MIT Libraries | DSpace@MIT

MIT Open Access Articles

SemAlign: Annotation-Free Camera-LiDAR Calibration with Semantic Alignment Loss

The MIT Faculty has made this article openly available. ***Please share*** how this access benefits you. Your story matters.

Citation: Liu, Zhijian, Tang, Haotian, Zhu, Sibo and Han, Song. 2021. "SemAlign: Annotation-Free Camera-LiDAR Calibration with Semantic Alignment Loss." 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).

As Published: 10.1109/iros51168.2021.9635964

Publisher: IEEE

Persistent URL: https://hdl.handle.net/1721.1/143675

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Terms of use: Creative Commons Attribution-Noncommercial-Share Alike

Massachusetts Institute of Technology

SemAlign: Annotation-Free Camera-LiDAR Calibration with Semantic Alignment Loss

Zhijian Liu*,1, Haotian Tang*,1, Sibo Zhu*,1 and Song Han[1]
https://semalign.mit.edu/

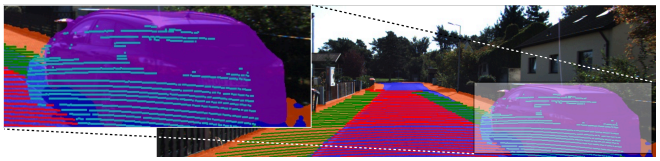
***Abstract*— Multi-sensor solution has been widely adopted in real-world robotics systems (*e.g.*, self-driving vehicles) due to its better robustness. However, its performance is highly dependent on the accurate calibration between different sensors, which is very time-consuming (*i.e.*, hours of human efforts) to acquire. Recent learning-based solutions partially address this yet still require costly ground-truth annotations as supervision. In this paper, we introduce a novel self-supervised *semantic alignment loss* to quantitatively measure the quality of a given calibration. It is well correlated with conventional evaluation metrics while it *does not require* ground-truth calibration annotations as the reference. Based on this loss, we further propose an annotation-free optimization-based calibration algorithm (*SemAlign*) that first estimates a coarse calibration with loss-guided initialization and then refines it with gradient-based optimization. SemAlign reduces the calibration time from *hours* of human efforts to only *seconds* of GPU computation. It not only achieves comparable performance with existing supervised learning frameworks but also demonstrates a much better generalization capability when transferred to a different dataset.**

I. Introduction

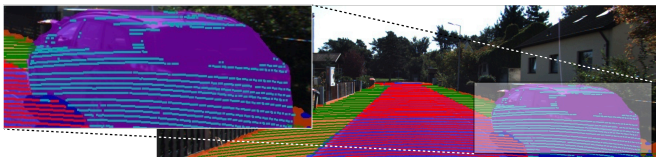
Real-world robotic systems (such as self-driving vehicles) are usually mounted with multiple sensors (including camera, radar and LiDAR scanners). This is because different sensors have different characteristics and failure modes: *e.g.*, camera sensor is very sensitive to the lighting condition while LiDAR sensor does not work reliably in the rainy weather. To improve the safety and robustness, most industrial perception solutions are based on multi-sensor fusion, whose performance is highly dependent on the accuracy of the calibration between sensors.

Accurate calibration, however, is notoriously hard to obtain in practice. On the one hand, it requires a lot of human efforts. Conventional calibration is based on matching the keypoints on the planar checkerboard from different sensors. During this process, human engineers have to constantly adjust the position and orientation of the checkerboard to collect data with different viewing angles. The whole procedure can take as much as *two hours* with human involved [19]. On the other hand, the time-consuming calibration is usually not a one-time effort. In the automotive industry, the mounted sensors need to be pre-calibrated by automakers during the manufacturing process and require to be re-calibrated regularly due to the potential bumps and thermal expansion/contraction.

To automate the tedious calibration process, researchers have introduced a series of learning-based solutions [9], [16], [24] that apply deep neural networks (DNNs) to extract the features for different sensors and predict the sensor calibration with multi-layer perceptrons (MLPs). However, most of these frameworks are based on *supervised learning* that still require the ground-truth calibration to guide the training. As ground-truth calibrations are expensive to obtain, most supervised learning frameworks have only seen a very limited amount of variations and usually cannot transfer well to the real-world data, hindering its practical usage.



(a) A bad calibration results in poor semantic alignment.



(b) A good calibration provides accurate semantic alignment.

Fig. 1: The quality of a calibration can be quantified by the **semantic overlapping** between 2D pixels and projected 3D points: *i.e.*, the number of 3D points projected into the 2D region with the same semantic category. We further relax this metric to make it differentiable so that we can directly optimize the calibration matrix based on gradient-based back propagation.

