I2Pmatch: Matching KeyPoints Across Image and Point Cloud Based On Feature Learning

YangXinjie 23020211153982, BaoXueye 31520211154022, QiuMengwei 23020211153960, HeKaiqun 23020211153929, WangXin 23020211153970

Abstract

Many direct and easy approaches have been proposed to solve the registration problem for pose estimation, which work with data from the same modality, i.e., image-to-image and point cloud-to-point cloud. However, these approaches have several limitations in terms of cost, accuracy, and the memory complexity, etc. Few approaches solve the cross-modality registration due to the difficulty in establishing cross-modality correspondences. In this paper, we proposed the I2PMatch: an improved end-to-end deep network architecture to jointly learn the descriptors for keypoints from image and point cloud, respectively. As a result, we are able to directly match and establish 2D-3D correspondences from the query image and 3D point cloud reference map for visual pose estimation. We references I2PMatch (Feng et al. 2019) and recurrent it. Meanwhile, we create our own 2D-3D patches datasets from the Oxford Robotcar dataset with the ground truth camera poses and 2D-3D image to point cloud correspondences for training and testing the deep network.

Instruction

Image-to-point cloud registration refers to the process of finding the rigid transformation, i.e., rotation and translation that aligns the projections of the 3D point cloud to the image. This process is equivalent to finding the pose, i.e., extrinsic parameters of the imaging device with respect to the reference frame of the 3D point cloud; and it has wide applications in many tasks in computer vision, robotics, augmented/virtual reality, etc.

Although the direct and easy approach to solve the registration problem is to work with data from the same modality, i.e., image-to-image and point cloud-to-point cloud, several limitations exist in these same-modality registration approaches. For point cloud-to-point cloud registration, it is impractical and costly to mount expensive and hard-to-maintain Lidars on large fleet of robots and mobile devices during operations. Furthermore, feature-based point cloud-to-point cloud registration (Deng, Birdal, and Ilic 2018a; Zeng et al. 2017; Li and Lee 2019; Yew and Lee 2018) usually requires storage of D-dimensional features ($D \gg 3$) in addition to the (x, y, z) point coordinates, which increases the memory complexity. For image-to-image registration, meticulous effort is required to perform SfM (Ullman 1979; Triggs et al. 1999; Fischler and Bolles 1981) and store the image feature descriptors (Rublee et al. 2011; Lowe 1999) corresponding to the reconstructed 3D points for feature matching. Additionally, image features are subjected to illumination conditions, seasonal changes, etc. Consequently, the image features stored in the map acquired in one season/time are hopeless for registration after a change in the season/time.

Cross-modality image-to-point cloud registration can be used to alleviate the aforementioned problems from the same modality registration methods. Specifically, a 3D point cloud-based map can be acquired once with Lidars, and then pose estimation can be deployed with images taken from cameras that are relatively low-maintenance and less costly on a large fleet of robots and mobile devices. Moreover, maps acquired directly with Lidars circumvents the hassle of SfM, and are largely invariant to seasonal/illumination changes.

In this proposal, we propose the I2PMatch - a deep network approach to jointly learn the keypoint descriptors of the 2D and 3D keypoints extracted from an image and a point cloud. We use the existing detectors from SIFT (Deng, Birdal, and Ilic 2018b) and ISS (Dorai and Jain 1997) to extract the keypoints of the image and point cloud, respectively. Similar to most deep learning methods, an image patch is used to represent an image keypoint, and a local point cloud volume is used to represent a 3D keypoint. We propose a triplet-like deep network to concurrently learn the keypoint descriptors of a given image patch and point cloud volume such that the distance in the descriptor space is small if the 2D and 3D keypoint are a matching pair, and large otherwise. The descriptors of the keypoints from both the image and point cloud are generated through our trained network during inference. The EPnP (Engel, Koltun, and Cremers 2017) algorithm is used to compute the camera pose based on the 2D-3D correspondences.

Related Work

Image-to-Image Registration. Images-to-image registrations (Shavit and Ferens 2019; Sattler, Leibe, and Kobbelt 2012) are done in the P^2 space because of the lack of depth

information. This is usually the first step to the computation of the projective transformation or SfM. Typical methods are usually based on feature matching. A set of features such as SIFT (Lowe 1999) or ORB (Rublee et al. 2011) are extracted from both source and target images. Correspondences are then established based on the extracted features, which can be used to solve for the rotation, translation using Bundle Adjustment (Triggs et al. 1999; Richard 2003), Perspective-n-Point solvers (Fischler and Bolles 1981), etc. Such techniques have been applied in modern SLAM systems (Engel, Schöps, and Cremers 2014; Mur-Artal, Montiel, and Tardos 2015; Engel, Koltun, and Cremers 2017). However, such methods are based on feature descriptors in the image modality to establish correspondences, and do not work for our general image-to-point cloud registration task.

