ISPRS Journal of Photogrammetry and Remote Sensing 211 (2024) 452–464

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

Bundle adjustment with motion constraints for uncalibrated multi-camera systems at the ground level

Debao Huang [a,b,c], Rongjun Qin [a,b,c,d,*], Mostafa Elhashash [a,c]

[a] Geospatial Data Analytics Laboratory, The Ohio State University, Columbus, USA
[b] Department of Civil, Environmental and Geodetic Engineering, The Ohio State University, Columbus, USA
[c] Department of Electrical and Computer Engineering, The Ohio State University, Columbus, USA
[d] Translational Data Analytics Institute, The Ohio State University, Columbus, USA

ARTICLE INFO

Keywords:
Structure from motion
Uncalibrated multi-camera systems
Bundle adjustment

ABSTRACT

Multi-camera systems for structure from motion (SfM) are widely deployed in many mapping applications. Existing solutions assume known rig calibration, synchronized frames among cameras, as well as overlapping field of views (FoVs). In this paper, we derive novel geometric constraints assuming minimal knowns about the multi-camera systems, to benefit low-cost and non-expert use cases where uncalibrated multi-camera systems with non-typical geometry setups present, i.e., no rig calibration, no overlapping FoVs. Assuming that these cameras are co-located and share the same motion of the platform, the proposed constraints utilize the parallelism and length proportionality of motion vectors of these co-located cameras and formulate them as translation constraints into the bundle adjustment (BA). The proposed constraints (called motion constraints) impose a first-order penalty to co-located cameras whose motion speeds and directions between frames do not match. With soft constraints, it can handle loosely synchronized frames (with an error within one second). The proposed constraints are integrated into the BA framework and experimented with different camera setups, i.e., on a group of casually co-located GoPro cameras with no rig calibration, and some with no overlapping views. Our results show that the constraints are extremely effective in improving the reconstruction and pose accuracy for ground motion images: in our self-collected open trajectories without loop closure, the proposed constraints are effective in correcting topographical errors (i.e., trajectory drifts) of the resulting models, and the dense point clouds achieve up to 11.34 m (86.12 %) of mean absolute error (MAE) improvement as compared to reference LiDAR point clouds; our results on KITTI-odometry and KITTI-360 datasets also show an improvement of up to 28.82 m (81.05 %) in terms of the root mean square error (RMSE) of absolute pose error (APE). We expect that the proposed constraints are significant not only as additional geometric constraints for image-based mobile mapping, but also will benefit the broader use of photogrammetry, since it empowers the possibility to harness the traditionally so-called low-quality stereo/multi-camera data (e.g., by non-photogrammetry citizen scientists) into improved 3D products.

1. Introduction

A multi-camera system refers to a set of co-located cameras simultaneously collecting images for mapping purposes. These cameras can be fixed through a rig to create parallaxes for stereo purposes and can be equipped with additional sensors such as Global Positioning System (GPS) and Inertial Measurement Unit (IMU), to form a well-integrated and well-calibrated sensor suite. Alternatively, a low-cost version can be as simple as a few casually co-located cameras whose relative positions are fixed at data collections but uncalibrated, e.g., several GoPro cameras mounted on a vehicle. Multi-camera systems have unique advantages in mapping applications (Harmat et al., 2015; Häne et al., 2017; Wierzbicki, 2018), due to their extended field of views (FoVs) and more stable camera networks for geometric reconstruction. Simultaneously using multiple cameras is also regarded as a good practice when performing structure from motion (SfM) or photogrammetric reconstruction at the ground level, as 3D reconstruction using a monocular camera (Engel et al., 2014; Mur-Artal et al., 2015; Z. Xu et al., 2022) is subject to drift problems caused by several factors. First, a monocular camera in a moving trajectory lacks a stable camera network to

* Corresponding author.
E-mail address: qin.324@osu.edu (R. Qin).

https://doi.org/10.1016/j.isprsjprs.2024.04.023
Received 23 May 2023; Received in revised form 24 March 2024; Accepted 22 April 2024
Available online 26 April 2024

accurately estimate lens distortions, causing, e.g., the "doming" effect by radial distortion errors (James & Robson, 2014). Second, the weak camera network itself may lead to less accurate pose estimation, e.g., it does not have image overlap in the direction orthogonal to motion, to resolve good tilt (or pitch) angle. Last but not least, bundle adjustment (BA) applied in incremental SfM is only locally optimal. These factors cause accumulated errors, which are eventually observed as trajectory drifts and 3D topographical distortions (Cornelis et al., 2004). A multi-camera system naturally provides more overlapping images that enhance the camera networks, and more redundancies to create parallaxes in different directions, hence is powerful in collecting convergent images for measurements.

Commercial-grade multi-camera systems are calibrated regularly in the factory, while customized multi-camera systems require calibration in the lab before use, typically through fiducial targets or control arrays (Heng, Lee, et al., 2015; Heng et al., 2019; Lichti et al., 2021; Dong et al., 2023). In addition, cameras are usually held on a special mounting rig, which is prone to errors due to minor sensor displacement during the motion (e.g., platform vibration), or environmental factors such as humidity and temperature. Thus, re-calibration is needed as the camera setup changes. However, this is considered an expensive process, and oftentimes not an option for users with no access to expertise and facilities for calibration.

Self-calibration has been a standard practice in photogrammetry to correct the camera lens distortions and interior parameters using the camera networks. However, calibrating multi-camera systems also involves the relative positions between different cameras, as well as the synchronization of their shutters. Oftentimes the synchronization of frames among multiple cameras requires a hardware solution, i.e., a trigger box (Nikolic et al., 2014) to control shutters. Thus, in addition to applying standard self-calibration for lens distortions, existing approaches exploited coded constraints assuming fixed relative orientations among synchronized frames from stereo or multiple cameras (Schonberger & Frahm, 2016; Maset et al., 2020), which were reported to be overall beneficial to improve the accuracy of the BA. For example, using the fixed relative orientation constraints, one can effectively "re-calibrate" a stereo rig during the reconstruction, which can supplement minor displacement that occurs after the lab calibration. Some of these methods were applied to commercial-grade systems such as aerial oblique camera systems (Maset et al., 2020). These varying approaches all assume aerial blocks with prescribed overlaps among these multiple cameras in aerial collections. Our earlier work (Huang, Elhashash, et al., 2022) proposed baseline constraints for uncalibrated stereo cameras in ground-level image collections: assuming an unknown baseline and only roughly synchronized video frames (with an error within one second), it implemented the constraints into BA that minimized the differences of the baselines of two co-located cameras at different locations, which reported to have increased the accuracy of 3D reconstruction. However, the implementation of the constraints required the FoVs of two cameras to have a certain overlap (i.e., tie points) to initiate baseline computations.

Ground-level multi-camera systems can be sophisticated, especially if such a system is in a "casual" setup. Also, there may exist no prior knowledge about the relative translation between pairs of cameras, no precise synchronization of the frames, and marginal or no overlapping FoVs among these cameras (e.g., two cameras facing in different directions). To our best knowledge, existing approaches were unable to place constraints on such a system, and to the maximum, existing approaches might independently estimate poses for images for each camera following monocular camera-based reconstruction, which apparently would continue to suffer from trajectory drifting and topographical errors as previously mentioned.

Although such a "casual" system provides little geometric constraints among these cameras, the fact that these cameras are co-located during collection still poses weak constraints that tie these frames. Therefore, this paper intends to propose a set of weak constraints characterizing this fact, and thus, integrate these constraints into BA. This can simultaneously resolve poses for image frames from uncalibrated multi-camera systems. Assuming that camera frames are only roughly synchronized (with an error within one second), here we present the motion constraints respectively corresponding to two facts: 1) length proportionality constraint: the length of motion vectors of two cameras should be similar, or at least up to scale. 2) motion parallelism constraint: the motion vectors of two cameras should be directionally parallel. Since the motion constraints only impose regularities on the motion vectors and do not require an explicit baseline to be known, it can be made more robust to synchronization errors, and more importantly, provide constraints for co-located cameras with non-overlapping FoVs in a multi-camera system.

We first introduce our proposed method in a two-camera case with non-overlapping FoVs, then extend it to a more sophisticated configuration with six cameras, containing mixed overlapping and non-overlapping FoVs. We performed experiments to compare reconstruction accuracy for the results obtained from a typical SfM/photogrammetry pipeline with and without our proposed motion constraints applied in BA. Two datasets were collected with LiDAR data available as a reference for both qualitative and quantitative evaluations of reconstruction accuracy, the KITTI-odometry (Geiger et al., 2012) and KITTI-360 datasets (Liao et al., 2022) with ground truth poses were used for evaluations of pose accuracy. The remainder of this paper is organized as follows: **Section 2** briefly introduces recent related works on SfM/photogrammetry using multi-camera systems; **Section 3** presents the formulation of our proposed motion constraints; **Section 4** presents the experiments, including the experimental design, qualitative and quantitative evaluation results, sensitivity analysis and ablation study; **Section 5** concludes this paper with our discussions on the potential usability of the motion constraints.

