

# Integration Framework of Monocular Vision-Based Drivable Region Detection and Contour-Based Vehicle Localization for Autonomous Driving Systems

Jia-En Lee <sup>\*</sup>, Feng-Li Lian <sup>\*</sup>, Hou-Tsan Lee <sup>†</sup>

## Abstract

Perception and localization are the keys in autonomous vehicle systems and driver assistance systems. The perception provides the information of environments around the vehicle, like other vehicles, pedestrians, and road signs. The localization provides the position and heading of vehicle, which can be used for path planning, navigation. With perception and localization process, the safety of vehicle driving could be increased. In this paper, an image segmentation method called region growing, using threshold estimated from previous indicated road region, is proposed to determine that the pixels in the image belong to road region or not. With a defined initial partial road region, the whole road region can be obtained. On the other hand, with a prior bird-eye view map of the area where the vehicle drives, the contours of road region extracted from captured images are matching with the contour on the map by iterative closest point to obtain the vehicle position. In addition, in order to increase the precision of matching, the movements of camera are also estimated by matching the contour in consecutive frames. Furthermore, the position estimated from visual information integrated with the information from GPS to obtain more accurate position. Comparing with vision-based localization only, the integration with GPS reduces the weight and influence of bad matching results, which make the estimated position more accurate. The experimental results show that in structured road, with the localization by road signs, stop lines, and lane lines, the global positions of vehicle can be estimated while the relative movements are very close to GPS data.

*Keywords:* Autonomous vehicle systems, monocular camera, road detection, localization, map matching, region growing, inverse perspective mapping (IPM), iterative closest point (ICP).

## 1 Introduction

Autonomous driving system is one of the most popular research in recent years. The system of autonomous driving includes environment perception, localization, planning, and control

---

<sup>\*</sup> National Taiwan University, Taipei, Taiwan

<sup>†</sup> Takming University of Science and Technology, Taipei, Taiwan

[10: Cheng. 2011]. For driving safety, perception and vehicle localization are the primary issues to consider. The common used sensors for environment perception of autonomous vehicle include cameras, lasers, radars, LiDAR, etc.

Among all of the sensors, camera is the most available and low-cost sensor, and provides a lot of color information which is mainly used in human driving. There are many applications which are implemented with only cameras, including road detection, obstacle detection, and traffic sign recognition [2: Álvarez et al. 2013]. This shows that environment perception is achievable with only image sequence.

For vehicle localization, the most common used way is Global Positioning System (GPS), and it might have drift and vulnerable in urban environment because of lack of local information. On the other hand, for human driving, the prediction of ego-vehicle position and drivable region detection are accomplished simultaneously by detecting the position of curbs or lane markings, which means the local information can be provided from the image captured from onboard cameras. By the road shape and features, the local position of vehicle can be found. Furthermore, in some scenes, the road markings are not fully captured by camera, the type of road markings are various. In such conditions, it is hard to use the road markings or lane lines for vehicle localization. So, the contour of asphalt road region is used, which is more general. Once the visible contour of road presents, such as “slow”, different type of arrows, and curve curbs, the vehicle localization process can be applied. Road region detection also ensures the extracted edges and boundaries are on the road surface, which is purer and more reliable because of the purpose of safety. Comparing with the features like scale-invariant feature transform (SIFT) and histogram of oriented gradient (HOG), localization based on road region contour is more instinctive to user.

For vision-based road region detection, the main problem is classifying the image pixels into road and non-road pixels. One of the most popular methods is machine learning algorithm. By a classifier which can be trained by support vector machine (SVM) [2: Álvarez et al. 2013], or AdaBoost [3: Fritsch et al. 2014], the image pixels are classified into different classes. However, it needs large amount of database to get accurate results, and the performance is highly dependent on training images. But, it is not practical to train the classifiers for all of the scenes that vehicle might encounter. Another kind of road detection method is likelihood classification. There are generally two important parameters in this method, which are the initial seeds or region and restriction. The seeds and initial region can be determined based on previous frames [1: Hsu et al. 2009], [4: Siogkas and Dermatas 2013]. The restriction can be bounded by the upper and lower bounded of previous road region pixels [1: Hsu et al. 2009]. However, if the vehicle drives cross two areas whose colors are slightly different, this method would fail.

The other issue in this thesis is localization of vehicles. Many researches on autonomous vehicle localization are based on the environment features observed by sensors. One of the typical methods for feature based localization is particle filter localization. Many researches have already shown that particle filter can be used for vehicle localization based on the observation from on-board sensors without precise vehicle kinematic model [7: Hata and Wolf 2016], [8: Sivaraman and Trivedi 2013], [9: Cui et al. 2016]. However, if the number of feature points is very large, the running time might be extremely large. Another kind of method is matching the feature points by iterative closest point algorithm [6: Durrant-Whyte and Madhavan 2005]. The pose of sensor can be found by matching the observation to the map, the matching might fail if the features of observation and map are not exactly the same or the initial state is unknown.

There are mainly two problems to solve in this paper: road region detection and vehicle localization. The system flow chart is shown in Figure 1. The information used for vehicle localization process include the image captured from on-board camera, a large bird-eye view map, and GPS position and heading. The road region and its boundary are obtained from image and map, and matched with each other to obtain the vehicle position. On the other hand, GPS position and heading are used in vehicle localization stage to increase the accuracy of matching.

In road region detection, a modified region growing algorithm is present. The growing is operated with square patch instead of a single pixel. It makes the algorithm be capable of tackling broken lane markings and low resolution. Furthermore, a threshold choosing method motivated by maximal stable external region (MSER) is proposed. The threshold is decided by the most stable region, so the restriction can be determined appropriately and based on previous results. The results show that in common vehicle driving environments with different road markings, such as, straight road, stop lines, and road signs, the accuracies of road region detection results are mostly over 90%.

For vehicle localization, the Iterative Closest Point (ICP) algorithm is applied to match the observation to a pre-built map. To tackle the problem of lack of initial state before ICP, the movement between every consecutive frames and the state at previous time step are considered. By transforming the observation to the close position to the map feature, it provides nice prediction for initial state. Furthermore, with the integration with GPS, the weights and influences of bad matching results are reduced, which make the estimated position more accurate, and lateral and longitudinal movement can still be estimated even in straight road. The experimental results show that in structured road, with the localization by road markings, the global positions of vehicle can be estimated while the relative movements are very close to GPS data.

