

Crowdsourced Generation of Parking Maps Using Mass-Production Vehicle Data for Parking-to-Parking Automated Driving

Yudai Kato¹⁾ Keisuke Maekawa²⁾ Iori Nagase²⁾ Naoki Fukuda¹⁾ Hiroshi Okabe¹⁾ Kazutaka Hayakawa¹⁾

1) AISIN CORPORATION

2-1 Asahi-machi, Kariya, Aichi 448-8650, Japan

2) Panasonic Advanced Technology Development Co., Ltd.

20F Matsushita IMP Building, 1-3-7 Shiromi, Chuo-ku, Osaka 540-6320, Japan

KEY WORDS: Software and Its Underlying Technologies, SLAM, Cloud Computing (E2)

To realize a “Parking-to-Parking” automated driving function covering public roads and parking areas from an origin to a destination parking space, this study investigates a crowdsourced parking-map generation technique using sensor data from mass-production vehicles, aiming for a low-cost, scalable map-building scheme that works even in GNSS-denied indoor environments without dedicated survey vehicles.

The proposed method uses only sensors mounted on mass-production vehicles: four surround-view fisheye cameras, an IMU, wheel-speed sensors, and single-point GNSS. A vehicle-side lightweight Semantic Visual SLAM^(1,2) using road markings as features generates a point-cloud map, which is sent to the cloud for merging.

In map merging, semantic landmarks from SLAM have low distinctiveness and are prone to mismatching. Existing research⁽³⁾ uses parking-slot numbers as a cue, but these are not always present and can be hidden by vehicles, limiting versatility. The proposed method instead evaluates the distinctiveness of each frame with a Semantic Pre-Qualification (SPQ) score⁽²⁾, selects high-quality frames as candidates, and merges them via Iterative Closest Point (ICP). For each correspondence pair, the relative pose between two source-map keyframes is compared with the relative pose after ICP matching across all pairs, and pairs that deviate beyond an allowable range are iteratively removed as geometrically inconsistent, yielding robust merging that does not depend on specific landmarks. The merging procedure is summarized in Algorithm 1.

We use mass-production vehicle data from three domestic facilities with different parking-area sizes (Facility A: approx. 3,000 m², B: approx. 3,200 m², C: approx. 7,000 m²), assuming GNSS is unavailable indoors, using the 3D position error of merged keyframes relative to GNSS ground truth (RTK-FIX) as the accuracy metric.

Table 1 summarizes the results for one merge per facility. The reliability check removes geometrically inconsistent mismatches: errors of several meters without the check (mean 2.8–7.6 m for A, B) improved to under 1 m, and Facility C improved from 3.4 m to 1.5 m. Comparing standalone and merged maps, Facility C (standalone mean 5.34 m) improved to 1.46 m (approx. 73 % reduction), and Facility B improved by approx. 9 %, while Facility A, already accurate (mean 0.11 m), slightly worsened from the source map’s relative error; larger standalone-map error thus yields a greater accuracy gain from merging.

The merging process (10-run average, Core(TM) i7-13700 CPU, 32 GiB RAM) is dominated by high-SPQ matching (ICP over candidate keyframe pairs), which increases sharply with keyframe count (approx. 11 s for 15/56 source/target keyframes, approx. 66 s for 40/127). Accelerating candidate search/optimization for large-scale maps and further reducing map size for distribution remain future work.

References

- (1) Qin, Tong, et al. “AVP-SLAM: Semantic visual mapping and localization for autonomous vehicles in the parking lot.” International Conference on Intelligent Robots and Systems (IROS). IEEE, 2020. p. 5939-5945.
- (2) Li, Ye, et al. “AVM-SLAM: Semantic visual SLAM with multi-sensor fusion in a bird’s eye view for automated valet parking.” International Conference on Intelligent Robots and Systems (IROS). IEEE, 2024. p. 7937-7943.
- (3) Jiang, ZhiHong, et al. “Dynamic indoor mapping for AVP: Crowdsourcing mapping without prior maps.” IET Intelligent Transport Systems, 2024, 18.12: 2397-2408.

Algorithm 1 Map Merging

Require: target map M_t , source map M_s
Ensure: merged map M

```

1:  $T_0 \leftarrow \text{InitAlign}(M_s, M_t)$ 
2:  $M_s \leftarrow \text{Transform}(M_s, T_0)$ 
3:  $C \leftarrow \emptyset$ 
4: for all  $q \in M_s$  s.t.  $\text{SPQ}(q) \geq \tau_{\text{spq}}$  do
5:    $P \leftarrow \{p \in M_t \mid \text{SPQ}(p) \geq \tau_{\text{spq}},$ 
      $\|q - p\| < d_{\text{max}}, |\Delta h| < h_{\text{max}}\}$ 
6:    $(T, s) \leftarrow \text{ICP}(q, P)$ 
7:   if  $s \geq \tau_{\text{score}} \wedge \Delta(T) \leq \delta$  then
      $C \leftarrow C \cup \{(q, T, s)\}$ 
8:  $C \leftarrow \text{ConsistencyFilter}(C)$ 
9: if  $C = \emptyset$  then
10:  return FAIL
11:  $M \leftarrow \text{Append}(M_t, M_s)$ 
12:  $M \leftarrow \text{Optimize}(M, C)$ 
13: for  $k = 1$  to 2 do
14:   $C \leftarrow C \cup \text{MatchDuplicates}(M, d_{\text{dup}});$ 
      $M \leftarrow \text{Optimize}(M, C)$ 
15:  $M \leftarrow \text{RemoveDuplicatePoints}(M)$ 
16: return  $M$ 
```

Table1 Position error (Mean/RMSE [m])

Facility	w/o check	with check	before merge
A	7.59 / 7.75	0.14 / 0.16	0.11 / 0.13
B	2.84 / 4.73	0.42 / 0.46	0.47 / 0.50
C	3.44 / 5.27	1.46 / 1.75	5.34 / 7.97