A Novel Method for Ground-truth Determination of Lane Information through a Single Web Camera

Keyu Ruan, Lingxi Li, Guobiao Song, Jing Xia, and Hongyu Pang

Abstract— **The high-definition (HD) map is critical for the localization and motion planning of connected and automated vehicles (CAVs). With all the road and lane information pre-scanned in a certain area, the vehicles can know its position with respect to the lane marks and roadside, and hence make better decisions on planning future trajectories. A common issue, however, is the accuracy of the scanned outputs from different data sources. Because of the limitations of online maps (e.g., zooming and stretching in their image layers), visualizing the data in the bird's eye view on maps cannot satisfy the accuracy requirement of being the ground-truth system. To this end, a feasible method that can combine sensing data from different sources and obtain reliable ground-truth information is necessary. In this paper, we develop a novel method to transform the data points from the bird's eye view to the view angle of a web camera installed on the windshield of the ego vehicle. In such a case, the position of landmarks from the captured frames of the camera can be used as the ground-truth. In particular, we take the lane marking detection outputs from the Mobileye system as the reference for a better accuracy. We evaluate the proposed method using the field data on highway I-75 in Michigan, USA. The results show that this method has achieved a very good accuracy of over 90% for location determination of lane information. The main contribution of this paper is that the proposed method can be more intuitive and reliable than using the traditional maps in bird's eye view.**

I. INTRODUCTION

With the rapid growth of the auto industry, many advanced techniques have been applied to self-driving vehicles. In [1], the authors presented a method of guidance control for parallel parking, which can help unskilled drivers with parking tasks. Authors in [2] studied the strategies of decision-making and driving state control of unmanned vehicles. Interval type-2 fuzzy sets, fuzzy comprehensive evaluation, and fuzzy control rules were used to realize the objectives. In [3], the authors surveyed the works on applying model predictive control algorithms to the traffic signal management and control. In [4], driver behavior data were collected to evaluate the in-vehicle camera-based driver state monitoring systems, which can be an objective and reliable source for the improvement of these products.

Besides aforementioned works, high-definition (HD) map is also an essential technique and is widely used for vehicle localization and path planning based on pre-obtained environment information. The authors in [5] presented a method for HD map updating with the cellular network that meets the requirements for highly automated driving. This work is claimed to be the first one that correlates the data requirements with the network infrastructure. In [6], the authors provided a control scheme for predictive cruise control, which utilized the HD map information. The proposed predictive cruise control system helped achieve a higher rate of fuel-saving. Other than highway and urban driving, HD map technique has also been used in many other situations. The authors in [7] used HD maps in transfer vehicles in smart factories, in order to perform better interactions with the environment. They proposed an HD map update strategy to solve the open problem for sustainability of the performance. The HD map technique has also been used for underwater vehicles in [8], where the authors tried to analyze water situations and find solutions for water resources management.

Although the HD map technique has been well received in the research community, its accuracy remains to be evaluated. The authors in [9] used a framework that contains an automated ground-truth process, which provides the user with a unique visualization of the captured video and quickly generates the ground-truth information. They have also used this framework in a night-time lane detection system presented in [10]. The authors in [11] presented an efficient solution in evaluating the road marking detection algorithms. They have validated this evaluation process with a virtual database.

The aforementioned works used HD map to obtain ground-truth information of land markings and road edges on highway. In this paper, we study a different problem. In our setup, the map provider gave us the dataset of lane locations based on latitude and longitude coordinates, whose precision level needs to be evaluated. Due to the intentionally zooming and stretching of their image layers, the accuracy of the data points cannot be simply evaluated from any online maps. Hence, we deployed a variety of onboard sensors on an experimental vehicle and collected field driving data on highway I-75 for the ground-truth evaluation. Among the sensors, the Mobileye camera system can provide reliable lane marking detection outputs. However, it is very difficult to obtain raw data from the Mobileye system for further calculations. Therefore, in this paper, we present a method to transform these sensor outputs along with many other bird's eye view data points into the view angle of a single web camera installed on the windshield of the vehicle. In this way, we can directly compare the data points from the Mobileye detection results with the camera video frames,

The work in this papr was supported in part by the QAII Joint Fund Project 16-7-1-4-JCH.

