# Multi-Lane Detection and Tracking Using Vision for Traffic Situation Awareness

Yukihiro Tsukamoto  Masahiro Ishizaki  Akihito Hiromori  Hirozumi Yamaguchi  Teruo Higashino
*Graduate School of Information Science and Technology*
*Osaka University*
Suita, Japan

***Abstract*—Situation awareness in the transport system has been significant for understanding the cause of traffic congestion, potential risks of accidents, and so on, and many efforts have been made to recognize surrounding situations by vehicle onboard cameras for low-cost sensing. These methods use computer vision technologies to recognize objects in the vicinity of the vehicle in the video image. However, to correctly recognize the positional relationship between the surrounding objects and the vehicle, lane detection, particularly multiple lane detection, should robustly be performed. In this paper, we propose a method for detecting multiple lanes from video images taken by onboard cameras. Since frame-by-frame spatial lane detection does not often work in a severe environment on multi-lane roads, the proposed method leverages temporal change detection, which enables complementing such lanes that are not detected in a frame and eliminating wrongly-detected ones for robust detection and tracking. In the experimental using the video images, we have achieved an accuracy of more than 90% in multi-lane environments with different conditions such as lane boundary occlusion by other vehicles, nighttime, rain, side road junctions.Besides, we achieved 96.47% accuracy to detect lanes on which vehicles are driving, while 88.70% in the comparative method.**

***Index Terms*—Lane detection, Dashcam, Spatiotemporal image**

## I. INTRODUCTION

In recent years, the number of vehicles equipped with in-vehicle cameras has been increasing, and it is expected that the video images taken by these cameras will help, in a crowdsensing manner, to detect obstacles on roads, to analyze regular traffic patterns for city planning and to recognize abnormal traffic situations such as local traffic congestion and potential risks of accidents. In particular, with the recent advance and enhancement of deep neural network-based image processing techniques, it has become possible to accurately recognize moving objects in a video image such as surrounding vehicles, bikes, and pedestrians in the vicinity of a vehicle. Furthermore, knowing the positions of those moving objects enables us to recognize the context in the scene. For example, in Japan, we have a serious social problem caused by aggressive drivers. They often drive with high speed and frequently change the lanes. Even worse, some malicious drivers do tailgating to intensively threaten others. Fortunately, as more vehicles have in-vehicle cameras for their safety and digital forensic purposes, such behavior is often recorded by some vehicles there and shared with others if necessary. On the other hand, since analyzing several high-volume videos is not realistic, automatic detection of the intended context is expected [1]. Specifically, to recognize the context in the in-vehicle camera views correctly and automatically, the positional relation between the ego-vehicle and each of the surrounding vehicles, bikes, and pedestrians is required, and lane information is most significant to identify their locations – in which lanes vehicles and bikes are, and where pedestrians are, either on sidewalks or crossing.

There has been a lot of vision-based lane detection and tracking methods so far. However, autonomous driving cars with powerful processors and multiple cameras on their roofs and corners use very different algorithms – they can use rich information taken from different angles, and no detection failure is allowed. Meanwhile, *ego-lane detection* using only a single camera has been investigated mainly for safe driving assistance purposes such as lane departure warning (LDW). Nevertheless, multi-lane analysis has not been investigated so far as it is more difficult to recognize the existence of other lanes. Without detailed road map information, it is not easy to know how many lanes are there and in which lane the ego-vehicle is. Therefore it is more challenging to detect and track not only the ego-lane but also other lanes only from in-vehicle camera vision.

In this paper, we propose a vision-based method for detecting and tracking multiple lanes. Most of the existing vision-based methods for lane detection and tracking have the issues perform lane detection for each captured frame. However, the detection often fails when lanes in frames are not photographed, particularly when a lane is drawn as a dotted line, the lane is blurred, or a strong edge exists temporarily due to the presence of light or shadow of sunlight. To tackle this challenge, our method leverages temporal change detection as well as frame-by-frame spatial detection of lanes and develop a novel technique to combine both detections. Notably, we focus on the fact that inter-frame features where the number of lanes and their positions do not change significantly in a short period, and the detection result in a frame has to be similar to the previous frame. Therefore, by inheriting the detection results in the previous frames, we may supplement missing lanes in some frames and eliminate falsely detected ones.

