Rolling Shutter Camera Self-Calibration

Yongcong Zhang[1,2,*], Navid Rabbani[1,*], Bangyan Liao[3], Chengbo Wang[4], Yizhen Lao[4], and Adrien Bartoli[1,2,†]

[1]DIA2M, DRCI, CHU Clermont-Ferrand, France
[2]SURGAR, Surgical Augmented Reality, Clermont-Ferrand, France
[3]School of Engineering, Westlake University, China
[4]School of Design, Hunan University, China

Abstract. Rolling shutter (RS) cameras are widely used in consumer devices, but their row-wise exposure causes distortions under motion, making geometric 3D vision problems dependent on both camera intrinsics and readout time ratio. Existing RS calibration methods rely on calibration targets or specialised hardware, limiting their use in unconstrained settings. We present the first *self*-calibration method for RS cameras that directly estimates camera intrinsics and the readout time ratio from image sequences, without requiring calibration targets. The method is implemented as a self-calibrating bundle adjustment (BA), which critically depends on the RS imaging model. We combine two known complementary models. The first formulates RS imaging as continuous-time trajectory estimation under a row-wise pose representation. The second interprets RS images as temporally distorted global shutter (GS) images and requires to estimate correction fields. The combination is non-trivial and results in a unified dual-projection model, in which each 3D point is simultaneously constrained at both row-dependent and reference timestamps along a shared continuous trajectory, enforcing stronger geometric and temporal consistency. Extensive simulations analyse the applicability of several implementations under varying conditions, and real data experiments demonstrate the accuracy, robustness, and practical effectiveness of the proposed approach.

Keywords: Rolling Shutter · Self-Calibration · Bundle Adjustment

1 Introduction

RS cameras are widely used in modern consumer imaging devices due to their low cost, low power consumption, and high frame rate [17, 32]. Unlike GS cameras, which capture the entire image at a single instant, RS cameras expose and read out sequentially in a row-wise manner. As a consequence, different image rows correspond to slightly different time points. Under camera or scene motion, this sequential readout introduces characteristic geometric distortions such as skew,

* Equal contribution
† Corresponding author: `adrien.bartoli@gmail.com`
Project page: https://github.com/Yongcong-Zhang/RSSC

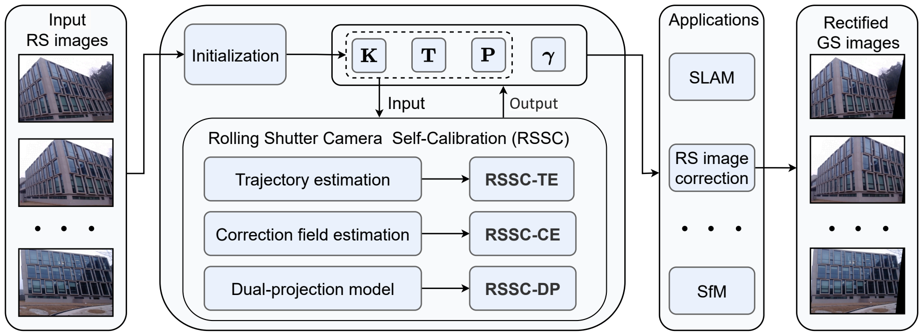

Fig. 1: Computation pipeline of the proposed Rolling Shutter Camera Self-Calibration (RSSC). Given a sequence of RS images, an initial reconstruction of the camera intrinsics $\mathbf{K}$, camera poses $\mathbf{T}$, and 3D point coordinates $\mathbf{P}$ is first obtained. It is then refined by the RSSC framework, where a self-calibrating BA jointly estimates the readout time ratio γ. The estimated RS camera parameters can subsequently be used for downstream tasks such as SLAM, SfM, and RS image correction.

wobble, and the well-known jello effect [29]. These RS distortions significantly complicate many fundamental vision tasks, including camera calibration [1, 15, 16, 30], absolute and relative pose estimation [2, 8], as well as BA [13, 24] in SfM [4, 20] and SLAM [19, 34] systems.

The availability of accurate camera parameters is essential for geometric 3D vision problem, such as 3D reconstruction [12,37]. Compared with the GS model, RS imaging requires not only accurate intrinsic calibration but also an explicit modeling of the temporal readout process, depending on the readout time ratio γ (Fig. 3(a)). This extra parameter is defined as the ratio between the exposure readout time and the inter-frame interval [23, 31, 32, 44]; it strongly controls the magnitude of the RS distortion and is widely adopted in RS image rectification [3, 23, 31, 32, 44] and camera pose estimation [43, 44]. However, unlike standard intrinsic parameters, γ cannot be obtained from GS calibration procedures.

Although RS camera calibration has been studied, existing methods rely on dedicated calibration patterns or auxiliary sensors, limiting their applicability in unconstrained environments. In practice, the readout time ratio γ is often missing, and purely vision-based self-calibration—without calibration targets or additional hardware—remains an open objective. To address this challenge, we propose a general RS self-calibration framework that jointly estimates camera intrinsics and the readout time ratio directly from image observations, as illustrated in Fig. 1.

2 Related Work and Contributions

Global shutter camera self-calibration. Self-calibration for GS cameras has been extensively studied. Methods fall into three groups: classical, deep learning-based, and hybrid methods.

Table 1: Comparison of representative methods, with applicable sensor type (GS or RS), the need for calibration targets or IMU, and the ability to estimate shutter parameters.

	DroidCalib [11]	SelfSup-Calib [9]	ARSC [1]	RSCC [30]	RSC-IMU [15]	**Ours**
Sensor type	GS	GS	RS	RS	RS	RS
Target-free	✓	✓	✗	✗	✓	✓
IMU-free	✓	✓	✓	✓	✗	✓
Shutter param.	✗	✗	✓	✓	✓	✓

Classical self-calibration. Classical methods are primarily based on multi-view geometric constraints to model image sequences. Early work estimated the intrinsic parameters of a pinhole camera from three views using the Kruppa equations [27, 28]. Triggs *et al.* [38] further derived self-calibration constraints for planar scenes. These methods were progressively refined over time, and a comprehensive survey is provided in [14]. More recent studies typically adopt motion-based pipelines [33, 39], in which camera intrinsics, 3D structure, and poses are jointly optimized within BA.

Learning-based self-calibration. Deep learning-based methods can be grouped into single-image and video-based methods. Single-image methods include image undistortion networks [41,42] and models that directly regress focal length, radial distortion, and horizon parameters [5, 22, 25, 45]. Video-based methods typically adopt self-supervised learning to jointly regress pixel-wise depth, camera motion, and intrinsic parameters [7, 9, 10, 18]. However, the accuracy of the estimated intrinsics is often limited [10].