Point Cloud-to-Point Cloud Registration. The availability of 3D information enables direct registration between point clouds without establishing feature correspondences. Methods like ICP (Besl and McKay 1992; Chen and Medioni 1992), NDT (Biber and Straßer 2003) work well with proper initial guess, and global optimization approaches such as Go-ICP (Yang, Li, and Jia 2013) work without initialization requirements. These methods are widely used in point cloud based SLAM algorithms like LOAM (Zhang and Singh 2014), Cartographer (Hess et al. 2016), etc. Recently data driven methods like DeepICP (Lu et al. 2019), Deep-ClosestPoint (Wang and Solomon 2019), RPM-Net (Yew and Lee 2018), etc, are also proposed. Although these approaches do not require feature correspondences, they still rely heavily on the geometrical details of the point structures in the same modality to work well. Consequently, these approaches cannot be applied to our task on cross-modal registration. Another group of common approaches is the two-step feature-based registration. Classical point cloud feature detectors and descriptors usually suffer from noise and clutter environments. Recently deep learning based feature detectors like USIP (Li and Lee 2019), 3DFeatNet (Yew and Lee 2018), and descriptors like 3DMatch (Zeng et al. 2017), PPF-Net (Deng, Birdal, and Ilic 2018b), PPF-FoldNet (Deng, Birdal, and Ilic 2018a), PerfectMatch (Gojcic et al. 2019), have demonstrated improved performances in point cloud-based registration. Similar to image-to-image registration, these approaches require feature descriptors that are challenging to obtain in cross-modality registration.

Image-to-Point Cloud Registration. I2PMatch (Feng et al. 2019) is the prior work for general image-point cloud registration. It extracts images keypoints with SIFT (Lowe 1999), and point cloud keypoints with ISS (Zhong 2009). The image and point cloud patches around the keypoints are fed into each branch of a Siamese-like network and trained with triplet loss to extract cross-modal descriptors. At inference, it is a standard pipeline that consists of RANSAC-based descriptor matching and EPnP (Lepetit, Moreno-Noguer, and Fua 2008) solver. Despite its greatly simplified experimental settings where the point clouds and images are captured at nearby timestamps with almost zero relative rotation, the low inlier rate of correspondences reveals the struggle for a deep network to learn common features across the drastically different modalities. Another work (Yu et al. 2020) establishes 2D-3D line correspondences between images and prior Lidar maps, but they requires accurate initialization, e.g., from a SLAM/Odometry system. In contrast, the general image-to-point cloud registration do not rely on another accurate localization system. Some other works (Pham et al. 2020; Cattaneo et al. 2020) focus on image-to-point cloud place recognition / retrieval without estimating the relative rotation and translation.

Method

In this section, we outline our pipeline for visual pose estimation with a 2D query image and 3D point cloud reference map. We first introduce the overview of our pipeline in subsection A. In subsection B, we describe I2PMatch - a deep network to jointly extract the descriptors of the 2D and 3D keypoints from an image and a point cloud. The training loss is given in subsection B. Finally, we discuss the pose estimation algorithm we use to compute the camera pose given at least three 2D-3D correspondences in subsection C.

A. Overview

Given a query image I and the 3D point cloud reference map M of the scene, the objective of visual pose estimation is to compute the absolute camera pose P = [R|t] of the query image I with respect to the coordinate frame of the 3D point cloud reference map M. Unlike existing visual pose methods which associate image-based descriptors, e.g. SIFT (Lowe 1999), to each 3D point in the reference map, we propose the I2PMatch - a deep network to jointly learn the descriptors directly from the 2D image and 3D point cloud. We first apply the SIFT detector on the query image I to extract a set of 2D keypoints $U = \{u_1, \ldots, u_N \mid u_n \in R^2\}$, and the ISS detector (Zhong 2009) on the 3D point cloud of the reference map M to extract a set of 3D keypoints $V = \{v_1, \ldots, v_N \mid v_m \in R^3\}$.Here, N and M are the total number of 2D and 3D keypoints extracted from the image I and point cloud M, respectively. Given the set of 2D image patches centered around each 2D keypoint and 3D local point cloud volume centered around each 3D keypoint, our I2PMatch learns the corresponding set of 2D and 3D descriptors denoted as $P = \{p_1, \ldots, p_N \mid p_n \in R^D\}$ and $Q = \{q_1, \ldots, q_M \mid q_m \in R^D\}$ for each corresponding 2D and 3D keypoint in U and V. D is the dimension of the descriptor. Furthermore, the descriptors P and Q learned from our network yield a much smaller similarity distance d(p,q) between a matching pair of 2D-3D descriptors (p,q) in comparison to the similarity distance $d(\bar{p}, \bar{q})$ between a non-matching pair of 2D-3D descriptors $(\bar{p}, \bar{q}), i.e. d(p, q) \ll d(\bar{p}, \bar{q})$, thus establishing the 2D-3D correspondences between P and Q. Finally, the 2D-3D correspondences found from our I2PMatch are used to estimate the absolute pose of the camera using a PnP algorithm. We run the PnP algorithm within RANSAC for robust estimation.