Our objective is to remove the need for the costly ground-truth calibrations. To achieve this, we need a quantitative metric to measure the quality of a given calibration. A perfect calibration should be able to precisely align the same instance in different sensors. Following this intuition, there have been explorations to align the keypoints extracted from different sensory inputs (*e.g.*, 2D images, depth maps rendered from 3D point clouds). However, it is difficult to extract the same keypoint (*i.e.*, with the same location) in different modalities. We will have to carefully design which keypoints to extract. Moreover, it might be challenging sometimes to accurately localize the keypoint due to the color and texture variations.

In this paper, we propose to use the *semantic alignment* since it is much easier to acquire given the large amount of available 2D and 3D segmentation models. As illustrated in Figure 1, the quality of a calibration can be quantitatively evaluated by **the semantic overlapping between 2D and**

∗ The first three authors have contributed equally to this work.

[1] Z. Liu, H. Tang, S. Zhu, and S. Han are with Massachusetts Institute of Technology, Cambridge, MA 02139, USA.

3D sensory inputs: *i.e.*, the number of 3D points that fall into the 2D region with the same category. Based on this intuition, we first define a counting-based semantic alignment metric to quantitatively measure the semantic overlapping and then relax it to the *differentiable semantic alignment loss* that can be incorporated into any gradient-based optimization framework. Our proposed loss function is highly correlated with conventional calibration metrics (*i.e.*, rotation errors), *without* the need of ground-truth calibrations as the reference.

We then further introduce a novel *annotation-free*, *training-free* calibration framework, *SemAlign*. Based on the semantic alignment, SemAlign first leverages *loss-guided initialization* to estimate a coarse calibration, and then iteratively refines it with *gradient-based optimization*. It achieves comparable performance with previous supervised learning approaches on the KITTI odometry dataset [4]. It also demonstrates great generalization capabilities since it does not rely on the labels: it offers precise calibrations on the KITTI detection dataset, where even the official calibration is not accurate.

The key contributions of this paper are as follows:

- We propose a novel calibration metric based on *semantic alignment*, which does not require ground-truth calibrations as reference. It can be easily incorporated into any gradient-based learning framework.
- We introduce a novel annotation-free calibration framework, *SemAlign*, which reduces the calibration time from *hours* of human labor to *seconds* of GPU computation.
- Our proposed framework achieves comparable results with previous supervised learning and has much better generalization ability when evaluated on the new dataset that has different distribution from the training data.

The remainder of the paper is structured as follows: we summarize the related work in Section II, formulate our metric and algorithm in Section III, describe our experimental results in Section IV, and provide concluding remarks in Section V.

II. Related Work

In this paper, we focus on the extrinsic calibration between LiDAR and camera: estimating the 6-DoF rigid body transformation matrix between these two sensors. Mainstream extrinsic calibration methods can be divided into two categories: target-based and target-less methods.

A. Target-Based Methods

For most target-based methods, the cost function is mainly built from features extracted from specific markers, such as checkerboard patterns [5], 3D boxes [23], self-printed circle [1], and custom polygonal planer board [22]. Target-based methods can be further divided into fully-automated and semi-automated approaches.

Fully-automated methods detect pre-defined target objects and then extract and match features without human intervention. For instance, Velas *et al*. [29] use the circular hole detection from planer boards, Park *et al*. [22] adopt white homogeneous target objects for calibration of LiDAR and camera, Guindel *et al*. [6] use a custom cut wood board with holes for LiDAR and camera to detect edges of the board, and researchers have also explored to use multiple checkerboards for corner and keypoint detection [5], [13]. Though fully-automated methods do not require human intervention, they rely on the existence of certain calibration targets, which can be complex setups such as the twelve checkerboards [5] and multiple LiDARs and stereo camera set [17].

Semi-automated methods are composed of many manual steps, such as having a human operator holding and moving the target calibration patterns in different locations and positions, manually localizing the target objects within the field-of-view of all sensors, then running calibration algorithms to adjust the parameters. For example, Kim *et al*. [12] require a human holding a checkerboard for LiDAR and camera to do edge and keypoints detection. Semi-automated methods are usually more accurate but very time-consuming [19]. The calibration results depend on the skill of the operators.

A well-calibrated sensor system periodically requires re-calibration due to temperature change, shaking, vibration, sensor deformation effects and other potential physical contacts. Therefore, the time-consuming target-based calibration methods with complex setup are inefficient when re-calibration is frequently needed.