Hybrid self-calibration. Recent works integrate learning with classical BA, which improves robustness and accuracy. BA-Net [36] introduces differentiable BA for end-to-end pose estimation, while DroidCalib [11] incorporates a self-calibrating BA layer into learning-based models. VGG-SfM [40] and FlowMap [35] further tighten the coupling between neural networks and geometric optimization, jointly enforcing multi-view consistency and optimizing poses, intrinsics, and depth or 3D points within differentiable frameworks.

Rolling shutter camera calibration. Despite its maturity for GS cameras, self-calibration does not exist for RS cameras; instead calibration methods are used, which typically require external calibration targets or additional hardware.

RS calibration from calibration board. Geyer *et al.* [29] proposed a geometric model for RS cameras, highlighting their differences from GS cameras and the need for precise LED frequency control for calibration. Huai *et al.* [1] extended this to full intrinsic calibration using an LED panel, estimating both intrinsics and row delay. Oth *et al.* [30] introduced a method using calibration board videos, jointly estimating intrinsics and row delay via continuous-time trajectory modeling and batch optimization.

RS calibration from IMU. Lee *et al.* [21] proposed a joint IMU–RS camera calibration using a gray-box framework and an unscented Kalman filter to estimate IMU noise and sensor parameters. Huai *et al.* [15] introduced a continuous-time

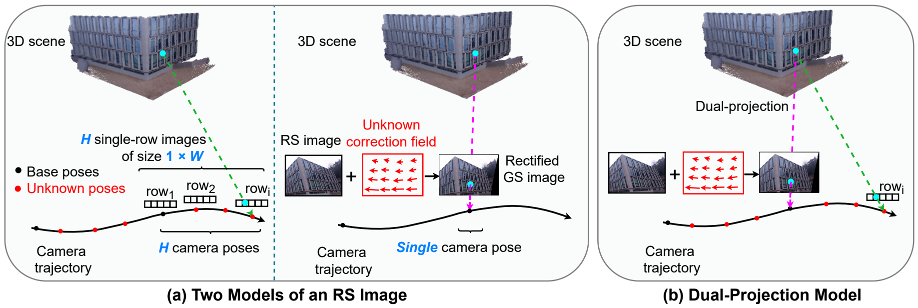

Fig. 2: (a) Two perspectives for modeling RS images. Left: the RS image is modeled as a composition of multiple independently captured single-row GS images; Right: the RS image is modeled as a single GS image with distortion. (b) The proposed dual-projection model projects each 3D point twice, onto its original RS 2D point and onto the rectified GS 2D point.

B-spline representation, assigning a unique camera pose to each feature, improving calibration accuracy.

Contributions. We propose an RS self-calibration framework where the imaging process is modeled from two complementary perspectives, resulting in three implementations, as illustrated in Fig. 2. The first model, denoted RSSC-TE, is based on continuous-time Trajectory Estimation with cumulative B-splines, where camera intrinsics, the readout time ratio, camera poses and 3D points are jointly optimized within a row-wise pose formulation. The second model, denoted RSSC-CE, is based on Correction fields Estimation to rectify RS images, followed by a GS BA with an adapted cost. The estimation of correction fields is performed by fitting projected points with a smooth distortion model. Building upon these complementary models, we further introduce RSSC-DP, based on a Dual-Projection model in which each 3D point is simultaneously constrained by its original RS observation and its rectified GS counterpart. This unified formulation enforces stronger geometric consistency and improves optimization stability. A comparison with representative methods is provided in Table 1. Our contributions are summarized as follows:

1. The first self-calibration method for RS cameras, jointly estimating the intrinsics and the readout time ratio without markers or special hardware.
2. The adaptation and combination of two RS camera models to enable self-calibration, namely pose-based continuous-time modeling and rectification-based observation modeling. The combination produces a dual-projection formulation, improving stability and consistency.

3 Background

We review the RS projection model and then cumulative B-splines on Lie groups.

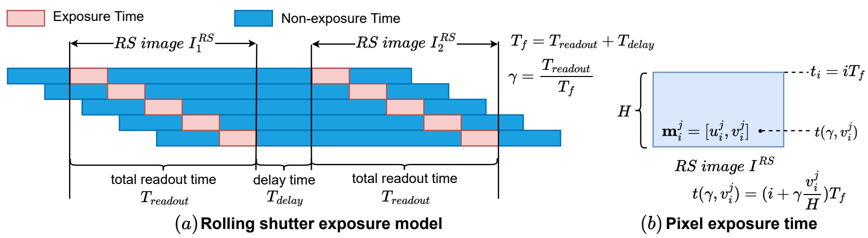

Fig. 3: Illustration of the RS imaging and pixel-wise exposure modeling. (a) Exposure mechanism between two RS consecutive frames. (b) Pixel exposure time formulation within an RS image.

3.1 Rolling Shutter Camera Projection Models

Unlike a GS camera, an RS camera captures images row by row. As illustrated in Fig. 3(a), we denote the total readout time and the delay time as T_{readout} and T_{delay}, respectively, and $T_f = T_{\text{readout}} + T_{\text{delay}}$ as the total time interval between two consecutive frames. The readout time ratio $\gamma = T_{\text{readout}}/T_f$ can be used to determine the timestamp of different image rows. As shown in Fig. 3(b), assuming that a 3D point $\mathbf{P}^j = [X^j, Y^j, Z^j]^\top$ is observed by an RS camera i represented by measurement $\mathbf{m}_i^j = [u_i^j, v_i^j]$ in the image domain, the timestamp of the first row of the i-th image is given by $t_i = iT_f$. For an image of height H, the timestamp corresponding to the point $\mathbf{m}_i^j$ can be expressed as:

$$t(\gamma, v_i^j) = \left(i + \gamma \frac{v_i^j}{H} \right) T_f. \tag{1}$$

Different rows in an RS image correspond to different timestamps, so representing the entire image with a single camera pose is inappropriate. Instead, each image row is associated with a distinct camera pose along the continuous motion trajectory. Specifically, for a point $\mathbf{m}_i^j$ located on the i-th RS image, the camera pose is $\mathbf{T}(i, v_i^j, \gamma)$. Given intrinsic parameters $\mathbf{K}(f_x, f_y, c_x, c_y)$ and denoting perspective projection as function $\pi(\cdot)$, RS projection is:

$$\mathbf{m}_i^j = \pi(\mathbf{K}\mathbf{T}(i, v_i^j, \gamma)\mathbf{P}^j). \tag{2}$$

3.2 Cumulative B-Splines on Lie Groups

We represent the continuous-time trajectory $\mathbf{T}(t)$ using cumulative cubic B-splines [26]. The trajectory is defined over the time interval $[t_{\min}, t_{\max}]$. Its shape is governed by a set of n control poses $\{\mathbf{T}_i \in \mathrm{SE}(3) \mid i = 0, \ldots, n-1\}$, which are associated with uniformly spaced knots $t_0, t_1, \ldots, t_{n-1}$ within the interval. The spacing between consecutive knots is denoted by Δt.