Through the experiment using the video images (12,000 frames as a total) taken in different situations and annotated by Amazon Mechanical Turk, we have achieved an accuracy of more than 90% in multi-lane environments with different conditions such as lane boundary occlusion by other vehicles, nighttime, rain, side road junctions. Besides, we achieved 96.47% accuracy to detect lanes on which vehicles are driving, while 88.70% in the comparative method.

## II. Related Work

Lane detection and tracking have been well investigated so far in different contexts and with different configurations. We survey these in the following.

Several methods have used line segment detection techniques along with edge detection. For example, the work in [2] emphasizes the boundaries of while/yellow lanes and applies the Hough transform. However, edge detection is primarily affected by extremely high or low illuminance regions in the image and is not robust to such images that include white guardrails (there are many of them in Japan) and shade of large buildings.

Some methods leverage vanishing points in the images, considering the properties that parallel lines converge to the vanishing point in a projected 2-D image. For example, the methods in [3] and [4] both use Gabor filters to compute the directions of scene texture and estimate vanishing points from them. Notably, [4] also calculates the likelihood of the directions of scene texture, where the likelihood in the areas apart from the vanishing point becomes low to determine the reliability of calculated vanishing points. However, it is known that scene texture directions do not often work in a city environment where scenes do not have apparent features of textures. The work in [5] estimates vanishing points by an edge detection technique and finds line segments that have certain overlapping with lines converging to the vanishing points. This method achieves high accuracy by limiting the angles of detected lanes in the images and taking the edges' length and width into account. The work in [6] focuses on mitigating lane detection processing overhead. It applies the stochastic Hough transform for lane detection and a particle filter for lane tracking to reduce the computational overhead of per-frame lane detection.

One drawback of line segment detection-based methods is that curved lanes cannot be detected. To tackle this challenge, the work in [7] modifies the target images that include detected edges using predefined nine different curvature radiuses and finds the curves that fit with them. This method also applies an improved line segment detection that combines region detection using colors of lanes to avoid uncertainty. However, color information is usually expensive for real-time processing of frames. The work in [8] extracts horizontal scan lines with a constant height in the sequence of frames and aligns them along time so that the boundaries of lanes at the height can be detected. Since this method is not dependant on line segment detection, it can be applied to the curved lanes.

On the contrary, there have been lane detection and tracking methodologies for autonomous driving cars. The work in [9] applies the inverse perspective mapping (IPM) [10] to generate top-view images, and lanes are detected by unifying those images using the vehicle's speed. The curved lanes can be identified from the fused images by a least-squares fitting algorithm. The work also mentions how to track the detected lanes by a Kalman filter. The work in [11] presents a lane departure warning system by using IPM-based lane detection. Although IPM needs angles and other parameters of cameras, it can eliminate the effect of perspective. Hence this is useful for such a vehicle like an autonomous driving car where cameras are mounted, adjusted, and calibrated with predetermined parameters.

### A. Our Contributions

Most of the existing lane detection and tracking methods have been focusing on lane departure warning systems. To this end, most of the methods are for detecting lanes on which vehicles are driving, called *ego-lane* detection. On the other hand, the proposed method tackles the issue of identifying multiple lanes in urban areas for situational awareness via each in-vehicle camera (dash-cam). The multi-lane detection only by a video from a single, non-calibrated camera is not straightforward since there are two significant challenges; (i) non-ego-lanes are likely to be obscured by other vehicles, and (ii) no information about the presence of other lanes is assumed. As for the first issue, we strongly need spatial complementing to recover the lost lanes by other vehicles (or by some other reasons such as weather and sunlight). For the second issue, we need a quick adaptation to lane changes performed by the recording vehicles and to the change of the number of lanes on the roads. As far as we investigate, this is the first approach that tackles multi-lane detection and tracking to solve these issues.

## III. LANE DETECTION METHOD

In our method, firstly, in the pre-processing step, we extract the area where lanes may be visible and create a spatiotemporal image that is used to extract the trajectory of a lane boundary position at a certain height in each frame. Following these processes, the lane detection uses the spatiotemporal image created in the pre-processing stage to estimate the positions of the lane boundaries at multiple heights on the frame and obtains a group of points indicating the lane boundary positions. Subsequently, lane detection is performed by clustering these point groups for each lane boundary line. Besides, to improve the accuracy of lane detection, a lane boundary-completing process that uses time-series information is performed to remove false detections and recover the lost detections by comparing them with past detection results.