2. Related work

Multi-camera systems in photogrammetry. Over the past decade, multi-camera systems have gained increasing attention in the photogrammetry industry, mounted on different platforms such as unmanned aerial vehicles (UAVs), ground vehicles, robots, etc. For example, multi-camera systems on UAV or aerial platforms contain a sensor suite consisting of nadir and oblique cameras, onboard GPS/IMU, and precise shutter control systems (Nikolic et al., 2014). On the one hand, it extends the FoVs of the aerial collection. On the other hand, it significantly improves the collection of data integrating oblique geometries of the scene. Because such a sensor suite is well-integrated and calibrated, and its collection pattern allows overlapping FoVs of images at different shots, classic photogrammetric methods can process oblique image blocks well. This naturally broadens the use of aerial multi-camera systems in many applications such as 3D modeling (Papakonstantinou et al., 2018; Xu, Qin, Huang, et al., 2023), land change monitoring (Bertin et al., 2020; Jenal et al., 2020; Jenal et al., 2021; Xu et al., 2021; Huang, Tang, et al., 2022), smart cities (Alshammari & Rawat, 2019; Sakamoto et al., 2022; Kaya et al., 2023), etc. In parallel, multi-camera systems are also deployed on ground vehicle platforms for use in applications such as autonomous driving. The work of Yang et al. (2020) proposed a simultaneous localization and mapping (SLAM) method using multiple cameras, and demonstrated that such a setup could improve the localization accuracy for off-road navigation. Other works (Häne et al., 2017; Heng et al., 2019) focused on using multi-camera systems for autonomous vehicles to enhance 3D visual perception and mapping by using the extended FoVs. Furthermore, multi-camera systems provide robotic vision systems for better environment perception. Zhu et al. (2020) proposed an autonomous method for robot navigation based on wider FoVs from the multi-camera setup. For ground-based multi-camera systems, due to that the platform typically has larger vibration, one critical pre-processing step is to perform stereo calibration prior to collection missions, to "re-calibrate" potential camera

displacement. In general, there are four ways to approach the calibration problem, 1) using ground control points (GCPs) manually identified on site, either through surveyed points or from geo-referenced external LiDAR scans (Triggs, 1999; Jones et al., 2002); 2) using specific scene features such as vanishing points (Caprile & Torre, 1990; Krahnstoever & Mendonca, 2005) to achieve control-free calibration; 3) using coded targets with known dimensions as metrics for calibration (Marcon et al., 2017; Xie et al., 2018; Heng et al., 2019); 4) cloud calibration directly using the natural features in the survey area (Heng, Furgale, et al., 2015; Häne et al., 2017; J. Xu et al., 2022). However, these methods all require extra work in data collection, making it less friendly for calibration tasks on-the-go and oftentimes inaccessible to non-expert users.

Geometric constraints for multi-camera systems. Multi-camera systems naturally pose geometric constraints among different cameras, which has been explored by a few existing works in the literature to improve either camera calibration or reconstruction accuracy. Maset et al. (2021) investigated relative orientation constraints for 3D reconstruction using multi-camera systems and reported that using BA with relative orientation constraints led to improved accuracy. Existing approaches can be generally categorized as enforcing explicit and implicit multi-camera constraints (Detchev et al., 2018). The explicit constraint pre-computes the relative orientation between a master camera to the rest and then enforces this relative orientation as a constant throughout the BA. The implicit constraints do not enforce specific relative orientation between cameras in the rig, while it minimizes the differences of the relative orientation at different rig locations.

Explicit multi-camera constraints: One family of approaches applies the multi-camera constraint explicitly. Maset et al. (2020) estimated the exterior orientation of the master camera and the relative orientations of the slave cameras in the BA at the last stage of SfM pipeline. The relative orientation was constant at different rig locations to enforce the rigidity of the multi-camera system, thus requiring a frame to house the cameras and rigorous synchronization across the cameras. In the work proposed by Cavegn et al. (2018), they exploited and claimed that the usage of calibrated or defined relative orientations in BA could improve the reconstruction accuracy and robustness for mobile mapping multi-camera systems. Another work proposed by Schonberger and Frahm (2016) computed the average relative orientation from the initial SfM process, and the relative orientation was fixed in BA to enforce the rigidity of the system. However, it could be problematic to compute the average values if the initial SfM generated inaccurate results. Another approach (Lerma et al., 2010) included the manually measured pairwise baseline distances among three cameras as constraints in the BA and reported that the calibration could be benefited when the full set of baseline distances of cameras was used.

Implicit multi-camera constraints: Another family of approaches applies the implicit multi-camera constraint. Lichti et al. (2020) presented constraints to enforce the stability of relative orientation. The relative orientations were derived from exterior orientations for each camera in the rig and the differences were minimized at different rig locations. Their approach improved the calibration for a multi-camera mobile mapping system. In the work proposed by Rupnik et al. (2017), they computed the average relative orientation of the cameras from initial SfM similar to (Schonberger & Frahm, 2016; Rupnik et al., 2017). However, they adjusted the exterior orientation of the cameras in BA to achieve constant relative orientation evolved from the average relative orientation. Therefore, it could also be problematic if the initial SfM generated inaccurate results. Other work (Huang, Elhashash, et al., 2022) proposed the baseline constraints which derived the relative orientations from the exterior orientations of the cameras. The method minimized the difference of baselines at adjacent time steps and punished the outliers. The baseline constraints were reported to improve the reconstruction accuracy significantly compared to the unconstrained solutions.

Most of the existing solutions assume either rigorous synchronization, a special frame to mount the cameras, overlapping views to derive the relative orientations, or pre-calibration to obtain the relative orientation to formulate the constraints. In our work, we aim to tackle these challenges and get rid of these requirements.

3. Methodology

Existing constraints (both explicit and implicit) derived from uncalibrated multi-camera systems require pre-computed relative orientation as a starting point. Thus, in order to utilize these constraints, the cameras must share overlapping FoVs. If two camera views do not overlap, these are often treated as separate cameras. To close the gap, our novel motion constraints effectively build connections between two co-located cameras without requiring them to share overlapping FoVs (see Fig. 1), since only the first-order motion between cameras is constrained. Instead of assuming a fixed relative orientation between two co-located cameras, our motion constraints entail the fact that the motion vectors of co-located cameras should follow the same direction and length proportionality. Further, we demonstrate that the proposed motion constraints can be used together with existing approaches (Huang, Elhashash, et al., 2022), to extend to multi-camera systems with an arbitrary number of cameras, with and without overlapping FoVs. In this case, it formulates the most comprehensive constraints respecting the nature of multi-camera co-location. The next two subsections introduce the mathematical formulation of the motion constraints in a two-camera basic scenario (**Section 3.1**) and a scenario with more than two cameras (**Section 3.2**).

3.1. The proposed motion constraints

As shown in Fig. 1**(a)**, a steadily moving platform (e.g., a vehicle) is mounted with two co-located cameras (green and yellow). We assume the motions of these two cameras are parallel to each other in each short period of time (e.g., one second) during the data collection. The assumption still holds to some extent when the camera system turns around a corner, as the motion vectors under our assumption could be numerical approximated of the tangent vector (or a secant), which would still be parallel to each other in such a circumstance. A few facts can safeguard the use of the constraints in general motion scenarios: first, very often the turning radius of a vehicle in ground motion is much larger than the distance between the cameras, thus the effect of curved motion is insignificant; second, our proposed parallelism and length proportionality constraints were used as soft constraints and were adaptive based on the time interval associated with the motion vector. Therefore, conditioning the BA based on this can augment the geometric compliance respecting this fact. These two cameras, illustrated as cameras i and j may or may not share overlapping FoVs. The motion constraints can be interpreted as the following: for time point s, the formed motion vector of the camera i with respect to the previous time point $s-m$, denoted as $\boldsymbol{t}_{i,s-m}$, is parallel to the motion vector $\boldsymbol{t}_{j,s-m}$ formulated with camera j. The motion vector with respect to the next time point can be similarly formulated, denoted as $\boldsymbol{t}_{i,s+m}$ and $\boldsymbol{t}_{j,s+m}$. Here m is a variable about time intervals controlling the motion vector formulated based on short- or long-range dependences. Further, their moving speed (first-order motion difference with respect to time) should be similar. However, given that these two cameras may not have shared FoVs, the constraints should be formulated as a scale-invariant term. Hence, we introduce the use of length proportionality between these motion vectors that ratios out the scale.

As shown in Fig. 1**(b)**, the incremental SfM pipeline will generate two separate models for two cameras without content overlap, and the reprojection errors of both cameras are formulated separately in the standard BA framework (without motion constraints) during the reconstruction stage. After the reconstructions of the two models are done, they are fed together into our proposed BA framework enhanced with the motion constraints to jointly minimize the reprojection errors

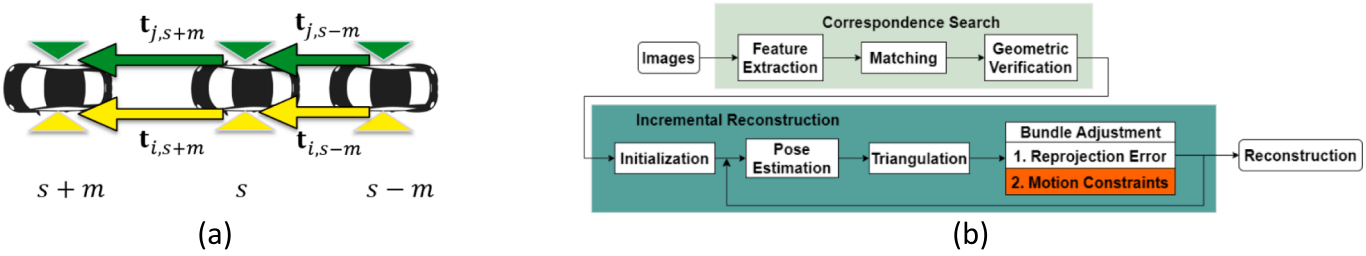


Fig. 1. The proposed motion constraints (a) into a typical BA framework (b). Details of the motion constraints in (a) are explained in the text.

and motion constraint errors for both models. The motion constraint errors serve as the link between the optimizations of reprojection errors of both cameras, as the optimizations of motion constraint errors formulated by motion vectors from both cameras guide the optimizations of the reprojection errors of both cameras and vice versa. To be more specific, the proposed motion constraints contain two factors: first, the motion speed between the two cameras should be similar or up to a scale; second, motion vectors should be directionally parallel to each other. Based on these two factors, we derive three error terms, 1) length proportionality term E_{prop}, and 2) motion cross-product error term, E_{cross}, and 3) motion dot-product error term, E_{dot}. E_{prop} corresponds to the first factor, i.e., the scale-invariant motion speed constraint; E_{cross} and E_{dot} respectively impose that the motion vectors should be parallel and should be pointing in the same direction. The sum of three error terms constructs the final motion constraints E_{motion} (**Equation (1)**), which is added to the total error E for BA, in addition to the regular reprojection error E_{reproj}.

$$\begin{aligned} E &= E_{reproj} + E_{motion} \\ E_{motion} &= \alpha E_{prop} + \beta E_{cross} + \gamma E_{dot} \end{aligned} \tag{1}$$

It should be noted that the other typical error terms such as for GCPs are not shown in **Equation (1)** but can be assumed when available. Moreover, these different terms are given tunable weights (α, β, γ) to impose only soft constraints directly into the energy term, since in our problem we only assume that these cameras are loosely synchronized, i.e., approximately at a level of one second with clock tuned solely by the GPS time.