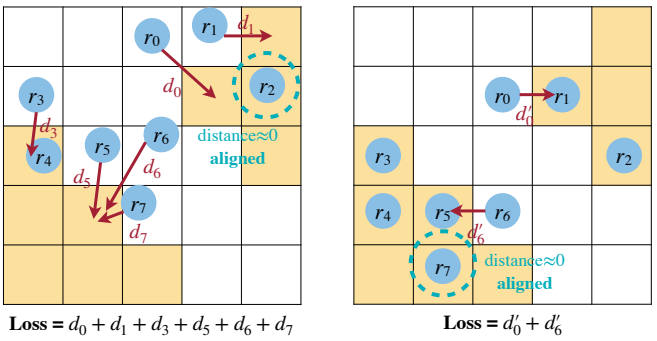
B. Target-Less Methods

Target-less methods extract features without the need for placement of calibration objects into the observed scene. Researchers have explored the usage of motion-based information [7], [8], [25], [26] and the usage of single-frame features for calibration [18], [20].

Motion-based methods first estimate trajectories of the camera and LiDAR sensors either by visual odometry or by employing IMU and GNSS measurements. Then, time-synchronized camera and LiDAR frames will be used to compute motion errors [8] or relative motions [25], with the assumption that all sensors are rigidly mounted. Motion-based calibration algorithms rely on accurate trajectory estimation, which is challenging in real-world applications due to sensor resolution, time synchronization and GPS failure.

Single-frame methods utilize correspondence of edge or line features extracted by LiDAR and camera for estimation of calibration parameters. One type of methods [10], [11], [18] require a tremendous amount of line correspondences to work well. Since far fewer line correspondences exist in outdoor scenes than in indoor scenes, line correspondence-based methods are usually more effective in indoor scenes than in driving scenarios. Other type of methods maximize mutual information of LiDAR intensity and camera grayscale values [20] or attempt to match 3D, 2D structural features from LiDAR and camera [21], [27].

End-to-end learning-based methods are also gaining popularity for extrinsic calibration [9], [16], [24]. These methods automatically estimate calibration parameters with sensory inputs. Annotated large datasets obtained from costly semi-automated calibration are used by those methods to train



(a) Before calibration (b) After calibration

Fig. 2: **Differentiable semantic alignment loss**. Here, blue color corresponds to projected 3D object pixels and yellow color represents 2D object pixels.

the supervised learning models. Despite being accurate on seen data, the trained models will usually have a hard time transferring and generalizing to new environments.

III. Method

In this section, we first define the objective of the calibration problem; we then introduce the self-supervised semantic alignment loss to measure the calibration quality; finally, we describe our optimization-based calibration algorithm.

A. Problem Definition

Based on the classical camera model, the homogeneous image coordinates of a 3D point can be obtained by multiplying its world coordinates with the product of camera intrinsic and extrinsic matrices. Since the camera intrinsic matrix is very easy to obtain, the objective of camera-LiDAR calibration then becomes to estimate the camera extrinsic matrix $\boldsymbol{M}_{\text{ex}} = \begin{bmatrix}\boldsymbol{R} & -\boldsymbol{R}\boldsymbol{t}\end{bmatrix}$, where $\boldsymbol{R}$ is the rotation matrix, and $\boldsymbol{t}$ is the translation vector.

As there is a one-to-one mapping between rotation matrices and 3-dimensional axis-angle representations $\boldsymbol{a}$ (whose norm is the magnitude of the rotation, and the direction is the rotation axis), the calibration problem can also be formulated as predicting the 6D calibration vector $\boldsymbol{c} = \begin{pmatrix}\boldsymbol{a} & \boldsymbol{t}\end{pmatrix}^{\text{T}}$.

B. Semantic Alignment Loss

Existing learning-based solutions are usually supervised by the reconstruction loss (*e.g.*, L1/L2 error) between the estimated and ground-truth calibrations. As obtaining calibration annotations is extremely costly, these models have only seen a limited amount of variation during training and therefore suffer from the poor generalization to the real-world data. To overcome limited annotations, we define a *self-supervised* loss function that accurately measures the calibration quality while, at the same time, does not require ground-truth calibration.

A good calibration should be able to precisely align the same instance in different sensors (see Figure 1): *e.g.*, the projected 3D LiDAR points of a car should be aligned with its corresponding 2D image pixels. Following this intuition, we introduce the *semantic alignment loss* to quantitatively measure the alignment between 2D pixels and projected 3D points that belong to the same semantic class (*e.g.*, car). The reason of using semantic category rather than instance ID to group pixels/points is due to the widely available pre-trained 2D/3D semantic segmentation models [14], [15].

For each semantic class c, we first gather all 2D image pixels of this class as $\mathcal{P}_c = \{\boldsymbol{p}_1, ..., \boldsymbol{p}_n\}$ and all projected 3D LiDAR points of this same class as $\mathcal{R}_c = \{\boldsymbol{r}_1, ..., \boldsymbol{r}_m\}$. Given that $\mathcal{P}_c$ and $\mathcal{R}_c$ are both 2D pixel sets on the same plane, one straightforward definition of the semantic alignment is the number of overlapping 2D and projected 3D pixels: $|\mathcal{P}_c \cap \mathcal{R}_c|$. However, this counting-based function is not differentiable and cannot provide the signal (*i.e.*, gradient) to improve the current calibration.

