

Self-Calibration of the Offset Between GPS and Semantic Map Frames for Robust Localization

Wei-Kang Tseng, Angela P. Schoellig, Timothy D. Barfoot
Institute for Aerospace Studies
University of Toronto
Toronto, Canada

gorden.tseng@mail.utoronto.ca, schoellig@utias.utoronto.ca, tim.barfoot@utoronto.ca

Abstract—In self-driving, standalone GPS is generally considered to have insufficient positioning accuracy to stay in lane. Instead, many turn to LIDAR localization, but this comes at the expense of building LIDAR maps that can be costly to maintain. Another possibility is to use semantic cues such as lane lines and traffic lights to achieve localization, but these are usually not continuously visible. This issue can be remedied by combining semantic cues with GPS to fill in the gaps. However, due to elapsed time between mapping and localization, the live GPS frame can be offset from the semantic map frame, requiring calibration. In this paper, we propose a robust semantic localization algorithm that self-calibrates for the offset between the live GPS and semantic map frames by exploiting common semantic cues, including traffic lights and lane markings. We formulate the problem using a modified Iterated Extended Kalman Filter, which incorporates GPS and camera images for semantic cue detection via Convolutional Neural Networks. Experimental results show that our proposed algorithm achieves decimetre-level accuracy comparable to typical LIDAR localization performance and is robust against sparse semantic features and frequent GPS dropouts.

I. INTRODUCTION

In autonomous driving applications, semantic maps have proven to be an invaluable component for most self-driving cars. They provide important prior knowledge of the surrounding environment, including the locations of drivable lanes, traffic lights, and traffic signs, as well as the traffic rules. This information is crucial for real-time behavioural planning of the vehicle under various traffic scenarios.

In order to effectively utilize semantic maps, the vehicle must be localized in the map frame down to decimetre accuracy. This proves to be challenging for the Global Positioning System (GPS), where even the best corrected version of GPS is generally considered inadequate in achieving the required accuracy consistently. Furthermore, GPS suffers from signal dropouts in situations such as inside tunnels or in dense urban environments. In light of these issues, many self-driving systems have adopted LIDAR (Light Detection and Ranging) localization methods, which require the construction of LIDAR maps prior to driving in a certain area. LIDAR localization has demonstrated great success in satisfying the stringent requirements of autonomous driving [1]–[3], but this comes at the cost of building detailed

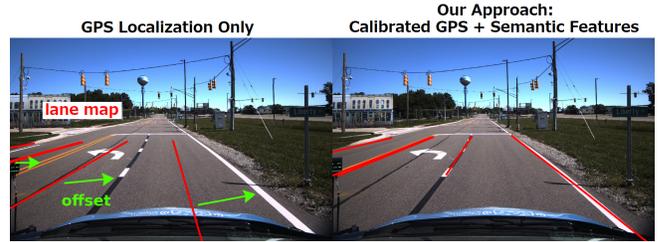


Figure 1. Vehicle localization using uncalibrated GPS (left) compared to our approach (right). The red lines are the projected lane boundaries from the semantic map. Our approach is able to self-calibrate for the GPS-to-map offset and achieve alignment between the observed lane markings and the projected lane boundaries [4].

geometric models of the world and keeping them up to date solely for the purpose of localization. Moreover, because the autonomous driving system, and in particular the planning component, ultimately requires the vehicle’s location with respect to the semantic map, it requires the additional step of aligning the LIDAR maps with the semantic maps.

An alternative to LIDAR localization is to directly take advantage of the semantic maps for localization, which the self-driving vehicle already utilizes for path planning and behavioural decision making. Through the detection of semantic cues (e.g., in the vehicle’s camera images) that are also present in the semantic maps, the vehicle location can be inferred. A major downside of such an approach is that these semantic cues are fairly sparse and not always present in enough numbers to ensure reliable localization. A potential solution is to adopt a hybrid approach that combines GPS and semantic cues. However, a new problem arises: the offset between the semantic map frame and the GPS frame, in which the vehicle position is reported, must be known accurately before fusing the two sources of information. This offset is a common issue and emerges because the semantic maps are aligned to the global frame using GPS data gathered at a different time/day than when the live drive occurs. Therefore, due to different positioning of the satellites in the sky and varying atmospheric conditions [5], there will be a map offset requiring calibration such that the GPS frame aligns with the semantic map frame. Manual calibration is generally not practical and reliable.

As an illustrative example, aUToronto, the team that won the self-driving competition by SAE International in 2019 [4], experienced an uncalibrated GPS-to-map offset in the magnitude of a few metres, which was corrected manually just in time for the competition run, see Figure 1.

To address these challenges, we propose a robust localization algorithm that integrates GPS and semantic cues while performing self-calibration of the offset between the GPS and semantic map frames. By folding the offset into our state estimation, we can properly fuse the two sources of information while benefitting from both. For this work, we assume detection of semantic cues using a front-facing monocular camera, and formulate the localization problem as a modified Iterated Extended Kalman Filter (IEKF), which improves upon the linearization of EKF. The system architecture is summarized in Figure 2.

The proposed approach has minimal computational impact because GPS is low-cost to process, and common semantic cues such as lane markings and traffic lights are already tracked for the purpose of vehicle behavioural planning, so the added cost of using them is also low. The result is an accurate and robust self-driving localization pipeline that uses GPS to fill in the gaps between sparse semantic observations, avoids the need for expensive maps specifically for localization, and relies on features in the environment that are actively maintained and designed to be highly visible. Experimental results in an urban environment using the Carla simulator [6] as well as on a real-world dataset collected by aUToronto during the SAE AutoDrive competition show that we are able to achieve 3 cm lateral and 5 cm longitudinal accuracy on average, and also maintain similar performance with frequent GPS dropouts.

The paper is organized as follows. Section II summarizes the related work on vehicle localization. Section III describes the preprocessing required for the semantic cues. Section IV presents the mathematical formulation of the localization algorithm. Section V provides the experimental results. Finally, Section VI concludes the paper.