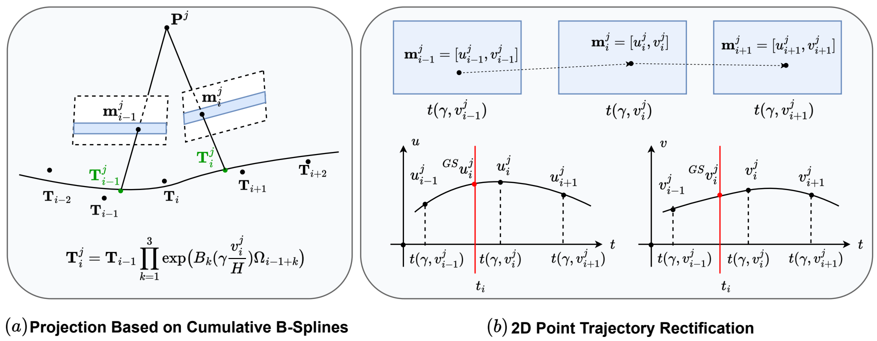

Fig. 4: Two different fitting strategies: (a) Representing the camera trajectory as a function of γ. (b) Representing the 2D points as a function of γ.

Each point along the trajectory is influenced by four consecutive control points. Specifically, for a given time $t \in [t_i, t_{i+1})$, the influencing control points correspond to knots $t_{i-1}, t_i, t_{i+1}, t_{i+2}$. The pose at time t is then given by:

$$\mathbf{T}(t) = \mathbf{T}_{i-1} \prod_{k=1}^{3} \exp\big(\mathbf{B}_k(s)\boldsymbol{\Omega}_{i-1+k}\big), \tag{3}$$

where $\boldsymbol{\Omega}_i = \log\big(\mathbf{T}_{i-1}^{-1}\mathbf{T}_i\big) \in \mathfrak{se}(3)$ and $s = (t - t_i)/\Delta t \in [0, 1)$ normalizes the time offset within the current knot interval. The cumulative basis functions $\mathbf{B}(s)$ are:

$$\mathbf{B}(s) = \frac{1}{6}\begin{pmatrix} 6 & 0 & 0 & 0 \\ 5 & 3 & -3 & 1 \\ 1 & 3 & 3 & -2 \\ 0 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} 1 \\ s \\ s^2 \\ s^3 \end{pmatrix}, \tag{4}$$

where $\mathbf{B}_k(s)$ denotes the k-th element of $\mathbf{B}(s)$, with indexing starting from 0. The resulting pose $\mathbf{T}(t)$ transforms points from local coordinates to world coordinates.

4 Proposed Method

We first model an RS image from two complementary perspectives, as illustrated in Fig. 2(a), from which we derive two RS self-calibration methods. We then integrate these models into a unified dual-projection model, as shown in Fig. 2(b), which jointly enforces motion continuity and observation consistency.

4.1 Continuous-time Trajection Estimation: RSSC-TE

We model an RS image as a collection of H individual single-row GS images (Fig. 2(a), left) and introduce RSSC-TE, RS self-calibration via continuous-time trajectory estimation with cumulative B-splines.

Continuous-time pose parameterization. As illustrated in Fig. 4(a), each row of the image corresponds to a different camera pose, and the camera trajectory is modeled using cumulative B-splines as described in Sec. 3. Assuming that n images are captured, we introduce n control poses $\{\mathbf{T}_i \in \mathrm{SE}(3) \mid i = 0, \ldots, n-1\}$, which parameterize the continuous camera trajectory over time. The temporal interval between two consecutive control poses is uniformly equal to T_f.

For the i-th image, we denote the pose corresponding to row v_i^j as $\mathbf{T}_i^j$. This pose is influenced by four consecutive control poses, namely $\mathbf{T}_{i-1}$, $\mathbf{T}_i$, $\mathbf{T}_{i+1}$, and $\mathbf{T}_{i+2}$. Their associated timestamps are $(i-1)T_f$, iT_f, $(i+1)T_f$, and $(i+2)T_f$, respectively. The timestamp of row v_i^j depends on γ and is given by $t(\gamma, v_i^j) = (i+\gamma\frac{v_i^j}{H})T_f$. According to Eq. (3), under the assumption of a reasonably smooth camera trajectory, the pose corresponding to row v_i is:

$$\mathbf{T}_i^j = \mathbf{T}_{i-1} \prod_{k=1}^{3} \exp\big(\mathbf{B}_k(\gamma \frac{v_i^j}{H})\boldsymbol{\Omega}_{i-1+k}\big). \tag{5}$$

Error function. The non-linear least squares optimizers are used to find a solution $\boldsymbol{\theta}_{\mathrm{TE}}^*$ including readout time ratio γ^*, camera intrinsic parameters $\mathbf{K}^*$, camera poses $\mathbf{T}^*$ and 3D points $\mathbf{P}^*$ by minimizing the reprojection error from point j to camera i over all the camera index in set $\mathcal{F}$ and corresponding 3D points index in subset $\mathcal{P}_i$:

$$\boldsymbol{\theta}_{\mathrm{TE}}^* = \{\gamma^*, \mathbf{K}^*, \mathbf{T}^*, \mathbf{P}^*\} = \underset{\boldsymbol{\theta}_{\mathrm{TE}}}{\arg\min} \sum_{i \in \mathcal{F}} \sum_{j \in \mathcal{P}_i} \left\| {}^{TE}\mathbf{e}_i^j \right\|_2^2, \tag{6}$$

where the reprojection residual ${}^{\mathrm{TE}}\mathbf{e}_i^j$ is expressed as:

$${}^{\mathrm{TE}}\mathbf{e}_i^j = \mathbf{m}_i^j - \pi\left(\mathbf{K}\mathbf{T}_{i-1} \prod_{k=1}^{3} \exp\left(\mathbf{B}_k(\gamma \frac{v_i^j}{H})\boldsymbol{\Omega}_{i-1+k}\right) \mathbf{P}^j\right). \tag{7}$$

4.2 Correction Field Estimation: RSSC-CE

We model an RS image as a distorted GS image associated with an unknown correction field (Fig. 2(a), right) and develop RSSC-CE, an RS self-calibration method based on correction field estimation.

