# Practical Auto-calibration for Spatial Scene-understanding from Crowdsourced Dashcamera Videos

Hemang Chawla, Matti Jukola, Shabbir Marzban, Elahe Arani and Bahram Zonooz
*Advanced Research Lab, Navinfo Europe, The Netherlands*

Keywords: Vision for Robotics, Crowdsourced Videos, Auto-calibration, Depth Estimation, Ego-motion Estimation.

Abstract: Spatial scene-understanding, including dense depth and ego-motion estimation, is an important problem in computer vision for autonomous vehicles and advanced driver assistance systems. Thus, it is beneficial to design perception modules that can utilize crowdsourced videos collected from arbitrary vehicular onboard or dashboard cameras. However, the intrinsic parameters corresponding to such cameras are often unknown or change over time. Typical manual calibration approaches require objects such as a chessboard or additional scene-specific information. On the other hand, automatic camera calibration does not have such requirements. Yet, the automatic calibration of dashboard cameras is challenging as forward and planar navigation results in critical motion sequences with reconstruction ambiguities. Structure reconstruction of complete visual-sequences that may contain tens of thousands of images is also computationally untenable. Here, we propose a system for practical monocular onboard camera auto-calibration from crowdsourced videos. We show the effectiveness of our proposed system on the KITTI raw, Oxford RobotCar, and the crowdsourced D$^2$-City datasets in varying conditions. Finally, we demonstrate its application for accurate monocular dense depth and ego-motion estimation on uncalibrated videos.

## 1 INTRODUCTION

Autonomous driving systems have progressed over the years with advances in visual perception technology that enables safer driving. These advances in computer vision have been possible with the enormous amount of visual data being captured for training neural networks applied to a variety of scene-understanding tasks such as dense depth and ego-motion estimation. Nevertheless, acquiring and annotating vehicular onboard sensor data is a costly process. One of the ways to design generic perception systems for spatial scene-understanding is to utilize crowdsourced data. Unlike most available datasets which contain limited hours of visual information, exploiting large scale crowdsourced data offers a promising alternative (Dabeer et al., 2017; Gordon et al., 2019)

Crowdsourced data is typically collected from low-cost setups such as monocular dashboard cameras. However, the robustness of modern multi-view perception systems used in dense depth estimation (Godard et al., 2019), visual odometry (Mur-Artal et al., 2015), lane detection (Ying and Li, 2016), object-specific distance estimation (Zhu and Fang, 2019), optical flow computation (Meister et al., 2018), and so on, depends upon the accuracy of their camera intrinsics. The lack of known camera intrinsics for crowdsourced data captured from unconstrained environments prohibits the direct application of existing approaches. Therefore, it is pertinent to estimate these parameters, namely focal lengths, optical center, and distortion coefficients automatically and accurately. Yet, standard approaches to obtaining these parameters necessitate the use of calibration equipment such as a chessboard, or are dependent upon the presence of specific scene geometry such as planes or lines (Wildenauer and Micusik, 2013). A variety of approaches to automatically extract the camera intrinsics from a collection of images have also been proposed. Multi-view geometry based methods utilize epipolar constraints through feature extraction and matching for auto-calibration (Gherardi and Fusiello, 2010; Kukelova et al., 2015). However, for the typical driving scenario with constant but unknown intrinsics, forward and planar camera motion results in critical sequences (Steger, 2012; Wu, 2014). Supervised deep learning methods instead require images with known ground truth (GT) parameters for training (Lopez et al., 2019; Zhuang

Chawla, H., Jukola, M., Marzban, S., Arani, E. and Zonooz, B.
Practical Auto-calibration for Spatial Scene-understanding from Crowdsourced Dashcamera Videos.
DOI: 10.5220/0010255808690880
In *Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2021) - Volume 5: VISAPP*, pages 869-880
ISBN: 978-989-758-488-6

et al., 2019). While Structure-from-Motion has also been used for auto-calibration, its direct application to long crowdsourced onboard videos is computationally expensive (Schonberger and Frahm, 2016), and hence unscalable. This motivates the need for a practical auto-calibration method for spatial scene-understanding from unknown dashcameras.

Recently, camera auto-calibration through Structure-from-Motion (SfM) on sub-sequences of turns from KITTI raw dataset (Geiger et al., 2012) was proposed for 3D positioning of traffic signs (Chawla et al., 2020a). However, the method was limited by the need for Global Positioning System (GPS) information corresponding to each image in the dataset. The GPS information may not always be available, or may be collected at a different frequency than the images, and may be noisy. Furthermore, no analysis was performed on the kind of sub-sequences necessary for successful and accurate calibration. Typical visibility of ego-vehicle in the onboard images also poses a problem to the direct application of SfM. Therefore, scalable accurate auto-calibration from onboard visual sequences remains a challenging problem.

In this paper, we present a practical method for extracting camera parameters including focal lengths, principal point, and radial distortion (barrel and pincushion) coefficients from a sequence of images collected using only an onboard monocular camera. Our contributions are as follows:

- We analytically demonstrate that the sub-sequences of images where the vehicle is turning provide the relevant structure necessary for a successful auto-calibration.
- We introduce an approach to automatically determine these turns using the images from the sequence themselves.
- We empirically study the relation of the frames per second (fps) and number of turns in a video sequence to the calibration performance, showing that a total $\approx 30\,\mathrm{s}$ of sub-sequences are sufficient for calibration.
- A semantic segmentation network is additionally used to deal with the variable shape and amount of visibility of ego-vehicle in the image sequences, improving the calibration accuracy.
- We validate our proposed system on the KITTI raw, the Oxford Robotcar (Maddern et al., 2017), and the D$^2$-City (Che et al., 2019) datasets against state-of-the-art.
- Finally, we demonstrate its application to chessboard-free dense depth and ego-motion estimation on the uncalibrated KITTI Eigen (Eigen and Fergus, 2015) and Odometry (Zhou et al., 2017) splits respectively.