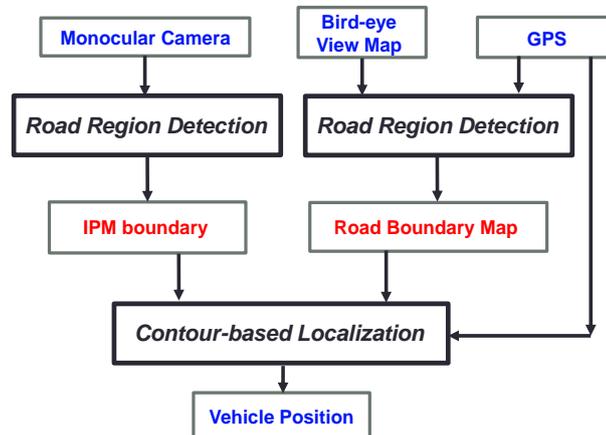


Figure 1: The illustration of proposed contour-based localization.

## 2 The Proposed Algorithms

### 2.1 Road Region Detection Based on Color Feature

An image segmentation method called region growing algorithm is used to classify image pixels belongs to road appearance or not, and determine the drivable region. The growing process has two steps, patch-based and pixel-based. Patch-based region growing detect a rough range of drivable region, and pixel-based region growing is applied to obtain more precise region which is close to the boundary of the road. The threshold of region is defined with the threshold of previous frames, considering the spatial and temporal continuity. After the road region is obtained, the boundary of the region is extracted, and considered as the contour of road which is used for vehicle localization in following stage. The flow chart of road region detection is shown in Figure 2.

The images captured from on-board camera are taken as the input of region growing algorithm, which includes two steps, patch-based that extracts the rough road region and pixel-based that makes the boundary of region more close to edges of road markings. The chosen threshold in current frame is used to determine the threshold of next frame. After the road region is detected, the boundary points of road region are extracted and mapped into IPM coordinate.

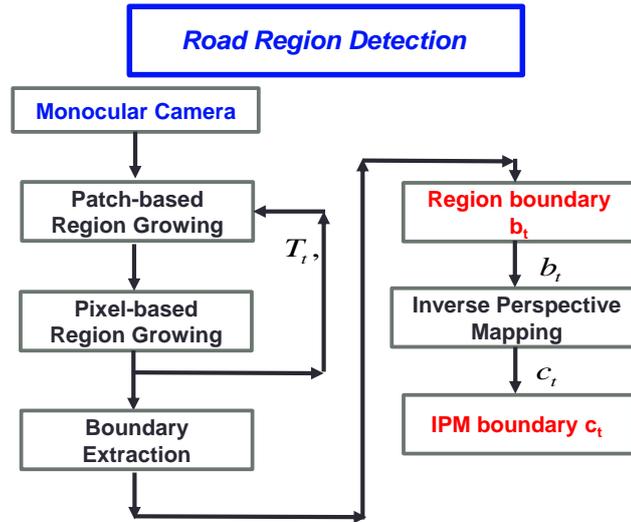


Figure 2: Road region detection system architecture.

### 2.2 Region Growing Algorithm

Methods of road detection using lane markings on the road surface, which is characteristic of a driving lane, extract the lane markings and define area between the road region as the road region [5: Alonso et al. 2012]. It highly depends on the existence and clearness of lane markings. In other words, when the lane markings do not exist or blur, this kind of road detection cannot distinguish the road region accurately.

Another method is using color features of road surface to distinguish the road region [3: Fritsch et al. 2014]. It does not have limitation of existence and clearness of lane markings. The method uses the characteristic that the color or material of drivable region are generally similar in a period of time. Machine learning algorithms classify the image pixels by trained classifier. However, machine learning algorithms need huge amount of training images to get more accurate results, and training the classifier for every different characteristic scenes is not practical. Another kind of road detection methods is likelihood-based classification. By finding the similar pixels to defined initial road pixels or seeds, a road region can be distinguished. A vision-based drivable region labeling method named region growing algorithm is also proposed [1: Hsu et al. 2009] which shows accurate results in various scenes.

The proposed method is based on region growing algorithm to detect road region and extract the boundary of drivable region. By using patch-based region growing and pixel-based region growing in order, a more reliable drivable region extraction can be achieved. For road detection, first, patch-based region growing is applied. The patch-grown region would be bounded by the lane markings or road curbs, but not close enough to the road boundary. The patch-grown region would dilate by half of patch size  $L/2$ , and the dilated region is called bounded road region. Finally, the pixel-based region growing is applied within bounded road region, and the drivable road region can be extracted. In Figure 3, an example of road region detection is shown. Starting from the captured image, the patch-based region growing detects the road region as shown in Figure 3(b) and (c). Then, the region dilates to include the boundary of road markings as shown in Figure 3(d) and (e). Finally, the pixel-based region growing is applied with dilated region, and the more precise road region is obtained.

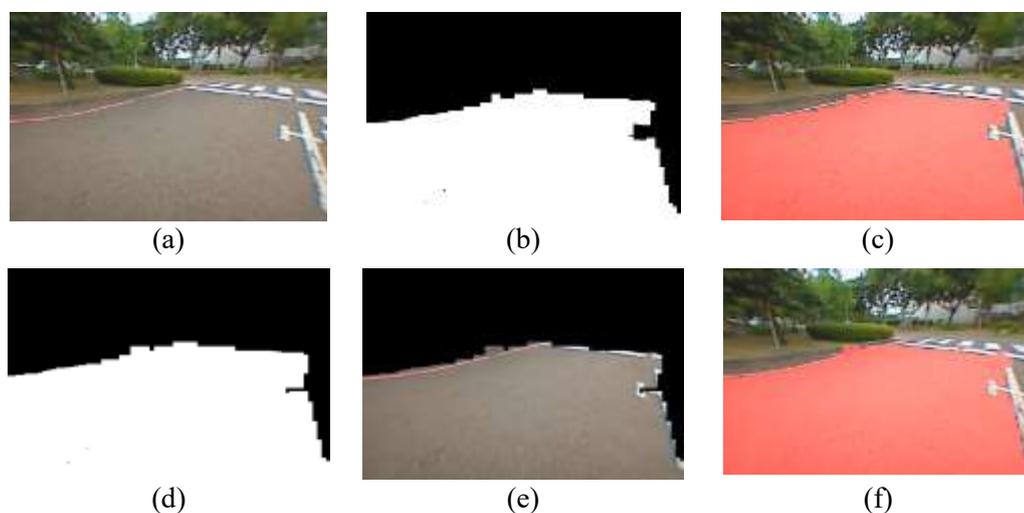


Figure 3: Example of road region detection result.

(a)Original image. (b) Binary image of patch-based grown result. (c) Patch-based grown result on original image. (d) Binary image of dilation of patch-based grown result. (e) Dilation region of patch-based grown result. (f) Apply pixel-based region growing on dilation region.

