Extrinsic Self-Calibration of the Surround-View System: A Weakly Supervised Approach

Yang Chen, Lin Zhang, *Senior Member, IEEE*, Ying Shen, Brian Nlong Zhao, *Student Member, IEEE*, and Yicong Zhou, *Senior Member, IEEE*

***Abstract*—An SVS usually consists of four wide-angle fisheye cameras mounted around the vehicle to sense the surrounding environment. From the images synchronously captured by cameras, a top-down surround-view can be synthesized, on the premise that both intrinsics and extrinsics of the cameras have been calibrated. At present, the intrinsic calibration approach is relatively complete and can be pipelined, while the extrinsic calibration is still immature. To fill such a research gap, we propose a novel extrinsic self-calibration scheme which follows a weakly supervised framework, namely WESNet (Weakly-supervised Extrinsic Self-calibration Network). The training of WESNet consists of two stages. First, we utilize the corners in a few calibration site images as the weak supervision to roughly optimize the network by minimizing the geometric loss. Then, after the convergence in the first stage, we additionally introduce a self-supervised photometric loss term that can be constructed by the photometric information from natural images for further fine-tuning. Besides, to support training, we totally collected 19,078 groups of synchronously captured fisheye images under various environmental conditions. To our knowledge, thus far this is the largest surround-view dataset containing original fisheye images. By means of learning prior knowledge from the training data, WESNet takes the original fisheye images synchronously collected as the input, and directly yields extrinsics end-to-end with little labor cost. Its efficiency and efficacy have been corroborated by extensive experiments conducted on our collected dataset. To make our results reproducible, source code and the collected dataset have been released.[1]**

***Index Terms*—Extrinsic calibration, photometric loss, surround-view dataset, surround-view system, weakly supervised learning.**

Manuscript received 12 August 2021; revised 1 December 2021 and 4 January 2022; accepted 15 January 2022. Date of publication 25 January 2022; date of current version 7 June 2023. This work was supported in part by the National Natural Science Foundation of China under Grants 61973235, 61972285, and 61936014, in part by the Natural Science Foundation of Shanghai under Grant 19ZR1461300, in part by the Shanghai Science and Technology Innovation Plan under Grant 20510760400, in part by the Dawn Program of Shanghai Municipal Education Commission under Grant 21SG23, in part by the Shanghai Municipal Science and Technology Major Project under Grant 2021SHZDZX0100, and in part by the Fundamental Research Funds for the Central Universities. The guest editor coordinating the review of this manuscript and approving it for publication was Mr. Chuang Gan. *(Corresponding author: Lin Zhang.)*

Yang Chen, Lin Zhang, and Ying Shen are with the School of Software Engineering, Tongji University, Shanghai 201804, China (e-mail: 2011439@tongji.edu.cn; cslinzhang@tongji.edu.cn; yingshen@tongji.edu.cn).

Brian Nlong Zhao is with the Department of Computer Science, University of Southern California, Los Angeles, CA 90089 USA (e-mail: briannlongzhao@gmail.com).

Yicong Zhou is with the Department of Computer and Information Science, University of Macau, Macau 999078, China (e-mail: yicongzhou@um.edu.mo).

Digital Object Identifier 10.1109/TMM.2022.3144889

I. Introduction

As AN indispensable component of modern ADAS [1], the surround-view system (SVS) has been installed by more and more vehicles. In the SVS, four wide-angle fisheye cameras that cover the 360-degree field of view around the vehicle are mounted. Based on the multi-view geometry knowledge [2], the top-down surround-view images can be synthesized at runtime from the multi-stream videos collected synchronously. With the surround-view, the driver can conveniently check whether there are obstacles around the vehicle without blind spots and grasp their relative orientations and distances. In this way, the occurrence of scraping, collision and other accidents can be avoided effectively. Besides, the surround-view also plays an important role in multiple computer vision tasks [3] towards driving assistance, such as parking-slot detection [4], [5], autonomous parking [6], [7], pedestrian detection [8], [9] and so on.

To synthesize high-quality surround-views, the accurate intrinsics and extrinsics of cameras in the SVS are indispensable. At present, the performance of intrinsic calibration schemes is relatively satisfactory. The manufactures can complete the cameras' manufacturing and intrinsic calibration in a streamlined manner. In addition, since the cameras are always tightly encapsulated, the intrinsics will usually remain fixed after the production of cameras. Therefore, the intrinsic re-calibration is not frequently required in most cases. Relatively speaking, the techniques for extrinsic calibration are still not mature yet to meet the demands in use. The existing schemes for extrinsic calibration mainly fall into two categories, the manual ones [10]–[15] and the self-calibration ones [16]–[22], and their limitations are mainly manifested in the following two aspects:

1) Although the existing manual calibration schemes often preform reliably, most of them are cumbersome and labor-consuming. When these methods are utilized, the vehicle needs to be driven by professionals to a specific calibration site, and then the calibration can be completed using the patterns with the known-scale regularly arranged in the site. It can be seen that except for the high labor cost, these methods also have specific restrictions on the working environment. As aforementioned, due to collisions or bumps, sometimes the extrinsics of the SVS may change, which leads to the result that the manual schemes can only work with the professional assistance in an offline manner, and are not applicable for the online environment during driving.

[1] [Online]. Available: https://cslinzhang.github.io/WESNet/WESNet.html

2) The existing self-calibration schemes generally have inherent limitations with respect to the robustness and the stability. This is easy to understand given that they mostly heavily rely on low-level geometric features on the ground, such as pixels, points, and lines, and estimate the extrinsics based on the relatively ideal mathematical model. On the one hand, such low-level features are sensitive to the natural noise and the accuracy of the system will evidently decline in nonideal environments. On the other hand, some features (such as lines) are not widely available in the natural environment. Without required features on the ground, the corresponding self-calibration methods will fail.

On account of the limitations aforementioned, as far as we know, there is still no existing extrinsics self-calibration scheme specially designed for the SVS that can be stably applicable in various environmental conditions. In most commercial solutions, drivers have to drive to 4S stores for calibration or re-calibration by professionals. This is undoubtedly troublesome for both customers and automobile manufacturers. Thus, many manufacturers are now looking for effective self-calibration schemes. As an attempt to fill in this research gap to some extent, we propose a novel weakly-supervised scheme towards the extrinsics self-calibration of the SVS. In summary, our contributions are mainly threefolds:

1) A weakly supervised network for extrinsics calibration of the SVS, namely WESNet, is proposed. Based on the prior knowledge learned from the training data, WESNet can yield the extrinsics of the SVS in an end-to-end manner with the input of original fisheye images. Since it is difficult to obtain the accurate extrinsics as the ground truth (GT), we do not directly label the training samples with the corresponding GT extrinsics for fully supervised learning, but instead follow a weakly supervised framework. Specifically, the corner information in the calibration site images (collected over the calibration site) is taken as the geometric supervision, and in the first stage of training, the network is optimized by minimizing the geometric loss.
2) A novel photometric loss as self-supervised information is designed, so as to mine more supervision information from the training images themselves. Inspired by Zhang *et al.*'s scheme in [21], OECS, we model the imaging discrepancy in the common-view regions of adjacent cameras in the SVS as the photometric loss, and expect to minimize this loss as much as possible so as to synthesize seamless and high-quality surround-views. When the training fully based on the geometric supervision converges, the self-supervised photometric loss will be introduced to fine-tune the network to further improve the estimation accuracy, which is what our second stage of training is for.
3) To facilitate the study of the extrinsics calibration or any other computer vision tasks relying on the surround-views, we collected a large-scale surround-view dataset covering a variety of environmental conditions. Such a dataset contains 19,078 groups of high-resolution fisheye-images and the corresponding surround-views synthesized under different environmental conditions, covering the ground with several kinds of lane-lines and tiles, the cement road, the narrow path, and the road exposed to strong sunlight. Besides, a data augmentation method based on the homography transformation is also proposed, for the sake of improving the richness of the extrinsics of collected data. It is worth mentioning that, to our knowledge, this is the largest surround-view dataset containing original fisheye images. To make our results reproducible, source code and the collected dataset in this paper are online available at https://cslinzhang.github.io/WESNet/WESNet.html.

The remainder of this paper is organized as follows. Section II introduces related studies. Section III makes an overview of the imaging principle of the SVS. Section IV and V present our proposed network, WESNet, and the collected dataset in detail, respectively. Experimental results are reported in Section VI. Finally, Section VII concludes the paper.

II. Related Work

A. Extrinsics Calibration of the Surround-View System

To synthesize a surround-view image, the SVS needs to be both intrinsically and extrinsically calibrated. Since the intrinsics calibration is relatively mature and can satisfy the industrial requirements in most cases, in this paper, we mainly focus on the aspect of extrinsics. Based on whether the calibration can be conducted automatically, existing extrinsics calibration methods are mainly divided into two categories, the manual ones and the self-calibration ones.

1) Manual Calibration Methods: In manual calibration methods, specific patterns, such as corners, circles or lines, are necessary so as to offer the reference information. These patterns are usually repeatedly and equidistantly printed on the calibration site or some portable reference targets like the chessboard, and the coordinates of each pattern in the world coordinate system can be easily obtained. The driver needs to park the vehicle equipped with the SVS at the appropriate position, and then captures the calibration site images. After that, the extrinsics can be solved by establishing the mapping relationships of patterns between the pixel coordinates and the world coordinates.

In [10], Liu *et al.* firstly proposed the basic theoretical model of the SVS, pointing out that the mapping relationship between the undistorted fisheye image and the surround-view can be determined by homography estimation. However, the author did not offer a specific calibration pipeline. In fact, as far as we know, at that time, the SVS was still in the early stage where the tiles of fixed size on the ground were usually considered as the simple calibration patterns, which was undoubtedly unsatisfactory in the accuracy. In [11], Hedi *et al.* presented a two-stage offline calibration pipeline. In its first stage, the vehicle was parked on the calibration site filled with the chessboard markers. Then the extrinsics were roughly estimated via the homography estimation. After that, an optimization approach to minimize the stitching loss was conducted, which was also the second stage of the pipeline. Zhang *et al.*'s solution proposed in [12] is a calibration-chart-based approach, which utilized Harris corners [23] and BRIEF descriptors [24] to find paired features between the calibration-chart and the collected calibration site images. One eminent feature of their work is that except for the geometric alignment, photometric alignment was also introduced. In the scheme presented by Shao *et al.* [13], instead of driving the vehicle to a fixed position in a specific calibration site, a single chessboard was the only demand. And a novel refinement procedure that jointly optimized camera poses in a closed-loop manner was adopted. In recent years, a new variant of the SVS, 3D SVS, has attracted a lot of research interest and the studies on extrinsics calibration of the 3D SVS naturally emerged, such as Gao *et al.*'s work [14] and Zhang *et al.*'s one [15]. Actually, these methods are not significantly different from the aforementioned schemes designed for the conventional SVS in the calibration aspect. Specifically, Gao *et al.*'s solution is based on the calibration site printed with chessboard markers, and Zhang *et al.*'s one relies on the calibration chessboard, which are similar to the designs in [11] and [13], respectively.