## 2 RELATED WORK

Over the years, multiple ways have been devised to estimate camera parameters from a collection of images for which the camera hardware is either unknown or inaccessible. Methods using two or more views of the scene have been proposed to estimate camera focal lengths (Gherardi and Fusiello, 2010). The principal point is often fixed at the center of the image, as its calibration is an ill-posed problem (de Agapito et al., 1998). To estimate radial distortion coefficients, two-view epipolar geometry constraints are often used (Kukelova et al., 2015). Note that forward camera motion is a degenerate scenario for distortion estimation due to ambiguity against scene depth. Similarly, planar camera motion is also a critical calibration sequence (Wu, 2014). Nevertheless, typical automotive visual data captured from dashcameras majorly constitute forward motion on straight paths and a few turns, within a mostly planar motion. Supervised learning based methods for camera auto-calibration have also been introduced (Lopez et al., 2019; Zhuang et al., 2019). However, their applicability is constrained by the variety of images with different combinations of ground truth camera parameters used in training. On the other-hand self-supervised methods (Gordon et al., 2019) do not achieve similar performance (Chawla et al., 2020b). SfM has also been utilized to estimate camera parameters from a crowdsourced collection of images (Schonberger and Frahm, 2016). However, the reconstruction of a complete long driving sequence of tens of thousands of images is computationally expensive, motivating the utilization of a relevant subset of images (Chawla et al., 2020a). Therefore, this work proposes a practical system for intrinsics auto-calibration from onboard visual sequences. Our work can be integrated with the automatic extrinsic calibration of (Tummala et al., 2019) for obtaining the complete camera calibration.

## 3 SYSTEM DESIGN

This section describes the components of the proposed practical system for camera auto-calibration using a crowdsourced sequence of $n$ images captured from an onboard monocular camera. We represent the camera parameters using the pinhole camera model

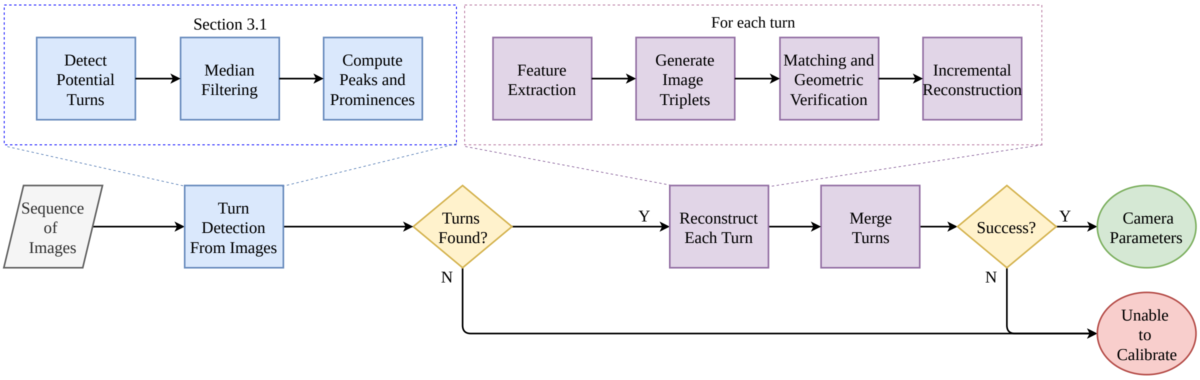

Figure 1: Practical Auto Calibration from on-board visual sequences. The input to the system is represented in gray. The components of turn detection step (Section 3.1) are shown in blue. The components of the turn reconstruction for parameter extraction (Section 3.2) are shown in purple.

and a polynomial radial distortion model with two parameters (Hartley and Zisserman, 2003). The intrinsic matrix is given by,

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{1}$$

where $f_x$ and $f_y$ are the focal lengths, and $(c_x, c_y)$ represents the principal point. The radial distortion is modeled using a polynomial with two parameters $k_1$ and $k_2$ such that,

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = (1 + k_1 r^2 + k_2 r^4) \begin{bmatrix} x_u \\ y_u \end{bmatrix}, \tag{2}$$

where $(x_d, y_d)$ and $(x_u, y_u)$ represent the distorted and rectified pixel coordinates respectively, while $r$ is the distance of the coordinate from the distortion center (assumed to be the same as the principal point). An overview of the proposed framework is shown in Figure 1.

Our system is composed of two broad modules. The first module is *turn detection* which outputs a list of $\varsigma$ sub-sequences corresponding to the turns in the video. The second module is *incremental reconstruction based auto-calibration* using these sub-sequences. This involves building the scene graph of image pairs with geometrically verified feature matches. The scene graph is then used to reconstruct each detected turn within a bundle adjustment framework, followed by a merging of the turns. This allows to extract the single set of camera parameters from the captured turn sequences.

Turns are necessary in extracting the camera parameters for two reasons:

1. As stated earlier, the pure translation and pure forward motion of the camera are degenerate scenarios for auto-calibration of the focal lengths and the distortion coefficients respectively (Steger, 2012; Wu, 2014).
2. Moreover, the error in estimating the focal lengths is bounded by the inverse of the rotation between the matched frames, as derived in Sec 3.2.

## 3.1 Turn Detection

Algorithm 1 summarizes the proposed method for turn detection. This method estimates the respective median images for the turns present in the full video. Thereafter for each median image (turn center), $k$ preceding and succeeding images are collated to form the turn sub-sequences.

Algorithm 1: Turn Detection.

```
input : a list of images $I_1 \dots I_n \in \mathrm{I}$
        max number of turns $\varsigma$
output: a list of turn_centers
turn_centers ← []
potential_turns ← []
turn_magnitudes ← []
potential_turns, turn_magnitudes ←
  computePotentialTurns(I)
turn_magnitudes ← medianFilter
  (turn_magnitudes)
peaks, prominences ←
  findPeaks(turn_magnitudes)
peaks.sort(prominences)
peaks.reverse()
peaks ← peaks[1 : min($\varsigma$, len(peaks))]
turn_centers ← potential_turns[peaks]
return turn_centers
```

To identify potential turns in the complete image sequence, we utilize a heuristic based on epipoles (Hartley and Zisserman, 2003). The image sequence is broken into triplets of consecutive images, and the epipoles are computed for the $C_2^3$ combinations of image pairs. The portions where the vehicle is undergoing pure translation is indicated by the failure to extract the epipoles. For the remaining

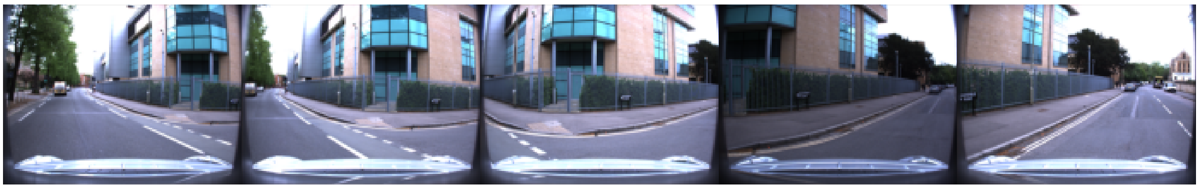

Figure 2: Sample images from a turn detected in the Oxford Robotcar dataset using Algorithm 1. The middle image corresponds to the detected turn median. For visualization, the remaining shown images are selected with a step size of 5 frames from the sub-sequence.