Keyu Ruan and Lingxi Li are with the Transportation Active Safety Institute (TASI), Department of Electrical and Computer Engineering, Indiana-University Purdue-University Indianapolis, Indianapolis, IN 46202, USA. Emails: {keyuruan, LL7}@iupui.edu.

Guobiao Song, Jing Xia, and Hongyu Pang are with the Department of Autonomous Driving, Aptiv PLC, Troy, MI 48098, USA. Emails: {guobiao.song, jing.xia, hongyu.pang}@aptiv.com.

which is more intuitive and efficient for post-processing after the driving data are collected.

Because the web camera and the vehicle sensing system have different coordinate systems, we firstly introduce a method to transform the sensor data into the camera coordinate system. Then, we develop a method to transform the data from bird's eye view into the camera's view (with different roll-pitch-yaw angles). The field data are then applied to evaluate the accuracy of the proposed approach.

The remainder of this paper is organized as follows. Section II introduces a method for coordinate transformation. Section III presents the approach for view angle transformation. An illustrative example is shown in Section IV. The evaluation of the proposed method using the field data is presented in Section V. Conclusion and future research directions are summarized in Section VI.

II. Coordinate Transformation

Since the web camera and the vehicle sensing system have different coordinates, a general approach for coordinate transformation is necessary. We use a minivan as our experimental vehicle and measure the distances in three dimensions between the camera and the origin of the vehicle sensor coordinate system (which has been pre-calibrated at the middle line of the testing vehicle and aligned with the position of the Mobileye processing unit). Fig. 1 shows the relationship between two coordinate systems before the transformation.

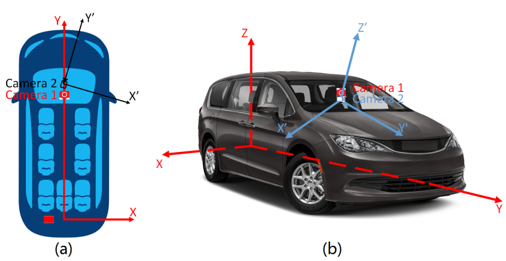

Fig. 1. Two coordinate systems before the transformation.

In Fig. 1, *X-Y-Z* plane represents the coordinates of the vehicle sensing system and *X'-Y'-Z'* plane represents the coordinates of the web camera. In particular, camera 1 is the Mobileye system camera and camera 2 is the web camera. The red box marks the position of the processing unit of the Mobileye system. The web camera was installed on the windshield right below the Mobileye camera. Based on the Mobileye system calibration manual and our in-vehicle measurements, several important position values are obtained and listed in Table I, where $Offset_X$, $Offset_Y$, $Offset_Z$ are the distance between the origin of plane *X-Y-Z* and camera 1 in X, Y, and Z directions, respectively. $Offset_C$ is the vertical distance between the two cameras. $Height_Z$ represents the distance from the origin of *X-Y-Z* plane to the ground.

The process of coordinate transformation includes the three-dimensional Euler angle rotation rules mentioned in [12] along with the origin position shifting. The order of the transformation should be rotating around $z-y-x$ axes and then shifting position. Therefore, based on the Euler angle rotation rules, we have:

TABLE I
Position Values in the Testing Vehicle

Parameters	Values
$Offset_X$	0
$Offset_Y$	2.437 m
$Offset_Z$	0.822 m
$Offset_C$	0.142 m
$Height_Z$	0.608 m

$$P' = R_z \times R_y \times R_x \times T \times P, \qquad (1)$$

where P is a point in plane *X-Y-Z* and P' is a point in plane *X'-Y'-Z'*. R_x, R_y, and R_z are rotation matrices around X, Y, and Z axes, respectively, and T is the position shifting matrix. The way to calculate these matrices follows the standard way of calculating Euler angle transformation.

