

# Registration of GPS and Stereo Vision for Point Cloud Localization in Intelligent Vehicles Using Particle Swarm Optimization

Vijay John<sup>1</sup>(✉), Yuquan Xu<sup>1</sup>, Seiichi Mita<sup>1</sup>, Qian Long<sup>2</sup>, and Zheng Liu<sup>3</sup>

<sup>1</sup> Research Center of Smart Vehicles, Toyota Technological Institute,  
Nagoya, Japan

{vijayjohn,yuquan.xu86,smita}@toyota-ti.ac.jp

<sup>2</sup> Chinese Academy of Sciences, Beijing, China

longqian@ynao.ac.cn

<sup>3</sup> University of British Columbia (Okanagan), Kelowna, BC V1V 1V7, Canada  
zheng.liu@ieee.org

**Abstract.** In this paper, we propose an algorithm for the registration of the GPS sensor and the stereo camera for vehicle localization within 3D dense point clouds. We adopt the particle swarm optimization algorithm to perform the sensor registration and the vehicle localization. The registration of the GPS sensor and the stereo camera is performed to increase the robustness of the vehicle localization algorithm. In the standard GPS-based vehicle localization, the algorithm is affected by noisy GPS signals in certain environmental conditions. We can address this problem through the sensor fusion or registration of the GPS and stereo camera. The sensors are registered by estimating the coordinate transformation matrix. Given the registration of the two sensors, we perform the point cloud-based vehicle localization. The vision-based localization is formulated as an optimization problem, where the “optimal” transformation matrix and corresponding virtual point cloud depth image is estimated. The transformation matrix, which is optimized, corresponds to the coordinate transformation between the stereo and point cloud coordinate systems. We validate the proposed algorithm with acquired datasets, and show that the algorithm robustly localizes the vehicle.

## 1 Introduction

Vehicle localization in the surrounding environment, such as dense 3D point clouds, is an important research problem for ADAS applications. In the standard localization algorithm, the GPS sensor is used. However, the GPS is error prone in environments, where satellite reception is limited, resulting in localization errors [2]. Researchers address this problem by performing sensor fusion with additional sensors such as camera, LIDAR, and Inertial Navigation Systems (INS) [13]. Since precise INS and LIDAR sensors are expensive, vision and GPS-based vehicle localization is increasingly receiving attention from the research community [13].

In this work, we propose to perform the sensor fusion of the stereo camera and GPS for depth-based vehicle localization within 3D dense point cloud using the particle swarm optimization algorithm. In the proposed algorithm, we first register the stereo and GPS coordinate systems in an offline phase. Given the estimated stereo-GPS transformation matrix, we subsequently localize the vehicle in the dense point cloud using depth information, in an online phase, using the particle swarm optimization algorithm. In this phase, stereo vision-based depth information is used within an optimization framework to perform the localization within the dense 3-D point clouds. Given the registered GPS and stereo camera, the localization is performed by estimating the optimal transformation matrix between the stereo and the point cloud coordinate system. The optimal world-stereo transformation matrix along with the GPS-stereo transformation matrix generates an optimal virtual depth map from the point cloud data. The transformation matrices are optimised by measuring the similarity between the corresponding virtual depth maps and the stereo depth maps. To efficiently measure the similarity, we prune the depth map using  $v$ -disparity [7] and restrict the similarity measurement to the unpruned regions. The proposed algorithm is validated with an acquired dataset, where we show the robustness of the proposed algorithm.

The rest of this paper is structured as follows. In Sect. 2, we present a literature review of the state-of-the-art. Details of the proposed algorithm are described in Sect. 3. The experimental results are presented in Sect. 4. Finally, we present our observations and directions for future work in Sect. 5.

## 2 Literature Review

The research problem of vehicle localization, typically, uses the GPS sensor. However, the GPS-based navigation systems suffer from low accuracy and intermittent missing signals under certain conditions [2]. The signal outages or localization errors, typically, occur when the vehicles moves through tunnels and roads surrounded by tall building. Additionally, the GPS errors are also introduced when the satellites are not widely positioned [2]. Researchers addressed this issue by incorporating inertial navigation systems (INS) [5] and visual odometry (VO) [8]. However these methods, inspite of improving the localization, are affected by drift errors.