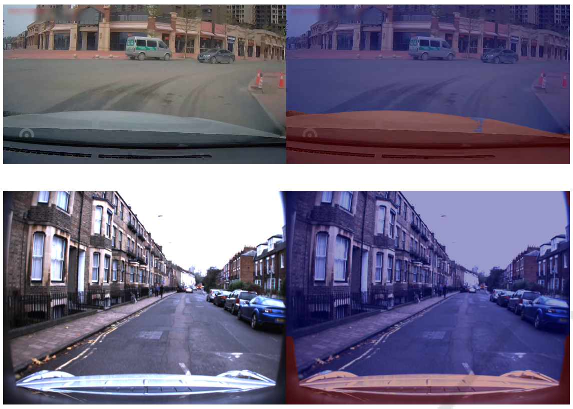

Figure 3: Masked ego-vehitcle using (Arani et al., 2021). Left: Input image. Right: Segmented image with masked ego-vehicle (red) and valid region (blue). Examples from D$^2$-City (top), and Oxford Robotcar (bottom) datasets. Note the varying shape, angles, and amount of ego-vehicle visibility. Only the valid region is used during feature extraction and matching for calibration.

portions, the average relative distance of the epipoles from the image centers is used as an indicator for the turning magnitude. We filter the estimated potential turn magnitudes across the sequence for noise, using a median filter with window size $2 \cdot \lfloor k/2 \rfloor - 1$.

Thereafter, we compute the peaks (local maximas) and the topographic prominences of the potential turn magnitudes. The prominences, sorted in a descending order are the proxy for the likelihood of each peak being a usable turn. Selecting the top $\varsigma$ turns based on the peak prominences, we construct the sub-sequences containing $2k+1$ images each. Figure 2 shows a set of sample images from a detected turn in the Oxford Robotcar dataset.

## 3.2 Calibration

Given the $\varsigma$ turn sub-sequences, the camera is calibrated within a SfM framework. Each turn is incrementally reconstructed, and the models for the turns are consequently merged to output the final camera parameters as shown in Figure 1. Utilizing multiple turns assists in increasing the reliability of the calibration and accounting for any false positive turn-detections that may not be successfully reconstructed.

Calibration through SfM involves building a scene graph for each turn. The images in the turn are the nodes of this graph, and the validated image pairs with their inlier feature correspondences are the edges. This requires extracting features from the images and matching them across the sequence, followed by a geometric verification of the matches.

**Ego-vehicle Masking.** One of the challenges in crowdsourced dashcamera sequences is the presence of the ego-vehicle in the images. This impacts the correspondence search negatively (Schonberger and Frahm, 2016). Upon sampling several crowdsourced sequences, we find that they have varying segments of car dashboard as well as A-pillars visible in the camera view, making a fixed crop of the images an infeasible solution. Therefore, we utilize a real-time semantic segmentation network (Arani et al., 2021) trained on Mapillary dataset (Neuhold et al., 2017) for masking out the ego-vehicle (see Figure 3) during feature extraction.

**Reconstruction.** After building the scene graph, we reconstruct each turn within an incremental SfM framework to output a set of camera poses $\mathcal{P} \in SE(3)$ and the scene points $\mathcal{X} \in \mathbb{R}^3$ (Schonberger and Frahm, 2016).

The Perspective-$n$-Point (P$n$P) problem is solved during turn reconstruction to simultaneously estimate the camera poses, scene structure, and the camera parameters, optimized within a bundle adjustment framework. This uses the correspondences between the 2D image features and the observed 3D world points for image registration.

Based on this, we show that the error in estimating the focal lengths is bounded by the inverse of the rotation between the matched frames. For simplicity of derivation, we keep the principal point fixed at the image center. The camera model maps the observed world points $P_w$ to the feature coordinates $p_c$ of the rectified image such that,

$$sp_c = KR_{cw}P_w + Kt_{cw}, \tag{3}$$

where the product of the camera intrinsic matrix $K$ (Eq. 1) and the rigid camera pose with rotation $R_{cw} \in SO(3)$ and translation $t_{cw} \in \mathbb{R}^3$ is the projection matrix, and $s$ denotes scale factor proportional to the projective depth of the features. The distortion model (Eq. 2) further relates the distorted feature coordinates with the rectified feature coordinates $p_c$. Accordingly, we can associate the feature matches of the observed 3D points in any two $(i, j)$ of the $m$ camera views correspondences through

$$sp_j = KRK^{-1}p_i + Kt. \tag{4}$$

In the scenario of pure translation without any rotation, Eq. 4 reduces to

$$sp_j = Kt, \tag{5}$$

showing that the correspondences cannot be utilized for calibration. Now consider the scenario where there is no translation. The feature/pixel shift is then only determined by the amount of rotation,

$$p_j = \frac{KRK^{-1}p_i}{(KRK^{-1}p_i)_3}, \tag{6}$$

where the subscript 3 represents the third component of the $3 \times 1$ homogeneous feature coordinates vector. Having correspondences across overlapping frames allows for the assumption that relative rotation is small. Hence, we can write

$$R = I + r, \tag{7}$$

$$\text{where } r = \begin{bmatrix} 0 & r_z & -r_y \\ -r_z & 0 & r_x \\ r_y & -r_x & 0 \end{bmatrix}. \tag{8}$$

To be able to finally derive an analytic expression, we expand Eq. 6 for $r$ using Taylor series to obtain,

$$p_j = p_i + (KrK^{-1}p_i) - p_i(KrK^{-1}p_i)_3, \tag{9}$$

where the subscript 3 represents the third component of the $3 \times 1$ homogeneous feature coordinates vector. Since vehicular motion primarily consists of the car turning about the $y$ axis, we only consider the yaw $r_y$ for this derivation. Substituting the camera matrix from Eq. 1 and the rotation matrix from Eq. 7 in Eq. 9, we obtain

$$p_{j(x)} = -f_x r_y - r_y \frac{(p_{i(x)} - c_x)^2}{f_x}, \tag{10}$$

$$p_{j(y)} = -r_y \frac{(p_{i(x)} - c_x)(p_{i(y)} - c_y)}{f_x}. \tag{11}$$

Similar equations can also be written for the estimated camera parameters and the reprojected feature coordinates. Bundle adjustment minimizes the re-projection error,

$$\varepsilon = \sum_j \rho_j \left( \|\hat{p}_j - p_j\|_2^2 \right) \tag{12}$$

$$= \sum_j \rho_j \left( |\delta p_{j(x)}|^2 + |\delta p_{j(y)}|^2 \right), \tag{13}$$

where $\rho_j$ down-weights the outlier matches, and $\hat{p}_j = p_j + \delta p_j$, are the reprojected feature coordinates. Thus, the parameters to be estimated, focal lengths $\hat{f}_x, \hat{f}_y$, camera rotation $\hat{r}_y$, and distortion coefficients $k_1$, $k_2$ appear implicitly in the error function above.

