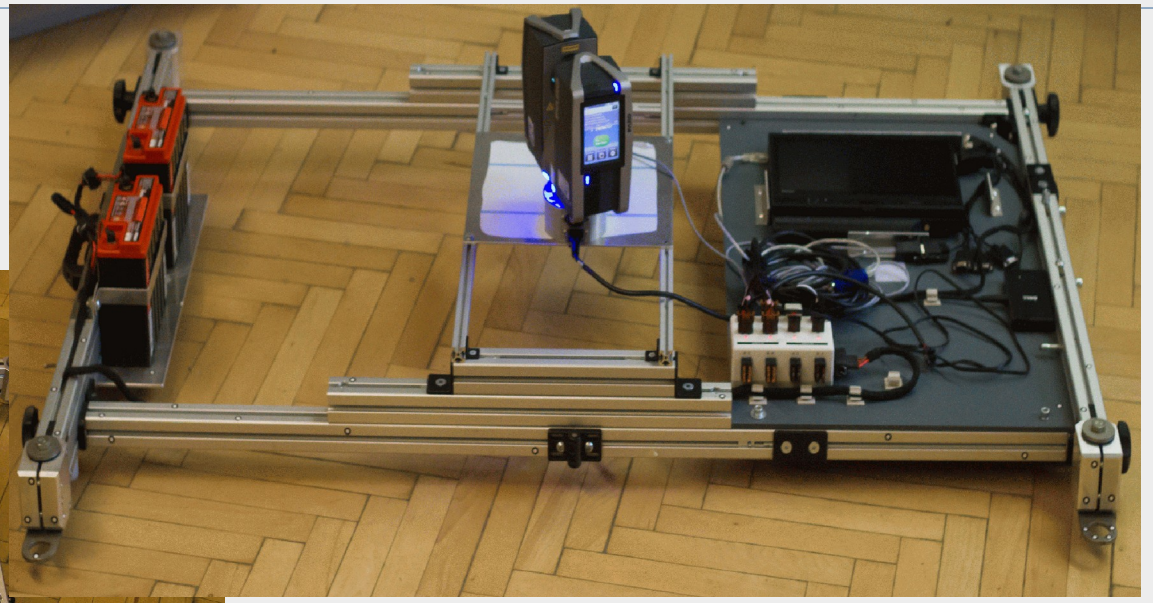


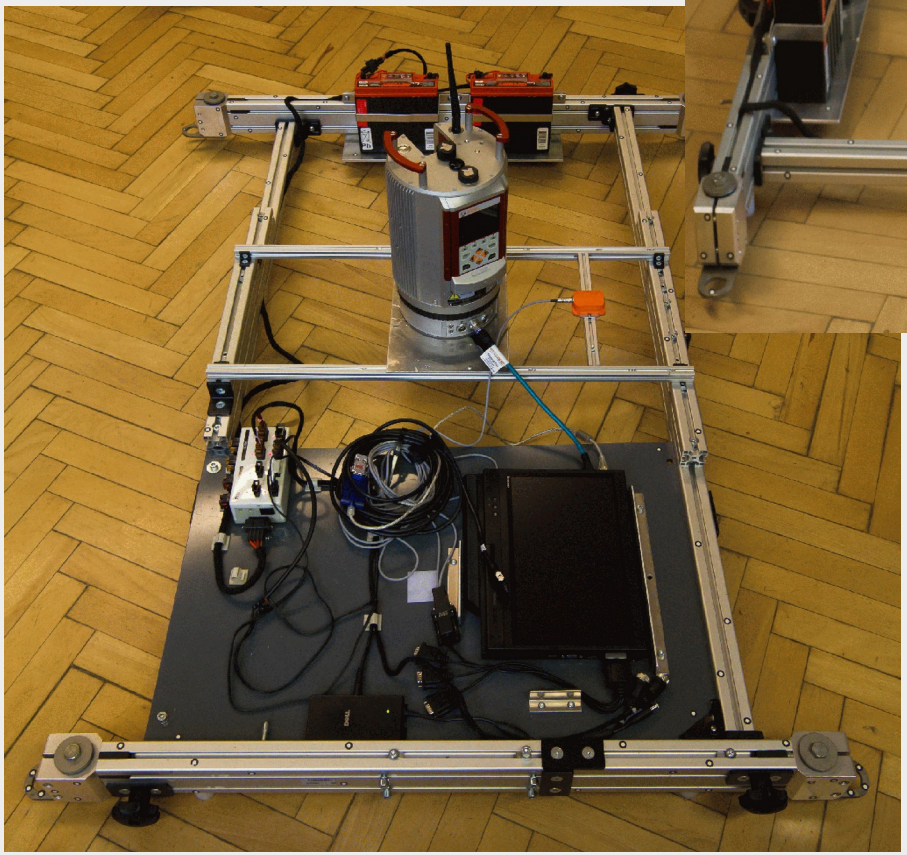
Algorithm: kd-PD-fast



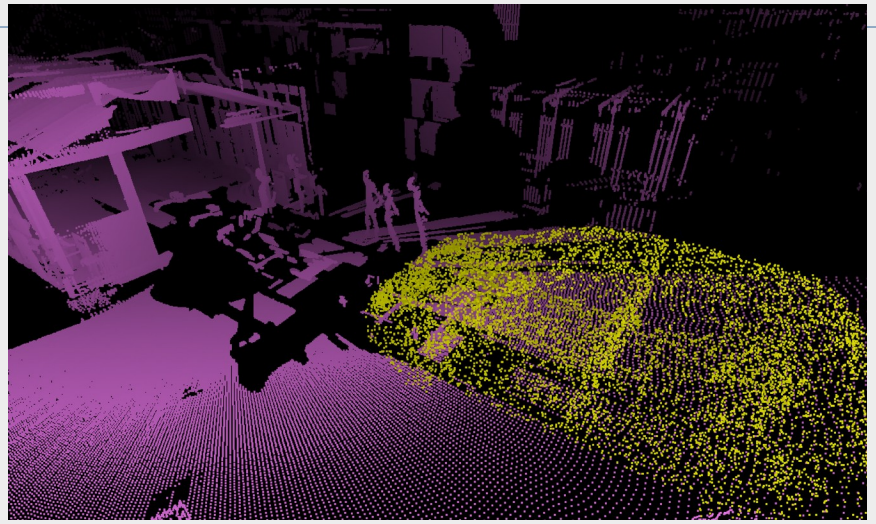
Algorithm: kd-PD

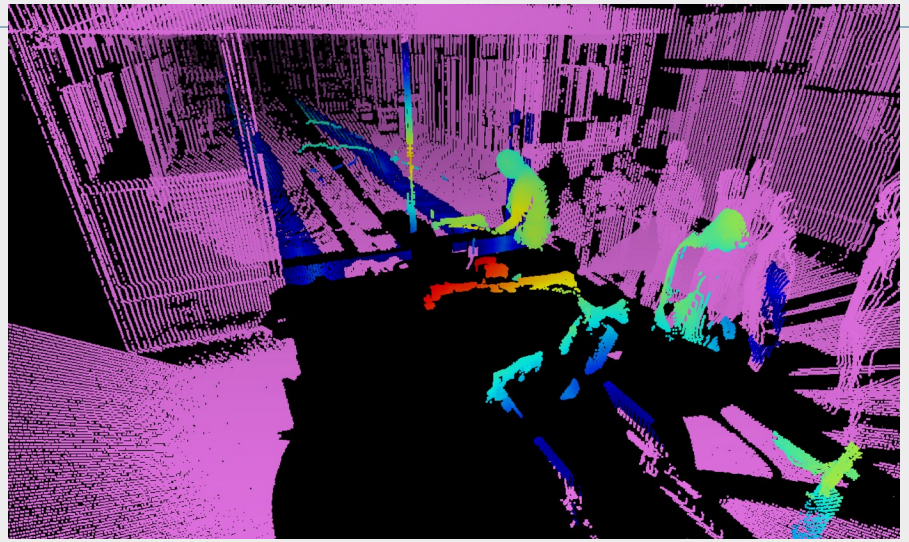
Sensor Skid

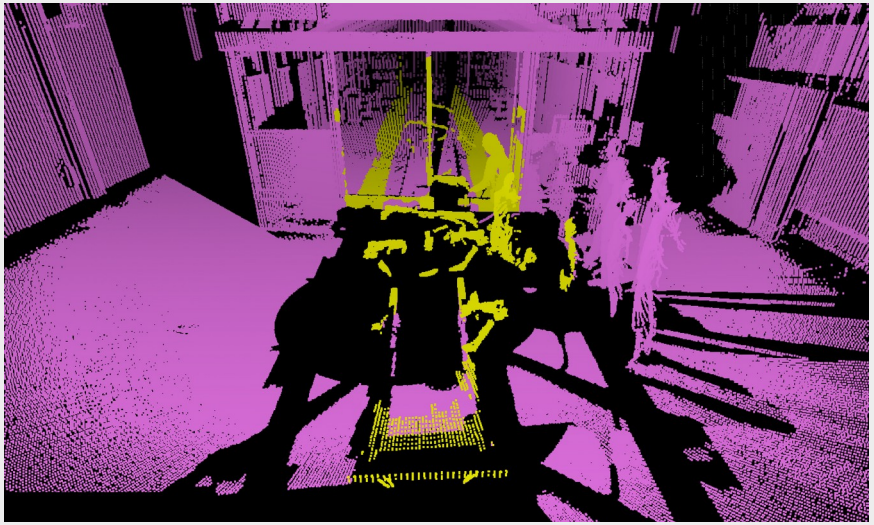


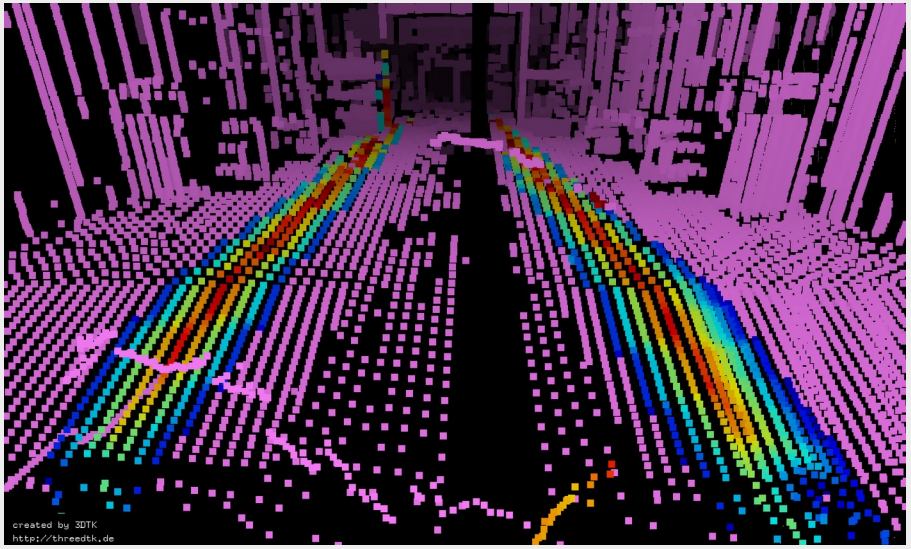


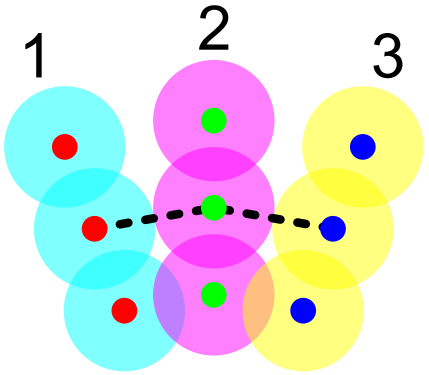
Elseberg, J., Borrmann, D., Schauer, J., Nüchter, A., Koriath, D., Rautenberg, U.: **A sensor skid for precise 3D modeling of production lines.** Proceedings of the Commision V Symposium Close-range imaging, ranging and applications. p. 117--122. , Riva del Garda, Italy (2014).








created by 3DTK
http://threedtk.de

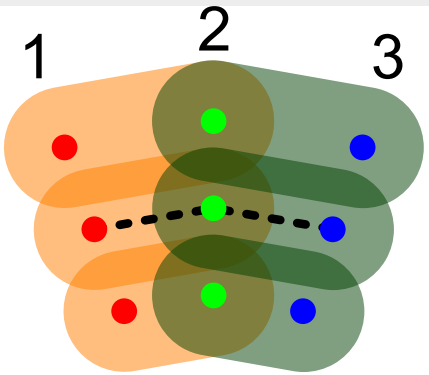


search areas (at pos. 1)

search areas (at pos. 2)

search areas (at pos. 3)