Since the orientation of plane *X'-Y'-Z'* is hard to be aligned with plane *X-Y-Z*, the values of the *Roll*, *Pitch*, and *Yaw* angles cannot be obtained precisely. Therefore, we combine the matrices R_x, R_y, R_z and T in Equation (1) to form the coordinate system transformation equation $P' = M \times P$ as:

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & t_x \\ a_{21} & a_{22} & a_{23} & t_y \\ a_{31} & a_{32} & a_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \qquad (2)$$

The transform matrix M in Equation (2) can be used to transform any point in *X-Y-Z* plane into the *X'-Y'-Z'* plane. In order to calculate the twelve variables in M, we need four points with their position values in both *X-Y-Z* plane and *X'-Y'-Z'* plane. The values in *X-Y-Z* can be obtained from the Mobileye system. The idea is illustrated in Fig. 2 (a), where the red dots between the two different shapes of the lane marks are the data points that can be detected by the Mobileye system.

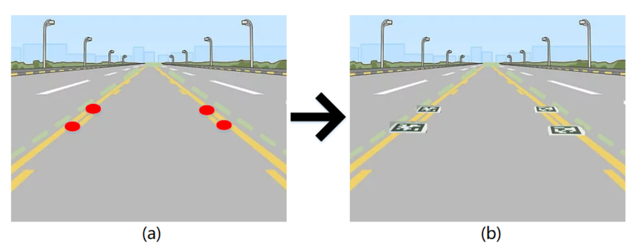

Fig. 2. Four data points in the Mobileye system for illustration.

The remaining task is to obtain the position values of the four points in plane *X'-Y'-Z'*. To this end, we apply the Apriltag algorithm proposed in [13], which enables distance measurement for the monocular camera, whose accuracy is claimed to be higher than 95% for distance detection. We place the recognizable patterns for the Apriltag algorithm

TABLE II

FOUR DATA POINTS IN BOTH COORDINATE SYSTEMS

ID	x	y	z	x'	y'	z'
1	-1.4838	6.7317	-0.608	-1.7511	4.1932	-0.660
2	1.4398	21.5681	-0.608	0.1330	19.1614	1.615
3	1.2851	37.0049	-0.608	-1.0955	34.5123	3.948
4	-1.5459	14.1499	-0.608	1.3635	11.5710	0.461

between the two different shapes, aligning the center of patterns with the red dots, as shown in Fig. 2 (b). The Apriltag will output the position information in the coordinate system of the camera, which is the *X'-Y'-Z'*. Therefore, we have the required position values of the four data points and can perform coordinate transformation. As an example, Table II shows the position values of the four data points that we use in both coordinate systems.

Use these four data points, we can calculate the coordinate system transformation matrix M in Equation (3) as follows.

$$M = \begin{bmatrix} 0.9690 & -0.0702 & -0.0996 & 0.1638 \\ 0.0698 & 0.9951 & 0.9548 & -1.5703 \\ 0.0106 & 0.1513 & 0.7212 & -1.1862 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3)$$

III. VIEWING ANGLE TRANSFORMATION

After we obtain the data points in the web camera coordinate system, we can perform the viewing angle transformation. The essential task in this part is to find the viewing angle transformation matrix N. In order to do that, a rectangular aluminum frame has been made in order to obtain the transformation matrix. As shown in Fig. 3, six pieces of recognizable patterns of Apriltag are attached to the frame: Four at the corners and two in the middle of two edges.

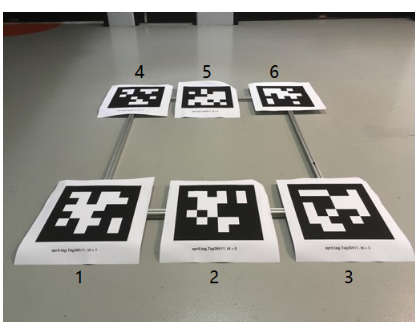

Fig. 3. Frame and distance measurement patterns.

