Volume estimation via integrating on a curve fitted point cloud

HiPEDS 2018 Cohort: G. Bisbas, L. Castiglione, D. Grumberg, S. Karolčík, L. Keeble, D. Kulon, B. Kwan, C. McMeel, R. Miles, J. Ortiz, N. Perez-Nieves, V. Pham Ngoc, J. Vandebon, D. Vink

Imperial College London

October 30, 2018

The problem and the goal

Figure: Royal vans. *Courtesy of https://www.pressandjournal.co.uk*

- **Problem:** Packaging in vans is not optimal → lots of empty space
- **Goal:** Fast estimation of available volume to ensure optimal packaging

Overall structure

HiPEDS Group workflow

- Cohort meetings on a regular basis
- Identify our goals and split into subgroups
- Integrate our progress
- Redefine goals

Point cloud integration team overall checkpoints

- Capture images
- Extract point cloud
- Fit a curve
- Find the volume inside

More details in the next slides...

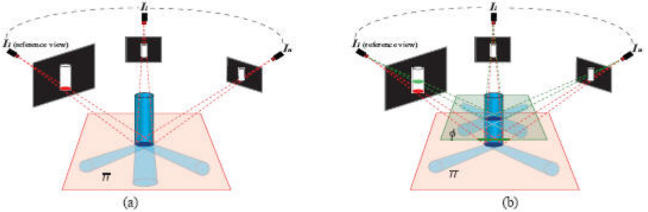
The hardware - Choosing the camera fitting our needs best

The Realsense *D*435 has been proved to be the best choice.

- \+ Relatively cheap.
- \+ Independent from lighting conditions.
- \- Still expensive but...
- \+ ...can be used in multiple vans!

Extracting the point cloud: PLYExtractor

Using the RealSense SDK, a piece of software has been written to automatise the process of extracting pointclouds from different points of view



PLYExtractor: Activity Flow

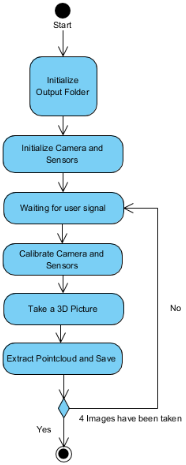


Figure: Software Activity Flow

PLYExtractor: in Action

(a)

(b)

Denoising the point cloud

Extracted point clouds are often very noisy, so, a need for removing outliers is needed.

- Step 1: Hard-cut denoising of points that are very far away of the mass centre.
- Step 2: Use a nearest neighbours approach with a predefined threshold to remove more outliers.

Resulted point clouds were found to be **95%** correctly denoised via simple parameter tuning.

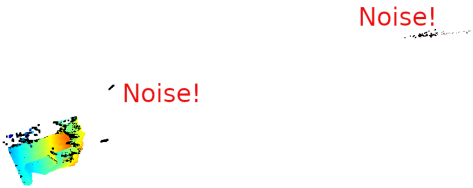


Figure: A noisy point cloud.

The ICP algorithm

The Iterative Closest Points algorithm was used to merge the different point clouds extracted from the camera.
For each point in the source point cloud, match the closest point in the reference point cloud (or a selected set).

1. Estimate the combination of rotation and translation using a root mean square point to point distance metric minimization technique which will best align each source point to its match found in the previous step. This step may also involve weighting points and rejecting outliers prior to alignment.
2. Transform the source points using the obtained transformation.
3. Iterate (re-associate the points, and so on).

Merging point clouds

Merging procedure: 2 steps
1st step: 4 to 2
2nd step: 2 to 1

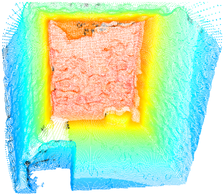
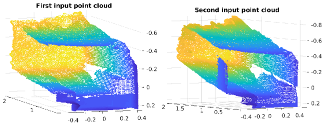


Figure: After merging.



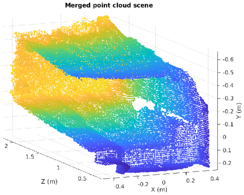


Figure: Intermediate merged pointclouds (left) and final point cloud (right). Denoised and downsampled version of final point cloud (bottom).

Curve fitting with Linear Interpolation

Using MATLAB's Curve Fitting Toolbox, we managed to

- Fit a polynomial surface
- Step 2: Use a nearest neighbours approach with a predefined threshold to remove more outliers.

Resulted point clouds were found to be **95%** correctly denoised via simple parameter tuning.