To extract precise road region, the threshold of region growing algorithm must be determined appropriately. The value of threshold  $T$  is determined by its stability and previous chosen threshold. The stability function  $q(i)$  of maximally stable extremal regions (MSER) is applied to choose the threshold of region growing, as shown in the following:

$$q(i) = \left| \frac{Q_{i+\Delta} - Q_{i-\Delta}}{Q_i} \right| \quad (1)$$

where  $Q_i$  is the number of pixels in grown region at threshold  $i$ , and  $\Delta$  is the interval of threshold. According to experiment, the appropriate threshold usually lies between 0.1 to 0.3, so only stability of threshold within this range is calculated. The interval of threshold  $\Delta$  is set as 0.1. Smaller interval can increase the precision, but it is not necessary and cost much more running time. The most appropriate threshold lies in threshold  $i$  where  $q(i)$  is local minimum.

The chosen threshold  $T_t$  of frame  $t$  is determined by threshold in previous frames, because in an area around a road marking, the difference between road marking and road surface is roughly the same.  $T_t$  is determined as the closest stable threshold  $t$  in frame  $i$  to the chosen threshold  $T_{t-d}$  in previous frames. The stable region whose boundary is representative can be determined with threshold  $i$  where  $q(i)$  is local minimum.

An example of threshold choosing is shown in Figure 4. The red area in Figure 4(b)-(e) is the detected road region with same initial seeds with local minimal stability. In Figure 4(f) and (g), the sizes of region area and the stabilities as vertical axis with different thresholds as horizontal axis are shown, where the local minimum is marked as red circle, local maximum is marked as blue star, and the chosen threshold is marked as magenta triangle. In this case, the previous chosen threshold is set as 0.16 which is the closest threshold with minimal stability to the threshold of previous frame.

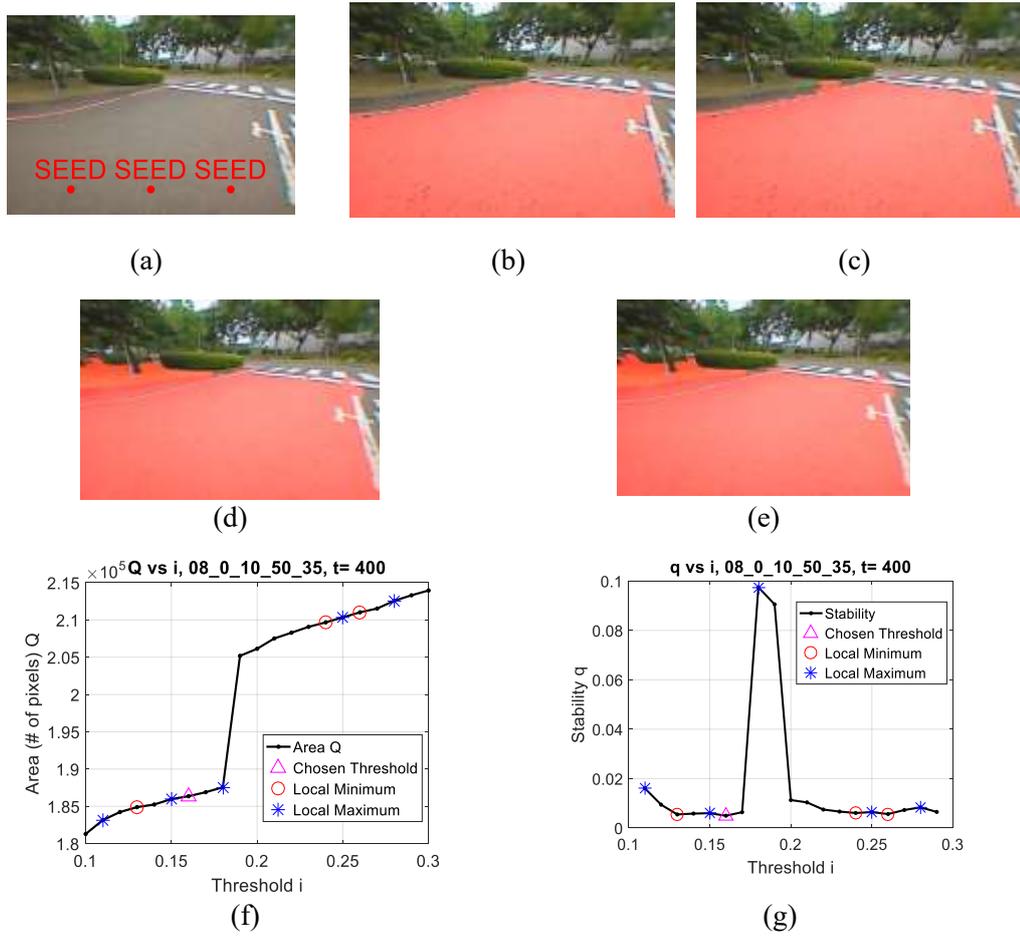


Figure 4: Region growing results with different threshold having local minimum stability in SL<sub>1</sub>.

(a) Seed position. (b)  $T = 0.13$ . (c)  $T = 0.16$  (chosen threshold). (d)  $T = 0.24$ . (e)  $T = 0.26$ .  
 (f) Relationship between region area and threshold. (g) Relationship between stability and threshold.

### 2.3 Region Boundary Extraction

Because the boundaries of road region must be the same in consecutive frames, they are extracted to localize the vehicle, and find the movement of vehicle between consecutive frames. To extract the boundaries of road region, Moore-Neighbor tracing algorithm is applied. However, there might be some very short boundaries because of the texture on the asphalt road, so the boundaries whose length are short than 10 pixels are neglected. In Figure 5, an example of boundary extraction is shown, and the extracted boundaries is shown in Figure 5(c). Although the region boundaries are extracted, there are some points in the boundaries which are not very stable. In other word, these points are very sensitive to the threshold. If the threshold increases a little, these boundary points might disappear. So, the restriction of boundary is defined that the boundary pixels whose distance to road region is less than the next local maximum stability larger than chosen threshold are removed from boundaries. In Figure 5(d) and (e), the remaining boundary

points are shown as red plus marker, and in Figure 5(f), they are plotted on the original image, which shows that the boundary points are mostly the edges of road markings.