With the assistance of calibration patterns, manual calibration methods often perform satisfactorily in both the stability and the accuracy. However, such calibration approaches usually need to be operated by professionals in specific sites, which causes high cost of manpower and materials. Besides, since these methods can only be applicable to the off-line environment, once the camera poses in the SVS changes due to collisions or bumps, the extrinsics obtained by the initial manual calibration will become inaccurate, and obvious geometric misalignment will appear in the synthesized surround-view.

2) Self-Calibration Methods: The self-calibration schemes are independent of the specific calibration patterns, and can recover the extrinsics only from the images taken in natural scenes, which effectively reduces the labor cost. In [16], Zhao *et al.* first detected multiple vanishing points of lane markings on the road via the weighted least squares method, and then with the estimated vanishing points, the pose of the multi-camera system relative to the world coordinate system was solved. In [17], Choi *et al.* also designed a lane-line based extrinsics self-calibration pipeline for the surround-view case, in which the SVS was calibrated by aligning lane markings across images of adjacent cameras. It can be seen that the aforementioned self-calibration frameworks both made an assumption for the target application environment, that is, there must be two parallel lane-lines clearly observed in the field of view. However, this is an assumption that cannot usually be satisfied. For example, lane-lines on the ground may be crooked and faded, or the car is likely to run on a rural path without lane-lines. Heng *et al.* [18], [19] resorted to visual SLAM systems to calibrate the extrinsics of the SVS and proposed an infrastructure-based pipeline. In their pipeline, there are no specific limitations on the application scope; however, the vehicle equipped with the SVS needs to travel in the calibration area for a while to establish the map, which is quite time-consuming and unlikely to satisfy the industrial portability requirement. Inspired by them, more and more studies trying to solve the task of camera pose estimation in multi-camera systems based on SLAM [26], [27] have emerged recently, but they can hardly be extended to SVS, which is also the reason why some solutions of camera calibration based on stereo images [28], [29] don't meet our requirement. As far as we know, the only three existing relatively lightweight self-calibration schemes which are applicable to the SVS are Liu *et al.*'s method [20] and Zhang *et al.*'s [21], [22]. They all deeply dissected the online extrinsics correction problem and offered effective solutions. In [20], Liu *et al.* proposed two models, namely the "Ground Model" and the "Ground-Camera Model", and both of them could correct extrinsics by minimizing photometric errors with the steepest descent [25]. In [21], Zhang *et al.* designed a novel model, the bi-camera model, to construct the least-square errors [30] on the imaging planes of two adjacent cameras and then optimize camera poses by the LM (Levenberg-Marquardt) algorithm [31]. And they further improved their work in [22] by utilizing multiple frames selected and stored in a local window rather than a single frame to build the overall error, so as to improve the system's robustness. Since the above three studies [20]–[22] focused on the "online correction" rather than the "calibration," a rough initial extrinsics needed to be offered to them as the input.

At present, most of the existing self-calibration methods can only utilize the low-level features, such as pixels, keypoints [32]–[34] or lines, to solve the extrinsics by aligning the features on different views. As discussed in Section I, these methods usually perform satisfactorily in ideal environments, but may fail without required textures on the ground. Compared with them, our proposed solution WESNet in this work follows a weakly supervised learning framework. Without any prior, it can effectively extract deep-level features and yield the accurate extrinsics end-to-end.

B. Learning-Based Calibration of the Camera System

In recent years, deep learning has shown superior performance in various computer vision tasks. Towards the calibration problem of camera systems, which is a classical task in the machine vision field, more and more learning-based solutions were proposed. In [35], Workman *et al.* proposed to regress the intrinsics of the camera directly from a single-shot via a convolutional neural network (CNN), namely FocalNet. Giering *et al.*'s approach in [36] is also an end-to-end CNN-based scheme, which took a multi-modal input including the point clouds from LiDAR, the optical flow maps and the RGB images. By solving a 9-class classification problem where each class corresponded to a particular x-y shift on an ellipse, the real-time lidar-video registration could be realized. In [37], Schneider *et al.* designed a network named RegNet, which is the first deep learning based work towards the extrinsics calibration of the LiDAR-camera system. Since it doesn't take geometric relationships into account, it has to be retrained each time the sensor intrinsics change. In contrast, the method presented in [38], namely CalibNet, deals with the problem in a weakly-supervised manner by attempting to reduce the dense photometric error and the point cloud distance error between the misaligned and the target depth maps. Despite some learning-based calibration schemes have been proposed, as far as we know, none of them are applicable to the surround-view case. Besides, most of the existing schemes are fully supervised, and the GTs are from traditional offline calibration solutions, implying that the accuracy of the trained networks is limited. For the consideration of the aforementioned limitations, our proposed WESNet, which is specially designed for the SVS, follows a weakly supervised framework. During its training phase, rather than generating GTs via existing offline calibration schemes, we take the re-projection loss of corners on the calibration site images as the weak supervision information. In addition, a novel photometric loss is also introduced as the self-supervision information to further improve the performance of the network.

III. Overview of the Surround-View System

This section describes the imaging process of a surround-view system, i.e., how to generate a surround-view from images captured by the cameras mounted around the vehicle. To synthesize a surround-view image, the mapping relationship of a point between its pixel coordinates on the original fisheye image and those on the bird's-eye-view should be established. Since such a relationship is relatively complex, we divide it into two parts, the mapping relationship from the pixel coordinates on the fisheye image to the ground coordinates and that from the 3D ground coordinates to the pixel coordinates in the bird's-eye-view. Next, we will introduce these two parts in detail.

Given the ground coordinate system O_G and a four-camera SVS (cameras are represented as C_1, C_2, C_3, and C_4), the poses of cameras in O_G are denoted by $\boldsymbol{T}_{C_1G}$, $\boldsymbol{T}_{C_2G}$, $\boldsymbol{T}_{C_3G}$, and $\boldsymbol{T}_{C_4G}$, respectively. The pose matrix $\boldsymbol{T}_{C_iG}$ is 4×4 and of 6 DOF (Degrees of Freedom), which can be expressed as,

$$\boldsymbol{T}_{C_iG} = \begin{bmatrix} \boldsymbol{R}_i & \boldsymbol{t}_i \\ \boldsymbol{0}^T & 1 \end{bmatrix}, i = 1, 2, 3, 4 \tag{1}$$

where $\boldsymbol{R}_i$ is an orthonormal 3×3 rotation matrix with $det(\boldsymbol{R}_i) = 1$ while $\boldsymbol{t}_i$ is a three dimensional translation vector.

For the transformation from the ground coordinate system to the coordinate system of the undistorted image, we formulate it with the pinhole camera model. Given an arbitrary point in the ground coordinate system $\boldsymbol{P}_G = [X_G, Y_G, Z_G, 1]^T$ in O_G, its corresponding pixel coordinate $\boldsymbol{p}_{C_i}$ on the imaging plane of C_i is given by,

$$\boldsymbol{p}_{C_i} = \frac{1}{Z_{C_i}} \boldsymbol{K}_{C_i} \boldsymbol{T}_{C_iG} \boldsymbol{P}_G, i = 1, 2, 3, 4 \tag{2}$$

where Z_{C_i} is the depth of $\boldsymbol{P}_G$ in camera C_i's coordinate system, and $\boldsymbol{K}_{C_i}$ is the 3×3 intrinsic matrix of camera C_i, which can be estimated together with the distortion coefficient matrix by Zhang's salient work [39] and some subsequent work of others [40], [41]. Concretely, the form of $\boldsymbol{K}_{C_i}$ is

$$\boldsymbol{K}_{C_i} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

where f_x, f_y, c_x, and c_y are camera intrinsic parameters. And it's worth noting that $\boldsymbol{p}_{C_i}$ is an undistorted point.

Compared with the above model, the transformation from the bird's-eye-view coordinate system to the ground coordinate system is much simpler, which is essentially a similarity transformation. The bird's-eye-view image can be generated by projecting a camera image to the ground, namely the plane $Z_G = 0$ in O_G. For a point $\boldsymbol{p}_G = [u_G, v_G, 1]^T$ in the bird's-eye-view coordinate system whose corresponding point in the ground coordinate system is $\boldsymbol{P}_G$, the transformation between them is given as,

$$\begin{bmatrix} u_G \\ v_G \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{d_{X_G}} & 0 & \frac{W}{2d_{X_G}} \\ 0 & -\frac{1}{d_{Y_G}} & \frac{H}{2d_{Y_G}} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_G \\ Y_G \\ 1 \end{bmatrix} \tag{4}$$

where X_G, Y_G, and Z_G are the coordinate values of $\boldsymbol{P}_G$, d_{X_G} and d_{Y_G} denote the size of each pixel[2], and W and H are the width and height of the scope covered by the surround-view image. Since that $Z_G = 0$, it is ignored implicitly here. Denote the transformation matrix from $\boldsymbol{P}_G$ to $\boldsymbol{p}_G$ by $\boldsymbol{K}_G$, and then (4) can be simplified as,

$$\boldsymbol{p}_G = \boldsymbol{K}_G \boldsymbol{P}_G \tag{5}$$

By combining (2) and (5), we can get,

$$\boldsymbol{p}_{C_i} = \frac{1}{Z_{C_i}} \boldsymbol{K}_{C_i} \boldsymbol{T}_{C_iG} \boldsymbol{K}_G^{-1} \boldsymbol{p}_G \tag{6}$$

With (6), we are able to establish a complete mapping between $\boldsymbol{p}_G$ on the bird's-eye-view and $\boldsymbol{p}_{C_i}$ in the undistorted imaging plane of C_i. With such a bijective relationship, considering each point $\boldsymbol{p}_G$ in the bird's-eye-view image $\boldsymbol{I}_{GC_i}$ captured by camera C_i, its corresponding pixel value can be obtained by,

$$\boldsymbol{I}_{GC_i}(\boldsymbol{p}_G) = \boldsymbol{I}_{C_i}(\boldsymbol{p}_{C_i}) \tag{7}$$

where $\boldsymbol{I}_{C_i}$ is the undistorted image captured by camera C_i. Mapping the images captured by cameras C_1, C_2, C_3, and C_4 to bird's-eye-views and then stitching them appropriately, a complete surround-view image can be synthesized.

IV. Weakly Supervised Camera Extrinsics Estimation

With accurate extrinsics, seamless surround-view images can be synthesized at runtime. However, as discussed in Section I, manual calibration schemes are usually laborious so that they can't be applied in the online manner, and off-the-shelf self-calibration schemes perform unsatisfactorily in the robustness and the generalization. To provide a robust and lightweight solution for the extrinsics calibration of SVS, in this paper, we propose a learning-based solution following the weakly supervised framework for camera extrinsics' estimation. Such a scheme is based on an end-to-end lightweight CNN, namely WESNet, which can yield the extrinsics directly from four input fisheye images captured synchronously by the cameras mounted around the vehicle. Under the weakly supervised framework, we mainly leverage the re-porjection loss of the corners of the calibration site instead of labelling every image in the dataset with its corresponding GT extrinsics.

