International Journal of Computer Vision (2021) manuscript No.
(will be inserted by the editor)

ROECS: A Robust Pipeline Towards Online Extrinsics Correction of the Surround-view System

Tianjun Zhang[1] · Lin Zhang[1] · Ying Shen[1] · Shengjie Zhao[1] · Yicong Zhou[2]

Received: 14 January 2021 / Accepted: 14 January 2021

Abstract

As an indispensable component of advanced driving assistant systems (ADAS), surround-view systems (SVS) have been equipped by more and more vehicles. Generally, an SVS consists of four to six wide-angle fisheye cameras, which are typically mounted facing different directions on the vehicle. As long as both intrinsics and extrinsics of all cameras have been calibrated, a top-down surround-view with the real scale can be synthesized in real time from fisheye images captured by these cameras. However, on account of collisions, bumps and even the "hot expansion and cold contraction" effects, relative poses between cameras in the SVS may change from the initial calibrated states. In case extrinsics' representations are not adjusted accordingly, on the surround-view, obvious geometric misalignment will appear. Currently, the researches on correcting the extrinsics of the SVS in an online manner are quite sporadic, and a mature and robust pipeline is still lacking. As an attempt to fill this research gap to some extent, in this work, we present a novel extrinsics correction pipeline designed specially for the SVS, namely ROECS (Robust Online Extrinsics Correction of the Surround-view system). Specifically, a "bi-camera error" model, measuring the photometric discrepancy between two corresponding pixels on images captured by two adjacent cameras, is firstly designed. Then, by minimizing the overall "bi-camera error" with any nonlinear optimization scheme, the SVS's extrinsics can be iteratively optimized and become accurate eventually. Besides, an innovative three-step pixel selection strategy is also proposed, by which both the speed and the accuracy of our scheme can be enhanced significantly. Since ORB features are matched to assist the pixel selection, ROECS can be regarded as semi-direct. The superior robustness and the generalization capability of ROECS are validated by both quantitative and qualitative experimental results. To make the results reported in this paper reproducible, the collected data and the source code have been released at **********.

Keywords Surround-view system · Extrinsics correction · Sparse semi-direct method · Pixel selection strategy

Communicated by Lin Zhang

Tianjun Zhang
E-mail: 1911036@tongji.edu.cn

✉ Lin Zhang
E-mail: cslinzhang@tongji.edu.cn

Ying Shen
E-mail: yingshen@tongji.edu.cn

Shengjie Zhao
E-mail: shengjiezhao@tongji.edu.cn

Yicong Zhou
E-mail: yicongzhou@um.edu.mo

[1] School of Software Engineering, Tongji University, Shanghai 201804, China
[2] Department of Computer and Information Science, University of Macau, Macau 999078, China

1 Introduction

A surround-view system (SVS) usually consists of four to six wide-angle fisheye cameras. These cameras are mounted on the vehicle facing different directions, so as to realize a 360-degree-perception of the surrounding environment around the vehicle. By calibrating the SVS's intrinsics and extrinsics accurately, relative poses between cameras can be determined and then a high-quality surround-view can be synthesized in real time. The surround-view cannot only broaden the driver's view to eliminate blind areas, but also can be employed in parking-slot detection [14, 22, 39], autonomous parking [16,23,36,38], pedestrian detection [13,19] and other related driving assistance tasks.

After being extrinsically calibrated, cameras in the SVS should be fixed to keep extrinsics, which refer to the relative poses among cameras, unchanged. However,

collisions, bumps, and even the "hot expansion and cold contraction" effects may destroy the initial spatial structure of the camera system. In this case, if the initial extrinsics are still used and not properly adjusted, in generated surround-views, there will be observable geometric misalignment, reflecting the abnormality of the perception of the SVS.

When geometric misalignment appears in the synthesized surround-views, prompt correction of the SVS's extrinsics is of great significance for the driving safety. But at present, effective online extrinsics correction solutions that can be applied to the SVS are still lacking. Existing schemes in this field mainly have the following limitations.

a) Most of existing online extrinsics correction methods are designed for common multi-camera systems like binocular cameras. Although the SVS also belongs to multi-camera systems, such methods usually can't be easily extended to make them applicable to the surround-view case due to the particularity in the structure of the SVS. Concretely, most of existing online extrinsics correction schemes extract feature points and descriptors in the common-view regions of different cameras, and then match them to solve accurate extrinsics. However, in the SVS, common fields of views between adjacent cameras are much narrower and more distorted than those of common binocular cameras, and accordingly it is difficult to resolve high-quality feature pairs from these regions. More details of the SVS's characteristics can be referred to in Sect. 2.1.
b) Existing solutions which are feasible to the surround-view case mostly require relatively ideal environments. For example, the approaches proposed in [3, 5, 17, 32, 41] require that, on the ground, there must be two parallel lane-lines that can be clearly detected. Thus, they usually have noticeable limitations in both the usability and the generalization capability. To the best of our knowledge, on the premise that the framework is applicable to the surround-view case, Liu *et al.*'s approach [25] and the one proposed in this paper are the only two that have quite relaxed requirements for the working conditions. Specifically, these two approaches both simply require a flat ground with relatively rich natural textures for them to work.

Currently, online extrinsics correction solutions are rarely embedded in the commercial products due to the technical immaturity. Thus, drivers usually have to drive to 4S shops for re-calibration to correct inaccurate extrinsics. It is too cumbersome for both users and manufacturers doubtlessly. Many automobile manufactures are also looking for ways to update extrinsics of the SVS in an online manner without re-calibration. To fill such a research gap to some extent, in this paper, we propose an online extrinsics correction pipeline for the SVS, namely "ROECS" (Robust Online Extrinsics Correction of the Surround-view system). Our contributions are summarized as follows:

a) A new error model "bi-camera error" is designed. The inaccuracy of SVS's extrinsics is mainly manifested in the geometric misalignment in bird's-eye common-view regions of adjacent cameras. Inspired by this observation, for a point $\boldsymbol{p}_G$ on the surround-view, we can construct a bi-camera error term which is actually the discrepancy of pixel values between two corresponding pixels $\boldsymbol{p}_{C_i}$ and $\boldsymbol{p}_{C_j}$ on fisheye images. The bi-camera error term effectively measures the degree of the geometric misalignment on the surround-view at $\boldsymbol{p}_G$.
b) Based on the "bi-camera error" model, we present the online extrinsics correction pipeline "ROECS", which follows a sparse and semi-direct framework. Each qualified point on the surround-view can be used to construct a bi-camera error term and by summing up all points' terms, the overall error of the system can be obtained. It's worth mentioning that we use ten frames selected by our frame selection strategy and stored in a local window rather than a single frame to build the overall error, so as to improve the system's robustness. The frame selection strategy will be introduced in Sect. 5.2. By iteratively minimizing the system's overall error with any non-linear optimization scheme, the optimal camera poses can then be figured out.
c) To further improve the speed and the accuracy of ROECS, we also propose a novel pixel selection strategy. The selection process mainly consists of three steps, common-view judgement, gradient screening and mismatched object elimination. Thanks to such a selection strategy, pixels with tiny gradient moduli and "mismatched" pixels, which will be demonstrated in detail in Sect. 5.1, can be effectively eliminated to reduce the computational cost and the effects of noise.

2 Related Work

2.1 Camera Pose Estimation

Since the essence of cameras' extrinsics calibration is to solve cameras' poses, here we will make a brief review on related researches of camera pose estimation. The estimation of camera poses is the core problem in many

fields, such as camera calibration, SFM (structure-from-motion), and visual SLAM (simultaneous localization and mapping). Most common camera pose estimation methods roughly fall into three categories, feature-point-based ones, optical-flow-based ones and direct ones.

Feature-point-based methods. Feature-point-based methods are generally composed of three steps, feature extraction, feature matching and pose estimation. The step of feature extraction is to detect feature points on the current frame, and then compute the local descriptor for each point. SIFT [27], SURF [2] and ORB [34] are all commonly utilized feature types. Among them, the matching accuracy of SIFT features is the highest, and the extraction speed of ORB features is the fastest. Therefore, users often trade off between the accuracy and the speed to find the most proper feature type. When the feature extraction is completed, features in the current frame should be matched with registered ones to obtain paired features. Finally, with a set of paired features, an estimation of the camera pose can be offered via the SVD decomposition, EPnP, ICP or any other proper schemes.

Feature-point-based methods are currently the most mature and widely adopted ones in this field. However, without the assistance of calibration sites or chessboards, they will be not suitable for the extrinsics' calibration of the SVS. This is because the accuracy of feature-point-based schemes largely depends on the quality of feature pairs. If there are mismatched feature pairs, the wrong information will be introduced, and such information will persist along with the entire process of the pose estimation. Thus, if high-quality paired features cannot be obtained, feature-point-based schemes are usually infeasible. Unfortunately, the SVS does have the following characteristics that are not conducive to the extraction and matching of features:

a) In the SVS, wide-angle fisheye cameras are mostly used. The distortion of such kind of cameras is often huge and difficult to be completely eliminated even by the undistortion algorithm designed specifically for fisheye cameras. Besides, this phenomenon will be more noticeable at fisheye images' boundaries, which are just common-view regions of adjacent cameras in the SVS. Since modern descriptors of feature points mostly can achieve the scale-and-rotation invariance but not "distortion invariance", they are unlikely to perform well on such regions.
b) In the vehicle-mounted SVS, cameras are usually mounted around the vehicle. Compared with other multi-camera systems such as binocular ones, cameras in the SVS are farther apart. Therefore, common-view regions between adjacent cameras are relatively narrow, which undoubtedly increases the difficulty of feature matching.

Optical-flow-based methods. Optical-flow-based methods are generally of two steps. Compared with feature-point-based ones, since the optical-flow-based ones track features based on the "grayscale invariance" assumption, such methods do not rely on the computation of descriptors. Thus, optical-flow-based schemes don't have the step of feature extraction but only includes two steps of feature tracking and pose estimation. Since the step of pose estimation in optical-flow-based schemes is similar to that in feature-point-based ones, we only discuss the feature tracking step here. LK optical-flow [28] proposed by Lucas and Kanade is a representative work in this field. One major shortcoming of the LK optical-flow is that it's a sparse optical-flow scheme and can't be applicable in the dense case. Horn and Schunck introduced a global smoothing hypothesis to the basic constraint equation of the optical-flow scheme, thereby proposing a dense optical-flow algorithm [18]. In Nagel's work [31], based on previous researches, they additionally smoothed the image by a weighted matrix and furtherly introduced a conditional smoothing constraint. Fleet and Jepson first presented the idea of using the phase information for the computation of the optical-flow [11]. Compared with the brightness, the phase is more reliable, and thus, the optical-flow fields obtained from the phase information usually exhibit better robustness. However, one major shortcoming of such a scheme is that, the time cost of optical-flow tracking will also increase by introducing the phase information.