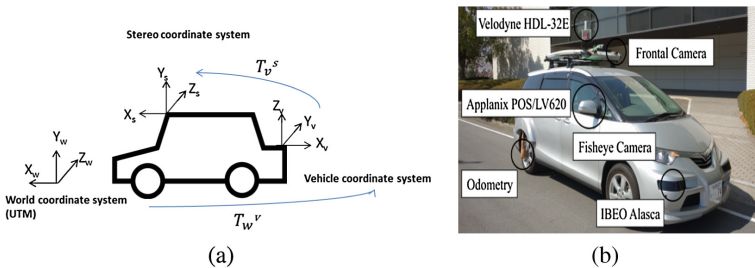
To address the issue of divergence, researchers have proposed to use environmental maps, in the form of satellite images and dense point clouds, for the vehicle localization [10, 11]. Typically, monocular cameras are used for the vehicle localization. In the work by Noda et al. [11] the localization was performed by matching the speeded-up robust features (SURF) observed from the on-board camera with the ones from satellite images. Similarly in [10], detected image lanes were matched with lanes in digital maps using GPS and INS information to perform localization. To enhance the accuracy of localization, Yoneda et al. [13] utilized the LIDAR sensor to perform vehicle localization in dense 3D point clouds, and achieved good localization accuracy. However, the main drawback

and limitation of the LIDAR sensor is its high cost. Thus, it is more feasible to use cheaper sensors, like stereo camera, to perform the matching in dense 3D point clouds. In spite of the recent advancements in vehicle localisation techniques, it can be seen that the problem is still not solved. In this paper, we propose to use stereo vision and GPS sensors to localize the vehicle in dense point clouds.

### 3 Depth-Based Vehicle Localization in Point Cloud Maps

We propose the sensor fusion of stereo and GPS sensor for vehicle localization in point cloud maps using depth information. The point cloud maps are generated as a representation of the vehicle’s environment [1]. The 3D point clouds,  $\mathbf{P}$ , contain latitude, longitude and altitude information, where the latitude and longitude are represented in the 2D Universal Transverse Mercator (UTM) system. To perform localization, we estimate the transformation matrices between the world, vehicle and stereo coordinates. The different coordinate systems are defined as follows. Firstly, the world coordinate system is defined at the origin of point cloud map’s UTM system. Secondly, the vehicle coordinate system corresponds to the GPS location in the vehicle. Finally, the stereo coordinate system is defined at the location of the left camera of the stereo pair. An illustration of the different coordinate systems are provided in Fig. 1-a.

The localization is achieved by generating virtual depth maps  $M_v$  from the point cloud  $\mathbf{P}$ , and measuring their similarity with the stereo depth map  $M_s$  within a particle swarm optimization (PSO). While,  $M_s$  is generated by the MPV algorithm [9],  $M_v$  are generated using a set of transformation matrices. This set of matrices correspond to the transformation between the world, vehicle and stereo coordinate systems. Apart from these transformation matrices, the stereo’s intrinsic calibration parameters are also needed for  $M_v$  generation. The transformation between the vehicle and stereo coordinate ( $T_v^s$ ) is fixed and corresponds to the sensor fusion of the GPS and stereo camera. The transformation between the word and vehicle ( $T_w^v$ ) varies as the vehicle moves. PSO localizes the vehicle in two phases, an off-line and online phase. The intrinsic calibration  $T_s^i$  and fixed transformation  $T_v^s$  are estimated in the offline phase. In the online



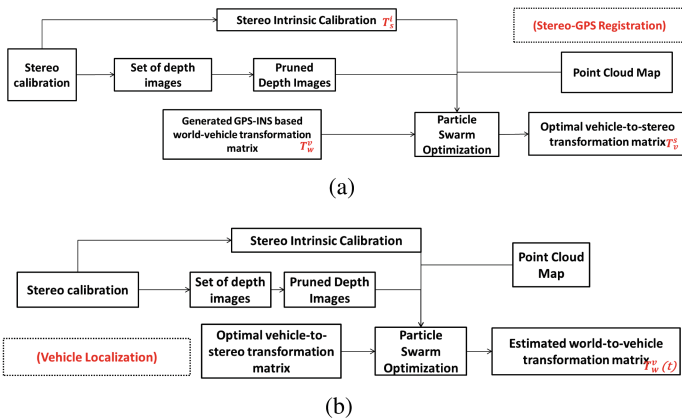
**Fig. 1.** (a) An illustration of the different coordinate system. (b) The intelligent vehicle used for the localization in our experiment.

phase, the varying  $T_w^v$  are estimated within an PSO-based tracker. The online tracker is initialized using the GPS-INS information. We propose a computationally effective PSO cost function, where the depth map is pruned using the u-v disparity.