Training with the weak supervision information, WESNet can offer a rough extrinsics estimation but the accuracy is still insufficient. To mine more information from the training data themselves, we also introduce the self-supervised photometric loss to fine-tune the network after the weakly supervised loss converges. Thus, the accuracy of the network can be further improved so as to synthesize seamless surround-views.

[2] More accurately, each pixel in the surround-view image corresponds to a $d_{X_G} \times d_{Y_G}$ physical area on the ground plane.

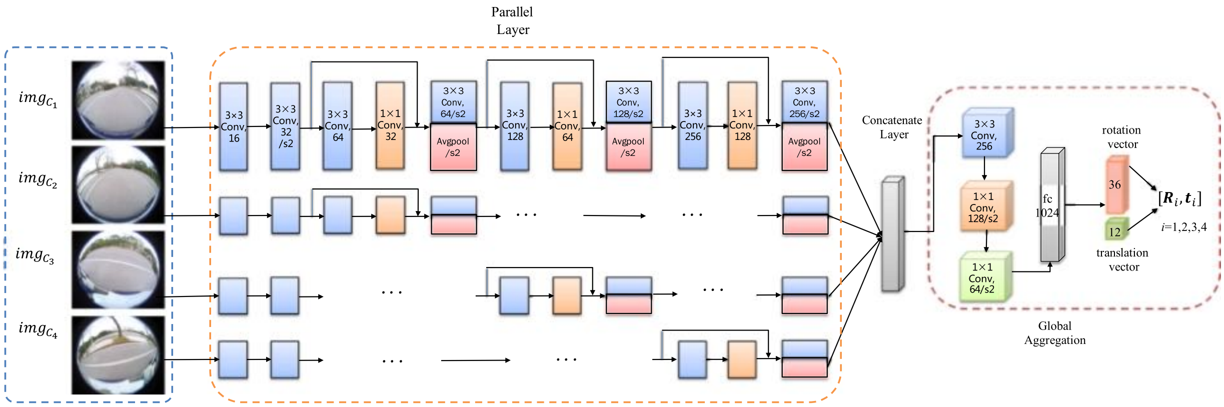


Fig. 1. The network architecture of WESNet. As illustrated in the figure, the parallel layer extracts deep-level features from four input images independently, and then the concatenate layer fuses the features from different images. Finally, the global aggregation layer maps the concatenated features to a 48-dimensional vector which stands for the extrinsics of the SVS. “s2” here means that the stride of the corresponding convolution operation is 2 while the default value is 1.

A. Network Architecture

The basic architecture of WESNet is designed mainly following the advice of regression frameworks for calibration [37], [38] as well as the common knowledge in CNN area. Fig. 1 illustrates its configurations. Here, a residual bottleneck [42] with a 3×3 convolution followed by a 1×1 one is used as the basic building block.

As shown in Fig. 1, the parallel layer independently extracts deep-level features from each input fisheye image. Then the feature maps from multipath will be concatenated together so as to fuse the extracted features from different views. Finally, the global aggregation layer, which consists of both convolutional layers and fully connected layers, will map the aggregated features to the extrinsics. As mentioned in (1), these extrinsics are formulated in the transformation matrix form, and for each camera, nine rotation parameters and three translation ones need to be regressed. Thus, WESNet will yield a 48-dimensional vector to represent the extrinsics of the SVS.

B. Loss Function

The loss function of WESNet is defined as the composition of three loss terms, the geometric loss, the orthogonal loss and the photometric loss, which is given as,

$$Loss = Loss_{geo} + \alpha Loss_{ortho} + \beta Loss_{pho} \tag{8}$$

where $Loss_{geo}$ is the geometric loss, $Loss_{ortho}$ stands for the orthogonal loss, and $Loss_{pho}$ refers to the photometric loss. The hyper-parameters are set to $\alpha = 0.1$ and $\beta = 0.15$. The geometric loss is actually the weakly supervised loss, which can promote the convergence of the network, while the photometric loss is to fine-tune the network so as to synthesize seamless surround-views. Besides, an orthogonal loss is also integrated to keep the internal constraints of the estimated rotation matrices. Next, we will introduce these three loss terms in detail.

1) Geometric Loss: As we know, the limitation of fully supervised learning is that its performance strongly depends on the accuracy of labels, while in the task of extrinsics calibration, it is difficult to obtain accurate GTs. Therefore, we do not take the extrinsics estimated by existing calibration solutions as direct labels, but utilize the geometric relationships in the calibration site images as weak supervision information to construct the geometric loss guiding the optimization of WESNet, so as to avoid its performance being limited by inaccurate supervision. We first collected the calibration site images over the calibration site filled with chessboard markers. With the known sizes of these markers, the 3D coordinates in the world coordinate system of each marker can be easily obtained. Based on the pairs of 3D world coordinates of corners and corresponding manually labelled 2D pixel coordinates in calibration site images, we form the re-projection loss to geometrically regress the output extrinsics from images collected under the same extrinsic configurations. To help understand, the sketch of the geometric loss is illustrated in Fig. 2.

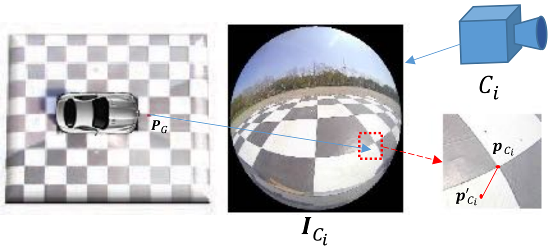


Fig. 2. Illustration of the geometric loss. For each selected corner $\boldsymbol{p}_{C_i}$ on the undistorted calibration site image, whose corresponding point in the ground coordinate system is $\boldsymbol{P}_G$, its geometric loss is constructed as the coordinates’ differences between $\boldsymbol{p}_{C_i}$ and the point projected from $\boldsymbol{P}_G$ to the undistorted image plane of C_i with estimated extrinsics by (2).

Given a selected corner on the calibration site, the relationship between its 3D coordinate $\boldsymbol{P}_G$ in the ground coordinate system $\boldsymbol{O}_G$ and its 2D pixel coordinate $\boldsymbol{p}_{C_i}$ on the undistorted fisheye image collected by camera C_i has been given in (2). Thus, with the yielded camera pose of WESNet, a corresponding projection $\boldsymbol{p}'_{C_i}$ can be generated and a loss term can be established, which is the distance between $\boldsymbol{p}_{C_i}$ and $\boldsymbol{p}'_{C_i}$. By summing up the error terms of all corners, we obtain the overall geometric loss of WESNet, which is given as,

$$Loss_{geo} = \sum_{i} \sum_{\boldsymbol{P}_G \in \mathcal{P}_i} \left\| norm(\boldsymbol{p}_{C_i} - \boldsymbol{p}'_{C_i}) \right\|_2$$

$$= \sum_{i} \sum_{\boldsymbol{P}_G \in \boldsymbol{\mathcal{P}}_i} \left\| norm(\boldsymbol{p}_{C_i} - \frac{1}{Z_{C_i}} \boldsymbol{K}_{C_i} \boldsymbol{T}_{C_i} \boldsymbol{P}_G) \right\|_2 \tag{9}$$

where i stands for the index of the camera in the SVS, ranging from one to four, $\boldsymbol{\mathcal{P}}_i$ stands for the set of all selected corners on the calibration site that can be seen by camera C_i, the function $norm(*)$ normalizes the re-projected pixel coordinates by dividing its coordinate on the corresponding axis with the width or the height of the image, respectively.

It's worth mentioning that the ground coordinate system should be determined manually. A common solution is to park the vehicle at an appropriate position over the calibration site to align the vehicle coordinate system and the ground one. From this perspective, except for labelling the training data in a weakly supervised manner, the geometric loss also offers an absolute reference to guarantee the convergence of the network. Specifically, by introducing the geometric loss, WESNet can learn to determine a specific ground coordinate system and to regress poses of different cameras in a unified reference system.

2) Orthogonal Loss: The rotation matrix consists of nine parameters but its DoF is only three, implying a strong internal constraint. Since it's difficult to solve the constrained optimization problem, we choose a relatively soft solution, that is, introducing the orthogonal loss to keep the constraint satisfied along with the training process. Motivated by [43], the orthogonal loss is defined as,

$$\begin{aligned} Loss_{org} = \sum_{i} \sum_{j=1}^{3} ((R^i_{j1})^2 + (R^i_{j2})^2 + (R^i_{j3})^2 - 1)^2 \\ + (R^i_{11}R^i_{21} + R^i_{12}R^i_{22} + R^i_{13}R^i_{23})^2 \\ + (R^i_{11}R^i_{31} + R^i_{12}R^i_{32} + R^i_{13}R^i_{33})^2 \\ + (R^i_{21}R^i_{31} + R^i_{22}R^i_{32} + R^i_{23}R^i_{33})^2 \end{aligned} \tag{10}$$

where R^i_{jk} is the element in the jth-row and the kth-column of the rotation matrix $\boldsymbol{R}_i$ of C_i.

3) Photometric Loss: Training only with the geometric loss, the network can converge to a barely satisfactory level. However, the results are still not accurate enough, and there will usually be obvious geometric misalignments in the synthesized surround-views. To further improve the performance of the network, we introduce an extra self-supervised photometric loss. The sketch of this photometric loss is illustrated in Fig. 3.