II. RELATED WORK

1) *LIDAR Localization*: One of the most popular localization approaches in self driving is LIDAR localization [7]–[10]. By constructing a database of the detailed geometry of the environment in advance, localization can be achieved using a point cloud registration algorithm, which matches the LIDAR scans against the database at test time. Because the localization performance greatly depends on the accuracy of the database in capturing the ever-changing appearance of the world, the database needs to be frequently updated. In response, many have developed algorithms that extract features that are more invariant to environmental changes in the LIDAR data [11]–[13]. More recently, [14] proposed a novel learning-based approach that directly takes LIDAR point clouds as inputs and learns descriptors for matching

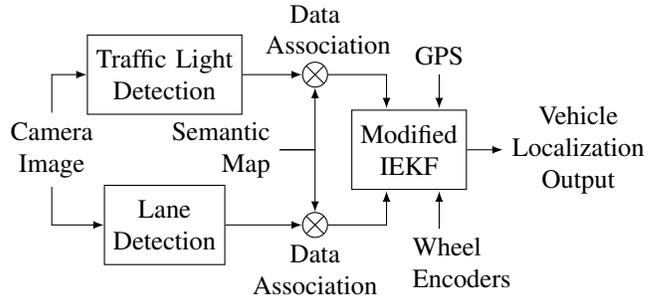


Figure 2. System architecture of our proposed localization pipeline. The camera image is passed through the lane and traffic light detectors. The data association step finds the correspondences between detection results and the semantic map projected into the image space. The results of the data association are then fused with GPS and wheel encoders in a modified IEKF to produce the final localization output.

in various driving scenarios. While these methods help mitigate the impact of outdated LIDAR database, they do not fundamentally address the issue of needing to maintain a separate database solely for localization.

2) *Semantic Localization*: Semantic localization exploits various common roadside semantic cues present in the semantic maps to achieve vehicle localization. In contrast to LIDAR localization, this method conveniently makes use of the same semantic maps already required by the autonomous vehicle for planning purposes. Therefore, no maintenance of a separate database of the environment is required. Among the various types of semantic cues, lane markings are most commonly utilized because they are abundant and provide important clues that keep the vehicle in the correct lane [15]–[19]. However, since lane markings tend to run parallel to the vehicle heading, the longitudinal localization accuracy is usually worse than lateral accuracy. Besides lane markings, other types of semantic cues have been exploited as well, including stop lines [20], [21], other road markings [22]–[24], traffic lights [25], [26], and traffic signs [27]–[30]. A common issue that all types of semantic cues suffer from is sparsity. In response, approaches that combine multiple types of semantic cues have been proposed, most of which include lane markings in combination with traffic lights or traffic signs [31]–[33]. Many of the semantic localization papers referenced in this section have incorporated GPS into their localization pipelines, but none of them addressed a possible offset between GPS and semantic map frames due to reasons discussed above, presumably because the GPS offset has been manually corrected prior to experiments. However, as experienced by aUToronto, manual GPS calibration is often unreliable, and can lead to localization failures [4].

3) *GPS Calibration with Semantic Cues*: In this work, the GPS measurements are regarded as reporting the vehicle positions with respect to a GPS frame, which is at an offset from the semantic map frame. Alternatively, we can treat the GPS as if it directly reports the vehicle position in the semantic map frame, but with a systematic bias. Some prior

works took this fact into account when developing their semantic localization pipelines. For instance, [34] simply modelled the GPS errors as a random constant since the change in the GPS bias is small. A more sophisticated model utilizing autoregressive process such as a random walk was shown by [35] to achieve superior performance compared to the random constant model, and was similarly adopted by [36] and [37]. All of these approaches only adopted road markings as the semantic cues. While our approach is similar in spirit to these papers, there are also notable differences, including the addition of traffic lights as part of the semantic cues, and their detections using Convolutional Neural Networks (CNNs). Furthermore, we formulate the localization problem in 3D, process the semantic cue detection results directly in the image space rather than in bird’s-eye view, and ensure robust localization against frequent GPS dropouts.

III. SEMANTIC CUE PREPROCESSING

1) *Semantic Map*: Our localization algorithm utilizes a lightweight HD semantic map that consists of a lane graph and traffic light locations. A lane graph is a set of polylines that defines all the lane boundaries of the road network. It corresponds to visually distinctive lane markings, which can be easily identified in a camera image. The traffic lights are treated as point landmarks with the coordinates of their centres recorded in the semantic map. In this work, we assume the semantic map has been provided.

2) *Traffic Light Detection*: Traffic lights appear regularly at road intersections and provide useful information for longitudinal localization. A traffic light CNN detector [38] outputs bounding boxes that locate traffic lights in the camera images. The centre of each bounding box is then obtained as the observed point landmark of a traffic light. To filter out false detections, only bounding boxes with a high confidence level are included for localization.

Before the traffic light detections can be made useful for localization, a data association scheme must first be devised to correctly associate the detections in the image with corresponding traffic lights in the semantic map. This is achieved by first projecting the locations of all nearby traffic lights in the semantic map to the image space using the estimated vehicle position. We then apply Iterative Closest Point followed by nearest neighbour to obtain the desired associations. Detections that have no nearby associations within a certain distance threshold are identified as outliers and discarded. Figure 3 illustrates the results of the traffic light data association process.

3) *Lane Marking Detection*: Lane markings are one of the most common type of semantic cues that primarily help with lateral localization. A lane marking CNN detector [39] produces a mask of the camera image that classifies each pixel as lane marking with a probability. A binary mask is then obtained by applying a probability threshold. Only the bottom portion of the mask, where the lane markings

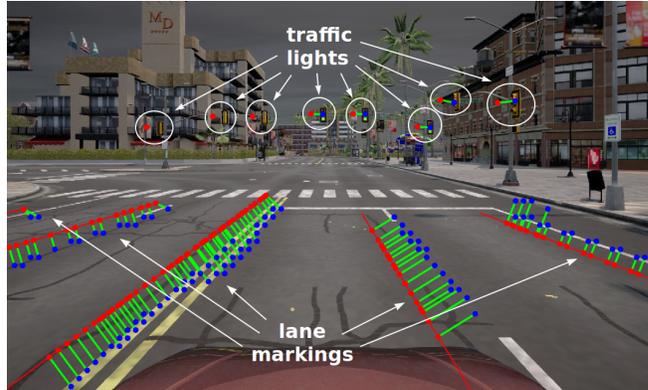


Figure 3. Data association process of traffic lights and lane markings. The red lines and points are known positions of semantic cues projected into the image using the estimated vehicle position; the blue points are semantic cue detections; and the green lines indicate the matching results. A few incorrect matches of outlier lane pixels can be observed on the right due to their proximity to a projected lane line.

are closer to the vehicle and can be clearly identified, is retained. The resulting image coordinates of the pixels classified as lane markings are then evenly subsampled to reduce computational burden.