### A. Preprocessing

*1) Region of Interest Selection in Frame:* To reduce the overhead to scan unnecessary regions in each frame where lanes do not usually appear, we first identify the regions of interest (RoI) that contain the lanes. For this purpose, we identify regions with flares (due to sunlight reflections on the windshield) and hoods. For flare detection, we find those regions with high illuminance over multiple frames. To realize this, we convert the frames into gray-scale images and inspect

Fig. 1. Region of Interest Selection Examples

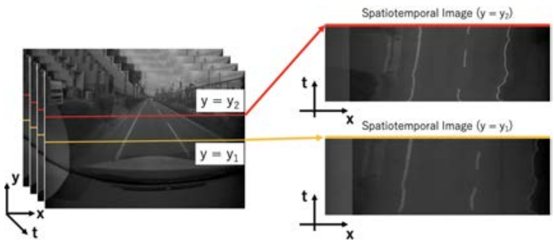


Fig. 2. Lane Boundaries on Scan Lines

the illuminance. The sky region can also be identified in this way. The hood regions can be identified as those without a specific change of colors over multiple frames. Then we identify RoI as a rectangle near the center of the frame that is not included in the above non-RoIs. Figure 1 shows RoIs examples as red rectangles, which are automatically detected by our methods.

*2) Spatiotemporal Image Creation:* The method of creating spatiotemporal images and detecting lane boundary positions at a certain height is based on the technique in [12]. Since vehicles generally follow lanes most of the time, it is assumed that the horizontal position of the lane boundary that appears on a horizontal line, called *scan line*, at a constant height in the video image, does not change significantly over time (Fig. 2). By aligning the scan lines at a certain height in chronological order, we can obtain the locus of the positions of the lane boundaries in the spatiotemporal image. Then each scan line in the spatiotemporal image is shifted horizontally to minimize the sum of the difference of the boundaries on the two subsequent scan lines at heights $y$ and $y+1$. Specifically, for the horizontal line at height $y+1$, we cut off both edges for $C$ pixels ($C = 30$ in our experiment) and calculate such an integer $\delta$ that minimizes SAD (Sum of Absolute Difference) of illuminance:

$$\delta = \arg \min_{\delta_x \in [-C,C]} \sum_{x \in C..X-C} |l(x+\delta_x, y+1) - l(x,y)| \quad (1)$$

where $l(x, y)$ is the illuminace at coordinates $(x, y)$ and $X$ is the width of the image. This process produces a spatiotemporal image in which the lane boundary positions are aligned.

We note that scan lines should be contained in RoI, and multiple scan lines are used for the robust detection of lanes. In our experiment, we have used eight scan lines.

### B. Lane Detection

*1) Estimation of Lane Boundary Position:* To estimate the lane boundary positions, for each scan line in a frame, we use a spatiotemporal image created using the sequence of scan lines in the last $k$ frames (that is, the height of the image is $k$) as explained in the previous section. In [12], which we refer to for the creation of spatiotemporal images using scan lines, each spatiotemporal image is binarized, and the Hough transform is applied to detect lane boundaries. On the other hand, the proposed method detects, for each horizontal line in a spatiotemporal image, such X-locations with a significant change of illuminance against their next X-locations. Then, from the detected X-locations, it identifies those with such significant changes at different heights as lane boundary positions. It is worth mentioning that the binarization of the entire spatiotemporal image does not often emphasize the boundaries of other lanes, which are apart from the camera, as those lines tend to appear thin due to the distance. Furthermore, binarization is not robust to temporary changes in lighting conditions, which frequently happen in driving.