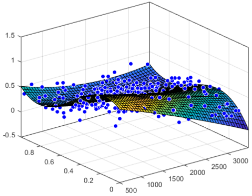


Figure: A simple curve fitting

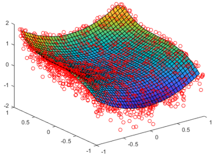
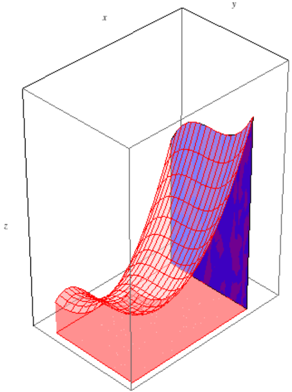


Figure: Linear interpolation

Integration - Keep it simple scientist

$$\iiint_V f(x, y, z)\, dx\, dy\, dz \qquad (1)$$



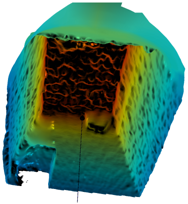


Figure: A full curve fitted surface

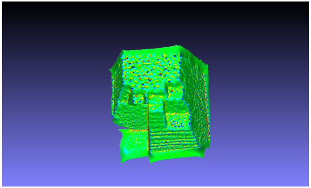


Figure: Curvature of the surface

Results-1

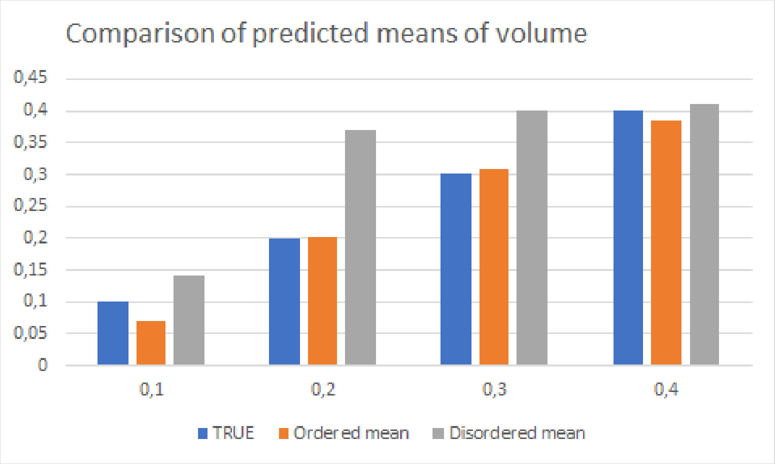


Figure: Comparison of ordered and disordered setups with the real occupied volume

Results-2

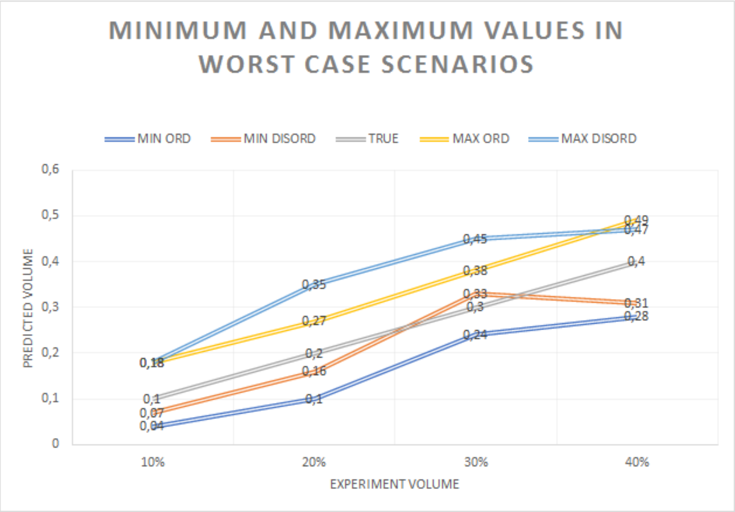


Figure: Worst case scenario differences

Conclusion-Our proposal

- 1 camera needs 10 seconds in every van - Up to 6 vans per minute sequentially.
- Occupied volume prediction in less than 45 seconds for every van.
- 3% - 4% error in 85% of cases - Worst case scenario errors up to 11%.
- With less than 200gbp we should be able to calculate the volume of 100 vans in less than 20 minutes, a second camera … 200 vans…etc…
- Not real time but requirements met.
- Simplicity - No installments
- Only external use
- Easy to transfer hardware
- Solution ready to be tested as soon as possible

Things to consider

- Time limitations
- Manual Procedure
- Free software to reduce cost?

Future extensions

- Depth picture quality
- Faster calculations
- Camera price trends

References

1. Z. Zhang, "Iterative point matching for registration of free-form curves and surfaces," Int. J. Comput. Vis., 1994.
2. E. Johnson and M. Hebert, "Surface registration by matching oriented points," Proceedings. Int. Conf. Recent Adv. 3-D Digit. Imaging Model. (Cat. No.97TB100134), 1997.
3. P. Besl and N. McKay, "A Method for Registration of 3-D Shapes," Tpami, 1992.
4. M. Kazhdan and H. Hoppe, "Screened poisson surface reconstruction," ACM Trans. Graph., vol. 32, no. 3, pp. 1–13, 2013.
5. N. Amenta, S. Choi, and R. K. Kolluri, "The power crust," Proc. sixth ACM Symp. Solid Model. Appl. - SMA '01, pp. 249–266, 2001.
6. M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson Surface Reconstruction," Proc. Symp. Geom. Process., pp. 61–70, 2006.

Acknowledgements

This project was proposed and supported by Royal Mail, EPSRC and Imperial College London. We would like to than all the HiPEDS 1st year PhD students for their cooperation and a special thanks to Jeremy Bradley and Ben Glocker for their support and advice throughout.