The aluminum frame is placed at a certain position in front of the vehicle (as shown in the left image of Fig. 4), which ensures that all six patterns can be detected and several requirements are satisfied: 1) Patterns 1, 2, and 3 should have the same z value. Similarly for Patterns 4, 5, and 6. 2) Patterns 2 and 5 should have x values equal to zero. 3) Patterns 1 and 3 should have opposite x values. Similarly for Patterns 4 and 6.

With these requirements satisfied, we can ensure that the frame is in the right position for later coordinates calculation. The Apriltag algorithm is applied again for distance measurement. The user interface of the Apriltag framework shows all the detected patterns, marked with red circles on the frame and yellow dashed lines connecting the camera and patterns on the 3D schematic, as seen in Fig. 4.

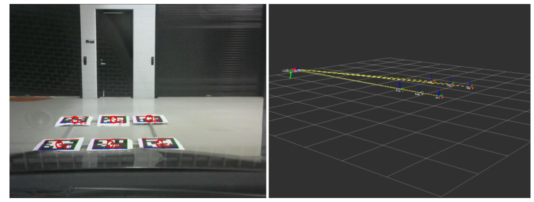

Fig. 4. The user interface which shows detected patterns. **Left**: Detected patterns marked with red circles in the camera frame. **Right**: Detected patterns and the relative positions towards the camera (connected with yellow lines).

The output of Apriltag framework also provides other information, such as the distance from each pattern to the camera in x (right+/left-), y (forward+/backward-), and z (upward+/downward-) direction, respectively; the Roll, Pitch, and Yaw angles for each pattern; and the pixel coordinates of the center of detected patterns on the video frame. The pixel coordinates are the key values to calculate the transformation matrix. In [14], a viewing angle transformation algorithm, called *Perspective Transform*, was proposed. In this algorithm, in order to obtain the transformation matrix, besides the detected pixel coordinates as the transformation destination, we need another set of pixel coordinates as the transformation source. Fig. 5 shows the idea of how the other set of pixel coordinates are chosen.

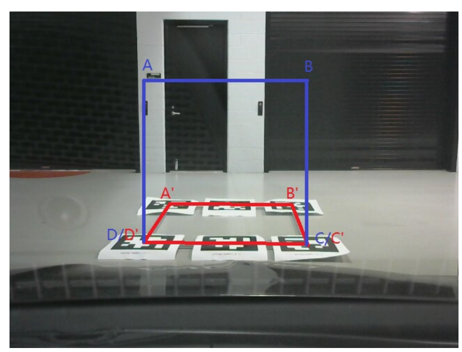

Fig. 5. Two pixel coordinates to calculate the transformation matrix: (1) **Blue ABCD**: the frame in the bird-eye viewing angle; (2) **Red A'B'C'D'**: the frame in the camera viewing angle.

In Fig. 5, the blue quadrilateral *A-B-C-D* spans the plane of the transformation source and the red quadrilateral *A'-B'-C'-D'* spans the plane of the transformation destination. Clearly, pixel coordinates of points A', B', C', and D' are obtained from the pattern detection. Since our goal is to transform the bird's eye view into the camera view, the shape of transformation source should match the way when seeing from the top. Hence, *A-B-C-D* should be in the shape of

TABLE III

SAMPLE DATA POINTS ON THE FRAME

Points	Coordinates in *A-B-C-D*	Coordinates in *A'-B'-C'-D'*
A/A'	(226, 276)	(193, 63)
B/B'	(399, 276)	(424, 63)
C/C'	(424, 330)	(424, 330)
D/D'	(193, 330)	(193, 330)

a rectangle, which has exactly the same dimension as the aluminum frame. We let *A-B-C-D* and *A'-B'-C'-D'* share the bottom edge. Thus, the unit length of each pixel can be calculated. In our case, the segment *CD* or $C'D'$ has 231 pixels on the image and 1.408 meters in length, thus each pixel represents 1.408/231=0.006095 meter. We then calculate the number of pixels on the segment *AD* or *BC* has. The length of *AD* and *BC* are both 1.627 meters, which implies that these two segments on image have 1.627/0.006095=267 pixels. Finally, since we have adjusted the position of the aluminum frame which ensures that the quadrilateral *A-B-C-D* is a strict rectangle, we can calculate the pixel coordinates for points *A* and *B*. In Table III, we list one set of pixel coordinates of the four patterns at corners of the aluminum frame.