To this end, we propose to convert the non-differentiable *counting-based* definition to a differentiable *distance-based* formulation. For each $\boldsymbol{r}_k$ in $\mathcal{R}_c$, we find its nearest neighbor in $\mathcal{P}_c$ and denote it as $\boldsymbol{p}_l$. If $\mathcal{P}_c$ and $\mathcal{R}_c$ overlap with each other entirely, $\boldsymbol{p}_l$ should have exactly the same coordinate as $\boldsymbol{r}_k$, resulting in a distance of zero between them. On the other hand, a non-zero distance between $\boldsymbol{r}_k$ and $\boldsymbol{p}_l$ indicates a misalignment for $\boldsymbol{r}_k$. Based on this, we can then formulate the semantic alignment loss as the total distance between $\boldsymbol{r}_k$ and $\boldsymbol{p}_l$ for all $\boldsymbol{r}_k$'s in $\mathcal{R}_c$:

$$\mathcal{L}_{\text{SA}} = \sum_{k=1}^{m} \min_{l} ||\boldsymbol{r}_k - \boldsymbol{p}_l||_2^2. \tag{1}$$

We provide an example of the loss computation in Figure 2. Before the calibration (Figure 2a), some projected 3D points ($\boldsymbol{r}_0$, $\boldsymbol{r}_1$, $\boldsymbol{r}_3$, $\boldsymbol{r}_5$, $\boldsymbol{r}_6$, $\boldsymbol{r}_7$) are misaligned with 2D pixels (in yellow). This then results in a large semantic alignment loss with six non-zero terms. After the calibration (Figure 2b), only two projected 3D points ($\boldsymbol{r}_0$ and $\boldsymbol{r}_6$) are not aligned, and the total distance will then be reduced to $d'_0 + d'_6$.

As $\mathcal{L}_{\text{SA}}$ is differentiable with respect to each $\boldsymbol{r}_k$, and $\boldsymbol{r}_k$ is computed with a differentiable projection with the calibration matrix, we can directly refine the calibration vector $\boldsymbol{c}$ with any gradient-based optimization. Besides, $\mathcal{L}_{\text{SA}}$ can be considered as the *uni-directional* Chamfer distance between $\mathcal{R}_c$ and $\mathcal{P}_c$. Thus, we can make use of the existing libraries for Chamfer distance to efficiently compute our $\mathcal{L}_{\text{SA}}$.

Implementation Details. Our loss formulation is compatible with any existing 2D/3D segmentation model. In this paper, we adopt SDCNet [30] and SPVNAS [28] as our 2D and 3D segmentation models. They are pre-trained on Cityscapes [3] and SemanticKITTI [2], respectively. As for semantic classes, we only consider cars, roads and sidewalks in our experiments as the segmentation quality on these classes is better.

C. Optimization-Based Calibration

Since our semantic alignment loss *does not* require any annotation, we are able to support gradient-based optimization of calibration parameters. No ground-truth label is required, no training is required, and the *generalization capability* is improved compared to conventional learning-based calibration methods.

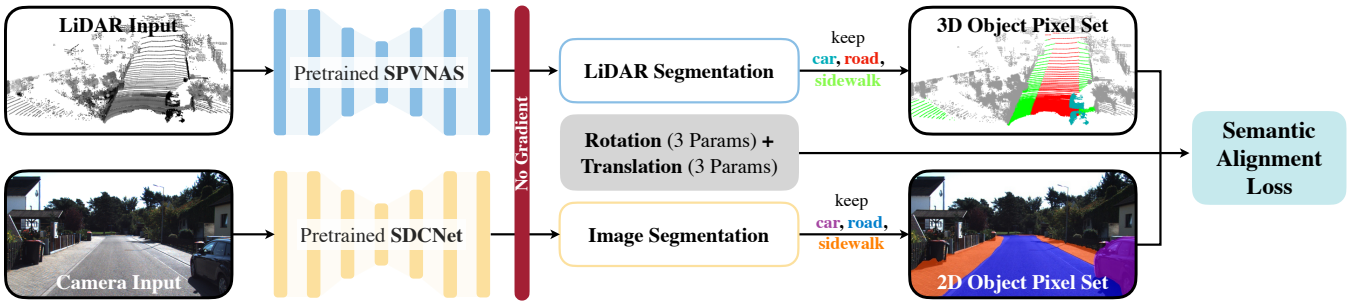


Fig. 3: Overview of **SemAlign**: we use pretrained 3D and 2d segmentation models to generate 3D / 2D *object pixel sets*, and then align them with the differentiable *Semantic Alignment Loss*. SemAlign *does not* require costly ground-truth calibration and time-consuming model training.