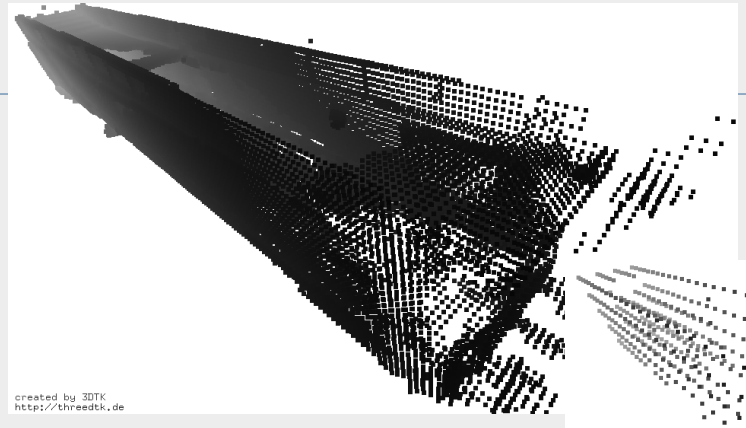
kd-CD-simple

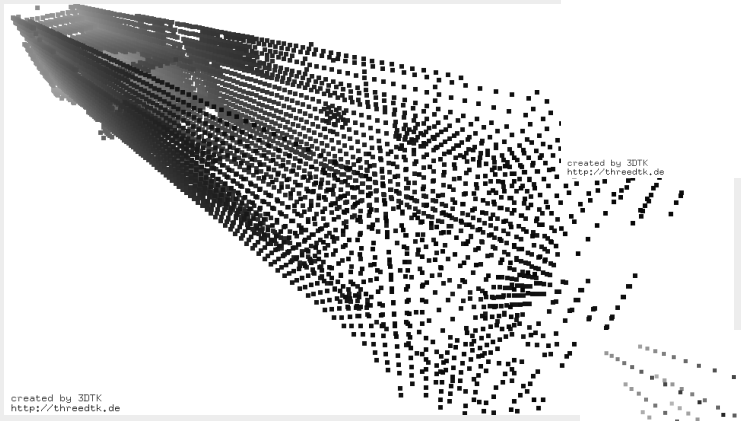


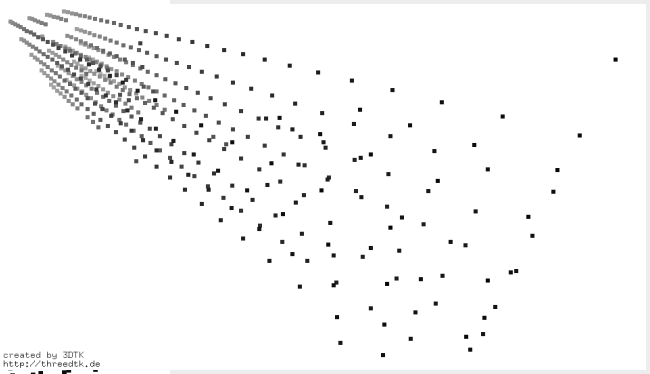
search areas (1st step)

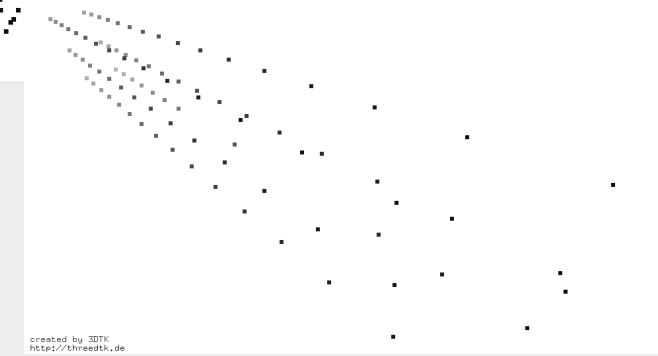
search areas (2nd step)

kd-CD



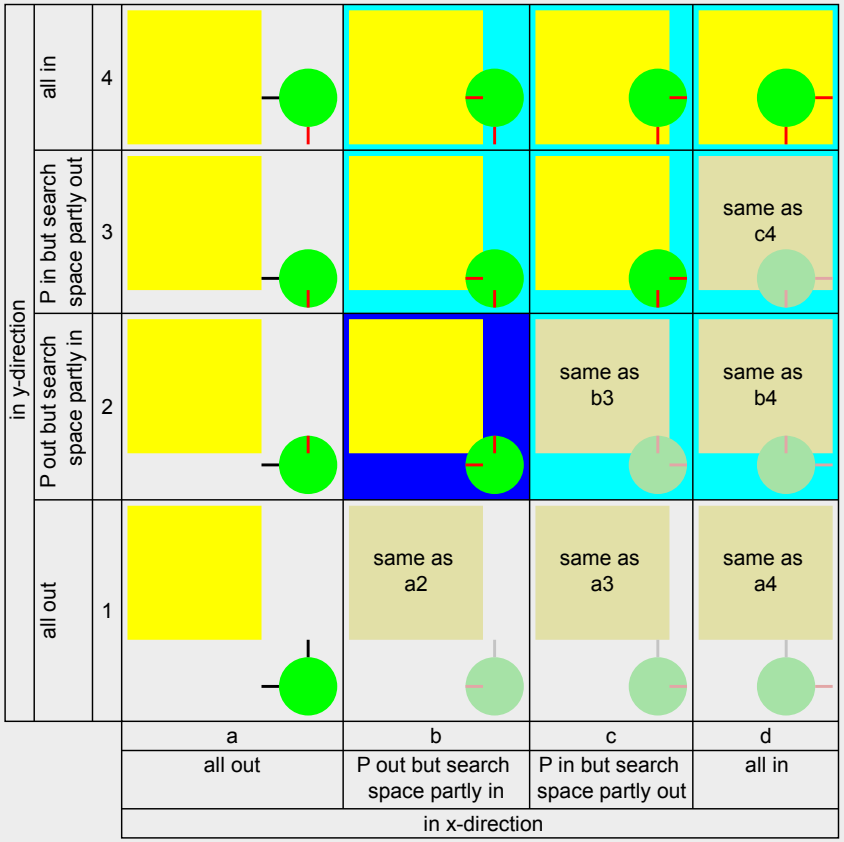


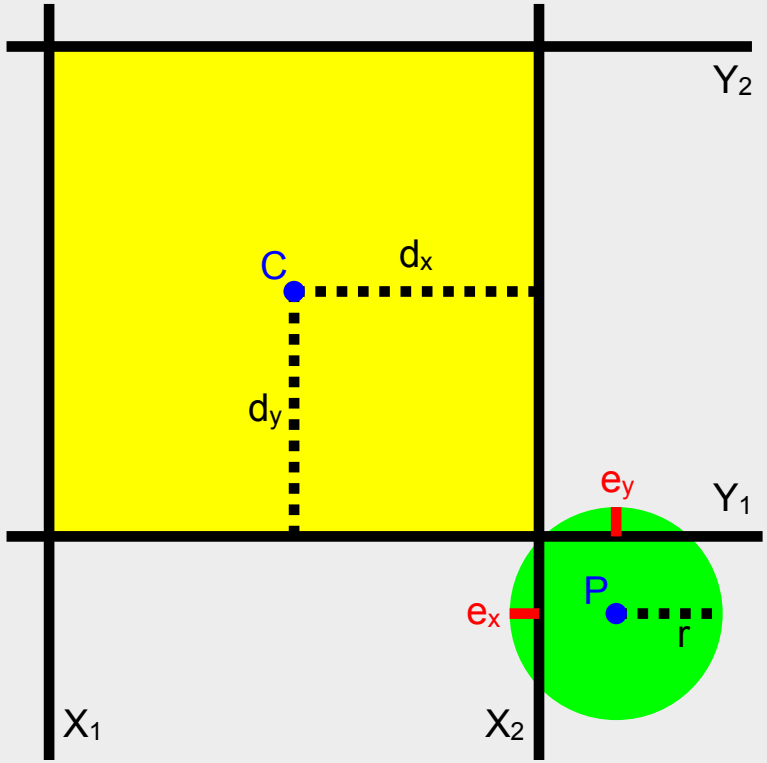




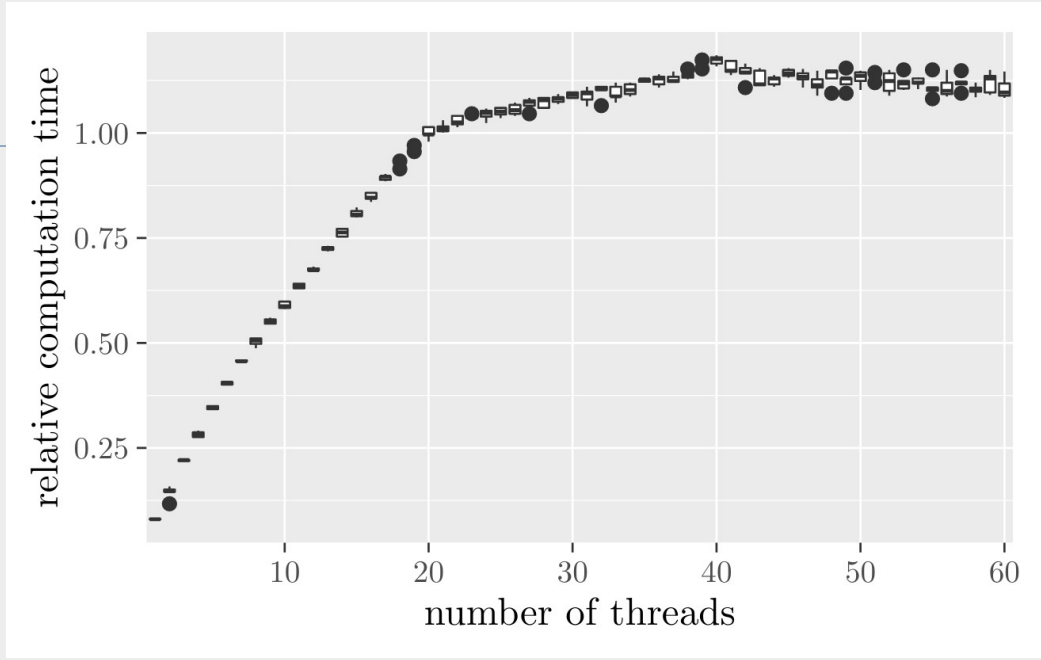
Schauer, J., Nüchter, A.: **Collision detection between point clouds using an efficient *k*-d tree implementation.** Journal Advanced Engineering Informatics (JAdvEI). 29, 440--458 (2015).

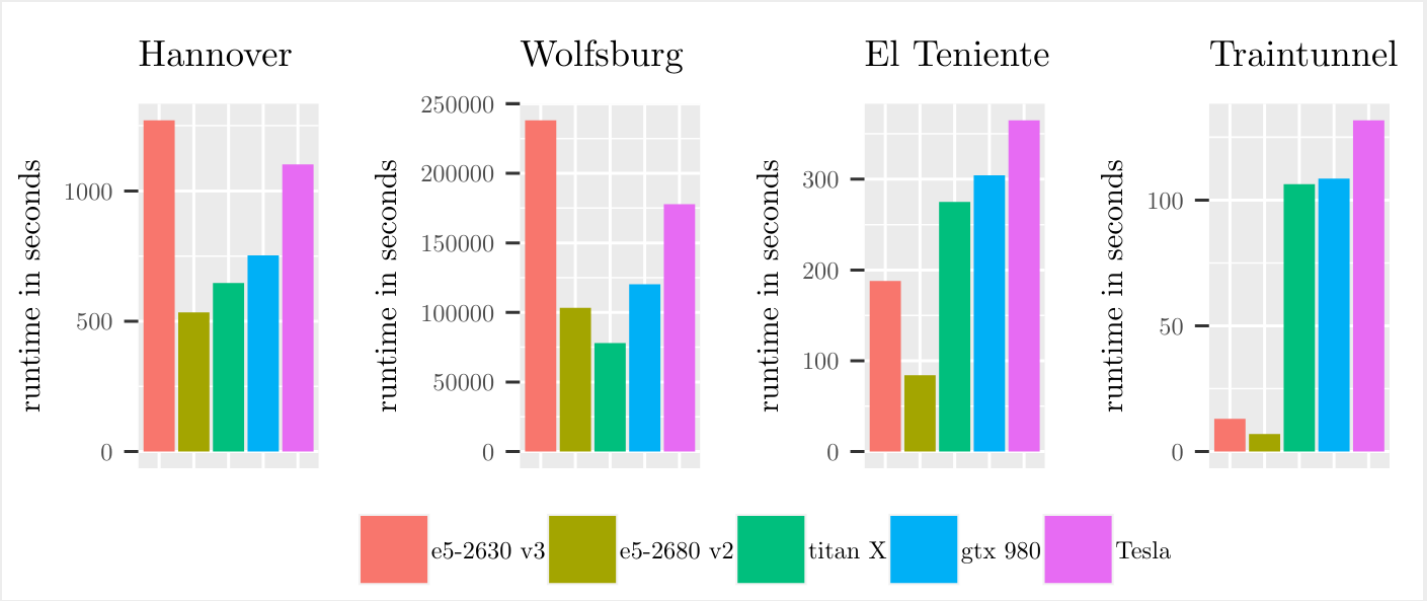
```
14   // quick test whether subtree has to be searched
15   double approx_dist_bbox =
16     max(max(fabs(params[threadNum].p[0]-node.center[0])-node.dx,
17     fabs(params[threadNum].p[1]-node.center[1])-node.dy),
18     fabs(params[threadNum].p[2]-node.center[2])-node.dz);
19   if (approx_dist_bbox >= 0 && sqr(approx_dist_bbox)
20       >= params[threadNum].maxdist_d2) return;
```