Figure 3 illustrates how the lane boundary estimation using spatiotemporal images is performed. We let $u$ denote this frame of interest (*i.e.*, the current target frame) in which we want to detect lane boundary positions. In this example, eight scan lines, $\#1$ – $\#8$ are set on the image where the height of $j$-th ($\#j$) scan line is denoted as $y_j$, and for each scan line, we have a corresponding spatiotemporal image denoted as $st_j$, which is a collection of the scan lines of the last $k$ frames. By obtaining the frame $u$, for each scan line $\#j$, we extract the line from $u$ and add the line to $st_j$ at height $k$ by shifting the image down by one pixel and apply SAD as explained in Section III-A2. Then at height $y_j$ in the updated image $st_j$, we detect a set $X_j$ of X-locations with significant changes of illuminance, that is,

$$X_j = \{\ x \ \mid \ |l(x+1, y_j) - l(x, y_j)| > \alpha\} \quad (2)$$

where $\alpha$ is a detection threshold. Then, we further filter $X_j$ to obtain $X'_j$ so that only X-locations where significant changes were also detected in some of the last $k$ frames. That is,

$$X'_j = \{\ x \ \mid \ x \in X_j \ ; \ \sum_{w \in 1..k} l(x, w) > \beta\} \quad (3)$$

where $\beta$ is a threshold. Finally, for each pair of $j$-th scan line and $x \in X'_j$, we regard $(x, y_j)$ as the lane boundary position.

Figure 3 shows two spatiotemporal images $st_2$ and $st_5$ that correspond to the scan lines $\#2$ and $\#5$, respectively (red lines highlight these lines while others are yellow ones). Yellow circles highlight the detected lane boundary positions.

*2) Ego Lane Detection:* We detect ego lanes and other lanes in this order for lightweight detection of ego lanes and other lanes. For ego lane detection, we attempt to exclude such points that are falsely-detected ones or that correspond to the other lanes. We employ the following heuristics; firstly, for each scan line, we leave the three points closest to the center of the frame, and the others are eliminated from this detection process. Furthermore, because curved lanes that appear in the upper region of the frame are usually hard to detect by line approximation, we may also eliminate those at a certain height in the frame. Finally, the remaining points are bisected into left and right based on the center of the frame. The left and right boundaries of the ego lane can be estimated from those points on the left and right sides, respectively.

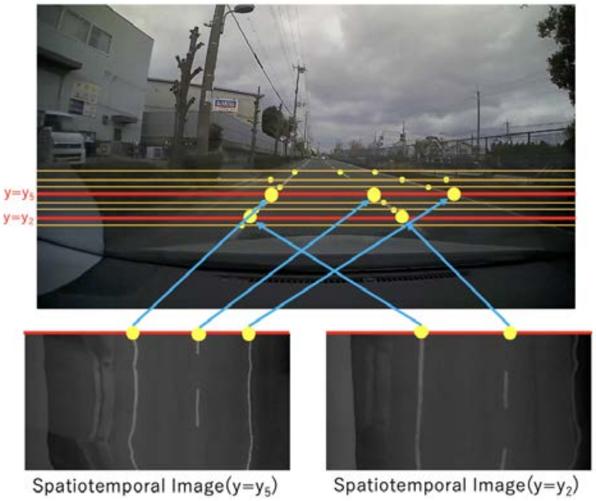


Fig. 3. Scan Lines in Frame and Spatiotemporal Images


Fig. 4. Case of Removing False Positives Based on Lane Width

Specifically, the estimation is done by line approximation over the set of points at each side. According to the investigation in [13], the angles of the left and right boundaries of ego lane are usually in $[30°, 55°]$ and $[125°, 150°]$, respectively, and we follow this insight. We let $N_{left}$ and $N_{right}$ denote the number of points at the left and right sides, respectively. Basically, we conduct an exhaustive search ($2^{N_{left}}$ and $2^{N_{right}}$ ) but take a greedy selection policy where the search is stopped if the coefficient of determination of the examined approximate line is close to one to avoid computational overhead.

*3) Ego Lane Detection Temporal Assessment:* For this purpose, we observe the significant change between the detection in the current frame and that in the last few frames, and reject the result in the current frame as erroneous if there is inconsistency. Specifically, we focus on the difference of the lane width in the current frame and the average width in the last few frames. If this difference is higher than a specific threshold value, the detection result in the previous frame is used instead of that in the current frame. We note that the possibility of actual lane width change on the road should be considered, and to cope with this, we update the average width even though the change is above the threshold, and the result is replaced by the last result. This enables us to adapt to the width change in the real world only with a few frames delay. Figure 4 shows example cases of acceptance and rejection of the estimated boundaries.