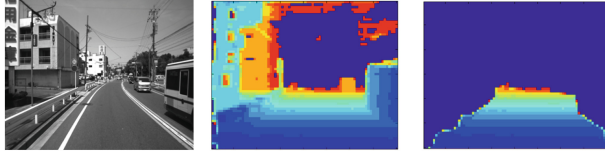
### 3.1 Algorithm Components

*Multipath Viterbi Algorithm.* The MPV algorithm is a stereo matching algorithm based on the dynamic programming-based Viterbi algorithm. The structural similarity (SSIM) is used to measure the matching cost or pixel difference between the stereo images on the epipolar lines. A total variation constraint is incorporated within the Viterbi algorithm. To estimate the disparity maps, which are considered as hidden states, the Viterbi process is performed in 4 bi-directional paths. A hierarchical structure is used to merge the multiple Viterbi search paths. Additionally, an automatic rectification process is also adopted to increase the robustness of the algorithm. We refer the authors to Long et al. [9] for details of the algorithm (Fig. 2).

*Depth Map Pruning.* Given the estimated depth map, we perform the pruning to remove the dynamic objects in the image using the real-time curb detection algorithm proposed by Long et al. [9]. To perform the pruning, we first estimate the histogram of disparity in horizontal and vertical direction or the U-disparity ( $H_y$ ) and V-disparity ( $H_x$ ). Given, the U and V-disparity maps, we first detect the road surface using the V-disparity. Subsequently, we detect the curbs in the depth map. After detecting the curbs and the road, the vertical foreground objects, such as car or pedestrians, are pruned from the depth map. Examples of the pruned depth map are shown in Fig. 3. The pruned depth map reduces the computational complexity during the PSO evaluation.



**Fig. 2.** A detailed overview of the (a) offline and (b) online phase of our proposed algorithm.



**Fig. 3.** An illustration of the depth pruning with (left) right stereo image, (middle) depth image and (right) pruned depth map.

*Particle Swarm Optimization.* In this paper, we use the varying inertia-based PSO proposed by Shi and Eberhart [12], which functions as global-to-local optimizer. Given a  $N$ -dimensional search space, PSO is used to identify the optimal solution using a swarm of  $M$  particles. Each particle in the PSO swarm is an  $N$ -dimensional vector ( $\mathbf{x}_m = \{x_m^n\}_{n=1}^N$ ) representing a position. Additionally, each particle has an associated  $N$ -dimensional velocity vector,  $\mathbf{v}_m = \{v_m^n\}_{n=1}^N$ , to facilitate the search. The best position of each particle and its fitness function value is given by  $\mathbf{p}_m = \{p_m^n\}_{n=1}^N$  and  $\lambda_m$ . The best particle in the swarm and its corresponding fitness function value is denoted by  $\mathbf{p}_g = \{p_g^n\}_{n=1}^N$  and corresponding fitness function is stored as  $g$  respectively. The inertia weight parameter  $w$  defines the exploration of the search space. The social and cognition components of the swarm are defined by the parameters  $\rho_1 = c_1 \text{rand}()$  and  $\rho_2 = c_2 \text{rand}()$ . In this algorithm, we used 5 PSO particles with  $c_1$  and  $c_2$  set at 2.  $A$  was set to 0.5 and  $C$  was set to 100 PSO iterations. A pre-defined search limits derived from the maximum possible inter-frame velocity was used.

### 3.2 Algorithm: Sensor Fusion

In the offline or sensor fusion phase, we first calibrate the cameras using the checkerboard. Subsequently, we estimate the fixed  $T_v^s$  using the PSO. To perform the estimation, we first acquire a dataset with our experimental vehicle known as the vehicle-stereo dataset. The vehicle-stereo dataset contains a set of stereo depth images  $M_s$  along with GPS-INS information. The GPS-INS information is used to generate the  $T_w^v$ , or world-vehicle transformation matrix. To eliminate GPS errors, the vehicle-stereo dataset is acquired in an area without any tall buildings. Given, the acquired  $T_w^v$  and the  $T_s^i$  (intrinsic), we estimate the  $T_v^s$  by generating and matching candidate virtual depth maps with the stereo depth map using the PSO. The candidate depth maps are generated from the Aisan-based point cloud data [1].

**Cost Function.** PSO generates the candidate matrix  $T_v^s$  or PSO particle  $\mathbf{x} = \mathbf{x}_v^s$ . The candidate virtual depth map and the corresponding depth map are evaluated by the PSO-based cost function at the unpruned depth indices, given as,

$$f(\mathbf{x}') = \text{dist}(M_s, M_v(\mathbf{x}')) \quad (1)$$

where  $\mathbf{x}'$  is the PSO particle which represents the vector representation of the candidate transformation matrix.  $\mathbf{x}' = [e_x, e_y, e_z, \theta, t_x, t_y, t_z]$ , where  $e_x, e_y, e_z, \theta$  represents the axis-angle representation of the rotation matrix and  $t_x, t_y, t_z$  represents the translation parameters.