When the extrinsics are absolutely accurate, the grayscale values of the adjacent cameras' imaging pixels of the same point tend to be consistent. Based on such an assumption, given a point $\boldsymbol{p}_G$ on the surround-view in the common-view region of camera C_i and C_j, we define the corresponding photometric loss term $\varepsilon_{\boldsymbol{p}_G}$ as,

$$\varepsilon_{\boldsymbol{p}_G} = \boldsymbol{I}_{GC_i}(\boldsymbol{p}_G) - \rho_{ij} \boldsymbol{I}_{GC_j}(\boldsymbol{p}_G) \tag{11}$$

where $\boldsymbol{I}_{GC_i}$ and $\boldsymbol{I}_{GC_j}$ are bird's-eye-views of C_i and C_j, respectively. With the preliminaries given by (6), (11) can also be reformulated as,

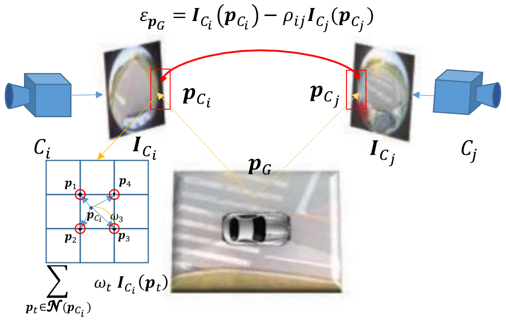

$\varepsilon_{\boldsymbol{p}_G} = \boldsymbol{I}_{C_i}(\boldsymbol{p}_{C_i}) - \rho_{ij}\boldsymbol{I}_{C_j}(\boldsymbol{p}_{C_j})$
$\boldsymbol{p}_{C_i}$
$\boldsymbol{p}_{C_j}$
C_i
$\boldsymbol{I}_{C_i}$
$\boldsymbol{p}_G$
$\boldsymbol{I}_{C_j}$
C_j
$\boldsymbol{p}_1$
$\boldsymbol{p}_4$
$\boldsymbol{p}_{C_i}$
ω_3
$\boldsymbol{p}_2$
$\boldsymbol{p}_3$
$\sum_{\boldsymbol{p}_t \in \boldsymbol{\mathcal{N}}(\boldsymbol{p}_{C_i})} \omega_t \boldsymbol{I}_{C_i}(\boldsymbol{p}_t)$

Fig. 3. Illustration of the photometric loss. It represents the grayscale difference of adjacent cameras' imaging pixels of the same point. It is introduced in the second stage of training to fine-tune the network, so as to synthesize seamless surround-views.

$$\begin{aligned} \varepsilon_{\boldsymbol{p}_G} = \boldsymbol{I}_{C_i}\left(\frac{1}{Z_{C_i}} \boldsymbol{K}_{C_i} \boldsymbol{T}_{C_i G} \boldsymbol{K}_G^{-1} \boldsymbol{p}_G\right) \\ - \rho_{ij} \boldsymbol{I}_{C_j}\left(\frac{1}{Z_{C_j}} \boldsymbol{K}_{C_j} \boldsymbol{T}_{C_j G} \boldsymbol{K}_G^{-1} \boldsymbol{p}_G\right) \end{aligned} \tag{12}$$

where ρ_{ij} is an exposure factor to weaken the negative impact brought by the discrepancy on lighting conditions and environmental reflections between different cameras. Actually, as discussed in [44], for an image taken of a physical object, except for the properties of the object itself, the imaging pixel values will also be determined by the exposure time, the vignette and the non-linear response function of the camera. Among them, the exposure time is the most important factor according to our experience. Thus, we model the exposure factor ρ_{ij} as,

$$\rho_{ij} = \frac{t_i}{t_j} \tag{13}$$

where t_i is the corresponding exposure time of $\boldsymbol{I}_{C_i}$ and t_j is that of $\boldsymbol{I}_{C_j}$. Even though the exposure time can't be obtained directly in general, the factor ρ_{ij} can be fitted as,

$$\rho_{ij} = \frac{\sum_{\boldsymbol{p_G} \in \boldsymbol{\mathcal{O}}_{ij}} \boldsymbol{I}_{GC_i}(\boldsymbol{p_G})}{\sum_{\boldsymbol{p_G} \in \boldsymbol{\mathcal{O}}_{ij}} \boldsymbol{I}_{GC_j}(\boldsymbol{p_G})} \tag{14}$$

where $\boldsymbol{\mathcal{O}}_{ij}$ is the set of all pixels in the common-view region of C_i and C_j on bird's-eye-view images.

To improve the robustness to outliers, we adopted the l_1 loss. In this way, by summing up the l_1-norm of photometric loss terms of all qualified points, the overall photometric loss can be obtained as,

$$Loss_{pho} = \sum_{(i,j) \in \boldsymbol{\mathcal{A}}_{ij}} \sum_{\boldsymbol{p}_G \in \boldsymbol{\mathcal{N}}_{ij}} |\varepsilon_{\boldsymbol{p}_G}| \tag{15}$$

where $\boldsymbol{\mathcal{A}}_{ij}$ is the set of all adjacent cameras' indices, and $\boldsymbol{\mathcal{N}}_{ij}$ is the set of all selected qualified pixels in the common-view region of C_i and C_j. More details regarding pixel selection can be found in Section IV-C.

C. Implementation Details

1) Pixel Selection: Our pixel selection strategy follows the criteria that each qualified point $\boldsymbol{p}_G$'s intensity gradient modulus $G(\boldsymbol{p}_G)$ should be large enough so as to keep the stability of the feature it can offer. Specifically,

$$G(\boldsymbol{p}_G) \geq 2G_{mean} + \sigma_g \tag{16}$$

where G_{mean} is the mean gradient modulus over $\mathcal{O}_{ij}$, and σ_g is the associated standard deviation.

Besides, as discussed in Section III, for any point $\boldsymbol{p}_G$ on the surround-view, it is assumed to be on the ground. However, some objects with non-negligible heights, such as pedestrians, lawns or curbs, may appear in the surround-view and break such a preliminary. For ease of representation, we call these objects as "mismatched objects" and the corresponding pixels as "mismatched pixels". Constructing the photometric loss with mismatched pixels may do harm to the final accuracy since such pixels do not follow the imaging model of the SVS. Thus, such pixels should be eliminated in the pixel selection approach. Motivated by [21], we adopted a color based strategy. Specifically, for any qualified point $\boldsymbol{p}_G$, the color discrepancy between $\boldsymbol{I}_{GC_i}(\boldsymbol{p}_G)$ and $\boldsymbol{I}_{GC_j}(\boldsymbol{p}_G)$ is supposed to be unobvious. Defining $\boldsymbol{I}^c_{GC_i}$ and $\boldsymbol{I}^c_{GC_j}$ as the maps of $\boldsymbol{I}_{GC_i}$ and $\boldsymbol{I}_{GC_j}$ of channel c, respectively, we measure the color discrepancy with the standard deviation of $\boldsymbol{p}_G$'s color ratios in different channels,

$$D_{color}\left(\boldsymbol{p}_G\right) = \sqrt{\frac{\sum_{c=1}^{n_c}\left(r_c\left(\boldsymbol{p}_G\right) - r_\mu\left(\boldsymbol{p}_G\right)\right)^2}{n_c}} \tag{17}$$

where n_c is the number of channels (normally 3) and $r_\mu(\boldsymbol{p}_G)$ returns the average of all $\boldsymbol{p}_G$'s color ratios. The color ratio $r_c(\boldsymbol{p}_G)$ is defined as,

$$r_c\left(\boldsymbol{p}_G\right) = \frac{\boldsymbol{I}^c_{GC_i}\left(\boldsymbol{p}_G\right)}{\boldsymbol{I}^c_{GC_j}\left(\boldsymbol{p}_G\right)} \tag{18}$$

For any qualified $\boldsymbol{p}_G$, it must satisfy,

$$D_{color}\left(\boldsymbol{p}_G\right) < D_{mean} - 2\sigma_d \tag{19}$$

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.

2) Derivatives of the Photometric Loss: During the back propagation of the network, the derivation of the aforementioned three types of losses to the yielded extrinsics is necessary. Different from the geometric loss and the orthogonal one, there is no perfect analytical solution of the derivatives of the photometric loss, thus some approximations are necessary. Take the derivative δ of the photometric loss term $\varepsilon_{\boldsymbol{p}_G}$ to the pose matrix $\boldsymbol{T}_{C_iG}$ as an example. The corresponding derivative δ can be decomposed into multiple simpler parts via the chain rule,

$$\begin{aligned}\delta &= \frac{\partial \varepsilon_{\boldsymbol{p}_G}}{\partial \boldsymbol{I}_{C_i}} \cdot \frac{\partial \boldsymbol{I}_{C_i}}{\partial \boldsymbol{p}^T_{C_i}} \cdot \frac{\partial \boldsymbol{p}_{C_i}}{\partial \boldsymbol{P}^T_{C_i}} \cdot \frac{\partial \boldsymbol{P}_{C_i}}{\partial \boldsymbol{T}_{C_iG}} \\ &= \frac{\partial \boldsymbol{I}_{C_i}}{\partial \boldsymbol{p}^T_{C_i}} \cdot \frac{\partial \boldsymbol{p}_{C_i}}{\partial \boldsymbol{P}^T_{C_i}} \cdot \frac{\partial \boldsymbol{P}_{C_i}}{\partial \boldsymbol{T}_{C_iG}}\end{aligned} \tag{20}$$

where $\boldsymbol{p}_{C_i}$ is the projection of $\boldsymbol{p}_G$ on the undistorted image $\boldsymbol{I}_{C_i}$ and $\boldsymbol{P}_{C_i}$ is the corresponding point in the camera coordinate system of C_i. The analytical solutions of the latter two terms can be derived easily, but not the first one. In [45], Irani *et al.* offered a general solution, that is utilizing the gradient of image intensities at $\boldsymbol{p}_{C_i}$ for approximation. In our implementations, to calculate the derivatives under the framework of neural network efficiently, every time after the forward propagation, we linearize the photometric loss term $\varepsilon_{\boldsymbol{p}_G}$ at the current projection $\hat{\boldsymbol{p}}_{C_i}$ with the differentiable bilinear sampling and $\boldsymbol{I}_{C_i}(\boldsymbol{p}_{C_i})$ is reformulated as,

$$\boldsymbol{I}_{C_i}(\boldsymbol{p}_{C_i}) = \sum_{\boldsymbol{p}_t \in \boldsymbol{\mathcal{N}}(\hat{\boldsymbol{p}}_{C_i})} \omega_t \cdot \boldsymbol{I}_{C_i}(\boldsymbol{p}_t) \tag{21}$$

where $\boldsymbol{\mathcal{N}}(\hat{\boldsymbol{p}}_{C_i})$ is the set of all neighboring points near $\hat{\boldsymbol{p}}_{C_i}$, and ω_t is linearly proportional to the spatial proximity between $\hat{\boldsymbol{p}}_{C_i}$ and $\boldsymbol{p}_t$. Thanks to the reformulation, the undifferentiable term $\boldsymbol{I}_{C_i}(\boldsymbol{p}_{C_i})$ is converted to a differentiable one, and the back propagation of WESNet can be efficiently conducted under the auto-differential framework.

V. Surround-View Dataset

Since there's no existing large-scale surround-view dataset containing original fisheye images, we collected our own dataset by an electric car equipped with four cameras mounted around, which is mainly composed of two parts, calibration site images and natural ones. It is worth emphasizing that both parts of our dataset are publicly available. The calibration site images were collected over the calibration site to provide weak supervision information, while the natural ones, which act as training and testing sets, were taken from natural scenes. For the relation between the chessboard corners in calibration site images and natural images, in the training phase, both kinds of data provide us with the necessary supervision signal. Specifically, chessboard corners offered us the required geometric supervision while natural images offered the photometric one. Hence, in the inference process, our network only takes the natural images as input to complete the extrinsics' estimation and has no dependence on the chessboard corners. And the original resolutions of all collected images are 1280×1080.

A. Calibration Site Images

As aforementioned, the calibration site images were taken over a calibration site. The calibration site is located on a flat field with 10×10 chessboard grids printed on it, and the size of each grid is $1m \times 1m$, as shown in Fig. 4. We parked the vehicle to a designated position where the midpoint of rear axle of wheels was five meters horizontally and six meters vertically from the upper left corner of the calibration site. Designating this midpoint as the origin, the world coordinate system was established and the 3D world coordinates of each chessboard corner could be easily obtained.

