

## ROLL-SENSITIVE ONLINE CAMERA ORIENTATION DETERMINATION ON THE STRUCTURED ROAD

Yuanxian Huang<sup>1</sup>, Jian Zhou<sup>1</sup>, Bijun Li<sup>1\*</sup>, Jinsheng Xiao<sup>2</sup>, Yongxing Cao<sup>1</sup>,

<sup>1</sup> State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, China,  
(yxhuang, jianzhou, lee, yongxing.cao)@whu.edu.cn

<sup>2</sup> Electronic Information School, Wuhan University, China, xiaojs@whu.edu.cn

**KEY WORDS:** Camera Calibration, Roll-sensitive, Autonomous Vehicles, Inverse Perspective Mapping, Lane Detection.

### ABSTRACT:

Online camera calibration technology can estimate the pose of the camera onboard in real time, playing an important role in many fields such as HD map production and autonomous vehicles. Some researchers use one vanishing point (VP) to calculate the pitch and yaw angle of the onboard camera. However, this method assumes that the roll angle is zero, which is impractical because of the inevitable installation error. This paper proposes a novel online camera orientation determination method based on a longitudinal vanishing point without the zero-roll hypothesis. The orientation of the camera is determined in two steps: calculating the pitch and yaw angles according to vanishing point theory, and then obtaining the roll angle with lane widths constraint which is modeled as an optimization problem. To verify the effectiveness of our algorithm, we evaluated it on the nuScenes dataset. As a result, the rotation error of the roll and pitch angle can achieve  $0.154^\circ$  and  $0.116^\circ$  respectively. Also, we deployed our method in the “Tuyou”, an autonomous vehicle developed by Wuhan University, and then tested it in the urban structured road. Our proposed method can reconstruct the ground space accurately compared with previous methods with zero-roll hypothesis.

### 1. INTRODUCTION

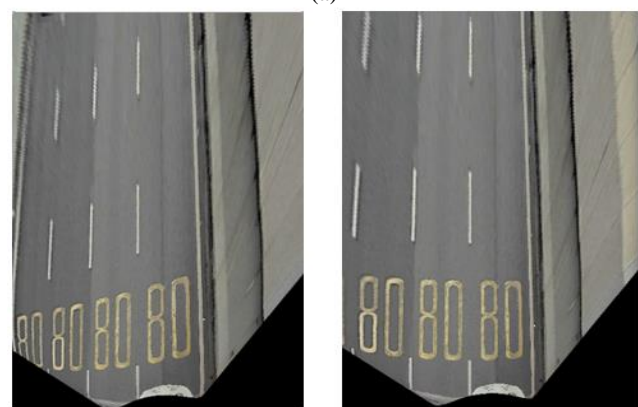
As cameras are widely equipped on mobile platforms like autonomous vehicles and robots, accurate extrinsic calibration becomes increasingly important for vision-based algorithms. Traditional manual calibration methods tend to use control points in the calibration field with special targets. However, these methods are impractical in outdoor scenes where cameras should be automatically calibrated online. Due to the ups and downs of the road, the poses of onboard cameras are changing continuously during driving, which introduces the errors related to the pose of the camera and influences the robustness of vision-based applications, such as lane keeping assist and lane departure warning.

Some researchers utilize VO (Vision Odometry) to measure the relative orientation change of the camera (Jeong and Kim, 2016). Based on VO, other researchers obtain the relative pose between camera and other onboard sensors whose poses are known, via hand-eye calibration (Tsai and Lenz, 1989; Wang et al, 2019). These methods require a large amount of computation and heavily rely on texture information. The main disadvantage of these methods is that they can only obtain relative poses between two adjacent frames. If one frame is lost, the calibration system is exposed to the risk of invalidation.

Three orthogonal vanishing points in the image can be used to calculate the intrinsic and extrinsic parameters of the camera (Hartley and Zisserman, 2003; Orghidan et al, 2012). However, for a camera mounted in the front of the vehicle, it is difficult to obtain three stable orthogonal vanishing points. On the structured road, the intersection of lanes in image space provides a stable



(a)



(b)

(c)

**Figure 1.** Bird's eye view (BEV) and the effect of camera's roll angle. (a) An image captured by an onboard camera whose roll angle is not zero. (b) BEV with zero-roll hypothesis. (c) BEV considering the roll angle

\* Corresponding author

longitudinal vanishing point. While another two orthogonal VPs, namely, the horizontal vanishing point and the vertical vanishing point, often intersect at infinity, which suffer numerical instability. Moreover, on the structured road, it is impractical to find three orthogonal VPs all the time. Therefore, many researchers only use the longitudinal VP to calculate the camera's pitch and yaw angle and assume that the roll angle is zero (Zhao et al, 2014; Lee and Zhou, 2016; Yang et al, 2016). However, this zero-roll hypothesis is inappropriate. On the one hand, the roll angle is not strictly zero due to the installation error. On the other hand, inevitably, the roll angle changes continuously during driving. This will directly affect the stability of autonomous vehicle applications such as lane keeping assist.

