Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.083412

ARTICLE

Real-Time Video Target Tracking via Geometric Coordinate Mapping

Na Li[1], **Yashu Zhang**[1], **Fengpu Lin**[1], **Liutao Zhao**[2,*], **Zhongshan Zhu**[3], **Chen Tom**[4] **and Tengfei Tu**[5]

[1]Beijing Big Data Center, Beijing Economic and Information Technology Bureau, Beijing, China
[2]College of Intelligence and Computing, Tianjin University, Tianjin, China
[3]Beijing Computing Center Company Ltd., Beijing Academy of Science and Technology, Beijing, China
[4]Manning College of Information & Computer Sciences, University of Massachusetts Amherst, Amherst, MA, USA
[5]School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing, China

*Corresponding Author: Liutao Zhao. Email: zhaolt@tju.edu.cn

Received: 03 April 2026; Accepted: 10 June 2026; Published: 23 July 2026

ABSTRACT: Accurate mapping of video imagery to physical space coordinates represents a fundamental challenge in dynamic target tracking and intelligent video analysis systems. Traditional methods struggle to maintain stable coordinate mapping in real-time video streams due to imaging distortion variations and changing environmental conditions. This paper presents a real-time coordinate mapping approach that integrates geometric constraints with online distortion correction to achieve stable pixel-to-target coordinate transformation for video target tracking applications. The proposed method introduces a planar geometric consistency constraint and an online distortion parameter update mechanism within a unified optimization framework, enabling adaptive adjustment of the mapping relationship under dynamic imaging conditions. Experimental results show that in static scenes, the root mean square error and mean absolute error of this method reach 1.16 and 0.77 mm, respectively, which are lower than those of deep learning regression methods. Under distortion conditions, the root mean square error remains stable at 0.22 ± 0.02 px, and the distortion parameter error is approximately 2×10^{-3}. In dynamic translation and rotational perturbation scenarios, the method still maintains low error and an accuracy of 84.5%. Furthermore, even with a feature point scale increased to 1000, the method still achieves real-time operation at 36 FPS with a count accuracy of 96.9%, validating its ability to support continuous tracking and counting of dynamic targets. This research contributes an engineering-feasible solution for real-time geometric coordinate mapping in video-based target tracking and intelligent monitoring applications.

KEYWORDS: Video target tracking; geometric coordinate mapping; geometric constraints; distortion correction; online optimization; real-time vision

1 Introduction

With the rapid advancement of digital twin technologies, smart city infrastructure, and intelligent video analysis systems, the ability to accurately track and count dynamic targets in video streams has become increasingly critical. In applications such as traffic monitoring, public safety surveillance, and industrial inspection, systems must continuously track moving objects including vehicles, pedestrians, and industrial components. The reliability of these tasks fundamentally depends on stable spatial coordinate mapping from video frames to a unified coordinate system. When targets move through different regions of the camera view, their pixel positions must be consistently transformed to physical space coordinates for meaningful analysis and decision-making.

The scope of this study is the coordinate-mapping component of a video tracking pipeline. Object detection, temporal association, and trajectory management can be connected as upstream or downstream modules, whereas the proposed method focuses on converting distorted pixel observations into stable target-frame coordinates in real time. This setting provides the spatial basis required for trajectory analysis and counting, and it also distinguishes the present work from general-purpose visual tracking or SLAM systems.

However, real-world deployment environments present significant challenges for maintaining accurate coordinate mapping. Camera lens distortion introduces non-linear geometric deformations that vary across the image field. Changes in camera installation position or orientation can alter the mapping relationship over time. Environmental factors such as temperature variations, vibrations, and mechanical settling further contribute to gradual degradation of calibration accuracy. Traditional approaches relying on offline calibration or static mapping models struggle to maintain accuracy under these dynamic conditions, leading to performance degradation in long-term video target tracking operations.

From the perspective of geometric coordinate mapping theory, transforming image pixels into target-frame coordinates depends on calibrated camera geometry, projective/multi-view geometry, temporally stable target localization, and explicit lens-distortion modeling [1–4]. Existing methods often separate camera calibration, target association, and distortion correction into independent steps, which makes it difficult to adapt the mapping relationship continuously when the camera view, lens distortion, or scene conditions change. Although high-precision calibration and rectification models are effective under controlled conditions, they are less suitable when real-time tracking and counting require stable, interpretable, and low-latency pixel-to-target coordinate conversion throughout a video stream.

Classical camera-calibration and distortion models demonstrate that radial, tangential, and decentering distortion can be explicitly parameterized and estimated from image observations [5,6]. These studies support the design philosophy of the proposed method: instead of using distortion correction only as an offline preprocessing step, the mapping and distortion parameters should be updated jointly so that the coordinate relationship remains stable under changing imaging conditions.

In the field of geometric reconstruction, incorporating geometric constraints into the reconstruction pipeline has been demonstrated as an effective approach to improve accuracy and robustness. Geometric constraints, such as planar consistency, collinearity, and coplanarity, provide additional regularization that constrains the solution space of the reconstruction problem. Meanwhile, distortion correction addresses the nonlinear geometric deformations introduced by the imaging system, which is essential for achieving accurate coordinate mapping from video imagery. To address the aforementioned challenges in video target tracking, this study proposes a real-time geometric coordinate mapping method that integrates geometric constraints and distortion correction. Planar geometric consistency constraints are introduced into a unified optimization framework, and mapping and distortion parameters are jointly updated online, thereby achieving stable spatiotemporal mapping of pixel observations under dynamic video stream conditions. This approach aligns with the principles of geometric reconstruction by treating coordinate mapping as a process of recovering the underlying geometric structure from distorted observations through constraint-driven optimization. The research aims to improve the accuracy and robustness of video target tracking while maintaining computational efficiency, providing reliable spatiotemporal support for the continuous tracking and counting of dynamic targets. Specifically, unlike existing SLAM-based approaches that primarily focus on camera pose estimation and sparse reconstruction, and unlike conventional distortion-aware mapping methods that treat distortion correction as an independent offline preprocessing step, the proposed method uniquely integrates planar geometric consistency constraints and online distortion parameter updates within a single unified real-time optimization framework. This joint modeling approach enables simultaneous adjustment of both geometric and distortion parameters at each frame, which is fundamentally different from prior sequential or decoupled processing pipelines. Furthermore, compared to existing joint optimization approaches, the proposed method introduces a closed-loop feedback mechanism between distortion parameters and mapping parameters, allowing continuous self-calibration during operation without requiring periodic offline recalibration.

2 Related Works

Video target tracking and coordinate mapping have been extensively studied in the computer vision community, with applications ranging from intelligent surveillance to autonomous navigation and industrial inspection [2,7]. The fundamental challenge lies in maintaining accurate and stable coordinate mapping from two-dimensional video observations to physical-space coordinates while handling target motion, geometric distortion, and environmental variations. In automatic detection and tracking of dynamic targets, spatial consistency across continuous frames is essential for stable trajectory analysis and counting. To improve reliability, existing studies have explored hierarchical scene-coordinate modeling, temporal map updating, multi-source fusion, dynamic-scene SLAM, and nonlinear distortion modeling.