The data association process for lane markings also begins by projecting all nearby lane lines from the semantic map to the image space. Next, the subsampled lane pixels are each matched to their closest projected lines. Outlier lane pixels without nearby matches are discarded. Figure 3 demonstrates the lane marking matching results. Given all the lane pixels matched to each projected line, a straight line is fitted using least squares in image space. Finally, data association is obtained between the pairs of fitted lines from lane marking detection and projected lines from the semantic map.

IV. VEHICLE LOCALIZATION

A. Problem Setup

We formulate the semantic localization problem with GPS offset by first discretizing the time denoted by subscript k . There are three reference frames. \mathcal{F}_M is the semantic map frame, $\mathcal{F}_{V,k}$ is attached to a moving vehicle, and $\mathcal{F}_{G,k}$ is the GPS frame, which is at an offset from \mathcal{F}_M . We then have three corresponding transformation matrices between the frames. $\mathbf{T}_{VG,k} \in SE(3)$ is the GPS measurement of the pose of vehicle, $\mathbf{T}_{GM,k} \in SE(3)$ is the GPS-to-map offset, which needs to be estimated for self-calibration, and $\mathbf{T}_{VM,k} \in SE(3)$ is the pose of vehicle with respect to the semantic map, which we ultimately desire. Figure 4 illustrates the described problem setup. At time step k , the j -th semantic cue, P^j , detected by the onboard camera has the pixel coordinates, $\mathbf{p}_{I,k}^j$, as well as its known location in the map frame, $\mathbf{p}_M^j \in \mathbb{R}^3$, obtained from the semantic map. Using $\mathbf{T}_{VM,k}$, we can transform and project the known location, \mathbf{p}_M^j , to the image space and obtain the reprojection error for localization and GPS-to-map offset calibration.

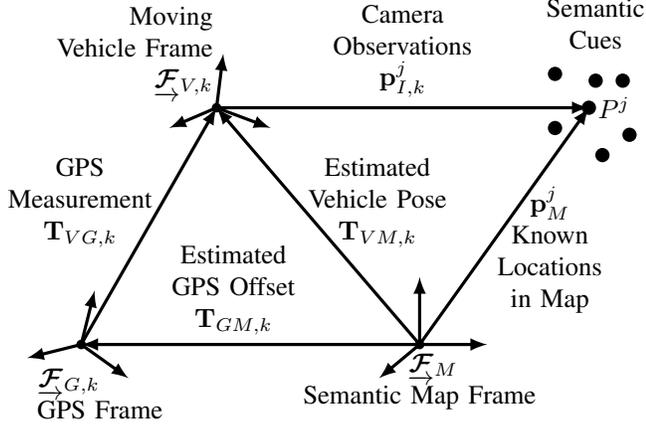


Figure 4. Definition of reference frames for the localization problem with semantic cues and offset between GPS and semantic map frames.

We adopt the mathematical notations from [40]. Notably, \wedge is an operator associated with the Lie algebra for $SE(3)$:

$$\xi^\wedge = \begin{bmatrix} \boldsymbol{\rho} \\ \boldsymbol{\phi} \end{bmatrix}^\wedge := \begin{bmatrix} \boldsymbol{\phi}^\wedge & \boldsymbol{\rho} \\ \mathbf{0}^T & 0 \end{bmatrix} \in \mathbb{R}^{4 \times 4}, \quad \boldsymbol{\rho}, \boldsymbol{\phi} \in \mathbb{R}^3, \quad (1)$$

and also for $SO(3)$:

$$\boldsymbol{\phi}^\wedge = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix}^\wedge := \begin{bmatrix} 0 & -\phi_3 & \phi_2 \\ \phi_3 & 0 & -\phi_1 \\ -\phi_2 & \phi_1 & 0 \end{bmatrix}. \quad (2)$$

B. Process and Observation Models

1) *Vehicle Dynamics Process Model*: We adopt the white-noise-on-acceleration model [41], which also estimates the vehicle velocity $\boldsymbol{\omega}_k \in \mathbb{R}^6$ in the vehicle frame $\mathcal{F}_{V,k}$:

$$\mathbf{T}_{VM,k} = \exp(\mathbf{w}_{VM}^\wedge) \exp(\Delta t_k \boldsymbol{\omega}_{k-1}^\wedge) \mathbf{T}_{VM,k-1}, \quad (3)$$

$$\boldsymbol{\omega}_k = \boldsymbol{\omega}_{k-1} + \mathbf{w}_\omega, \quad (4)$$

where $\Delta t_k = t_k - t_{k-1}$ is the time interval, and $\mathbf{w}_{VM}, \mathbf{w}_\omega \in \mathbb{R}^6$ are zero mean Gaussian process noises for vehicle pose and velocity, respectively. As formulated in [41], \mathbf{w}_{VM} and \mathbf{w}_ω are correlated with the joint distribution

$$\begin{bmatrix} \mathbf{w}_{VM} \\ \mathbf{w}_\omega \end{bmatrix} \sim \mathcal{N} \left(\mathbf{0}_{12 \times 1}, \underbrace{\begin{bmatrix} \frac{1}{3} \Delta t_k^3 \mathbf{Q}_C & \frac{1}{2} \Delta t_k^2 \mathbf{Q}_C \\ \frac{1}{2} \Delta t_k^2 \mathbf{Q}_C & \Delta t_k \mathbf{Q}_C \end{bmatrix}}_{\mathbf{Q}_{VM}} \right), \quad (5)$$

where the tunable parameter $\mathbf{Q}_C \in \mathbb{R}^{6 \times 6}$ is a diagonal matrix with non-zero values in its first and last diagonal entries corresponding to the vehicle's translational and rotational accelerations in the vehicle frame, which are along the x -axis (tangential to its motion) and about the z -axis (normal to the ground plane), respectively.