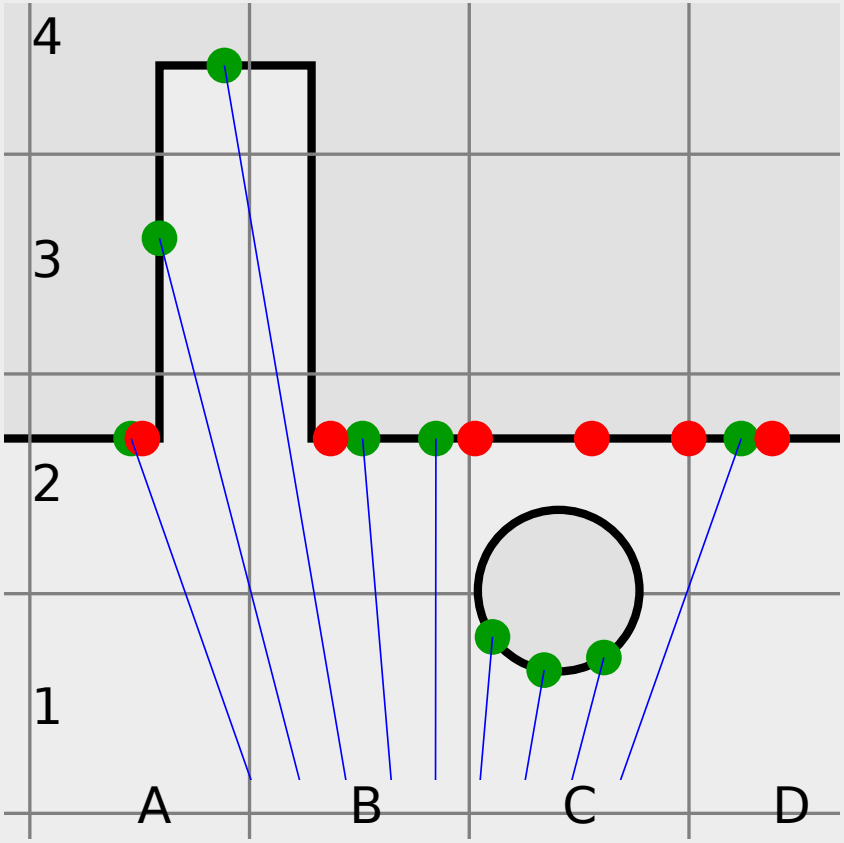
Schauer, J., Bedkowski, J., Majek, K., Nüchter, A.:
Performance comparison between state-of-the-art point-cloud based collision detection approaches on the CPU and GPU. Proceedings of the 4th IFAC Symposium on Telematics Applications (TA '13). p. 54--59. , Porto Alegre, Brazil (2016).



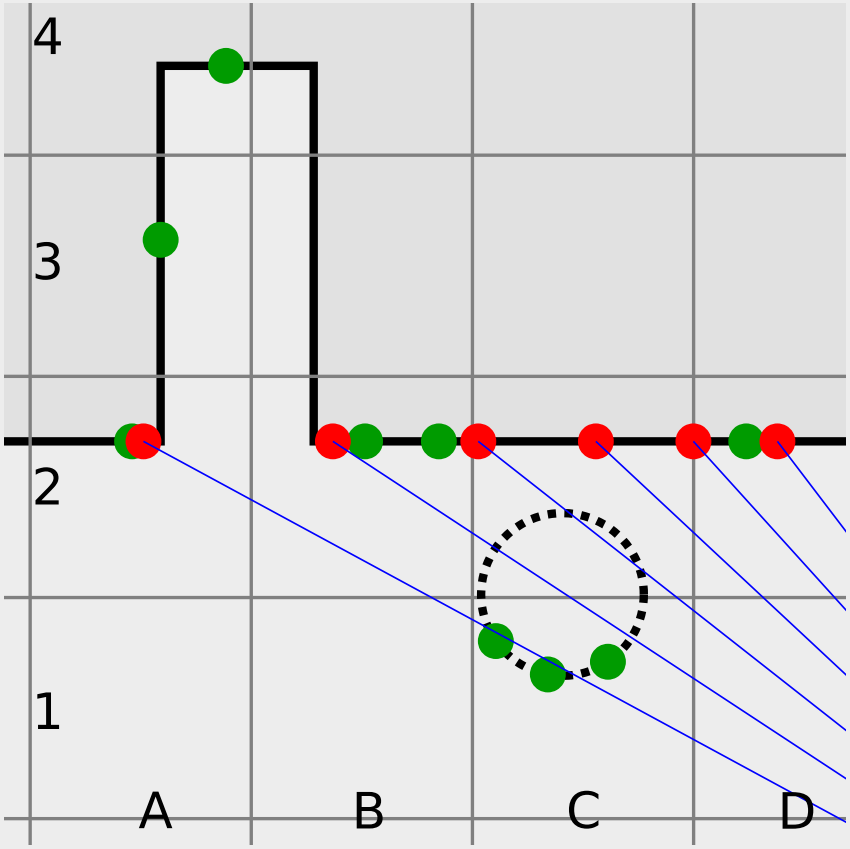


Schauer, J., Nüchter, A.: **Digitizing automotive production lines without interrupting assembly operations through an automatic voxel-based removal of moving objects.** Proceedings of the 13th IEEE International Conference on Control and Automation (ICCA '17). p. 701--706. , Ohrid, Macedonia (2017).

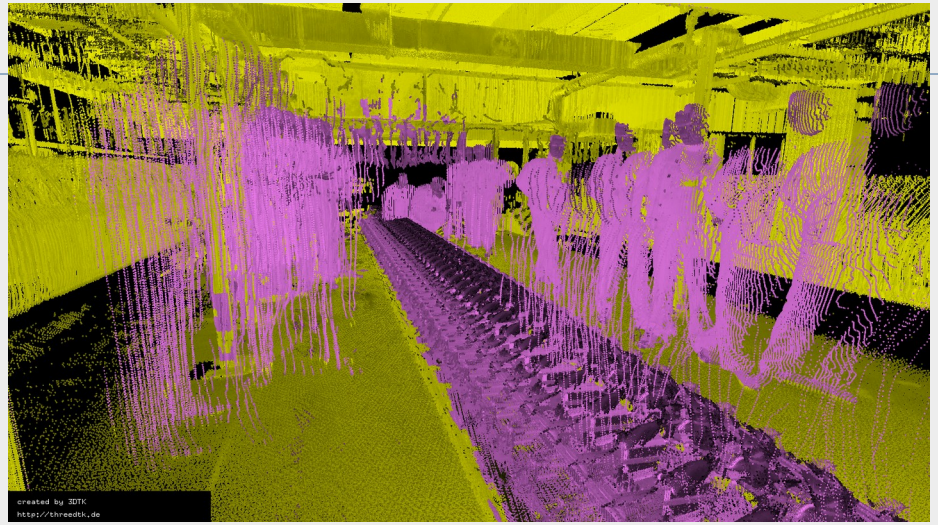
Voxel-based change detection in a nutshell

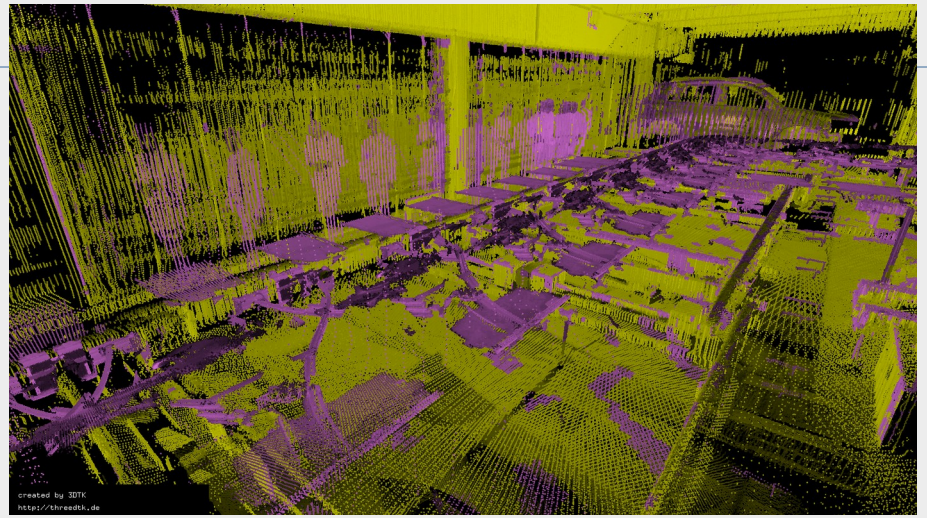


First scan position (green points)



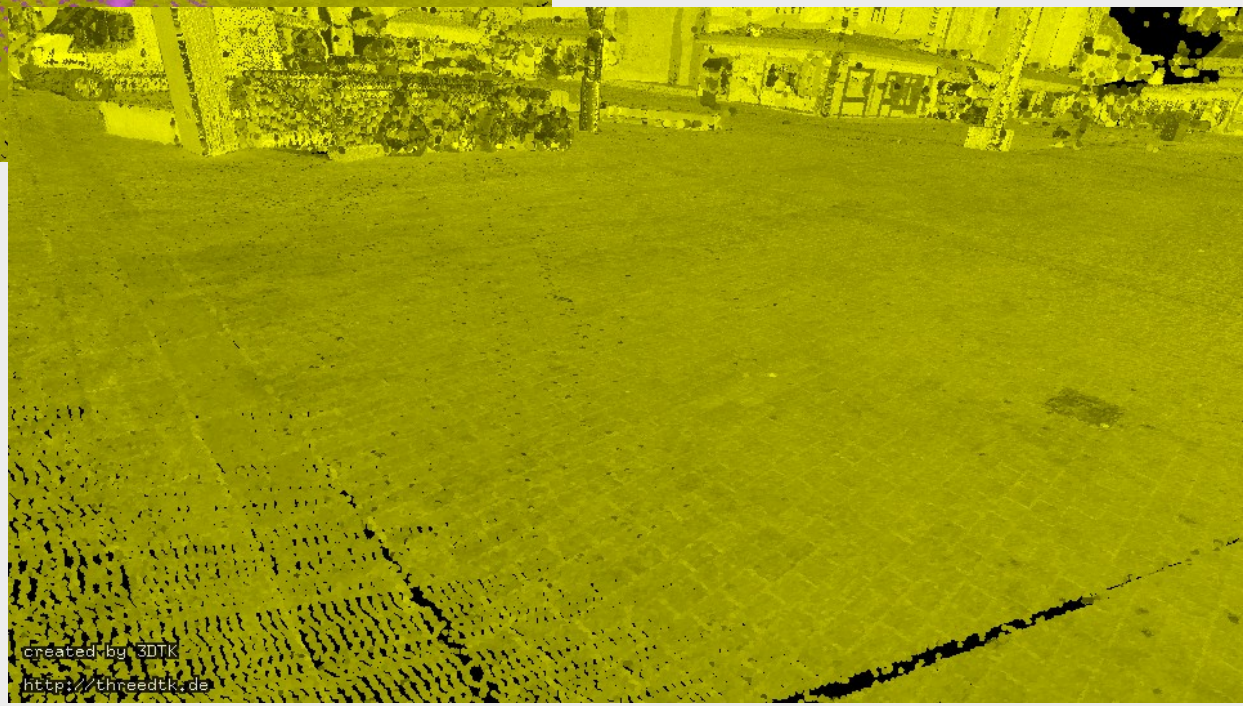
Second scan position (red points)