Researchers usually use inverse perspective mapping (IPM) to map image space into ground space (Bertozzi and Broggi, 1998; Aly, 2008; Oliveira et al, 2015) and obtain bird's eye view (BEV) image to measure the distance to the adjacent lane lines, in applications like lane keeping assist. In BEV, targets on the road's surface have the same scale as the real world. Generally, the widths of lanes on the structured road are equal. As a result, lanes in BEV usually have the same width. Besides, if the actual lanes' widths are known, the widths measured in the BEV should be equal to their actual values. However, the inappropriate zero-roll hypothesis will introduce errors. The error of the roll angle will deform the lanes' widths in BEV as shown in Figure 1. When the camera rotates anticlockwise around the optical axis of the camera, the lanes on the left become narrower and the right lanes become wider. The zero-roll hypothesis will bring more errors to vehicle lateral positioning.

This paper firstly deduces a formula for calculating the pitch and yaw angle based on VP without the zero-roll hypothesis. Then, a pipeline is proposed to estimate VP and measure lanes' widths. Finally, fixing the pitch and yaw angle, we further model the solution of the roll angle as an optimization problem with lane width constrain in BEV.

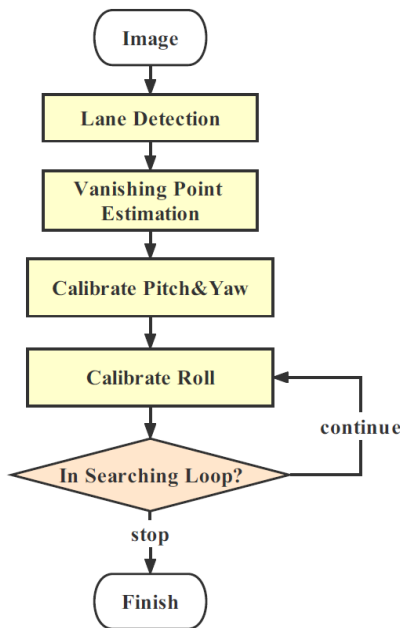


Figure 2. Flow chart of the proposed method

## 2. GENERAL FORMULA TO CALCULATE PITCH AND YAW

### 2.1 Problem Statement

As shown in Figure 3, there are mainly three coordinate systems in this paper: the road coordinate system  $(X, Y, Z)$ , the camera coordinate system  $(X_c, Y_c, Z_c)$  and the image coordinate system  $(x, y)$ . The origin of the camera coordinate system coincides with the origin of the road coordinate system. The camera is mounted at a height  $H$  above the ground. The rotation matrix is represented by  $\mathbf{R}$  and the translation vector is represented by  $\mathbf{t}$ . Since the origin of the road coordinate system and the camera coordinate system overlaps,  $\mathbf{t} = \mathbf{0}$ .  $\mathbf{R}$  can be expressed as a function of the roll angle  $\theta_r$ , pitch angle  $\theta_p$  and yaw angle  $\theta_y$ . In photogrammetry, the Euler angular order of the  $\mathbf{R}$  matrix is often roll-pitch-yaw. In this paper, the Euler angular order is adjusted to pitch-yaw-roll, to avoid the influence of the roll angle on the pitch and yaw angles. The  $\mathbf{R}$  matrix is defined as follows.

$$\begin{aligned} \mathbf{R}(\theta_p, \theta_y, \theta_r) &= \mathbf{R}_{pitch} \mathbf{R}_{yaw} \mathbf{R}_{roll} \\ &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & -\sin \theta_p & -\cos \theta_p \\ 0 & \cos \theta_p & -\sin \theta_p \end{pmatrix} \begin{pmatrix} \cos \theta_y & -\sin \theta_y & 0 \\ \sin \theta_y & \cos \theta_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ &\quad \begin{pmatrix} \cos \theta_r & 0 & -\sin \theta_r \\ 0 & 1 & 0 \\ \sin \theta_r & 0 & \cos \theta_r \end{pmatrix} \end{aligned} \quad (1)$$

Where,  $\mathbf{R}_{pitch}$ ,  $\mathbf{R}_{yaw}$ ,  $\mathbf{R}_{roll}$  are the rotation matrices corresponding to each angle. In order to be consistent with the definition in most IPM literatures, this paper defines the pitch angle in the horizontal direction as 0, which differs from the value defined by Euler's angle by  $\pi/2$ , so the  $\mathbf{R}_{pitch}$  has different forms.