B. Our I2PMatch: Network Architecture

Our I2PMatch is a triplet-like deep network that jointly learns the similarity between a given pair of image patch

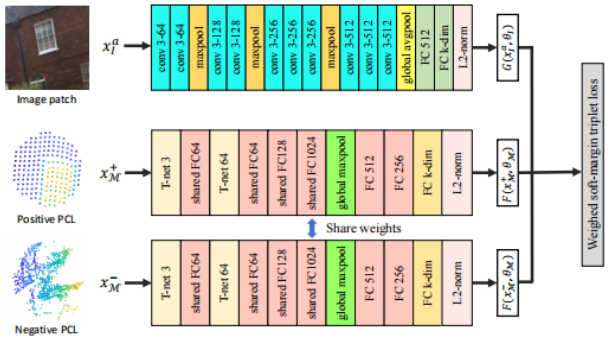

Figure 1: Our triplet-like I2PMatch

and local point cloud volume. The network consists of three branches as illustrated in Figure 1. One of the branches learns the descriptor for the 2D image keypoint and the other two branches with shared weights learn the descriptor for the 3D point cloud keypoint. The inputs to the network are (1) image patches centered on the 2D image keypoints, and (2) local volume of point cloud within a fixed radius sphere centered on the 3D keypoints. Details on keypoints extraction and the definitions of image patch and point cloud sphere are given in Sec. IV-B. The image patches and local volume of point clouds are fed into the network during training as tuples of anchor image patch, and positive and negative local volume point cloud. We denote the training tuple as $\{x_I^a, x_M^+, x_M^-\}$. Given a set of training tuples, our network learns the image descriptor function $Gx_I;\theta_I$: $\mathrm{x}_I \mapsto p$ that maps an input image patch x_I to its descriptor p, and the point cloud descriptor function $Fx_M;\theta_M$: $\mathrm{x}_M \mapsto p$ that maps an input local point cloud volume x_M to its descriptor q. θ_I and θ_M are the weights of the network learned during training.

C. Pose Estimation

The pose of the camera is computed from the putative set of 2D-3D correspondences obtained from our 2D3DMatchNet. Specifically, we obtain the 2D keypoints of the 2D query image with the SIFT detector, and the 3D keypoints of the 3D point cloud with the ISS detector. We compute the 2D and 3D keypoint descriptors with our network from the imagepatches and local point cloud volume extracted around the keypoints. The similarity distance is computed for every pair of 2D and 3D keypoints, and we find the top K closest 3D point cloud keypoints for every 2D image keypoint. Finally, we apply the EPnP algorithm to estimate the camera pose with all the putative 2D-3D correspondences. The EPnP algorithm is ran within RANSAC for robust estimation to eliminate outliers.

Dataset

In this section, we introduce how to create our benchmark dataset - Oxford 2D-3D Patches Dataset. There are 432,982 image patch to pointcloud pairs in the dataset, which allows the training and evaluation sufficient.

A.Oxford 2D-3D Patches dataset

The Oxford 2D-3D Patches dataset is created based on the Oxford RobotCar Dataset (Maddern et al. 2017). The Oxford RobotCar Dataset collects data from different kinds of sensors, including cameras, Lidar and GPS/INS. We use the images from the two (left and right) Point Grey Grasshopper2 monocular cameras, the laser scans from the front SICK LMS-151 2D Lidar, and the GPS/INS data from the NovAtel SPAN-CPT ALIGN inertial and GPS navigation system. Ignoring the traversals collected with poor GPS, night and rain, we get 36 traversals for over a year with sufficiently challenging lighting, weather and traffic conditions. We synchronize the images from the left and right cameras, and 2D laser scans from the Lidar with the timestamps, and get their global poses using the GPS/INS data.