Length proportionality error term (E_{prop}): Given two co-located cameras (i and j) capturing images while moving, This term minimizes the ratio differences of the lengths of the motion vectors, such that their speeds are up to scale, as formulated in **Equation (2)**:

$$\begin{aligned} E_{prop} &= \frac{1}{2}\sum_{s=s_{st}}^{s_e}\sum_{m=1}^{n}\rho\left(\left\| w_{s,m}\left(\left\|\widetilde{\boldsymbol{t}}_{i,s+m}\right\|_2 \cdot \left\|\widetilde{\boldsymbol{t}}_{j,s-m}\right\|_2 - \left\|\widetilde{\boldsymbol{t}}_{i,s-m}\right\|_2 \cdot \left\|\widetilde{\boldsymbol{t}}_{j,s+m}\right\|_2\right)\right\|_2^2\right) \\ w_{s,m} &= \frac{1}{m} e^{\frac{s-s_{st}}{s_e - s_{st}}} \end{aligned} \tag{2}$$

where $\widetilde{\boldsymbol{t}}_{i,s+m}$, $\widetilde{\boldsymbol{t}}_{j,s+m}$, $\widetilde{\boldsymbol{t}}_{i,s-m}$, $\widetilde{\boldsymbol{t}}_{j,s-m}$ denote the normalized motion vectors by the maximum length of $\boldsymbol{t}_{i,s+m}$, $\boldsymbol{t}_{j,s+m}$, $\boldsymbol{t}_{i,s-m}$, $\boldsymbol{t}_{j,s-m}$, which are the motion vectors of camera i and j at time point s with respect to its neighboring keyframes (defined as m frames away), and $\| \bullet \|_2$ refers to the L-2 norm. Normalization is performed to make sure E_{prop} has similar scales for different range dependencies controlled by m. **Equation (2)** suggests that these vectors should have an equivalent ratio, i.e., $\frac{\|\widetilde{\boldsymbol{t}}_{i,s+m}\|_2}{\|\widetilde{\boldsymbol{t}}_{j,s+m}\|_2} = \frac{\|\widetilde{\boldsymbol{t}}_{i,s-m}\|_2}{\|\widetilde{\boldsymbol{t}}_{j,s-m}\|_2}$, such that it is invariant to reconstructed models with scale differences. The error is built through the Huber loss function (ρ) with δ set to 4 (Huber, 1992), which is particularly effective in handling outliers. To further enhance the robustness of this error to outliers, we aggregate this proportionality error using variable time interval m, meaning that we select neighboring m frames to compute the motion vectors. $w_{s,m}$ is an adaptive weight that determines the contribution of each proportionality error based on the time interval and how far they are from the first frame of the reconstruction. This is formulated through two factors: 1) $\frac{s-s_{st}}{s_e-s_{st}}$, which is positively correlated to the distance between the current frame s and the starting frame s_{st} of the reconstruction ($s-s_{st}$), normalized by the entire collection trace (s_e-s_{st}), where s_e denotes the ending frame of the reconstruction; 2) $\frac{1}{m}$, which is inversely proportional to the time interval m. The first factor considers giving higher weight on this term as the error accumulates as the frame progresses through the reconstruction of the trajectory, and the second factor considers giving lower weight for motion vectors calculated using a larger time interval, as the numerical differentiation has a higher error as the interval enlarges. Finally, we place a global and constant weight α to leverage the importance of this entire error term, which is empirically set to 10^2 in the experiments. This value is set based on the sensitivity analysis (see **Section 4.6**).

Motion cross-product error term (E_{cross}): As mentioned earlier, E_{cross} enforces parallelism of motion vectors between separate cameras. Assuming two co-located cameras i and j, E_{cross} is defined as **Equation (3)**:

$$E_{cross} = \frac{1}{2}\sum_{s=s_{st}}^{s_e}\sum_{m=1}^{n}\rho\left(\|w_{s,m}\|\widehat{\boldsymbol{t}}_{i,s+m} \times \widehat{\boldsymbol{t}}_{j,s+m}\|_2\|_2^2\right) \tag{3}$$

where $\widehat{\boldsymbol{t}}_{i,s+m}$ and $\widehat{\boldsymbol{t}}_{j,s+m}$ are unit-length motion vectors for $\boldsymbol{t}_{i,s+m}$ and $\boldsymbol{t}_{j,s+m}$. $w_{s,m}$ is the same adaptive weight as defined in **Equation (2)**. Similar to α, β is a global weight of this error term leveraging its importance, which is empirically set to 10^5 in the experiments based on the sensitivity analysis (see **Section 4.6**).

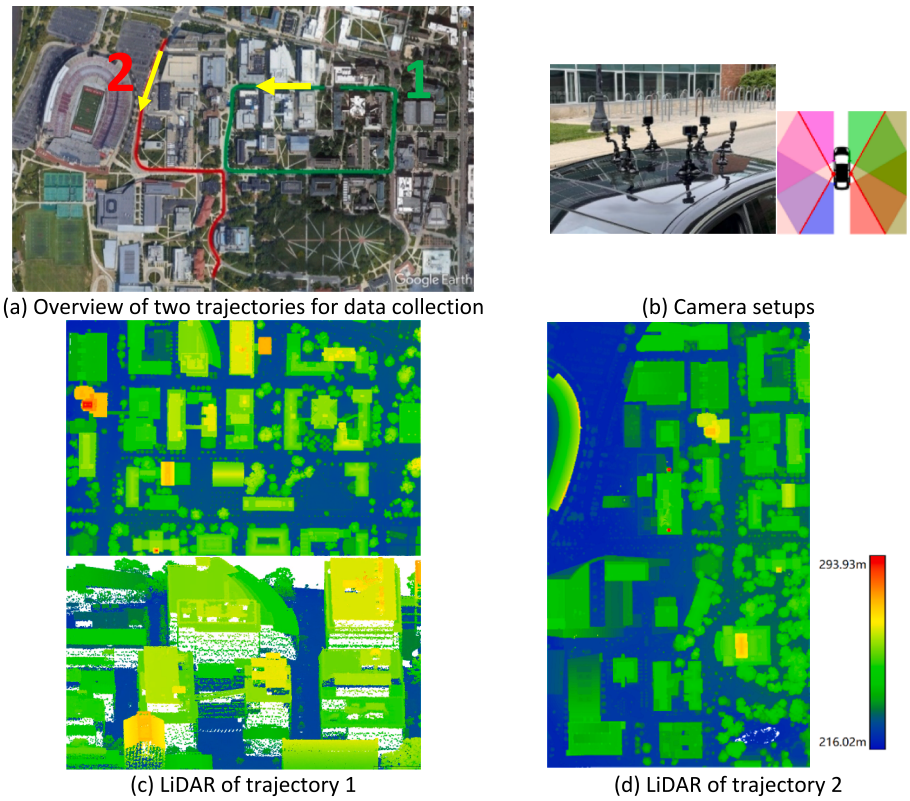
Motion dot-product error term (E_{dot}): It is possible that two motion vectors traveling in opposite directions may still lead to a small E_{cross}, hence we use E_{dot} as a supplemental constraint enforcing the directions of the motion vector to be equivalent. E_{dot} computes the dot product between $\widehat{\boldsymbol{t}}_{i,s+m}$ and $\widehat{\boldsymbol{t}}_{j,s+m}$, as described in **Equation (4)**, where "1″ is a constant enforcing the direction of the two vectors to coincide.

$$E_{dot} = \frac{1}{2}\sum_{s=s_{st}}^{s_e}\sum_{m=1}^{n}\rho\left(\|w_{s,m}(\widehat{\boldsymbol{t}}_{i,s+m}\cdot\widehat{\boldsymbol{t}}_{j,s+m} - 1)\|_2^2\right) \tag{4}$$

$w_{s,m}$ is similarly defined as the other two error terms, and γ is a constant weight indicating the contribution of this term, empirically set as 10^1 in the experiments, which is also based on the sensitivity analysis (see **Section 4.6**).

3.2. Application of motion constraints to multi-camera systems

The above-described motion constraints, due to their flexibility of not requiring overlapping FoVs, can be easily extended to a system that contains an arbitrary number of cameras. As an example, Fig. 2**(b)** contains mixed co-located cameras with and without overlapping FoVs, and the extension of our motion can be simply an enumeration of possible pairs of cameras (i and j) as described in **Equation (5)**:



(a) Overview of two trajectories for data collection (b) Camera setups

(c) LiDAR of trajectory 1 (d) LiDAR of trajectory 2

Fig. 2. Overview of data collection and LiDAR datasets. (a) Two moving trajectories of the vehicle. Yellow arrows point to the moving directions. (b) cameras "casually" mounted on the vehicle, with two groups of each three cameras, and no overlapping FoVs between groups. Two cameras facing left and right are highlighted in red lines for their FoVs, which were used to test the two-camera case. (c-d) Reference LiDAR point clouds color-coded by height based on the height scale bar in (d). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$$E_{motion} = \sum_i \sum_j \left(\alpha E^{ij}_{prop} + \beta E^{ij}_{cross} + \gamma E^{ij}_{dot} \right) \tag{5}$$

Further, when possible, we update the $w_{s,m}$ defined in these error terms (**Equation (2–4)**), to include any possible prior information about the distance between two cameras, as $w_{s,m}{}' = \lambda^{ij} w_{s,m}$. λ^{ij} weights the camera heavier if camera i and j are closer. These weights can be decided empirically, otherwise as equivalent if no prior information is given.

