Referenties	
[1] Elaborative Evaluation of RGB-D based Point Cloud Registration for Personal Robots	http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.391.2433&rep=rep1&type=pdf
[2] An evaluation of the RGB-D SLAM system	http://ieeexplore.ieee.org/abstract/document/6225199/
[3] Open source Visual SLAM evaluation	http://nicolovaligi.com/open-source-visual-slam-evaluation.html
[4] An emprical point error model for TLS derived point clouds	http://www2.isikun.edu.tr/personel/akca/devrim/2016CZ_Ozendi_etal_ISPRS_Prague.pdf
[6] On Measuring The Accuracy of SLAM Algorithms	http://www.diva-portal.org/smash/get/diva2:459937/FULLTEXT02
[7] Towards a benchmark for RGB-D SLAM evaluation	http://vision.in.tum.de/_media/spezial/bib/sturm11rss-rgbd.pdf
[8] ORB-SLAM: a Versatile and Accurate Monocular SLAM System	https://arxiv.org/pdf/1502.00956
g$_2$o: A General Framework for Graph Optimization	https://www.cct.lsu.edu/~kzhang/papers/g2o.pdf

Validatie Mapping

Methodieken	**In tekst**
[1] Distance Error Metric	*We used a **distance error metric** to evaluate each test, which was calculated by first finding point correspondences using nearest neighbour search, rejecting correspondences over a distance of 1cm[2] (as to ignore non overlapping sections of the clouds) and normalizing the error over the number of correspondences.* *It is important to note this metric evaluates the structural alignment of clouds, and does not take matching point colour into account. Therefore flat surfaces may obtain very good results even when the textures are completely misaligned.*
[4] Iterative closest point	*However, the former approach proposes a method which aims to refine data iteratively, similar to **ICP** algorithm, and the latter approach estimates the uncertainty by determining the precision of range measurement with respect to intensity of the observed light as described in (Sagawa et al., 2005).*
[4] Point Error Model	***Anisotropic Point Error Model*** *TLS system operates in spherical coordinate system measuring range (ρ), horizontal (θ) and vertical (α) angles where any point is defined as $r_i = [\rho, \theta, \alpha]^T$. Most of the current TLSs provide the point cloud data in Cartesian coordinate system, which are computed from spherical coordinates as defined in ...*
[6] Evaluate the quality of the resulting map (differences in the projected sensor data)	$error = \frac{1}{n}\sum_{i=1}^{n}\sum_{u=1}^{640}\sum_{v=1}^{480}[(\hat{\mathbf{x}}_{i+\Delta} \ominus \hat{\mathbf{x}}_i)P_i(u,v) - (\mathbf{x}_{i+\Delta} \ominus \mathbf{x}_i)P_i(u,v)]^2$ $\hat{\mathbf{x}}_{1:n}$ estimated traj. $\mathbf{x}_{1:n}$ ground-truth traj. $P_{1:n}$ sequence of sensor data (3D point clouds)

Validatie Trajectory

Methodieken	In tekst
[2] Een combinatie van: ● Root Mean Squared Error ● Scale Invariant Feature Transform ● Fast Library for Approximate Nearest Neighbors	*To evaluate our system, we use the RGB-D benchmark* *[29] which provides a dataset of Kinect sequences with synchronized ground truth. Furthermore, the benchmark* *provides an evaluation tool that computes the **root mean*** ***square error** (RSME) given an estimated trajectory.* *Accuracy of the Trajectory Estimation* *In our first rounds of experiments, we evaluated the accuracy of our system on all sequences using SIFTGPU* *feature extraction and approximate matching using FLANN.*
[6] Evaluate the quality of the camera trajectory (differences in relative poses)	$error = \frac{1}{n}\sum_{i=1}^{n}[(\hat{\mathbf{x}}_{i+\Delta} \ominus \hat{\mathbf{x}}_i) \ominus (\mathbf{x}_{i+\Delta} \ominus \mathbf{x}_i)]^2$ $\hat{\mathbf{x}}_{1:n}$ estimated traj. $\mathbf{x}_{1:n}$ ground-truth traj. $P_{1:n}$ sequence of sensor data (3D point clouds)

Metric	Ground Truth
[2] Accuracy of the Trajectory Estimation	n.v.t
[1] Reliability of the Associations	n.v.t
[3] Accuracy of the Trajectory Estimation	Accurate mapping from a laser system.
[8, 2] Root Mean Squared Error (RMSE)	
[4] Point Error Ellipsoids	n.v.t
[4] Least Square Plane RMSE	n.v.t

Begrippen	
[2] Root Mean Squared	Een manier om de afwijking tussen waardes te berekenen gebaseerd op een “ground truth”.
[2] Scale Invariant Feature Transform	Een algoritme om kenmerken zoals een hoek of een punt te herkennen in een afbeelding.
[2] Fast Library (FLANN)	Een collectie van algoritmes die het beste werken om “nearest neighbor search” uit te voeren. Ook bevat het een systeem om het beste algoritme en optimaal configureren van dit algoritme te selecteren aan de hand van de geleverde dataset.
[3] Ground-truth	Training set met accurate observatie(s) voor het evalueren van supervised learning technieken.
[4] Iterative Closest Point	Een algoritme om de verschillen tussen twee point clouds te minimaliseren.
[6] Monte Carlo Localization	Een algoritme voor robots om locatiebepaling te doen door middel van een particle filter.
[4] Anisotropic	
[2] $\ominus$	Wiskundig symbool voor https://en.wikipedia.org/wiki/Erosion_(morphology)
[6] $\ominus$	Inverse of standard motion composition operator