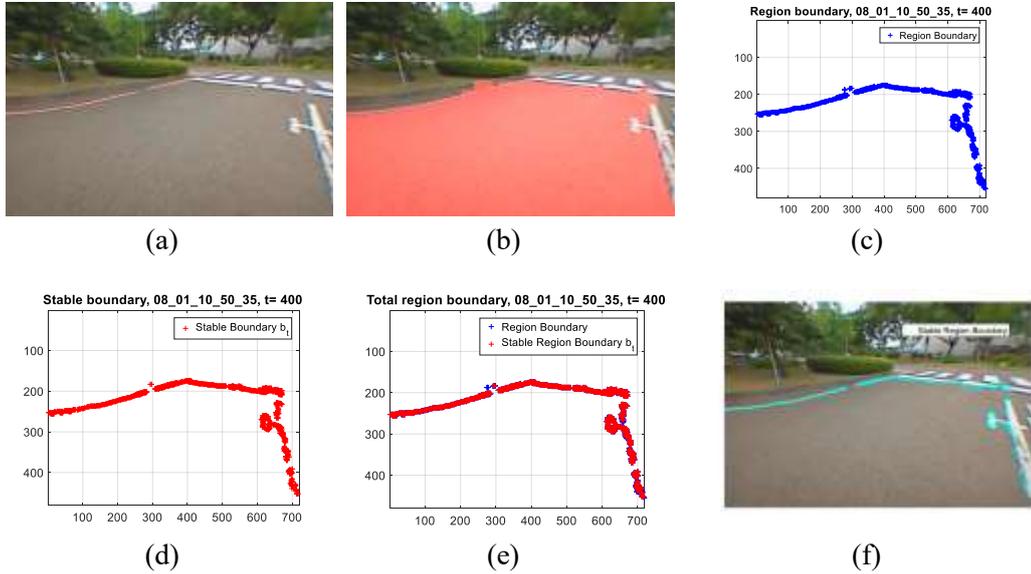


Figure 5: Example of road region boundary extraction.

(a) Original image. (b) Road region detection result. (c) Boundary extraction of road region detection result. (d) Stable boundary extraction. (e) Comparison of stable and total boundary. (f) Stable boundary on original image.

## 2.4 Localization based on Road Boundary Map Matching

The flow chart of ICP localization process is shown in Figure 6. By matching the boundary of IPM images  $ct$  and road region map  $cmap$ , the vehicle can be localized. The estimated movement between consecutive frames is used to make road boundary and road boundary map closer, and the process of map boundary extraction and kNN search make sure the clouds for matching indicate same objects. The ICP algorithm is applied to obtain a more precise position by matching the boundary point clouds.

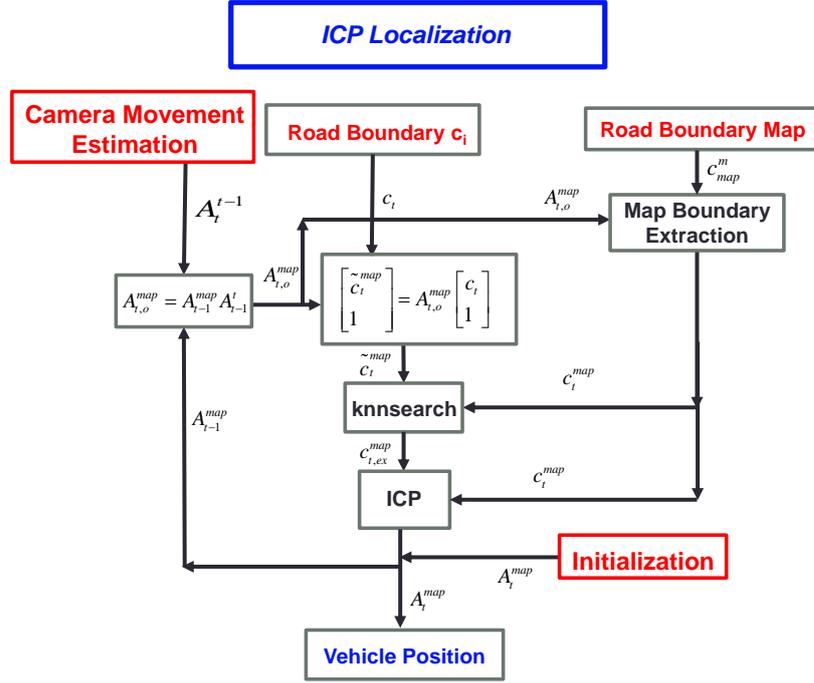


Figure 6: Flow chart of ICP localization.

The boundary points of bird-eye view map are also detected by road region detection with patch size 21, and threshold 0.18, which is determined by experiments. In addition, the seed of region growing algorithm is defined by the adjusted position of GPS data. The position of GPS data is adjusted as  $\tilde{p}_{t,gps}$  by the previous estimated position, and represented as Eq. (2), where  $p_{t-1}$  is the estimated position in previous frame and  $\Delta p_{t,gps}$  is the translation of GPS data.

$$\tilde{p}_{t,gps} = p_{t-1} + \Delta p_{t,gps} \quad (2)$$

In the bird-eye view map, many markers for vehicle localization exist, which are shown within red circles in Figure 7, and their position on the map and the range of the local map surround each marker are recorded in the database. The position of marker  $m$  is represented as  $p_m$ .



Figure 7: Road markings on the road.

When the vehicle drives close to the markers, the vehicle localization process starts, and the road region detection of local map is also operated. This condition can be represented as Eq. (3).

$$\left| \tilde{p}_{t,gps} - p_m \right| < dist_m \quad (3)$$

Considering the limitation of field of view of camera, the distance  $dist_m$  is determined as 3 meters. When the camera is 3 meters close to markers, the vehicle localization, and the road region detection of local map starts. In Figure 8, an example of road marker, its local map, and the range to start vehicle localization are shown as red diamond, magenta rectangle, and red circle respectively.



Figure 8: Road markings position, local road markings map, and range to start vehicle localization.

## 2.5 Boundary Matching by Iterative Closest Point

By matching the boundaries with ICP, the position of vehicle can be determined. The transformation matrix which maps the boundaries in current frame  $C_t$  into the boundary map at each time step  $t$  is defined as  $A_t^{map}$ . An example of boundary extraction result is shown in Figure 9. Figure 9(c) shows the extracted boundary  $C_t$  in the IPM coordinate, where the origin is the position of camera.

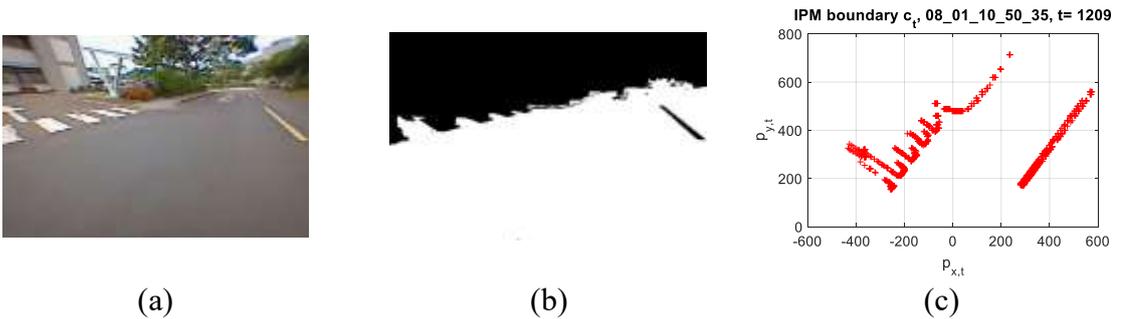


Figure 9: Extracted boundary  $c_t$  in IPM coordinate.