In the domain of visual localization and geometric reconstruction, hierarchical scene-coordinate methods have proven effective for handling complex mapping relationships. Coarse-to-fine scene-coordinate classification and regression can reduce the difficulty of directly predicting 3D coordinates from pixels in large or ambiguous scenes [8], while differentiable robust estimation further connects learned coordinate prediction with geometric pose estimation [9]. For long-term operation, temporal map prediction and recalibration strategies help maintain localization consistency when the environment changes over time [10]. These studies indicate that hierarchical modeling and temporal updating are useful for improving coordinate stability, but they do not directly solve the problem of real-time pixel-to-target mapping with online distortion correction in fixed-camera tracking scenes.

To address illumination changes, weak texture, occlusion, and dynamic interference in complex scenes, visual SLAM and visual-odometry methods improve localization robustness by strengthening feature selection, map maintenance, and dynamic-object handling. Agricultural stereo SLAM illustrates the need for robust perception in outdoor environments [11], ORB-SLAM3 shows that real-time visual and visual-inertial SLAM can operate with pinhole and fisheye camera models [12], and DN-SLAM improves mapping consistency in dynamic scenes by suppressing moving-object interference [13]. These works are relevant to spatial perception robustness, but their main outputs are camera pose or map representations rather than a direct, lightweight coordinate-mapping module for target counting.

Multi-source fusion has emerged as an important direction for robust geometric estimation. Radar-visual odometry enhances pose estimation in complex dynamic environments through multi-modal and multi-scale adaptive fusion [14], and visual-inertial estimation improves metric localization by combining camera observations with inertial measurements in a nonlinear optimization framework [15]. These studies confirm that additional constraints or complementary sensing cues can stabilize spatial estimation; however, the present study focuses on a monocular video setting and therefore introduces geometric consistency and distortion parameters as internal constraints within the coordinate-mapping model.

From the perspective of visual modeling mechanisms, classical computer vision literature emphasizes that image formation, projection geometry, spatial constraints, and camera calibration are the basis of reliable coordinate recovery [16]. The treatment of nonlinear geometric distortions is particularly important: line-straightness self-calibration can estimate distortion from structured scenes [17], learning-based frameworks can correct image distortion under complex conditions [18], and joint distortion estimation and removal networks further show the benefit of modeling mixed distortion effects [19]. Collectively, these studies show that geometric structure and nonlinear distortion should be considered together when accurate coordinate mapping is required.

In summary, existing research has explored tracking, localization, geometric reconstruction, sensor fusion, and distortion correction from different perspectives. However, most methods either emphasize general target localization, estimate camera pose and maps, or handle distortion as a separate preprocessing problem. They rarely unify planar geometric consistency constraints, distortion parameter estimation, and online parameter updating in a single real-time coordinate-mapping framework. This gap is significant for fixed-camera video target tracking and counting applications, where the system requires an interpretable and continuously updated pixel-to-target mapping relationship rather than only object identities or camera trajectories. Therefore, this study proposes a real-time geometric coordinate mapping method that integrates geometric constraints and distortion correction, constructing a joint modeling framework for geometric constraints and distortion parameters and achieving dynamic adjustment of the coordinate mapping relationship through online parameter updates.

Recent learning-based and SLAM-based studies further show the trend toward combining data-driven perception with geometric constraints in tracking and localization. Deep digital-twin systems have been used for pedestrian tracking and evacuation-load assessment in public spaces [20], and TVG-SLAM introduces tri-view geometric constraints into Gaussian-splatting SLAM [21]. These works are useful references for end-to-end tracking or localization. The present study addresses a narrower engineering problem: for fixed-camera scenes, it builds an interpretable pixel-to-target coordinate mapping module in which geometric consistency and online distortion correction are optimized together.

3 Methods and Materials

3.1 Construction of Coordinate Mapping Model Incorporating Geometric Constraints

To address the problem of stable mapping of pixel observations to the target coordinate system in video target tracking applications, this study proposes a real-time coordinate mapping method that integrates geometric constraints and distortion correction to improve the stability and consistency of the mapping process under complex operating conditions. In the coordinate mapping modeling stage, if there is a lack of effective geometric constraints, the mapping relationship is often too free, and it is difficult to guarantee the structural consistency between different spatial quantities [22]. Therefore, this study first starts with the description of coordinate system relationships and geometric consistency constraint modeling, and constructs a coordinate mapping model that integrates geometric constraints. In order to facilitate the description of coordinate mapping relationships and research objects, this study describes the various coordinate systems involved in the system and their mapping targets, and their overall relationship is shown in Fig. 1.

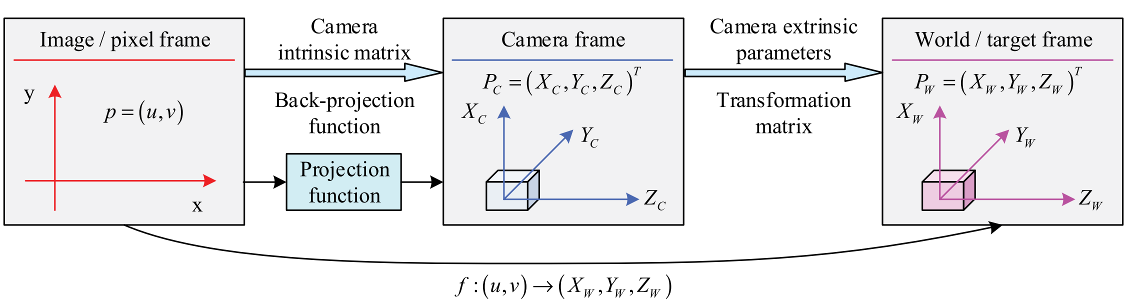

Figure 1: Schematic diagram of multi-coordinate system definition and coordinate mapping target.

As shown in Fig. 1, the coordinate mapping process involved in the study consists of the image coordinate system, the camera coordinate system, and the target coordinate system. The image information acquired by the visual sensor is first represented in the image/pixel coordinate system, where the pixel points are described by two-dimensional coordinate $p = (u, v)$ [23]. Under the influence of the camera intrinsic parameter matrix K and the imaging projection relationship, the pixel coordinates are mapped to the camera coordinate system through the back projection process, and the corresponding spatial point $P_C = (X_C, Y_C, Z_C)^T$ in the camera coordinate system is obtained. Subsequently, the spatial points in the camera coordinate system are transformed to the target coordinate system to obtain the target coordinate $P_W = (X_W, Y_W, Z_W)^T$. Based on the above coordinate system relationship, the goal of the study is to construct a direct mapping function $f: (u, v) \to (X_W, Y_W, Z_W)$ from the image pixel coordinates to the target coordinate system, so as to lay a unified coordinate description foundation for the subsequent introduction of geometric constraints and real-time mapping models.