Correction field estimation using 2D point trajectory fitting. As illustrated in Fig. 4(b), we model the trajectory of each 2D point as a function of time and temporally rectify observations to align them to a unified GS timestamp. For a 3D point $\mathbf{P}^j$, its 2D observation in frame i is denoted as $\mathbf{m}_i^j$ with timestamp $t(\gamma, v_i^j)$ (Eq. (1)). Given a target GS time t_i, we estimate the rectified point location ${}^{\mathrm{GS}}\mathbf{m}_i^j$ by fitting a local trajectory and evaluating it at t_i. We consider two interpolation schemes for this purpose. Both schemes can be used to approximate and rectify RS pixel trajectories, and their performance will be compared in the experiments.

Quadratic Polynomial Interpolation. Following [31], a quadratic polynomial is fitted using three temporally adjacent observations from frames $i-1$, i, and $i+1$. The resulting method is referred to as RSSC-CEQ. The rectified point is obtained via Lagrange interpolation:

$$ {}^{\mathrm{GS}}\mathbf{m}_i^j(t_i) = L_0(t_i)\mathbf{m}_{i-1}^j + L_1(t_i)\mathbf{m}_i^j + L_2(t_i)\mathbf{m}_{i+1}^j, \tag{8} $$

where the Lagrange basis functions are:

$$ \begin{cases} L_0(t_i) = \frac{(t_i - t(\gamma, v_i^j))(t_i - t(\gamma, v_{i+1}^j))}{(t(\gamma, v_{i-1}^j) - t(\gamma, v_i^j))(t(\gamma, v_{i-1}^j) - t(\gamma, v_{i+1}^j))}, \\ L_1(t_i) = \frac{(t_i - t(\gamma, v_{i-1}^j))(t_i - t(\gamma, v_{i+1}^j))}{(t(\gamma, v_i^j) - t(\gamma, v_{i-1}^j))(t(\gamma, v_i^j) - t(\gamma, v_{i+1}^j))}, \\ L_2(t_i) = \frac{(t_i - t(\gamma, v_{i-1}^j))(t_i - t(\gamma, v_i^j))}{(t(\gamma, v_{i+1}^j) - t(\gamma, v_{i-1}^j))(t(\gamma, v_{i+1}^j) - t(\gamma, v_i^j))}. \end{cases} \tag{9} $$

Cubic Hermite Interpolation. We also explore a cubic Hermite interpolation scheme using four consecutive observations from frames $i-1$, i, $i+1$, and $i+2$. The resulting method is referred to as RSSC-CEH. The rectified point is:

$$ {}^{\mathrm{GS}}\mathbf{m}_i^j(t_i) = h_{00}(\tau)\mathbf{m}_i^j + h_{10}(\tau)\Delta t\,\mathbf{d}_i + h_{01}(\tau)\mathbf{m}_{i+1}^j + h_{11}(\tau)\Delta t\,\mathbf{d}_{i+1}, \tag{10} $$

where τ is the interpolation parameter:

$$ \tau = \frac{t_i - t(\gamma, v_i^j)}{t(\gamma, v_{i+1}^j) - t(\gamma, v_i^j)}, \tag{11} $$

and the Hermite basis functions are:

$$ \begin{cases} h_{00}(\tau) = 2\tau^3 - 3\tau^2 + 1, \\ h_{10}(\tau) = \tau^3 - 2\tau^2 + \tau, \\ h_{01}(\tau) = -2\tau^3 + 3\tau^2, \\ h_{11}(\tau) = \tau^3 - \tau^2. \end{cases} \tag{12} $$

The tangent vectors are approximated using central differences:

$$ \mathbf{d}_i = \frac{\mathbf{m}_{i+1}^j - \mathbf{m}_{i-1}^j}{t(\gamma, v_{i+1}^j) - t(\gamma, v_{i-1}^j)}, \quad \mathbf{d}_{i+1} = \frac{\mathbf{m}_{i+2}^j - \mathbf{m}_i^j}{t(\gamma, v_{i+2}^j) - t(\gamma, v_i^j)}. \tag{13} $$

Error function. Following Sec. 4.1, we estimate the parameters $\boldsymbol{\theta}_{\mathrm{CE}}^*$ via nonlinear least squares optimization. The cost is the reprojection error over all camera indices in $\mathcal{F}$ and corresponding 3D point indices in $\mathcal{P}_i$:

$$ \boldsymbol{\theta}_{\mathrm{CE}}^* = \left\{\gamma^*, \mathbf{K}^*, {}^{\mathrm{GS}}\mathbf{T}^*, \mathbf{P}^*\right\} = \underset{\boldsymbol{\theta}_{\mathrm{CE}}}{\arg\min} \sum_{i \in \mathcal{F}} \sum_{j \in \mathcal{P}_i} \left\| {}^{\mathrm{CE}}\mathbf{e}_i^j \right\|_2^2, \tag{14} $$

where the reprojection residual ${}^{\mathrm{CE}}\mathbf{e}_i^j$ is expressed as:

$$ {}^{\mathrm{CE}}\mathbf{e}_i^j = {}^{\mathrm{GS}}\mathbf{m}_i^j(t_i) - \pi(\mathbf{K}\,{}^{\mathrm{GS}}\mathbf{T}_i\mathbf{P}^j). \tag{15} $$

4.3 Combined Method: RSSC-DP

We combine RSSC-TE, which captures motion continuity via continuous-time pose modeling, and RSSC-CE, which enforces observation consistency through 2D points rectification, into a unified dual-projection formulation RSSC-DP (Fig. 2(b)). Under this model, each 3D point is constrained by both its original RS observation and the corresponding rectified GS observation. As shown in Sec. 5, this additional constraint improves the robustness and accuracy of self-calibration.

Continuous-time constrained GS poses. In RSSC-CE, the rectified GS pose ${}^{\mathrm{GS}}\mathbf{T}_i$ at timestamp t_i was previously treated as an independent optimization variable. In RSSC-DP, we instead parameterize it using the same cumulative B-spline trajectory defined in Sec. 4.1. Specifically, let t_i denote the reference GS timestamp of frame i. The corresponding pose is obtained by evaluating the continuous trajectory:

$$\mathbf{T}(t_i) = \mathbf{T}_{i-1} \prod_{k=1}^{3} \exp\big(\mathbf{B}_k(s_i)\boldsymbol{\Omega}_{i-1+k}\big), \tag{16}$$

where:

$$s_i = \frac{t_i - iT_f}{T_f},$$

and $\mathbf{T}_{i-1}, \dots, \mathbf{T}_{i+2}$ are the four control poses influencing t_i. Accordingly, we replace ${}^{\mathrm{GS}}\mathbf{T}_i$ with the B-spline controlled trajectory $\mathbf{T}(t_i)$. As a result, the rectified GS poses are no longer independent parameters, but are fully determined by the continuous-time trajectory.