B.Training Data Generation

Keypoint Detection. We build a point cloud based reference map from the laser scans for every submap, where the coordinates of the first laser scan is used as the reference frame. We detect the ground plane and remove all points lying on it. This is because this plane is unlikely to contain any good 3D keypoint and descriptor. The ISS keypoint detector is applied on the remaining point cloud to extract all 3D keypoints. We apply the SIFT detector on every image to extract all 2D keypoints. Fig. 2 shows the visualization of 3D keypoints extracted by ISS detector and Fig. 3 shows the visualization of 2D keypoints extracted by SIFI detector.

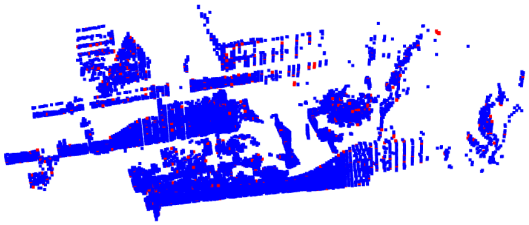

Figure 2: 3D keypoints

Figure 3: 2D keypoints

2D-3D Correspondences. To establish the 2D-3D correspondences, we project each ISS keypoint to all images

within its view and find the nearest neighbour of SIFT keypoint in each image. To increase the confidence of the correspondences, we require the distance of the projected nearest neighbour to be smaller than 3 pixels and each ISS point must have at least SIFT correspondences in three different views within each submap. The ISS points and their corresponding SIFT points that satisfy these requirements are reserved for further processing.

ISS Volume and SIFT Patch Extraction. We remove all ISS keypoints that are within 4m from a selected ISS keypoint in each submap, and remove all SIFT keypoints within 32 pixels from a selected SIFT keypoint in each image. Since larger scale results in smaller patch size, we discard SIFT keypoints with scale larger than a threshold value. In our experiments, we set this threshold value as 4 and the patch size at the basic scale as 128 × 128. We extract the ISS volume and its corresponding SIFT patch if the number of points within the ISS volume is larger than 100 and the SIFT patch is at suitable scale. We discard both the ISS volume and SIFT patch otherwise. Fig. 4 shows the visualization of projecting point cloud to image with ground truth pose and Four examples of our data consist of the local ISS point cloud volumes and their corresponding image patches with different scales, viewpoints and lightings.

Figure 4: Project point cloud to image with ground truth pose. Four examples of our dataset. The first image of each example shows the ISS volume. The other three are some corresponding SIFT patches across multiple frames with different scale, viewpoint and lighting.

Data Pre-processing. Before training, we rescale all the SIFT patches with different scales to the same size, i.e. 128×128, and zero-center by mean subtraction. We subtract each point within each ISS point cloud volume with the associated ISS keypoint, thus achieving zero-center and unit norm sphere. Additionally, we pad the number of points to 1024 for each local volume in our experiments.

C.Testing Data Generation

We use the GPS/INS poses of the images as the ground truth pose for verification. The ground truth 2D-3D correspondences are computed as follows: (1) We detect all ISS keypoints from the point cloud of each submap and retain keypoints with more than 100 neighboring 3D points within 1m radius. (2) We detect SIFT keypoints on each image and extract the corresponding patches with scale smaller than the threshold value, i.e. 4 as mentioned above. (3) Each ISS keypoint is projected to all images within its view and the nearest SIFT keypoint with a distance smaller than 3 pixels is selected as the correspondence. We discard an ISS to SIFT keypoint correspondence if a nearest SIFT within 3 pixels is found in less than 3 image views.

Experiments

Our image-to-point cloud registration approach is evaluated with Oxford Robotcar (Maddern et al. 2017).

Our network is implemented in pythorch with 2× GeForce RTX 3090 GPUs. We train the whole network in an end-to-end manner. For each triplet input, we choose an image patch as the anchor, and its corresponding 3D point cloud volume as positive sample. The negative point cloud volume is randomly sampled from the rest of the point clouds. We initialize the image descriptor network branch with DenseNet model pre-trained on ImageNet (Deng et al. 2009).Both descriptor extraction networks are optimized with Adam optimizer and the initial learning rate is 6×10^{-5}. In our experiments, we train and test with the output feature dimension 128.