As aforementioned, SVSs have several characteristics that are not conducive to the extraction and the matching of features. Though there is no step of feature extraction in optical-flow-based schemes, feature matching is still indispensable. Thus, in most cases, optical-flow schemes are also improper to be used in the extrinsics' correction of the surround-view case.

Direct methods. Direct methods do not require feature extraction or feature matching, and can directly estimate cameras' poses. Direct methods are essentially improved versions of optical-flow-based methods and were first proposed by Irani and Anandan in [20]. As optimization-based methods, the core idea of such schemes is to minimize the photometric error, thereby offering accurate camera poses' estimations. In recent years, more and more scholars are willing to replace feature-point-based methods or optical-flow-based methods by direct ones, especially in the field of SLAM. Engel *et al.*'s work presented in 2014, LSD-SLAM [10], is a typical semi-direct SLAM system. It was one of the most advanced monocular SLAM systems at that time. In the same year, Forster *et al.* proposed SVO [12]. This is an

influential visual odometry with distinguished processing speed which also follows the semi-direct framework. In 2017, Engel *et al.* [9] proposed DSO. To this day, it is still one of the most advanced SLAM systems. Engel *et al.* claimed that DSO is five times faster than ORB-SLAM [30,33], a representative feature-point-based monocular SLAM system.

There is no feature matching step in direct methods, or to be more exact, the feature matching is performed simultaneously with the pose estimation. Therefore, the matching relationships of features are not fixed along with the optimization evolvement. During the optimization, pixels that brought wrong information temporarily will not always introduce wrong information in the future, as long as the vast majority of pixels give the correct "guidance". As the accuracy of camera poses increases, the probability of giving wrong information by these pixels will decrease accordingly. As aforementioned, it is difficult to obtain high-quality paired features between adjacent cameras in the SVS. Hence, the performance of feature-point-based schemes and optical-flow-based ones is doomed to be limited. By contrast, since direct methods have no dependence on feature matching, compared with their feature-point-based and optical-flow-based counterparts, they will have inherent advantages in the SVS's extrinsics correction task.

2.2 Online Extrinsics Correction for the Multi-camera System

The SVS, which we are going to focus on, belongs to a particular type of multi-camera systems, which are sensors composed of at least two cameras. Binocular cameras and panoramic cameras are both typical multi-camera systems. To determine the relative poses between different cameras, the extrinsics calibration is always indispensable. However, since most offline calibration frameworks are cumbersome, it will take a lot of time and labor cost for each time to complete the whole calibration process. Thus, once extrinsics change after the offline calibration, how to update extrinsics in an online manner without resorting to re-calibration becomes quite significant. Based on the type of features utilized, existing online extrinsics correction methods for multi-camera systems roughly fall into two categories, manmade-feature-based ones and natural-feature-based ones, and we will review these two categories of schemes in this subsection, respectively. Besides, in Sect. 7, we also compared the characteristics of related researches in this field and ROECS qualitatively and summarized the results in Table 1.

Manmade-feature-based methods. Since manmade-feature-based methods often strongly rely on some specific assumptions, relatively ideal environmental conditions are indispensable for them. One of the earliest work in this field is Collado *et al.*'s in [5]. To begin with, they extracted patterns from two parallel lane-lines with the Sobel operator and the Hough transform. Then camera poses can be estimated in an online manner with lane-lines' patterns. In [32], Nedevschi *et al.* proposed a solution based on the vanishing point estimation. They first estimated the vanishing point of two lane-lines parallel to each other, and then extrinsics of the multi-camera system can be calibrated with the position of the estimated vanishing point. In Hold *et al.*'s work [17], a method of the online extrinsics calibration also using ground lane-lines was presented. They detected lane-lines and then sampled them with the scanning line to obtain a set of equidistant feature points. After that, the distance of these lane points will be measured by the fast Fourier transform. Finally, they estimated the extrinsics of the camera system with these lane points. In [41], Zhao *et al.* estimated multiple vanishing points rather than a single one to calibrate cameras' orientations. Although most of the aforementioned manmade-feature-based frameworks perform satisfactorily in both the speed and the accuracy, they are all not applicable to the SVS. In [3], Choi *et al.* proposed an online extrinsics calibration pipeline which is specially designed for the surround-view case. They aligned lane-line markings across images captured by adjacent cameras and then the SVS can be extrinsically online calibrated.

Manmade features, such as parallel lane lines, may be more stable than those natural features. However, in reality, environmental conditions are not always ideal. For example, lane-lines on the ground may be crooked and faded, or maybe the car is running on a rural path without lane-lines. Thus, the application scope of these manmade-feature-based schemes is actually quite limited.

Natural-feature-based methods. On account of the limited application scope of manmade-feature-based approaches, more and more researchers focus on substituting manmade features in specific environments with natural features that could be extracted in common scenes. We refer to these solutions as "natural-feature-based" ones.

One of the earliest relevant researches in this field can be traced back to Dang *et al.*'s work in [6]. They formulated a Gauss-Helmert model for the self-recalibration task. The model consists of three different categories of constraint equations, the bundle-adjustment constraint, the epipolar constraint, and the trilinear constraint. Hansen *et al.*'s approach in [15] is a typical natural-feature-based method. They matched feature points among

different frames and then estimated extrinsics through the bundle adjustment. To guarantee the efficiency of the scheme, features are sampled sparsely. And since the pose estimation based on sparse features is not robust, they exploited a sequence of frames rather than a single one to weaken the effect of the noise and the information from multiple frames was fused by the Extended Kalman Filter. Knorr *et al.* [21] established an optimization algorithm which seats on a recursive structure. Similar to Hansen *et al.*'s approach, they also resorted to the Extended Kalman Filter to correct relative camera poses, thus a sequence of frames are required for the pipeline to converge. After that, the relationship between the camera system and the ground is determined via the homography estimation. In Ling and Shen's approach [24], taking the initial offline calibration result as the starting point, they minimized the epipolar error by the non-linear optimization to find accurate camera poses. Besides, the accuracy of the calibration was evaluated by the minimum eigenvalue of the covariance matrix. It is worth mentioning that this method takes all cameras as a whole and supposes that relative poses among them are fixed and will not change. Consequently, it actually does not consider the optimization on relative poses between cameras.

However, these methods are all not applicable to the SVS. As far as we know, the only natural-feature-based method which is applicable to the SVS is Liu *et al*'s method [25]. They studied the online extrinsics correction problem in depth and their work is quite relevant to ours in this paper. They proposed two models, the "Ground Model" and the "Ground-Camera Model", and both of them can correct extrinsics by minimizing photometric errors. However, since they didn't take the interference of possible noise in various environments into consideration, their method still has limitations in both the accuracy and the robustness.

3 Bi-Camera Error Model

We propose a new error model named "bi-camera error", which is the core of ROECS. The bi-camera error term measures the photometric discrepancy of 2 corresponding points on original fisheye images captured by adjacent cameras. By minimizing the systems's overall error, which is mainly summed up by the square of all bi-camera error terms, accurate extrinsics of the SVS can be obtained. Since the final form of the error term is complex, in this section, firstly we will analyze the basic form of the error model, and other necessary refinements will then be shown incrementally. The basic structure of the error model is illustrated in Fig. 1.

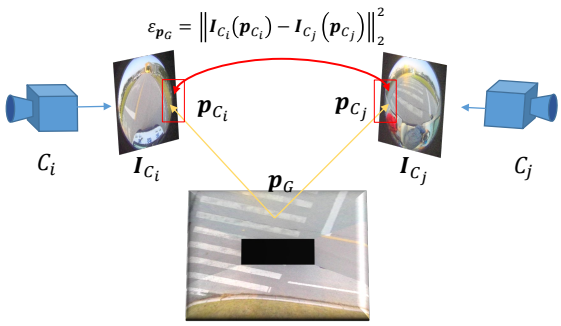


Fig. 1 Illustration of the bi-camera model. This is a kind of photometric error essentially. For any qualified point selected by our pixel selection strategy, a corresponding bi-camera error term can be constructed.

3.1 Basic Form

The surround-view calibration mainly consists of two components, the intrinsics calibration and the extrinsics calibration. Since the intrinsics calibration is irrelevant to the crux of this paper, it won't be discussed due to its complexity. More details can be found in [8, 40, 42]. It's to be observed that with the calibrated intrinsics, fisheye images can then be undistorted. To simplify notations, without special instructions, all pixels are sampled in the undistorted image coordinate system in this paper.

Expressed by a 6 DOF rigidbody transform matrix $\boldsymbol{T}$ in the homogeneous coordinate system [35], the extrinsics of a camera in the SVS are essentially the quantized relative spatial relationship between its camera coordinate and the ground coordinate. Suppose that an SVS is composed of four fisheye cameras, C_1, C_2, C_3 and C_4. For a camera C_i, the mapping relationship between an observed point $\boldsymbol{p}_G$ on the surround-view coordinate system and a corresponding point $\boldsymbol{p}_{C_i}$ on the undistorted image $\boldsymbol{I}_{C_i}$ is given by,

$$\boldsymbol{p}_{C_i} = \frac{1}{Z_{C_i}} \boldsymbol{K}_{C_i} \boldsymbol{T}_{C_i G} \boldsymbol{K}_G^{-1} \boldsymbol{p}_G \tag{1}$$

where $\boldsymbol{K}_{C_i}$ is the intrinsic matrix of C_i. The extrinsics of camera C_i consist of all elements in the matrix $\boldsymbol{T}_{C_i G}$, which is the pose of camera C_i with respect to the ground coordinate system. Z_{C_i} is the depth of $\boldsymbol{p}_G$ in C_i's coordinate system. $\boldsymbol{K}_G$ is the transform matrix from the ground coordinate system to the surround-view coordinate system, which is given by,

$$\boldsymbol{K}_G = \begin{bmatrix} \frac{1}{d_{X_G}} & 0 & \frac{W}{2d_{X_G}} \\ 0 & -\frac{1}{d_{Y_G}} & \frac{H}{2d_{Y_G}} \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

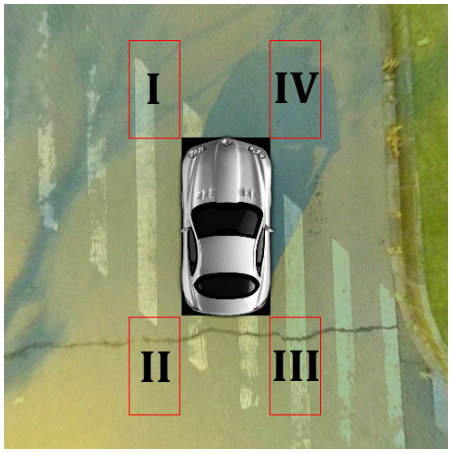


Fig. 2 The surround-view image and common-view regions on the surround-view. There are 4 common-view regions marked on the figure as the Roman numericals I, II, III and IV.

where (d_{X_G}, d_{Y_G}) stands for the size of the physical area on the ground plane corresponding to each pixel of the surround-view, and W and H are the width and height of the surround-view, respectively.