(a) Original image. (b) Detected road region. (c) Extracted boundary  $c_t$ .

To use ICP matching for vehicle localization, the target and source point cloud must be very similar, and consist of same objects. However, the obtained point clouds are from different

sources. The pre-transform boundary cloud is obtained from on-board camera, on the other hand, the boundary map is obtained from unmanned aerial vehicle. When some obstacles which are not captured in bird-eye view map show in the on-board camera, the matching results could be extremely terrible. In order to reduce the influence of present environment, the objects or boundaries which are not present in bird-eye view map and absence in boundary map must be removed from pre-transform boundary cloud. So, the k-nearest neighbor (kNN) algorithm is applied. In this case, the kNN search is also used for finding the nearest point in current transform boundary cloud for each point in boundary map, so the parameter k is defined as 1. The point set which consists of the nearest points is defined. Because pre-transform boundary cloud is obtained by previous position, heading and relative movement estimation between current state and previous state, it must be close to its actual state. By removing the extra obstacles which is obtained from on-board camera and vulnerable to present environment, the result of the ICP matching with boundary map can be more accurate. The example of nearest boundary points to the map, and the boundary map in field of view at time  $t$  are shown in Figure 10.

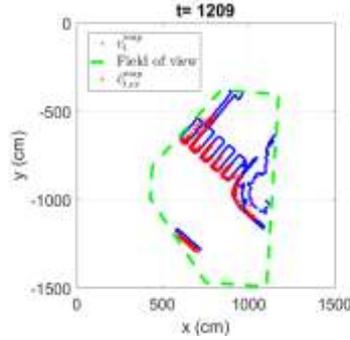


Figure 10: Nearest boundary points by the kNN search.

In order to matching the boundary accurately, ICP is applied again. After ICP, the rotation matrix and the translational vector can be obtained. The relationship is shown as follows:

$$\begin{bmatrix} \hat{C}_t^{map} \\ C_t \\ 1 \end{bmatrix} = \begin{bmatrix} R_t^{map} & T_t^{map} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{C}_t^{map} \\ 1 \end{bmatrix} \quad (4)$$

All the transformation matrix from IPM boundary at time  $t$  to matched boundary can be represented as follows:

$$\begin{bmatrix} \hat{C}_t^{map} \\ C_t \\ 1 \end{bmatrix} = \begin{bmatrix} R_t^{map} & T_t^{map} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_{t,o}^{map} & T_{t,o}^{map} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} C_t \\ 1 \end{bmatrix} = A_t^{map} A_{t,o}^{map} \begin{bmatrix} C_t \\ 1 \end{bmatrix} = A_t^{map} \begin{bmatrix} C_t \\ 1 \end{bmatrix} \quad (5)$$

After the transformation matrix which transforms the IPM boundary from the IPM coordinate to the map coordinate is obtained, the position of camera, which is the origin (0, 0) in the IPM coordinate, is also transformed into the map coordinate. Then, the position of camera on the bird-eye view map can be obtained as follows:

$$\begin{bmatrix} p_{x,t,icp} \\ p_{y,t,icp} \\ 1 \end{bmatrix} = A_t^{map} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (6)$$

## 2.6 Integration with GPS Data

Because the positions estimated by ICP matching might not be accurate and precise for every frame, the results of ICP localization are integrated with GPS data with different weights. Considering the relative translation of GPS data is comparatively accurate, the translations of ICP localization results are compared with GPS data to determine their accuracy. The relative different ratio  $r_i$  between translations of the ICP localization results  $\Delta p_{t,icp}$  and the translation of GPS data  $\Delta p_{t,gps}$  is defined as follows:

$$r_t = \frac{|\Delta p_{t,icp} - \Delta p_{t,gps}|}{\Delta p_{t,gps}} \quad (7)$$

In addition, the qualities of ICP matching can be indicated by the root mean square errors ERT of ICP algorithm. Higher root mean square errors  $ER_t$  is,  $t$  So, the weights of ICP position are defined with  $ER_t$  and relative different  $r_i$ , and represented as follows:

$$W_{t,icp} = \exp(-ER_t \times r_t) \quad (8)$$

As a result, the weight of ICP results  $W_{t,icp}$  is higher to 1, when  $ER_t$  and  $r_i$ , are closer to 0, and the weights are close to 1, when  $ER_t$  and  $r_i$  are higher. On the other hand, the weight of GPS data  $W_{t,gps}$  is defined as follows to make the sum of weights be 1.

$$W_{t,gps} = 1 - W_{t,icp} \quad (9)$$

The integrated position  $p_t$  of GPS and ICP matching and  $\tilde{p}_{t,gps}$  are defined as follows:

$$p_t = W_{t,gps} \tilde{p}_{t,gps} + W_{t,icp} p_{t,icp} \quad (10)$$

$$\tilde{p}_{GPS,i} = p_{i-1} + \Delta p_{GPS,i} \quad (11)$$

## 3 Experimental Results

The geometry of experimental vehicle, on-board camera, and GPS is shown in top of Figure 11. The camera is equipped in the front of the vehicle, where the height is 0.97-meter to the ground plane, and the pitch is 29.4°. There are two GPS sensors at the top central of the vehicle, where

the height is 2.06 meters, and the GPS data is obtained by combining the information from two sensors. The horizontal distance between camera and GPS is 1.96 meters. The entire system illustration is shown in the bottom of Figure 11, where the system output is the estimated position by ICP. And, the following experiments are thus conducted.