B. Natural Images

The natural images form our training and testing data. In practical application, to reduce manpower and time cost, we expect that the cameras can be calibrated while the vehicle drives on the normal ground rather than the specific calibration site, so we take natural images as the input of our network, which is

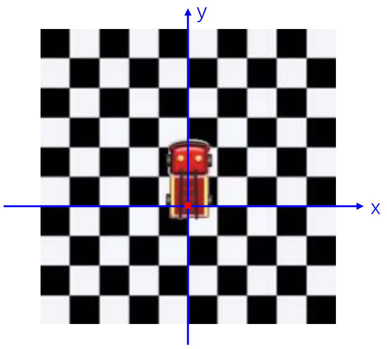


Fig. 4. Illustration of the calibration site and how to establish the world coordinate system. Over the site, calibration site images, which offer weakly supervised information, are collected.

TABLE I
QUANTITIES OF NATURAL IMAGES COLLECTED UNDER DIFFERENT ENVIRONMENTAL CONDITIONS

index	environment	specific type	numbers	total
V0	lane-line	——	6,873	6,873
V1	tile	square rectangle granite	1,599 2,001 312	3,912
V2	cement	——	526	526
V3	narrow	——	1,256	1,256
V4	sunlight	——	1,692	1,692
V5	parking-site	small medium large	736 1,486 2,597	4,819
***total*: 19,078**				

TABLE II
HARDWARE CONFIGURATIONS OF THE COMPUTATION PLATFORM

hardware	specification
CPU	Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40 GHz
GPU	NVIDIA TITAN Xp
Memory	32.0 GB
Motherboard	Z10PE-D16 Series

TABLE III
MAIN SOFTWARE CONFIGURATIONS OF THE COMPUTATION PLATFORM

software environment	version
Operating System	Ubuntu 18.04
GPU API	CUDA 10.1.243
GPU Driver	GeForce Driver 418.77

consistent with our actual application scenario. Moreover, the natural images themselves also provide a self-supervised photometric loss term, so that we can get the seamless surround-view image. We simulated the driving in reality and collected a large number of images in the natural environment on campus, namely natural images for short. The data we collected covers a variety of common environments, including the ground with different lane-lines and tiles, the cement road, the narrow path, and the road exposed to strong sunlight. In addition, due to the obvious difference between the underground environment and the above outdoor scenes, we also collected data from three underground parking-sites with different scales. Some typical examples of different categories are shown in Fig. 5 while the specific quantities of images are given in Table I.

C. Data Pre-Processing

Preparing for the training, the collected data needs to be pre-processed, including labelling and data augmentation. With respect to labelling, we recorded the pixel coordinates and the corresponding 3D world coordinates of the manually selected corners in the calibration site images as weakly supervised labels. For each frame, about 20$\sim$30 corners were chosen. It is worth noting that this is the only necessary manual operation in our scheme. For data augmentation, in reality, the extrinsics of the SVS equipped by different vehicles usually vary, thus the collected data should cover extrinsics as widely as possible. However, it is quite time- and labor-consuming to expand the diversity of extrinsics of the dataset by manually adjusting the camera poses repeatedly and then collect data under different extrinsic configurations for many times. Therefore, in our practice, the extrinsics of cameras were always fixed during the collection, while the homography transformation was applied to improve the richness of the extrinsics of collected data. Concretely, we applied the homography transformation to the normalized planes of undistorted images and then distorted them again to synthesize fisheye images corresponding to new extrinsics. Fig. 6 shows an example of the fisheye image before and after the augmentation.

In our implementation, the rotation disturbance was within ±0.2 rd (about ±11.5 degrees), with an interval of 0.05 rd while the translation disturbance was within ±0.2˜m, with an interval of 0.05m. Finally, the extrinsic settings were augmented into nine different types, and a dataset containing 171,702 groups of images (each group consists of four fisheye images collected synchronously) was established. Considering that the similarity between consecutive video frames is relatively high, we take the first 90% of video frames in V0 $\sim$ V5 settings as the training set and the last 10% as the testing set instead of using a random sampling strategy.

It is worth mentioning that we can transform the labeling information corresponding to the original calibration site images to generate that for augmented data instead of manually labeling it, so that the data augmentation is fully automatic.

VI. EXPERIMENTAL RESULTS

A. Experiment Setup

All experiments in this paper were conducted on the same desktop computer, and its detailed hardware and software configurations are shown in Table II and Table III, respectively. The training and evaluation codes of WESNet were implemented using PyTorch [46]. Before training, all the input images were resized to 512×512. During training, we initialized the weights of all neural network layers randomly and used Adam optimizer [47] with 10^{-4} as the initial learning rate. Finally, we trained our network with the batch size of 16 for 20 epochs.

B. Qualitative Experiment

1) Traits of the Methods: As we have reviewed in Section II, there are several studies in the literature that are relevant to WESNet. In order to understand the different characteristics of these

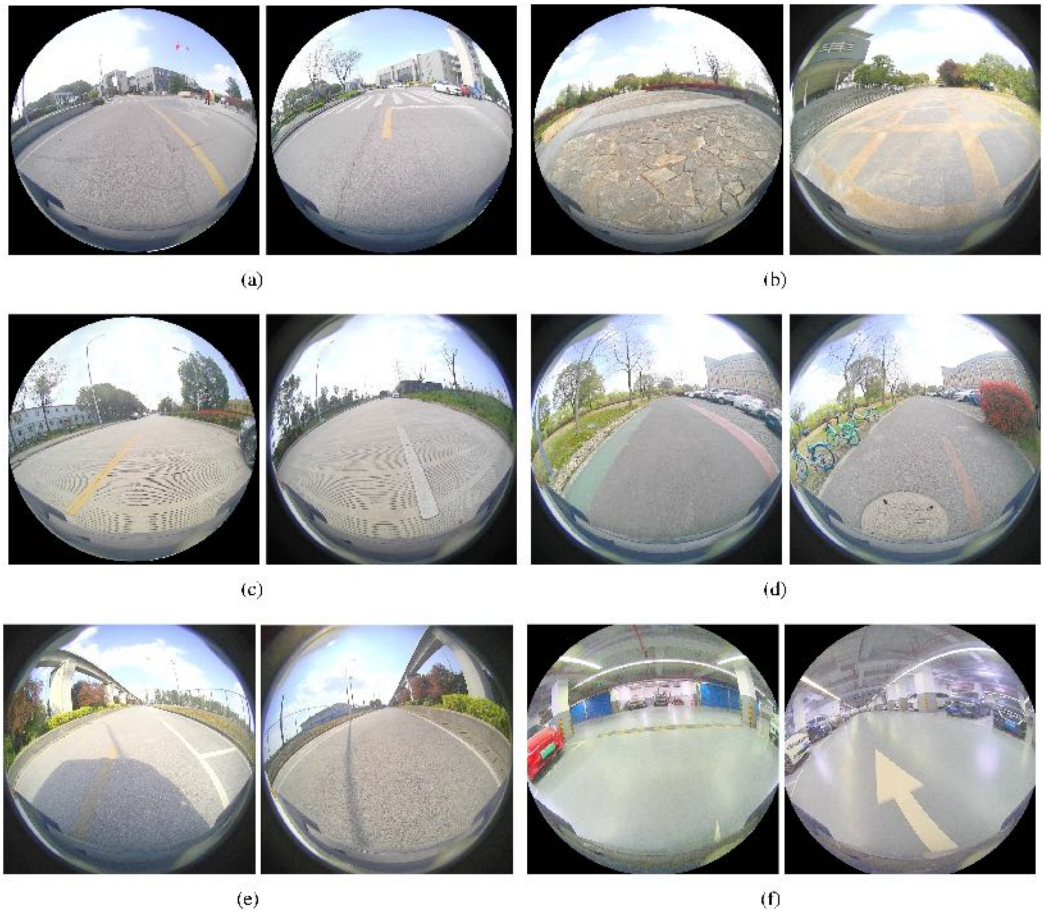


Fig. 5. Typical samples of natural images collected under different environments. The shown images are all captured by the front camera. From (a)∼(f), the samples are selected from groups V0∼V5, respectively.

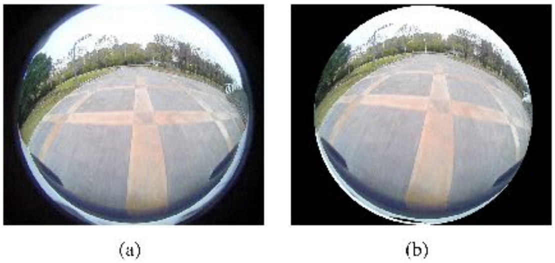


Fig. 6. An example of synthesizing fisheye images based on our proposed data augmentation pipeline. (a) is the original fisheye image, and (b) is the augmented result, which can represent the fisheye image under a new extrinsics' configuration.

TABLE IV
QUALITATIVE COMPARISON WITH RELATED METHODS

method	prior	automatic?	objects assisting calibration
Liu *et al.* [10]	×	manual	tiles
Hedi *et al.* [11]	×	manual	calibration site
Zhang *et al.* [12]	×	manual	calibration chart
Shao *et al.* [13]	×	manual	chessboard
Gao *et al.* [14]	×	manual	calibration site
Zhang *et al.* [15]	×	manual	chessboard
Zhao *et al.* [16]	×	self-calibration	ground lane
Choi *et al.* [17]	×	self-calibration	ground lane
Heng *et al.* [18]	×	self-calibration	SLAM system
Heng *et al.* [19]	×	self-calibration	SLAM system
Liu *et al.* [20]	√	self-calibration	natural images
OECS [21]	√	self-calibration	natural images
ROECS [22]	√	self-calibration	natural images
WESNet	×	self-calibration	natural images

methods more clearly, in Table IV we compare them in three aspects: 1) Does it require the prior information of extrinsics? 2) Does it belong to manual calibration schemes or self-calibration ones? and 3) What kind of objects assisting calibration does it rely on? It can be seen that Liu *et al.*'s method [20], OECS [21], ROECS [22] and our scheme, WESNet, can yield the extrinsics of the SVS just from natural images. Among them, WESNet does not need any prior information of the extrinsics, implying a wider application scope. It's worth mentioning that Heng *et al.*'s schemes [18] and [19] also rely on natural images. However, as mentioned in Section II, since a large quantity of frames are required for their SLAM systems to converge, these two schemes are quite cumbersome. By contrast, our WESNet takes a single group of frames collected by the SVS synchronously as the input, corroborating its lightweightness.