Since the camera rotation and intrinsics are optimized simultaneously to minimize the re-projection error, we can assume that the estimated $\hat{r}_y$ balances the estimated camera parameters. For simplicity, we choose a yaw that at least the features close to the principal point remain unchanged, where the impact of distortion is also negligible. Therefore, from Eqs. 10 and 11, we understand

$$\hat{r}_y \hat{f}_x = r_y f_x. \tag{14}$$

Now we can write the equations for the estimated feature point coordinates replacing $\hat{r}_y$ to obtain,

$$\hat{p}_{j(x)} = -f_x r_y - f_x r_y \frac{(p_{i(x)} - c_x)^2}{\hat{f}_x^2}, \tag{15}$$

$$\hat{p}_{j(y)} = -f_x r_y \frac{(p_{i(x)} - c_x)(p_{i(y)} - c_y)}{\hat{f}_x^2}. \tag{16}$$

Accordingly, by subtracting Eqs. 10 and 11 from Eqs. 15 and 16 respectively, we get the first order approximation of $\delta p_j$,

$$\delta p_{j(x)} = 2\delta f_x r_y \frac{(p_{i(x)} - c_x)^2}{f_x^2}, \tag{17}$$

$$\delta p_{j(y)} = 2\delta f_x r_y \frac{(p_{i(x)} - c_x)(p_{i(y)} - c_y)}{f_x^2}. \tag{18}$$

For the errors $|\delta p_j(x)| \ll 1$ and $|\delta p_j(y)| \ll 1$, we need to simultaneously satisfy

$$|\delta f_x| \ll \frac{f_x^2}{2r_y(p_{i(x)} - c_x)^2} \text{ and,} \tag{19}$$

$$|\delta f_x| \ll \frac{f_x^2}{2r_y|p_{i(x)} - c_x||p_{i(y)} - c_y|} \tag{20}$$

with the features closer to the boundary providing the tightest bound on $|\delta f_x|$. Since $c_x = w/2$ and $c_y = h/2$, we obtain,

$$\begin{bmatrix} \delta f_x & \delta f_y \end{bmatrix}^\intercal < \frac{2}{\max(h, w)} \begin{bmatrix} \frac{f_x^2}{w \cdot r_y} & \frac{f_y^2}{h \cdot r_x} \end{bmatrix}^\intercal. \tag{21}$$

This result is similar to the Eq. 3 obtained in (Gordon et al., 2019). Thus the errors in focal lengths $f_x$ and

$f_y$ are bounded by the inverse of rotation about the $y$ and $x$ axes, respectively.

Since the vehicular motion is planar with limited pitch rotations, we perform the calibration in two steps:

1. As modern cameras are often equipped with nearly square pixels, we fix $f_x = f_y$ and utilize the yaw rotations about the $y$ axis to calibrate the focal lengths. The principal point is also fixed to the center of the image. This step outputs a common focal length and the refined distortion coefficients.
2. Thereafter, we relax these assumptions and refine all the parameters simultaneously within bounds. This allows to slightly update $f_y$ and also refine the principal point to minimize the reprojection error.

The final output is a set of six parameters namely, focal lengths $f_x$, and $f_y$, principal point ($c_x$, $c_y$), and the radial distortion coefficients $k_1$ and $k_2$. Thereafter, the reconstructed models of the turns are merged. Thus, instead of optimizing the intrinsics as separate parameters for each turn, they are estimated as a single set of constants. In the few scenarios where overlapping turn sub-sequences may be present, they are also spatially merged and optimized together.

# 4 EXPERIMENTS

We validate our proposed system for auto-calibration of onboard cameras on the KITTI raw (Geiger et al., 2012), the Oxford Robotcar (Maddern et al., 2017), and the D$^2$-city (Che et al., 2019) datasets. Ego-vehicle is visible in the onboard camera of Oxford Robotcar and the dashcamera of D$^2$-City datasets, respectively. Corresponding GPS information (used in competing methods) is not availbale for the crowdsourced D$^2$-City dataset.

We compare our system against (Chawla et al., 2020a) which uses GPS based turn detection and SfM for calibration, as well as (Santana-Cedrés et al., 2017) which relies upon detecting lines, curves and Hough transform in the scene. We also demonstrate the necessity of masking the ego-vehicle for accurate calibration. To further evaluate our proposed system, we empirically study the impact of the number of turns used, as well as the frame rate of the videos on the calibration performance. We show that our system is superior to (Santana-Cedrés et al., 2017) as well as (Chawla et al., 2020a; Chawla et al., 2020b), which in turn outperforms the self-supervised (Gordon et al., 2019). Finally, we demonstrate the application of our system for chessboard-free accurate monocular dense depth and ego-motion estimation on uncalibrated videos.

## 4.1 Datasets

**KITTI Raw.** This dataset contains sequences from the urban environment in Germany with a right-hand drive. The images are captured at 10 fps with the corresponding GPS at 100 Hz. The ground truth (GT) camera focal lengths, principal point, and the radial distortion coefficients and model are available for the different days the data was captured. The GT camera parameters corresponding to Seq 00 to 02 are $\{f_x = 960.115\,\text{px}, f_y = 954.891\,\text{px}, c_x = 694.792\,\text{px}, c_y = 240.355\,\text{px}, k_1 = -0.363, k_2 = 0.151\}$, and for Seq 04 to 10 are $\{f_x = 959.198\text{px}, f_y = 952.932\,\text{px}, c_x = 694.438\,\text{px}, c_y = 241.679\,\text{px}, k_1 = -0.369, k_2 = 0.158\}$. Seq 03 is not present in the raw dataset. Also, the ego-vehicle is not visible in the captured data.

**Oxford Robotcar.** This dataset contains sequences from the urban environment in the United Kingdom with a left-hand drive. The images are captured at 16 fps, and the GPS at 16 Hz. However, some of the sequences have poor GPS or even do not have corresponding GPS available. A single set of GT camera focal lengths and the principal point is available for all the recordings, $\{f_x = 964.829\,\text{px}, f_y = 964.829\,\text{px}, c_x = 643.788\,\text{px}, c_y = 484.408\,\text{px}\}$. Instead of the camera distortion model and coefficients, a look-up table (LUT) is available that contains the mapping between the rectified and distorted images. The ego-vehicle is also visible in the captured data.