2) *GPS Offset Process Model*: The GPS-to-map offset, which very gradually varies over time, is modelled as a random walk. This is a convenient way to handle the estimation of such a time-dependent unknown parameter:

$$\mathbf{T}_{GM,k} = \exp(\mathbf{w}_{GM}^\wedge) \mathbf{T}_{GM,k-1}, \quad (6)$$

where $\mathbf{w}_{GM} \sim \mathcal{N}(\mathbf{0}_{6 \times 1}, \mathbf{Q}_{GM})$ is the process noise.

3) *GPS Observation Model*: In this work, a GPS measurement refers to a preprocessed quantity that is a three-dimensional transformation matrix $\mathbf{T}_{VG,k}$ with three degrees of freedom each in position and orientation. This is the output of commercial GPS-based localization solutions such as Applanix POS LV, which integrates GPS and IMU information. The observation model of GPS measurement, $\mathbf{T}_{VG,k}$, is

$$\mathbf{T}_{VG,k} = \exp(\mathbf{n}_{VG}^\wedge) \mathbf{T}_{VM,k} \mathbf{T}_{GM,k}^{-1}, \quad (7)$$

where the measurement noise is $\mathbf{n}_{VG} \sim \mathcal{N}(\mathbf{0}_{6 \times 1}, \mathbf{R}_{VG})$.

4) *Traffic Light Observation Model*: For the j -th traffic light pixel measurement, the observation model is simply

$$\mathbf{p}_{I,k}^j = \mathbf{g}(\mathbf{p}_M^j, \mathbf{T}_{VM,k}) + \mathbf{n}_{\text{light}}, \quad (8)$$

where $\mathbf{g}(\cdot)$ projects the known traffic light location \mathbf{p}_M^j from semantic map to image space of the onboard camera given vehicle pose estimation, $\mathbf{T}_{VM,k}$. The pixel measurement noise $\mathbf{n}_{\text{light}} \sim \mathcal{N}(\mathbf{0}_{2 \times 1}, \mathbf{R}_{\text{light}})$ is assumed to be Gaussian.

5) *Lane Marking Observation Model*: Using the data association process described in Section III-3, we obtain lane marking observations as straight lines in the image space, which can each be represented by two distinct points on the line. For the j -th observed line, the points are selected by choosing two different vertical pixel coordinates $\mathbf{y}_I^j = [y_1^j \ y_2^j]^T$. The observation model for the corresponding horizontal pixel coordinates is

$$\mathbf{x}_{I,k}^j = [x_{1,k}^j \ x_{2,k}^j]^T = \mathbf{f}(\mathbf{g}(\boldsymbol{\ell}_M^j, \mathbf{T}_{VM,k}), \mathbf{y}_I^j) + \mathbf{n}_{\text{lane}}, \quad (9)$$

where $\boldsymbol{\ell}_M^j$ is the known lane line from the semantic map, and $\mathbf{f}(\cdot)$ produces the horizontal pixel coordinates given \mathbf{y}_I^j . The associated measurement noise is $\mathbf{n}_{\text{lane}} \sim \mathcal{N}(\mathbf{0}_{2 \times 1}, \mathbf{R}_{\text{lane}})$.

6) *Wheel Encoders Observation Model*: The onboard wheel encoders provide measurements on the vehicle's longitudinal velocity, v_k , and angular velocity, ω_k , in yaw. Wheel encoders are included to improve robustness against GPS dropouts. The observation model is

$$\boldsymbol{\omega}_{\text{wheel},k} = [v_k \ \omega_k]^T = \mathbf{h}_\omega(\boldsymbol{\omega}_k) + \mathbf{n}_\omega, \quad (10)$$

where $\mathbf{h}_\omega(\cdot)$ extracts the corresponding vehicle velocities from $\boldsymbol{\omega}_k$, and the noise term is $\mathbf{n}_\omega \sim \mathcal{N}(\mathbf{0}_{2 \times 1}, \mathbf{R}_\omega)$.

7) *Pseudo-Measurement Observation Model*: Some pseudo-measurements are introduced by leveraging the fact that the vehicle always stays on the ground, which is assumed to be the xy -plane of the map frame. Therefore, its elevation, roll, and pitch are all soft-constrained to zero with respect to the map. This effectively reduces the localization problem down to a 2D space while maintaining the problem formulation in 3D. Additionally, we assume that the rear wheels do not slip sideways, thus the lateral vehicle velocity is also near zero. The observation models of the pseudo-measurements are straightforward:

$$u_{\text{pseudo}} = h_u(\mathbf{T}_{VM,k}, \boldsymbol{\omega}_k) + n_u, \quad (11)$$

where u is elevation, roll, or pitch of the vehicle with respect to the map frame, or lateral velocity in the vehicle frame. $h_u(\cdot)$ extracts the corresponding quantity from the current vehicle pose and velocity estimations. The corresponding pseudo-measurement noise is $n_u \sim \mathcal{N}(0, r_u)$, with small r_u to effectively constraint each quantity.

C. Modified IEKF Formulation

1) *Prediction Step*: The prediction step follows the standard IEKF formulation by jointly estimating the vehicle pose, velocity, and GPS offset using the process models (3), (4), and (6). We linearize them to obtain the predicted means

$$\check{\mathbf{T}}_{VM,k} = \exp(\Delta t_k \hat{\boldsymbol{\omega}}_{k-1}) \hat{\mathbf{T}}_{VM,k-1}, \quad (12)$$

$$\check{\boldsymbol{\omega}}_k = \hat{\boldsymbol{\omega}}_{k-1}, \quad (13)$$

$$\check{\mathbf{T}}_{GM,k} = \hat{\mathbf{T}}_{GM,k-1}, \quad (14)$$

as well as the predicted joint covariance matrix

$$\check{\mathbf{P}}_k = \mathbf{F}_{k-1} \hat{\mathbf{P}}_{k-1} \mathbf{F}_{k-1}^T + \begin{bmatrix} \mathbf{Q}_{VM} & \mathbf{0}_{12 \times 6} \\ \mathbf{0}_{6 \times 12} & \mathbf{Q}_{GM} \end{bmatrix}, \quad (15)$$

where \mathbf{F}_k is the combined Jacobian matrix of the linearized process models (3), (4), and (6) at time step k .