We now calculate the transformation matrix with the two sets of pixel coordinates. The authors in [14] proposed a method to calculate the transformation matrix, as shown in Equation (4) below.

$$\begin{cases} a_{11}x + a_{12}y + a_{13} - a_{31}xX - a_{32}Xy = X \\ a_{21}x + a_{22}y + a_{23} - a_{31}xY - a_{32}yY = Y \end{cases} \tag{4}$$

where coefficients a_{ij} form the transform matrix A; $[x\ y\ 1]^T$ is the coordinate of the source point that needs to be projected; $[X\ \ Y\ \ 1]^T$ represents the coordinate of the point as the projection destination. We have:

$$\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{5}$$

Use the four pairs of source and destination points shown in Table III, we have eight equations to calculate eight parameters in matrix N, which is obtained below.

$$N = \begin{bmatrix} 0.7070 & -0.2880 & 95.0484 \\ 8.0439 & -0.1020 & 266.9913 \\ 2.9311 & -0.0009 & 1 \end{bmatrix} \tag{6}$$

Matrix N transforms every data point in the plane spanned by *A'-B'-C'-D'* to the plane spanned by *A-B-C-D*. Since our final objective is to obtain plottable values on video frames, we need to convert the coordinate values of the data points from meters into pixels.

To do this, we need to use the four points A, B, C, and D as the reference points. Then, by using the closest reference from the data point and applying Equations (7) and (8) below, we can convert the coordinate values from meters to pixels.

TABLE IV

VALUES OF THE REFERENCE POINTS

Points	$x_{ref-meter}$	$y_{ref-meter}$	$x_{ref-pixel}$	$y_{ref-pixel}$
A	-0.7061	6.6197	193	63
B	0.7152	6.5743	424	63
C	0.6993	4.9195	193	330
D	-0.7083	4.9226	424	330

TABLE V

RAW MOBILEYE BIRD'S EYE VIEW DATA POINTS IN VEHICLE COORDINATE SYSTEM

ID	Y (L&R&C)	X (L)	X (R)	X (C)
1	6.14376	-1.9713	1.5326	-0.2149
2	11.6766	-2.0543	1.4434	-0.3004
3	17.2094	-2.1549	1.3362	-0.4037
4	22.7421	-2.2725	1.2112	-0.5245
5	28.8394	-2.4212	1.0530	-0.6774
6	34.9366	-2.5894	0.8739	-0.8506
7	41.0338	-2.7765	0.6742	-1.0436

$$x_{pixel} = x_{ref-pixel} + \left(\frac{x_{meter} - x_{ref-meter}}{l_{unit}}\right) \tag{7}$$

$$y_{pixel} = y_{ref-pixel} - \left(\frac{y_{meter} + y_{ref-meter}}{l_{unit}}\right) \tag{8}$$

where $[x_{meter},\ y_{meter}]^T$ and $[x_{pixel},\ y_{pixel}]^T$ represents the data point before and after conversion, respectively. l_{unit} represents the length per pixel. Note that in the bird's eye view, the length distribution is uniform in all directions. $[x_{ref-meter},\ y_{ref-meter}]^T$ and $[x_{ref-pixel},\ y_{ref-pixel}]^T$ are the reference points, the choices of values depend on which reference point is the closest. Table IV shows the options of different reference points.

After we obtain the pixel coordinates of the data points, the transformation in Equation (5) can be applied. Thus, the final data points can be calculated. Next, we use a set of raw sensor and map data as an example to illustrate the proposed transformation process.