**D$^2$-City.** This dataset contains crowdsourced sequences collected from dashboard cameras onboard DiDi taxis in China. Therefore, the ego-vehicle is visible in the captured data. Different sequences have different amount and shape of this dashboard and A-pillar visibility. There is no accompanying GPS. The images are collected at 25 fps across varying road and traffic conditions. Consequently, no GT camera parameters are available as well.

## 4.2 Performance Evaluation

Table 1 summarizes the results of performance evaluation of our proposed onboard monocular camera auto-calibration system. We evaluate on ten KITTI raw sequences 00 to 10 (except sequence 03, which is missing in the dataset), and report the average calibration performance. Furthermore, we evaluate on three sequences from the Ox-

Table 1: Comparing calibration performance measured through median absolute percentage error and mean SSIM across datasets. M represents ego-vehicle masking. The best estimates for each parameter on the datasets are highlighted in gray.

| Dataset | Method | $\downarrow \mathbf{f_x}$ | $\downarrow \mathbf{f_y}$ | $\downarrow \mathbf{c_x}$ | $\downarrow \mathbf{c_y}$ | $\downarrow \mathbf{k_1}$ | $\downarrow \mathbf{k_2}$ | $\uparrow \mathbf{SSIM}_\mu$ |
|---|---|---|---|---|---|---|---|---|
| **KITTI Raw** | (Chawla et al., 2020a) | 1.735 | 0.967 | 0.884 | 0.051 | 11.435 | 34.488 | - |
| | Ours (w/o M) | 1.670 | 0.525 | 0.717 | 0.522 | 11.421 | 34.034 | - |
| **Oxford (Good GPS)** | (Chawla et al., 2020a) | 7.065 | 6.018 | 0.715 | 0.151 | - | - | 0.889 |
| | Ours (w/o M) | 8.068 | 6.394 | 1.152 | 0.433 | - | - | 0.890 |
| | Ours (with M) | 7.598 | 6.286 | 0.587 | 0.426 | - | - | 0.893 |
| **Oxford (Poor GPS)** | (Chawla et al., 2020a) | 6.763 | 6.156 | 0.418 | 0.017 | - | - | 0.908 |
| | Ours (w/o M) | 1.975 | 0.703 | 0.513 | 3.010 | - | - | 0.906 |
| | Ours (with M) | 1.892 | 0.361 | 0.386 | 2.516 | - | - | 0.909 |
| **Oxford (No GPS)** | Ours (w/o M) | 13.937 | 18.956 | 0.857 | 1.639 | - | - | 0.897 |
| | Ours (with M) | 6.610 | 6.408 | 1.207 | 0.549 | - | - | 0.899 |

ford Robotcar dataset with different GPS qualities, `2014-11-28-12-07-13` with good GPS measurements, `2015-03-13-14-17-00` with poor GPS measurements, and `2015-05-08-10-33-09` without any accompanying GPS measurements. We repeat each calibration 5 times are report the median absolute percentage error for the estimated focal lengths, principal point, and the distortion coefficients. Since the GT distortion model and coefficients are not provided in the Oxford robotcar dataset, we instead measure the mean structural similarity (Wang et al., 2004) between the images rectified using the provided Look-up-table (LUT) and our estimated parameters. Finally, we calibrate 10 turn-containing sequences of 30 s each, from the D$^2$-City dataset.

We successfully calibrate all sequences except for KITTI 04 which does not contain any turns. Our method performs better than (Chawla et al., 2020a) in all the cases except for Oxford (Good GPS). This case can be attributed to better turn detection with good GPS quality used by that method, which is often not available for crowdsourced data. Also, note that the calibration is better when using the proposed ego-vehicle masking. This effect is most pronounced for Oxford (No GPS) where the focal length errors become nearly half when masking out the ego-vehicle. The calibration without ego-vehicle masking is less effective with the presence of ego-vehicle in the images, as it acts as a watermark and negatively impacts feature matching (Schonberger and Frahm, 2016), even with the use of RAndom SAmple Consensus (RANSAC) (Hartley and Zisserman, 2003).

Following the comparison protocol of (Santana-Cedrés et al., 2017)[1], Figure 4 shows some qualitative auto-calibration results comparing our system against (Chawla et al., 2020a; Santana-Cedrés et al., 2017). Note that our system performs better, as visibly demonstrated by the rectified structures. Moreover, our method is applicable even when no GPS information is available, as we successfully calibrate all the selected sequences from D$^2$-City. However, our system relies upon the features in the images and thus is more suitable for urban driving sequences. Additional qualitative results for the dashcamera videos from D$^2$-city can be found in the Appendix.

## 4.3 Design Evaluation

Here, we further evaluate our proposed system design for the impact of the number of turns used during calibration, and the frame rate of the onboard image sequence, on the calibration performance. We carry out these analyses on KITTI sequence 00. For all these experiments the value of $k$ is set to 30.

For each experiment repeated multiple times, we report the *calibration error metric* as the median of absolute percentage error normalized by the number of times the calibration was successful. This unitless metric is used to capture the accuracy as well as the consistency of the auto-calibration across the aforementioned settings.

**Number of Turns.** Here, we study the impact of the number of turns on the calibration error metric. For this analysis, we first create a set $S_{turns}$ of the top 15 turns (at 10 fps) extracted using Algorithm 1. Thereafter, we vary the number of turns $j$ used for calibration by randomly selecting them from $S_{turns}$. We repeat each experiment 10 times and report the calibration error metric (see Figure 6). Note that the focal lengths estimation improves up to two turns. The principal point estimation improves up to four turns. The distortion coefficients estimation improves up to five turns. Therefore, successful auto-calibration with our approach requires a minimum of 5 turns, consisting of only a few hundred (≈300) images. Figure 5 shows the top 5 of the extracted turns for auto-calibration of KITTI Sequence 00. This reinforces the practicality and scalability of our system.