*4) Detection of Other lanes:* To detect other lanes, the estimated ego lane is used. Firstly, the vanishing point is

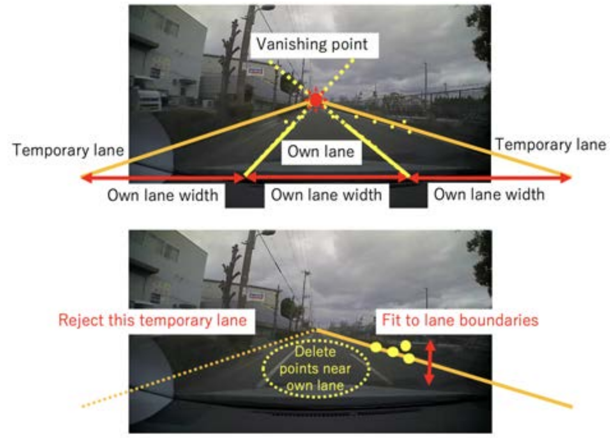


Fig. 5. Detecting Other Lanes

calculated by finding the intersection of the boundaries of the ego lane. Secondly, the width of the ego lane at a certain height is calculated, and using this width, lanes on the left and right of the ego lane are temporarily assumed (Fig. 5). Thirdly, the distance from each assumed lane boundary to each point detected in the previous section is calculated. The lane is regarded to be present if multiple points are within a certain distance from the assumed boundary. Fourthly, the assumed boundary is best fitted to those points by rotating and shifting. Finally, the lane detection is completed by completing the boundaries of other lanes by using the time series information described later.

### C. Lane Boundary Completion Using Time Series Information

It is expected that the boundaries of ego and other lanes do not change significantly in a short time. Therefore, the time series information can be used to supplement the lane boundary detection results. In this section, we describe the completion of lane boundaries using time-series information.

*1) Backward Lane Boundary Completion:* If the lane boundaries that were detected in the last few frames are not detected, the *backward lane boundary completion* is performed. To verify that the two boundaries in consecutive frames are identical, the position and angle of the lane boundary in the frame is compared with that in the last frames. Figure 6 shows an example of the completion of the other lanes. The upper left image of Figure 6 shows the detected boundaries in the $(N-1)$-th frame (say $L1$ to $L3$) , and the upper right shows those in the $N$-th frame (say $L4$ and $L5$). Then the bottom image shows the result of completion in the $N$-th frame using the previous frame. The lane boundaries L4 and L5 correspond to L1 and L2, respectively, but that corresponding to L3 is not detected in the $N$-th frame. Therefore, L3 is used to supplement the missing one in the $N$-th frame.

*2) Forward Lane Boundary Completion:* Besides the backward completion, forward completion can be performed using buffered frames. We let $N$ denote the current frame. With the buffer of size $h$, allowing the delay of $h$-frames to output the final result, the result of $(N-h)$-th frame can be complemented using the detection results of $(N-h+1)$-th to $N$-th frames.

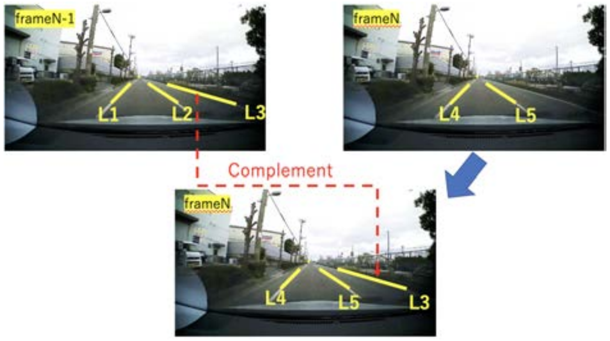


Fig. 6. Example of completion of a detected leakage lane

For such a scenario that does not need time-critical analysis, this completion is effective to improve the accuracy.

We may consider different types of completions, but in this paper, we develop an algorithm by majority vote in the future frame set. Based on the vote's decisions, the missing boundaries are completed using the nearest frames that have the boundaries, and the falsely-detected ones are eliminated. This completion is particularly useful in situations such as the just after the initialization of the detection (no buffered frames for backward completion) or just after an intersection (no boundaries are detected when an ego-vehicle passes an intersection).