2) Typical Samples of Synthesized Surround-Views: In order to qualitatively examine the performance of WESNet, we compared it with three representative competitors, including the manual calibration scheme [13] and the self-calibration one,

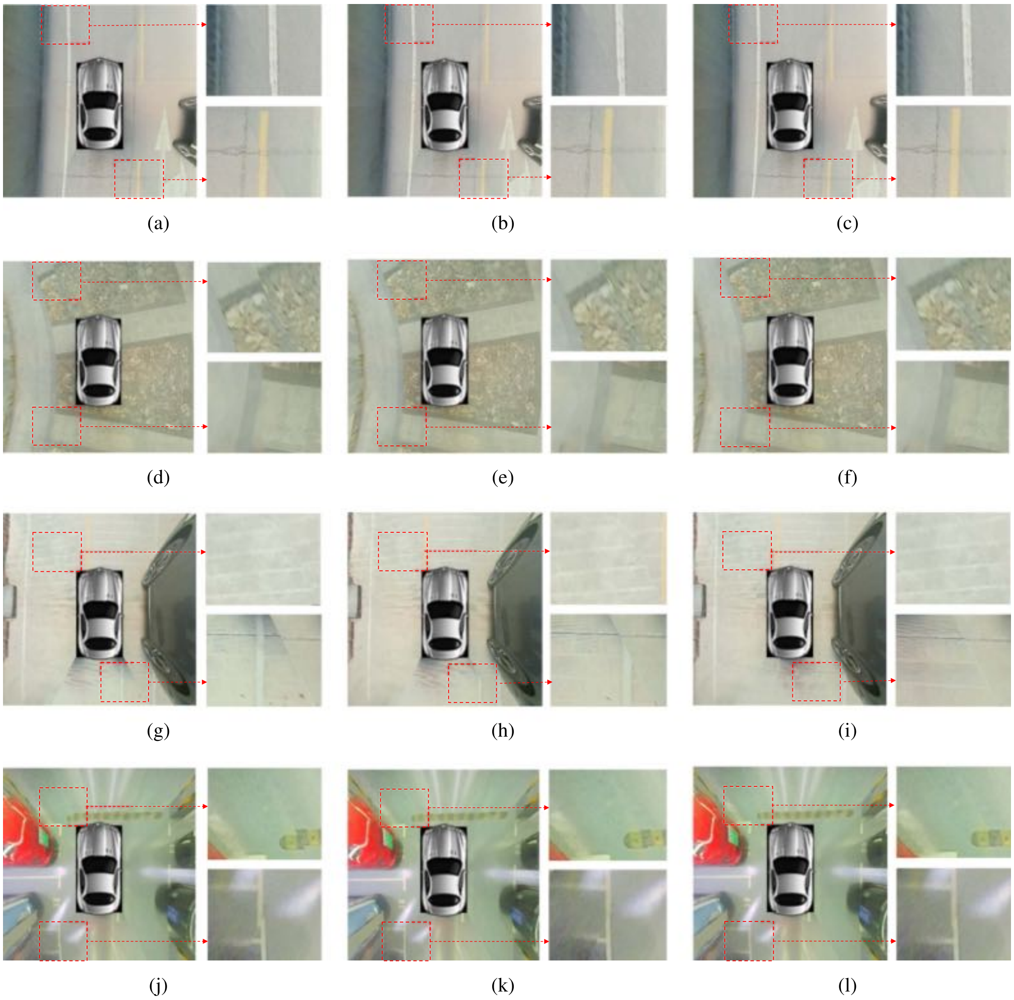


Fig. 7. Comparisons of the surround-views synthesized with yielded extrinsics of WESNet and two representative competitors. In each row, from left to right the surround-views are generated by Shao *et al.*'s scheme [13], OECS [21], and WESNet, respectively.

OECS [21] on our testing set. Some typical samples of the surround-views synthesized with the yielded extrinsics of our scheme and these two competitors are shown in Fig. 7. From Fig. 7, it can be clearly seen that there are often geometric misalignments of different severity in the surround-views synthesized by Shao *et al.*'s scheme [13]. The reason accounting for this phenomenon is that with Shao *et al.*'s scheme [13], the offline calibration is only conducted over the calibration site, and is not adaptive to some naturally occurring interference factors, such as the change of the tire pressure or the vehicle load. Apart from that, such a scheme utilizes only a limited number of corners during the calibration, and hence it suffers from the sensitivity to labelling impreciseness of selected corners. In contrast, OECS [21] works well on the ideal road surface with clear textures, such as Fig. 7(b). However, it only uses a single frame and strongly relies on the imaging hypothesis of the SVSs. When the road surface is uneven (as shown in Fig. 7(e) and (h)) or there are obvious objects with non-negligible heights (as shown in Fig. 7(k)) in the surround-view, it may underperform, implying its unsatisfactory robustness. For WESNet, as aforementioned, since it can learn richer and more general features from a large number of data thanks to the superiority of the learning mechanism, in most cases, it performs significantly better than those two competitors. Fig. 7 also supports our claim.

C. Quantitative Experiment

1) Metrics: To measure the accuracy of the extrinsics estimated by the compared methods, four metrics were employed for evaluation, two absolute values and two relative ones. Specifically, they are:

a) $\mathcal{E}_{ARE}$, the absolute re-projection error. It is the distance between the projections of 3D corners from the ground coordinate system to the undistorted image plane with the output extrinsics and the pixel coordinates of the corresponding 2D points in undistorted calibration site images, which is given by (9).

b) $\mathcal{E}_{APE}$, the absolute photometric error. It is defined as (15) to represent the grayscale difference between the corresponding keypoints, which lay in the common-view regions of the SVS.

c) $\mathcal{E}_{RRE}$, the relative re-projection error. As the accuracy of offline calibration methods is generally satisfactory now,

TABLE V
QUANTITATIVE COMPARISON WITH REPRESENTATIVE COMPETITORS

group index	method	$\mathcal{E}_{ARE}$	$\mathcal{E}_{APE}$	$\mathcal{E}_{RRE}$	$\mathcal{E}_{RPE}$
V0	Shao *et al.* [13]	8.2	29.5	0	0
	OECS [21]	4.5	15.7	-3.7	-13.8
	ROECS [22]	4.8	14.9	-3.4	-14.6
	WESNet	**3.2**	**13.1**	**-5.0**	**-16.4**
V1	Shao *et al.* [13]	8.2	30.6	0	0
	OECS [21]	6.6	22.0	-1.6	-8.6
	ROECS [22]	5.8	21.9	-2.4	-8.7
	WESNet	**3.1**	**21.4**	**-5.1**	**-9.2**
V2	Shao *et al.* [13]	8.2	31.4	0	0
	OECS [21]	5.6	**19.6**	-2.6	**-11.8**
	ROECS [22]	4.7	23.2	-3.5	-8.2
	WESNet	**3.6**	20.3	**-4.6**	-11.1
V3	Shao *et al.* [13]	8.2	33.2	0	0
	OECS [21]	5.5	27.6	-2.7	-5.6
	ROECS [22]	5.1	27.5	-3.1	-5.7
	WESNet	**4.5**	**27.2**	**-3.7**	**-6.0**
V4	Shao *et al.* [13]	8.2	40.2	0	0
	OECS [21]	3.9	27.1	-4.3	-13.1
	ROECS [22]	4.3	26.8	-3.9	-13.4
	WESNet	**3.5**	**26.2**	**-4.7**	**-14.0**
V5	Shao *et al.* [13]	8.2	26.5	0	0
	OECS [21]	4.2	17.6	-4.0	-8.9
	ROECS [22]	3.7	17.0	-4.5	-9.5
	WESNet	**3.4**	**16.9**	**-4.8**	**-9.6**
weighted average	Shao *et al.* [13]	8.2	30.6	0	0
	OECS [21]	5.0	19.6	-3.2	-11.0
	ROECS [22]	4.8	19.2	-3.4	-11.4
	WESNet	**3.4**	**18.3**	**-4.8**	**-12.3**

TABLE VI
TIME COSTS AND COMPUTATION PLATFORMS OF COMPARED METHODS

	Shao *et al.* [13]	OECS [21]	ROECS [22]	WESNet
time cost	About $2min$	$2.0s$	$23.3s$	$32.6ms$
platform	Manual operation + CPU	CPU	CPU	GPU

we take the absolute re-projection error over surround-views synthesized with camera poses offline calibrated by Shao *et al.*'s scheme [13] as the baseline, denoted by $\mathcal{E}^{base}_{ARE}$. Given the absolute re-projection error of an evaluated method $\mathcal{E}'_{ARE}$, the relative re-projection error of this method can be defined as $\mathcal{E}_{RRE} = \mathcal{E}'_{ARE} - \mathcal{E}^{base}_{ARE}$.

d) $\mathcal{E}_{RPE}$, the relative photometric error. Similar to $\mathcal{E}_{RRE}$, denoting the absolute photometric error of the baseline and that of a compared method by $\mathcal{E}^{base}_{APE}$ and $\mathcal{E}'_{APE}$, respectively, the relative photometric error can be calculated by $\mathcal{E}_{RPE} = \mathcal{E}'_{APE} - \mathcal{E}^{base}_{APE}$.

It is worth mentioning that if the values of the two latter metrics are negative, it implies that the tested method is better than the baseline.

2) Accuracy and Generalization: Over each group of data collected from different environments, we tested the compared methods with the metrics aforementioned, and reported the experimental results in Table V. The best results are highlighted in bold. Additionally, in order to make a fair and comprehensive comparison, for each performance metric, Table V also tabulates the weighted-average errors of all methods over the whole data groups. The weight assigned to each group depends linearly on the number of fisheye images contained in that group. From Table V, it can be seen that compared with all counterparts, WESNet shows an overwhelming performance on nearly all data groups. In the group V2 composed of samples of cement roads where WESNet is not the best, its results nearly match OECS's which is the best one. However, OECS's lower photometric errors on group V2 in fact are due to the "overfitting" caused by the low-texture feature of group V2. Specifically, OECS takes only the photometric error as the guidance to correct camera poses. However, in low-texture environment, the photometric error is mainly determined by noise rather than inaccurate poses of cameras. In this case, OECS can yield extrinsics with low photometric errors but their accuracy can't be guaranteed. To conclude, the excellent accuracy of extrinsics estimation and the generalization capability of WESNet has been nicely demonstrated.