[1] http://dev.ipol.im/~asalgado/ipol_demo/workshop_perspective/

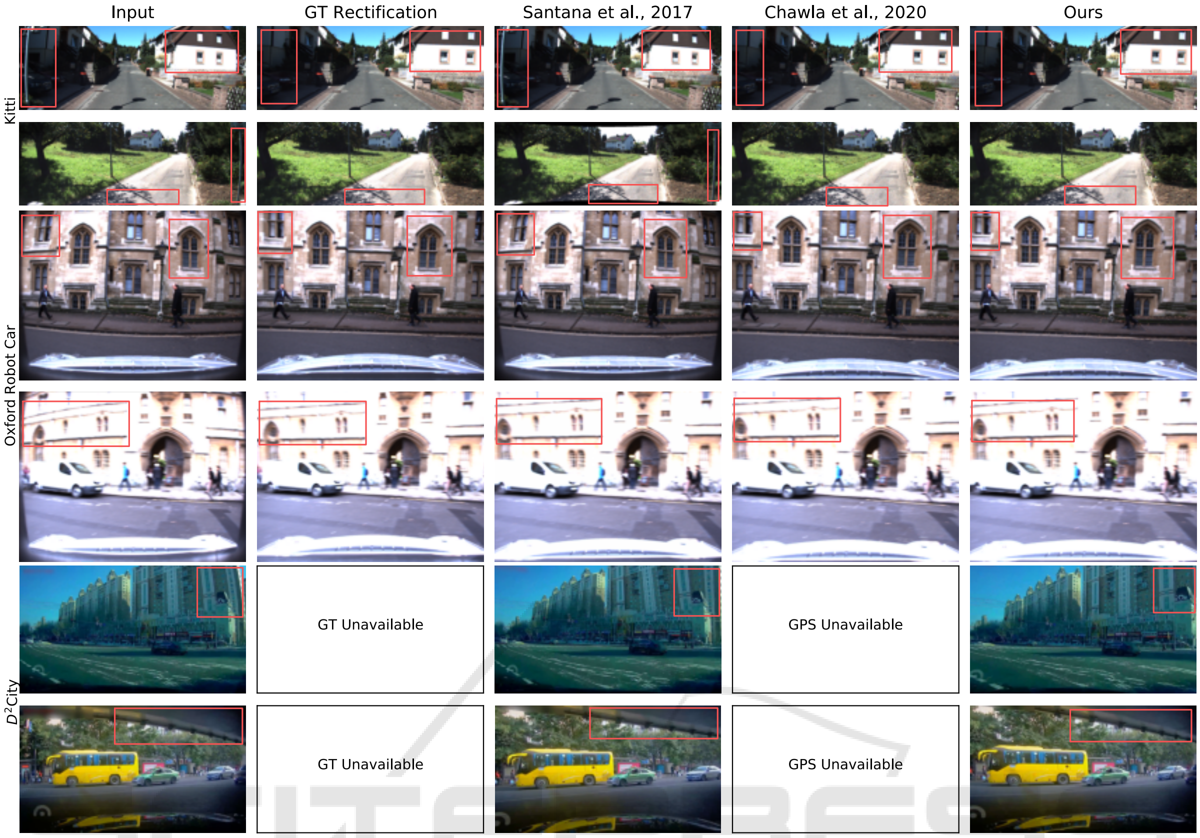

Figure 4: Qualitative comparison of camera auto-calibration on the KITTI raw (top), the Oxford Robotcar (middle), and the $D^2$-city (bottom) datasets.

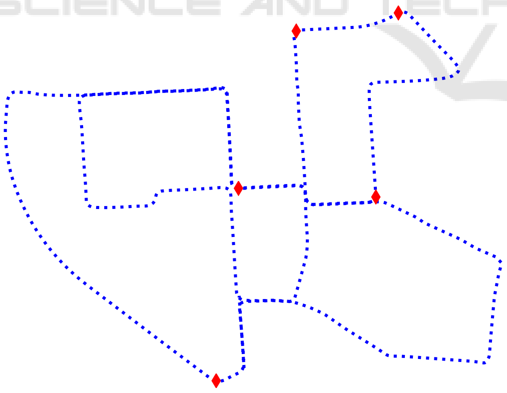

Figure 5: Using the top 5 turns for auto-calibration of KITTI Seq 00. The red diamonds denote the turn centers, and the dotted blue line is the corresponding GPS trajectory. Note that our method does not use this GPS trajectory for turn estimation.

**Frame Rate.** We study the impact of the frame rate by sampling the image sequence across a range of 1 fps to 10 fps, and report the calibration metric as before (see Figure 6). Note that calibration is unsuccessful when the frame rate is less than 3 fps. Thereafter, the calibration parameters improve up to 5 fps, thereby demonstrating the efficacy of our method even for low-cost onboard cameras.

## 4.4 Spatial Scene-understanding on Uncalibrated Sequences

We apply our auto-calibration system for training monocular dense depth and ego-motion estimation on the KITTI Eigen (Eigen and Fergus, 2015) and Odometry splits (Zhou et al., 2017) respectively, without prior knowledge of the camera parameters. Tables 2 and 3 compare depth and ego-motion estimation using Monodepth-2 (Godard et al., 2019) with different calibrations on the metrics defined in (Zhou et al., 2017), respectively. Note that depth and ego-motion estimation using our parameters is better than that using (Chawla et al., 2020a). Figure 7 provides some

Table 2: Chessboard-free monocular dense depth estimation with metrics defined in (Zhou et al., 2017). Better results are highlighted.

| **Calib** | **↓Abs Rel Diff** | **↓RMSE** | **↑$\delta < 1.25$** |
|---|---|---|---|
| **(Chawla et al., 2020a)** | 0.1142 | 4.8224 | 0.8783 |
| **Ours** | 0.1137 | 4.7895 | 0.8795 |

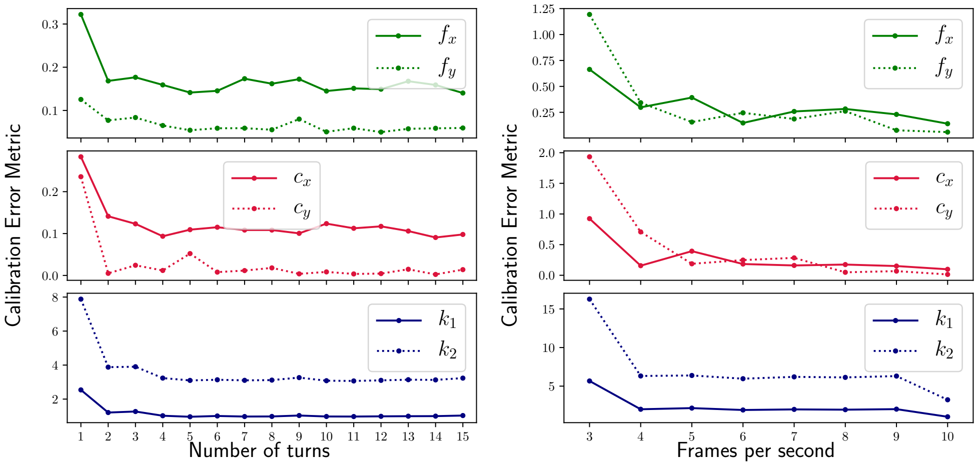

Figure 6: Impact of the number of turns used and the frame rate of captured image sequence on the calibration performance. The calibration error metric (unitless) measures the median of absolute percentage error normalized by the number of times the calibration was successful.