2) *Correction Step*: The iterative correction step of IEKF is modified by replacing it with a batch optimization formulation with time window size of one (the current time step) [41]. The cost function to optimize is $J = J_v + J_y$, where

$$J_v = \frac{1}{2} \mathbf{e}_{v,k}^T \check{\mathbf{P}}_k^{-1} \mathbf{e}_{v,k}, \quad (16)$$

$$J_y = \sum_i \frac{1}{2} \mathbf{e}_{y,k}^i{}^T \mathbf{R}^{i-1} \mathbf{e}_{y,k}^i, \quad (17)$$

are the prior and measurement cost terms, respectively. The prior errors, $\mathbf{e}_{v,k}$, are computed using the predicted means, $\check{\mathbf{T}}_{VM,k}$, $\check{\boldsymbol{\omega}}_k$, and $\check{\mathbf{T}}_{GM,k}$, from the prediction step (12), (13), and (14). This encourages a consistent trajectory that respects the vehicle dynamics between the estimated poses. The overall measurement cost term J_y is a sum of the cost terms derived from the sensor measurements, including GPS (7), semantic cues (8)(9), wheel encoders (10), and pseudo-measurements (11). For the i -th measurement, $\mathbf{e}_{y,k}^i$ is the measurement error and \mathbf{R}^i is the associated observation covariance matrix: \mathbf{R}_{VG} , $\mathbf{R}_{\text{light}}$, \mathbf{R}_{lane} , \mathbf{R}_{ϖ} , or r_u .

In order to minimize the impact of bad data association of semantic cues, a Cauchy M-estimator is deployed [42]. For each semantic cue measurement cost term in (17), \mathbf{R}^{i-1} is replaced by $\mathbf{Y}_k^{i-1} = (1 + \mathbf{e}_{y,k}^i{}^T \mathbf{R}^{i-1} \mathbf{e}_{y,k}^i)^{-1} \mathbf{R}^{i-1}$. Given a reasonably well initialized vehicle position, this scheme effectively prevents localization failures by scaling down the importance of outliers, which produce large measurement errors, via the associated \mathbf{Y}_k^{i-1} .

The cost function, J , is optimized using the Gauss-Newton method. In each iteration, we first obtain the combined Jacobian matrices, \mathbf{E}_k and \mathbf{G}_k , after linearizing

the error terms, $\mathbf{e}_{v,k}$ and $\mathbf{e}_{y,k}^i$, about the operating point, $\mathbf{x}_{\text{op}} = \{\hat{\mathbf{T}}_{VM,\text{op},k}, \hat{\boldsymbol{\omega}}_{\text{op},k}, \hat{\mathbf{T}}_{GM,\text{op},k}\}$. We then substitute the linearized error terms into the cost function and set its derivative with respect to the perturbation term, $\delta \mathbf{x} = [\delta \boldsymbol{\xi}_{VM,k}^T \quad \delta \boldsymbol{\omega}_k^T \quad \delta \boldsymbol{\xi}_{GM,k}^T]^T \in \mathbb{R}^{18}$, to zero. This produces the update equation

$$\underbrace{(\mathbf{E}_k^T \check{\mathbf{P}}_k^{-1} \mathbf{E}_k)}_{\mathbf{A}_{\text{pri}}} + \underbrace{(\mathbf{G}_k^T \mathbf{R}_k^{-1} \mathbf{G}_k)}_{\mathbf{A}_{\text{meas}}} \delta \mathbf{x}^* = \mathbf{E}_k^T \check{\mathbf{P}}_k^{-1} \mathbf{e}_{v,k} + \mathbf{G}_k^T \mathbf{R}_k^{-1} \mathbf{e}_{y,k}, \quad (18)$$

where $\delta \mathbf{x}^*$ is the optimal perturbation, and \mathbf{R}_k is a block diagonal matrix that combines all \mathbf{R}^i and \mathbf{Y}_k^i . Solving for $\delta \mathbf{x}^*$, we update the operating point as follows:

$$\hat{\mathbf{T}}_{VM,\text{op},k} \leftarrow \exp((\delta \boldsymbol{\xi}_{VM,k}^*)^\wedge) \hat{\mathbf{T}}_{VM,\text{op},k}, \quad (19)$$

$$\hat{\boldsymbol{\omega}}_{\text{op},k} \leftarrow \hat{\boldsymbol{\omega}}_{\text{op},k} + \delta \boldsymbol{\omega}_k^*, \quad (20)$$

$$\hat{\mathbf{T}}_{GM,\text{op},k} \leftarrow \exp((\delta \boldsymbol{\xi}_{GM,k}^*)^\wedge) \hat{\mathbf{T}}_{GM,\text{op},k}. \quad (21)$$

Finally, the results of the correction step, $\hat{\mathbf{T}}_{VM,k}$, $\hat{\boldsymbol{\omega}}_k$, and $\hat{\mathbf{T}}_{GM,k}$, are output after convergence. The corresponding covariance is computed as $\hat{\mathbf{P}}_k = (\mathbf{A}_{\text{pri}} + \mathbf{A}_{\text{meas}})^{-1}$ from the last iteration.

V. EXPERIMENTS

Quantitative simulations were carried out to verify the method using the Carla simulator [6]. We also gathered anecdotal results from a real-world dataset to validate the feasibility of our approach in reality.

A. Parameter Tuning

Our localization pipeline involves numerous parameters, including the outlier distance threshold in data association, and matrices related to the state estimator. In the process models, \mathbf{Q}_C is associated with the state covariance of vehicle pose and velocity, and \mathbf{Q}_{GM} affects the magnitude of the random walk of GPS frame. The observation noise parameters consist of \mathbf{R}_{VG} , $\mathbf{R}_{\text{light}}$, \mathbf{R}_{lane} , \mathbf{R}_{ϖ} , and r_u that are associated with GPS, traffic light, lane markings, wheel encoders, and pseudo-measurements, respectively. These parameters are manually tuned by initializing them with reasonable values, followed by adjustments to achieve optimal performance evaluated on a validation dataset generated from Carla.