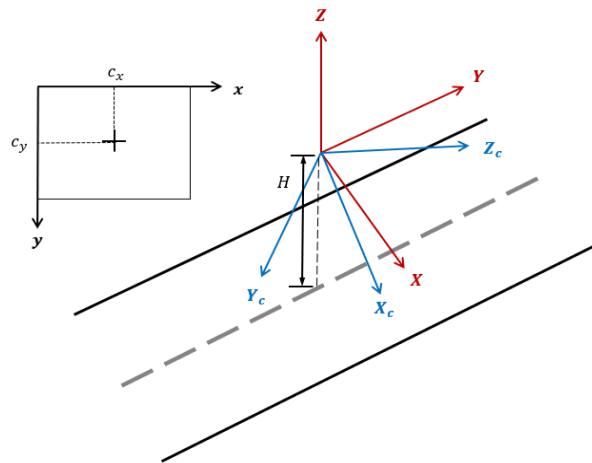


Figure 3. Definition of coordinate systems.

For any given point  $\mathbf{P} = (X, Y, Z)$  in space. The relationship between the road coordinates and image coordinates of  $\mathbf{P}$  can be described by the pinhole camera model.

$$\lambda \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \mathbf{K} \mathbf{R}_{pitch} \mathbf{R}_{yaw} \mathbf{R}_{roll} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (2)$$

where  $\mathbf{K}$  = intrinsic matrix,  $\lambda$  = normalization scale

## 2.2 Definition of Vanishing Point

As shown in Figure 4, the point on the line passing through the point  $X_0$  and with direction  $D$  can be expressed in the following formula.

$$X(\tau) = X_0 + \tau D \quad (3)$$

Vanishing point  $V$  is the point on the line at infinity (Hartley and Zisserman, 2003).

$$V = \lim_{\tau \rightarrow \infty} X(\tau) \quad (4)$$

The image coordinate of VP is only related to the direction  $D$ . For simplicity, let  $X_0 = 0$ . Also, since the  $Y$  axis is parallel to the direction of the road, let  $D = (0, 1, 0)^T$ . Combining equation (3)(4), the VP can be expressed in equation (5).

$$V = \lim_{\tau \rightarrow \infty} (0, \tau, 0)^T \quad (5)$$

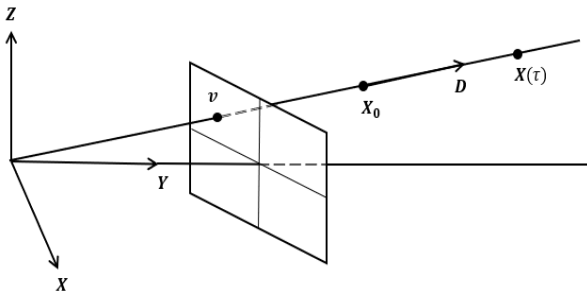


Figure 4. Definition of the vanishing point

## 2.3 Calibration Based on Vanishing Point

Combining equation (2) and (5), VP's image coordinates  $(v_x, v_y)$  can be expressed in equation (6).

$$\begin{aligned} \begin{pmatrix} v_x \\ v_y \\ 1 \end{pmatrix} &= \lim_{\tau \rightarrow \infty} KR_{pitch}R_{yaw}R_{roll} \begin{pmatrix} 0 \\ \tau \\ 0 \end{pmatrix} \\ &= \lim_{\tau \rightarrow \infty} KR_{pitch}R_{yaw} \begin{pmatrix} 0 \\ \tau \\ 0 \end{pmatrix} \end{aligned} \quad (6)$$

Let,

$$KR_{pitch}R_{yaw} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

Then equation (6) can be expressed in the form of the collinear equation.

$$\begin{cases} v_x = \lim_{\tau \rightarrow \infty} \frac{a_{12}\tau}{a_{32}\tau} = \frac{a_{12}}{a_{32}} = c_x - \frac{f_x}{\cos \theta_p} \tan \theta_y \\ v_y = \lim_{\tau \rightarrow \infty} \frac{a_{22}\tau}{a_{32}\tau} = \frac{a_{22}}{a_{32}} = c_y - f_y \tan \theta_p \end{cases} \quad (7)$$

where  $(c_x, c_y)$  = camera optical center  
 $(f_x, f_y)$  = the focal length

Based on equation (7), the pitch and yaw angles can be calculated.

$$\begin{cases} \theta_p = \arctan \frac{c_y - v_y}{f_y} \\ \theta_y = \arctan \left( \frac{c_x - v_x}{f_x} \cos \theta_p \right) \end{cases} \quad (8)$$

In equation (8), the pitch and yaw angles are independent of the roll angle. During the derivation, only the order of Euler angles is specified, and no assumptions are made on the roll angle. In other words, the pitch and yaw angles can be obtained from equation (8) regardless of the change of the roll angle. This is important for the further calculation of the roll angle.