IV. AN ILLUSTRATIVE EXAMPLE

With the information of coordinate system transformation matrix M, viewing angle transformation matrix N, and the method to convert the coordinates into pixels, we could convert all the data points collected by the sensors into the camera view, which can then be plotted on the video frame captured by the web camera. We use one of the video frames along with the corresponding sensor data points that are at the same timestamp. Tables V and Table VI list several sets of data points, which include the left and right lane marking points detected by the Mobileye system along with the calculated center line, and the left and right lane marking points provided by the HD map provider along with the calculated center line, whose values are shown in columns marked with labels L, R, and C. These data points are all in plane *X-Y-Z*, i.e., the vehicle coordinate system.

We transform these raw data points from the vehicle coordinate system into the camera coordinate system by

TABLE VI
RAW HD MAP BIRD'S EYE VIEW DATA POINTS IN VEHICLE COORDINATE SYSTEM

ID	Y (L)	X (L)	Y (R)	X (R)	Y (C)	X (C)
1	None	None	6.831	1.6608	6.1473	-0.1972
2	18.1356	-2.3003	22.515	1.3616	11.6790	-0.3084
3	41.2890	-2.9019	42.258	0.8147	17.2107	-0.4196
4	55.2323	-3.3913	60.119	0.1328	22.7424	-0.5308
5	71.1554	-4.1141	75.984	-0.6507	28.8376	-0.6933
6	90.4356	-5.1849	89.861	-1.4823	34.9328	-0.8557
7	107.413	-6.2992	107.70	-2.7349	41.0281	-1.0182

TABLE VII
MOBILEYE DATA POINTS IN PIXELS

ID	Y (L)	X (L)	Y (R)	X (R)	Y (C)	X (C)
1	377	-115	362	549	369	228
2	225	100	222	383	224	244
3	185	157	183	336	184	247
4	166	182	165	314	166	248
5	154	198	154	299	154	249
6	147	207	147	289	147	248
7	142	213	142	282	142	247

multiplying the transformation matrix M. Then convert the obtained results from meters into pixels in the bird's eye view. At last, by multiplying the transformation matrix N, the calculated pixel coordinates in the camera view are the final data points, which are shown in Tables VII and VIII.

We plot these data points onto the video frame that corresponds to the same timestamp. The plot is shown in Fig. 6, where the two blue lines represent the left and right lane markings detected by the Mobileye camera and the two red lines represent the left and right lane markings from the HD map data. We also see the orange and green lines, both representing the middle line of the lane, which are calculated from the Mobileye and HD map lane marking data, respectively. From Fig. 6, we can see that the Mobileye data fits the lane markings in the video frame well.

V. EVALUATION OF THE PROPOSED METHOD

In this section, we use real highway driving data to evaluate the accuracy of the proposed method. All data points are obtained from the driving route on highway I-75, starting from Aptiv headquarter at Troy, Michigan and ending at Flint, Michigan. Fig. 7 shows the driving route on the Google map.

Our objective for the evaluation is to check whether the detected lane marking data points from the Mobileye sensors

TABLE VIII
MAP DATA POINTS IN PIXELS

ID	Y (L)	X (L)	Y (R)	X (R)	Y (C)	X (C)
1	None	None	327	529	369	231
2	181	156	166	320	224	243
3	142	210	141	294	184	247
4	135	218	133	270	166	248
5	130	222	129	261	154	248
6	127	222	127	255	147	248
7	125	221	125	248	142	248

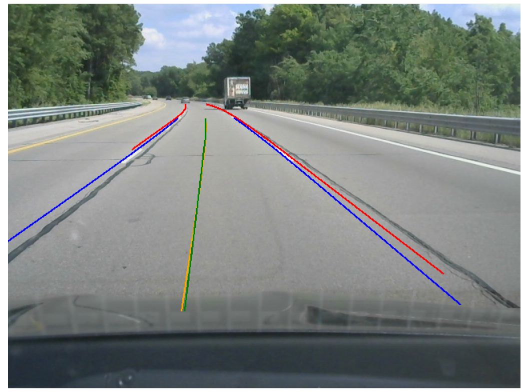

Fig. 6. Transformed data points. Color coding: (1) **Blue**: Left and right lane markings detected by the Mobileye camera; (2) **Red**: Left and right lane markings from the HD map data; (3) **Orange**: Middle line of the lane calculated from the Mobileye data; (4) **Green**: Middle line of the lane calculated from the HD map data.