B. Carla Simulation

Our localization algorithm was first tested using Carla, an autonomous driving simulator [6]. In particular, the experiments were conducted using the map ‘‘Town10HD’’, which offers a photorealistic urban driving environment and a perfect semantic map. The benefit of using a simulator is the availability of ground truth, which simplifies the analysis of localization results. Due to Carla not supporting an offset in GPS measurements, we instead manually injected one with 2 m in both longitude and latitude. The time-dependency of

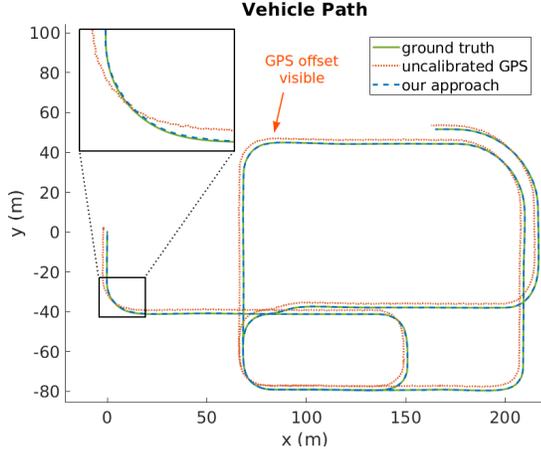


Figure 5. Vehicle path of Carla simulation with total length of roughly 1 km. The ground truth path is compared with results from uncalibrated GPS and our proposed approach. One of the turns is zoomed in to show that the estimated path using our approach very closely overlap with the ground truth path, while the uncalibrated GPS path significantly diverges from it.

the offset is ignored since it is negligible in the duration of a simulation run. With this setup, the simulation data was obtained from roughly 1 km of driving. The vehicle path comparing ground truth with our method and uncalibrated GPS is shown in Figure 5.

1) *Localization Results:* The longitudinal, lateral, and heading localization errors computed using ground truth are summarized in Table I. Our proposed method achieves highly accurate results with a median longitudinal error of 0.053 m, a median lateral error of 0.031 m, and a median heading error of 0.004 radians. When shown as a histogram in Figure 6, we observe that the longitudinal errors have a larger spread than lateral errors, and the vehicle heading always remains very accurate. This is in line with our expectations since lane markings, the most abundant type of semantic cues, only provide lateral and heading corrections. In contrast, longitudinal corrections offered by traffic lights are only available around road intersections.

2) *GPS Offset Estimation:* Being the key motivation for developing the proposed localization algorithm, achieving accurate estimation of the GPS-to-map offset is crucial. The blue line in Figure 7 shows the GPS offset estimation error. Starting from a poor initial guess, our localization algorithm successfully refines the estimates and drops the error down to just a few centimetres, with no manual calibration required.

3) *GPS Dropouts:* To evaluate the robustness of our localization algorithm, we introduced periodic GPS dropouts lasting for 30 seconds in every 60-second interval, i.e., half of the GPS measurements are lost. Under such conditions, the proposed approach is still able to achieve accurate estimation of the GPS offset as shown by the orange line in Figure 7, albeit at a slower pace. Furthermore, the localization results are shown in Figure 6 and summarized in Table I. Compared to the scenario without any GPS

Table I
CARLA LOCALIZATION ACCURACY WITH & WITHOUT GPS DROPOUTS

| Errors | Experimental Scenarios | | | | | |
|------------------|------------------------|-------|-------|--------------|-------|-------|
| | No Dropouts | | | GPS Dropouts | | |
| | Median | 95% | 99% | Median | 95% | 99% |
| Longitudinal (m) | 0.053 | 0.145 | 0.185 | 0.069 | 0.370 | 0.504 |
| Lateral (m) | 0.031 | 0.104 | 0.172 | 0.032 | 0.158 | 0.270 |
| Heading (rad) | 0.004 | 0.014 | 0.025 | 0.004 | 0.015 | 0.028 |

dropout, we observe virtually no increase in the median lateral and heading errors largely due to frequent occurrences of lane markings, which keep the vehicle in the correct lane. This highlights the importance and effectiveness of lane markings as a crucial type of semantic cues in semantic localization. On the other hand, there is a significant decline in performance over the worst case scenario in terms of longitudinal error, where it increases from 0.185 m to 0.504 m. This can be attributed to the infrequent appearance of traffic lights, which help with longitudinal localization, when road intersections are not nearby. In this case, the vehicle can only rely on wheel odometry for relative localization during GPS dropouts, which accumulates longitudinal errors. Nevertheless, the localization accuracy is still acceptable for autonomous driving. This demonstrates the robustness of our proposed approach against frequent GPS dropouts by leveraging semantic cues.

C. Real-World Experiments

Unfortunately, the vast majority of the publicly available self-driving datasets do not provide semantic maps. Those that do all lack other components necessary for our experiments. For instance, nuScenes dataset does not provide raw GPS data [43]. Therefore, to demonstrate real-world feasibility, experiments are conducted using aUToronto’s dataset collected during the SAE AutoDrive competition where the incident of uncalibrated GPS occurred [4]. However, due to the lack of localization ground truth in the dataset for comparison, this will only serve as anecdotal results to verify the effectiveness of our approach. Figure 1 provides a side-by-side comparison of a snapshot of the camera image overlaid with projection of the lane boundaries from the semantic map, which indicates the estimated location of the vehicle with respect to the map. Visually, we see that the GPS data is unusable on its own while our approach is able to self-calibrate for the GPS offset and has the projected lane boundaries well aligned with the lane markings to achieve accurate localization.