## IV. EVALUATION

### A. Experimental Settings

We used two vehicles with the commercial off-the-shelf car-mounted cameras. We drove the vehicles on the roads where the lane boundaries were drawn. To evaluate the performance of lane detection, video images were taken on various road conditions – on straight roads, on roads with side streets, on roads with large curves, on roads with vehicles in the vicinity, in rainy environments, in the nighttime, and with road surface markings. The total number of frames used in the evaluation was 12,000. For reference, the video images in these conditions are shown in Figure 7 with lane boundary detection results by our method.

### B. Evaluation and Discussion of The Proposed Method

To evaluate the proposed method, we have obtained a detection result every second, and assess the detection accuracy. The results are shown in Tables I and II.

Table I shows the detection accuracy of ego lanes and other lanes, without the forward completion, while Table II shows that with the forward completion having 30 frames (one second).

From these results, it is shown that sufficient accuracy is achieved (96.47% and 90.76 on average for ego and other lanes, respectively) even with severe conditions. The accuracy in the nighttime is lower compared with other cases since only several lane boundary positions were detected due to insufficient lighting. Besides, the accuracy in the nighttime for other lanes is almost 70% because the headlights did not illuminate the lanes.

TABLE I
EXPERIMENTAL RESULT (W/O FORWARD COMPLETION)

| Condition | Ego [%] | Other [%] |
|---|---|---|
| Straight Roads | 97.30 | 91.89 |
| with Side Roads | 94.20 | 89.86 |
| with Large Curvature | 94.12 | 87.06 |
| with Surrounding Vehicles | 95.16 | 88.71 |
| Rain | 94.41 | 88.11 |
| Night | 92.31 | 71.79 |
| Average | 94.55 | 86.64 |

TABLE II
EXPERIMENTAL RESULT (W/ FORWARD COMPLETION)

| Condition | Ego [%] | Other [%] |
|---|---|---|
| Straight Roads | 98.11 | 98.11 |
| with Side Roads | 97.10 | 85.51 |
| with Large Curvature | 95.23 | 91.67 |
| with Surrounding Vehicles | 96.72 | 93.44 |
| Rain | 95.77 | 91.55 |
| Night | 92.30 | 84.62 |
| Average | 96.47 | 90.76 |

The forward completion contributes to improving accuracy by 1.92% and 4.12% in ego and other lanes, respectively, but even without the completion, we can enjoy less delayed detection with sufficient accuracy as shown in Table I.

As for comparison, the method in [12] from which we exploit the concept of spatiotemporal image creation achieved **88.70%**, while we achieved 96.46%. This is because we have the following advantage and enhancement. Firstly, we use illuminance to detect boundaries in spatiotemporal images. This significantly improves robustness. Secondly, we design a fully new algorithm for boundary estimation considering temporal dependencies. Fusing several techniques, our method could achieve stable accuracy under different situations, including severe cases such as rain, night, and with side roads (lane boundaries disappear at the side road merging points, which may often confuse detection algorithms).

Finally, our method estimated the boundaries with an average of 0.125 s in our current implementation. Since some parts of the implementation can be parallelized, we expect that our method can predict boundaries up to 30 frames in realtime with a modern multi-core processor.

To demonstrate the detection results, we stored the video in the following URL: https://www.dropbox.com/sh/kadu7e3vmq1pjim/AACaRQMfeAUPYEJbDfbN00Yua?dl=0.

## V. CONCLUSION AND FUTURE WORKS

We have proposed a method for detecting and tracking multiple lanes by effectively fusing both frame-by-frame lane detection and temporal dependencies over the sequence of frames, using a video taken by a single dash-cam. The several techniques that leverage temporal dependencies and illuminance-based lane detection in spatiotemporal images enable us to robustly detect multiple lanes, under a variety of situations in normal driving in urban areas (rain, curves, with windshield shining, lane occlusion by other vehicles, side roads, etc.). The experimental results using 12,000 frames in the videos recorded by dash-cams in our daily commute, we

Fig. 7. Video Images Used for Evaluation

have shown that even with severe situations, more than 90% accuracy is achieved only using a single camera video.