### 3.3 Algorithm: Online Phase

Given the estimated  $T_v^s$  and  $T_s^i$ , we localize the vehicle by estimating  $T_w^v$  at each time instant  $t$  using a PSO-based tracker. Similar to the offline phase, the unknown transformation matrix is optimized by generating and matching virtual point cloud depth images with the stereo depth image. The PSO-based tracker has two phases, an automatic initialization phase and estimation phase.

**Automatic Initialization.** The online PSO tracker is initialized at time instant  $t = 1$  using the parameters of the  $T_w^v$  matrix or  $\mathbf{x}_w^v(t)$  obtained from the GPS-INS. Please note that, following initialization, the PSO tracker estimates the  $\mathbf{x}_w^v(t)$  for the remaining frames  $t > 1$  *without* the GPS information.

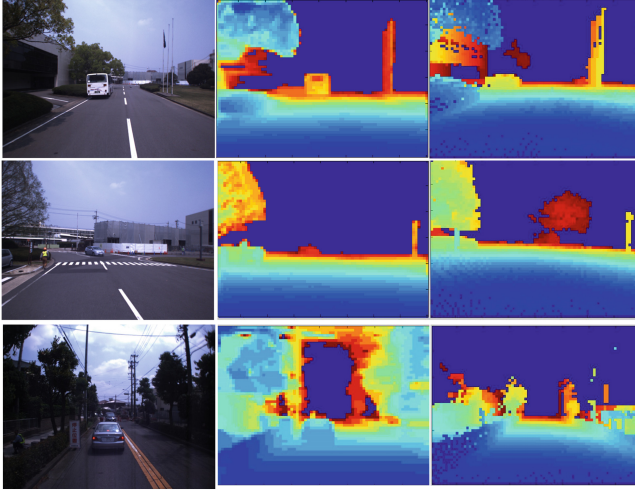
**PSO Estimation.** Candidate parameters generated by the PSO algorithm, along with the previously estimated fixed transformation matrices, are used to generate the candidate virtual point cloud depth image. By measuring the similarity between the candidate and stereo depth images, the optimal  $\mathbf{x}_w^v(t)$  transformation parameters are estimated and the vehicle is localized. The initialized PSO tracker estimates the optimal candidate for a given frame  $t$ , using the cost function Eq. 1, where  $\mathbf{x} = \mathbf{x}_w^v(t)$ .

**Propagation.** Once the transformation matrix is estimated for a given frame  $t$ , the PSO swarm at every subsequent frame  $t + 1$  is initialized by sampling from a zero-mean Gaussian distribution centered around the previous frame's estimate  $\hat{\mathbf{x}}_w^v(t)$ . The zero mean Gaussian distribution is represented by a diagonal covariance matrix with covariance value 0.01. By iteratively estimating the optimal transformation matrix and propagating the optimal candidate, the PSO tracker performs online tracking. Examples of the localization results are shown in Fig. 4.

## 4 Experimental Results

A comparative analysis with baseline algorithms is performed for the offline and the online phase. The Euclidean distance-based error measure between the estimated parameters and the ground truth parameters are reported. The ground truth parameter for  $\mathbf{x}_v^s$  in the offline phase is calculated directly from the distance and orientation between the GPS and stereo camera on the experimental vehicle. The ground truth parameter for the online phase corresponds to the parameters obtained from the GPS. We implement the algorithm using Matlab and Windows (3.5 GHz Intel *i7*).

**Dataset and Algorithm Parameters.** The experimental vehicle was used to acquire multiple datasets with multiple sequences for the online phase.



**Fig. 4.** Localization results. The left column is the left stereo image, the middle column is the disparity. The right column is the optimal virtual depth map.

The online phase was evaluated with three datasets. The first dataset contains 4 sequences with total of 300 frames. The second dataset contains 5 sequences with total of 1200 frames, while the third dataset contains 3 sequences with a total of 1200 frames. The dataset for the offline phase contains 15 stereo depth maps and corresponding  $T_w^v$  parameters derived from the GPS sensor.

#### 4.1 Offline Phase

We validate the offline phase by performing a comparative analysis with baseline optimization algorithms, the genetic algorithm (GA) and the simulated annealing algorithm (SA) [6]. The number of generations and populations in GA along with the number of iterations in SA was kept similar to the total number of PSO evaluations. Similarly, the search limits were uniform across the algorithms. The results as shown in Table 1, show that the PSO algorithm is better than the baseline algorithm.