## 3. VANISHING POINT ESTIMATION AND LANE WIDTH MEASUREMENT

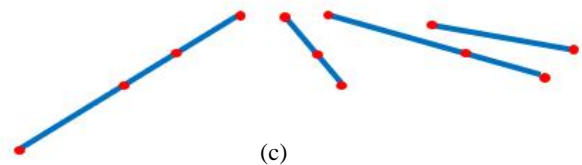
Up until now, the pitch and yaw angles of the camera can be calculated from the VP of the lane lines. Besides, as mentioned before, to further estimate the roll angle, the lane width is used as a constraint to solve the optimization problem. Therefore, it is necessary to detect the lane lines on the image, estimate the vanishing point, and measure lanes' widths.



(a)



(b)



(c)

Figure 5. Pipeline of lane detection. (a) Original image. (b) Segmentation mask. (c) Lanes expressed as piecewise linear model

### 3.1 Lane Detection and Vanishing Point Estimation

Recently, lane detection methods based on convolutional neural networks (CNN) have become mainstream and achieve state-of-

the-art performances (Chen et al, 2017; Zou et al, 2019; Qin et al, 2020). As shown in Figure 5, a self-developed CNN is used to get the segmentation mask of the input image. Then, following a series of post-processing processes including denoising (Xiao, J et al, 2018), contour detection (Suzuki, 1985), Douglas-Puke sampling, and center points extraction, we finally obtain lanes in the image and describe it as a piecewise linear model.

Then, we estimate VP based on Gaussian spheres (Collins and Weiss, 1990; Rother, 2002) using the lanes detected before. As shown in Figure 6, a Gaussian sphere is a unit sphere whose center is the camera's optical center. Each line in the image corresponds to a great circle on the Gaussian sphere. Two great circles of two parallel lines in space will intersect at a point on the Gaussian sphere. The ray from the sphere's center to the intersect is the vanishing direction (VD)  $D_v$  (Barnard, 1983; Lee and Yoon, 2019). VD is orthogonal to the normal of great circles of parallel lines. Suppose the two endpoints of a line segment  $s$  in the image are  $P_1$  and  $P_2$ . Then the normal vector  $n$  of the great circle corresponding to the line segment  $s$  can be expressed as  $n = (K^{-1}P_1) \times (K^{-1}P_2)$ . By orthogonality, we have  $n \cdot D_v = 0$ . If there are  $N$  line segments, we have

$$AD_v = 0 \quad (9)$$

where  $A = [n_1, \dots, n_N]^T$

Equation (9) can be solved by singular value decomposition (SVD). Then VP's image coordinate  $v = (v_x, v_y)^T$  can be calculated.

$$v = KD_v \quad (10)$$

Since lane lines are represented in the piecewise linear model, this method can also work on unstructured roads. The time complexity of this algorithm is  $o(N)$ .

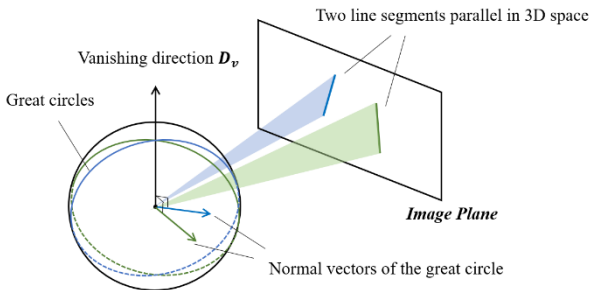


Figure 6. The Gaussian sphere.

### 3.2 Roll-Sensitive Inverse Perspective Mapping

IPM is a technology which can map image coordinates  $(x, y)$  to road coordinates  $(X, Y)$ . For simplicity, previous studies (Bertozzi and Broggi, 1998; Aly, 2008) usually assume that the roll angle is zero. However, such an assumption may bring non-negligible errors to the subsequent lane width calculation when the roll angle is large.