As part of our future work, we are working to leverage this detection technique for situation awareness using vision, as we have done using smartphones [14]. Our final goal is to understand the traffic situations automatically – identifications of objects (vehicles, pedestrians, and bikes) in the scenes using the start-of-the-art object detection techniques, and localization of those objects in terms of the traffic regulations – which lanes the surrounding vehicles are driving, the pedestrians' walking positions on narrow roads, etc. for collecting and sharing the potential risks about accidents. These information can be shared with other vehicles through connected cars, which are expected to become increasingly popular. This may allow the integration of information obtained from various vehicle perspectives to generate new information. Our lane detection is useful for this purpose, as well as other general use cases.

## VI. Acknowledgement

This work was supported by "Empirical Research and Development for Solving Regional Issues by Data Utilization (No. 200)", the Commissioned Research of National Institute of Information and Communications Technology (NICT), JAPAN.

## References

[1] H. Yamaguchi, "Keynote: Situational awareness platform for city transportation," in *Proceedings of 5th IEEE International Workshop on Pervasive Context-Aware Smart Cities and Intelligent Transport System (PerAwareCity2020)*, March 2020, p. 1.

[2] N.S.Aminuddin, M.M.Ibrahim, N.M.Ali, S.A.Radzi, W.H.M.Saad, and A. M. Darsono, "A New Apprpach To Highway Lane Detection By Using Hough Transform Technique," *Journal of ICT*, vol. 16, no. 2, pp. 244 – 260, Dec. 2017.

[3] C.Rasmussen, "Grouping dominant orientations for ill-structured road following," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, July. 2004, pp. 470–477.

[4] H.Kong, J.Y.Audibert, and J. Ponce, "General Road Detection From a Single Image," *IEEE Transactions on Image Processing*, vol. 19, pp. 2211 – 2220, Aug. 2010.

[5] J. H. Yoo, S.-W. Lee, S.-K. Park, and D. H. Kim, "A Robust Lane Detection Method Based on Vanishing Point Estimation Using the Relevance of Line Segments," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, pp. 3254 – 3266, Mar. 2017.

[6] G. Liu, F. Wörgötter, and I. Markelić, "Combining Statistical Hough Transform and Particle Filter for robust lane detection and tracking," *IEEE Intelligent Vehicles Symposium*, Aug. 2010.

[7] Y.He, H.Wang, and B.Zhang, "Color-based road detection in urban traffic scenes," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 4, pp. 309–318, Dec. 2004.

[8] S. Jung, J. Youn, and S. Sull, "Efficient Lane Detection Based on Spatiotemporal Images," *IEEE Transactions on Industrial Informatics*, vol. 17, pp. 289–295, 2016.

[9] J.Wang, B.Kong, and H. T.Mei, "Lane detection algorithm based on temporal–spatial information matching and fusion," *CAAI Transactions on Intelligence Technology*, vol. 2, pp. 154 – 165, Nov. 2017.

[10] A.M.Muad, A.Hussain, S.A.Samad, M.M.Mustaffa, and B.Y.Majlis, "Implementation of inverse perspective mapping algorithm for the development of an automatic lane tracking system," in *Proceedings of the 2004 IEEE Region 10 Conference (TENCON)*, vol. 1, Nov. 2004, pp. 207–210.

[11] Y. C.Y.Kuo and S.M.Yang, "On the Image Sensor Processing for Lane Detection and Control in Vehicle Lane Keeping Systems," *Sensors*, vol. 19, Apr. 2019.

[12] A. Rasouli, I. Kotseruba, and J. K. Tsotsos, "Understanding Pedestrian Behavior in Complex Traffic Scenes," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 1, pp. 61–70, Mar. 2018.

[13] Z. Fang, D. Vázquez, and A. M. López, "On-Board Detection of Pedestrian Intentions," *Sensors (Basel)*, vol. 17, no. 10, p. 2193, Sep. 2017.

[14] R. Akikawa, A. Uchiyama, A. Hiromori, H. Yamaguchi, T. Higashino, M. Suzuki, Y. Hiehata, and T. Kitahara, "Smartphone-based risky traffic situation detection and classification," in *Proceedings of International Workshop on Pervasive Computing for Vehicular Systems (PerVehicle2020)*, March 2020, pp. 1–8.