3) Robustness to Intrinsic Disturbance: To evaluate the robustness of WESNet to the accuracy of the intrinsics, we first introduced varying degrees of disturbance to the offline calibrated intrinsics of each camera in the SVS. The disturbance can be represented as an intrinsics' disturbance factor d and we added it to the focal length of the camera. Accordingly, the disturbed intrinsic matrix $K^d_{C_i}$ of camera C_i can be expressed as,

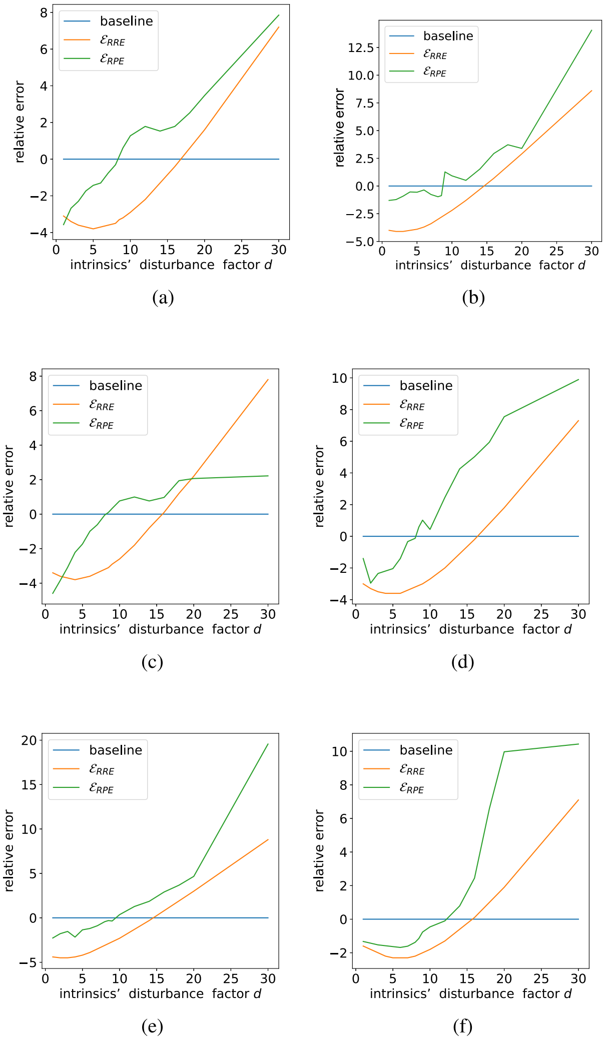
$$K^d_{C_i} = \begin{bmatrix} f_x + d & 0 & c_x \\ 0 & f_y + d & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{22}$$

Then under different d's settings and environmental conditions we ran our scheme and recorded the corresponding $\mathcal{E}_{RRE}$s and $\mathcal{E}_{RPE}$s. The relationship between $\mathcal{E}_{RRE}$s (or $\mathcal{E}_{RPE}$s) and the settings of d is shown in Fig. 8. The figure illustrates that, as long as d is lower than 8 pixels, the camera poses estimated by WESNet are always more accurate than the offline calibrated results. Based on our experience, the camera's focal length variation caused by the natural collisions or bumps won't exceed 5 pixels in general. Therefore, it can be concluded that WESNet is robust to the variations of intrinsics.

4) Time Cost Analysis: Following a weakly supervised framework, the repetitive workload in the calibration process is greatly reduced, and accordingly WESNet shows the speed performance far exceeding that of other competitors. To support our claim, we summarized the time costs to complete the whole calibration process of evaluated methods, and the results and platforms are reported in Table VI. From this table, it can be seen that as a representative manual solution, Shao *et al.*'s method [13] takes about two minutes to finish the task, which confirms our claim that manual schemes are usually cumbersome. OECS [21] and ROECS [22] can effectively correct imprecise extrinsics of the SVS in an online manner, but a few seconds are still required. Compared with them, as a lightweight CNN, WESNet can run on GPU and regress the extrinsics end-to-end with a total time cost of about 32.6ms, demonstrating its superior efficiency.

D. Ablation Study of Loss Terms

We demonstrate how loss terms in our framework affect the results by comparing WESNet with two networks trained with different loss terms on our dataset. Specifically, these networks are 1) **GeoNet**: The network trained with $Loss_{geo}$ only; 2) **PhoNet**: The network trained with $Loss_{pho}$ only. Table VII provides the



(a) (b)

(c) (d)

(e) (f)

Fig. 8. $\mathcal{E}_{RRE}$s and $\mathcal{E}_{RPE}$s under different disturbance factor d's settings. From (a)∼(f), the evaluations were conducted on groups V0∼V5, respectively. The relationship between $\mathcal{E}_{RRE}$s and different settings of d is shown as the orange curve while that of $\mathcal{E}_{RPE}$s is shown as the green curve. The blue line is the offline baseline.

TABLE VII
PERFORMANCE OF NETWORKS TRAINED WITH VARIOUS COMBINATIONS OF LOSS TERMS ON OUR DATASET

group index	method	$\mathcal{E}_{ARE}$	$\mathcal{E}_{APE}$	$\mathcal{E}_{RRE}$	$\mathcal{E}_{RPE}$
V0	GeoNet	3.4	16.6	-4.8	-12.9
	PhoNet	6955.2	42.0	6947.0	12.5
	WESNet	**3.2**	**13.1**	**-5.0**	**-16.4**
V1	GeoNet	3.3	23.5	-4.9	-7.1
	PhoNet	2430.0	53.4	2421.8	22.8
	WESNet	**3.1**	**21.4**	**-5.1**	**-9.2**
V2	GeoNet	4.9	21.8	-3.3	-9.6
	PhoNet	4201.3	42.6	4193.1	11.2
	WESNet	**3.6**	**20.3**	**-4.6**	**-11.1**
V3	GeoNet	5.1	30.7	-3.1	-2.5
	PhoNet	15,504.5	33.4	15,496.3	0.2
	WESNet	**4.5**	**27.2**	**-3.7**	**-6.0**
V4	GeoNet	3.8	34.3	-4.4	-5.9
	PhoNet	12,772.1	28.7	12,763.9	-11.5
	WESNet	**3.5**	**26.2**	**-4.7**	**-14.0**
V5	GeoNet	4.1	21.8	-4.1	-4.7
	PhoNet	4622.4	27.6	4614.2	1.1
	WESNet	**3.4**	**16.9**	**-4.8**	**-9.6**
weighted average	GeoNet	3.8	22.3	-4.4	-8.3
	PhoNet	6422.6	39.4	6414.4	8.8
	WESNet	**3.4**	**18.3**	**-4.8**	**-12.3**

TABLE VIII
QUANTITATIVE COMPARISON OF GEONET WITH WESNET ON CALIBRATION SITE IMAGES

method	$\mathcal{E}_{ARE}$	$\mathcal{E}_{APE}$	$\mathcal{E}_{RRE}$	$\mathcal{E}_{RPE}$
Shao *et al.* [13]	6.7	25.3	0	0
GeoNet	5.3	22.6	-1.4	-2.7
WESNet	**3.0**	**19.5**	**-3.7**	**-5.8**

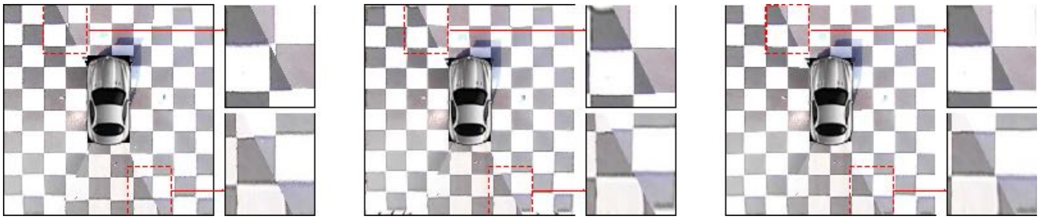
results of the evaluated variants in terms of the four metrics mentioned in Section VI-C on our dataset covering different environments. For each metric, the best result is highlighted in bold.

From the results presented in Table VII, we make the following observations. First, highly relying on the geometric loss, **GeoNet** achieves performance comparable with WESNet in $\mathcal{E}_{ARE}$ and $\mathcal{E}_{RRE}$. However, its evaluation results in $\mathcal{E}_{APE}$ and $\mathcal{E}_{RPE}$ are unsatisfactory, implying that the extrinsics are not accurate enough for synthesizing a seamless surround-view. Second, training with the photometric loss only, **PhoNet** delivers significantly poor performance in all metrics. Indeed, it can't converge at all. The underlying reason is that the approximation given in (20) guiding the optimization in the back propagation works well when the extrinsics approach the accurate values, otherwise this approximation is unreasonable. This also corroborates our claim that the geometric loss can help with the fast convergence. Third, thanks to our two-stage framework, WESNet exhibits clear performance advantages over all counterparts with all errors reduced to a sufficiently low level. Under this framework, the geometric loss is utilized in the first stage to ensure relatively accurate extrinsics. Then the photometric loss is introduced to fine-tune the network, which indeed improves the accuracy of extrinsics to some extent according to the evaluation results in Table VII. These results lead us to express the belief that the loss function of WESNet is well designed and both the geometric loss and the photometric loss play essential and effective roles in it.

Since **GeoNet** could generate good results on natural images, to show the comparison results more clearly, we have also conducted quantitative and qualitative experiments to see how **GeoNet** works on calibration site images compared with our WESNet, and the experimental results are shown in Table VIII and Fig. 9, respectively. The results of Shao *et al.* [13] (the baseline) are also provided for reference. It can be seen that **GeoNet**'s performance is comparable to the baseline since their optimizations both mainly rely on re-projection errors. In addition, our WESNet outperforms **GeoNet** both quantitatively and qualitatively on calibration site images, further demonstrating the effectiveness of the loss function of WESNet.

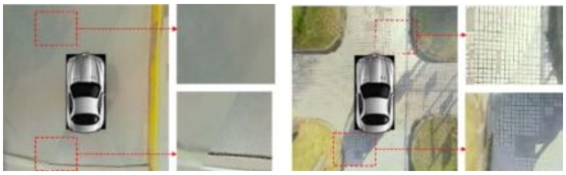
E. Failure Case Analysis

By observing and analyzing the experimental results, we found that the textures of the ground have an obvious influence on WESNet's performance. On the one hand, when the ground texture is weak, the information contained in the image is relatively scarce. In this case, the impact of noise will be more notable, and WESNet will not be able to extract high-quality features in deep level, resulting in poor performance as shown in Fig. 10(a). On the other hand, when the ground is filled with



(a) (b) (c)

Fig. 9. Qualitative comparison of surround-views synthesized on calibration site images with extrinsics yielded by Shao *et al.* [13], **GeoNet** and WESNet, respectively.



(a) (b)

Fig. 10. Examples of failure cases. The synthesized surround-views are shown on the left while enlarged local regions are on the right.

repetitive fine-grained textures, WESNet may also fail as shown in Fig. 10(b). The underlying reason is that our training framework, especially in its second stage with the photometric loss, tends to align the adjacent views in the SVS. In the failure case as illustrated in Fig. 10(b), the alignment may be "mismatched," implying the failure of the extrinsics estimation. Thus, it should be emphasized that in order to make our WESNet work successfully, the vehicle needs to be parked on flat ground with clearly observable and coarse grained textures.

VII. Conclusion

In this paper, we studied a practical problem, extrinsic self-calibration for the surround-view system, emerging from the field of ADAS. Following a weakly supervised framework, we proposed a novel learning-based solution, namely WESNet. Taking the original fisheye images captured by cameras in the SVS as the input, it can yield extrinsics end-to-end. During training, we first optimize the network fully based on the weakly supervised geometric loss for fast convergence, and then the self-supervised photometric loss is introduced to further fine-tune the network. With the two-stage training, seamless surround-views can be synthesized with the yielded extrinsics. An outstanding merit of WESNet is that, since the only required input is a single group of natural fisheye images and the forward propagation of the network can be completed within milliseconds, it does not require additional apparatuses or calibration sites and can be easily applied in the online manner. As long as the vehicle is driving on a normal flat road with relatively rich textures, WESNet will work. Besides, to facilitate the study of the extrinsic calibration or other surround-view based computer vision tasks, a surround-view dataset comprising 19,078 groups of surround-views and the associated original fisheye images in high resolution was also collected where typical types of environments were covered. Though WESNet can work well in most cases, its performance is still not satisfied when working in environments having low textures or strong texture repeatability. Thus we will continue to devote efforts in this area.