Error function. With the continuous-time constrained GS poses introduced above, RSSC-TE and RSSC-CE are no longer two independent formulations, but become two complementary projection instances defined over the same underlying trajectory. For each 3D point $\mathbf{P}^j$ observed in frame i, the unified trajectory naturally induces two geometrically related projection events: 1) a row-dependent RS projection at timestamp $t(\gamma, v_i^j)$, and 2) a reference GS projection at timestamp t_i. The RSSC-TE residuals captures consistency with the original RS observation:

$${}^{\mathrm{TE}}\mathbf{e}_i^j = \mathbf{m}_i^j - \pi\Big(\mathbf{K}\mathbf{T}(t(\gamma, v_i^j))\mathbf{P}^j\Big)\,, \tag{17}$$

while the RSSC-CE residuals captures consistency with its rectified GS counterpart:

$${}^{\mathrm{CE}}\mathbf{e}_i^j = {}^{\mathrm{GS}}\mathbf{m}_i^j(t_i) - \pi\big(\mathbf{K}\mathbf{T}(t_i)\mathbf{P}^j\big)\,. \tag{18}$$

These two residuals arise from two temporally distinct projections of the same 3D point along a shared continuous trajectory. The resulting objective is formulated as a weighted least-squares problem:

$$\boldsymbol{\theta}_{\mathrm{DP}}^* = \{\gamma^*, \mathbf{K}^*, \mathbf{T}^*, \mathbf{P}^*\} = \underset{\boldsymbol{\theta}_{\mathrm{DP}}}{\arg\min} \sum_{i\in\mathcal{F}} \sum_{j\in\mathcal{P}_i} \Big(\|{}^{\mathrm{TE}}\mathbf{e}_i^j\|_2^2 + \lambda\|{}^{\mathrm{CE}}\mathbf{e}_i^j\|_2^2\Big), \tag{19}$$

where λ balances the contributions of the two projection models.

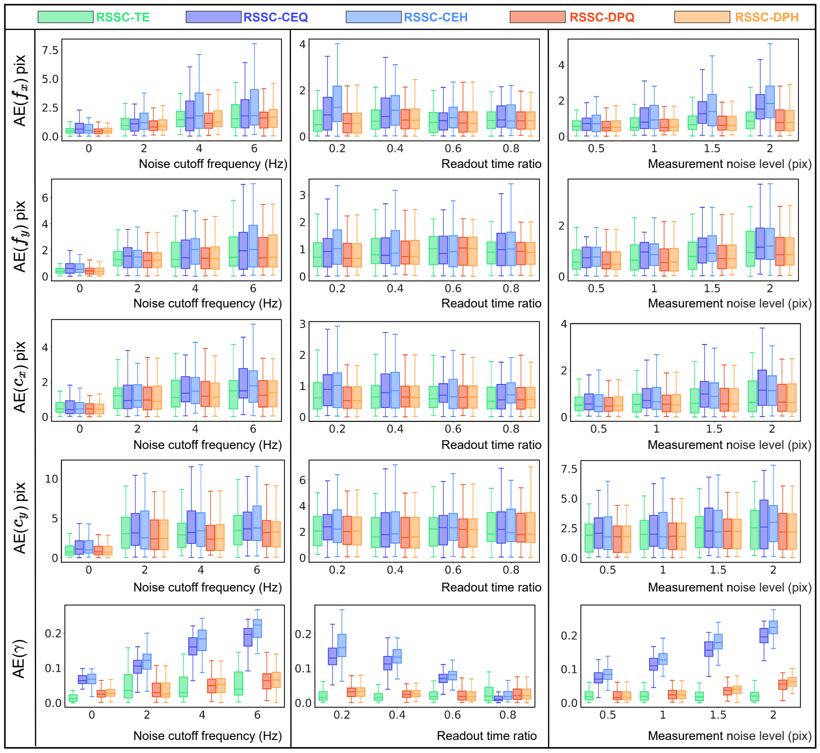

Fig. 5: Intrinsic parameters and readout time ratio errors for the proposed RS camera self-calibration methods under varying experimental conditions, with, from left to right, trajectory smoothness, readout time ratio, and measurement noise.

4.4 Discussion

The two constraints in RSSC-DP are complementary and compatible in practice. A non-smooth 3D trajectory can still yield locally smooth 2D projections, allowing RSSC-CE to compensate for inaccuracies in RSSC-TE's global trajectory representation; conversely, when the 2D motion field is locally irregular, RSSC-TE's global smoothness prior can mitigate per-point fitting errors in RSSC-CE. If the objectives were conflicting, RSSC-DP would be expected to perform substantially worse than at least one individual model in some scenarios. However, as shown in Sec. 5, such degradation was never observed, and the performance gap remained small even when RSSC-DP was not the best performer.

5 Experimental Results

We first investigate the influence of different conditions on the proposed RSSC methods through simulated data. We then evaluate on real datasets and compare with state-of-the-art methods.

Table 2: Optimization efficiency comparison. The rows correspond to per-iteration time (seconds), iterations to convergence, and total optimization time, respectively. All results are medians over 50 runs. Bold is best and underlined second-best.

Method	RSSC-TE	RSSC-CEQ	RSSC-CEH	RSSC-DPQ	RSSC-DPH
Iteration time (s)	0.5142	**0.0841**	<u>0.0851</u>	0.7668	0.7735
Nb. iterations	32	**5**	<u>6</u>	9	14
Total time (s)	5.8657	**0.2321**	<u>0.2765</u>	2.6048	3.9185

5.1 Synthetic Data Experiments

Simulation setup and compared methods. The 3D scene is defined within a $4 \times 4 \times 4$ cubic volume. For each trial, a smooth continuous-time camera trajectory is generated, and we conduct a total of 50 independent trials, each consisting of 50 images. The weighting parameter λ in Eq. (19) is set to 0.5. The proposed methods—RSSC-TE, RSSC-CEQ, RSSC-CEH, RSSC-DPQ, and RSSC-DPH—are evaluated under varying simulation conditions.
Experimental variables. We evaluate the proposed methods under three simulation factors, which include trajectory smoothness, RS strength, and measurement noise.