In the process of coordinate mapping, the mapping from pixel coordinates to the target coordinate system involves multi-parameter coupling, and it is difficult to maintain structural consistency under dynamic conditions by simply relying on the imaging model and coordinate transformation [24]. Therefore, this study introduces geometric constraints in the mapping modeling stage, and uses spatial geometric relationships as constraints to participate in the modeling, so as to limit the degree of freedom of mapping and enhance the stability of mapping. Its mechanism is shown in Fig. 2.

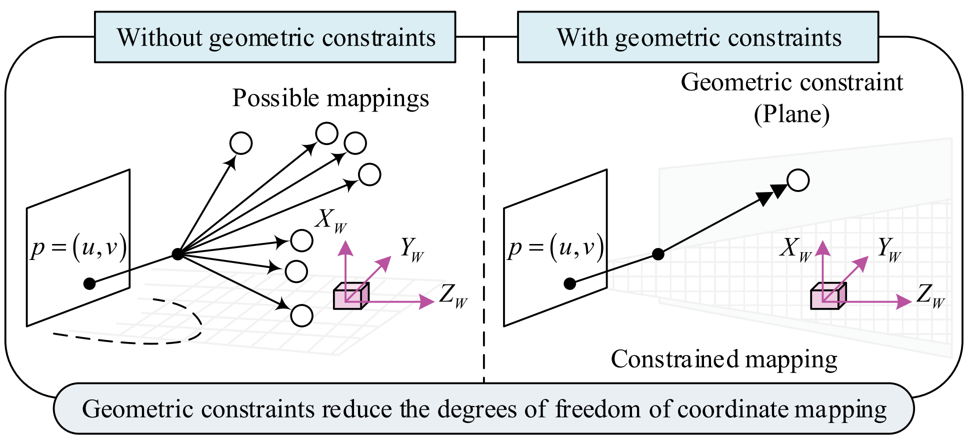

Figure 2: Schematic diagram of the mechanism of geometric constraints in coordinate mapping.

As shown in Fig. 2, without the introduction of geometric constraints, the same pixel point $p = (u, v)$ can correspond to multiple spatial positions in the target coordinate system under the action of back projection and coordinate transformation. The mapping result can be represented as a set of possible solutions P, as shown in Eq. (1).

$$P = \left\{ P_W \middle| P_W = T_{WC} \pi^{-1} (u, v) \right\} \tag{1}$$

In Eq. (1), $\pi^{-1}(\cdot)$ represents the back projection function, used to map two-dimensional pixel coordinates to spatial rays in the camera coordinate system. The matrix T_{WC} is the coordinate transformation matrix from the camera coordinate system to the target coordinate system. At this time, the mapping relationship has a high degree of freedom, making it difficult to guarantee the consistency of the spatial structure. When geometric constraints are introduced in the coordinate mapping modeling process, the mapping result must simultaneously satisfy the imaging geometric relationship and the spatial geometric constraint conditions. The planar constraint can be expressed as shown in Eq. (2).

$$n^T P_W + d = 0 \tag{2}$$

In Eq. (2), n is the normal vector of the constraint plane, and d is the plane offset term. By introducing the above geometric constraints into the mapping model, the originally unrestricted mapping solution space is compressed into a subspace that satisfies the geometric relationship, thereby effectively reducing the degrees of freedom of coordinate mapping and making the mapping results from pixel coordinates to the target coordinate system more consistent in structure. The study integrates the coordinate system relationship description, imaging model, and geometric constraint conditions to construct a coordinate mapping model that integrates geometric constraints. The overall process is shown in Fig. 3.

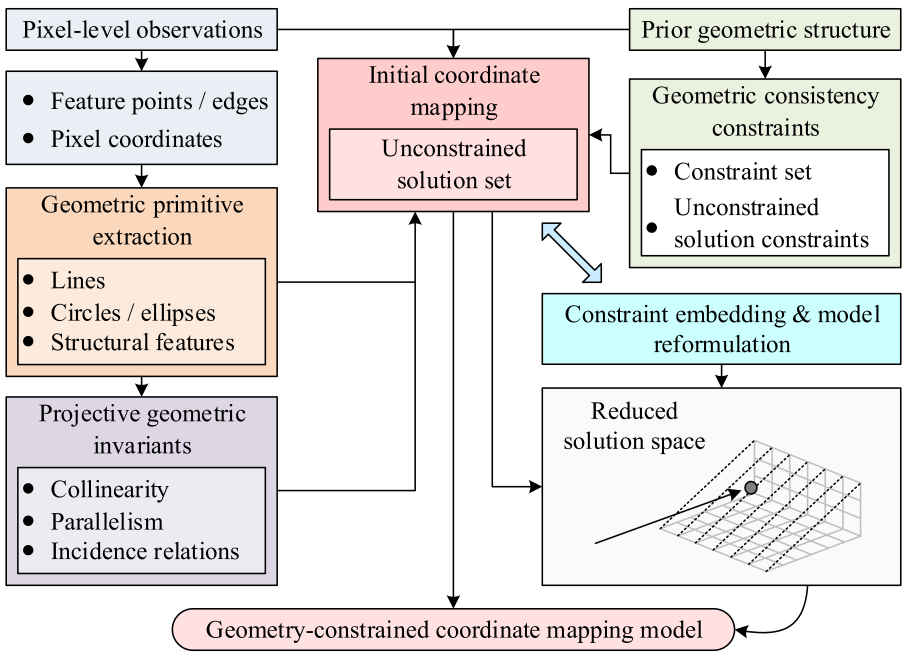

Figure 3: Flowchart of the coordinate mapping model construction integrating geometric constraints.

As shown in Fig. 3, the model takes pixel-level observations as input, first organizes discrete observations into geometrically meaningful element representations, and extracts projective geometric invariants from them to construct a set of geometrically consistent constraints. Then, in the model construction stage, the constraint embedding and mapping relationship reconstruction are completed, and finally a coordinate mapping model with geometric constraints is obtained. Specifically, pixel observations are denoted by Eq. (3).

$$p_i = (u_i, v_i), \quad i = 1, \ldots, N \tag{3}$$

In Eq. (3), N represents the total number of pixel observations involved in the modeling. Under the influence of imaging geometry and coordinate transformation, an initial mapping model from pixel to target coordinate system is established, as shown in Eq. (4).

$$P_{w,i} = f_0 (p_i; \theta) \tag{4}$$

In Eq. (4), $P_{w,i}$ represents the three-dimensional spatial point in the target coordinate system obtained by mapping from the pixel point p_i. $f_0(\cdot)$ is the initial coordinate mapping model without introducing geometric constraints, used to describe the basic mapping relationship from pixel coordinates to target coordinates. θ represents the parameter vector of the mapping model, used to characterize the geometric or pose parameters involved in the mapping relationship. Based on geometric feature extraction and projective invariant modeling, the geometric consistency constraint is expressed as a set of constraint residual functions, as shown in Eq. (5).