4. Experiments

We first performed experiments with two self-collected datasets under two types of camera configurations (two-camera and multi-camera (six) cases). The accuracy of the reconstruction was evaluated against LiDAR reference data. Additionally, we performed experiments on the KITTI-odometry (Geiger et al., 2012) and KITTI-360 datasets (Liao et al., 2022) to evaluate the absolute pose error (APE) against the provided ground truth poses. **Section 4.1** briefly introduces the two datasets we collected and the reference LiDAR point clouds, and the KITTI-odometry and KITTI-360 datasets. **Section 4.2** and **Section 4.3** describe the qualitative and quantitative evaluations of reconstruction accuracy for the two-camera case and the multi-camera cases, respectively. **Section 4.4** and **Section 4.5** describe the qualitative and quantitative evaluations of pose accuracy using the KITTI-odometry and KITTI-360 datasets. A sensitivity test for each error term of motion constraints is discussed in **Section 4.6**. An ablation study was also performed to understand the contribution of each component of the motion constraints, which is discussed in **Section 4.7**.

To evaluate the reconstruction accuracy using our self-collected datasets, we first ran the SfM pipeline with and without the motion constraints, followed by a dense reconstruction using the open-source OpenMVS library (Cernea, 2022). For all the experiments of the multi-camera system with six cameras, we also incorporated our previously proposed baseline constraints (Huang, Elhashash, et al., 2022) to leverage the advantages of cameras with overlapping FoVs. The baseline constraints build an error term that minimizes the differences of baseline lengths of stereo cameras in different collection time, which can be used on camera pairs that share overlapping FoVs. The dense results were then metrically registered to the LiDAR point clouds using iterative closest point (ICP) algorithm (Besl & McKay, 1992; Xu, Qin, & Song, 2023) initiated by a few manually selected reference points, in which we considered rotation, translation, and scaling. The quantitative evaluation was performed by measuring the mean absolute error (MAE) between the dense point clouds and reference LiDAR point clouds. MAE is derived by a slightly modified chamfer distance, which is the mean of absolute distances between each point in the dense point clouds and the quadric fitted surface using the nearest 6 points from the reference LiDAR point clouds. It is noted that such an evaluation metric is not derived by the correct correspondences between the dense point clouds and the reference LiDAR point clouds due to the lack of color and sparseness of the LiDAR point clouds. However, it still reflects the level of non-rigid distortion as the lower bound errors. To have a more comprehensive understanding of evaluation, we also picked several corresponding subsections from the dense points and LiDAR point clouds. The misalignment was then measured as a supplementary evaluation to emphasize the improvement quantitatively. The registration and quantitative evaluation were performed using the open-source CloudCompare software (Girardeau-Montaut, 2022).

To evaluate the pose accuracy using the KITTI-odometry and KITTI-360 datasets, we first ran the SfM pipeline with and without the motion constraints. A tool named evo (Grupp, 2017) was used to evaluate and compare the poses from the SfM pipeline to the ground truth poses provided in the KITTI-odometry and KITTI-360 datasets. APE was

adopted as the metric to evaluate the global consistency of a trajectory. As defined in **Equation (6)**, APE is based on the absolute relative pose between the reference and estimated poses $P_{ref,i}, P_{est,i} \in SE(3)$ at timestamp i (Lu & Milios, 1997):

$$E_i = P_{ref,i}^{-1} P_{est,i} \in SE(3) \tag{6}$$

Since our motion constraints were formulated as translation constraints into the BA, only the translation part of E_i was used to compute the APE, which is defined as follows:

$$APE_i = \|trans(E_i)\| \tag{7}$$

Before evaluating the APE, the poses from the SfM pipeline were aligned to the ground truth by using least square-based Umeyama alignment algorithm (Umeyama, 1991), including rotation, translation, and scaling. The alignments and APE derivation were done using the evo tool.

4.1. Dataset

OSU image datasets. Two datasets were self-collected on part of The Ohio State University (OSU) campus, as shown in Fig. 2**(a)**: trajectory 1 consisted of data in a loop (without closing it) and trajectory 2 consisted of data with forward motion (open trajectory). Six GoPro cameras were mounted on a vehicle with a configuration shown in Fig. 2**(b)**, two of which (facing left and right, as their FoVs highlighted in red lines) were used to test the basic two-camera case as described in **Section 3.1**. Data from all the cameras were then used to evaluate the motion constraints in the multi-camera case as described in **Section 3.2**. We considered this as a "casual" setup because it had cameras with both overlapping and non-overlapping FoVs, and no special mounting rig was used. The videos were recorded at a frame rate of 30 FPS, and we uniformly extracted 1/6 of the video frames to constitute the image datasets, thus the time interval between consecutive frames was about 0.2 s. The video clocks were synchronized via GPS time with an estimated synchronization error of less than one second. The resolution of the images was downsampled by half to 2000 × 1500. The detailed information is provided in Table 1.

OSU LiDAR datasets. The high-resolution LiDAR point clouds were collected in 2015 for the City of Columbus, Ohio as part of (Ohio Statewide Imagery Program (OSIP)). The LiDAR data was collected by a Leica ALS70 LiDAR system onboard aircraft with a nominal pulse spacing (NPS) of 0.7 m. The horizontal accuracy is 1.182 m at a 95 % confidence level, with an average density of 5.76 pts./m^2. We prepared the LiDAR dataset covering the trajectories as shown in Fig. 2**(c-d)**. Although it is airborne LiDAR, the façade points are sufficient for evaluation as shown in Fig. 2**(c)**.

KITTI-odometry and KITTI-360 datasets. KITTI-odometry (Geiger et al., 2012) and KITTI-360 datasets (Liao et al., 2022) are large-scale suburban driving datasets for various tasks such as semantic scene understanding, novel view synthesis, and SLAM. It contains rich information from multiple sensors. KITTI-odometry dataset contains 11 trajectories with ground truth poses available and it has two perspective stereo cameras to the front, while KITTI-360 dataset contains 9 trajectories with ground truth poses available and it has two perspective stereo cameras to the front and two fisheye cameras to each side. The ground truth poses for the camera frames are derived from the GPS/IMU measurements. The detailed information is provided in Table 1.

4.2. Evaluation of motion constraints in two-camera case

Qualitative evaluation. Fig. 3 shows the visual comparison of the reconstruction results with and without applying the motion constraints for two trajectories. It should be noted that since there are no overlapping FoVs between these two cameras, the reconstructed models are in separate coordinate systems but here they are co-located for visualization through similarity transformation, including rotation, translation, and scaling. Here we hide the reference LiDAR point clouds in Fig. 3 to better emphasize the distortion and correction visually, and show the visual comparison to LiDAR point clouds in Fig. 4 for four manually selected subsections. Thanks to our motion constraints, the reconstructions are mutually constrained by the motion vectors from each other. As can be seen in Fig. 3**(a)**, there is a noticeable drift at the end of the trajectory for trajectory 1, in both horizontal and vertical directions: from the bird-eye view, one model tends to drift outwards while the other inwards due to the doming effect caused by accumulated errors (James & Robson, 2014). The results of our proposed motion constraints have significantly improved the drift (Fig. 3**(b)**). Although both cameras do not share overlapping FoVs, the process can be understood as that each camera provides an approximate motion vector that guides the BA, thus it can effectively prevent the cameras from deviating from each other at each time step and cancels out the drift for both models. Results of trajectory 2 in Fig. 3**(c-d)** show similar improvement for BA with the proposed motion constraints, as shown in the focused region outlined in the rectangle where parallel streets tend to diverge for reconstruction without the proposed motion constraints. The evaluations also indicate that our motion constraints can handle the case of curve motion, as can be seen at the corners of both trajectories. It should be noted that our motion constraints do not register 3D models from these two cameras under the same coordinate frame, rather, they utilize the parallelism and length proportionality constraints from each other to correct the topographical distortions of their respective geometries. The distortion for reconstruction without motion constraints and the correction for reconstruction with motion constraints can be further emphasized by comparing the four manually selected subsections from the dense point clouds to the corresponding subsections from LiDAR point clouds, as shown in Fig. 4. These subsections are selected from building façades. Among these subsections, the ones with motion constraints are better aligned with the corresponding subsections from LiDAR point clouds.

Quantitative evaluation. The quantitative evaluation was carried out by two means: 1) measuring the MAE between the whole dense reconstruction results and the reference LiDAR point clouds; 2) measuring the misalignments between the corresponding subsections from dense reconstruction results and the reference LiDAR point clouds. We consider that the reference LiDAR data are more metrically accurate. As mentioned earlier, the dense point clouds were registered to the LiDAR data prior to evaluation. As demonstrated in Table 2**,** reconstructions with motion constraints resulted in a substantial improvement in the MAE and standard deviation of both datasets. The

Table 1
Information of OSU image datasets, KITTI-odometry and KITTI-360 datasets.

Dataset ID	GPS length [m]	Driving direction	Camera configuration	Video frames	Resolution	Time interval [s]
OSU Trajectory 1	1200	Anticlockwise	2 cams	1400	2000 × 1500	0.2
			6 cams	4200		
OSU Trajectory 2	860	Southward	2 cams	1060		
			6 cams	3180		
KITTI-odometry, 11 Trajectories	22,179	Various	2 cams	11,614	1226 × 370	0.2 – 0.3
KITTI-360,	66,591	Various	4 cams	72,154	1408 × 376	0.2 – 0.3
9 Trajectories					1400 × 1400	