Trajectory smoothness is controlled by injecting band-limited Gaussian noise into the continuous-time trajectory $\mathbf{x}(t) \in \mathbb{R}^6$. The perturbed trajectory is:

$$\tilde{\mathbf{x}}(t) = \mathbf{x}(t) + \tilde{\mathbf{n}}(t), \quad \tilde{\mathbf{n}}(t) = [\tilde{n}_1(t), \ldots, \tilde{n}_6(t)]^\top. \tag{20}$$

Let $n_d(t) \sim \mathcal{N}(0,1)$ denote zero-mean unit-variance Gaussian noise. The band-limited Gaussian noise $\tilde{n}_d(t)$ is computed as

$$\tilde{n}_d(t) = \lambda_d \, \mathcal{F}^{-1}(H(f; f_c, m) \, \mathcal{F}(n_d(t))) \, , \tag{21}$$

where $H(f; f_c, m)$ denotes the frequency response of a low-pass Butterworth filter with cutoff frequency f_c and order m. The filter order is fixed to $m = 4$, and the parameter λ_d controls the noise amplitude, set to 0.015 rad for rotation and 1.5 for translation. The cutoff frequency $f_c \in \{0, 2, 4, 6\}$ Hz controls the trajectory smoothness, where larger f_c results in less smooth motion.

RS distortion strength is controlled by the readout time ratio $\gamma \in \{0.2, 0.4, 0.6, 0.8\}$, and measurement noise by the standard deviation $\sigma \in \{0.5, 1.0, 1.5, 2.0\}$ pixels of Gaussian noise added to the 2D measurements.
Evaluation metrics. We evaluate the proposed methods using two key quantitative metrics. First, the absolute error (AE) is computed for the camera intrinsic parameters and the readout time ratio, $\theta \in \{f_x, f_y, c_x, c_y, \gamma\}$, defined as $\mathrm{AE}(\hat{\theta}) = |\hat{\theta} - \theta|$, where θ and $\hat{\theta}$ denote the true and estimated values, respectively. Second, we assess optimization efficiency by measuring the per-iteration time, the number of iterations to convergence, and the total optimization time. Convergence is declared when the parameter update is below 10^{-8}, the infinity norm of the gradient is below 10^{-10}, or the change in the objective function is below 10^{-8}.

Analysis of experimental results. The results of the proposed methods under varying simulation conditions are given in Fig. 5 and Table 2.

Regarding trajectory smoothness, estimation errors increase for all methods as trajectory perturbation grows, indicating reduced stability under less smooth motion. The RSSC-DP methods consistently achieve the lowest errors, showing superior robustness from the dual-projection formulation that jointly constrains pose and projection, while RSSC-CE degrades more rapidly due to its sensitivity to motion inaccuracies.

For variations in the readout time ratio γ, intrinsic parameter estimation remains largely unaffected for all methods. However, RSSC-CE exhibits increasing errors in the estimated γ as the readout time decreases, whereas RSSC-TE and RSSC-DP maintain stable performance.

When considering measurement noise, RSSC-CE is highly sensitive, with errors increasing rapidly as 2D noise grows. In contrast, RSSC-TE and RSSC-DP remain robust, benefiting from continuous-time modeling and dual-projection constraints that provide inherent regularization.

In terms of computational efficiency, Table 2 shows that RSSC-CE achieves the fastest per-iteration time and requires the fewest iterations, yielding the lowest total optimization time. RSSC-TE has slightly shorter per-iteration time than RSSC-DP but requires substantially more iterations, resulting in the highest total time. Overall, RSSC-DP balances accuracy and efficiency, offering competitive performance at moderate computational cost.

Overall, the results highlight that RSSC-TE excels under smooth trajectories, RSSC-CE prioritizes computational efficiency, and RSSC-DP achieves the best accuracy and robustness across all conditions, making it the most versatile choice for practical applications.

5.2 Real Data Experiments

Datasets and compared methods. We conduct self-calibration evaluation on two publicly available RS datasets, WHU-RSVI [6] and TUM-RSVI [8]. For each sequence, 50 consecutive frames are selected for calibration. Our method uses COLMAP [33] to obtain an initial reconstruction, with the readout time ratio initialized to 0.5, and the weighting parameter λ in Eq. (19) is set to 0.5. The comparison methods include *1)* COLMAP [33], *2)* DroidCalib [11], and *3)* SelfSup-Calib [9]. Since no publicly available methods exist for self-calibration of RS cameras, only approaches designed for GS cameras are included in the comparison.

Evaluation metrics. We evaluate the accuracy of intrinsic parameters and the readout time ratio using the median absolute error (MAE), together with the median absolute trajectory error (MATE) after alignment to the ground-truth trajectory using a Sim(3) similarity transformation.

Experiment results. Table 3 presents the intrinsic parameters and readout time ratio MAEs, and MATE computed over all sequences in each dataset, and Fig. 6 shows a comparison between the trajectories obtained by different methods and the ground truth (full results are provided in the supplementary ma-