$$C \triangleq \left\{c_j(\theta) = 0\right\}_{j=1}^{M} \tag{5}$$

In Eq. (5), C represents the set of geometric consistency constraints, consisting of M constraint terms, where $c_j(\theta)$ is the jth geometric constraint residual function, used to characterize the spatial geometric relationship that the mapping result must satisfy. Taking the most common planar consistency as an example, it can be written as shown in Eq. (6).

$$c_i^{plane}(\theta) = n^T P_{W,i}(\theta) + d = 0 \tag{6}$$

In Eq. (6), when the mapping point $P_{w,i}$ satisfies the condition $n^T P_W + d = 0$, it means that the point is located on the defined geometric constraint plane. In order to achieve constraint embedding and model reconstruction, the initial mapping error and geometric constraint residual are unified into the same objective function, resulting in a mapping modeling form that integrates geometric constraints, as shown in Eq. (7).

$$\min_{\theta} \sum_{i=1}^{N} \left\| P_{w,i} - f_0(p_i; \theta) \right\|^2 + \lambda \sum_{j=1}^{M} \left\| c_j(\theta) \right\|^2 \tag{7}$$

In Eq. (7), the first term represents the mapping error term of the initial mapping model, which is used to measure the consistency between the mapping result and the model prediction; the second term is the geometric constraint penalty term, which is used to measure the degree to which the mapping parameters satisfy the geometric consistency constraints, where λ is the constraint weight coefficient, which is used to balance the relative influence of the mapping error and the geometric constraints in the overall modeling process.

3.2 Distortion Correction-Driven Real-Time Mapping Joint Optimization Method

Based on the coordinate mapping model that integrates geometric constraints, the mapping relationship is constrained in terms of structural consistency. However, the inherent nonlinear distortion of the imaging system in dynamic application scenarios and its variation with operating conditions still significantly affect the mapping accuracy and stability [25,26]. To address this, this study further introduces a distortion correction mechanism, jointly models and optimizes the distortion parameters and mapping parameters online, and constructs a distortion correction-driven real-time coordinate mapping method, the structure of which is shown in Fig. 4.

As shown in Fig. 4, this study treats imaging distortion as a unified internal factor in the coordinate mapping modeling process, rather than as an independent preprocessing step. The model takes distorted pixel observations as input, characterizes the impact of imaging distortion on pixel observations through a distortion modeling module, and incorporates the distortion effect and geometric constraint information into the joint coordinate mapping model. During model operation, the distortion parameters establish a feedback relationship with the coordinate mapping model through an online update mechanism, enabling the mapping parameters to adaptively adjust with changes in imaging conditions, thereby achieving collaborative modeling and real-time updating of distortion correction and coordinate mapping. Based on the mapping model structure that integrates distortion and geometric constraints, the study further designs the online solution and update process for the model parameters, and its overall solution flow is shown in Fig. 5.

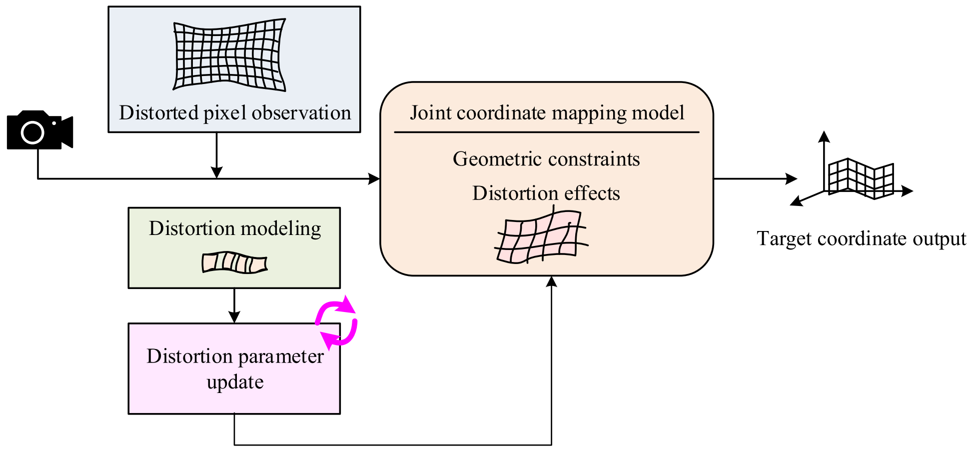

Figure 4: Schematic diagram of the structure of the distortion constraint embedded coordinate mapping model.

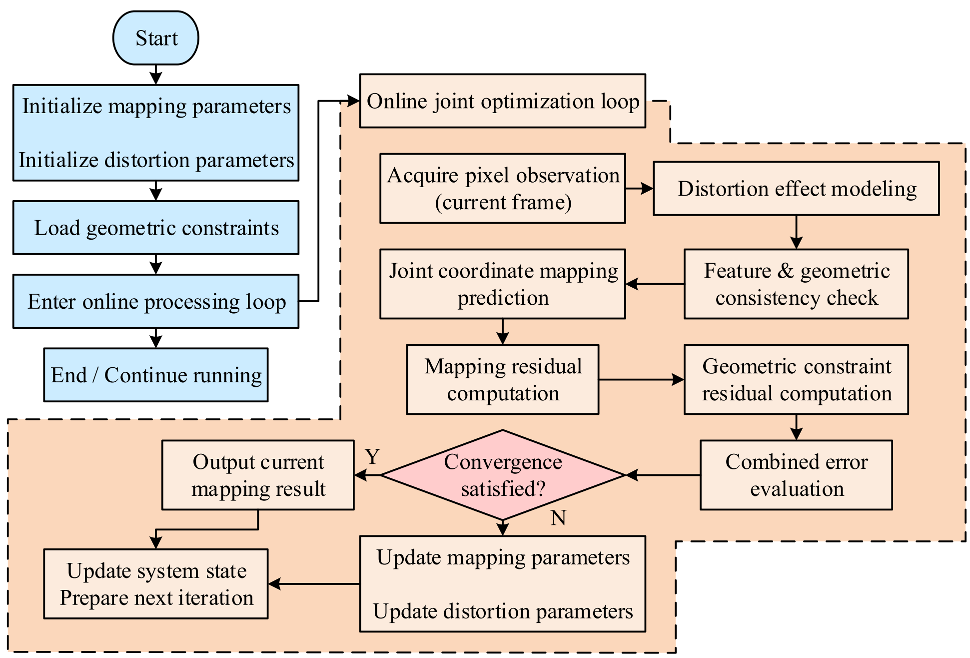

Figure 5: Flowchart for online solution of coordinate mapping parameters.