If $\boldsymbol{p}_G$ can be seen by both C_i and C_j, its two projections $\boldsymbol{p}_{C_i}$ and $\boldsymbol{p}_{C_j}$ on undistorted images $\boldsymbol{I}_{C_i}$ and $\boldsymbol{I}_{C_j}$ can then be obtained using Eq. 1. For $\boldsymbol{p}_G$, we define its corresponding bi-camera error term $\varepsilon^{bi}_{\boldsymbol{p}_G}$ as,

$$\varepsilon^{bi}_{\boldsymbol{p}_G} = \boldsymbol{I}_{C_i}\left(\boldsymbol{p}_{C_i}\right) - \boldsymbol{I}_{C_j}\left(\boldsymbol{p}_{C_j}\right) \tag{3}$$

By combining Eq. 1 and Eq. 3, we have obtained the basic form of the bi-camera error term. To minimize the error under the non-linear optimization framework effectively, we introduce the "Lie algebra representation" [7] and the "inverse depth" [4] to reformulate the bi-camera error as,

$$\begin{aligned}\varepsilon^{bi}_{\boldsymbol{p}_G} =& \boldsymbol{I}_{C_i}\left(\lambda^{C_i}_{\boldsymbol{p}_G}\boldsymbol{K}_{C_i}\exp\left(\boldsymbol{\xi}^{\wedge}_{C_iG}\right)\boldsymbol{K}_G^{-1}\boldsymbol{p}_G\right)\\ &- \boldsymbol{I}_{C_j}\left(\lambda^{C_j}_{\boldsymbol{p}_G}\boldsymbol{K}_{C_j}\exp\left(\boldsymbol{\xi}^{\wedge}_{C_jG}\right)\boldsymbol{K}_G^{-1}\boldsymbol{p}_G\right)\end{aligned} \tag{4}$$

where $\boldsymbol{\xi}_{C_iG}$ and $\boldsymbol{\xi}_{C_jG}$ are Lie algebra forms of C_i's pose and C_j's, respectively. $\lambda^{C_i}_{\boldsymbol{p}_G}$ is $\boldsymbol{P}_{C_i}$'s inverse depth and $\lambda^{C_j}_{\boldsymbol{p}_G}$ is $\boldsymbol{P}_{C_j}$'s. It's worth mentioning that $\lambda^{C_i}_{\boldsymbol{p}_G}$ and $\lambda^{C_j}_{\boldsymbol{p}_G}$ are not independent to each other. Their relationship can be represented as,

$$\lambda^{C_i}_{\boldsymbol{p}_G} = \frac{1}{[\boldsymbol{T}_{C_iC_j}(\lambda^{C_j}_{\boldsymbol{p}_G})^{-1}\boldsymbol{K}_{C_j}^{-1}\boldsymbol{p}_{C_j}]_3} \tag{5}$$

where the symbol $[*]_3$ stands for the coordinate value in the Z axis of the point, $\boldsymbol{T}_{C_iC_j}$ is the relative pose between C_i and C_j.

Since $\boldsymbol{p}_{C_i}$ and $\boldsymbol{p}_{C_j}$ are imaging points of the same physical object, for any qualified point $\boldsymbol{p}_G$ (actually selected by the pixel selection strategy discussed in Sect. 5.1), $\varepsilon^{bi}_{\boldsymbol{p}_G}$ should be equal to zero ideally. Thus, by summing up the square of all qualified points' error terms, the basic form of the objective function of the optimization in ROECS can then be obtained.

3.2 Necessary Refinements

Based on the basic form of the bi-camera error model, the basic form of the objective function of ROECS can then be formulated. However, to guarantee the performance of the optimization, some refinements are still necessary. Concretely, we introduce an exposure time factor and compute the error on a patch rather than on a single pixel.

Exposure correction. Since there should be definite discrepancies between different cameras' internal constructions in the SVS, for a same point $\boldsymbol{p}_G$ on the ground, corresponding imaging pixel values $\boldsymbol{I}_{C_i}\left(\boldsymbol{p}_{C_i}\right)$ and $\boldsymbol{I}_{C_j}\left(\boldsymbol{p}_{C_j}\right)$ won't be completely the same, even extrinsics are absolutely accurate. Actually, for an image of a physical object, except for the properties of the object itself, the corresponding pixel value will also be determined by the exposure time, the vignette and a non-linear response function of the camera [9]. Based on our experience, the exposure time is the most important factor among them. Therefore, we define a factor $\boldsymbol{\gamma}_{ij}$ as the ratio of exposure times of C_i and C_j. $\boldsymbol{\gamma}_{ij}$ is defined as,

$$\boldsymbol{\gamma}_{ij} = \frac{t_i}{t_j} \tag{6}$$

where t_i is C_i's exposure time and t_j is that of C_j's. Then the bi-camera error term can be rewritten as,

$$\begin{aligned}\varepsilon^{bi}_{\boldsymbol{p}_G} =& \boldsymbol{I}_{C_i}\left(\lambda^{C_i}_{\boldsymbol{p}_G}\boldsymbol{K}_{C_i}\exp\left(\boldsymbol{\xi}^{\wedge}_{C_iG}\right)\boldsymbol{K}_G^{-1}\boldsymbol{p}_G\right)\\ &- \boldsymbol{\gamma}_{ij}\boldsymbol{I}_{C_j}\left(\lambda^{C_j}_{\boldsymbol{p}_G}\boldsymbol{K}_{C_j}\exp\left(\boldsymbol{\xi}^{\wedge}_{C_jG}\right)\boldsymbol{K}_G^{-1}\boldsymbol{p}_G\right)\end{aligned} \tag{7}$$

Though the exposure time of a camera can't be obtained directly without the photometric calibration in general, the factor $\boldsymbol{\gamma}_{ij}$ can be fitted as,

$$\gamma_{ij} = \frac{\sum_{\boldsymbol{p}_G\in\boldsymbol{\mathcal{O}}_{ij}}\boldsymbol{I}_{GC_i}(\boldsymbol{p}_G)}{\sum_{\boldsymbol{p}_G\in\boldsymbol{\mathcal{O}}_{ij}}\boldsymbol{I}_{GC_j}(\boldsymbol{p}_G)} \tag{8}$$

where $\boldsymbol{I}_{GC_i}$ and $\boldsymbol{I}_{GC_j}$ are bird's-eye view images of camera C_i and C_j, respectively. $\boldsymbol{\mathcal{O}}_{ij}$ is the set of all pixels in the common-view region of C_i and C_j on bird's-eye views. In sum, there are four such regions, which are

shown in Fig. 2, on the surround-view. With the exposure correction, the negative influence of intensity discrepancies aroused by different lighting conditions or environmental reflections can thereby be weakened effectively.

Compute loss on a patch. In most cases, the function of the pixel intensity of an image won't be absolutely smooth. Constructing the loss term with a single pixel, the optimization may easily fall into the local optimum due to the non-smoothness of the image. Therefore, to improve the robustness, rather than computing the error with just one pixel pair $\boldsymbol{p}_{C_j}$ and $\boldsymbol{p}_{C_i}$, we construct the loss term with $\boldsymbol{p}_{C_j}$ and nine points in a patch whose center is $\boldsymbol{p}_{C_i}$,

$$\begin{aligned}\varepsilon_{\boldsymbol{p}_G}^{bi} = \sum_{\boldsymbol{p}_s \in \boldsymbol{\mathcal{P}}} \frac{1}{|\boldsymbol{\mathcal{P}}|} \boldsymbol{I}_{C_i} \left(\lambda_{\boldsymbol{p}_G}^{C_i} \boldsymbol{K}_{C_i} \exp\left(\boldsymbol{\xi}_{C_iG}^{\wedge}\right) \boldsymbol{K}_G^{-1} \boldsymbol{p}_G + \boldsymbol{p}_s\right) \\ - \boldsymbol{\gamma}_{ij} \boldsymbol{I}_{C_j} \left(\lambda_{\boldsymbol{p}_G}^{C_j} \boldsymbol{K}_{C_j} \exp\left(\boldsymbol{\xi}_{C_jG}^{\wedge}\right) \boldsymbol{K}_G^{-1} \boldsymbol{p}_G\right)\end{aligned} \tag{9}$$

where $\boldsymbol{\mathcal{P}}$ can be considered as the set of relative pixel coordinates' shift of all points in the local patch to the patch center, and $|\boldsymbol{\mathcal{P}}|$ is the size of set $\boldsymbol{\mathcal{P}}$. $\boldsymbol{\mathcal{P}}$ is defined as,

$$\boldsymbol{\mathcal{P}} = \{[i, j]^T \,|i, j = -2, 0, 2\} \tag{10}$$

4 Optimization

4.1 Objective of the Optimization

In this subsection, we will mainly introduce the form of ROECS's objective function in the optimization. We consider C_i as the target camera and C_j as the reference camera. In each iteration of the optimization, to keep the optimal solution unique, only C_i's pose $\boldsymbol{\xi}_{C_iG}$ and $\boldsymbol{P}_{C_j}$'s inverse depth $\lambda_{\boldsymbol{p}_G}^{C_j}$ will be optimized. Both $\boldsymbol{\xi}_{C_jG}$ and $\boldsymbol{p}_{C_j}$ are fixed. It's worth mentioning that $\boldsymbol{p}_G$ is not fixed, but changes with $\lambda_{\boldsymbol{p}_G}^{C_j}$.

In a single frame, for each of qualified points chosen by the pixel selection strategy, a bi-camera error term can be built. To improve the robustness of the pipeline, we will utilize pixels from ten frames, which are selected by our proposed frame selection strategy and stored in a local window, rather than a single one during optimization. Besides, there is also a prior knowledge that after the pixel selection, most of remained pixels should be from the flat ground. So for each point $\boldsymbol{p}_G$, we also add a prior error term $\varepsilon_{\boldsymbol{p}_G}^{prior}$ to the overall error to prevent the inverse depth $\lambda_{\boldsymbol{p}_G}^{C_j}$ from drastic changes. The prior error term $\varepsilon_{\boldsymbol{p}_G}^{prior}$ is defined as,

$$\varepsilon_{\boldsymbol{p}_G}^{prior} = \lambda_{\boldsymbol{p}_G}^{C_j} - \lambda_{\boldsymbol{p}_G}^{C_j*} \tag{11}$$

and the prior inverse depth $\lambda_{\boldsymbol{p}_G}^{C_j*}$ is defined as,

$$\lambda_{\boldsymbol{p}_G}^{C_j*} = \frac{1}{\left[\boldsymbol{P}_{C_j}\right]_3} = \frac{1}{\left[\exp\left(\boldsymbol{\xi}_{C_jG}^{\wedge}\right) \boldsymbol{K}_G^{-1} \boldsymbol{P}_G\right]_3} \tag{12}$$

where the symbol $[*]_3$ stands for the coordinate value in Z axis of the point.