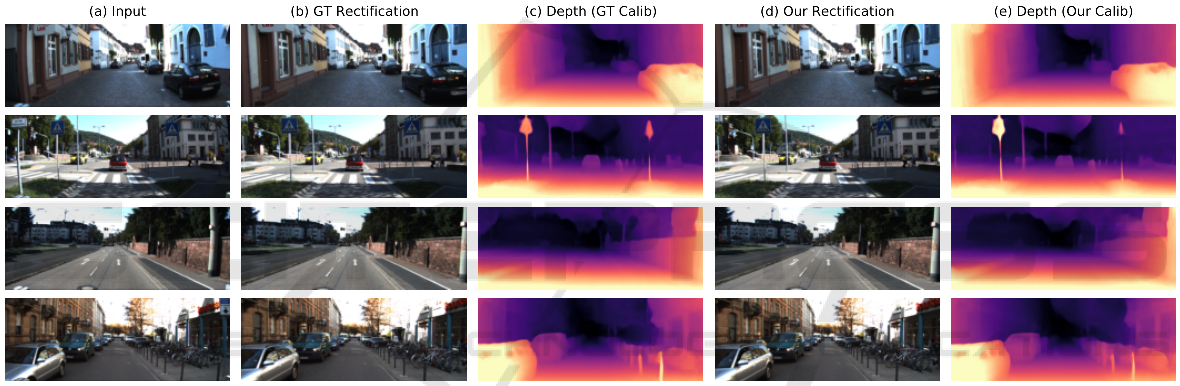
(a) Input (b) GT Rectification (c) Depth (GT Calib) (d) Our Rectification (e) Depth (Our Calib)

Figure 7: Comparison of monocular dense depth estimation when training Monodepth-2 using GT and our estimated camera parameters.

Table 3: Absolute Trajectory Error (ATE-5) (Zhou et al., 2017) for chessboard-free monocular ego-motion estimation on KITTI sequences 09 and 10. Better results are highlighted.

| Calib | Seq 09 | Seq 10 |
|---|---|---|
| **(Chawla et al., 2020a)** | 0.0323 ± 0.0103 | 0.0228 ± 0.0132 |
| **Ours** | 0.0299 ± 0.0109 | 0.0210 ± 0.0125 |

qualitative examples of depth maps estimated using our proposed system, which are comparable to those when using GT calibration.

## 5 CONCLUSIONS

In this work, we demonstrated spatial-scene understanding through practical monocular camera auto-calibration of crowdsourced onboard or dashcamera videos. Our system utilized the structure reconstruction of turns present in the image sequences for successfully calibrating the KITTI raw, Oxford robotcar, and the D$^2$-city datasets. We showed that our method can accurately extract these turn sub-sequences of total length ≈ 30 s from long videos themselves, without any assistance of corresponding GPS information. Moreover, our method is effective even on low fps videos for low-cost camera applications. Furthermore, the calibration performance was improved by automatically masking out the ego-vehicle in the images. Finally, we demonstrated chessboard-free monocular dense depth and ego-motion estimation for uncalibrated videos through our system. Thus, we contend that our system is suitable for utilizing crowdsourced data collected from low-cost setups to accelerate progress in autonomous vehicle perception at scale.

## REFERENCES

Arani, E., Marzban, S., Pata, A., and Zonooz, B. (2021). Rgpnet: A real-time general purpose semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 4

Chawla, H., Jukola, M., Arani, E., and Zonooz, B. (2020a). Monocular vision based crowdsourced 3d traffic sign positioning with unknown camera intrinsics and distortion coefficients. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. 2, 6, 7, 8, 9, 11

Chawla, H., Jukola, M., Brouns, T., Arani, E., and Zonooz, B. (2020b). Crowdsourced 3d mapping: A combined multi-view geometry and self-supervised learning approach. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2, 6

Che, Z., Li, G., Li, T., Jiang, B., Shi, X., Zhang, X., Lu, Y., Wu, G., Liu, Y., and Ye, J. (2019). D $^2$-city: A large-scale dashcam video dataset of diverse traffic scenarios. *arXiv preprint arXiv:1904.01975*. 2, 6

Dabeer, O., Ding, W., Gowaiker, R., Grzechnik, S. K., Lakshman, M. J., Lee, S., Reitmayr, G., Sharma, A., Somasundaram, K., Sukhavasi, R. T., et al. (2017). An end-to-end system for crowdsourced 3d maps for autonomous vehicles: The mapping component. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 634–641. 1

de Agapito, L., Hayman, E., and Reid, I. D. (1998). Self-calibration of a rotating camera with varying intrinsic parameters. In *British Machine Vision Conference (BMVC)*, pages 1–10. Citeseer. 2

Eigen, D. and Fergus, R. (2015). Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2650–2658. 2, 8

Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361. 2, 6

Gherardi, R. and Fusiello, A. (2010). Practical autocalibration. In *European Conference on Computer Vision (ECCV)*, pages 790–801. Springer Berlin Heidelberg. 1, 2

Godard, C., Mac Aodha, O., Firman, M., and Brostow, G. J. (2019). Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3828–3838. 1, 8

Gordon, A., Li, H., Jonschkowski, R., and Angelova, A. (2019). Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 8977–8986. 1, 2, 5, 6

Hartley, R. and Zisserman, A. (2003). *Multiple view geometry in computer vision*. Cambridge university press. 3, 7

Kukelova, Z., Heller, J., Bujnak, M., Fitzgibbon, A., and Pajdla, T. (2015). Efficient solution to the epipolar geometry for radially distorted cameras. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2309–2317. 1, 2

Lopez, M., Mari, R., Gargallo, P., Kuang, Y., Gonzalez-Jimenez, J., and Haro, G. (2019). Deep single image camera calibration with radial distortion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11817–11825. 2

Maddern, W., Pascoe, G., Linegar, C., and Newman, P. (2017). 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, 36(1):3–15. 2, 6

Meister, S., Hur, J., and Roth, S. (2018). Unflow: Unsupervised learning of optical flow with a bidirectional census loss. In *Thirty-Second AAAI Conference on Artificial Intelligence*. 1

Mur-Artal, R., Montiel, J. M. M., and Tardós, J. D. (2015). Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163. 1