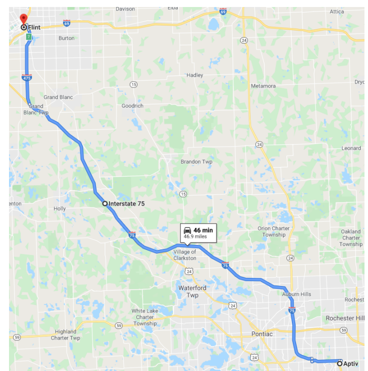

Fig. 7. Driving route for data collection.

are accurate or not after transformation (in the view angle and coordinate of the web camera). In Fig. 8, we show the way of accuracy labeling. The lane markings have the boundaries marked with the white lines. When the data points are within the area between two white lines (including borders), as shown in Fig. 8(a), we have a full accuracy. When the data points are outside of the area, as shown in Fig. 8(b), we find the distances from the data points to the closest border and estimate the errors.

The inaccuracy in these scenarios can be the consequence of extreme uneven road surface, e.g., bumps, pit-holes, cracks, or slopes, which can lead to strong shakes in the y and z directions. Since our camera is calibrated in a stationary vehicle inside the garage, which is in the ideal scenario, the calculated transformation matrices might not fit the real driving scenarios well when the vehicle drives on an uneven surface. Hence, the inaccurate situations happen mostly at lane-changing cases (since the road surface between lanes is usually uneven). To make the error values more intuitive, we can roughly convert the unit from pixels into meters using Equation (9):

$$\Delta d_{meter} = \Delta d_{pixel} * \frac{x_{right-meter} - x_{left-meter}}{x_{right-pixel} - x_{left-pixel}} \tag{9}$$

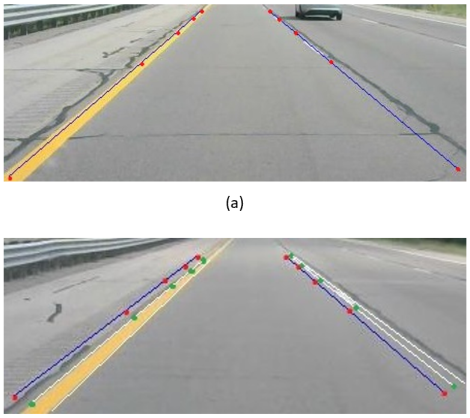

Fig. 8. Example of data labeling.

TABLE IX
SUMMARY OF LABELING RESULT OF ALL 204 CASES

Term	Value
#Total	204
#Lane Keeping	187
#Lane Changing	17
#Full Accurate(Lane Keeping)	184
#Full Accurate(Lane Changing)	0
%Full Accurate(Over All)	90.19%
%Full Accurate(Lane Keeping)	98.40%
Error (Lane Keeping)	$<$ 3 centimeters
Error (Lane Changing)	$>$ 10 centimeters

With the proposed method, we manually labeled 204 sets of the Mobileye data points to assess the performance. From the labeling results, we found that we should divide the driving conditions into lane-keeping and lane-changing. For the lane-keeping cases, there are 184 cases with full accuracy and three inaccurate cases, which have errors less than 3 centimeters. On the other hand, the 17 lane-changing cases are all inaccurate. The smallest error is around 10 centimeters, which is much larger than lane-keeping cases. We summarize our results in Table IX.

The above analysis shows that our method has over 90% fully accurate cases among the driving data sets, which will be even higher for lane-keeping cases only. Thus, this method is effective for determining the ground-truth information using a single web camera.

VI. CONCLUSIONS

In this paper, we proposed a method to transform the road and lane data from bird's eye view to the view of a single web camera, which can be used as the ground-truth of vehicle sensing data. This method is based on the detected lane marking information from the Mobileye system. Two view angle transformation algorithms have been applied. More than two hundred sets of highway driving data have been used to evaluate the proposed method. The evaluation results showed that this method has achieved an overall accuracy of over 90% for all data samples.