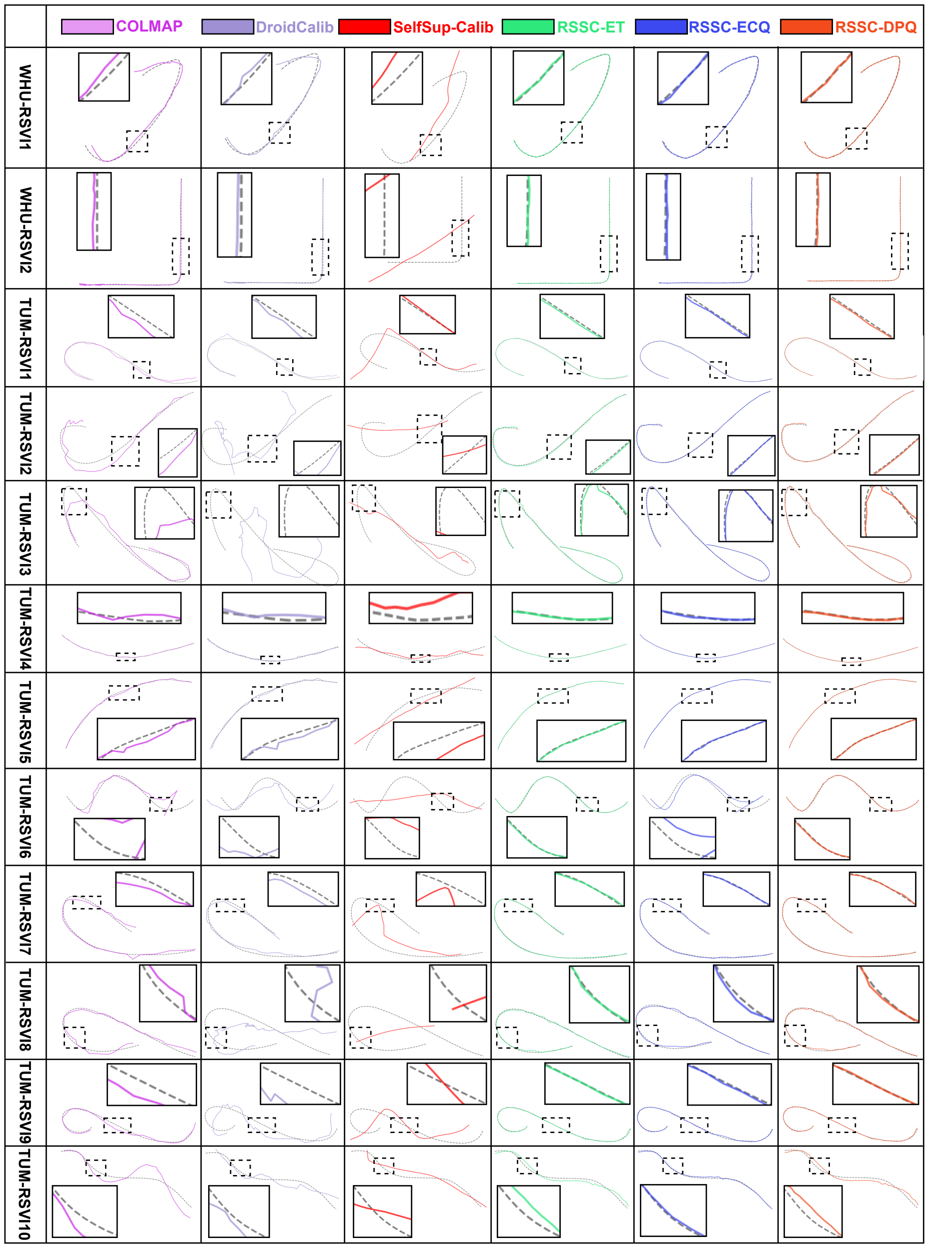

Fig. 6: Trajectory comparison on the WHU-RSVI [6] and TUM-RSVI [8] datasets. Each row denotes a different trajectory sequence, while each column corresponds to a specific method. Selected local regions (highlighted by dashed boxes) are magnified and displayed within solid boxes for detailed inspection.

Table 3: Self-calibration results on WHU-RSVI [6] and TUM-RSVI [8] datasets. All results are median absolute errors (MAE) of the intrinsic parameters and readout time ratio, and median absolute trajectory errors (MATE) over all sequences in each dataset.

Dataset	Method	MAE(f_x)	MAE(f_y)	MAE(c_x)	MAE(c_y)	MAE(γ)	MATE
WHU	COLMAP [33]	3.05	11.26	4.36	8.17	—	0.0453
	DroidCalib [11]	0.56	22.33	3.13	13.80	—	0.0317
	SelfSup-Calib [9]	195.98	129.28	13.85	23.78	—	0.4585
	RSSC-TE	<u>0.07</u>	**0.31**	<u>0.23</u>	**0.60**	<u>0.0485</u>	0.0170
	RSSC-CEQ	0.16	2.82	0.27	1.04	0.0814	0.0131
	RSSC-CEH	0.67	2.43	0.71	1.60	0.1789	0.0238
	RSSC-DPQ	**0.03**	<u>0.44</u>	**0.19**	<u>0.65</u>	**0.0483**	**0.0101**
	RSSC-DPH	0.30	**0.31**	0.24	1.50	0.0945	<u>0.0126</u>
TUM	COLMAP [33]	33.58	14.82	16.01	28.5	–	0.0374
	DroidCalib [11]	71.96	67.56	17.54	41.76	–	0.0963
	SelfSup-Calib [9]	359.77	338.17	59.885	9.85	–	0.1432
	RSSC-TE	<u>3.16</u>	**1.48**	1.76	**1.12**	0.1144	<u>0.0063</u>
	RSSC-CEQ	3.79	2.67	<u>1.71</u>	1.25	**0.0609**	0.0066
	RSSC-CEH	3.71	3.31	**1.40**	1.52	0.0852	0.0081
	RSSC-DPQ	**3.10**	<u>2.39</u>	1.93	<u>1.16</u>	<u>0.0814</u>	**0.0055**
	RSSC-DPH	4.17	3.06	2.16	2.14	0.1374	0.0072

terial). We observe that, for all evaluation metrics, our self-calibration methods consistently outperform the baselines. Among the proposed methods, RSSC-TE and RSSC-DPQ achieve the best performance. These results clearly validate the effectiveness, accuracy, and robustness of the proposed RS self-calibration approach. Additional comparisons with target-based RS calibration methods are provided in the supplementary material, where our approach achieves comparable performance.

6 Conclusion

We have proposed the first RS camera self-calibration framework, jointly estimating intrinsics and the readout time ratio from a target-free general video. Our approach models self-calibration from two complementary perspectives: RSSC-TE performs trajectory estimation using continuous-time B-splines to optimize intrinsics, readout time ratio, poses and 3D points; RSSC-CE estimates correction fields by fitting 2D pixel trajectories to rectify RS images into a GS formulation. These strategies are unified in a dual-projection model, RSSC-DP, which enforces geometric consistency between row-dependent and rectified observations, thereby enhancing both estimation accuracy and robustness under challenging motion and imaging conditions. Extensive experiments on synthetic and real data demonstrate the accuracy, robustness, and practical effectiveness of our method. Future work includes jointly optimizing RS camera parameters and dense 3D scene representations, such as 3D Gaussian models, as well as studying degeneracies in RS camera self-calibration.

References

1. Automated rolling shutter calibration with an led panel. Optics Letters **48**(4), 847–850 (2023)
2. Albl, C., Kukelova, Z., Larsson, V., Pajdla, T.: Rolling shutter camera absolute pose. IEEE transactions on pattern analysis and machine intelligence **42**(6), 1439–1452 (2019)
3. Albl, C., Kukelova, Z., Larsson, V., Polic, M., Pajdla, T., Schindler, K.: From two rolling shutters to one global shutter. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2505–2513 (2020)
4. Albl, C., Sugimoto, A., Pajdla, T.: Degeneracies in rolling shutter sfm. In: European Conference on Computer Vision. pp. 36–51. Springer (2016)
5. Bogdan, O., Eckstein, V., Rameau, F., Bazin, J.C.: Deepcalib: A deep learning approach for automatic intrinsic calibration of wide field-of-view cameras. In: Proceedings of the 15th ACM SIGGRAPH European Conference on Visual Media Production. pp. 1–10 (2018)
6. Cao, L., Ling, J., Xiao, X.: The whu rolling shutter visual-inertial dataset. IEEE Access **8**, 50771–50779 (2020)
7. Chen, Y., Schmid, C., Sminchisescu, C.: Self-supervised learning with geometric constraints in monocular video: Connecting flow, depth, and camera. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 7063–7072 (2019)
8. Dai, Y., Li, H., Kneip, L.: Rolling shutter camera relative pose: Generalized epipolar geometry. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4132–4140 (2016)
9. Fang, J., Vasiljevic, I., Guizilini, V., Ambrus, R., Shakhnarovich, G., Gaidon, A., Walter, M.R.: Self-supervised camera self-calibration from video. In: 2022 International Conference on Robotics and Automation (ICRA). pp. 8468–8475. IEEE (2022)
10. Gordon, A., Li, H., Jonschkowski, R., Angelova, A.: Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 8977–8986 (2019)
11. Hagemann, A., Knorr, M., Stiller, C.: Deep geometry-aware camera self-calibration from video. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3438–3448 (2023)
12. Hartley, R.: Multiple view geometry in computer vision, vol. 665. Cambridge university press (2003)
13. Hedborg, J., Forssén, P.E., Felsberg, M., Ringaby, E.: Rolling shutter bundle adjustment. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1434–1441. IEEE (2012)
14. Hemayed, E.E.: A survey of camera self-calibration. In: Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance, 2003. pp. 351–357. IEEE (2003)
15. Huai, J., Zhuang, Y., Lin, Y., Jozkow, G., Yuan, Q., Chen, D.: Continuous-time spatiotemporal calibration of a rolling shutter camera-imu system. IEEE Sensors Journal **22**(8), 7920–7930 (2022)
16. Ito, E., Okatani, T.: Self-calibration-based approach to critical motion sequences of rolling-shutter structure from motion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 801–809 (2017)