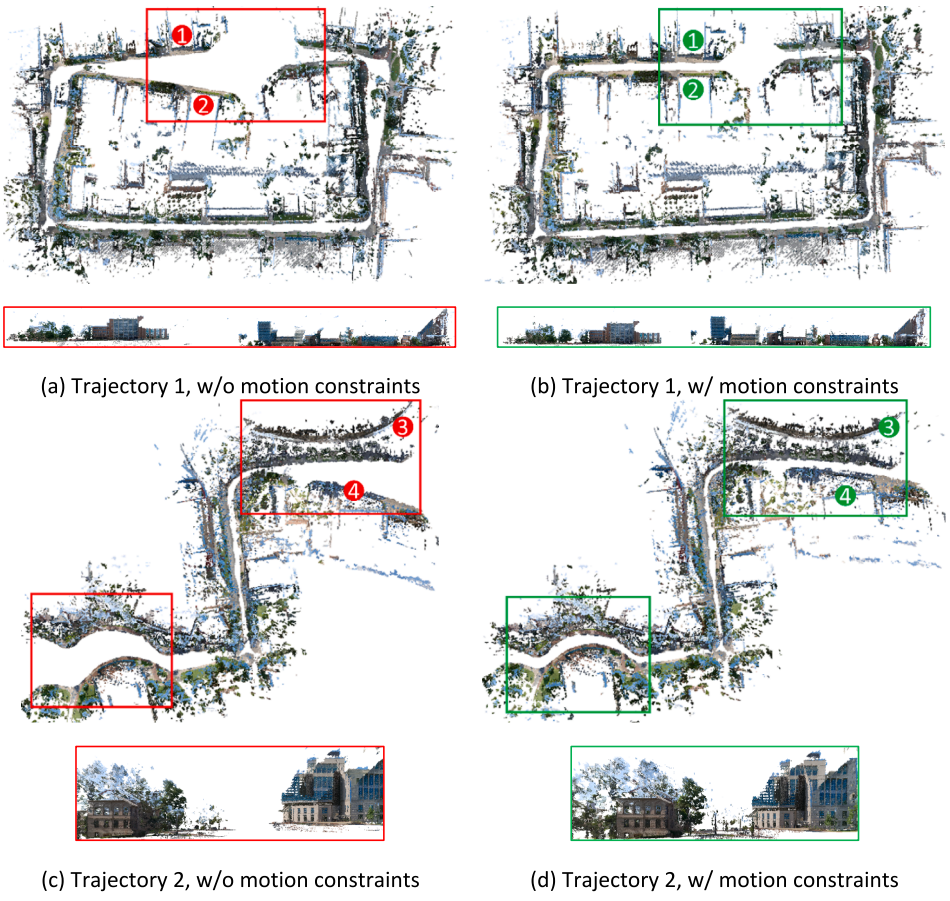


Fig. 3. Visual comparison of the whole reconstructions with and without motion constraints. The red rectangle in (a) shows the drifts in both horizontal (top) and vertical (bottom) directions for the reconstruction without motion constraints, which are reduced after using the motion constraints, as outlined in the green rectangles in (b). The red rectangles in (c) outline the drift regions at both ends of the trajectory, in both horizontal (top) and vertical (bottom) directions. It is mitigated after using motion constraints as outlined in the green rectangles in (d). The numbers indicate different subsections of distortion (indexed by red numbers) and correction (indexed by green numbers), which are visually compared to the corresponding subsections from the LiDAR point clouds as shown in Fig. 4. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

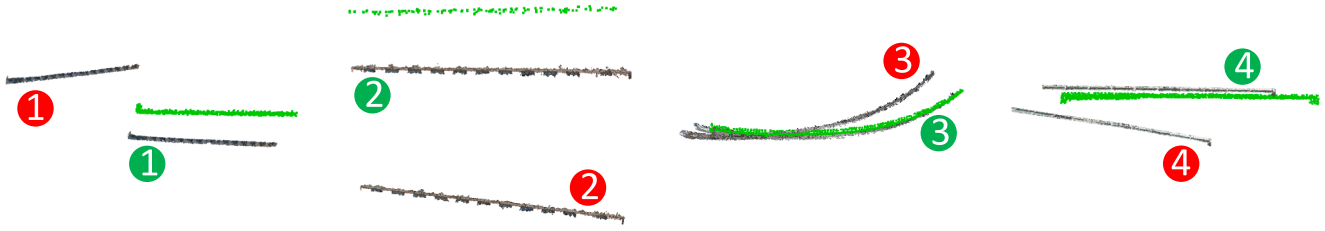


Fig. 4. Visual comparison showing a top view of four selected subsections of the generated reconstruction results against the LiDAR as a reference. The subsections before and after applying the motion constraints are indexed by red and green numbers, respectively. The drift and distortions are reduced after applying the motion constraints leading to better aligned results with LiDAR point clouds. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

improvement is up to 0.97 m (42.46 %) for MAE and 0.91 m (28.48 %) for standard deviation for trajectory 1, and up to 0.92 m (47.53 %) for MAE and 1.13 m (45.26 %) for standard deviation for trajectory 2. While MAE of the whole reconstruction indicates the lower bound errors, the misalignments of the subsections further reveal the significant improvement brought by the motion constraints. The improvement is up to 12.90 m (73.43 %) for MAE and 7.30 m (79.15 %) for standard deviation for subsections in trajectory 1, and up to 11.34 m (86.12 %) for MAE and 4.91 m (80.00 %) for standard deviation for subsections in trajectory 2.

4.3. Evaluation of motion constraints in multi-camera case

In this experiment, we used the images from all six cameras and applied the aggregated constraints among each pair of cameras (as described in Section 3.2). The previously proposed baseline constraints were also incorporated in all the experiments. Both the qualitative and quantitative results are shown as follows.

We first compared the reconstruction results without motion constraints between two-camera and multi-camera cases. For the multi-camera case, baseline constraints were applied additionally. Visual

Table 2
Statistics of the reconstruction accuracy in the two-camera case. For each trajectory, two separate models were reconstructed from the left and right cameras and evaluated respectively, which we call models #1 and #2 in the table. "Sub" refers to the subsections of reconstruction, indices of which are shown in Figs. 3 - 4.

Dataset	model	MAE [m]		Standard deviation [m]	
		w/o motion constraints	w/ motion constraints	w/o motion constraints	w/ motion constraints
Trajectory 1	#1	2.277	**1.310**	3.206	**2.293**
	#2	1.952	**1.403**	2.384	**2.056**
	Sub #1	17.565	**4.667**	9.227	**1.924**
	Sub #2	22.343	**7.073**	7.886	**2.310**
Trajectory 2	#1	1.939	**1.017**	2.485	**1.360**
	#2	1.083	**0.777**	1.711	**1.188**
	Sub #3	5.976	**1.107**	6.145	**1.229**
	Sub #4	13.172	**1.828**	6.825	**2.349**

inspection shows that the reconstruction of six cameras for both trajectories has a better appearance in terms of scene coverage and drifts at the ends of both trajectories (outlined in red rectangles in Fig. 3**(a, c)** and Fig. 5**(a, c)**), thanks to the advantages of multi-camera systems that provide extended FoVs and tighter camera networks. The results of the visual inspection are also correspondingly reflected in the statistical results, which will be discussed later.

Qualitative evaluation. Because the six cameras were mounted in a manner where three of the six cameras were on one side and the other three were on an opposite side, there is a natural lack of overlapping FoVs between cameras from each side, thus it will produce two separate models mapping different side of the street. Fig. 5 shows similar observations demonstrated in the two-camera case (Fig. 3). First of all, the six-camera reconstruction without motion constraints naturally shows better geometry (i.e., less drift) than that generated from a two-camera case (e.g., comparing Fig. 3**(a)** with Fig. 5**(a)**), thanks to the redundant cameras with overlapping FoVs. However, drifts still exist as shown in the red rectangle region in Fig. 5**(a)** in the horizontal direction while being slightly better in the vertical direction. Our proposed approach can improve such errors to a notable level especially in the horizontal direction, as seen in Fig. 5**(b).** For trajectory 2, we can observe notable differences in results with and without our proposed motion constraints,

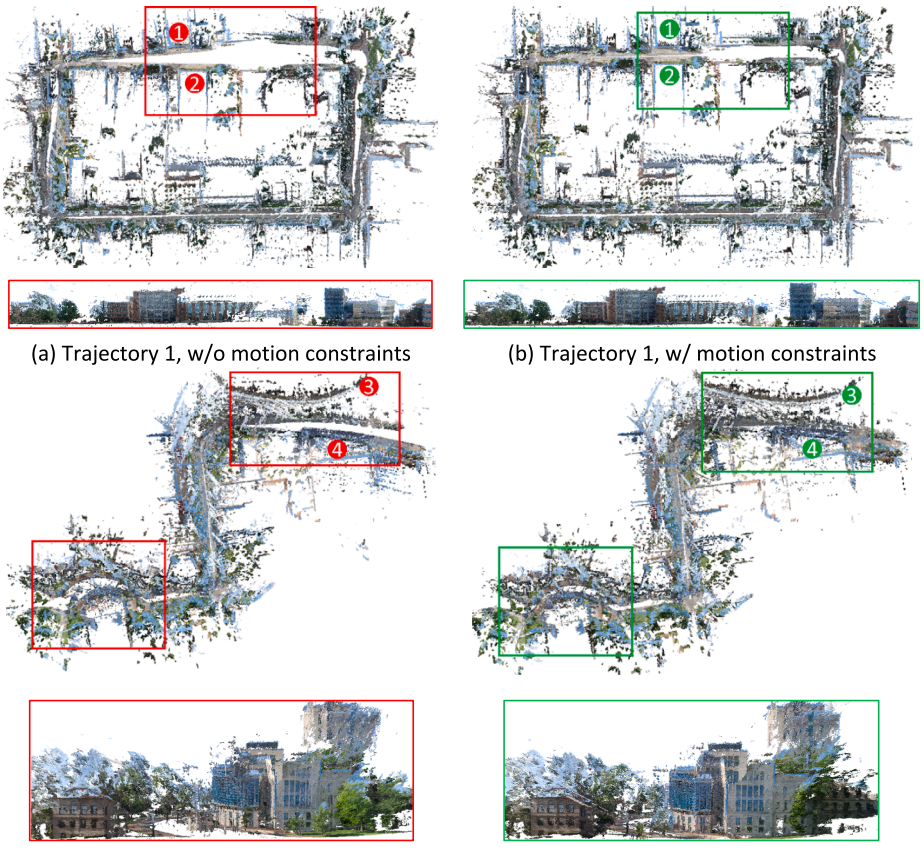


Fig. 5. Visual comparison of the whole reconstruction with and without motion constraints. The red rectangles in (a) show the drift mainly in the horizontal direction (top) and less in the vertical direction (bottom) for the reconstruction without motion constraints, which is reduced after using the extended motion constraints, as outlined in the green rectangles in (b). The red rectangles in (c) also show the drift mainly in the horizontal direction (top) and less in the vertical direction (bottom) at both ends of the trajectory. It is mitigated with extended motion constraints, as outlined in the green rectangles in (d). It is worth noting that the reconstruction in the multi-camera case has more points than in the two-camera case. The numbers indicate different subsections of distortion (indexed by red numbers) and correction (indexed by green numbers), which are visually compared to the corresponding subsections from the LiDAR point clouds as shown in Fig. 6. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and the reader may focus on the rectangle regions where distorted point clouds are corrected after using our proposed approach. Similar observations for visual comparison of the same four corresponding subsections from dense point clouds and LiDAR point clouds were found in Fig. 6. The subsections of six-camera reconstructions with and without motion constraints show better alignment than those of two-camera reconstructions, among which the ones with motion constraints are even closer to the corresponding subsections from the LiDAR point clouds.