The traditional ground-truth methods that use online GPS maps have the problems of image layer stretching and zooming. Our method compares the collected data with the lane markings in the same frames. The lane markings are much easier to be recognized in the collected frames than any open-source maps. Hence, we can ensure that the comparison is intuitively and reliably.

This work can be extended in several ways. We can use a stereo camera instead of the current monocular one. In such a case, the accuracy of distance detection can be improved. It is also interesting to develop a simpler and more flexible process for camera calibration in order to deal with the bad road conditions in the future.

REFERENCES

[1] J. Tan, C. Xu, L. Li, F.-Y. Wang, D. Cao, and L. Li, “Guidance control for parallel parking tasks,” *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 301-306, January 2020.

[2] X. Zhao, K. Yan, H. Mo, and L. Li, “Type-2 fuzzy control for driving state and behavioral decision of unmanned vehicle,” *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 178-186, January 2020.

[3] B.-L. Ye, W. Wu, K. Ruan, L. Li, T. Chen, H. Gao, and Y. Chen, “A survey of model predictive control methods for traffic signal control,” *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 3, pp. 623-640, May 2019.

[4] R. Tian, K. Ruan, L. Li, J. Le, J. Greenberg, and S. Barbat, “Standardized evaluation of camera-based driver state monitoring systems,” *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 3, pp. 716-732, May 2019.

[5] F. Jomrich, J. Schmid, S. Knapp, A. Höß, R. Steinmetz, and B. Schuller, “Analysing communication requirements for crowd sourced backend generation of HD Maps used in automated driving,” in *Proc. 2018 IEEE Vehicular Networking Conference*, pp. 1-8, Taipei, Taiwan, 2018.

[6] H. Chu, L. Guo, B. Gao, H. Chen, N. Bian, and J. Zhou, ”Predictive cruise control using high-definition map and real vehicle implementation,” *IEEE Transactions on Vehicular Technology*, vol. 67, no. 12, pp. 11377-11389, Dec. 2018.

[7] M. O. TAS, H. Serhan YAVUZ, and A. YAZICI, ”Updating HD-maps for autonomous transfer vehicles in smart factories,” in *Proc. 2018 IEEE International Conference on Control Engineering Information Technology*, pp. 1-5, Istanbul, Turkey, 2018.

[8] A. Yaemjaem, N. Sutthisangiam, A. Sompan, N. Sa-ngiam, and P. Jindasee, ”Development of high-definition 3D mapping system for water resources management,” in *Proc. 2018 IEEE International Conference on Advanced Informatics: Concept Theory and Applications*, pp. 219-223, Krabi, Thailand, 2018.

[9] A. Borkar, M. Hayes, and M. T. Smith, ”An efficient method to generate ground truth for evaluating lane detection systems,” in *Proc. 2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1090-1093, Dallas, TX, 2010.

[10] A. Borkar, M. Hayes, and M. T. Smith, ”A novel lane detection system with efficient ground truth generation,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 365-374, March 2012.

[11] R. Marc, G. Dominique, and P. Evangeline, ”Generator of road marking textures and associated ground truth applied to the evaluation of road marking detection,” in *Proc. 2012 IEEE International Conference on Intelligent Transportation Systems*, pp. 933-938, Anchorage, AK, USA, 2012.

[12] K. Shoemake, “Euler angle conversion,” in *Graphics Gems IV (Paul S. Heckbert, Ed.)*, pp. 222-229, 1994.

[13] Edwin Olson, “AprilTag: A robust and flexible visual fiducial system”, in *Proc. IEEE International Conference on Robotics and Automation*, pp. 3400-3407, Shanghai, China, May 2011.

[14] I. Carlbom and J. Paciorek, ”Planar geometric projections and viewing transformations,” *ACM Computing Surveys*, vol. 10, no. 4, pp. 465-502, 1978.