Neuhold, G., Ollmann, T., Bulò, S. R., and Kontschieder, P. (2017). The mapillary vistas dataset for semantic understanding of street scenes. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5000–5009. 4

Santana-Cedrés, D., Gomez, L., Alemán-Flores, M., Salgado, A., Esclarín, J., Mazorra, L., and Alvarez, L. (2017). Automatic correction of perspective and optical distortions. *Computer Vision and Image Understanding*, 161:1–10. 6, 7, 11

Schonberger, J. L. and Frahm, J.-M. (2016). Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113. 2, 4, 7

Steger, C. (2012). Estimating the fundamental matrix under pure translation and radial distortion. *ISPRS journal of photogrammetry and remote sensing*, 74:202–217. 1, 3

Tummala, G. K., Das, T., Sinha, P., and Ramnath, R. (2019). Smartdashcam: automatic live calibration for dashcams. In *Proceedings of the 18th International Conference on Information Processing in Sensor Networks (IPSN)*, pages 157–168. 2

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612. 7

Wildenauer, H. and Micusik, B. (2013). Closed form solution for radial distortion estimation from a single vanishing point. In *British Machine Vision Conference (BMVC)*. 1

Wu, C. (2014). Critical configurations for radial distortion self-calibration. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 25–32. 1, 2, 3

Ying, Z. and Li, G. (2016). Robust lane marking detection using boundary-based inverse perspective mapping. In *2016 IEEE International Conference on Acoustics,* *Speech and Signal Processing (ICASSP)*, pages 1921–1925. IEEE. 1

Zhou, T., Brown, M., Snavely, N., and Lowe, D. G. (2017). Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1851–1858. 2, 8, 9

Zhu, J. and Fang, Y. (2019). Learning object-specific distance from a monocular image. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3839–3848. 1

Zhuang, B., Tran, Q.-H., Ji, P., Lee, G. H., Cheong, L. F., and Chandraker, M. K. (2019). Degeneracy in self-calibration revisited and a deep learning solution for uncalibrated slam. *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3766–3773. 2

# APPENDIX

We provide additional qualitative results on the calibration of the $D^2$-City dataset. Here, we don't show the results from (Chawla et al., 2020a) because it requires corresponding GPS headings that are missing for this dataset. Note that (Santana-Cedrés et al., 2017) is run with their default parameters. As shown in Fig. 8, sometimes (Santana-Cedrés et al., 2017) fails to completely undistort the structures, while our method performs consistently across a variety of real-world situations. Note that the average estimated focal length for D$^2$-City cameras is $\approx 1400$ px, different from the range of values for the KITTI and the Oxford Robotcar datasets.

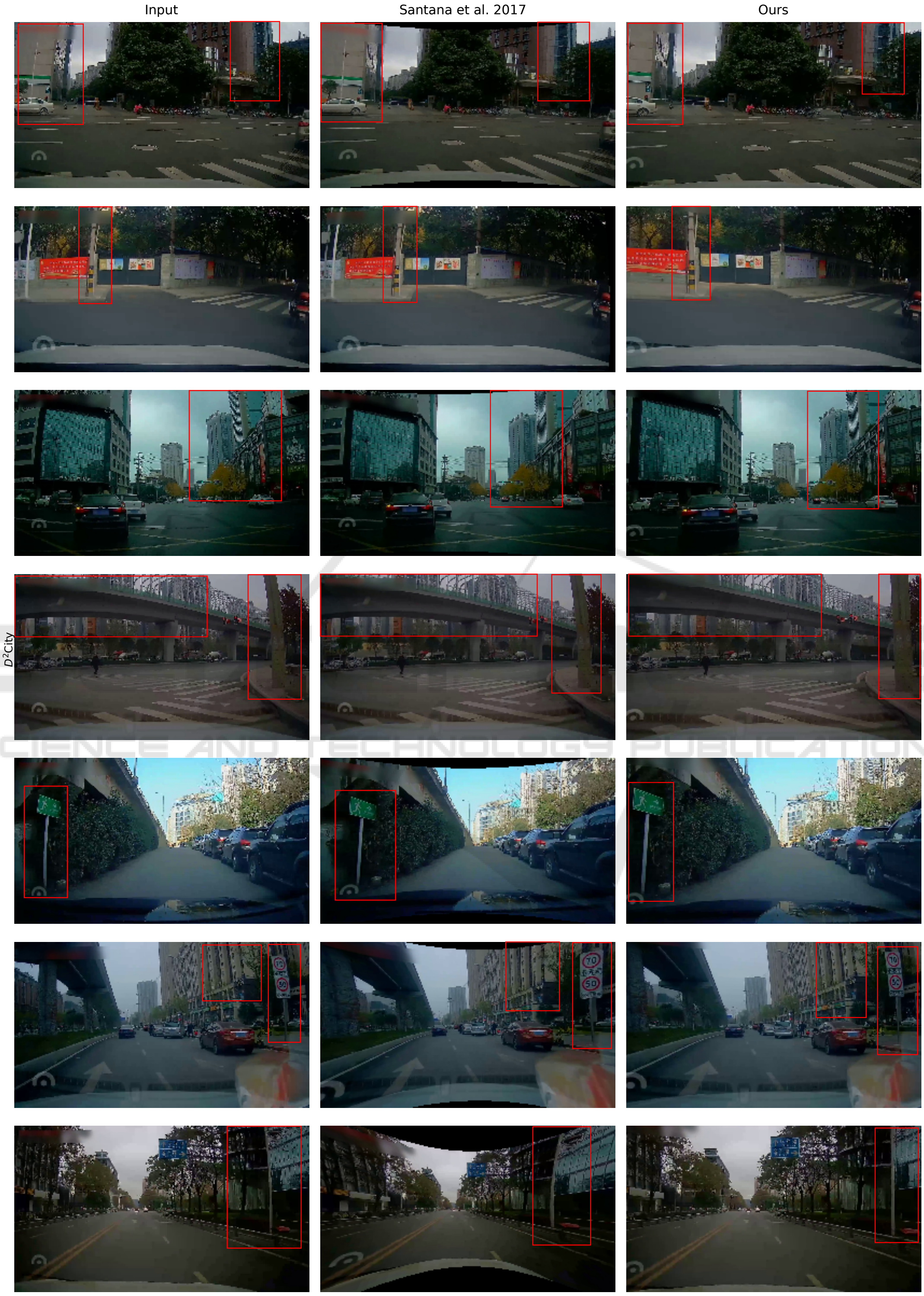

Figure 8: Additional qualitative comparison of camera auto-calibration on the $D^2$-city dataset.