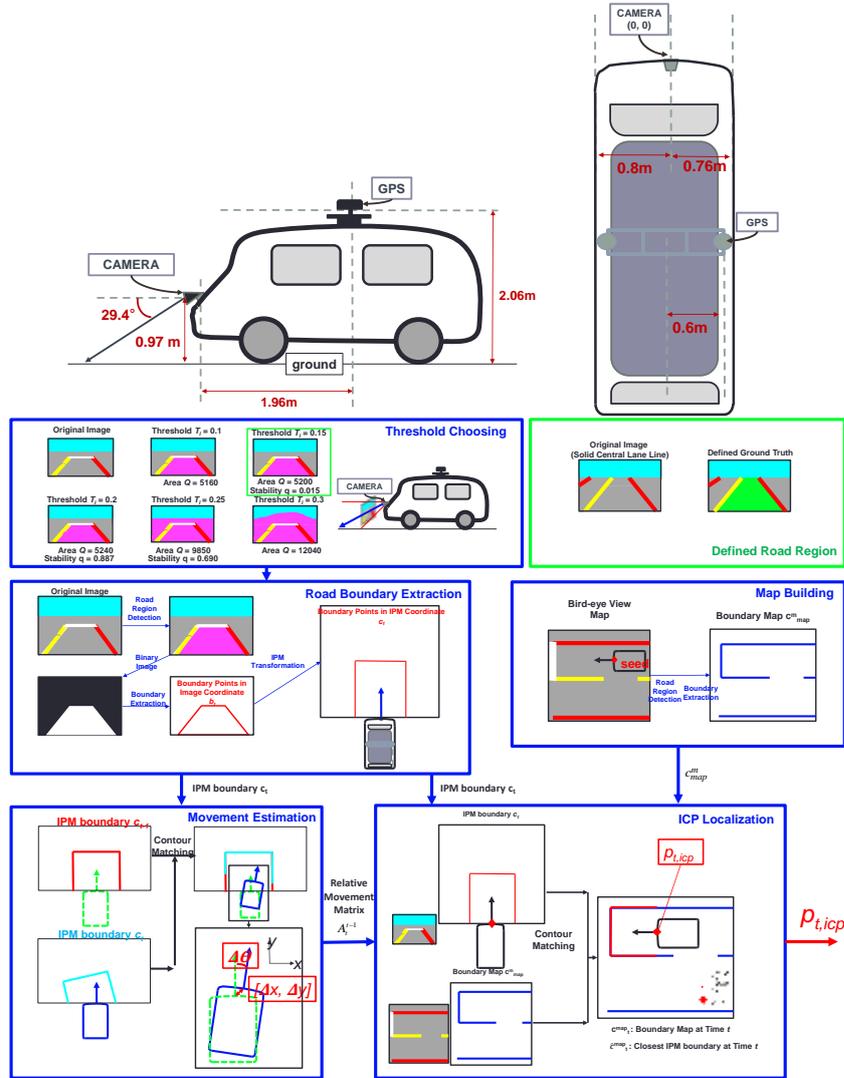


Figure 11: Experimental vehicle setup and system illustration.

### 3.1 Boundary Points Extraction on Road Region Map

Figure 12(a) shows an example of local marker map, and the position of seed, as blue star. Figure 12(b) and (c) show the result of road region detection as red region, and the extracted boundary as blue stars respectively. Finally, the extracted boundary is transformed into bird-eye view map and the boundary map surround marker is obtained as shown in Figure 12(d) as blue stars.

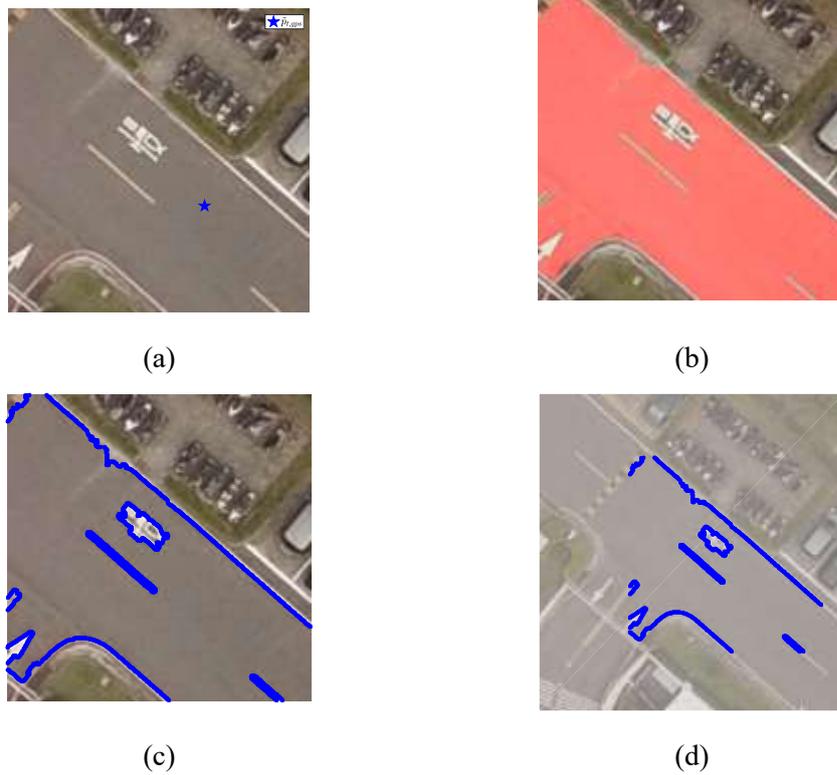


Figure 12: Local boundary map extraction.

(a) Position of seed (camera position). (b) Detected road region. (c) Extracted boundary points (local road marking coordinate). (d) Extracted boundary points (global coordinate).

### 3.2 Boundary Matching by Iterative Closest Point

By matching the boundaries with ICP, the position of vehicle can be determined. The transformation matrix maps the boundaries in current frame into the boundary map at each time step. An example of boundary extraction result is shown in Figure 13. Figure 14 shows the ICP matched boundary at different time, and the position of camera on the boundary map. Figure 11 shows the results of the vehicle localization. In Figure 15(a), the trajectory of GPS data and ICP matching GPS data. In Figure 15(b)-(d), the differences ratio between ICP and GPS, the error of ICP matching GPS, and the weight of ICP position are shown respectively. Figure 15(d) shows the trajectory of integrated localization results. The integrated positions are closer to the ICP localization results when the differences ratio between ICP and GPS and the error of ICP matching are low, or the integrated positions reach the ICP localization results when the time is long enough.

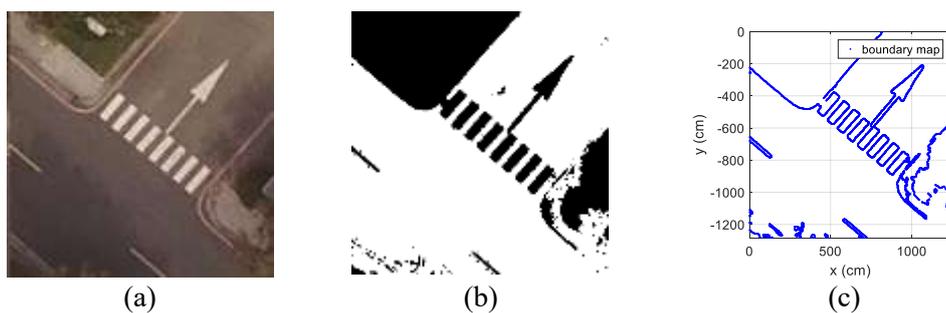


Figure 13: Local boundary map.

(a) Local bird-eye view map. (b) Detected region. (c) Extracted local boundary map.