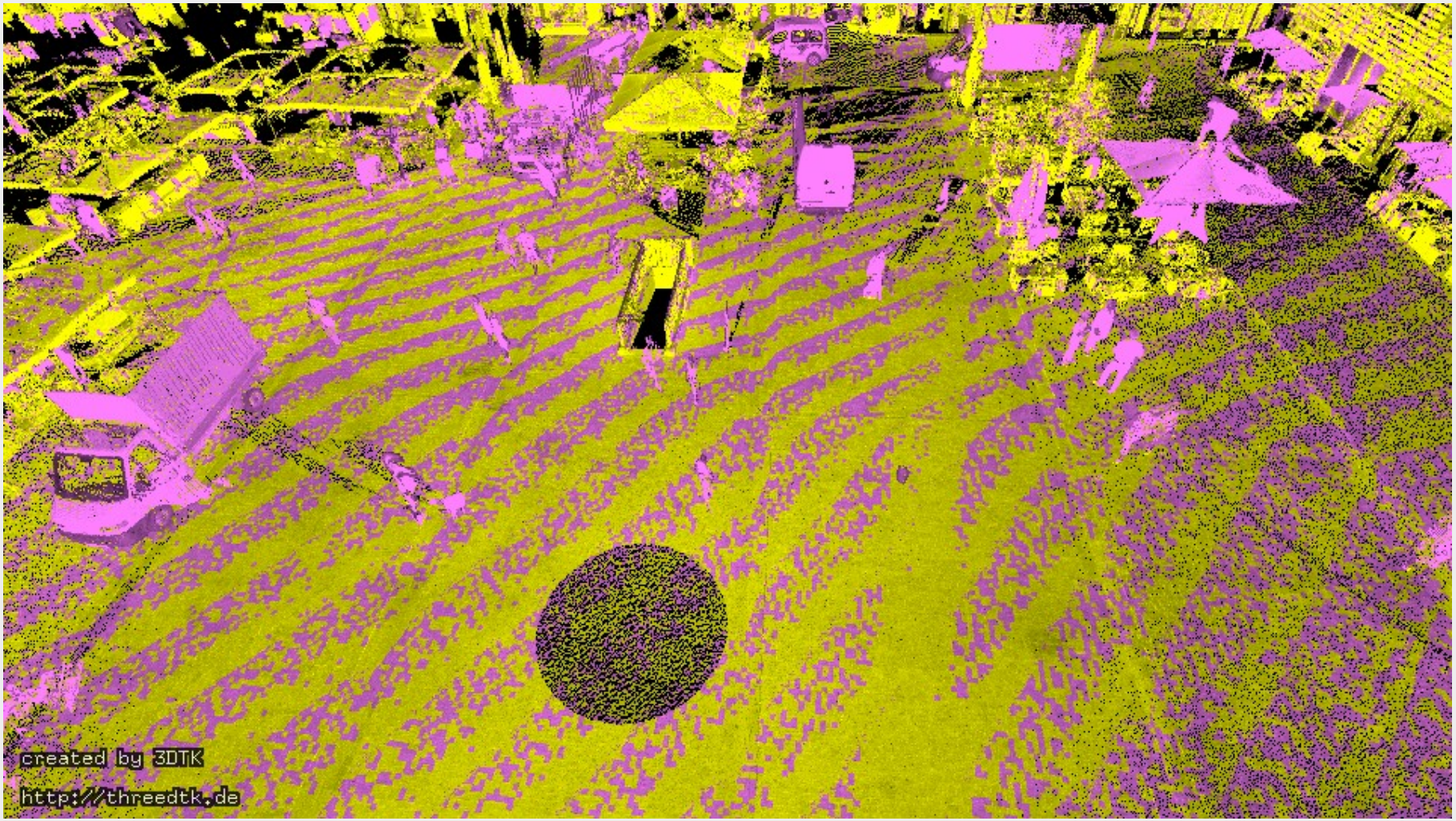


Yellow: static points, Magenta: dynamic points

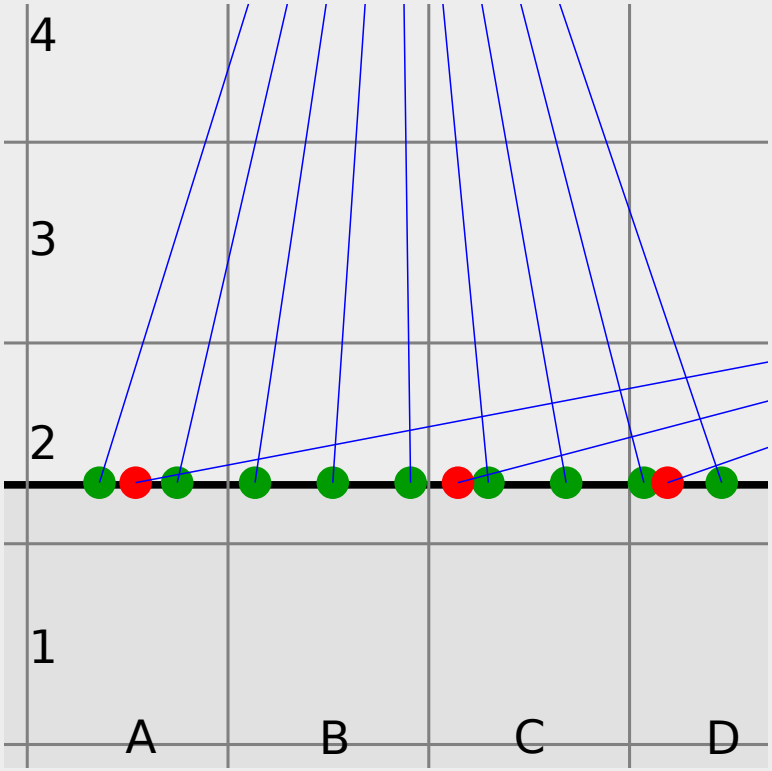
Dynamic points removed: only static points remain

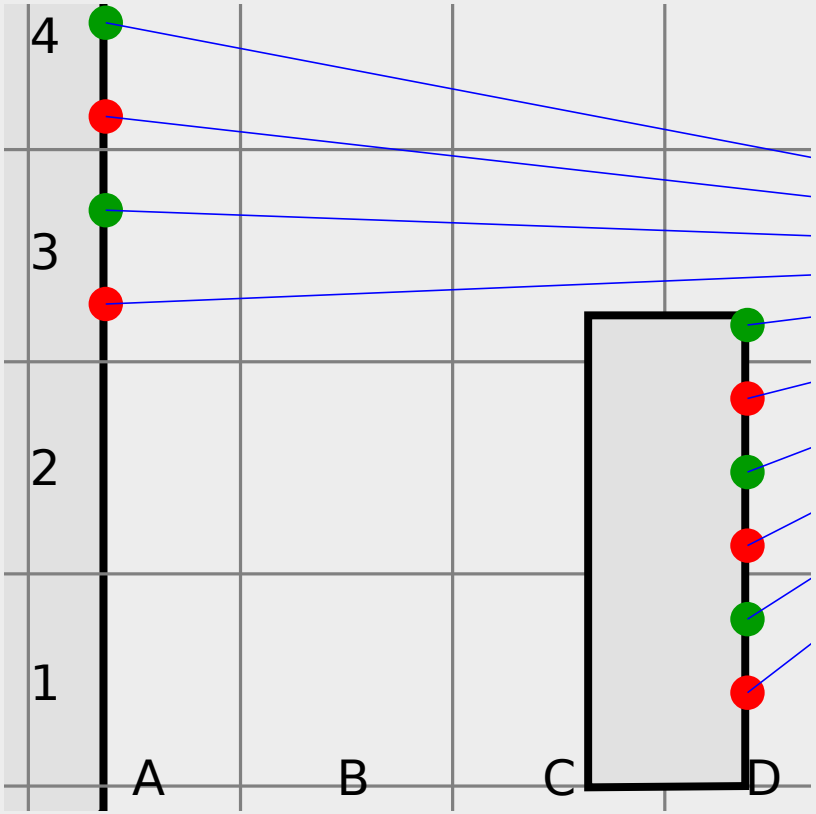


Schauer, J., Nüchter, A.: **The Peopleremover --- Removing Dynamic Objects From 3-D Point Cloud Data by Traversing a Voxel Occupancy Grid.** IEEE Robotics and Automation Letters (RAL). 3, 1679--1686 (2018).