According to equation (2), the relationship between  $(x, y)$  and  $(X, Y)$  can be expressed by the following equation.

$$\begin{pmatrix} X \\ Y \\ -H \end{pmatrix} = \frac{1}{\lambda} R(\theta_p, \theta_y, \theta_r)^{-1} K^{-1} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (11)$$

Let

$$R^{-1}K^{-1} = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} \quad (12)$$

Substituting into equation (11), we get

$$\begin{cases} X = -H \frac{b_{11}x + b_{12}y + b_{13}}{b_{31}x + b_{32}y + b_{33}} \\ Y = -H \frac{b_{21}x + b_{22}y + b_{23}}{b_{31}x + b_{32}y + b_{33}} \end{cases} \quad (13)$$

### 3.3 Lane Width Measurement

In this paper, the piecewise linear model is used to describe the lane lines, i.e., the lane lines are represented as broken lines. A broken line consists of vertices and edges. Denote  $L = \{P_1, \dots, P_n, s_1, \dots, s_{n-1}\}$ , where  $L$  denotes a broken line,  $P_i$  denotes a vertex,  $s_i$  denotes an edge and  $n$  denotes the number of the vertices. For any given point  $P$ , the distance  $d(P, L)$  from the point  $P$  to the broken line  $L$  is defined as the shortest distance to all edges.

$$d(P, L) = \min_{i=1, \dots, n-1} dist(P, s_i) \quad (14)$$

Where,  $dist(P, s_i)$  is the distance from the point  $P$  to the line where edge  $s_i$  is located.

Suppose there are two broken lines  $L_1, L_2$ . Let  $L_1 = \{P_1^1, \dots, P_n^1, s_1^1, \dots, s_{n-1}^1\}$  and  $L_2 = \{P_1^2, \dots, P_m^2, s_1^2, \dots, s_{m-1}^2\}$ . The distance between two broken lines  $d(L_1, L_2)$  is defined as the average distance from each vertex to another broken line.

$$d(L_1, L_2) = \frac{1}{mn} \left( \sum_i^n d(P_i^1, L_2) + \sum_i^m d(P_i^2, L_1) \right) \quad (15)$$

## 4. CALCULATE ROLL USING LANE WIDTH CONSTRAINT

In Section 2, the pitch and yaw angles are calibrated based on the VP. Then we need to estimate the roll angle. In BEV, the lane widths are influenced by the roll angle. When the camera rotates clockwise around the  $Y$  axis, the lanes on the left become wider and the right lanes become narrower. When the camera calibration parameters are accurate and the ground is approximately flat, the lanes in BEV obtained by IPM should match their scale in real world. This characteristic can be used to accomplish the calibration of the roll angle.

### 4.1 Roll Calibration Without Prior Lane Width

In structured roads, the widths of adjacent lanes are usually equal. This means that the widths of different lanes in the BEV should be equal and the variance of the widths of multiple lanes should be zero. Utilize this property, an optimization problem can be built.

$$\theta_r^* = \operatorname{argmin}_{\theta_r} \sum_{i=1}^{N-1} (d_i(\theta_r) - \bar{d})^2 \quad (16)$$

where  $N$  = the number of lane lines  
 $d_i(\theta_r) = d(L_i, L_{i+1})$ ,  $\bar{d}$  = the mean of  $d_i$

When fixing the pitch and yaw angle, the lane’s width is a function of the roll angle  $\theta_r$ .

Since the relationship between  $\theta_r$  and  $d_i$  is difficult to express explicitly, the above optimization problem is difficult to solve directly in the continuous numerical space. Therefore, we use a search-based algorithm to solve  $\theta_r$ . Assume that  $\theta_r$  obeys the normal distribution, i.e.,  $\theta_r \sim N(\mu, \sigma)$ . According to the property of normal distribution, 99.7% of the values lie within three standard deviations of the mean, which is called as three-sigma rule. So, let  $\theta_r \in [\mu - 3\sigma, \mu + 3\sigma]$ . After limiting the upper and lower bounds of  $\theta_r$ , if the resolution of  $\theta_r$  is further specified, the value of  $\theta_r$  can only be taken in a finite set of real numbers. Then, the brute-force searching method can be used to solve the problem.

The pipeline of solving the roll angle is shown in Figure 2. Firstly, using CNN and a series of post-processing processes,  $N$  sorted lane lines  $\mathcal{L} = \{L_1, L_2, \dots, L_N\}$  are obtained. According to equation (9)(10), the image coordinate of VP ( $v_x, v_y$ ) is estimated. Based on the VP, the pitch and yaw angles ( $\theta_p, \theta_y$ ) of the camera can be calculated according to equation (8). Fixing  $\theta_p$  and  $\theta_y$ , the optimal roll angle  $\theta_r^*$  is determined by searching which minimizes the variance of lane widths in BEV. As for how to calculate the lane widths, we first project lanes  $\mathcal{L}$  from image space to ground space according to equation (13), and then calculate the distance between adjacent lanes and calculate the variance from equation (15).