Loss-Guided Initialization. Given an initial 6D calibration vector (which might be way off from the correct calibration), we first sample N random 4×4 rigid body transformations and apply them to the initial 6D vector. For each transformed calibration vector, we use it to project 3D object points onto the 2D image plane, and compute the semantic alignment loss between 3D and 2D object pixel sets. We skip randomly transformed calibration vectors that lead to less than M 3D points projected onto the 2D image. As such, we avoid getting stuck in extremely bad starting points where no 3D points are in the camera FoV, and the semantic alignment loss is 0. We keep the transformed calibration vector that leads to lowest semantic alignment loss, and use it as the starting point for gradient-based refinement. Such loss-guided search can greatly improve the performance of gradient-based refinement especially when the initial calibration is largely off, as shown later in Table II.

Gradient-Based Optimization. With a better starting calibration vector, we perform gradient-based optimization to further improve the calibration precision. During forward propagation, we compute semantic alignment loss for road, sidewalk and cars separately, and normalize their scales to 1:1:1 dynamically: we notice that both the scale of semantic alignment loss for different classes and the scale of total semantic alignment loss for different scenes have drastic variations. With such loss normalization strategy, we empirically find that the same set of learning hyperparameters can generalize well to all samples and different datasets. This makes it easy to quickly deploy our method in real-world applications. The optimization propagates the gradient back to the 6D calibration parameters; the weights of the 2D and 3D segmentation networks are frozen. We also optimize the memory cost of semantic alignment loss by reusing GPU registers and shared memory to store intermediate distances in Equation 1. This avoids finding minimum distances in a large $|\mathcal{P}_c| \times |\mathcal{R}_c|$ matrix. As such, SemAlign is memory-efficient (≈1GB memory consumption) and can scale up well to high resolution camera and LiDAR inputs. We also notice that SemAlign usually converges very quickly within 500 to 1,000 iterations, which corresponds to merely 15-30 seconds on an NVIDIA RTX2080Ti GPU.

IV. Experiments

We evaluate our proposed calibration algorithm on both KITTI odometry and detection datasets [4]. We compare our SemAlign against three state-of-the-art *supervised learning* solutions [9], [16], [24].

A. KITTI Odometry

The KITTI odometry benchmark is composed of 22 sequences from different environments. Since we are using cars, sidewalks, and roads as our targets, frames without the existence of those objects are filtered out by running semantic segmentation networks on image and point cloud frames beforehand. We use sequences 16 to 21 (6,741 frames in total after filtering) to evaluate all methods. Each sequence is composed of RGB images captured by the left PointGrey Flea2 video camera and 3D LiDAR point cloud captured by Velodyne HDL-64E.

Optimization Details. We sample N = 5,000 random transformations for the starting point search and filter out all transformations that leads to less than M = 8,000 LiDAR points projected onto the camera FoV. We then use the Adam optimizer with a constant learning rate (10^{-3}) to optimize the 6D calibration vector for 1,000 iterations on each sample.

Evaluation Metrics. Since the KITTI odometry benchmark has very accurate calibration, we manually add random roll, pitch, yaw rotations within $[-10°, 10°]$ and $[-20°, 20°]$ to the original calibration and measure how well calibration algorithms can restore such random rotation. We report both quaternion angle distance and the Euler angle difference and provide both mean and median rotation errors.

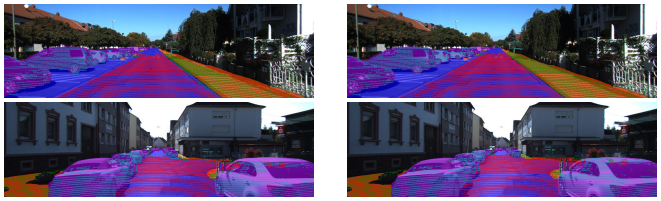
Main Results. We summarize our results in Table I in comparison with learning-based methods [9], [16], [24]. Notice that all previous methods require ground-truth calibration annotations and a standard supervised training pipeline before deployment. In contrast, SemAlign does not require calibration annotation and can directly run on unseen data. In Table I, we observe that even if SemAlign does not have access to calibration annotations, it still achieves the same level of performance comparing with state-of-the-art LCCNet [16]: notice that rotation error around 1° is almost

	Annotation Free	Training Free	Mis-Calibrated Rotation	Quaternion Rotation		Euler Rotation	
				Mean Error	Median Error	Mean Error	Median Error
RegNet [24]	✗	✗	$[-20, 20]$	–	–	0.28	–
CalibNet [9]	✗	✗	$[-10, 10]$	–	–	0.41	–
LCCNet [16]	✗	✗	$[-10, 10]$	0.59	0.45	0.39	0.22
			$[-20, 20]$	1.24	0.90	0.48	0.40
SemAlign (Ours)	✓	✓	$[-10, 10]$	1.14	0.46	0.62	0.23
			$[-20, 20]$	2.59	0.49	1.49	0.24

Tab. I: Quantitative results on KITTI odometry [4]. Our SemAlign does not require *costly calibration labels* or *time-consuming model training*, while achieving comparable performance with existing calibration methods. The results of the three baselines are all cited from the paper of LCCNet [16] (Tables 1 and 3).