By summing up the square of all bi-camera error terms and prior error terms, the overall error of the system can be obtained and the optimal pose $\boldsymbol{\xi}_{C_iG}^*$ of camera C_i is given as,

$$\boldsymbol{\xi}_{C_iG}^* = \underset{\boldsymbol{\xi}_{C_iG}, \boldsymbol{\lambda}_{C_j}}{\arg\min} \sum_{(i,j) \in \boldsymbol{\mathcal{A}}} \sum_{f \in \boldsymbol{\mathcal{F}}} \sum_{\boldsymbol{p}_G \in \boldsymbol{\mathcal{N}_{ij}}} (\varepsilon_{\boldsymbol{p}_G}^{bi})^2 + \rho_h((\varepsilon_{\boldsymbol{p}_G}^{prior})^2) \tag{13}$$

where ρ_h is the Huber kernel function, $\boldsymbol{\mathcal{A}}$ is the set of all adjacent camera pairs, $\boldsymbol{\mathcal{F}}$ is the set of all frames used in the optimization and $\boldsymbol{\mathcal{N}}_{ij}$ is the set of qualified points in the common-view region of C_i and C_j. $\boldsymbol{\lambda}_{C_j}$ is the vector of all qualified points' inverse depth values. As a least-square problem, it can be solved by any non-linear optimization method [26], like the steepest descent [1], the Gauss-Newton method [37] and the Levenberg-Marquardt (LM) method [29]. To achieve a rather better performance, in our implementations, we adopted the LM scheme.

4.2 Derivatives

To minimize the objective function in Eq. 13, the derivative relationships between the error and optimized vairables, which consists of camera poses and the inverse depth of each point, need to be determined. Since the form of the prior error is straightforward, in this paper we just analyze the derivation process of the bi-camera error term.

Derivative to the pose. The jacobian $\boldsymbol{J}_p$ of the bi-camera error term $\varepsilon_{\boldsymbol{p}_G}^{bi}$ to camera C_i's pose $\boldsymbol{\xi}_{C_iG}$ can be expressed as,

$$\boldsymbol{J}_p = \frac{\partial \varepsilon_{\boldsymbol{p}_G}^{bi}}{\partial \boldsymbol{\xi}_{GC_i}^T} \tag{14}$$

It can be decomposed to 4 parts with the chain rule,

$$\boldsymbol{J}_p = \frac{\partial \varepsilon_{\boldsymbol{p}_G}^{bi}}{\partial \boldsymbol{I}_{C_i}} \cdot \frac{\partial \boldsymbol{I}_{C_i}}{\partial \boldsymbol{p}_{C_i}^T} \cdot \frac{\partial \boldsymbol{p}_{C_i}}{\partial \boldsymbol{P}_{C_i}^T} \cdot \frac{\partial \boldsymbol{P}_{C_i}}{\partial \boldsymbol{\xi}_{C_iG}^T} \tag{15}$$

Then we will discuss these 4 parts one by one,

(1) $\partial \varepsilon_{\boldsymbol{p}_G}^{bi} / \partial \boldsymbol{I}_{C_i}$ is the derivative of the error $\varepsilon_{\boldsymbol{p}_G}^{bi}$ to pixel intensities of image $\boldsymbol{I}_{C_i}$. Obviously, this term can be ignored since it's equal to one.

$$\frac{\partial \varepsilon_{\boldsymbol{p}_G}^{bi}}{\partial \boldsymbol{I}_{C_i}} = 1 \tag{16}$$

(2) $\partial \boldsymbol{I}_{C_i}/\partial \boldsymbol{p}_{C_i}^T$ is the average intensity gradient, which is generally computed by the Sobel operator, of image $\boldsymbol{I}_{C_i}$ at the all pixels in the local window $\boldsymbol{\mathcal{P}}$ whose center is $\boldsymbol{p}_{C_i}$. Actually, this term can also be approximated just by the intensity gradient at $\boldsymbol{p}_{C_i}$ (the window of the Sobel operator needs to be enlarged accordingly). Thus, $\partial \boldsymbol{I}_{C_i}/\partial \boldsymbol{p}_{C_i}^T$ can be given as,

$$\frac{\partial \boldsymbol{I}_{C_i}}{\partial \boldsymbol{p}_{C_i}^T} = \left[\begin{array}{cc} \frac{\partial \boldsymbol{I}_{C_i}}{\partial u_{C_i}} & \frac{\partial \boldsymbol{I}_{C_i}}{\partial v_{C_i}} \end{array} \right] \triangleq \left[\begin{array}{cc} \nabla \boldsymbol{I}_{C_i}^{u_{C_i}} & \nabla \boldsymbol{I}_{C_i}^{v_{C_i}} \end{array} \right] \tag{17}$$

where u_{C_i} and v_{C_i} are both coordinate values of $\boldsymbol{p}_{C_i}$.

(3) $\partial \boldsymbol{p}_{C_i}/\partial \boldsymbol{P}_{C_i}^T$ is the derivative of a pixel's 2D coordinate to its 3D position in the camera coordinate system. From the pin-hole camera model, we have

$$\frac{\partial \boldsymbol{p}_{C_i}}{\partial \boldsymbol{P}_{C_i}^T} = \left[\begin{array}{ccc} \frac{f_x^i}{Z_{C_i}} & 0 & -\frac{f_x^i X_{C_i}}{Z_{C_i}^2} \\ 0 & \frac{f_y^i}{Z_{C_i}} & -\frac{f_y^i Y_{C_i}}{Z_{C_i}^2} \end{array} \right] \tag{18}$$

where f_x^i and f_y^i are focal lengths of C_i. X_{C_i}, Y_{C_i} and Z_{C_i} are coordinate values in three axes of $\boldsymbol{P}_{C_i}$ in C_i's coordinate system.

(4) $\partial \boldsymbol{P}_{C_i}/\partial \boldsymbol{\xi}_{C_iG}^T$ is the derivative of the 3D point $\boldsymbol{P}_{C_i}$ to the camera pose $\boldsymbol{\xi}_{C_iG}$,

$$\frac{\partial \boldsymbol{P}_{C_i}}{\partial \boldsymbol{\xi}_{C_iG}^T} = \left[\boldsymbol{I}_{3\times 3} \; -\boldsymbol{P}_{C_i}^{\wedge} \right] \tag{19}$$

where $\boldsymbol{I}$ is a 3×3 identity matrix and $\boldsymbol{P}_{C_i}^{\wedge}$ is the 3×3 anti-symmetric matrix generated from $\boldsymbol{P}_{C_i}$. By merging the four terms in Eqs. 16∼19, we can get the final form of the jacobian $\boldsymbol{J}_p$.

Derivative to the inverse depth. The jacobian $\boldsymbol{J}_d$ of the bi-camera error term $\varepsilon_{\boldsymbol{p}_G}^{bi}$ to point $\boldsymbol{p}_{C_j}$'s corresponding inverse depth $\lambda_{\boldsymbol{p}_G}^{C_j}$ can be expressed as,

$$\boldsymbol{J}_d = \frac{\partial \varepsilon_{\boldsymbol{p}_G}^{bi}}{\partial \lambda_{\boldsymbol{p}_G}^{C_j}} \tag{20}$$

With the chain rule, it can also be decomposed as,

$$\boldsymbol{J}_d = \frac{\partial \varepsilon_{\boldsymbol{p}_G}^{bi}}{\boldsymbol{P}_{C_i}^T} \cdot \frac{\partial \boldsymbol{P}_{C_i}}{\partial \boldsymbol{P}_{C_j}^T} \cdot \frac{\partial \boldsymbol{P}_{C_j}}{\partial \lambda_{\boldsymbol{p}_G}^{C_j}} \tag{21}$$

Next, these three simpler parts will be discussed one by one,

(1) $\partial \varepsilon_{\boldsymbol{p}_G}^{bi}/\partial \boldsymbol{P}_{C_i}^T$ is the derivative of the error $\varepsilon_{\boldsymbol{p}_G}$ to $\boldsymbol{p}_G$'s corresponding 3D position $\boldsymbol{P}_{C_i}$ in C_i's camera coordinate system. This term can be obtained by combining Eqs. 16 ∼ 18.

(2) $\partial \boldsymbol{P}_{C_i}/\partial \boldsymbol{P}_{C_j}^T$ is the derivative of $\boldsymbol{P}_{C_i}$'s 3D coordinate in C_i's coordinate system to its corresponding 3D point in C_j's coordinate system,

$$\frac{\partial \boldsymbol{P}_{C_i}}{\partial \boldsymbol{P}_{C_i}^T} = \boldsymbol{T}_{C_iG} \boldsymbol{T}_{C_jG}^{-1} = \exp\left(\boldsymbol{\xi}_{C_iG}^{\wedge}\right) \exp\left(\boldsymbol{\xi}_{C_jG}^{\wedge}\right)^{-1} \tag{22}$$

(3) $\partial \boldsymbol{P}_{C_j}/\partial \lambda_{\boldsymbol{p}_G}^{C_j}$ is the derivative of a 3D point $\boldsymbol{P}_{C_j}$ to its inverse depth. It can be expressed as,

$$\frac{\partial \boldsymbol{P}_{C_j}}{\partial \lambda_{\boldsymbol{p}_G}^{C_j}} = -\frac{1}{(\lambda_{\boldsymbol{p}_G}^{C_j})^2} \boldsymbol{K}_{C_j}^{-1} \boldsymbol{p}_{C_j} \tag{23}$$

As all derivative relationships between bi-camera error terms and optimized variables have been deduced, the objective function in Eq. 13 can then be minimized with any non-linear optimization scheme to get accurate extrinsics of the SVS.

5 Pixel Selection and Frame Selection Strategies

5.1 Pixel Selection Strategy

To improve the speed and the robustness of the system, the pipeline we proposed follows a sparse semi-direct framework. That is to say, pixels which meet specific requirements rather than all pixels will be chosen to build the overall error. The selected pixels on the surround-view should meet 3 requirements:

a) The pixel must be able to be at least (usually at most) in the field of view of two cameras.
b) The pixel should have enough intensity gradient modulus.
c) The pixel should be taken from the flat ground.