This proposed method does not require scale information and only requires that there are at least two lanes in the image. The actual width of the lanes and the mounting height of the camera do not affect the solution of the roll angle.

#### 4.2 Roll Calibration with Prior Lane Width

When the widths of lanes are known, the optimization model can be obtained by simply modifying equation (16) as follows.

$$\theta_r^* = \underset{\theta_r}{\operatorname{argmin}} \sum_{i=1}^{N-1} (d_i(\theta_r) - d_i)^2 \quad (17)$$

where  $d_i$  = real width of the  $i$  th lane

Compared with the method proposed in Section 4.1, this method can even work in the scenes with only one lane and in the roads whose lanes’ widths are not equal, if the widths of lanes and the height of the camera are known.

### 5. EXPERIMENT

We quantitatively evaluated our algorithm on the nuScenes dataset (Caesar, H., 2020) which contains a large amount of synchronized camera and pose data. The onboard front camera collects images in urban scenarios including structured roads. The pose data is obtained from an accurate localization system that takes into account IMU, GPS, and HD LiDAR maps. We take the pose data as ground truth. The proposed method estimates the absolute orientation in road coordinate system. However, the orientation provided by the dataset is relative to the global coordinate system. As a result, the yaw angle is different in the most of time, and the pitch and roll angle will differ when driving along a slope or inclination. Therefore, we take the error

of relative orientation (Geiger, A., 2012) as the metric to describe the accuracy of the calibration algorithm.

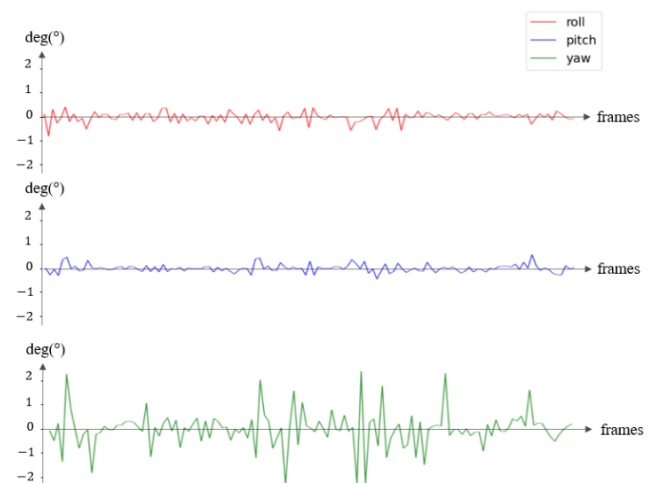


Figure 7. Rotation errors

The rotation errors are shown in Figure 7 and their mean values are shown in Table 1. The error of the yaw angle is relatively larger than the other two, which is influenced by the fact that the horizontal location of the vanishing point is relatively sensitive to the lane centerlines extraction error. Also, the non-zero curvature of the lane line will introduce errors to VP estimation.

Rotation Error	Roll (°)	Pitch (°)	Yaw (°)
Mean	0.154	0.116	0.568

Table 1. Rotation errors analysis

We also tested the proposed method on “Tuyou”, an autonomous vehicle that equips cameras mounted on the roof of the car (see Figure 8.). The focal length of the camera we used is 3.6 mm and the size of the image is 1920 x 1080. We tested our algorithms on real urban road scenarios, including main roads, highways, and intersections.