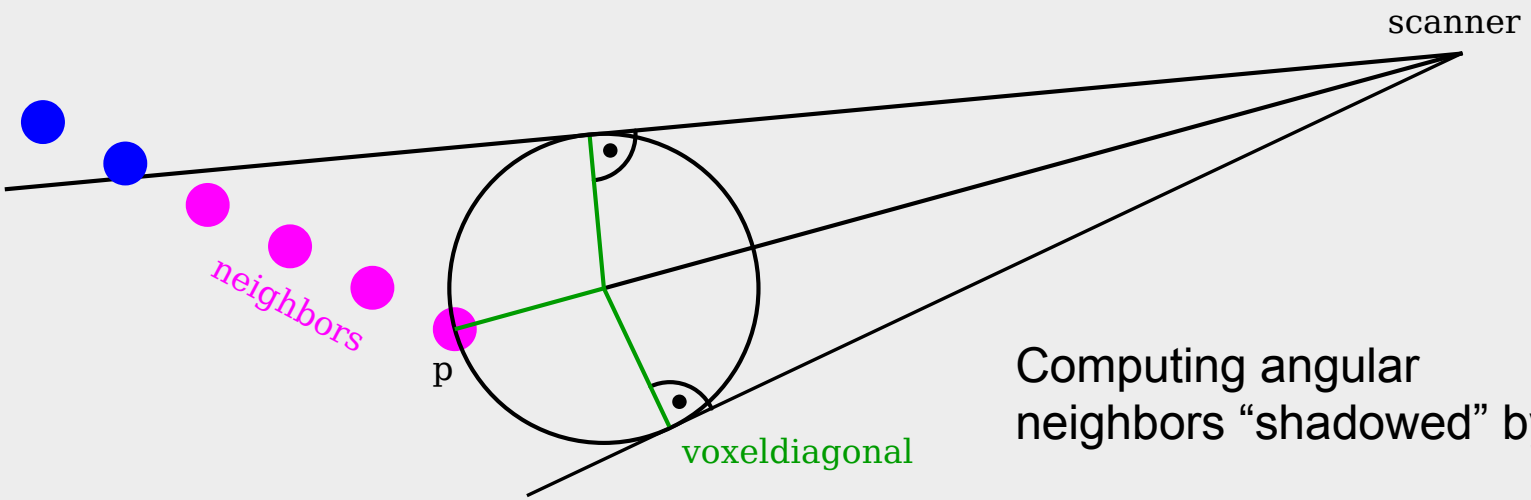

created by 3DTK
http://threedtk.de

Problem with terrestrial scan data

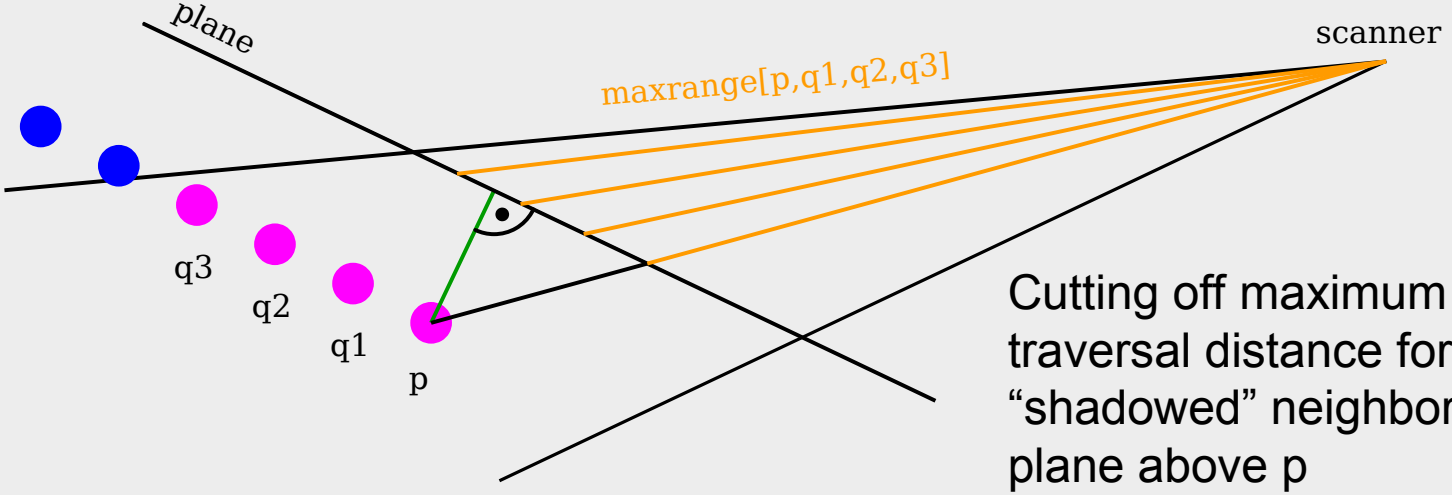




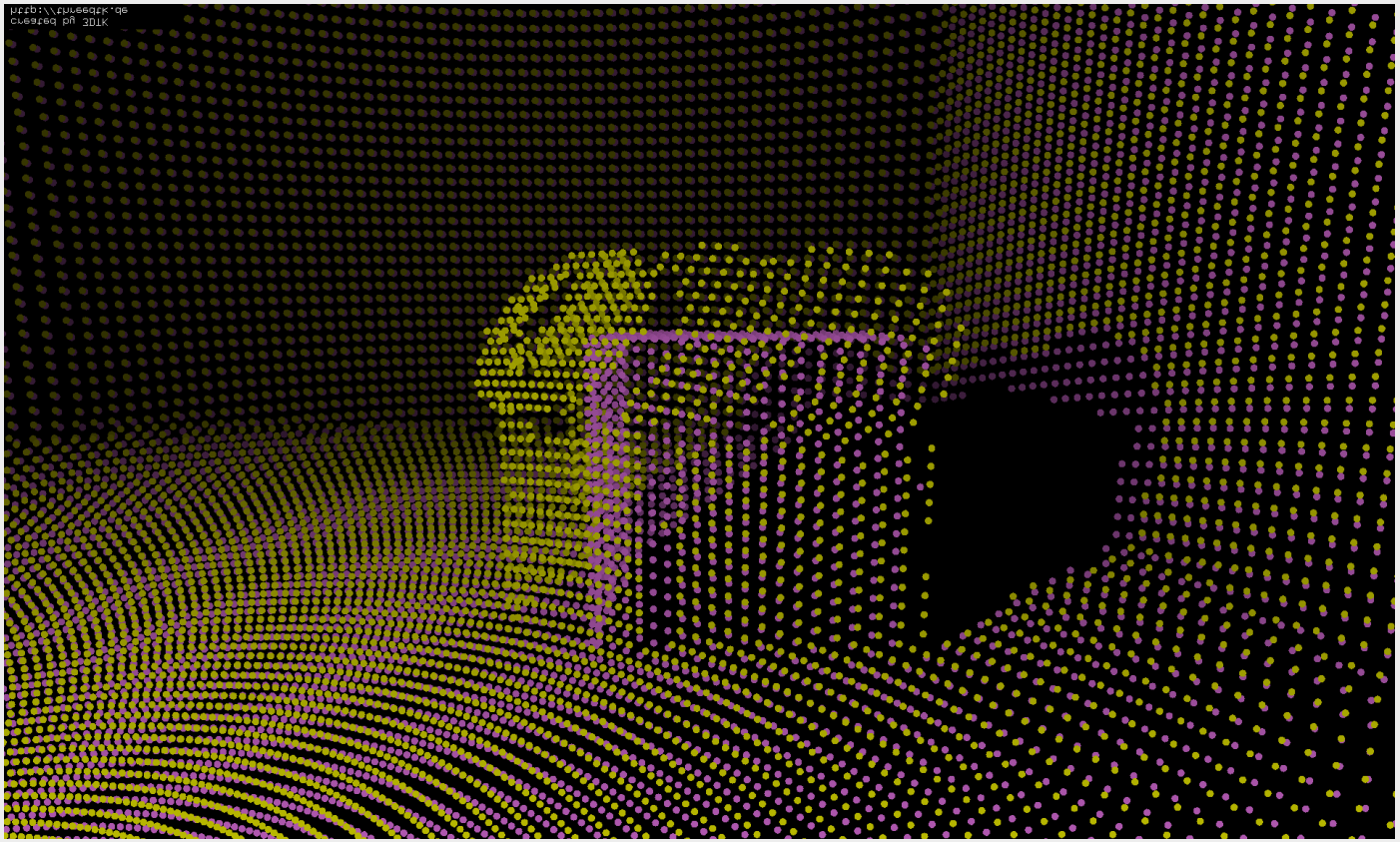
Computing “point shadows”



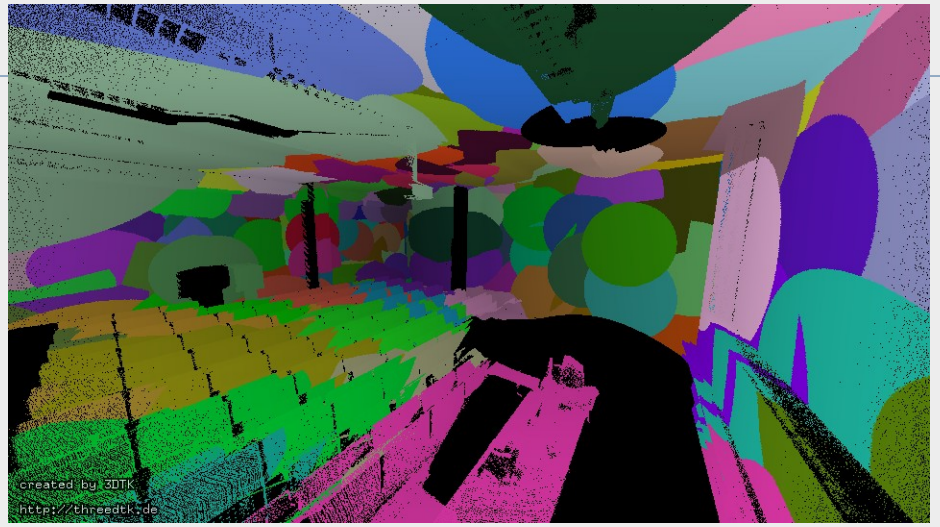
Computing angular neighbors “shadowed” by p

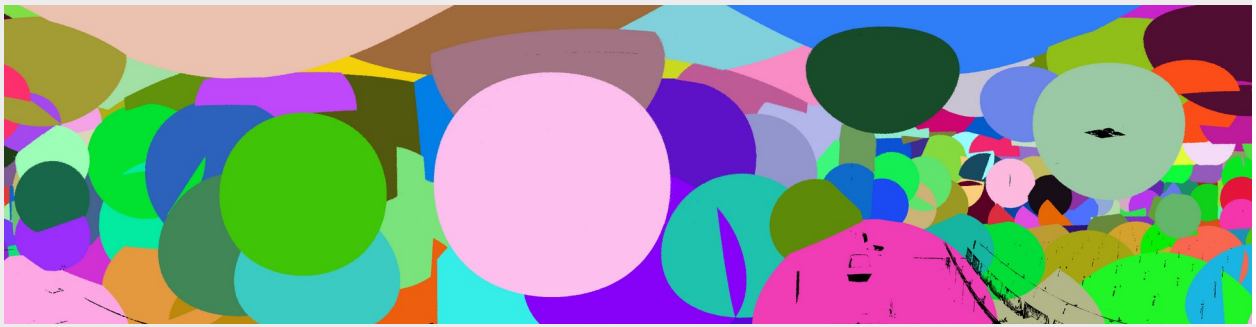


Cutting off maximum traversal distance for all “shadowed” neighbors at the plane above p



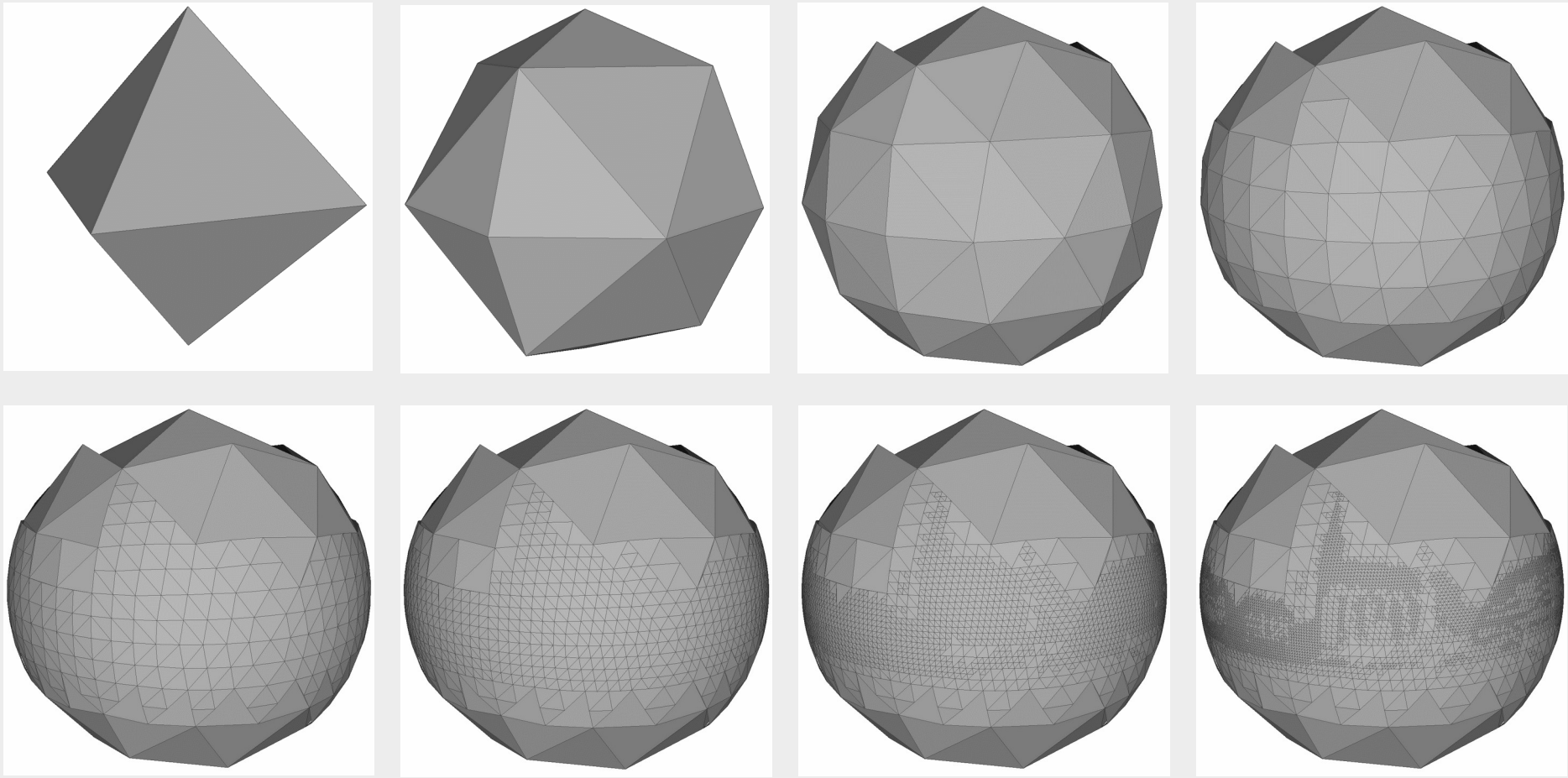
Magenta points: synthetic model
Yellow points: traversal distance cut-off point
Scanner is in the upper-left



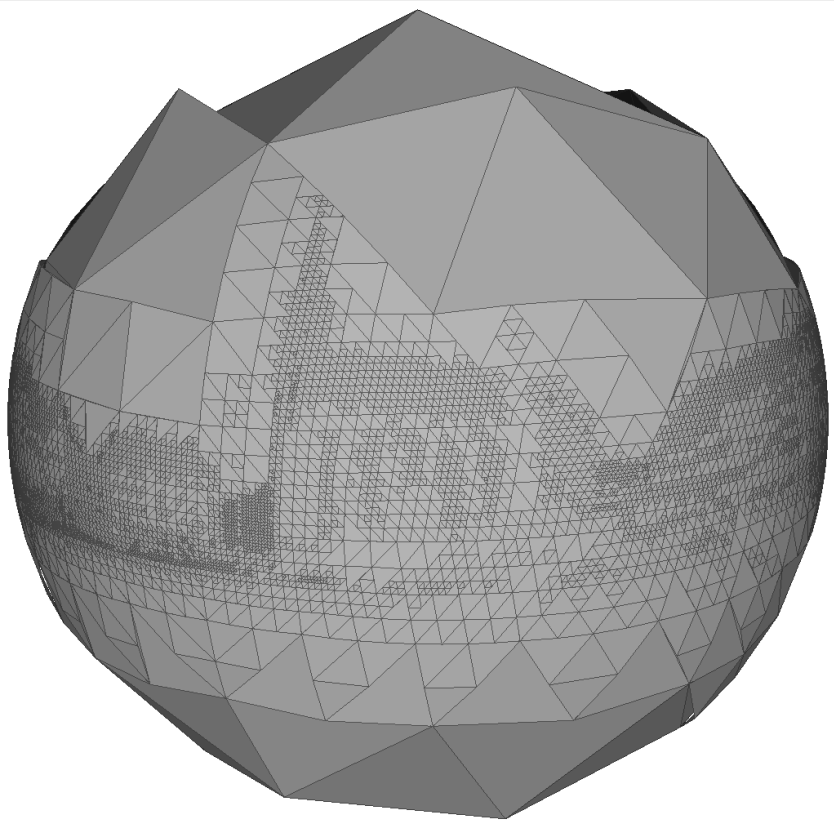


dataset	normals(%)
sim	8.03
lab	0.02
carpark	0.23
lecturehall	0.003
campus	0.16
würzburg	0.21
bremen	0.222
randersacker	0.010