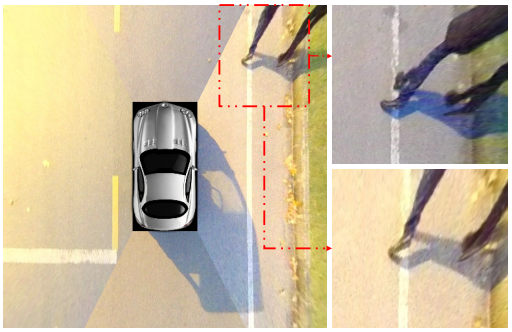
With the three requirements above, we established a novel pixel selection strategy. Take a pair of adjacent cameras C_i and C_j as an example. A set of pixels $\boldsymbol{\mathcal{N}}_{ij}$ will be selected out by the strategy. For any pixel $\boldsymbol{p}$ in $\boldsymbol{\mathcal{N}}_{ij}$, it must pass a three-step check, including common-view judgement, gradient screening, and mismatched object elimination. In this subsection, we will discuss each part of the pixel selection strategy in detail one by one.

Common-view Judgement. This is the first and simplest but most important rule. Common-view judgement means that the pixel must in the common-view region between adjacent cameras, and it is also the precondition of constructing the bi-camera error term. To describe the criterion, the point should be in region $\boldsymbol{\mathcal{O}}_{ij}$,

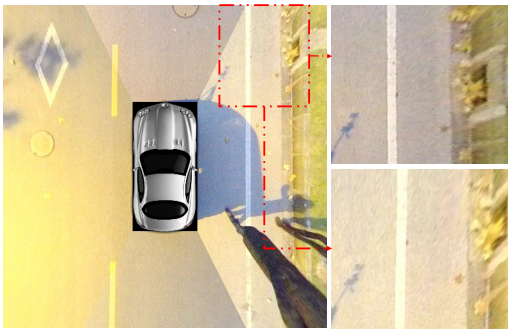
$$\boldsymbol{p} \in \boldsymbol{\mathcal{O}}_{ij} \tag{24}$$

where $\boldsymbol{\mathcal{O}}_{ij}$ is one of the common-view regions shown in Fig. 2.

Gradient Screening. Gradient screening is an approach that selects pixels with high gradient moduli while abandons those low ones. To the consideration of



(a)



(b)

Fig. 3 Typical examples of mismatched objects. The pedestrians in (a) and the curb in (b) are both typical mismatched objects, and they are marked in surround-views on the left. Enlarged areas on the right in each group from top to bottom are captured from the front-view and the right-view, respectively. It can be seen that there are obvious parallaxes between observations of mismatched objects on adjacent bird's-eye views.

the discrepancy in both overall pixel intensities and contrast between different images, a single constant threshold won't be always appropriate. Thus, we use a combination of the dynamic threshold and the constant threshold to select pixels. Concretely, this criterion can be formulated as,

$$G_i(\boldsymbol{p}) > max\{G_{mean} + \sigma_g, 70\} \tag{25}$$

where G_{mean} is the mean intensity gradient modulus of all pixels in $\boldsymbol{\mathcal{O}}_{ij}$ over the surround-view and σ_g is the associated standard deviation. 70 is an empirical value.

Mismatched Object Elimination. As aforementioned, the effectiveness of ROCES is based on the assumption that the ground is flat to a great extent. However, in reality, some objects with non-negligible heights, such as lawns, curbs or pedestrians, may appear on the surround-view. As shown in Fig. 3, such objects will cause parallax between the observations of adjacent cameras, thereby resulting in pixels' invalid matching in bird's-eye common-view regions. For the convenience of instruction, such objects are named "mismatched objects". Pixels from mismatched objects are named "mismatched pixels". And for those objects with negligible heights and their corresponding pixels, they are named "ground objects" and "ground pixels", respectively.

The existence of mismatched pixels will have a destructive effect on the performance of the system since it breaks the premise of the establishment of the bi-camera error model. Therefore, we design such a "mismatched object elimination" approach to cull mismatched pixels. Such a step consists of two sub-steps, homography alignment and color matching.

The first step is homography alignment. The 2D homography estimation can provide a good fitting for the plane motion. Therefore, since the vehicle is travelling on the flat ground, we can solve a homography matrix to estimate the motion of the vehicle. For 2 consecutive frames $\boldsymbol{I}^t_{GC_i}$ and $\boldsymbol{I}^{t+1}_{GC_i}$, we extract ORB features over them and then match these features on the hamming distance. After that, a homography matrix $\boldsymbol{H}^{t+1}_t$ is estimated via the "4-point" method. For the consideration of the robustness, the estimation seats on a RANSAC framework. It's worth mentioning that, in this sub-step, ORB feature points are used, so our method is "semi-direct" rather than "direct".

The second step is color matching. After the homography alignment, we warp $\boldsymbol{I}^{t+1}_{GC_i}$ by $\boldsymbol{H}^{t+1}_t$ to generate $\boldsymbol{I}^{t^{'}}_{GC_i}$. For any point $\boldsymbol{p}$ on $\boldsymbol{I}^{t^{'}}_{GC_i}$, there should be,

$$\boldsymbol{I}^{t^{'}}_{GC_i}(\boldsymbol{p}) = \boldsymbol{I}^{t+1}_{GC_i}(\boldsymbol{H}^{t+1}_t \boldsymbol{p}) \tag{26}$$

Since the color information is more discriminative than the gray information, the "mismatched object elimination" step is conducted based on color images rather than gray ones. Ideally, for any ground pixel $\boldsymbol{p}$, we have,

$$\boldsymbol{I}^{t^{'}}_{GC_i}(\boldsymbol{p}) = \boldsymbol{I}^{t}_{GC_i}(\boldsymbol{p}) \tag{27}$$

However, the mismatched pixels' source physical object are not in the same plane with the ground pixels'. Thus, for an mismatched pixel $\boldsymbol{p}$, $\boldsymbol{H}^{t+1}_t$ can't provide an outstanding motion estimation, and there will be obvious differences between $\boldsymbol{I}^{t^{'}}_{GC_i}(\boldsymbol{p})$ and $\boldsymbol{I}^{t}_{GC_i}(\boldsymbol{p})$. To measure this discrepancy in quantity, we firstly propose a coefficient, named "color ratio", and then use the standard deviation of $\boldsymbol{p}$'s color ratios in different channels as the measurement of its corresponding color discrepancy.

To simplify the notation, we use $\boldsymbol{I}_t$ to represent $\boldsymbol{I}^t_{GC_i}$, and use $\boldsymbol{I}_{t'}$ to represent $\boldsymbol{I}^{t^{'}}_{GC_i}$. Besides, let $\boldsymbol{I}^c_t$ and $\boldsymbol{I}^c_{t'}$ be the channel map of $\boldsymbol{I}_t$ and $\boldsymbol{I}_{t'}$ of channel c, respectively. The color ratio $r_c(\boldsymbol{p})$ of point $\boldsymbol{p}$ is defined as,

$$r_c(\boldsymbol{p}) = \frac{\boldsymbol{I}^c_{t'}(\boldsymbol{p})}{\boldsymbol{I}^c_t(\boldsymbol{p})} \tag{28}$$

Then as aforementioned, we use the standard deviation of $\boldsymbol{p}$'s color ratios in different channels as the measurement of its corresponding color discrepancy. It's worth mentioning that to improve the robustness, the color discrepancy is computed in a 4×4 local window $\boldsymbol{\mathcal{P}}_{\boldsymbol{p}}$ at $\boldsymbol{p}$,

$$D_{color}\left(\boldsymbol{p}\right) = \sum_{\boldsymbol{p}_w \in \boldsymbol{\mathcal{P}}_{\boldsymbol{p}}} \frac{1}{|\boldsymbol{\mathcal{P}}|} \sqrt{\frac{\sum_{c=1}^{n_c} \left(r_c\left(\boldsymbol{p}_w\right) - r_\mu\left(\boldsymbol{p}_w\right)\right)^2}{n_c}} \tag{29}$$

where n_c is the number of channels (normally 3) and r_μ is the average of all $\boldsymbol{p}$'s color ratios. For any $\boldsymbol{p} \in \boldsymbol{\mathcal{N}}_{ij}$, it must satisfy,

$$D_{color}\left(\boldsymbol{p}\right) < D_{mean} - \sigma_d \tag{30}$$

where D_{mean} is the average color discrepancy of all the points in $\boldsymbol{\mathcal{O}}_{ij}$ and σ_d is the associated standard deviation.

Our proposed pixel selection strategy can improve both the speed and the accuracy of the algorithm effectively. Besides, via such a pixel-selection approach, mismatched pixels can also be efficaciously distinguished and then culled to improve the robustness of ROECS.

5.2 Frame Selection Strategy

To keep the richness of textures and the uniformity of their distribution, we use multiple frames in a local window rather than a single one to build the overall error and then activate the optimization. The candidate frame that can be added to the local window must meet following three criteria:

a) There should be at least five frames between the candidate frame and the last frame in the local window.
b) There should be enough features in the candidate frame. In our implementations, 6000 qualified pixels are required under the 1080p resolution. For lower resolutions, such a threshold can also be turned down accordingly.
c) The ground should be relatively flat, and there should not be too many mismatched objects in the field of the surround-view.

The first two constraints are straightforward, so here we will just explain how to build the third constraint, which is "the ground should be relatively flat". Although the pixel selection strategy we proposed can eliminate "mismatched pixels" through the "mismatched object elimination" step, such elimination can in fact only play an auxiliary role. Because when the ground has serious unevenness or there are too many mismatched objects, the accuracy of the homography estimation, which is an essential part in the "mismatched object elimination" process, cannot be guaranteed. In fact, when the ground is flat, the vehicle is approximately moving parallel to the imaging plane of the SVS, so the estimated homography matrix should be very close to the identity transform matrix. Thus, we utilize a simple heuristic determinant-based method to check the identity of the transform matrix. The homography matrix $\boldsymbol{H}_t^{t+1}$ mentioned in Sect. 5.1 can be expressed as,

$$\boldsymbol{H}_t^{t+1} = \begin{bmatrix} \boldsymbol{A}_{2\times 2} & \boldsymbol{t} \\ \boldsymbol{u} & 1 \end{bmatrix} \tag{31}$$

If $\boldsymbol{H}_t^{t+1}$ is an identity transform matrix, $\boldsymbol{A}_{2\times 2}$ should be an unit orthogonal matrix. Thus, we use the determinant of $\boldsymbol{A}_{2\times 2}$ to check the flatness of the ground. For a candidate frame, its corresponding matrix $\boldsymbol{H}_t^{t+1}$ should satisfy,

$$\left(Det\left(\boldsymbol{A}_{2\times 2}\right) - 1\right)^2 < \theta \tag{32}$$

where θ is a threshold. In our implementations, θ is set to 0.2.

6 Overall Pipeline and Implementation Details of ROECS

In Sect. 3 $\sim$ 5, we have presented details about our online extrinsics correction approach ROECS, which is specifically designed for the surround-view case. To provide the reader with a clear and overall understanding of our work, the overall pipeline, which is illustrated in Fig. 4, of ROECS will be demonstrated in this section.