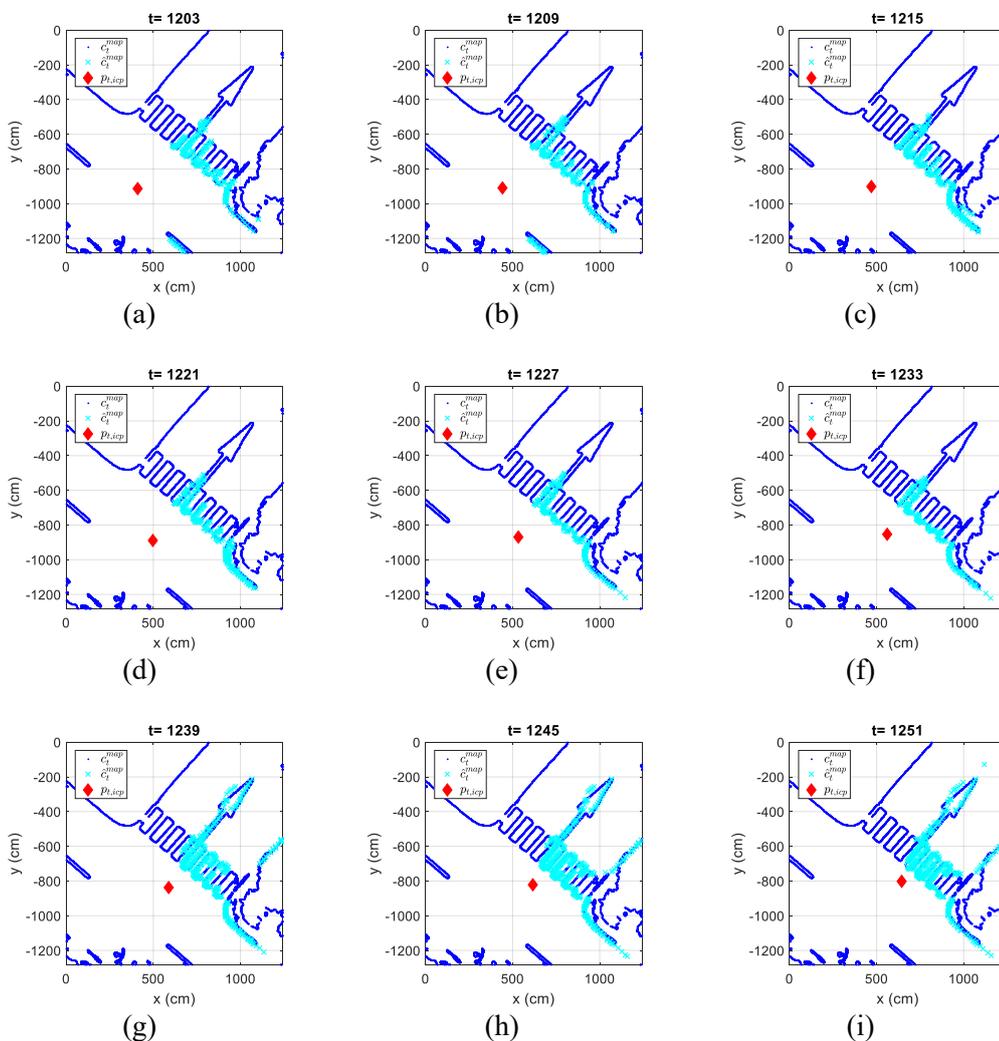


Figure 14: ICP matched boundary.

(a)  $t = 1203$ . (b)  $t = 1209$ . (c)  $t = 1215$ . (d)  $t = 1221$ . (e)  $t = 1227$ . (f)  $t = 1233$ .  
 (g)  $t = 1239$ . (h)  $t = 1245$ . (i)  $t = 1251$ .

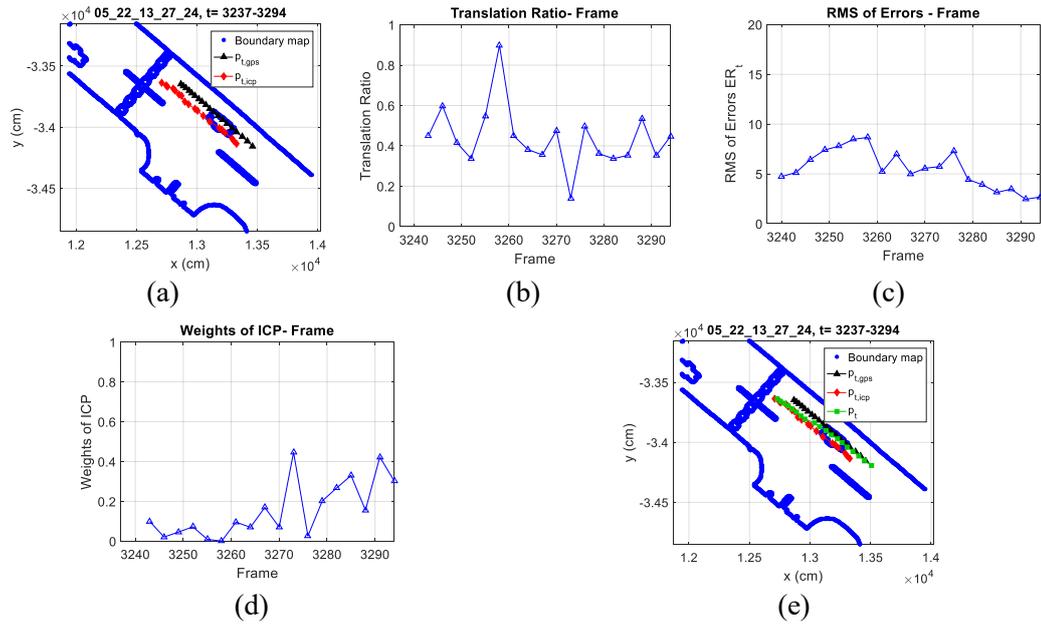


Figure 15: Integration of ICP localization and GPS.

- (a) Trajectory of ICP localization and GPS. (b) Differences ratio between ICP and GPS.  
(c) Errors of ICP matching. (d) Weights of ICP position. (e) Integration result.

### 3.3 Complete Test in Two Scenarios

Figures 16 and 17 show the whole trajectory of contour-based localization, and original GPS trajectory in large scenes, B51 and B58, respectively in global coordinate. In Figure 16, the whole trajectory of the vehicle moves to right by contour-based localization comparing with GPS trajectory. Furthermore, the drift of GPS on longitudinal direction of the road is about half of the width of the road, and with the localization process, the drift is adjusted. In Figure 17, the drift of GPS in this scene is large which makes the position of the vehicle is false when it drives into the court in front of the building. The localization process successfully estimates the actual position where the vehicle drives through, which can prevent from hitting the road curbs.