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed a method capable of localizing an autonomous vehicle while self-calibrating for an offset between GPS and semantic map frames. This is achieved by using a lightweight semantic map containing locations of lane boundaries and traffic lights, which are complementary in correcting for lateral and longitudinal

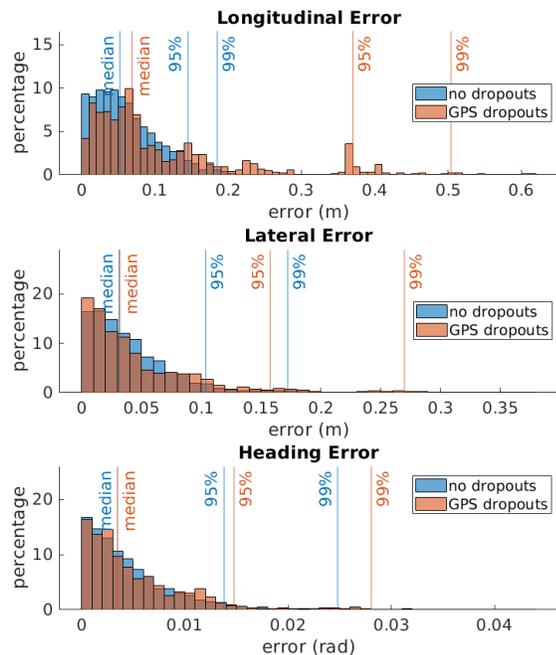


Figure 6. Histograms of longitudinal (top), lateral (middle), and heading (bottom) localization errors of Carla simulation comparing scenarios with and without GPS dropouts.

position of the vehicle. These semantic cues are detected via a monocular camera and integrated with GPS and wheel encoders. Our approach is evaluated using Carla simulator, which demonstrates robustness against GPS dropouts in addition to achieving decimetre-level accuracy. The real-world feasibility of our approach is also validated with a dataset collected by a vehicle with a GPS not calibrated to the semantic map. Next steps include the addition of other types of semantic cues to decrease the gap between semantic cue observations due to sparsity, as well as the closed-loop implementation of our semantic localization system on an autonomous vehicle.

ACKNOWLEDGMENT

This work is funded by the Natural Sciences and Engineering Research Council of Canada (NSERC), and supported by aUToronto who provided access to their internal software and dataset.

REFERENCES

[1] Z. J. Chong, B. Qin, T. Bandyopadhyay, M. H. Ang, E. Frazzoli, and D. Rus, “Synthetic 2D lidar for precise vehicle localization in 3D urban environment,” in *IEEE International Conference on Robotics and Automation*, 2013, pp. 1554–1559.

[2] R. W. Wolcott and R. M. Eustice, “Fast lidar localization using multiresolution Gaussian mixture maps,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 2814–2821.

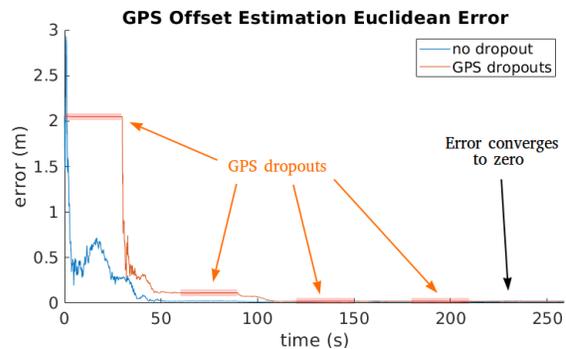


Figure 7. Euclidean error of GPS-to-map offset estimation of Carla simulation. By taking advantage of semantic cues, our localization algorithm is able to estimate the GPS measurement offset with decimetre-level accuracy even with the presence of periodic GPS dropouts.

[3] N. Akai, L. Y. Morales, T. Yamaguchi, E. Takeuchi, Y. Yoshihara, H. Okuda, T. Suzuki, and Y. Ninomiya, “Autonomous driving based on accurate localization using multilayer lidar and dead reckoning,” in *IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, 2017, pp. 1–6.

[4] K. Burnett, J. Qian, X. Du, L. Liu, D. J. Yoon, T. Shen, S. Sun, S. Samavi, M. J. Soroocky, M. Bianchi, K. Zhang, A. Arkhangorodsky, Q. Sykora, S. Lu, Y. Huang, A. P. Schoellig, and T. D. Barfoot, “Zeus: A system description of the two-time winner of the collegiate sae autodrive competition,” *Journal of Field Robotics*, vol. 38, no. 1, pp. 139–166, 2021.

[5] J. Laneurit, R. Chapuis, and F. Chausse, “Accurate vehicle positioning onto a numerical map,” *International Journal of Control, Automation and Systems*, vol. 3, pp. 15–31, 2005.

[6] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, “CARLA: An open urban driving simulator,” in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.

[7] D. Fontanelli, L. Ricciato, and S. Soatto, “A fast RANSAC-based registration algorithm for accurate localization in unknown environments using lidar measurements,” in *IEEE International Conference on Automation Science and Engineering*, 2007, pp. 597–602.

[8] A. Y. Hata and D. F. Wolf, “Feature detection for vehicle localization in urban environments using a multilayer lidar,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 420–429, 2015.

[9] P. Egger, P. V. Borges, G. Catt, A. Pfrunder, R. Siegwart, and R. Dubé, “Posemap: Lifelong, multi-environment 3D lidar localization,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 3430–3437.

[10] C. Le Gentil, T. Vidal-Calleja, and S. Huang, “IN2LAAMA: Inertial lidar localization autocalibration and mapping,” *IEEE Transactions on Robotics*, 2020.

[11] K. Yoneda, H. Tehrani, T. Ogawa, N. Hukuyama, and S. Mita, “Lidar scan feature for localization with highly precise 3-D map,” in *IEEE Intelligent Vehicles Symposium Proceedings*, 2014, pp. 1345–1350.

[12] J.-H. Im, S.-H. Im, and G.-I. Jee, “Vertical corner feature based precise vehicle localization using 3D lidar in urban area,” *Sensors*, vol. 16, no. 8, p. 1268, 2016.