The pipeline of ROECS mainly consists of three parts, and the first is the data preprocessing. While the vehicle is driving on the road, the SVS will continuously capture images and synthesize surround-views. Fisheye images captured by different cameras in the SVS at the time t are recorded as a group of images, G_t. After ROECS is activated, each time the SVS acquires the image group G_t, the step a) and the step c) in the frame selection is are firstly performed. If G_t can fit these two requirements, we will select pixels on G_t with the pixel selection strategy to find all qualified pixels that can be used in the optimization. Finally, the step b) of the frame selection will be conducted to ensure that there are enough qualified pixels. After G_t has passed the frame selection, we store G_t and related pixel selection results into the local window. When the number of image groups in the local window reaches the preset threshold n, the subsequent two steps in ROECS'

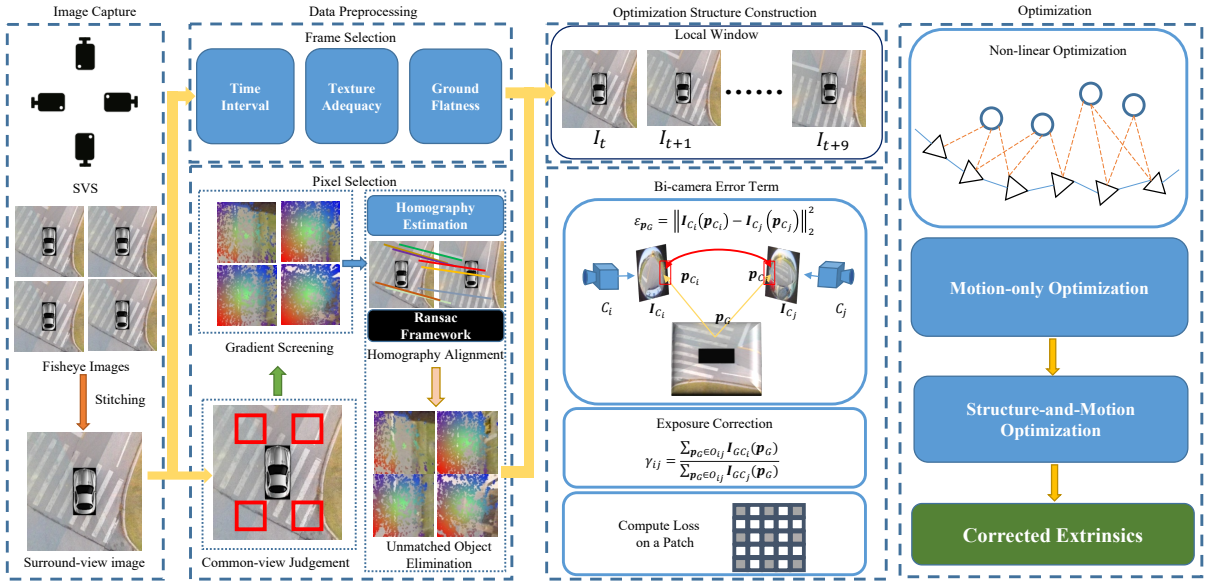


Fig. 4 The overall pipeline of ROECS. Each pixel $\boldsymbol{p}_G$ on the surround-view can be used to build a bi-camera error term. By minimizing the overall error of the system, which is composed of error terms from all frames in the local window, with the non-linear optimization, extrinsics can be iteratively optimized.

pipeline will then be executed. In our implementations, n is set to 10.

The second part of the pipeline is the establishment of optimized structure. Suppose I_t^i is one of the image in group G_t, which is stored in the local window. All qualified pixels on I_t^i will be projected onto the correct adjacent cameras' view to construct a term of bi-camera error. Corresponding prior errors can also be built. By summing up all error terms, the overall error of the system can then be obtained.

When the first two steps of the pipeline have been performed, the optimization can then be conducted, and this is also the last component of the pipeline. Since we derived all the related derivatives, the overall error can be minimized by any nonlinear optimization scheme. In our implementation, the LM method is chosen. For the optimization, firstly one hundred iterations of the motion-only optimization is conducted, that is, fix the inverse depth of pixels and only optimize the camera pose. After that, the camera poses and pixels' inverse depth will be jointly optimized for another fifty iterations.

7 Experimental Results

7.1 Experiment Setup

To validate the performance of our proposed pipeline, we performed experiments on an electric car equipped with an SVS. Our SVS consists of four leopard LI-OV10640-490-GMSL fisheye cameras. The resolution, the field-of-view, and the acquisition frequency of cameras are 1920 × 1080, 190 degrees and 30 FPS, respectively.

We collected four groups of surround-views, and for each group, there are one hundred frames. Each group of frames corresponds to a specific environmental condition, which are characterized by (1) with rich textures, (2) with relatively rich textures, (3) with sparse textures, and (4) with obvious mismatched objects, respectively. All experiments mentioned in this section are based on these data.

7.2 Qualitative Experimental Results

Table 1 Qualitative comparison with related methods

method	method type	prior	SVS	feature type
Collado *et al.* [5]	manmade-feature	×	×	ground lanes
Nedevschi *et al.* [32]	manmade-feature	√	×	ground lanes
Hold *et al.* [17]	manmade-feature	×	×	ground lanes
Zhao *et al.* [41]	manmade-feature	×	√	ground lane
Choi *et al.* [3]	manmade-feature	×	√	ground lane
Dang *et al.* [6]	natural-feature	√	×	feature point
Hansen *et al.* [15]	natural-feature	×	×	feature point
Knorr *et al.* [21]	natural-feature	√	×	feature point
Ling and Shen [24]	natural-feature	√	×	feature point
Liu *et al.* [25]	natural-feature	√	√	dense pixels
ROECS	natural-feature	√	√	sparse pixels

Traits of Methods. From those four viewpoints shown in Table 1, we compared all methods discussed in Sect. 2.2 to demonstrate their characteristics more clearly. 1) Is this method manmade-feature-based or natural-feature-based? 2) Does it reuse the prior information

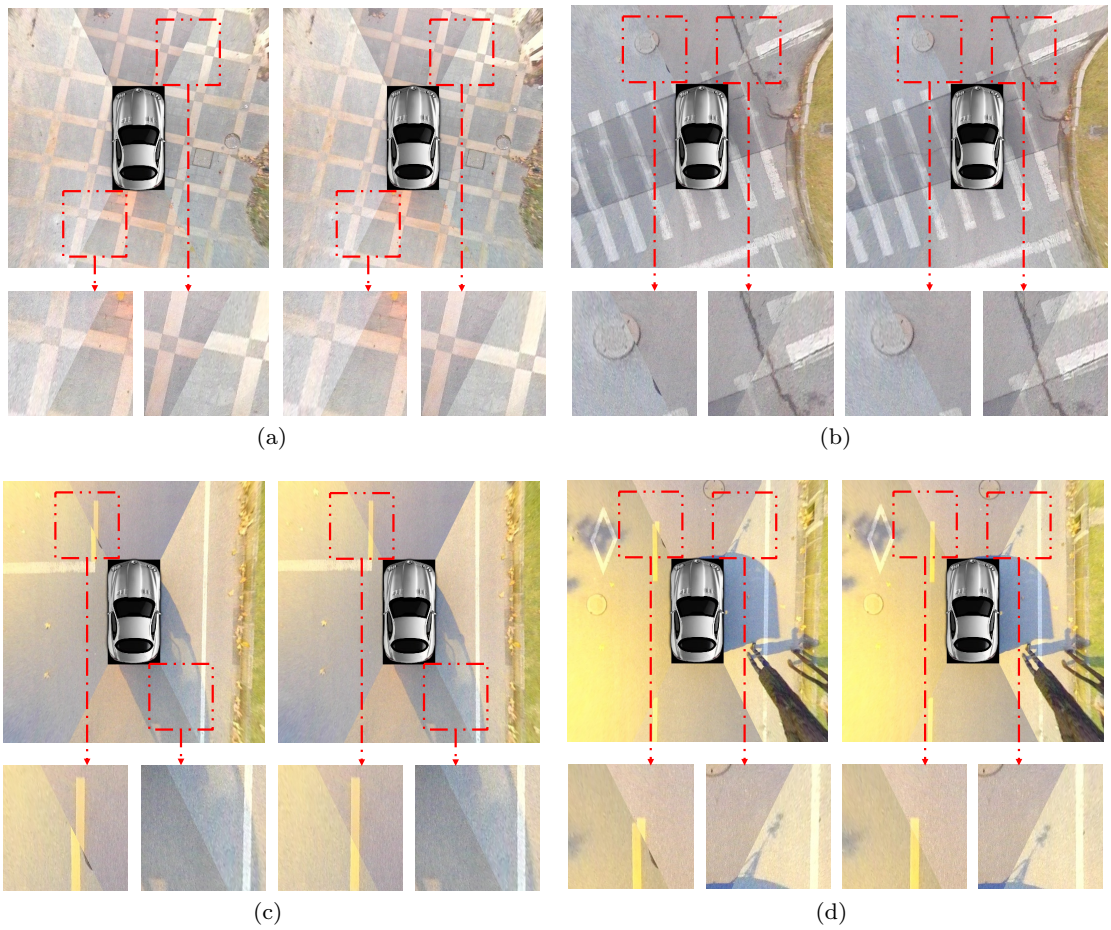


Fig. 5 Comparison of surround-views before and after extrinsics correction by ROECS in various environments. From (a) to (d), four pairs of images belong to four groups of the collected data mentioned in Sect. 7.1, respectively. For each pair, the upper left image is generated with inaccurate extrinsics while the upper right one is the result after the optimization. Enlarged local areas are shown on the bottom.

from the offline calibration? 3) Without complex extensions, can it be applicable to the SVS? 4) What kind of features does it rely on? It can be seen that only Liu *et al.*'s method and our ROECS can both reuse the offline calibration information as a prior and be applicable to the SVS. Compared with Liu *et al.*'s method, ROECS follows a sparse semi-direct framework. With our novel pixel selection strategy, mismatched pixels can be eliminated effectually. Furthermore, by using multiple frames rather than a single one, ROECS performs much better in terms of the robustness.

Typical Samples. To intuitively demonstrate the correction effect of ROECS, for each of the four groups of data aforementioned, we select a typical sample and show surround-views generated with both inaccurate extrinsics and corrected extrinsics in Fig. 5, respectively. It can be seen that the geometric misalignment in surround-views has been eliminated evidently through the correction, which qualitatively shows the effectiveness of ROECS.