(a) Initial calibration (b) After loss-guided initialization (c) After grad.-based optimization (d) Ground truth

Fig. 4: Example of online calibration on the KITTI odometry dataset: loss-guided initialization quickly improves the quality of initial calibration and gradient-based optimization further refines the rotation estimation.



(a) 1° Error (b) Ground truth

Fig. 5: 1° rotation error is close enough to the ground truth.

	Quaternion Error		Euler Error	
	±10°	±20°	±10°	±20°
SemAlign	**0.46**	**0.49**	**0.23**	**0.24**
w/o Loss-Guided Init.	0.53	0.93	0.27	0.46
w/o Constraint on #Points	0.51	7.97	0.25	4.72

Tab. II: Loss-guided initialization and constraint on the number of projected 3D points are critical to the performance.

invisible (shown in Figure 5). Our median error at larger mis-calibrated rotation angle (20°) is even better than LCCNet. We believe that the relatively larger mean error is caused by a small number of outliers: we find that SemAlign achieves worse performance after calibration on around 5% samples.

Ablation Analysis. We also analyze the effectiveness of different components in our algorithm. In Figure 4, loss-guided initialization quickly improves the quality of initial calibration. Gradient-based optimization further improves the calibration quality in border regions. Results in Table II also demonstrate that both loss-guided initialization and constraint on the number of projected 3D points (M) help improve the calibration performance significantly, especially at larger mis-calibration rotation angles.

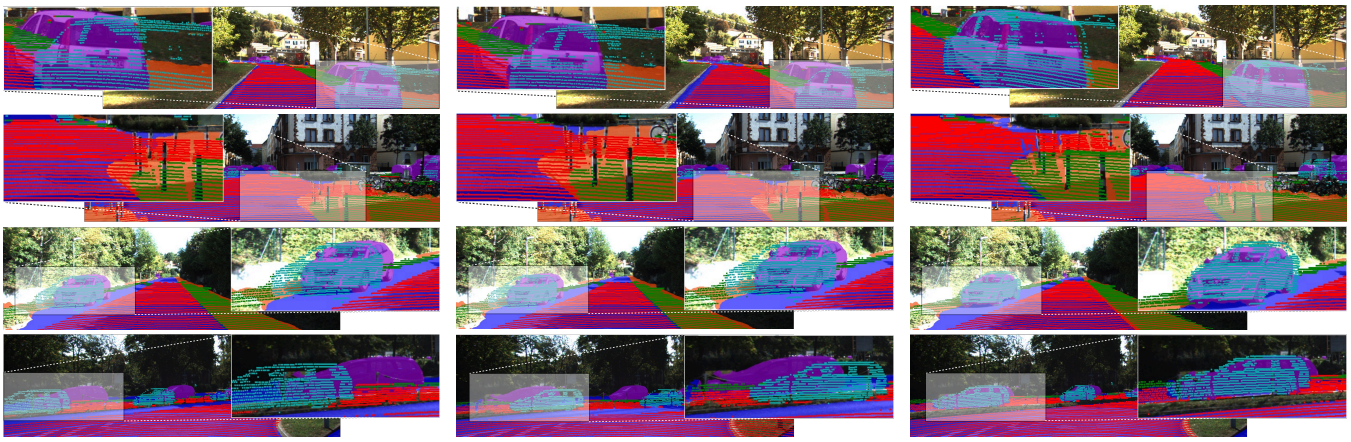
	Official Calibration	LCCNet	SemAlign
Geomean Loss Ratio	1.00	1.26	**0.74**

Tab. III: Results on KITTI detection [4] calibration. SemAlign has better generalization comparing with LCCNet.

B. KITTI Detection

We also perform experiments on the KITTI detection benchmark. The calibration performance is evaluated on the official validation set of KITTI consisting of 3,769 image-LiDAR pairs. We directly transfer pretrained LCCNet [16] models on KITTI odometry for comparison.

Evaluation Metrics. Unlike KITTI odometry benchmark where the ground-truth calibration annotation is accurate, calibrations on the KITTI detection dataset are often misaligned (*e.g.* Figure 6a is rendered from KITTI detection calibration annotations). Instead, we resort to *semantic alignment loss* as the criterion for the calibration quality. We quantitatively



(a) Official Calibration (b) LCCNet (c) **SemAlign (Ours)**

Fig. 6: Example of calibration results from KITTI detection: LCCNet is trained with supervised learning and fails to transfer to a different dataset, while SemAlign generalizes well on unseen data and significantly outperforms official calibration.