As shown in Fig. 5, the online solution process for the coordinate mapping parameters is based on a joint optimization framework, which synchronously updates the mapping parameters and distortion parameters in a continuous time series. The system first initializes the mapping model parameters and distortion parameters, and continuously receives pixel observation data of the current frame during operation. For the input of the tth frame, the joint coordinate mapping model $P_w^{(t)}$ can be expressed as shown in Eq. (8).

$$P_w^{(t)} = f\left(p^{(t)}, \theta^{(t)}, \kappa^{(t)}\right) \tag{8}$$

In Eq. (8), $p^{(t)}$ represents the input set of pixel observations, $\theta^{(t)}$ represents the coordinate mapping parameter vector, and $\kappa^{(t)}$ represents the distortion parameter vector at the corresponding time. Based on this prediction result, the study calculates the mapping consistency error and geometric constraint residual respectively to evaluate the mapping quality under the current parameter state. The mapping residual term is used to measure the deviation between the predicted mapping result and the model consistency, and can be uniformly expressed as shown in Eq. (9).

$$E_{map}^{(t)} = \sum_i \left\| P_{w,i}^{(t)} - \hat{P}_{w,i}^{(t)} \right\|^2 \tag{9}$$

In Eq. (9), $E_{map}^{(t)}$ represents the mapping consistency error term, $P_{w,i}^{(t)}$ represents the prediction result of the ith mapping point in the target coordinate system, and $\hat{P}_{w,i}^{(t)}$ represents the reference prediction value calculated by the current mapping model. Meanwhile, the geometric constraint residual is used to measure the degree to which the mapping result satisfies the spatial geometric consistency condition, and its form is shown in Eq. (10).

$$E_{geo}^{(t)} = \sum_j \left\| c_j\left(\theta^{(t)}, \kappa^{(t)}\right) \right\|^2 \tag{10}$$

In Eq. (10), $E_{geo}^{(t)}$ represents the comprehensive measure of the residuals of all geometric constraints. To achieve a unified evaluation of mapping error and geometric constraints, a comprehensive cost function is constructed during the online optimization process, as shown in Eq. (11).

$$E^{(t)} = E_{map}^{(t)} + \zeta E_{geo}^{(t)} \tag{11}$$

In Eq. (11), ζ is the weight coefficient, used to balance the relative influence of mapping consistency and geometric constraints in the optimization process. In each frame iteration, the study determines whether the convergence or threshold condition is met based on the change of the comprehensive cost function $E^{(t)}$; if not, the mapping parameters and distortion parameters are jointly updated simultaneously, and the online update process can be abstractly represented as shown in Eq. (12).

$$\left(\theta^{(t+1)}, \kappa^{(t+1)}\right) = M\left(\theta^{(t)}, \kappa^{(t)}, \nabla E^{(t)}\right) \tag{12}$$

In Eq. (12), $M(\cdot)$ represents the parameter update operator, and $\nabla E^{(t)}$ represents the gradient information of the comprehensive cost function relative to the model parameters at time t, which reflects the direction of parameter adjustment. Through the above online loop mechanism, the coordinate mapping parameters and distortion parameters can be adaptively adjusted according to the imaging conditions and observation changes, thereby achieving real-time and stable coordinate mapping result output. Finally, the proposed real-time coordinate mapping method that integrates geometric constraints and distortion correction is shown in Fig. 6.

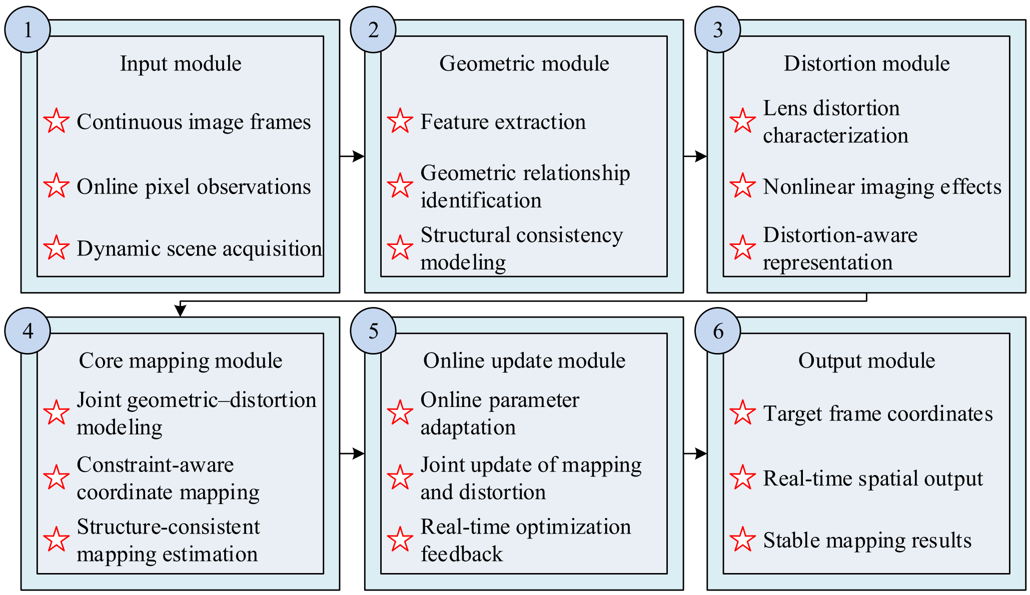

Figure 6: Real-time coordinate mapping method fused with geometric constraints and distortion correction.

As shown in Fig. 6, the proposed method takes a continuous visual image stream as input, processes online pixel observations during operation, and acquires spatial structure and imaging characteristic information from two levels: geometric perception and structural modeling, and imaging distortion modeling. Subsequently, the geometric constraints and distortion modeling results are uniformly introduced into the coordinate mapping core module. Through online joint optimization and parameter update mechanisms, the mapping model is continuously adjusted, ensuring the coordinate mapping process remains stable despite scene changes. Finally, the method outputs the spatial mapping result in the target coordinate system, forming a complete workflow from visual input, geometric and distortion modeling, joint mapping to online updates, demonstrating the method's overall synergy and real-time performance in complex dynamic scenes.

Algorithm 1 summarizes the complete procedure of the proposed real-time geometric coordinate mapping method. The algorithm takes a continuous video stream as input and outputs target coordinates for each pixel observation at every frame through an online joint optimization loop.