Table 2 Generalization comparison with related methods

method	Group 1	Group 2	Group 3	Group 4
Collado *et al.* [5]	×	×	×	×
Nedevschi *et al.* [32]	×	×	×	×
Hold *et al.* [17]	×	×	×	×
Dang *et al.* [6]	×	×	×	×
Hansen *et al.* [15]	×	×	×	×
Knorr *et al.* [21]	×	×	×	×
Ling and Shen [24]	×	×	×	×
Zhao *et al.* [41]	×	×	√	√
Choi *et al.* [3]	×	×	√	√
Liu *et al.* [25]	√	√	√	√
ROECS	√	√	√	√

Application Scope. As aforementioned, ROECS performs better in the adaptation ability on environmental conditions compared with other existing related schemes. To make our demonstration more convincible, we have made qualitative comparisons in application scopes of ROECS and other related works. We recorded the applicability of those compared related methods discussed in Sect. 2.2 to the four sets of data, and the results are summarized in Table 2. It is worth noting that for those methods that are not applicable to the SVS, we con-

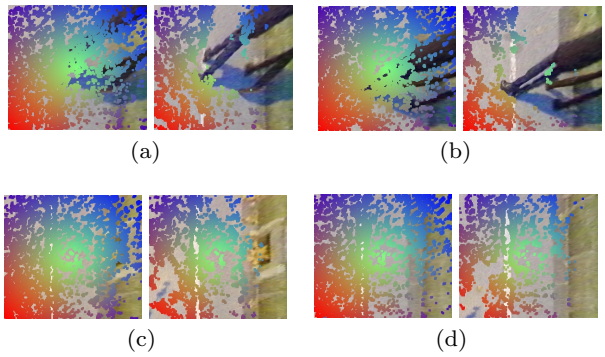


Fig. 6 Typical samples of the elimination effect of our pixel selection strategy over "mismatched pixels". For each group, the left image shows the selected pixels through the pixel selection without the step of mismatched object elimination, while right ones are results with the elimination.

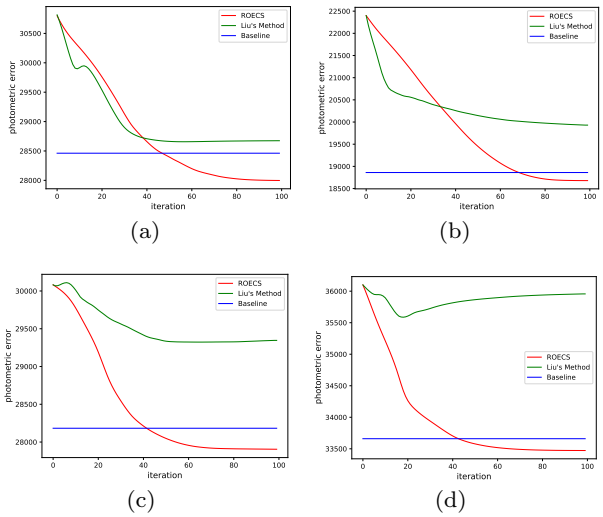


Fig. 7 (a)∼(d) are average photometric errors over all surround-views corresponding to group (a)∼(d) of the collected data, respectively.

sider that they cannot be used on all of the data. From the table, it can be seen that the generalization ability of the manmade-feature-based schemes is relatively poor. Among them, only Zhao *et al.*'s work and Choi *et al.*'s can be applicable to the SVS, but these schemes can only be used in the third and fourth sets of data, since there are no parallel lane-lines on the ground in first two groups. Among natural-feature-based schemes, only Liu *et al*'s framework and ours are designed for the surround-view case, and they can complete the extrinsics' correction on all of the data. Therefore, the experimental results corroborate that ROECS has loose requirements on external working environments, and thus has a good usability and a strong adaptation ability on environmental conditions.

Mismatched Pixels' Elimination. The mismatched object elimination step in the pixel selection can effectively eliminate mismatched pixels in surround-views, thereby improving the system's robustness. To show the effectiveness of such a process more intuitively, we select some typical samples and show them in Fig. 6. It can be seen that after the elimination, mismatched pixels from pedestrians, lawns and curbs are effectively eliminated, and it implies that ROECS will have better robustness with such a mismatched object elimination approach.

7.3 Quantitative Experimental Results

In this section, we will quantitatively analyze the performance of ROECS and compare it with other related methods. It is worth noticing that, as mentioned in Sect. 2.2, the existing related methods roughly fall into two categories, manmade-feature-based ones and natural-feature-based ones. Among them, manmade-feature-based ones usually have higher requirements on environmental conditions, while natural-feature-based ones are more flexible. As far as we know, among all natural-feature-based methods, only Liu *et al.*'s work and ours can be applicable to the surround-view case. Thus, in this section, we mainly quantitatively compare ROECS with Liu *et al.*'s framework.

Effectiveness and Robustness. In this experiment, with each group of images we collected, we tried to optimize the system's extrinsics with Liu *et al*'s scheme and ROECS, respectively. For each examined approach, its average photometric errors over all surround-views at sampled iterations during the optimization are summarized in Table 3 and trends of errors along with the optimization evolvement are shown in Fig. 7. For reference, we also offered an "offline baseline". The baseline is the average photometric error over all surround-views generated by offline calibrated extrinsics. Obviously, in most cases, ROECS performs much better than Liu *et al*'s method, especially in group (3) and (4). Besides, after 100 iterations of the optimization, extrinsics are even more accurate than offline calibrated results. Thus, the experimental results show the excellent effectiveness and robustness of ROECS quantitatively.

Generalization Ability. The generalization ability means that the extrinsics' correction can reduce photometric errors over most surround-views, even over those frames that are not utilized in the optimization. Without powerful generalization ability, the optimization is actually an "over fitting approach" rather than the effective correction. For each compared correction approach, we define those frames utilized in the optimization as "training frames" and other frames as "testing frames". For each training frame, we can randomly choose a corresponding testing frame. The only

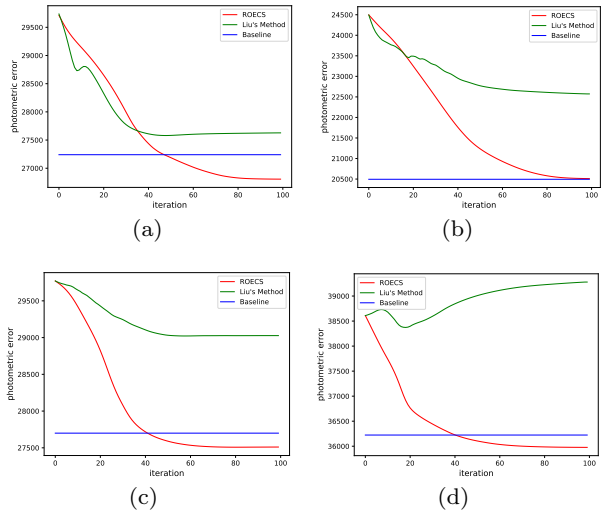


Fig. 8 (a)∼(d) are average photometric errors over corresponding "testing frames" in group (a)∼(d), respectively, along with the optimization evolvement of examined approaches.

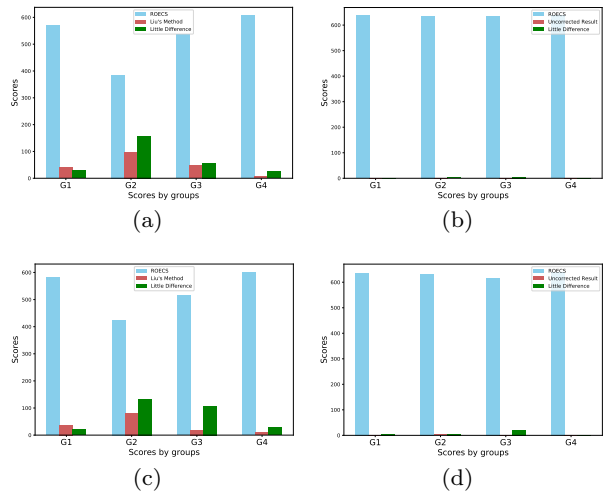


Fig. 9 The subjective scores of compared methods. The method will gain one point if one assessor think the corresponding corrected surround-view of this method is better than other compared ones. (a)∼(b) are results on the training frames while (c)∼(d) are on the testing frames.

discrepancy between this experiment and the previous one is that recorded photometric errors do not come from training frames, but from testing frames so as to evaluate the generalization ability. The result is illustrated in Fig. 8. It reflects that ROECS shows the generalization ability far beyond Liu *et al*'s work under various environmental conditions.

Subjective Evaluation. Since the photometric error can only reflect the accuracy of extrinsics to some extent, and there are no other better quantitative evaluation indicators to our knowledge, we conducted a subjective evaluation to measure the performance of compared schemes more comprehensively. Totally 8 people joined the experiment. Generated with the same group of original images, the uncorrected surround-view, the corrected result of ROECS, and the corrected result of Liu *et al*'s pipeline are all stored in one set. After all of the data was disorganized, for each set of the compared data, the assessor will subjectively select a surround-view with relatively slighter geometric dislocations. The statistical results are shown in Fig. 9. It can be seen that in most cases, ROECS can effectively correct the geometric misalignment, and its performance is far beyond that of Liu *et al*'s method.

Table 3 Comparison of photometric errors by examined approaches

Group	Method	iter = 0	iter = 50	iter = 100
1	Liu *et al*'s Method	30815.97	28664.91	28674.33
	ROECS	30815.97	28416.67	27997.77
	Offline Baseline	28461.83	28461.83	28461.83
2	Liu *et al*'s Method	22399.82	20155.71	19930.52
	ROECS	22399.82	19494.47	18679.48
	Offline Baseline	18861.56	18861.56	18861.56
3	Liu *et al*'s Method	30081.19	29340.13	29345.86
	ROECS	30081.19	28060.60	27904.57
	Offline Baseline	28182.59	28182.59	28182.59
4	Liu *et al*'s Method	36099.83	35861.47	35956.81
	ROECS	36099.83	33585.77	33472.62
	Offline Baseline	33660.04	33660.04	33660.04

Ablation Study of the Pixel Selection Strategy. Actually, without the pixel selection, the optimization in ROECS becomes a dense direct approach rather than the sparse one. In short, we call the optimization approach of our method with and without the pixel selection as the "sparse approach" and the "dense approach", respectively. Two factors are mainly considered in the evaluation of the pixel selection strategy, the speed and the accuracy. For the speed, since the speed of ROECS is directly related to the image resolution, we recorded time costs of both "sparse approaches" and "dense approaches" under different resolutions in Table 4. From the result, it can be seen that under the same experimental conditions, the optimizations in "sparse approaches" are all obviously faster than those in "dense approaches" regardless of the resolution. Though the solution usually does not asked to be real-time in the task of online extrinsics correction, faster speed is still an advantage. For the accuracy, the accuracy of each compared method is evaluated by the average photometric error calculated over all surround-views. The photometric errors along with the optimization evolvement of both the "sparse approach" and the "dense approach" are illustrated in Fig. 10. It can be seen that for all groups of the data, "sparse approaches" always perform better than "dense approaches" in accuracy. To sum up, the experimental results corroborate