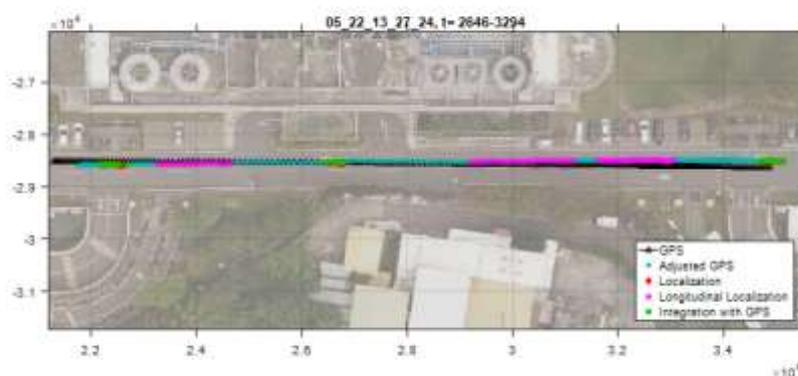


Figure 16: The results of localization, and original GPS trajectory in large scene B51.

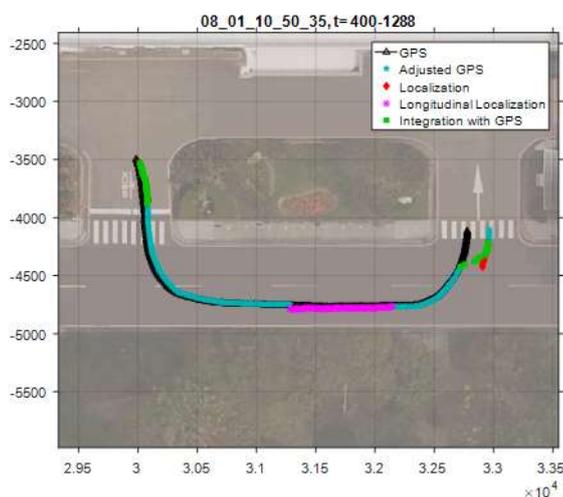


Figure 17: The results of localization, and original GPS trajectory in large scene B58.

## 4 Concluding Remarks

In this paper, a monocular vision-based drivable region detection method with region growing algorithm is proposed. The proposed method shows that it detects the road region and its contour precisely in various common ground vehicle driving scenes which include ‘stop lines,’ ‘road markings,’ ‘crosswalks,’ and ‘pure straight road,’ by choosing the threshold based on temporal continuity. Comparing with traditional region growing algorithm, with proposed patch-based region growing method, the leak of road region caused by broken road markings can be avoided. The experimental results show that the accuracies of road region detection results are over 95% in the scenes that are consistent in road surface color, and even when the road color changes. The average accuracies are still higher than 90% for whole image sequence in the scenes.

After the road region is detected, the contour of the region is obtained by boundary points extraction, and only the robust boundary points remain in extraction, which ensure the contour is reliable for vehicle localization. Then, the vehicle position is estimated by a road contour-based

localization method with iterative closest point (ICP) algorithm. With a bird-eye view map which contains the scenes and their location that vehicle might drive through, the vehicle can be localized by matching the contour of road region with the map. In addition, the observation is extracted by k-nearest neighbor (kNN) search algorithm to neglect the objects, which is not present in map building stage, in ICP matching step. With the help of kNN search and GPS, the input point clouds for ICP indicate possibly the same object, which ensures the system working even the image and bird-eye view have slight difference. To tackle with the uncertainty of matching result, it is integrated with GPS data whose relative movement is very reliable. As a result, the vehicle can be localized by road markings and on the straight road which is lack of lateral information. The experimental results show that the position of vehicle can be estimated precisely with matching errors which are lower than 5 cm, in scenes with road markings that have simple contour. Even for the scenes with Mandarin road sign, whose contour is relatively complicated, the errors of matching are still lower than 10 cm.

## Acknowledgement

This work was supported in part by the Ministry of Education, and the Ministry of Science and Technology: MOST 105-2221-E-002 -135 -MY3 and MOST 108-2634-F-002-0024, Industrial Technology Research Institute.

## References

- [1: Hsu et al. 2009] Chih-Ming Hsu, Fei-Hong Chao, Feng-Li Lian, and Jong-Hann Jean, "Monocular vision-based drivable region labeling using adaptive region growing." in Proceedings of the Society of Instrument and Control Engineers Annual Conference, pp. 2108-2112, Sapporo, Japan, Sep. 9 - 12, 2009.
- [2: Álvarez et al. 2013] José M. Alvarez, Theo Gevers, Ferran Diego, and Antonio M. Lopez, "Road Geometry Classification by Adaptive Shape Models," IEEE Transactions on Intelligent Transportation Systems, vol. 14, no. 1, pp. 459-468, Mar., 2013.
- [3: Fritsch et al. 2014] Jannik Fritsch, Tobias Kühnl, and Franz Kummert, "Monocular Road Terrain Detection by Combining Visual and Spatial Information," IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 4, pp. 1586-1596, Aug., 2014.
- [4: Siogkas and Dermatas 2013] George K. Siogkas, and Evangelos S. Dermatas, "Random-Walker Monocular Road Detection in Adverse Conditions Using Automated Spatiotemporal Seed Selection," IEEE Transactions on Intelligent Transportation Systems, vol. 14, no. 2, pp. 527-538, Jun., 2013.
- [5: Alonso et al. 2012] Ignacio Parra Alonso, David Fernández Llorca, Miguel Gavilan, Sergio Álvarez Pardo, Miguel Ángel Garcia-Garrido, Ljubo Vlacic, and Miguel Ángel Sotelo, "Accurate Global Localization Using Visual Odometry and Digital Maps on Urban Environments," IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 4, pp. 1535-1545, Dec., 2012.

- [6: Durrant-Whyte and Madhavan 2005] Hugh Durrant-Whyte, and Raj Madhavan, “2D map-building and localization in outdoor environments,” *Journal of Robotic Systems*, vol. 22, no. 1, pp. 45-63, Jan., 2005.
- [7: Hata and Wolf 2016] Alberto Y. Hata, and Denis 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, Feb., 2016.
- [8: Sivaraman and Trivedi 2013] Sayanan Sivaraman, and Mohan Manubhai Trivedi, “Integrated Lane and Vehicle Detection, Localization, and Tracking: A Synergistic Approach,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 906-917, Jun., 2013.
- [9: Cui et al. 2016] Dixiao Cui, Jianru Xue, and Nanning Zheng, “Real-Time Global Localization of Robotic Cars in Lane Level via Lane Marking Detection and Shape Registration,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1039-1050, Apr., 2016.
- [10: Cheng 2011] Hong Cheng, “Autonomous Intelligent Vehicles Theory, Algorithms, and Implementation,” Springer, London, 2011.