Algorithm 1: Real-time geometric coordinsate mapping with joint optimization

---

Input: Video stream $V = \{I_1, I_2, ..., I_T\}$; initial camera intrinsics K_0; initial extrinsics T_0; initial distortion parameters $\delta_0 = [k_1, k_2, p_1, p_2] = [0, 0, 0, 0]$

Output: Target coordinates P_t for each pixel observation at frame t

1: Initialize mapping parameters θ_0 using DLT; set $\delta_0 = 0$, $\lambda = 0.5$

2: For each frame $t = 1, 2, ..., T$ do

3: Detect feature points and extract pixel observations O_t using Shi-Tomasi detector

4: Compute back-projected rays from pixel observations using current distortion model

5: Calculate mapping consistency error Emap(θ_t, δ_t) via Eq. (9)

6: Calculate geometric constraint residual $\Phi(\theta_t)$ via Eq. (10)

(Continued)

Algorithm 1 (continued)

7: Evaluate comprehensive cost $C(\theta_t, \delta_t)$ = Emap + $\lambda \cdot \Phi$ via Eq. (11)
8: Jointly update θ_t and δt using Levenberg-Marquardt via Eq. (12)
9: If $|C_t - C_t{-}1|/C_t < 10^{-6}$ or iterations $\geq$ 5, output mapping result
10: Return target coordinates $\{P_t\}$ for all frames

3.3 Implementation Details

Feature points are selected with a spatially uniform strategy. Shi-Tomasi responses are computed on each frame, and the image is divided into regular grids to avoid concentrating points in a single textured region. The default number of points is N = 200 in the accuracy test, and N = 200, 500, 800, and 1000 in the scalability test. Candidates in each grid are sorted after non-maximum suppression. If occlusion or weak texture reduces the number of reliable detections, short-term tracks from the previous frame are retained and checked by the geometric constraint. When fewer than four reliable points remain, the online update is suspended and the last stable mapping parameters are used for output.

The reference plane is initialized from calibration correspondences. Given 3D reference points on the target plane, the normal vector n is obtained from the eigenvector corresponding to the smallest eigenvalue of the point covariance matrix, and the offset is calculated as $d = -n^T\ \overline{p}$, where $\overline{p}$ is the centroid of the reference points. The initial mapping parameter θ_0 is estimated by direct linear transformation and then refined with the first valid observation batch. The weight λ is set to 0.5 according to the ablation results in Section 4.2, where the weak- and strong-constraint variants show only limited accuracy changes around this value.

The initial distortion vector is set to $\delta_0 = [k_1, k_2, p_1, p_2] = [0, 0, 0, 0]$ for the moderate-field industrial lens used in the experiments. This setting is not intended as a universal assumption. For wide-angle or fisheye lenses, an offline calibration result can be used as the initial value, and additional radial, tangential, or fisheye parameters can be added without changing the main optimization structure.

The Levenberg-Marquardt update minimizes the residual vector formed by the mapping residual and the weighted geometric residual. The Jacobian of the plane constraint is derived from the plane residual, and the distortion-related Jacobian is computed with respect to the active distortion parameters. The damping factor is increased if a trial update raises the total cost and decreased after an accepted update. The relative convergence threshold is 10^{-6}, and each frame is limited to five iterations. These settings were chosen to keep the per-frame computation within the real-time budget; more iterations gave only minor error reduction while increasing latency.

4 Results

4.1 Experimental Environment and Parameter Settings

To verify the effectiveness and superiority of the proposed real-time coordinate mapping method that integrates geometric constraints and distortion correction under video streaming conditions, experimental verification was conducted. The specific experimental environment configuration and parameter settings are shown in Table 1.

Based on Table 1, the study selected Zhang's Camera Calibration Dataset and the TUM RGB-D public dataset as experimental data sources. These datasets were input into the system frame-by-frame as continuous video streams to simulate the real-time push interface of camera video data. Before data use, all video sequences underwent a standardized preprocessing procedure, including image resolution standardization, camera parameter format conversion, and outlier frame removal, to ensure consistency and fairness of video stream input under different experimental conditions.

Table 1: Experimental environment and parameter settings.

Category	Parameter	Value	Category	Parameter	Value
Camera parameters	Camera model	Basler acA1920-40gc	Software environment	Operating system	Windows 10 (64-bit)
	Lens model	Computar M1214-MP2		Programming language	Python 3.9
	Image resolution	1920 × 1080		Image processing library	OpenCV 4.5.5
Computing platform	Frame rate	30 fps	Online optimization parameters	Maximum iterations	5 iterations per frame
	CPU	Intel Core i7-10700		Update frequency	Per-frame update
	Memory	32 GB	Experimental settings	Number of feature points	50–300 per frame
	GPU	NVIDIA RTX 3060 (12 GB)		Number of repeated runs	5

The comparison methods are implemented under the same input and hardware settings. OCCH uses Zhang-style offline camera calibration followed by homography-based planar mapping with OpenCV 4.5.5. NLS-VCM uses the same correspondences and refines the mapping parameters by nonlinear least-squares optimization. DL-CR is a data-driven coordinate-regression baseline trained on the same calibration correspondences, using normalized pixel coordinates as input and target-frame coordinates as output. The data split, image resolution, hardware platform, and feature-point settings are kept the same for all methods.

4.2 Verification of the Effectiveness and Robustness of the Mapping Model

The study first conducted static accuracy benchmark experiments. Three representative and directly comparable baselines were selected: Offline Camera Calibration and Homography-based Mapping (OCCH), Nonlinear Least Squares-based Visual Coordinate Mapping (NLS-VCM), and Deep Learning-based Coordinate Regression (DL-CR). OCCH represents the widely used geometric calibration and homography pipeline for fixed-camera mapping; NLS-VCM represents a strong optimization-based coordinate-mapping baseline; and DL-CR represents a learning-based regression baseline trained under the same data partitions. These baselines cover the main technical routes relevant to this task, namely offline geometric calibration, nonlinear optimization-based correction, and data-driven coordinate regression. They are therefore more directly comparable to the proposed pixel-to-target mapping module than SLAM, monocular depth, or general pose-estimation systems, which solve different output problems. The results are shown in Fig. 7.

As shown in Fig. 7a, the root mean square error (RMSE) of the OCCH, NLS-VCM, and DL-CR methods were 1.73, 1.45, and 1.31 mm, respectively, indicating that nonlinear optimization and data-driven models could improve the accuracy of static mapping. In contrast, the average RMSE of the research method was further reduced to 1.16 mm, a decrease of 11.2% compared to DL-CR, and the standard deviation was only 0.09 mm, demonstrating higher stability. As shown in Fig. 7b, the mean absolute error (MAE) of the comparative methods was 1.32, 1.02, and 0.93 mm, respectively, while the research method was 0.77 mm, a decrease of 17.3% compared to DL-CR, and with the smallest error fluctuation. The results showed that introducing geometric constraints and jointly modeling imaging distortion in static scenes could effectively reduce mapping errors and improve stability.