Spherical quadtree to compute range neighbors

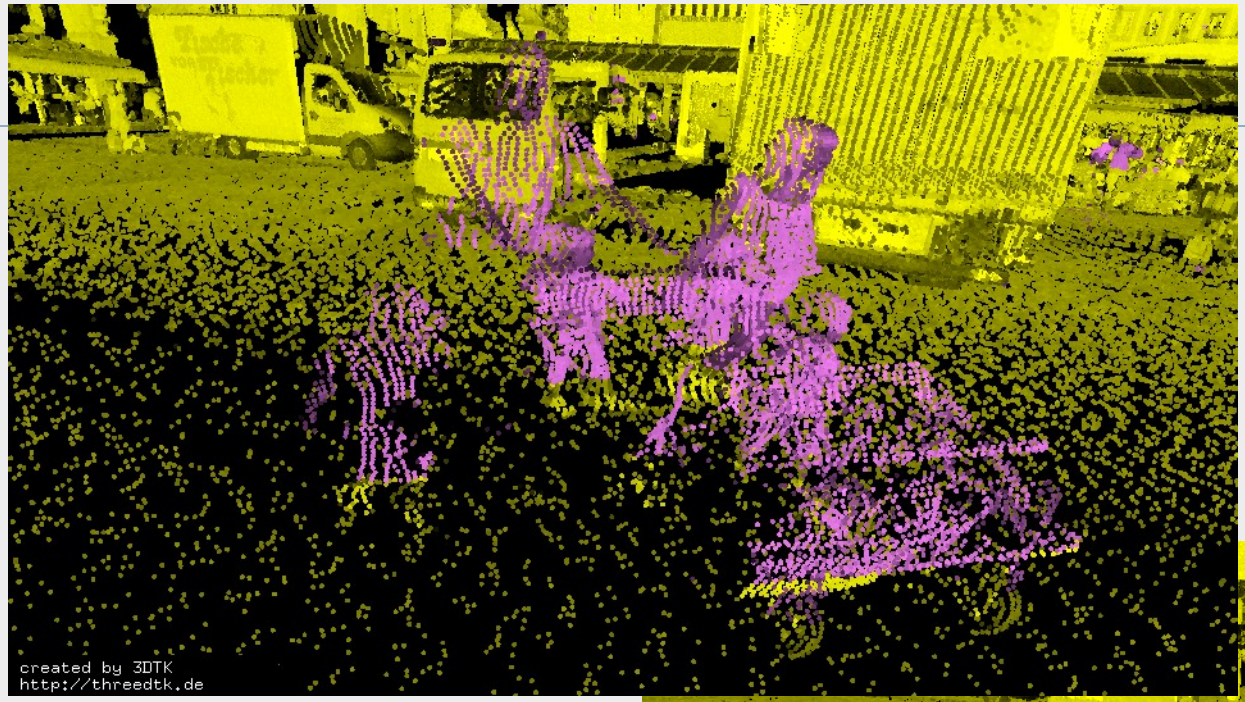


Subdivisions of the faces of an octrahedron up to depth 8

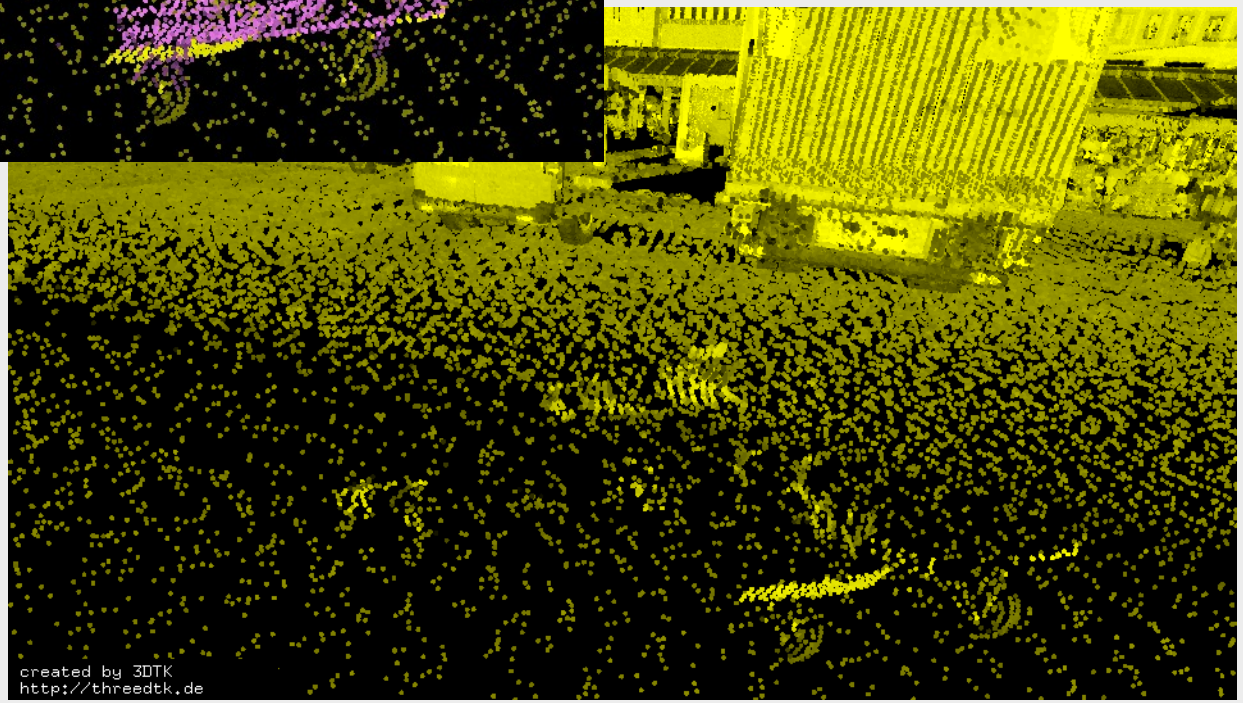


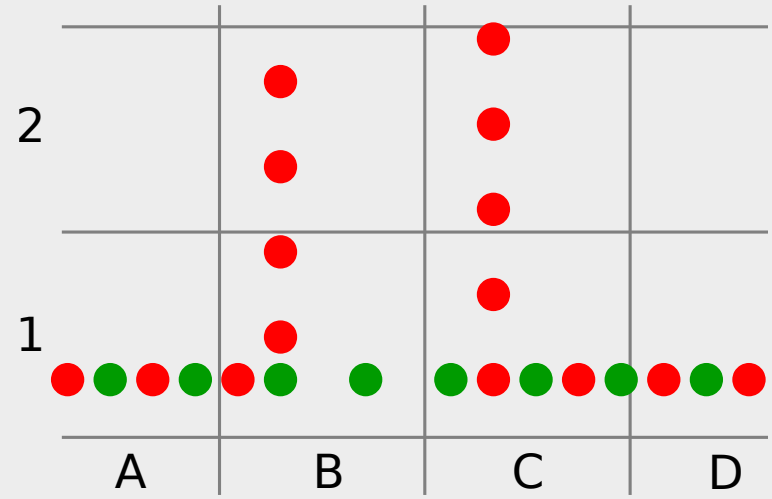
Ninth subdivision

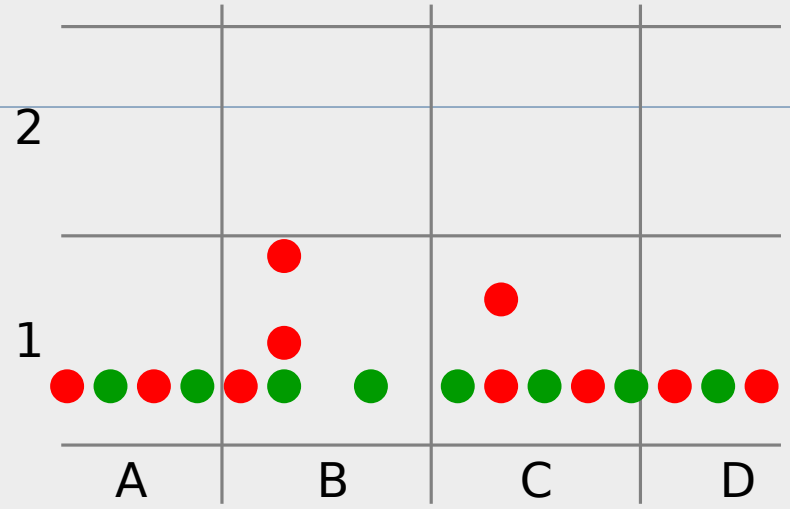
Points projected onto perfect sphere with reflectivity information