Quantitative evaluation. Table 3 shows the statistical comparison between the dense results with and without the proposed motion constraints. For trajectory 1, the proposed motion constraints achieve up to 0.06 m (5.11 %) improvement for MAE and 0.06 m (2.72 %) for standard deviation. For trajectory 2, motion constraints achieve statistically up to 0.53 m (36.70 %) improvement for MAE and 1.10 m (45.93 %) for standard deviation. As for the misalignments of subsections, the improvement is up to 8.08 m (81.50 %) for MAE and 4.63 m (73.90 %) for standard deviation for subsections in trajectory 1, and up to 3.58 m (76.00 %) for MAE and 2.82 m (67.98 %) for standard deviation for subsections in trajectory 2. This level of improvement is in line with the qualitative evaluation but also suggests that the improvement is more significant for cases with fewer cameras (e.g., the two-camera case).

Table 3
Statistics of the reconstruction accuracy in the multi-camera case. For each trajectory, two separate models were reconstructed from the left and right camera groups and evaluated respectively, which we call models #1 and #2 in the table. "Sub" refers to the subsections of reconstruction, indices of which are shown in Figs. 5 - 6.

dataset	model	MAE [m]		Standard deviation [m]	
		w/o motion constraints	w/ motion constraints	w/o motion constraints	w/ motion constraints
Trajectory 1	#1	1.078	**1.022**	2.187	**2.127**
	#2	1.398	**1.364**	2.185	**2.147**
	Sub #1	9.915	**1.834**	6.271	**1.637**
	Sub #2	2.210	**0.766**	0.778	**0.615**
Trajectory 2	#1	1.434	**0.908**	2.388	**1.291**
	#2	0.702	**0.631**	**1.227**	1.238
	Sub #3	3.612	**0.946**	4.148	**1.328**
	Sub #4	4.716	**1.132**	2.922	**1.373**

4.4. Evaluation of motion constraints in KITTI-odometry and KITTI-360 datasets

Table 4 shows the statistical results for the APE evaluation of poses with and without motion constraints for both KITTI-odometry and KITTI-360 datasets. It should be noted that both datasets have overlapping FoVs among all the cameras, where the KITTI-odometry collection came from two cameras while KITTI-360 data came from four (better connected). We can see that the improvement on KITTI-odometry dataset is much more significant than that on KITTI-360 dataset, with update to 28.82 m (81.05 %) improvement on Trajectory 01 and an average improvement of 4.48 m (24.39 %) for all the trajectories in KITTI-odometry dataset. For KITTI-360 dataset, because the four cameras are better connected with strong correspondences, the improvement of RMSE of APE is rather marginal: up to 0.14 m (4.28 %) for Trajectory 03 and the average improvement is only 0.18 m (0.26 %). This mere improvement by our motion constraint is also affected by several challenging such as trajectory 05 and 07, due to a section of frames full of moving vehicles (an example is shown in Fig. 7).

The quantitative results indicate that our motion constraints work more effectively in short (<3km) and regular (smooth turns, no revisiting) trajectories (e.g., Trajectory 01, 03, 04, 06, 09, 10 in KITTI-odometry dataset). Fig. 8 shows the visual comparison of the poses with and without our motion constraints compared to the ground truth poses for Trajectory 01 and 09 in KITTI-odometry dataset. The trajectories of poses with motion constraints are better aligned with the ground truth poses, especially at the beginning and the end of the trajectories.

The results also indicate that for long trajectories with complicated road condition, the benefit brought by our motion constraints is less noticeable, e.g., Trajectory 02 in KITTI-odometry dataset and most trajectories in KITTI-360 dataset. These trajectories cover longer distances and contain more turns and revisiting of some sections of road. Additionally, the traffic condition in KITTI-360 is much busier with more moving objects in the scenes, thus the initial reconstruction (before applying motion constraints) is relatively poorly determined (see Fig. 7). Our motion constraints improve upon initial reconstructions, in cases where poor camera network is generated, the fraction of improvement of the motion constraints (probably built on poorly determined initial geometry), is also impacted. On the other hand, our motion constraints are designed to work best for co-located cameras that do not share overlapping FoVs, e.g., side-looking cameras at opposite directions (explained with experiments shown in **Section 4.2**). As we explained earlier, the KITTI-360 dataset, however, has co-located cameras that share overlapping FoVs, wherein the features connecting these four cameras kick in. While our motion constraints still show benefits in these cases, it is less.

4.5. Loop closure and incremental incorporation of motion constraints

Generally, the reconstruction can be easily improved when common strategies such as loop closure (when available) or more sophisticated ones such as an incremental bundle adjustment (incremental BA incorporating the motion constraint) are considered. As an example, we tested on KITTI-360 Trajectory #5, where, for the purpose of this testing, we removed the section of frames with heavy moving traffic (red box in Fig. 7). In this experiment, we included: 1) loop closure among intersecting frames; 2) incrementally incorporate the motion constraints

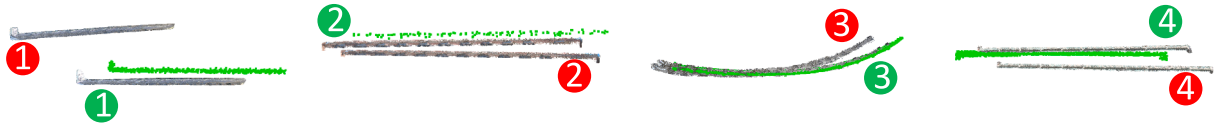


Fig. 6. Visual comparison showing a top view of four selected subsections of the generated reconstruction results against the LiDAR as a reference. The subsections before and after applying the motion constraints are indexed by red and green numbers, respectively. The drift and distortions are reduced after applying the motion constraints leading to better aligned results with LiDAR point clouds. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 4
RMSE of APE (m) for KITTI-odometry and KITTI-360 datasets, with and without motion constraints. Bold results indicate the trajectories with most improvements in each dataset.

KITTI-Odometry Trajectory#	00	01	02	03	04	05	06	07	08	09	10	Ave
w/o motion constraints	25.80	35.56	41.08	2.29	0.71	20.38	19.20	5.78	23.21	22.74	5.30	18.37
w/ motion constraints	25.75	**6.74**	40.38	1.22	0.36	19.71	14.50	5.05	23.12	12.02	3.99	13.89
KITTI-360 Trajectory#	**00**	**02**	**03**	**04**	**05**	**06**	**07**	**09**	**10**	**Ave**		
w/o motion constraints	70.62	67.55	3.27	8.14	156.58	32.10	176.97	80.63	17.41	68.14		
w/ motion constraints	70.10	67.42	**3.13**	8.00	156.57	32.07	176.88	80.56	16.90	67.96		

Fig. 7. Left: reconstruction results of KITTI-360 Trajectory #5. The region where the pose estimation becomes problematic is outlined in red box. Right: an example image in the problematic region that causes incorrect pose estimation. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

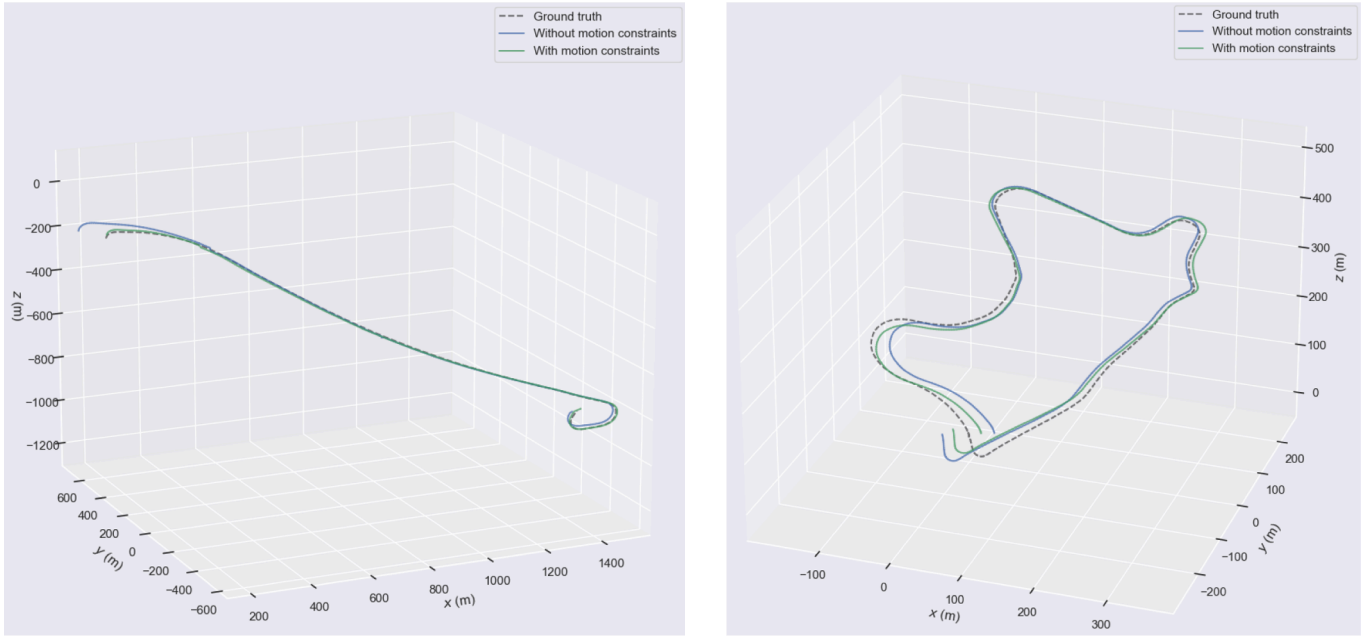


Fig. 8. Visual comparison of poses with and without motion constraints to the ground truth poses. (a) and (b) show trajectories with significant improvement in KITTI-odometry dataset. The blue lines represent the trajectories of poses without motion constraints. The green lines represent the trajectories of poses with motion constraints. The dashed lines represent the trajectories of the ground truth poses. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

during the reconstruction. The results are shown in Table 5. As expected, we can see that both strategies can notably improve the accuracy in cases where the cameras are already well connected (KITTI-360 data): namely up to 0.3 m of improvement with incremental motion constraint strategy, 0.2 m of improvement with loop closure, and 0.5 m combined. This shows that more sophisticated strategies can easily catalyze the effectiveness of our proposed motion constraints in its ability to improve the reconstruction accuracy.