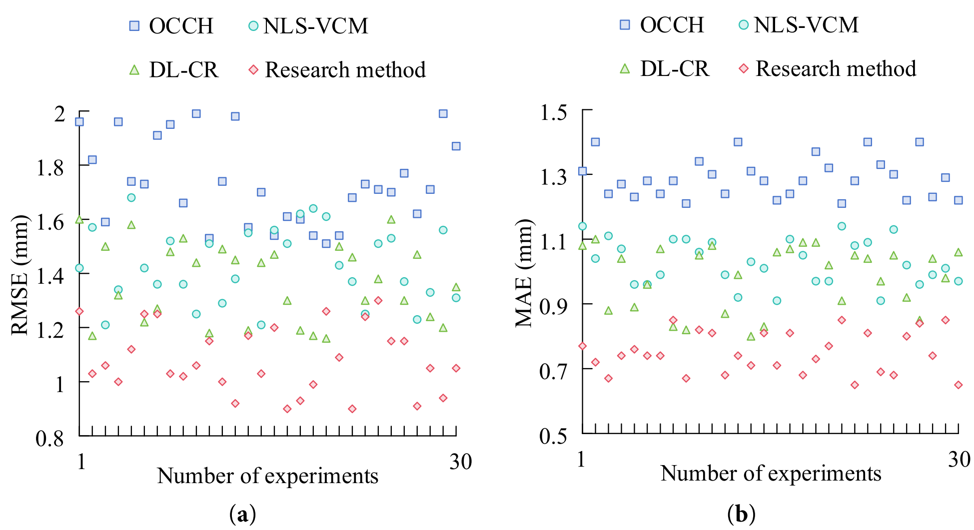

Figure 7: Static accuracy reference experiment: (**a**) comparison of RMSE; (**b**) comparison of MAE.

To systematically evaluate the contribution of each key component to the overall performance, a comprehensive ablation study was conducted. First, to analyze the impact of geometric constraints on coordinate mapping performance, a comparative analysis was performed using a model without geometric constraints, a planar consistency constraint model, a weak constraint model, and a strong constraint model. The weak constraint model uses geometric constraints with lower weights for coordinate mapping, while the strong constraint model strengthens the coordinate mapping restrictions by increasing the weights of the geometric constraints. The four models are denoted as G1–G4, and the comparison results are shown in Fig. 8.

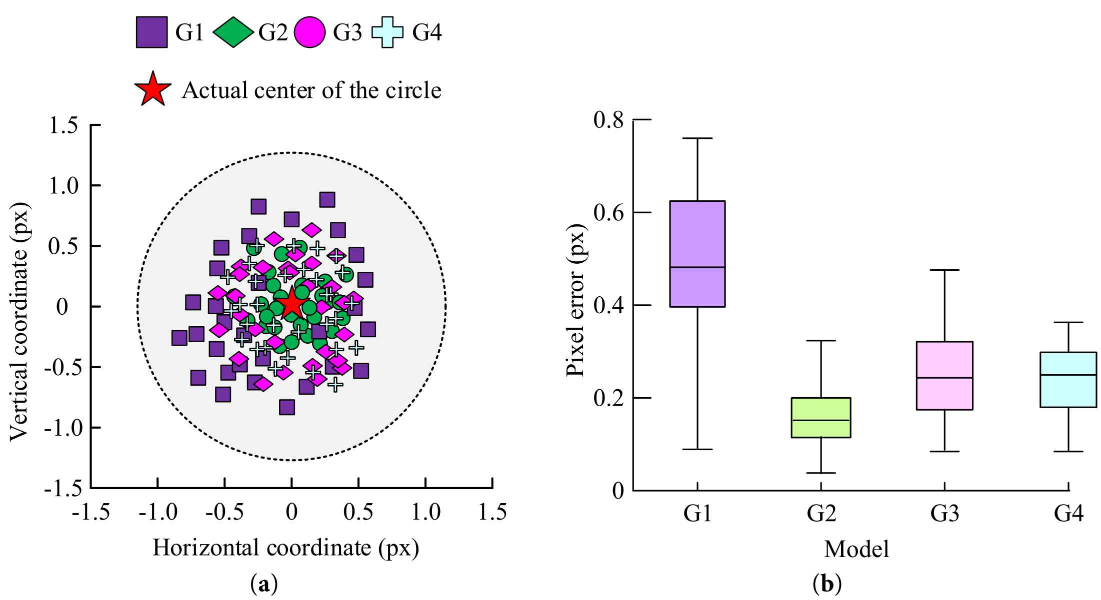

Figure 8: Effects of geometric constraints on coordinate mapping performance: (**a**) the spatial distribution of projection points; (**b**) error statistics results.

Based on this, to analyze the impact of geometric constraints on coordinate mapping performance, a comparative study was conducted using a model without geometric constraints, a planar consistency constraint model, a weak constraint model, and a strong constraint model. The weak constraint model uses geometric constraints with lower weights for coordinate mapping, while the strong constraint model strengthens the coordinate mapping restrictions by increasing the weights of the geometric constraints. The four models are denoted as G1–G4, and the comparison results are shown in Fig. 8.

As shown in Fig. 8a, the projection points of the model without geometric constraints were the most dispersed, while the projection points converged significantly towards the true position after introducing appropriate geometric constraints; the convergence effect was limited when the constraints were too weak or too strong. As shown in Fig. 8b, the median pixel error decreased from 0.47 px to about 0.16 px and then stabilized; the median errors under weak or strong constraints were about 0.23 and 0.24 px, respectively. The results showed that there was a reasonable range of geometric constraints, and appropriate constraints could significantly improve the accuracy and stability of coordinate mapping. Furthermore, to verify the stability and adaptability of the proposed method under the condition of varying imaging distortion, a robustness experiment of distortion variation was conducted. Multiple groups of methods were set up, namely, a fixed distortion parameter group, a group that only updates the mapping parameters, and a joint update group. A learning-based distortion compensation method was introduced for comparison, and the methods were named D1–D4, respectively. The comparison results are shown in Fig. 9. These ablation results on constraint strength confirm that the planar consistency constraint with appropriate weighting is a critical component, as both its absence and improper weighting lead to significant performance degradation.

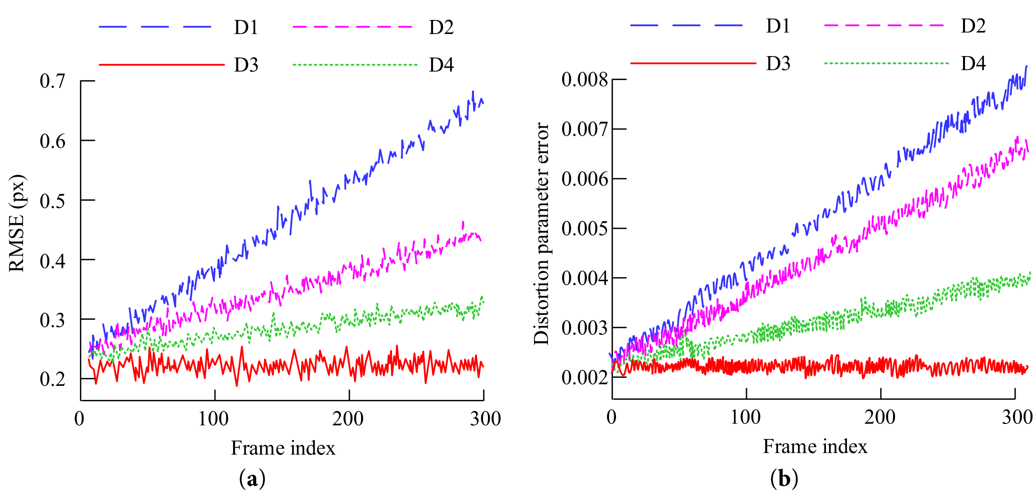

Figure 9: Results of the distortion variation robustness experiment: (**a**) RMSE variation over time; (**b**) distortion parameter tracking error over time.