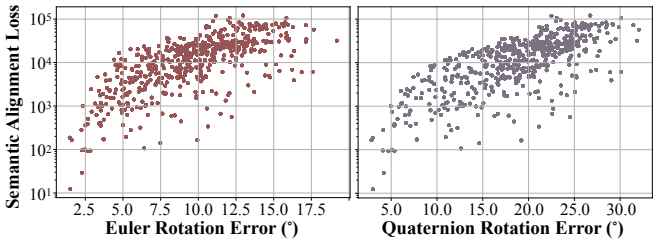


Fig. 7: Semantic Alignment Loss is loglinear to both Euler and quaternion rotation errors on the accurately annotated KITTI odometry benchmark.

study the correlation between semantic alignment loss and rotation, translation errors on well-annotated KITTI *odometry* benchmark and find in Figure 7 that semantic alignment loss is *loglinear* to both errors. Specifically, the Spearman's index is 0.72 and 0.71, indicating strong correlation. As a result, we believe that semantic alignment loss is a good surrogate for rotation and translation errors, and particularly useful when the ground truth annotation is not reliable. Since semantic alignment loss has large fluctuation across different samples, we choose to report the geomean ratio between semantic alignment loss after calibration and semantic alignment loss using official calibration on the entire dataset. The lower such ratio is, the better camera and LiDAR inputs are aligned.

Main Results. We compare the geomean semantic alignment loss ratio on KITTI detection to compare the generalization of different methods on a poorly annotated dataset. As shown in Table III, the official calibration provided by KITTI detection is inaccurate comparing with SemAlign due to poor alignment between 2D and 3D object pixel sets. The supervised learning model LCCNet [16] achieves even worse calibration comparing with the official calibration. This finding indicates that LCCNet is overfitting on KITTI odometry benchmark and cannot generalize well. Contrarily, our SemAlign achieves lower semantic alignment loss, which suggests better calibration quality.

In Figure 6 we visualize the calibration on KITTI detection. We find that even with the official annotated calibration, points belonging to the cars in 3D are projected to incorrect positions in 2D, which will potentially hinder the performance of camera-LiDAR fusion algorithms. The situation is similar for LCCNet [16]. In contrast, our SemAlign achieves better qualitative results comparing with LCCNet (see Figure 6c). Visualizations on KITTI detection further justifies our quantitative results in Table III and proves the generalizability of SemAlign.

V. Conclusion

In this paper, we study the annotation-free sensor calibration to reduce the human labor from the tedious calibration process. We first introduce a novel differentiable semantic alignment loss to measure the calibration quality without the need of ground-truth calibration as reference. Based on this loss, we further propose a novel annotation-free calibration framework, SemAlign, that reduces the calibration time from hours of human efforts to only seconds of GPU computation. Evaluated on the KITTI dataset, our annotation-free, self-supervised solution achieves comparable results with previous supervised learning frameworks. When transferred to the new dataset with different distribution from the training data, SemAlign demonstrates strong generalization ability and provides accurate calibration.

Despite encouraging results on the benchmark datasets, SemAlign still has some limitations. The formulation of semantic alignment loss is not sensitive to translation errors and requires both 2D and 3D segmentation models. Furthermore, gradient-based optimization cannot run in real time on GPUs, making it difficult to deploy SemAlign directly in latency-sensitive applications. We hope that our work can inspire future explorations that improve SemAlign in these aspects.

Acknowledgements. We thank Ji Lin for helpful discussions. This research is supported by NVIDIA, Samsung, Hyundai Motors and NSF.

REFERENCES

[1] Hatem Alismail, L Douglas Baker, and Brett Browning. Automatic Calibration of a Range Sensor and Camera System. In *International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission*, 2012.

[2] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Juergen Gall. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In *International Conference on Computer Vision*, 2019.

[3] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[4] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are We Ready For Autonomous Driving? The KITTI Vision Benchmark Suite. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[5] Andreas Geiger, Frank Moosmann, Ömer Car, and Bernhard Schuster. Automatic Camera and Range Sensor Calibration Using a Single Shot. In *International Conference on Robotics and Automation*, 2012.

[6] Carlos Guindel, Jorge Beltrán, David Martín, and Fernando García. Automatic Extrinsic Calibration for Lidar-Stereo Vehicle Sensor Setups. In *IEEE International Conference on Intelligent Transportation Systems*, 2017.