4.6. Sensitivity analysis on different error terms

A sensitivity analysis is conducted to evaluate the contribution of different error terms controlled by the weight parameters (introduced in **Section 3.1**) on the reconstruction accuracy. The two-camera dataset of trajectory 1 was used for the sensitivity test. The weight was determined as those achieved the smallest MAE by grid search, which were at the order of 10^2 for proportionality parameter α, 10^5 for cross-product parameter β, and 10^1 for dot-product parameter γ. To understand the

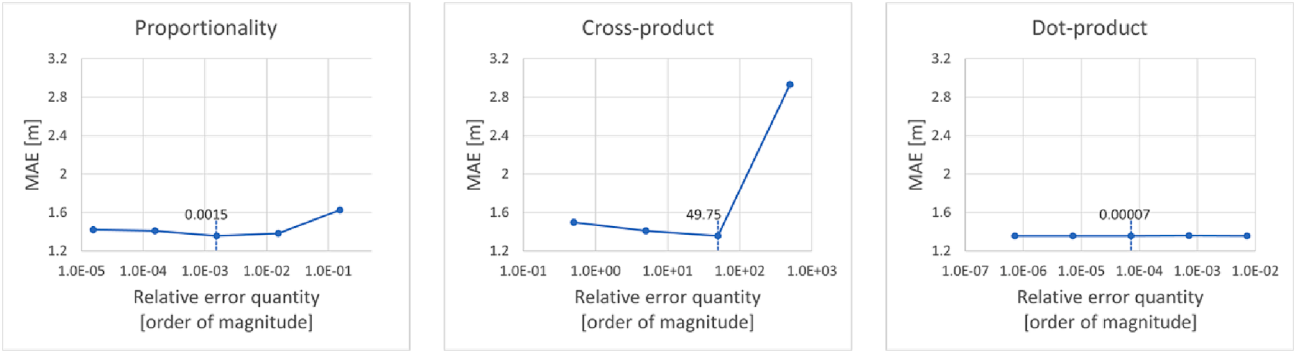


Fig. 9. Sensitivity analysis on different error terms. The x-axis refers to the relative error quantity to the reprojection errors. The x-axis is presented on a base-10 log scale. The y-axis refers to the reconstruction accuracy in terms of MAE. The numbers above the minimum points of the lines indicate the relative error quantity of each error term with the optimal weight value. Detailed explanation in the text.

Table 5
Statistics results for reconstruction KITTI-360 Trajectory #5 without the section with heavy traffic (red box in Fig. 7).

	RMSE of APE [m]
No loop closure, no motion constraint	3.94
No loop closure, motion constraint	3.92
No loop closure, incremental motion constraint	3.66
Loop closure, no motion constraint	2.41
Loop closure, motion constraint	2.23
Loop closure, incremental motion constraint	1.93

sensitivity of each error term, we adjusted the weight of each error term and evaluated the performance of our method. In Fig. 9, the performance is shown by varying the weight of each error term, such that the actual quantity of each error term varies from 0.00001 % to 100000 % of the reprojection error budget (we call this percentage relative error quantity), as reflected at the x-axis of the figures presented on a base-10 log scale. As indicated in the middle of Fig. 9, when set to the optimal weight values (determined by grid search), the cross-product errors constitute the majority of the BA problem, being 50 times (equivalently 5000 %) of the reprojection errors. This gives a strong constraint to stabilize the BA to respect the motion constraint. The proportionality is comparatively, much smaller, at only 0.15 % of the reprojection error, while the dot-product errors are even less (0.00007, equivalently 0.007 % of the reprojection error). The dot-product error has a very minor impact on the error term, while may play roles in regularizing the motion to be in the same direction. Both the proportionality and cross-product terms are relatively robust when their relative error quantity is within an order of magnitude of 10, while the cross-product error term is slightly more sensitive since the error may increase when this error term contributes 100 times more or less of the optimal value. Therefore, the adjustment of the weight parameter, as they follow what is suggested by Fig. 9, may produce the best MAE in the experiment. To obtain the parameter values for other datasets, it is recommended that the users first obtain the quantity for each error term and the reprojection error by setting the weight parameters to the default unit value (e.g., 1), and then follow our weight suggestions based on the recommended percentage relative error quantity values in the sensitivity analysis, which are 5000 % for cross-product errors, 0.15 % for the proportionality errors, and 0.007 % for the dot-product errors.

4.7. Ablation study on different error terms

An ablation study was conducted to understand the contribution of different error terms in our motion constraints. The experiment was also performed on the two-camera dataset of trajectory 1. We first added each error term individually to BA and evaluated the improvement achieved by each error term. Then we gradually added the error terms to BA in descending order of contribution and evaluated their intersection effects. As shown in Table 6, the results indicate that the cross-product error term has the largest contribution, resulting in an improvement of 0.685 m or equivalently 32.40 % compared to the results without motion constraints. The proportionality error and dot-product error achieved 10.17 % and 8.14 % improvement, respectively. By gradually adding the error terms in order of decreasing induvial contribution, the results show that the aggregated terms have a positive impact towards reducing the errors, yet the improvement of adding proportionality errors reduces by 2/3 and the improvement of adding dot-product error becomes marginal, which can be regarded as a supplemental constraint when cross-product errors exist. Dot-product term provides extra enforcement if the non-cooperative case occurs, i.e., motion directions are wrongly estimated in an opposite direction.

5. Conclusion

In this paper, we attempt to address the problem of BA for uncalibrated multi-camera systems to achieve improvement in metric accuracy. By observing the fact that co-located cameras share the same motion, we propose novel motion constraints incorporated into a BA framework to enforce the optimization to respect this fact. A significant difference between our motion constraints, as compared to similar works in the literature, is that our constraints have a high degree of flexibility and do not even require cameras with overlapping FoVs, which allows multi-camera systems with any “casual” setups to benefit from our proposed constraints. Our experiments show that, with cameras not sharing overlapping FoVs that generate separate 3D reconstructions, our proposed constraints can still positively improve the metric accuracy, in that the motion vectors of co-located cameras provide guidance for each other to travel in paralleling speeds and directions. With two datasets containing over 7,000 video frames in total and a LiDAR reference, we experimented with our proposed approach and have shown that this resulted in reconstruction accuracy of up to 11.34 m (86.12 %) of MAE improvement for a two-camera system, and up to 8.08 m (81.50 %) of MAE improvement for a six-camera system. Our motion constraints were also tested with the KITTI-odometry and KITTI-360 datasets to evaluate the pose accuracy, which achieved an

Table 6
Ablation over each component of the motion constraints.

BA	E_{prop}	E_{cross}	E_{dot}	MAE | improvement [-m]
w/o motion constraints	×	×	×	2.114
w/ motion constraints	√	×	×	1.899|-0.215
	×	√	×	1.429|-0.685
	×	×	√	1.942|-0.172
	√	√	×	1.360|-0.754
	√	√	√	1.356|-0.758

improvement of up to 28.82 m (81.05 %) in terms of the RMSE of APE. We also showed that more sophisticated strategies commonly used in structure from motion, such as loop closure (when available) and incremental bundle adjustment incorporating our motion constraints, can further improve the reconstruction accuracy. Our proposed motion constraints are under the context of low-cost mapping, which can be expanded with a broader impact that enables citizen scientists to capture 3D information with improved accuracy. For example, non-experts can casually place multiple co-located cameras (or even smartphone cameras) without the need to go through a rigorous rig calibration to use the stereo capability. It should also be advised that for two or more sets of cameras with non-overlapping FoVs, our proposed motion constraints may still result in separated models, while the motion constraints are able to utilize the trajectory of each other to improve the geometric reconstruction of their respective models. In our future work, we aim to further test the capability of the proposed approach for non-expert user cases to expand the photogrammetry applications.

CRediT authorship contribution statement

Debao Huang: Data curation, Methodology, Writing – original draft, Writing – review & editing, Investigation. **Rongjun Qin:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Writing – review & editing, Supervision, Validation, Visualization, Project administration, Resources, Software. **Mostafa Elhashash:** Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was partially supported by the Office of Naval Research (ONR, Award No. N00014-20-1-2141 and N00014-23-1-2670). The authors would like to acknowledge the Ohio Statewide Imagery Program (OSIP) for making the LiDAR Dataset available. We also thank Xinyi Wu for her assistance in this work.

References

Alshammari, A., & Rawat, D. B. (2019). Intelligent multi-camera video surveillance system for smart city applications. *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*.

Bertin, S., Levy, B., Gee, T., Delmas, P., 2020. Geomorphic change detection using cost-effective structure-from-motion photogrammetry: Evaluation of direct georeferencing from consumer-grade UAS at Orewa beach (New Zealand). Photogramm. Eng. Remote Sens. 86 (5), 289–298.

Besl, P. J., & McKay, N. D. (1992). Method for registration of 3-D shapes. *Sensor fusion IV: control paradigms and data structures*.

Caprile, B., Torre, V., 1990. Using vanishing points for camera calibration. Int. J. Comput. Vis. 4 (2), 127–139.

Cavegn, S., Blaser, S., Nebiker, S., Haala, N., 2018. Robust and accurate image-based georeferencing exploiting relative orientation constraints. ISPRS Ann. Photogramm. Remote Sens. Spatial Inform. Sci. 4 (2).