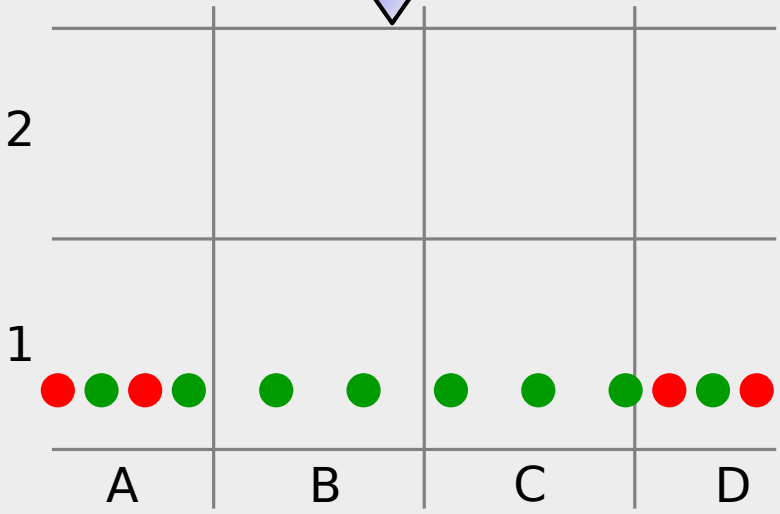


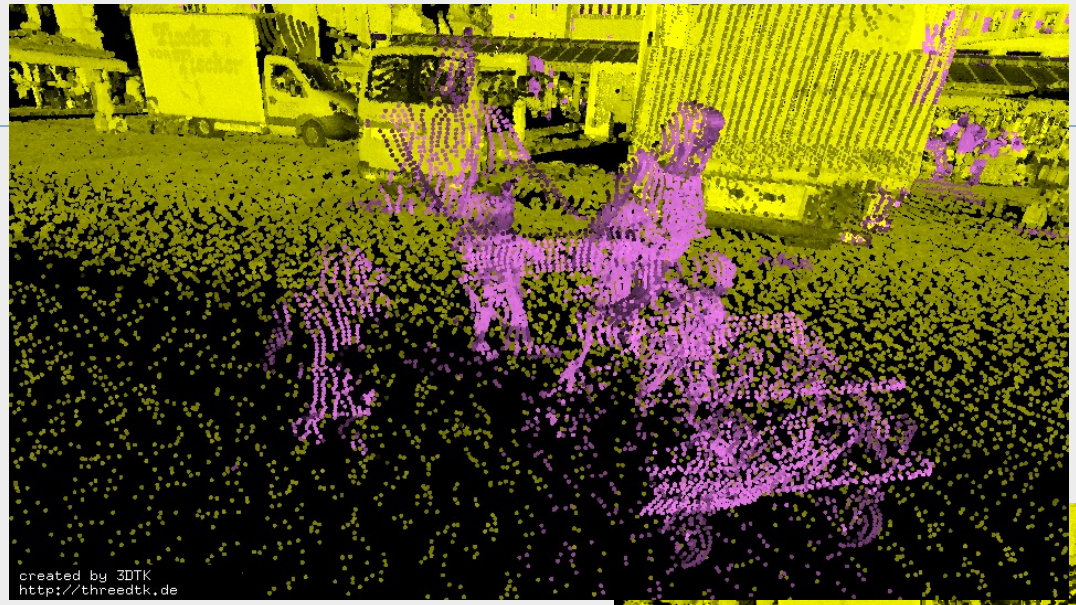
Artifacts remain



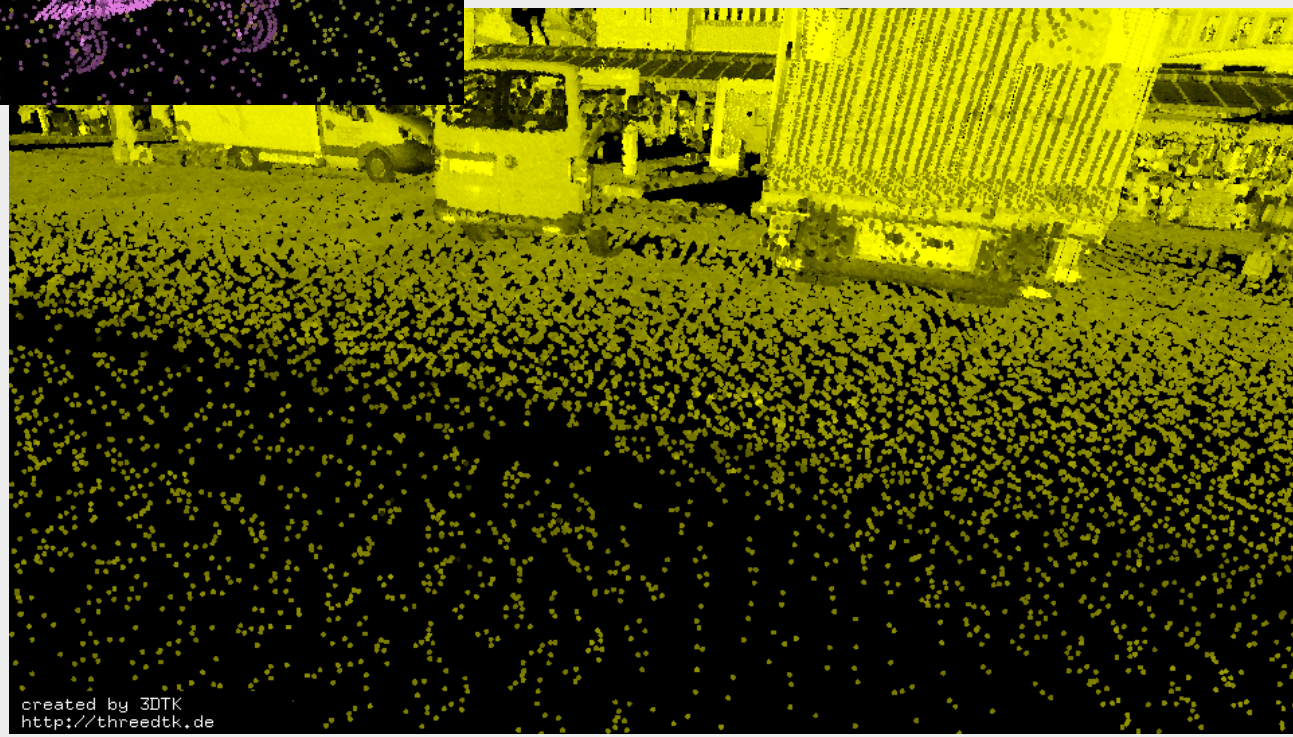

2
1
A
B
C
D


2
1
A
B
C
D


2
1
A
B
C
D



Artifacts are removed



Improvements of voxel traversal

- Original algorithm by Amanatides et al. (1987)
- Corner cases not covered by any existing implementation (Octomap, MRPT, PCL, yt)
- Improvements:
  - Consistent results when ray crosses voxel boundaries
  - Consistent results independent of ray direction
  - Avoiding accumulation of floating point errors
  - Support for rays starting at exactly a voxel boundary
  - No measurable performance impact

Schauer, J., Nüchter, A.: **Removing non-static objects from 3D laser scan data.** ISPRS Journal of Photogrammetry and Remote Sensing (JPRS). 143, 15--38 (2018).

Clustering for noise removal

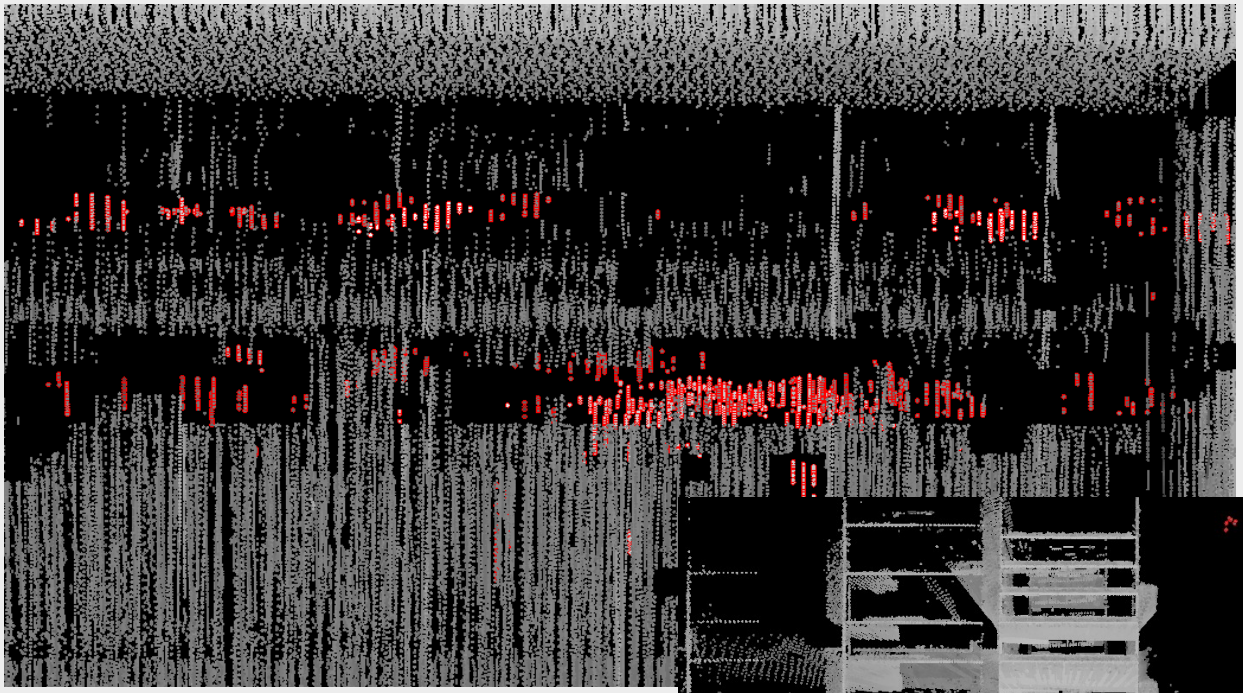
- Creation of clusters of adjacent dynamic voxels
- Removal of clusters with less than N dynamic voxels
- Very fast by working in voxel space

Quantitative Results (F_1-score and runtime)

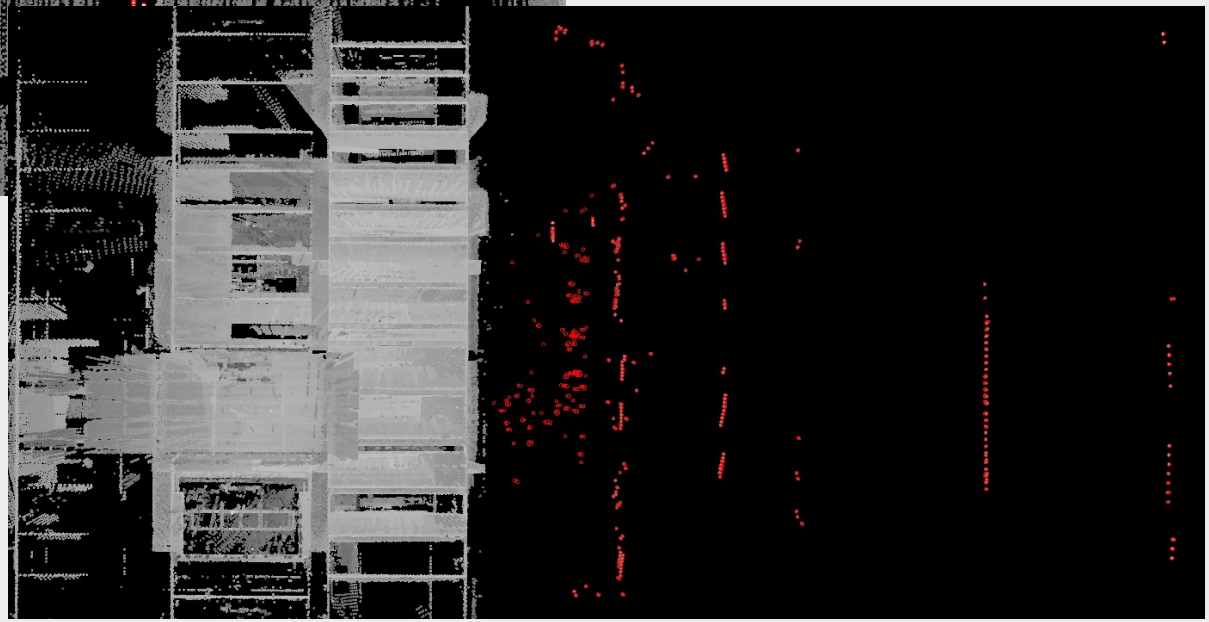
dataset	Underwood			3DTK	
name	T_a	$T_r(m)$	F_1-score	voxel size(m)	F_1-score
sim	1.4	0.1	**0.98**	0.6	**0.98**
lab	1.2	0.2	**0.71**	0.175	0.42
carpark	1.0	0.35	0.78	0.125	**0.83**
lecturehall	0.8	0.3	**0.96**	0.1	**0.96**

dataset	Underwood	3DTK
name	$t(s)$	$t(s)$
sim	25	**6**
lab	405	**29**
carpark	34	**23**
lecturehall	837	**687**
campus	12.8 days	**13.1 hours**
würzburg	7961	**4967**

Reflecting Surfaces...



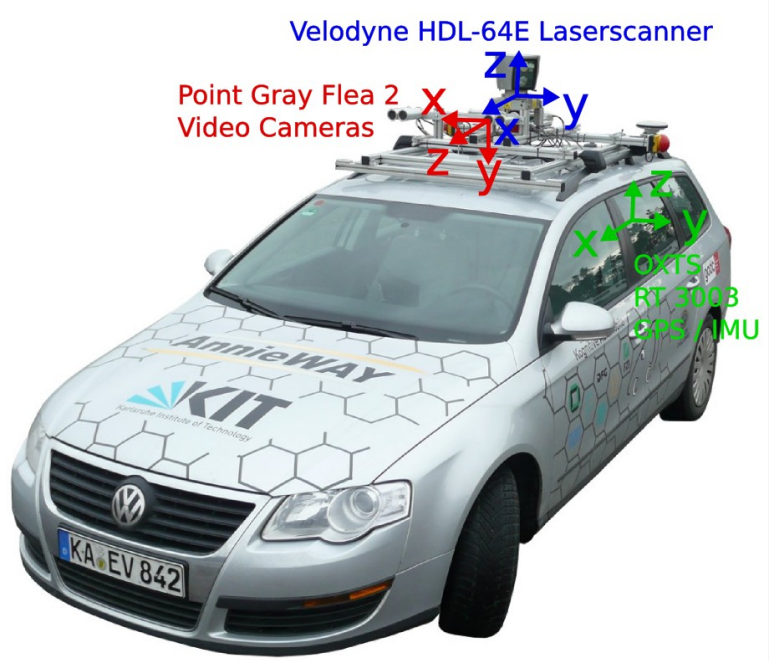
“holes” in a solid wall...



…because of mirrored points (red) behind it

Change detection on the KITTI dataset

Schauer, J., Nüchter, A.: **Analytical Change Detection on the KITTI dataset.** The 16th International Conference on Control, Automation, Robotics and Vision, ICARCV 2020 (2020).

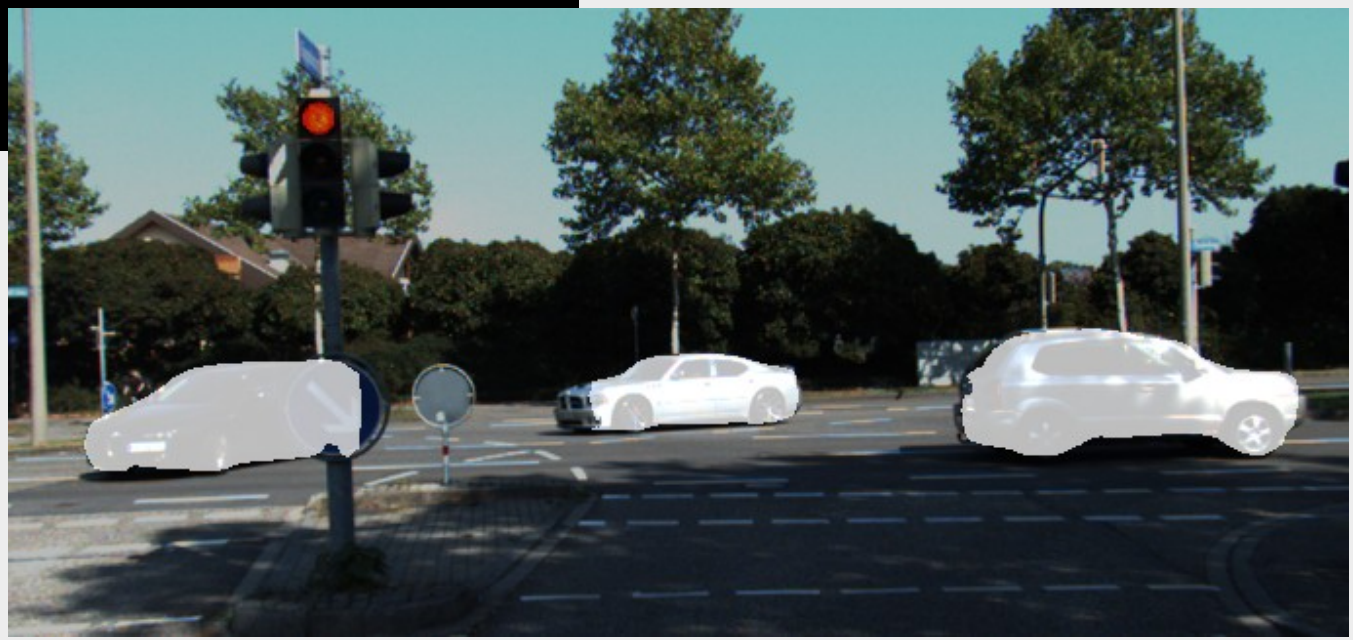


KITTI setup by Geiger et al.