- [13] H. Liu, Q. Ye, H. Wang, L. Chen, and J. Yang, "A precise and robust segmentation-based lidar localization system for automated urban driving," *Remote Sensing*, vol. 11, no. 11, p. 1348, 2019.
- [14] W. Lu, Y. Zhou, G. Wan, S. Hou, and S. Song, "L3-net: Towards learning based lidar localization for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6389–6398.
- [15] D. Gruyer, R. Belaroussi, and M. Revilloud, "Map-aided localization with lateral perception," in *IEEE Intelligent Vehicles Symposium Proceedings*, 2014, pp. 674–680.
- [16] D. Gruyer, R. Belaroussi, and M. Revilloud, "Accurate lateral positioning from map data and road marking detection," *Expert Systems with Applications*, vol. 43, pp. 1–8, 2016.
- [17] F. Chausse, J. Laneurit, and R. Chapuis, "Vehicle localization on a digital map using particles filtering," in *IEEE Proceedings. Intelligent Vehicles Symposium*, 2005, pp. 243–248.
- [18] R. Vivacqua, R. Vassallo, and F. Martins, "A low cost sensors approach for accurate vehicle localization and autonomous driving application," *Sensors*, vol. 17, no. 10, p. 2359, 2017.
- [19] K. Shunsuke, G. Yanlei, and L. Hsu, "GNSS/INS/on-board camera integration for vehicle self-localization in urban canyon," in *IEEE 18th International Conference on Intelligent Transportation Systems*, 2015, pp. 2533–2538.
- [20] M. Schreiber, C. Knöppel, and U. Franke, "LaneLoc: Lane marking based localization using highly accurate maps," in *IEEE Intelligent Vehicles Symposium (IV)*, June 2013, pp. 449–454.
- [21] S. Nedeveschi, V. Popescu, R. Danescu, T. Marita, and F. Oniga, "Accurate ego-vehicle global localization at intersections through alignment of visual data with digital map," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 673–687, 2013.
- [22] K. Jo, Y. Jo, J. K. Suhr, H. G. Jung, and M. Sunwoo, "Precise localization of an autonomous car based on probabilistic noise models of road surface marker features using multiple cameras," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3377–3392, 2015.
- [23] T. Wu and A. Ranganathan, "Vehicle localization using road markings," in *IEEE Intelligent Vehicles Symposium (IV)*, 2013, pp. 1185–1190.
- [24] J. K. Suhr, J. Jang, D. Min, and H. G. Jung, "Sensor fusion-based low-cost vehicle localization system for complex urban environments," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1078–1086, 2017.
- [25] A. Vu, A. Ramanandan, A. Chen, J. A. Farrell, and M. Barth, "Real-time computer vision/DGPS-aided inertial navigation system for lane-level vehicle navigation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 2, pp. 899–913, 2012.
- [26] C. Wang, H. Huang, Y. Ji, B. Wang, and M. Yang, "Vehicle localization at an intersection using a traffic light map," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 4, pp. 1432–1441, April 2019.
- [27] A. Welzel, P. Reisdorf, and G. Wanielik, "Improving urban vehicle localization with traffic sign recognition," in *IEEE 18th International Conference on Intelligent Transportation Systems*, Sep. 2015, pp. 2728–2732.
- [28] X. Qu, B. Soheilian, and N. Paparoditis, "Vehicle localization using mono-camera and geo-referenced traffic signs," in *IEEE Intelligent Vehicles Symposium (IV)*, June 2015, pp. 605–610.
- [29] M. Sefati, M. Daum, B. Sondermann, K. D. Kreisköther, and A. Kampker, "Improving vehicle localization using semantic and pole-like landmarks," in *IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 13–19.
- [30] K. Choi, J. K. Suhr, and H. G. Jung, "FAST pre-filtering-based real time road sign detection for low-cost vehicle localization," *Sensors*, vol. 18, no. 10, p. 3590, 2018.
- [31] W.-C. Ma, I. Tartavull, I. A. Bârsan, S. Wang, M. Bai, G. Matyus, N. Homayounfar, S. K. Lakshmikanth, A. Pokrovsky, and R. Urtasun, "Exploiting sparse semantic HD maps for self-driving vehicle localization," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 5304–5311.
- [32] M. J. Choi, J. K. Suhr, K. Choi, and H. G. Jung, "Low-cost precise vehicle localization using lane endpoints and road signs for highway situations," *IEEE Access*, vol. 7, pp. 149 846–149 856, 2019.
- [33] H. Li, F. Nashashibi, and G. Toulminet, "Localization for intelligent vehicle by fusing mono-camera, low-cost GPS and map data," in *13th International IEEE Conference on Intelligent Transportation Systems*, 2010, pp. 1657–1662.
- [34] B.-H. Lee, J.-H. Song, J.-H. Im, S.-H. Im, M.-B. Heo, and G.-I. Jee, "GPS/DR error estimation for autonomous vehicle localization," *Sensors*, vol. 15, p. 20779, 08 2015.
- [35] Z. Tao, P. Bonnifait, V. Frémont, and J. Ibañez-Guzman, "Mapping and localization using GPS, lane markings and proprioceptive sensors," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov 2013, pp. 406–412.
- [36] K. Jo, K. Chu, and M. Sunwoo, "GPS-bias correction for precise localization of autonomous vehicles," in *IEEE Intelligent Vehicles Symposium (IV)*, June 2013, pp. 636–641.
- [37] Z. Tao and P. Bonnifait, "Road invariant extended Kalman filter for an enhanced estimation of GPS errors using lane markings," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2015, pp. 3119–3124.
- [38] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv:1804.02767*, 2018.
- [39] T. Takikawa, D. Acuna, V. Jampani, and S. Fidler, "Gated-SCNN: Gated shape CNNs for semantic segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 5229–5238.
- [40] T. D. Barfoot, *State Estimation for Robotics*, 1st ed. New York, NY, USA: Cambridge University Press, 2017.
- [41] S. Anderson and T. D. Barfoot, "Full STEAM ahead: Exactly sparse Gaussian process regression for batch continuous-time trajectory estimation on SE(3)," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 157–164.
- [42] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel, *Robust statistics: the approach based on influence functions*. John Wiley & Sons, 1986.
- [43] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A multimodal dataset for autonomous driving," *arXiv preprint arXiv:1903.11027*, 2019.