References

[1] L. Duan and F. Chen, "The future of advanced driving assistance system development in China," in *Proc. IEEE Int. Conf. Veh. Electron. Saf.*, 2011, pp. 238–243.

[2] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ., 2003.

[3] R. Klette, A. Koschan, and K. Schluns, *Computer Vision: Three-Dimensional Data From Images*. Singapore: Springer, 1998.

[4] L. Li, L. Zhang, X. Li, X. Liu, Y. Shen, and L. Xiong, "Vision-based parking-slot detection: A benchmark and a learning-based approach," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2017, pp. 649–654.

[5] L. Zhang, J. Huang, X. Li, and L. Xiong, "Vision-based parking-slot detection: A. DCNN-based approach and a large-scale benchmark dataset," *IEEE Trans. Image Process.*, vol. 27, no. 11, pp. 5350–5364, Nov. 2018.

[6] C. Lin and M. Wang, "A vision based top-view transformation model for a vehicle parking assistant," *Sensors*, vol. 12, no. 4, pp. 4431–4446, 2012.

[7] J. Xu, G. Chen, and M. Xie, "Vision-guided automatic parking for smart car," in *Proc. IEEE Intell. Veh. Symp.*, 2000, pp. 725–730.

[8] M. Gressmann, G. Palm, and O. Löhlein, "Surround view pedestrian detection using heterogeneous classifier cascades," in *Proc. Int. IEEE Conf. Intell. Transp. Syst.*, 2011, pp. 1317–1324.

[9] C. Hou, H. Ai, and S. Lao, "Multiview pedestrian detection based on vector boosting," in *Proc. Asian Conf. Comput. Vis.*, 2007, pp. 18–22.

[10] Y. Liu, K. Lin, and Y. Chen, "Bird's-eye view vision system for vehicle surrounding monitoring," in *Proc. Int. Workshop Robot Vis.*, 2008, pp. 207–218.

[11] A. Hedi and S. Lonari, "A system for vehicle surround view," in *Proc. IFAC Symp. Robot Control Int. Federation Autom. Control*, 2012, pp. 120–125.

[12] B. Zhang *et al.*, "A surround view camera solution for embedded systems," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2014, pp. 676–681.

[13] X. Shao, X. Liu, L. Zhang, S. Zhao, Y. Shen, and Y. Yang, "Revisit surround-view camera system calibration," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2019, pp. 1486–1491.

[14] Y. Gao, C. Lin, Y. Zhao, X. Wang, S. Wei, and Q. Huang, "3-D surround view for advanced driver assistance systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 320–328, Jan. 2018.

[15] L. Zhang, J. Chen, D. Liu, Y. Shen, and S. Zhao, "Seamless 3D surround view with a novel burger model," in *Proc. IEEE Int. Conf. Image Process.*, 2019, pp. 4150–4154.

[16] K. Zhao, U. Iurgel, M. Meuter, and J. Pauli, "An automatic online camera calibration system for vehicular applications," in *Proc. Int. IEEE Conf. Intell. Transp. Syst.*, 2014, pp. 1490–1492.

[17] K. Choi, H. Jung, and J. Suhr, "Automatic calibration of an around view monitor system exploiting lane markings," *Sensors*, vol. 18, no. 9, pp. 2956–2982, 2018.

[18] L. Heng, B. Li, and M. Pollefeys, "CamOdoCal: Automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 1793–1800.

[19] L. Heng, M. Bürki, G. Lee, P. Furgale, R. Siegwart, and M. Pollefeys, "Infrastructure-based calibration of a multi-camera rig," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2014, pp. 4912–4919.

[20] X. Liu, L. Zhang, Y. Shen, S. Zhang, and S. Zhao, “Online camera pose optimization for the surround-view system,” in *Proc. ACM Int. Conf. Multimedia*, 2019, pp. 383–391.

[21] T. Zhang, L. Zhang, Y. Shen, Y. Ma, S. Zhao, and Y. Zhou, “OECS: Towards online extrinsics correction for the surround-view system,” in *Proc. IEEE Int. Conf. Multimedia Expo*, 2020, pp. 1–6.

[22] T. Zhang, N. Zhao, Y. Shen, X. Shao, L. Zhang, and Y. Zhou, “ROECS: A robust semi-direct pipeline towards online extrinsics correction of the surround-view system,” in *Proc. ACM Int. Conf. Multimedia*, 2021, pp. 3153–3161.

[23] C. Harris and M. Stephens, “A combined corner and edge detector,” in *Proc. Alvey Vis. Conf.*, 1988, pp. 147–151.

[24] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, “BRIEF: Binary robust independent elementary features,” in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 778–792.

[25] R. Battiti, “First- and second-order methods for learning: Between steepest descent and Newton’s method,” *Neural Comput.*, vol. 4, no. 2, pp. 141–166, 1992.

[26] X. Yang, Z. Yuan, D. Zhu, C. Chi, K. Li, and C. Liao, “Robust and efficient RGB-D SLAM in dynamic environments,” *IEEE Trans. Multimedia*, vol. 23, pp. 4208–4219, 2021.

[27] X. Gong, Y. Liu, Q. Wu, J. Huang, H. Zong, and J. Wang, “An accurate, robust visual odometry and detail-preserving reconstruction system,” *IEEE Trans. Multimedia*, vol. 23, pp. 2820–2832, 2021.

[28] E. Izquierdo, “Efficient and accurate image based camera registration,” *IEEE Trans. Multimedia*, vol. 5, no. 3, pp. 293–302, 2003.

[29] Y. Yu, K. Wong, S. Or, and J. Chen, “Controlling virtual cameras based on a robust model-free pose acquisition technique,” *IEEE Trans. Multimedia*, vol. 11, no. 1, pp. 184–190, Sep. 2009.

[30] M. Lourakis, “Sparse non-linear least squares optimization for geometric vision,” in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 43–56.

[31] J. Moré, “The Levenberg-Marquardt algorithm: Implementation and theory,” in *Numerical Analysis* G. A. Watson Eds., Berlin, Germany: Springer, 1978, pp. 105–116.

[32] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[33] H. Bay, T. Tuytelaars, and L. Gool, “SURF: Speeded up robust features,” in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.

[34] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: An efficient alternative to SIFT or SURF,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2564–2571.

[35] S. Workman, C. Greenwell, M. Zhai, R. Baltenberger, and N. Jacobs, “DeepFocal: A method for direct focal length estimations,” in *Proc. IEEE Int. Conf. Image Process.*, 2015, pp. 1369–1373.

[36] M. Giering, V. Venugopalan, and K. Reddy, “Multi-modal sensor registration for vehicle perception via deep neural networks,” in *Proc. High Perform. Extreme Comput. Conf.*, 2015, pp. 1–6.

[37] N. Schneider, F. Piewak, C. Stiller, and U. Franke, “RegNet: Multi-modal sensor registration using deep neural networks,” in *Proc. IEEE Intell. Veh. Symp.*, 2017, pp. 1803–1810.

[38] G. Iyer, R. Ram, J. Murthy, and K. Krishna, “CalibNet: Geometrically supervised extrinsic calibration using 3D spatial transformer networks,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 1110–1117.

[39] Z. Zhang, “Flexible camera calibration by viewing a plane from unknown orientations,” in *Proc. Int. Conf. Comput. Vis.*, 1999, pp. 666–673.

[40] F. Du and M. Brady, “Self-calibration of the intrinsic parameters of cameras for active vision systems,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1993, pp. 477–482.

[41] H. Zhu, J. Yang, and Z. Liu, “Fisheye camera calibration with two pairs of vanishing points,” in *Proc. Int. Conf. Inf. Technol. Softw. Eng.*, 2009, pp. 321–324.

[42] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[43] M. Ahmed, E. Hemayed, and A. Farag, “Neurocalibration: A neural network that can tell camera calibration parameters,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 1999, pp. 463–468.

[44] J. Engel, V. Koltun, and D. Cremers, “Direct sparse odometry,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 611–625, Mar. 2018.

[45] M. Irani and P. Anandan, “About direct methods,” in *Proc. Int. Workshop Vis. Algorithms*, 1999, pp. 267–277.

[46] A. Paszke *et al.*, “PyTorch: An imperative style, high-performance deep learning library,” 2019. [Online]. Available: https://arxiv.org/pdf/1912.01703

[47] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

Yang Chen received the B.S. degree in 2020 from the School of Software Engineering, Tongji University, Shanghai, China, where she is currently working toward the Ph.D. degree. Her research interests include SLAM systems, computer vision, and machine learning.

Lin Zhang (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees from the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2003 and 2006, respectively, and the Ph.D. degree from the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, in 2011. From March 2011 to August 2011, he was a Research Associate with the Department of Computing, The Hong Kong Polytechnic University. In August 2011, he joined the School of Software Engineering, Tongji University, Shanghai, China, where he is currently a Full Professor. His current research interests include environment perception of intelligent vehicle, pattern recognition, computer vision, and perceptual image/video quality assessment. He is an Associate Editor for the IEEE ROBOTICS AND AUTOMATION LETTERS, and *Journal of Visual Communication and Image Representation*. He was awarded as a Young Scholar of Changjiang Scholars Program, Ministry of Education, China.

Ying Shen received the B.S. and M.S. degrees from Software School, Shanghai Jiao Tong University, Shanghai, China, in 2006 and 2009, respectively, and the Ph.D. degree from the Department of Computer Science, City University of Hong Kong, Hong Kong, in 2012. In 2013, she joined the School of Software Engineering, Tongji University, Shanghai, China, where she is currently an Associate Professor. Her research interests include bioinformatics and pattern recognition.

Brian Nlong Zhao (Student Member, IEEE) is an Undergraduate Student with the University of Southern California, Los Angeles, CA, USA, majoring in computer engineering & computer science and applied mathematics. His research interests include machine learning and computer vision.

Yicong Zhou (Senior Member, IEEE) received the B.S. degree in electrical engineering from Hunan University, Changsha, China, and the M.S. and Ph.D. degrees in electrical engineering from Tufts University, Medford, MA, USA. He is currently a Full Professor and the Director of the Vision and Image Processing Laboratory with the Department of Computer and Information Science, University of Macau, Macau, China. His research interests include chaotic systems, multimedia security, computer vision, and machine learning. He is an Associate Editor for the *Neurocomputing*,*Journal of Visual Communication and Image Representation*, and*Signal Processing: Image Communication.* He is a Co-Chair of the Technical Committee on Cognitive Computing in the IEEE Systems, Man, and Cybernetics Society. He is a Senior Member of the International Society for Optical Engineering. He was the recipient of the Third Price of Macau Natural Science Award in 2014.