As mentioned in before, we reconstruct the point cloud from GPS/INS data for each submap. Next, we detect all the 3D keypoints and infer the corresponding descriptors from the point cloud descriptor network. Given a query image, we extract the 2D SIFT keypoints and feed all the corresponding image patches into the image descriptor network to get the descriptors of the query image.

For each image descriptor, we find its top K nearest point descriptor from our database thus establishing the 2D-3D correspondences. The selection of K can largely effect the localization results. With a larger K, we have more point feature candidates for each image feature. Consequently, the RANSAC algorithm is more likely to find the correct match. On the other hand, a larger K unfavorably increases the number of iterations of RANSAC exponentially. Considering the trade-off, we choose K = 5 for our experiments.

Finally, we solve the camera pose using the EPnP algorithm (Lepetit, Moreno-Noguer, and Fua 2008). the registration is evaluated with two criteria: average Relative Translational Error (RTE) and average Relative Rotation Error (RRE). The results are shown in Table 1.

Due to the lack of existing approaches in solving the image-to-point cloud registration problem under the same setting, we further compare our I2PMatch with 4 other approaches.

Direct Regression. Direct Regression uses a deep network to directly regress the relative poses. The global point cloud

feature and global image feature are concatenated into a single vector and processed by a MLP that directly regresses Gˆ. See the supplementary materials for more details of this method.

Monodepth2+USIP. Monodepth2+USIP converts the cross-modality registration problem into point cloud-based registration by using Monodepth2 (Godard et al. 2019) to estimate a depth map from a single image. The Lidar point cloud is used to calibrate the scale of depth map from MonoDepth2, i.e. the scale of the depth map is perfect. Subsequently, the poses between the depth map and point cloud are estimated with USIP (Li and Lee 2019). This is akin to same modality point cloud-to-point cloud registration. Nonetheless, Table 1 shows that this approach underperforms. This is probably because the depth map is inaccurate and USIP does not generalize well on depth maps.

Monodepth2+GT-ICP. Monodepth2+GT-ICP acquires a depth map with absolute scale in the same way as Monodepth2+USIP. However, it uses Iterative Closest Point (ICP) (Besl and McKay 1992; Chen and Medioni 1992) to estimate the pose between the depth map and point cloud. Note that ICP fails without proper initialization, and thus we use the ground truth (GT) relative pose for initialization.

2D3D-MatchNet. 2D3D-MatchNet (Feng et al. 2019) is the only prior work for crossmodal image-to-point cloud registration to our best knowledge. However, the rotation between camera and Lidar is almost zero in their experiment setting. This is because the images and point clouds are taken from temporally consecutive timestamps without additional augmentation. In contrast, the point clouds in our experiments are always randomly rotated.

Table 1: Registration accuracy on the Oxford datasets.

	RTE (m)	**RRE (°)**
Direct Regression	5.02 ± 2.89	10.45 ± 16.03
MonoDepth2 + USIP	33.2 ± 46.1	142.5 ± 139.5
MonoDepth2 + GT-ICP	1.3 ± 1.5	6.4 ± 7.2
2D3D-MatchNet	1.41	6.40
Ours	6.72	10.23

Conclusion

We presented a novel method for camera pose estimation given a 3D point cloud reference map of the outdoor environment. Instead of the association of local image descriptors to points in the reference map, we proposed to jointly learn the image and point cloud descriptors directly through our deep network model, thus obtaining the 2D-3D correspondences and estimating the camera pose with the EPnP algorithm. We demonstrated that our network is able to map crossdomain inputs (i.e. image and point cloud) to a discriminative descriptor space where their similarity / dissimilarity can be easily identified.

References

Besl, P. J.; and McKay, N. D. 1992. Method for registration of 3-D shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, 586–606. International Society for Optics and Photonics.

Biber, P.; and Straßer, W. 2003. The normal distributions transform: A new approach to laser scan matching. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)*, volume 3, 2743–2748. IEEE.

Cattaneo, D.; Vaghi, M.; Fontana, S.; Ballardini, A. L.; and Sorrenti, D. G. 2020. Global visual localization in LiDAR-maps through shared 2D-3D embedding space. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 4365–4371. IEEE.

Chen, Y.; and Medioni, G. 1992. Object modelling by registration of multiple range images. *Image and vision computing*, 10(3): 145–155.

Deng, H.; Birdal, T.; and Ilic, S. 2018a. Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 602–618.

Deng, H.; Birdal, T.; and Ilic, S. 2018b. Ppfnet: Global context aware local features for robust 3d point matching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 195–205.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

Dorai, C.; and Jain, A. K. 1997. COSMOS-A representation scheme for 3D free-form objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(10): 1115–1130.