**Table 1.** Error mean and variance over 3 trials on the offline dataset.

Algorithms	PSO	GA	SA
Error	$0.62 \pm 0.08$	$1.52 \pm 0.6$	$1.32 \pm 0.5$
Time taken (min)	35	37.25	69.75

**Table 2.** Error mean and variance for the localization experiment

Dataset	PSO	PF	APF
1	$0.63 \pm 0.17$	$0.89 \pm 0.27$	$1.57 \pm 0.7$
2	$0.69 \pm 0.10$	$1.3 \pm 0.53$	$2.04 \pm 1.0$
3	$0.89 \pm 0.06$	$1.1 \pm 0.3$	$2.3 \pm 1.6$

## 4.2 Online Phase

In this section, we validate the PSO tracker by performing a comparative analysis with the widely used particle filter [3] and the APF [4]. The number of particles and iterations of both the particle filter (PF) and APF were kept the same as the PSO. The PF contains 250 particles, while the APF contains 125 particles and 2 layers. In the experiment, the GPS was used only for the initialization, and subsequent localization is only stereo-based. The results tabulated in Table 2, show that the performance of the PSO is better than the baseline algorithms. The low accuracy of the PF can be attributed to the divergence error.

## 5 Conclusion and Future Works

In this paper, we proposed a sensor fusion and vehicle localization algorithm using particle swarm optimization. To perform the localization, we matched the stereo depth map with virtual point cloud depth maps within a particle swarm optimization framework. Virtual depth maps are generated using a series of transformation matrices. The fixed transformation matrix between the GPS and calibrated stereo camera is estimated in an offline phase. Subsequently, the vehicle is localized in the online phase using a particle swarm optimization based tracking algorithm. We increase the computational efficiency by pruning the depth map for the evaluation. We performed a comparative evaluation with state-of-the-art techniques. Based on our results, we demonstrated the improved performance of the proposed algorithm. In the future work, we will implement the tracker phase on the GPU to achieve real-time performance.

## References

1. Aisan Technology Co. Ltd. (2013). <http://www.whatmms.com/>
2. Colombo, O.: Ephemeris errors of GPS satellites. *Bull. Geodesique* **60**(1), 64–84 (1986)
3. Dailey, M., Parnichkun, M.: Simultaneous localization and mapping with stereo vision. In: *International Conference on Control, Automation, Robotics and Vision* (2006)
4. Deutscher, J., Blake, A., Reid, I.: Articulated body motion capture by annealed particle filtering. In: *CVPR* (2000)
5. Farrell, J., Barth, M.: *The Global Positioning System and Inertial Navigation*. McGraw-Hill, New York (1999)
6. Franconi, L., Jennison, C.: Comparison of a genetic algorithm and simulated annealing in an application to statistical image reconstruction. *Stat. Comput.* **7**(3), 193–207 (1997)
7. Labayrade, R., Aubert, D., Tarel, J.P.: Real time obstacle detection in stereovision on non flat road geometry through “v-disparity” representation. In: *Intelligent Vehicle Symposium* (2002)
8. Kneip, L., Chli, M., Siegwart, R.: Robust real-time visual odometry with a single camera and an IMU. In: *British Machine Vision Conference* (2011)



9. Long, Q., Xie, Q., Mita, S., Tehrani, H., Ishimaru, K., Guo, C.: Real-time dense disparity estimation based on multi-path viterbi for intelligent vehicle applications. In: British Machine Vision Conference (2014)
10. Mattern, N., Schubert, R., Wanielik, G.: High accurate vehicle localization using digital maps and coherency images. In: IVS (2010)
11. Noda, M., Takahashi, T., Deguchi, D., Ide, I., Murase, H., Kojima, Y., Naito, T.: Vehicle ego-localization by matching in-vehicle camera images to an aerial image. In: Koch, R., Huang, F. (eds.) ACCV 2010. LNCS, vol. 6469, pp. 163–173. Springer, Heidelberg (2011). doi:[10.1007/978-3-642-22819-3\\_17](https://doi.org/10.1007/978-3-642-22819-3_17)
12. Shi, Y., Eberhart, R.: A modified particle swarm optimizer. In: International Conference on Evolutionary Computation, pp. 69–73 (1998)
13. Yoneda, K., Tehrani, H., Ogawa, T., Hukuyama, N., Mita, S.: Lidar scan feature for localization with highly precise 3-D map. In: Intelligent Vehicles Symposium (2014)