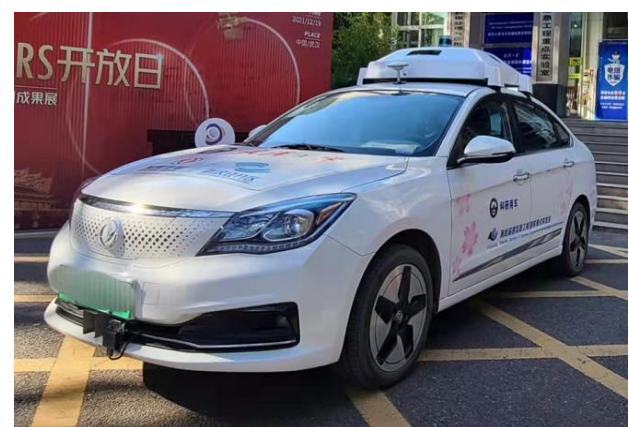
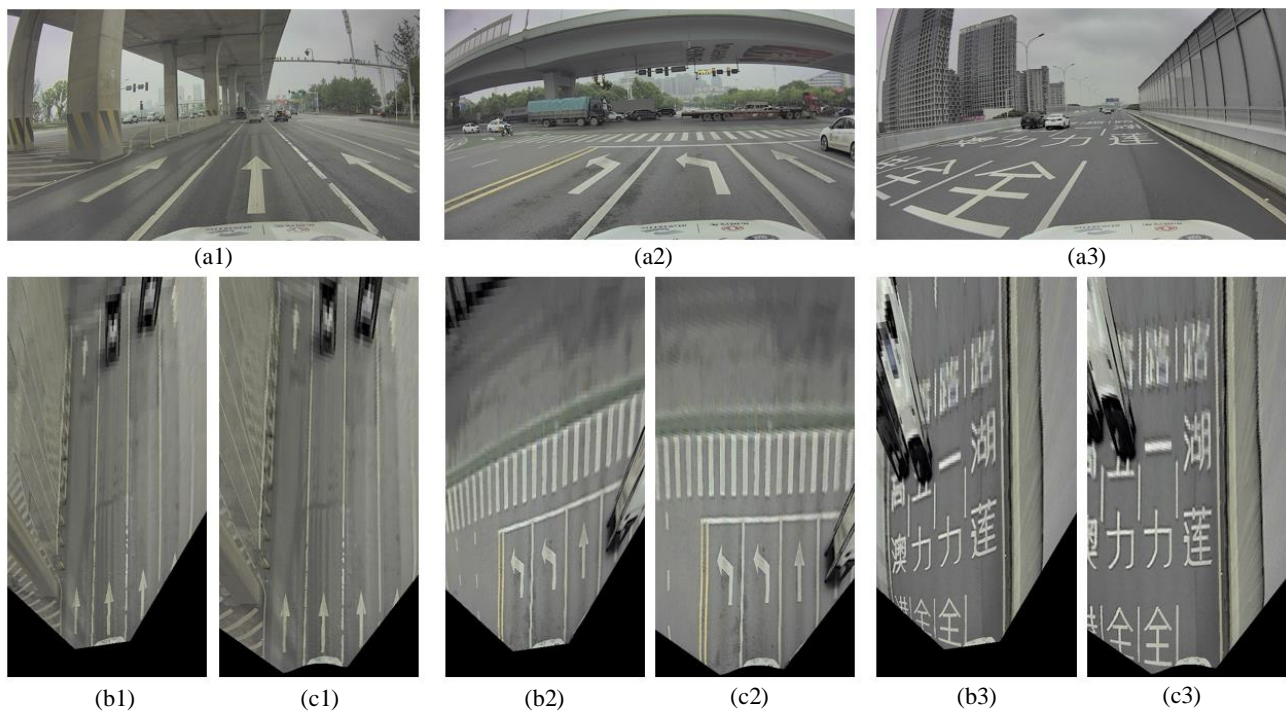


Figure 8. “Tuyou” autonomous vehicle

It’s difficult to obtain the ground truth of the camera’s orientation, especially in outdoor scenes without special targets. We use the proposed roll-sensitive IPM to reconstruct the ground and



**Figure 9.** BEV generated by different methods. (a) Perspective view. (b) BEV generated by the method proposed by Yang et al. (c) BEV generated by our method.

compare the result with other online calibration methods with the zero-roll hypothesis. Applying IPM, the bird's-eye view can be easily generated. Figure 9(b) shows BEVs using the camera parameters calibrated by Yang's method (Yang et al. 2016) which assumes the roll angle is zero. Figure 9(c) demonstrates the result using the parameters calculated by our proposed method. Because of the installation error, our onboard camera did not strictly mount horizontally and the roll angle was about  $2^\circ$ . As shown in the figure, due to the inappropriate zero-roll assumption, the BEV generated by Yang's method is deformed systematically. On the contrary, by calibrating the roll angle, our method successfully reconstructs the ground space. It indicates that we can use the proposed method to rectify the installation error.

The total running time of the proposed algorithm is about 75ms where lane detection and corresponding post-processing pipeline take 40ms, camera calibration takes about 35ms. In camera calibration, the calculation of the roll angle takes up most of the running time, about 34.5ms. Here we choose  $\mu = 0$  and  $\sigma = 1^\circ$ . And we set the resolution of roll angle when searching is  $0.1^\circ$ . The rapid running speed allows the proposed algorithm to run in real time onboard.