Engel, J.; Koltun, V.; and Cremers, D. 2017. Direct sparse odometry. *IEEE transactions on pattern analysis and machine intelligence*, 40(3): 611–625.

Engel, J.; Schöps, T.; and Cremers, D. 2014. LSD-SLAM: Large-scale direct monocular SLAM. In *European conference on computer vision*, 834–849. Springer.

Feng, M.; Hu, S.; Ang, M. H.; and Lee, G. H. 2019. 2d3d-matchnet: Learning to match keypoints across 2d image and 3d point cloud. In *2019 International Conference on Robotics and Automation (ICRA)*, 4790–4796. IEEE.

Fischler, M. A.; and Bolles, R. C. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6): 381–395.

Godard, C.; Aodha, O. M.; Firman, M.; and Brostow, G. 2019. Digging Into Self-Supervised Monocular Depth Estimation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 3827–3837.

Gojcic, Z.; Zhou, C.; Wegner, J. D.; and Wieser, A. 2019. The perfect match: 3d point cloud matching with smoothed densities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5545–5554.

Hess, W.; Kohler, D.; Rapp, H.; and Andor, D. 2016. Real-time loop closure in 2D LIDAR SLAM. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 1271–1278. IEEE.

Lepetit, V.; Moreno-Noguer, F.; and Fua, P. 2008. EPnP: An Accurate O(n) Solution to the PnP Problem. *International Journal of Computer Vision*, 81: 155–166.

Li, J.; and Lee, G. H. 2019. Usip: Unsupervised stable interest point detection from 3d point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 361–370.

Lowe, D. G. 1999. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, 1150–1157. Ieee.

Lu, W.; Wan, G.; Zhou, Y.; Fu, X.; Yuan, P.; and Song, S. 2019. Deepvcp: An end-to-end deep neural network for point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 12–21.

Maddern, W. P.; Pascoe, G.; Linegar, C.; and Newman, P. 2017. 1 year, 1000 km: The Oxford RobotCar dataset. *The International Journal of Robotics Research*, 36: 15 – 3.

Mur-Artal, R.; Montiel, J. M. M.; and Tardos, J. D. 2015. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE transactions on robotics*, 31(5): 1147–1163.

Pham, Q.-H.; Uy, M. A.; Hua, B.-S.; Nguyen, D. T.; Roig, G.; and Yeung, S.-K. 2020. Lcd: Learned cross-domain descriptors for 2d-3d matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 11856–11864.

Richard, I. 2003. Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 673 p.

Rublee, E.; Rabaud, V.; Konolige, K.; and Bradski, G. 2011. ORB: An efficient alternative to SIFT or SURF. In *2011 International conference on computer vision*, 2564–2571. Ieee.

Sattler, T.; Leibe, B.; and Kobbelt, L. 2012. Improving image-based localization by active correspondence search. In *European conference on computer vision*, 752–765. Springer.

Shavit, Y.; and Ferens, R. 2019. Introduction to camera pose estimation with deep learning. *arXiv preprint arXiv:1907.05272*.

Triggs, B.; McLauchlan, P. F.; Hartley, R. I.; and Fitzgibbon, A. W. 1999. Bundle adjustment—a modern synthesis. In *International workshop on vision algorithms*, 298–372. Springer.

Ullman, S. 1979. The interpretation of structure from motion. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 203(1153): 405–426.

Wang, Y.; and Solomon, J. M. 2019. Deep closest point: Learning representations for point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3523–3532.

Yang, J.; Li, H.; and Jia, Y. 2013. Go-icp: Solving 3d registration efficiently and globally optimally. In *Proceedings of the IEEE International Conference on Computer Vision*, 1457–1464.

Yew, Z. J.; and Lee, G. H. 2018. 3dfeat-net: Weakly supervised local 3d features for point cloud registration. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 607–623.

Yu, H.; Zhen, W.; Yang, W.; Zhang, J.; and Scherer, S. 2020. Monocular camera localization in prior lidar maps with 2d-3d line correspondences. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4588–4594. IEEE.

Zeng, A.; Song, S.; Nießner, M.; Fisher, M.; Xiao, J.; and Funkhouser, T. 2017. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1802–1811.

Zhang, J.; and Singh, S. 2014. LOAM: Lidar Odometry and Mapping in Real-time. In *Robotics: Science and Systems*, volume 2.

Zhong, Y. 2009. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, 689–696. IEEE.