17. Janesick, J., Pinter, J., Potter, R., Elliott, T., Andrews, J., Tower, J., Cheng, J., Bishop, J.: Fundamental performance differences between cmos and ccd imagers: part iii. In: Astronomical and space optical systems. vol. 7439, pp. 47–72. SPIE (2009)
18. Jeong, Y., Ahn, S., Choy, C., Anandkumar, A., Cho, M., Park, J.: Self-calibrating neural radiance fields. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5846–5854 (2021)
19. Kim, J.H., Cadena, C., Reid, I.: Direct semi-dense slam for rolling shutter cameras. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). pp. 1308–1315. IEEE (2016)
20. Lao, Y., Ait-Aider, O., Araujo, H.: Robustified structure from motion with rolling-shutter camera using straightness constraint. Pattern Recognition Letters **111**, 1–8 (2018)
21. Lee, C.R., Yoon, J.H., Yoon, K.J.: Calibration and noise identification of a rolling shutter camera and a low-cost inertial measurement unit. Sensors **18**(7), 2345 (2018)
22. Lee, J., Go, H., Lee, H., Cho, S., Sung, M., Kim, J.: Ctrl-c: Camera calibration transformer with line-classification. In: Proceedings of the IEEE/CVF International conference on computer vision. pp. 16228–16237 (2021)
23. Liang, C.K., Chang, L.W., Chen, H.H.: Analysis and compensation of rolling shutter effect. IEEE transactions on image processing **17**(8), 1323–1330 (2008)
24. Liao, B., Qu, D., Xue, Y., Zhang, H., Lao, Y.: Revisiting rolling shutter bundle adjustment: Toward accurate and fast solution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4863–4871 (2023)
25. Lopez, M., Mari, R., Gargallo, P., Kuang, Y., Gonzalez-Jimenez, J., Haro, G.: Deep single image camera calibration with radial distortion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11817–11825 (2019)
26. Lovegrove, S., Patron-Perez, A., Sibley, G.: Spline fusion: A continuous-time representation for visual-inertial fusion with application to rolling shutter cameras. In: BMVC. vol. 2, p. 8 (2013)
27. Luong, Q.T., Faugeras, O.D.: Self-calibration of a moving camera from point correspondences and fundamental matrices. International Journal of computer vision **22**(3), 261–289 (1997)
28. Maybank, S.J., Faugeras, O.D.: A theory of self-calibration of a moving camera. International journal of computer vision **8**(2), 123–151 (1992)
29. Meingast, M., Geyer, C., Sastry, S.: Geometric models of rolling-shutter cameras. arXiv preprint cs/0503076 (2005)
30. Oth, L., Furgale, P., Kneip, L., Siegwart, R.: Rolling shutter camera calibration. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1360–1367 (2013)
31. Qu, D., Lao, Y., Wang, Z., Wang, D., Zhao, B., Li, X.: Towards nonlinear-motion-aware and occlusion-robust rolling shutter correction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10680–10688 (2023)
32. Qu, D., Liao, B., Zhang, H., Ait-Aider, O., Lao, Y.: Fast rolling shutter correction in the wild. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(10), 11778–11795 (2023)
33. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4104–4113 (2016)

34. Schubert, D., Demmel, N., Usenko, V., Stuckler, J., Cremers, D.: Direct sparse odometry with rolling shutter. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 682–697 (2018)
35. Smith, C., Charatan, D., Tewari, A., Sitzmann, V.: Flowmap: High-quality camera poses, intrinsics, and depth via gradient descent. In: 2025 International Conference on 3D Vision (3DV). pp. 389–400. IEEE (2025)
36. Tang, C., Tan, P.: Ba-net: Dense bundle adjustment network. arXiv preprint arXiv:1806.04807 (2018)
37. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: a factorization method. International journal of computer vision **9**(2), 137–154 (1992)
38. Triggs, B.: Autocalibration from planar scenes. In: European conference on computer vision. pp. 89–105. Springer (1998)
39. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment—a modern synthesis. In: International workshop on vision algorithms. pp. 298–372. Springer (1999)
40. Wang, J., Karaev, N., Rupprecht, C., Novotny, D.: Vggsfm: Visual geometry grounded deep structure from motion. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 21686–21697 (2024)
41. Xue, Z., Xue, N., Xia, G.S., Shen, W.: Learning to calibrate straight lines for fisheye image rectification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1643–1651 (2019)
42. Yin, X., Wang, X., Yu, J., Zhang, M., Fua, P., Tao, D.: Fisheyerecnet: A multi-context collaborative deep network for fisheye image rectification. In: Proceedings of the European conference on computer vision (ECCV). pp. 469–484 (2018)
43. Zhang, Y., Liao, B., Qu, D., Wu, J., Lu, X., Li, W., Xue, Y., Lao, Y.: Ego-motion estimation for vehicles with a rolling shutter camera. IEEE Transactions on Intelligent Vehicles (2024)
44. Zhuang, B., Cheong, L.F., Hee Lee, G.: Rolling-shutter-aware differential sfm and image rectification. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 948–956 (2017)
45. Zhuang, B., Tran, Q.H., Lee, G.H., Cheong, L.F., Chandraker, M.: Degeneracy in self-calibration revisited and a deep learning solution for uncalibrated slam. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 3766–3773. IEEE (2019)