## 6. CONCLUSION

In this paper, we proposed a roll-sensitive online camera calibration method without the zero-roll hypothesis. Previous studies assume that the roll angle is zero, and calibrate the pitch and yaw angles by a longitudinal vanishing point. With lane width constrain, our proposed method can calculate the three Euler angles of the camera based on only one VP in the image. This method can be used to rectify the camera's installation angular errors and determine the orientation of the camera online.

## ACKNOWLEDGEMENTS

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grants 42101448 and the Key Research and Development Projects in Hubei Province under Grants 2021BLB149.

## REFERENCES

- Aly, M., 2008. Real time detection of lane markers in urban streets. In *2008 IEEE Intelligent Vehicles Symposium* (pp. 7-12). IEEE.
- Barnard, S. T., 1983. Interpreting perspective images. *Artificial intelligence*, 21(4), 435-462.
- Bertozzi, M., Broggi, A, 1998. GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection. *IEEE transactions on image processing*, 7(1), 62-81.
- Caesar, H. , V Bankiti, Lang, A. H. , Vora, S. , & Beijbom, O. , 2020. nuScenes: A Multimodal Dataset for Autonomous Driving. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.
- Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A. L., 2017. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4), 834-848.
- Collins, R. T., Weiss, R. S., 1990. Vanishing point calculation as a statistical inference on the unit sphere. In *ICCV* (Vol. 90, pp. 400-403).
- Geiger, A. , Lenz, P., Urtasun, R. , 2012. Are we ready for Autonomous Driving? The KITTI Vision Benchmark

- Suite. *2012 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.
- Hartley, R., & Zisserman, A, 2003. *Multiple view geometry in computer vision*. Cambridge university press.
- Jeong, J., Kim, A, 2016. Adaptive inverse perspective mapping for lane map generation with SLAM. In *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)* (pp. 38-41). IEEE.
- Lee, B. , Zhou, J. , Ye, M. , Guo, Y., 2016. A novel approach to camera calibration method for smart phones under road environment. *ISPRS International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLI-B5*, 49-54.
- Lee, J. K., Yoon, K. J., 2019. Joint estimation of camera orientation and vanishing points from an image sequence in a non-manhattan world. *International Journal of Computer Vision*, 127(10), 1426-1442.
- Oliveira, M., Santos, V., Sappa, A. D., 2015. Multimodal inverse perspective mapping. *Information Fusion*, 24, 108-121.
- Orghidan, R., Salvi, J., Gordan, M., Orza, B, 2012. Camera calibration using two or three vanishing points. In *2012 Federated Conference on Computer science and information systems (FedCSIS)* (pp. 123-130). IEEE.
- Qin, Z., Wang, H., & Li, X., 2020. Ultra fast structure-aware deep lane detection. In *European Conference on Computer Vision* (pp. 276-291). Springer, Cham.
- Rother, C., 2002. A new approach to vanishing point detection in architectural environments. *Image and Vision Computing*, 20(9-10), 647-655.
- Suzuki, S., 1985. Topological structural analysis of digitized binary images by border following. *Computer vision, graphics, and image processing*, 30(1), 32-46.
- Tsai, R. Y., Lenz, R. K, 1989. A new technique for fully autonomous and efficient 3 d robotics hand/eye calibration. *IEEE Transactions on robotics and automation*, 5(3), 345-358.
- Wang, X., Chen, H., Li, Y., Huang, H, 2019. Online extrinsic parameter calibration for robotic camera–encoder system. *IEEE Transactions on Industrial Informatics*, 15(8), 4646-4655.
- Wang, Z., Dong, M., Qin, Y., Du, Y., Zhao, F., Gu, L, 2017. Suspension system state estimation using adaptive Kalman filtering based on road classification. *Vehicle System Dynamics*, 55(3), 371-398.
- Xiao, J. , Hong, T. , Zhang, Y. , Zhou, Y. , & Lei, J. . 2018. Blind video denoising via texture-aware noise estimation. *Computer Vision and Image Understanding*, 169(apr.), 1-13.
- Yang, W., Fang, B., Tang, Y. Y, 2016. Fast and accurate vanishing point detection and its application in inverse perspective mapping of structured road. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(5), 755-766.
- Zhao, K., Iurgel, U., Meuter, M., Pauli, J, 2014. An automatic online camera calibration system for vehicular applications. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (pp. 1490-1492). IEEE.
- Zou, Q., Jiang, H., Dai, Q., Yue, Y., Chen, L., Wang, Q., 2019. Robust lane detection from continuous driving scenes using deep neural networks. *IEEE transactions on vehicular technology*, 69(1), 41-54.