Cernea, D. (2022). *OpenMVS: Multi-View Stereo Reconstruction Library*. https://cdcseacave.github.io/openMVS.

Cornelis, K., Verbiest, F., Van Gool, L., 2004. Drift detection and removal for sequential structure from motion algorithms. IEEE Trans. Pattern Anal. Mach. Intell. 26 (10), 1249–1259.

Detchev, I., Habib, A., Mazaheri, M., Lichti, D., 2018. Practical in situ implementation of a multicamera multisystem calibration. J. Sens. 2018, 1–12.

Dong, H., Yao, J., Gong, Y., Li, L., Cao, S., Li, Y., 2023. Learning-based encoded target detection on iteratively orthorectified images for accurate fisheye calibration. Photogram. Rec.

Engel, J., Schöps, T., & Cremers, D. (2014). LSD-SLAM: Large-scale direct monocular SLAM. *European conference on computer vision*.

Geiger, A., Lenz, P., & Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. *2012 IEEE conference on computer vision and pattern recognition*.

Girardeau-Montaut, D. (2022). *CloudCompare*. (Version 2.12.4) http://www.cloudcompare.org/.

Grupp, M. (2017). *evo: Python package for the evaluation of odometry and SLAM.* https://github.com/MichaelGrupp/evo.

Häne, C., Heng, L., Lee, G.H., Fraundorfer, F., Furgale, P., Sattler, T., Pollefeys, M., 2017. 3D visual perception for self-driving cars using a multi-camera system: Calibration, mapping, localization, and obstacle detection. Image Vis. Comput. 68, 14–27.

Harmat, A., Trentini, M., Sharf, I., 2015. Multi-camera tracking and mapping for unmanned aerial vehicles in unstructured environments. J. Intell. Rob. Syst. 78 (2), 291–317.

Heng, L., Choi, B., Cui, Z., Geppert, M., Hu, S., Kuan, B., Liu, P., Nguyen, R., Yeo, Y. C., & Geiger, A. (2019). Project autovision: Localization and 3d scene perception for an autonomous vehicle with a multi-camera system. *2019 International Conference on Robotics and Automation (ICRA)*.

Heng, L., Furgale, P., Pollefeys, M., 2015. Leveraging image-based localization for infrastructure-based calibration of a multi-camera rig. J. Field Rob. 32 (5), 775–802.

Heng, L., Lee, G.H., Pollefeys, M., 2015. Self-calibration and visual slam with a multi-camera system on a micro aerial vehicle. Auton. Robot. 39 (3), 259–277.

Huang, D., Elhashash, M., Qin, R., 2022. Constrained bundle adjustment for structure from motion using uncalibrated multi-camera systems. ISPRS Ann. Photogramm. Remote Sens. Spatial Inform. Sci. 2, 17–22.

Huang, D., Tang, Y., Qin, R., 2022. An evaluation of PlanetScope images for 3D reconstruction and change detection–experimental validations with case studies. Gisci. Remote Sens. 59 (1), 744–761.

Huber, P. J. (1992). Robust estimation of a location parameter. *Breakthroughs in statistics: Methodology and distribution*, 492-518.

James, M.R., Robson, S., 2014. Mitigating systematic error in topographic models derived from UAV and ground-based image networks. Earth Surf. Proc. Land. 39 (10), 1413–1420.

Jenal, A., Lussem, U., Bolten, A., Gnyp, M.L., Schellberg, J., Jasper, J., Bongartz, J., Bareth, G., 2020. Investigating the potential of a newly developed UAV-based VNIR/SWIR imaging system for forage mass monitoring. PFG–J. Photogramm. Remote Sens. Geoinform. Sci. 88 (6), 493–507.

Jenal, A., Hüging, H., Ahrends, H.E., Bolten, A., Bongartz, J., Bareth, G., 2021. Investigating the potential of a newly developed UAV-mounted vnir/swir imaging system for monitoring crop traits—A case study for winter wheat. Remote Sens. (Basel) 13 (9), 1697.

Jones, G., Renno, J., & Remagnino, P. (2002). Auto-calibration in multiple-camera surveillance environments. *Third IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*.

Kaya, Y., Şenol, H.İ., Yiğit, A.Y., Yakar, M., 2023. Car Detection from Very High-Resolution UAV Images Using Deep Learning Algorithms. Photogramm. Eng. Remote Sens. 89 (2), 117–123.

Krahnstoever, N., & Mendonca, P. R. (2005). Bayesian autocalibration for surveillance. *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*.

Lerma, J.L., Navarro, S., Cabrelles, M., Seguí, A.E., 2010. Camera calibration with baseline distance constraints. Photogram. Rec. 25 (130), 140–158.

Liao, Y., Xie, J., Geiger, A., 2022. KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. IEEE Trans. Pattern Anal. Mach. Intell. 45 (3), 3292–3310.

Lichti, D.D., Jarron, D., Tredoux, W., Shahbazi, M., Radovanovic, R., 2020. Geometric modelling and calibration of a spherical camera imaging system. Photogram. Rec. 35 (170), 123–142.

Lichti, D.D., Tredoux, W., Maalek, R., Helmholz, P., Radovanovic, R., 2021. Modelling extreme wide-angle lens cameras. Photogram. Rec. 36 (176), 360–380.

Lu, F., Milios, E., 1997. Globally consistent range scan alignment for environment mapping. Auton. Robot. 4, 333–349.

Marcon, M., Sarti, A., Tubaro, S., 2017. Multicamera rig calibration by double-sided thick checkerboard. IET Comput. Vis. 11 (6), 448–454.

Maset, E., Magri, L., Toschi, I., Fusiello, A., 2020. Bundle block adjustment with constrained relative orientations. ISPRS Ann. Photogramm. Remote Sens. Spatial Inform. Sci. 5 (2).

Maset, E., Rupnik, E., Pierrot-Deseilligny, M., Remondino, F., Fusiello, A., 2021. Exploiting multi-camera constraints within bundle block adjustment: AN experimental comparison. Int. Arch. Photogramm. Remote Sens. Spatial Inform. Sci. 43, 33–38.

Mur-Artal, R., Montiel, J.M.M., Tardos, J.D., 2015. ORB-SLAM: a versatile and accurate monocular SLAM system. IEEE Trans. Rob. 31 (5), 1147–1163.

Nikolic, J., Rehder, J., Burri, M., Gohl, P., Leutenegger, S., Furgale, P. T., & Siegwart, R. (2014). A synchronized visual-inertial sensor system with FPGA pre-processing for accurate real-time SLAM. *2014 IEEE international conference on robotics and automation (ICRA)*.

Ohio Statewide Imagery Program (OSIP). Retrieved 10/26/2022 from https://das.ohio.gov/technology-and-strategy/ogrip/projects/osip.

Papakonstantinou, A., Doukari, M., Roussou, O., Drolias, G. C., Chaidas, K., Moustakas, A., Athanasis, N., Topouzelis, K., & Soulakellis, N. (2018). UAS multi-camera rig for post-earthquake damage 3D geovisualization of Vrisa village. *Sixth International Conference on Remote Sensing and Geoinformation of the Environment (RSCy2018)*.

Rupnik, E., Daakir, M., & Pierrot Deseilligny, M. (2017). MicMac–a free, open-source solution for photogrammetry. *Open Geospatial Data, Software and Standards*, *2*(1), 1-9.

Sakamoto, T., Ogawa, D., Hiura, S., Iwasaki, N., 2022. Alternative procedure to improve the positioning accuracy of orthomosaic images acquired with agisoft metashape and DJI P4 multispectral for crop growth observation. Photogramm. Eng. Remote Sens. 88 (5), 323–332.

Schonberger, J. L., & Frahm, J.-M. (2016). Structure-from-motion revisited. *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Triggs, B. (1999). Camera pose and calibration from 4 or 5 known 3d points. *Proceedings of the Seventh IEEE International Conference on Computer Vision*.

Umeyama, S., 1991. Least-squares estimation of transformation parameters between two point patterns. IEEE Trans. Pattern Anal. Mach. Intell. 13 (04), 376–380.

Wierzbicki, D., 2018. Multi-camera imaging system for UAV photogrammetry. Sensors 18 (8), 2433.

Xie, Y., Shao, R., Guli, P., Li, B., & Wang, L. (2018). Infrastructure based calibration of a multi-camera and multi-lidar system using apriltags. *2018 IEEE Intelligent Vehicles Symposium (IV)*.

Xu, N., Huang, D., Song, S., Ling, X., Strasbaugh, C., Yilmaz, A., Sezen, H., Qin, R., 2021. A volumetric change detection framework using UAV oblique photogrammetry–A case study of ultra-high-resolution monitoring of progressive building collapse. Int. J. Digital Earth 14 (11), 1705–1720.

Xu, J., Li, R., Zhao, L., Yu, W., Liu, Z., Zhang, B., Li, Y., 2022. CamMap: Extrinsic calibration of non-overlapping cameras based on SLAM map alignment. IEEE Rob. Autom. Lett. 7 (4), 11879–11885.

Xu, Z., Lu, X., Wang, W., Xu, E., Qin, R., Niu, Y., Qiao, X., Yang, F., Yan, R., 2022. Monocular video frame optimization through feature-based parallax analysis for 3D pipe reconstruction. Photogramm. Eng. Remote Sens. 88 (7), 469–478.

Xu, N., Qin, R., Huang, D., Remondino, F., 2023. Enabling Neural Radiance Fields (NeRF) for large-scale aerial images–A multi-tiling approach and the geometry assessment of NeRF. Photogram. Rec.

Xu, N., Qin, R., Song, S., 2023. Point cloud registration for LiDAR and photogrammetric data: A critical synthesis and performance analysis on classic and deep learning algorithms. ISPRS Open J. Photogramm. Remote Sens. 8, 100032.

Yang, Y., Tang, D., Wang, D., Song, W., Wang, J., Fu, M., 2020. Multi-camera visual SLAM for off-road navigation. Rob. Auton. Syst. 128, 103505.

Zhu, K., Chen, W., Zhang, W., Song, R., & Li, Y. (2020). Autonomous robot navigation based on multi-camera perception. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.