[7] Markus Horn, Thomas Wodtko, Michael Buchholz, and Klaus Dietmayer. Online Extrinsic Calibration based on Per-Sensor Ego-Motion Using Dual Quaternions. *IEEE Robotics and Automation Letters*, 2021.

[8] Kaihong Huang and Cyrill Stachniss. Extrinsic Multi-Sensor Calibration for Mobile Robots Using the Gauss-Helmert Model. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2017.

[9] Ganesh Iyer, R Karnik Ram, J Krishna Murthy, and K Madhava Krishna. CalibNet: Geometrically Supervised Extrinsic Calibration Using 3D Spatial Transformer Networks. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2018.

[10] Jaehyeon Kang and Nakju L Doh. Automatic Targetless Camera-LIDAR Calibration by Aligning Edge with Gaussian Mixture Model. *Journal of Field Robotics*, 2020.

[11] Archana Khurana and KS Nagla. Improved Auto-Extrinsic Calibration Between Stereo Vision Camera and Laser Range Finder. *International Journal of Image and Data Fusion*, 2020.

[12] Eung-su Kim and Soon-Yong Park. Extrinsic Calibration between Camera and LiDAR Sensors by Matching Multiple 3D Planes. *Sensors*, 2020.

[13] Kiho Kwak, Daniel F Huber, Hernan Badino, and Takeo Kanade. Extrinsic Calibration of a Single Line Scanning LiDAR and a Camera. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011.

[14] Zhijian Liu. Hardware-Efficient Deep Learning for 3D Point Cloud. Master's Thesis, 2020.

[15] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-Voxel CNN for Efficient 3D Deep Learning. In *Conference on Neural Information Processing Systems*, 2019.

[16] Xudong Lv, Boya Wang, Dong Ye, and Shuo Wang. LCCNet: LiDAR and Camera Self-Calibration using Cost Volume Network. *arXiv preprint arXiv:2012.13901*, 2020.

[17] Subodh Mishra, Gaurav Pandey, and Srikanth Saripalli. Extrinsic Calibration of a 3D-LIDAR and a Camera. In *IEEE Intelligent Vehicles Symposium*, 2003.

[18] Peyman Moghadam, Michael Bosse, and Robert Zlot. Line-Based Extrinsic Calibration of Range and Image Sensors. In *IEEE International Conference on Robotics and Automation*, 2013.

[19] Balázs Nagy and Csaba Benedek. On-the-Fly Camera and Lidar Calibration. *Remote Sensing*, 2020.

[20] Gaurav Pandey, James R McBride, Silvio Savarese, and Ryan M Eustice. Automatic Extrinsic Calibration of Vision and Lidar by Maximizing Mutual Information. *Journal of Field Robotics*, 2015.

[21] Chanoh Park, Peyman Moghadam, Soohwan Kim, Sridha Sridharan, and Clinton Fookes. Spatiotemporal Camera-LiDAR Calibration: A Targetless and Structureless Approach. *IEEE Robotics and Automation Letters*, 2020.

[22] Yoonsu Park, Seokmin Yun, Chee Sun Won, Kyungeun Cho, Kyhyun Um, and Sungdae Sim. Calibration Between Color Camera and 3D LIDAR Instruments with a Polygonal Planar Board. *Sensors*, 2014.

[23] Zoltán Pusztai, Iván Eichhardt, and Levente Hajder. Accurate Calibration of Multi-LiDAR-Multi-Camera Systems. *Sensors*, 2018.

[24] Nick Schneider, Florian Piewak, Christoph Stiller, and Uwe Franke. RegNet: Multimodal Sensor Registration Using Deep Neural Networks. In *IEEE Intelligent Vehicles Symposium*, 2017.

[25] Yiu Cheung Shiu and Shaheen Ahmad. Calibration of Wrist-Mounted Robotic Sensors by Solving Homogeneous Transform Equations of the Form AX=XB. 1987.

[26] Klaus H Strobl and Gerd Hirzinger. Optimal Hand-Eye Calibration. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006.

[27] Levente Tamas and Zoltan Kato. Targetless Calibration of a Lidar - Perspective Camera Pair. In *IEEE International Conference on Computer Vision Workshop*, 2013.

[28] Haotian Tang, Zhijian Liu, Shengyu Zhao, Yujun Lin, Ji Lin, Hanrui Wang, and Song Han. Searching Efficient 3D Architectures with Sparse Point-Voxel Convolution. In *European Conference on Computer Vision*, 2020.

[29] Martin Vel'as, Michal Španěl, Zdeněk Materna, and Adam Herout. Calibration of RGB Camera with Velodyne LiDAR. 2014.

[30] Yi Zhu, Karan Sapra, Fitsum A. Reda, Kevin J. Shih, Shawn D. Newsam, Andrew Tao, and Bryan Catanzaro. Improving Semantic Segmentation via Video Propagation and Label Relaxation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.