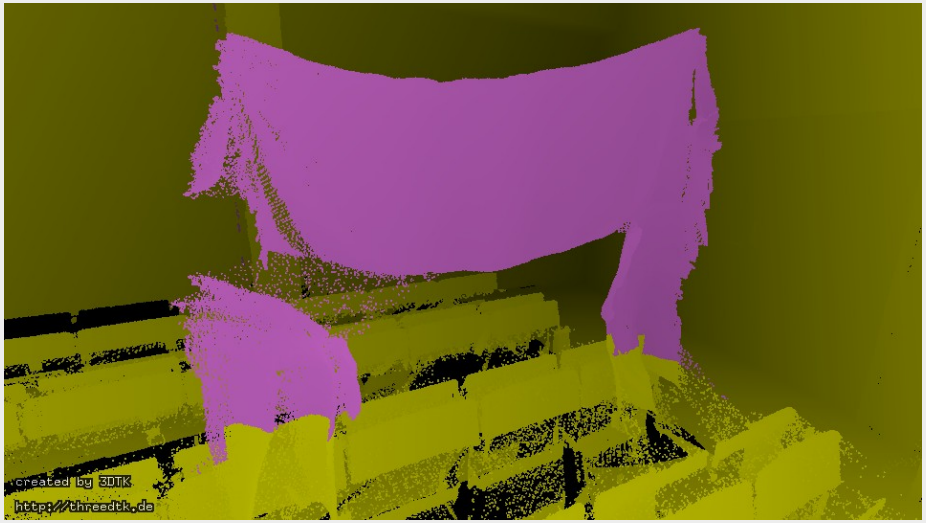
#	Underwood	3DTK
9	0.3005	**0.3658**
11	0.4385	**0.5544**
13	0.2610	**0.5684**
15	0.4906	**0.6331**
17	**0.7122**	0.6285
18	0.4409	**0.4975**
39	**0.2994**	0.2739
46	0.2373	**0.6562**
48	0.2051	**0.5656**
51	0.2161	**0.6529**
59	0.4090	**0.4449**
overall	0.3269	**0.5290**

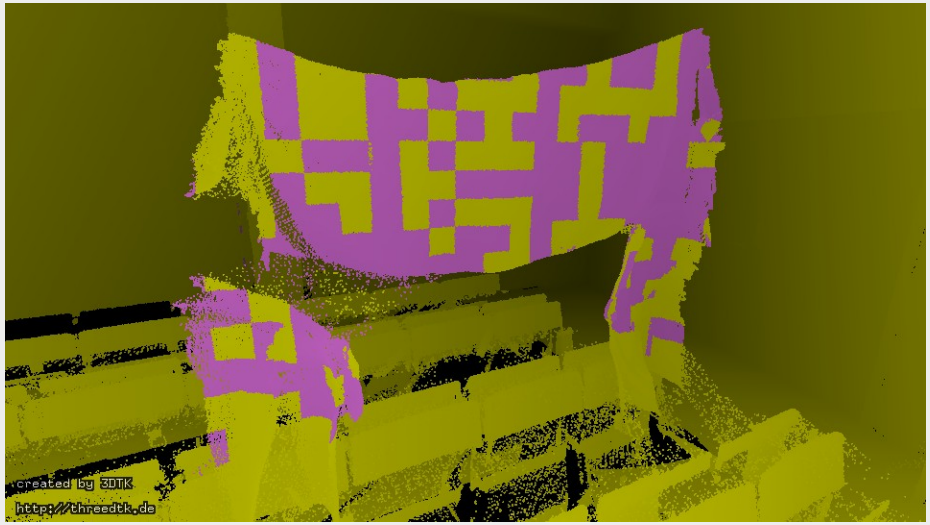
Problems with KITTI Dataset

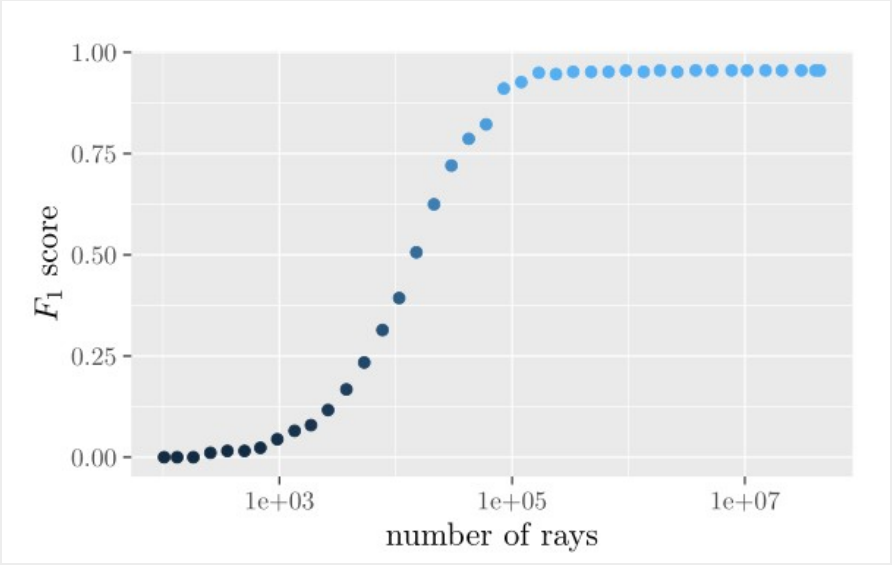
- Noisy data (Velodyne)
- Reflections (image on the left)
- Problems with ground truth (image below)

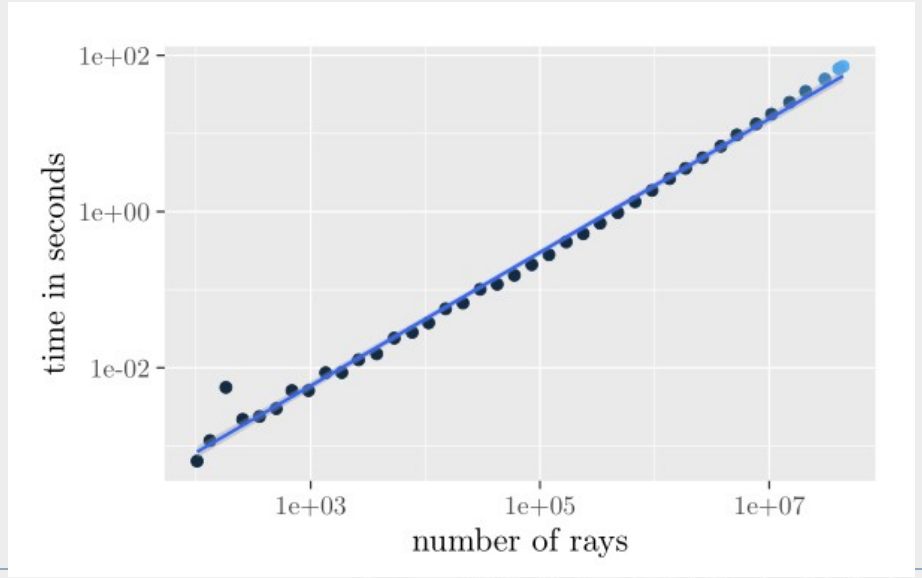


Spherical Quadtree reduction









Limitations

- Data must be correctly registered (no “double” walls)
- No mirrored points
- Data must contain supposedly free volume seen as free

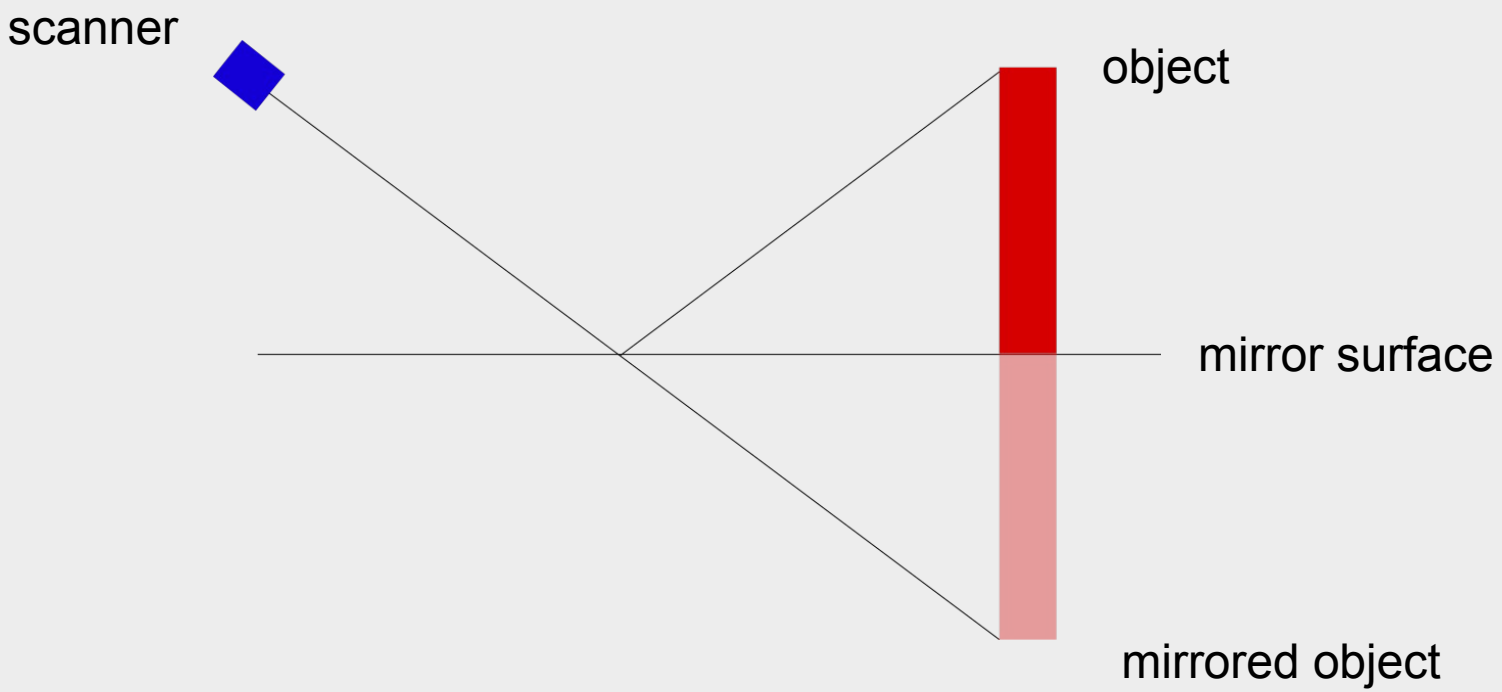
Change detection: Contributions

- A new algorithm for voxel-based change detection
  - Similar or better F_1-scores compared to existing solutions
  - Better runtime compared to existing solutions
- Spherical Quadtree datastructure for angular range searches (to compute “point shadows”) and point reduction (for runtime improvements of multiple orders of magnitude)
- Voxel based artifact removal for improved qualitative results
- Improved voxel traversal algorithm
- All code is freely available under the terms of the GPL3
- Scripts to reproduce all quantitative results from freely available point cloud datasets

https://robotik.informatik.uni-wuerzburg.de/telematics/download/RA-L_2018/
https://robotik.informatik.uni-wuerzburg.de/telematics/download/isprs2018/
https://robotik.informatik.uni-wuerzburg.de/telematics/download/icarcv2020/

Future Work

Mirrors



Questions

Questions?