Table 4 Time cost analysis of ROECS

Sparsity	Resolution	Time cost	Pixel number
Dense	1080p	2.8236s/iter	216000/frame
Sparse	1080p	0.2331s/iter	13476/frame
Dense	900p	1.8638s/iter	150968/frame
Sparse	900p	0.1613s/iter	9073/frame
Dense	720p	1.1332s/iter	96480/frame
Sparse	720p	0.0942s/iter	5896/frame
Dense	540p	0.6058s/iter	54000/frame
Sparse	540p	0.0437s/iter	3253/frame
Dense	360p	0.2729s/iter	24120/frame
Sparse	360p	0.0142s/iter	1415/frame
Dense	270p	0.1539s/iter	13600/frame
Sparse	270p	0.0077s/iter	804/frame

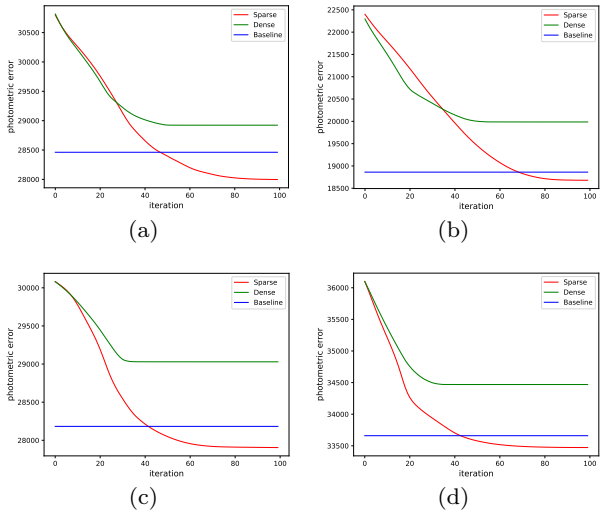


Fig. 10 The photometric errors of both "dense approaches" and "sparse approaches" along with the optimization evolvement. From (a) to (d) are the results corresponding to group (a)∼(d) of the data, respectively.

that both the speed and the accuracy can be enhanced effectively by our proposed pixel selection strategy.

8 Conclusion

In this paper, we studied a practical problem, online correction of cameras' extrinsics for the surround-view system, emerging from the field of ADAS, and proposed a novel solution namely ROECS. With ROECS, by minimizing the system's overall error over multiple frames chosen by our frame selection strategy, cameras' extrinsics can be optimized effectively. ROECS follows a sparse and semi-direct framework and fuses the prior information inherited from the offline calibration. One eminent feature of the proposed solution is that unmatched pixels can be eliminated effectively by our novel pixel selection strategy. Experimental results corroborated ROECS's superiority over the state-of-the-art competitors in this area.

References

1. Battiti, R.: First- and second-order methods for learning: Between steepest descent and newton's method. Neural Computation **4**(2), 141–166 (1992)
2. Bay, H., Tuytelaars, T., Gool, L.V.: Surf: Speeded up robust features. In: Proc. European Conf. Comput. Vis., pp. 404–417 (2006)
3. Choi, K., Jung, H., Suhr, J.: Automatic calibration of an around view monitor system exploiting lane markings. Sensors **18**(9), 2956:1–26 (2018)
4. Civera, J., Davison, A., Montiel, J.: Inverse depth parametrization for monocular slam. IEEE Trans. Robotics **24**(5), 932–945 (2008)
5. Collado, J., Hilario, C., Escalera, A., Armingol, J.: Self-calibration of an on-board stereo-vision system for driver assistance systems. In: Proc. Int'l IEEE Conf. Intell. Vehicles Symposium, pp. 156–162 (2006)
6. Dang, T., Hoffmann, C.: Tracking camera parameters of an active stereo rig. In: Proc. DAGM, p. 627–636 (2006)
7. Dennis, J.E., Schnabel, R.B.: The lie algebra of visual perception. J. Mathematical Psychology **3**(1), 65–98 (1966)
8. Du, F., Brady, M.: Self-calibration of the intrinsic parameters of cameras for active vision systems. In: Proc. IEEE Int'l Conf. Comput. Vis. Pattern Recognit., pp. 477–482 (1993)
9. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry. IEEE Trans. Pattern Analysis and Machine Intell. **40**(3), 611–625 (2018)
10. Engel, J., Schöps, T., Cremers, D.: Lsd-slam: Large-scale direct monocular slam. In: Proc. European Conf. Comput. Vis., pp. 834–849 (2014)
11. Fleet, D.J., Jepson, A.D.: Stability of phase information. IEEE Trans. Pattern Analysis and Machine Intell. **15**(12), 1253–1268 (1993)
12. Forster, C., Pizzoli, M., Scaramuzza, D.: Svo: Fast semi-direct monocular visual odometry. In: Proc. IEEE Int'l Conf. Robotics and Automation, pp. 15–22 (2014)
13. Gressmann, M., Palm, G., Löhlein, O.: Surround view pedestrian detection using heterogeneous classifier cascades. In: Proc. Int'l IEEE Conf. Intell. Transportation Systems, pp. 1317–1324 (2011)
14. Hamada, K., Hu, Z., Fan, M., Chen, H.: Surround view based parking lot detection and tracking. In: Proc. IEEE Intell. Vehicles Symposium, pp. 1106–1111 (2015)
15. Hansen, P., Alismail, H., Rander, P., Browning, B.: Online continuous stereo extrinsic parameter estimation. In: Proc. IEEE Int'l Conf. Comput. Vis. Pattern Recognit., pp. 1059–1066 (2012)
16. Hecker, S., Dai, D., Gool, L.V.: End-to-end learning of driving models with surround-view cameras and route planners. In: Proc. European Conf. Comput. Vis., p. 435–453 (2018)
17. Hold, S., Görmer, S., Kummert, A., Meuter, M., Muller-Schneiders, S.: A novel approach for the online initial calibration of extrinsic parameters for a car-mounted camera. In: Proc. Int'l IEEE Conf. Intell. Transportation Systems, pp. 420–425 (2009)
18. Horn, K.P., Schunck, B.G.: Determining optical flow. In: Proc. Int'l Joint Conf. Artificial Intell., pp. 185–203 (1981)
19. Hou, C., Ai, H., Lao, S.: Multiview pedestrian detection based on vector boosting. In: Proc. Asian Conf. Comput. Vis., pp. 18–22 (2007)
20. Irani, M., Anandan, P.: About direct methods. In: Proc. Int'l Workshop on Vis. Algorithms, pp. 267–277 (1999)

21. Knorr, M., Niehsen, W., Stiller, C.: Online extrinsic multi-camera calibration using ground plane induced homographies. In: Proc. IEEE Intell. Vehicles Symposium, pp. 236–241 (2013)
22. Li, L., Zhang, L., Li, X., Liu, X., Shen, Y., Xiong, L.: Vision-based parking-slot detection: A benchmark and a learning-based approach. In: Proc. IEEE Int'l Conf. Multimedia and Expo, pp. 649–654 (2017)
23. Lin, C., Wang, M.: A vision based top-view transformation model for a vehicle parking assistant. Sensors **12**(4), 4431–4446 (2012)
24. Ling, Y., Shen, S.: High-precision online markerless stereo extrinsic calibration. In: Proc. IEEE/RSJ Int'l Conf. Intell. Robots and Systems, pp. 1771–1778 (2016)
25. Liu, X., Zhang, L., Shen, Y., Zhang, S., Zhao, S.: Online camera pose optimization for the surround-view system. In: Proc. ACM Int'l Conf. Multimedia, pp. 383–391 (2019)
26. Lourakis, M.: Sparse non-linear least squares optimization for geometric vision. In: Proc. European Conf. Comput. Vis., pp. 43–56 (2019)
27. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int'l J. Comput. Vis. **60**(2), 91–110 (2004)
28. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proc. Int'l Joint Conf. Artificial Intell., pp. 404–417 (2006)
29. Moré, J.: The levenberg-marquardt algorithm: Implementation and theory. In: Numerical Analysis (Eds: G.A. Watson), pp. 105–116 (1978)
30. Mur-Artal, R., Tardós, J.D.: Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. Int'l J. Comput. Vis. **33**(5), 1255–1262 (2017)
31. Nagel, H.H.: On the estimation of optical flow: Relations between different approaches and some new results. In: Proc. Int'l Joint Conf. Artificial Intell., pp. 299–324 (1987)
32. Nedevschi, S., Vancea, C., Marita, T., Graf, T.: Online extrinsic parameters calibration for stereovision systems used in far-range detection vehicle applications. IEEE Trans. Intell. Transportation Systems **8**(4), 651–660 (2007)
33. R. Mur-Artal, J.M.M.M., Tardós, J.D.: Orb-slam: A versatile and accurate monocular slam system. IEEE Trans. Robotics **31**(5), 1147–1163 (2015)
34. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: Orb: An efficient alternative to sift or surf. Proc. IEEE Int'l Conf. Comput. Vis. pp. 2564–2571 (2011)
35. Shao, X., Liu, X., Zhang, L., Zhao, S., Shen, Y., Yang, Y.: Revisit surround-view camera system calibration. In: Proc. IEEE Int'l Conf. Multimedia and Expo, pp. 1486–1491 (2019)
36. Wang, C., Zhang, H., Yang, M., Wang, X., Ye, L., Guo, C.: Automatic parking based on a bird's eye view vision system. Advances in Mechanical Engineering **6**, 847406:1–13 (2014)
37. Wedderburn, R.W.M.: Quasi-likelihood functions, generalized linear models, and the gauss-newton method. Biometrika **61**(3), 439–447 (1974)
38. Xu, J., Chen, G., , Xie, M.: Vision-guided automatic parking for smart car. In: Proc. IEEE Intell. Vehicles Symposium, pp. 725–730 (2000)
39. Zhang, L., Huang, J., Li, X., Xiong, L.: Vision-based parking-slot detection: A dcnn-based approach and a large-scale benchmark dataset. IEEE Trans. Image Processing **27**(11), 5350–5364 (2018)
40. Zhang, Z.: Flexible camera calibration by viewing a plane from unknown orientations. In: Proc. IEEE Int'l Conf. Comput. Vis., pp. 666–673 (1999)
41. Zhao, K., Iurgel, U., Meuter, M., Pauli, J.: An automatic online camera calibration system for vehicular applications. In: Proc. Int'l IEEE Conf. Intell. Transportation Systems, pp. 1490–1492 (2014)
42. Zhu, H., Yang, J., Liu, Z.: Fisheye camera calibration with two pairs of vanishing points. In: Proc. Int'l Conf. Inf. Tech. Comput. Sci., pp. 321–324 (2009)