As shown in Fig. 9a, the RMSE of D1 continuously increased with frame number, growing from an initial 0.25 px to approximately 0.68 px at the end of the sequence. D2 could mitigate the error growth to some extent, but the RMSE still rose to 0.42 px. D4 further reduced the error growth rate, with its RMSE remaining at approximately 0.32 px at the end. In contrast, the proposed D3 remained stable throughout the entire sequence, with the RMSE basically maintained within the range of 0.22 ± 0.02 px, showing no significant error drift over time and demonstrating stronger robustness against distortion changes. As shown in Fig. 9b, the distortion parameter error of D1 gradually increased over time, reaching approximately 8×10^{-3} at the end. Although D2 reduced the error growth slope, the error at the end was still close to 6×10^{-3}. D4 could further suppress error growth, with its distortion parameter error at approximately 4×10^{-3} at the end of the sequence. In contrast, the distortion parameter error of the proposed D3 remained stable at around 2×10^{-3}, with only small random fluctuations, indicating that the method could effectively track changes in distortion parameters.

4.3 Online Joint Optimization and Real-Time Availability Assessment

To verify the real-time adaptability and usability of the proposed online joint optimization strategy under dynamic working conditions, an online adaptation experiment under dynamic perturbation was further conducted when the camera or target underwent continuous motion, and the viewing angle and imaging conditions changed rapidly. The results are shown in Fig. 10.

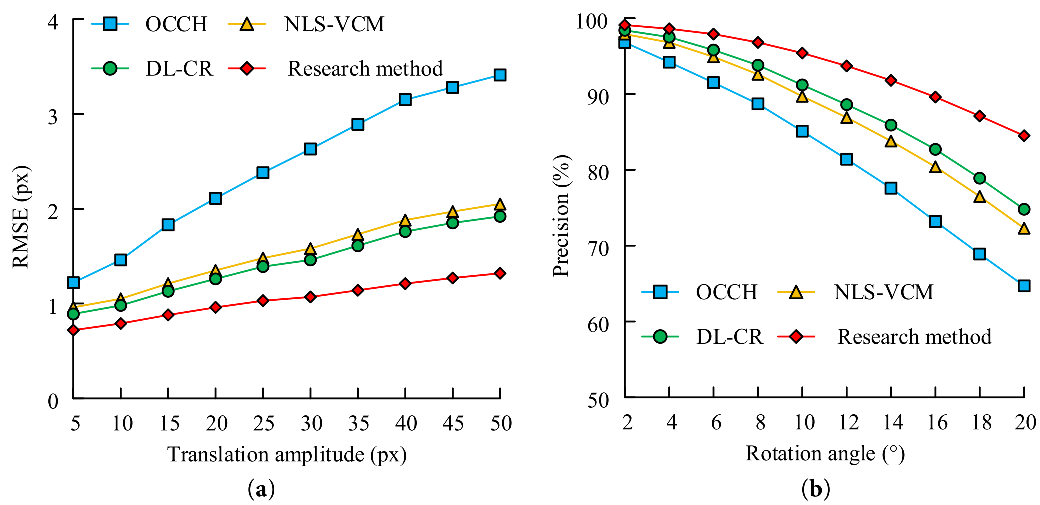

Figure 10: Online adaptation experiment under dynamic disturbance: (**a**) RMSE comparison under the variation of translation perturbation intensity; (**b**) comparison of precision rates under the variation of rotational disturbance intensity.

As shown in Fig. 10a, the RMSE of the OCCH method rapidly increased from 1.22 to 3.41 px, indicating its greatest sensitivity to translational perturbations. Under maximum translation conditions, the RMSEs of NLS-VCM and DL-CR increased to 2.05 and 1.92 px, respectively. In contrast, the research method maintained the lowest error across the entire perturbation range, with its RMSE increasing only from 0.72 to 1.32 px, demonstrating that online joint optimization could effectively adapt to translational perturbations. As shown in Fig. 10b, the precision of each method gradually decreased as the rotation angle increased from 2° to 20°. The precision of the OCCH method decreased most significantly, from 96.8% to 64.7%. Under 20° rotation, the precision of NLS-VCM and DL-CR decreased to 72.3% and 74.8%, respectively. In contrast, the research method maintained a precision of 84.5% under large-angle rotation conditions, demonstrating stronger online adaptability. Therefore, the online joint optimization method exhibited superior adaptability and stability under dynamic perturbation conditions such as translation and rotation. Finally, to address the requirements of continuous video stream processing and scalability for real-time tracking and counting of dynamic targets in video applications, real-time performance and scalability experiments were conducted. The focus was on evaluating the algorithm's real-time support capability and operational stability for continuous tracking and counting of dynamic targets under conditions of continuously increasing feature point size. The results are shown in Table 2.

Table 2: Experimental results of real-time performance and scale expansion.

Number of Feature Points N (Scale)	Metric	OCCH	NLS-VCM	DL-CR	Research Method
200	Per-frame time (ms)	3.6	19.2	8.9	11.4
	Frame rate (FPS)	278.0	52.0	112.0	88.0
	Counting accuracy (%)	97.9	96.8	98.4	99.0
500	Per-frame time (ms)	4.9	46.3	10.2	16.9
	Frame rate (FPS)	204.0	22.0	98.0	59.0
	Counting accuracy (%)	95.2	89.6	96.3	98.1
800	Per-frame time (ms)	6.1	78.5	11.8	23.6
	Frame rate (FPS)	164.0	13.0	85.0	42.0
	Counting accuracy (%)	93.4	83.7	95.1	97.5
1000	Per-frame time (ms)	6.8	95.4	12.9	27.9
	Frame rate (FPS)	147.0	10.0	78.0	36.0
	Counting accuracy (%)	91.8	78.9	94.2	96.9

As shown in Table 2, the single-frame computation time of each method increased with the feature point size from 200 to 1000, but their support capabilities for dynamic target tracking and data lookup tasks differed significantly. The OCCH method maintained a high frame rate (≥147 FPS) at all scales, but its data counting accuracy decreased from 97.9% to 91.8%, indicating that mapping stability decreased with scale when lacking online adaptability. The NLS-VCM method experienced the fastest increase in computational overhead, with the frame rate decreasing from 52 to 10 FPS, corresponding to a data counting accuracy decrease from 96.8% to 78.9%, making it difficult to meet continuous statistical requirements. The DL-CR method maintained 78 FPS at 1000 feature points, but its data counting accuracy decreased from 98.4% to 94.2%, demonstrating some stability. In contrast, the method presented in this paper maintained a real-time processing capability of 36 FPS even at the largest scale, and the counting accuracy only decreased from 99.0% to 96.9%, achieving the highest statistical accuracy under all scale conditions, indicating that it achieved a better balance between real-time performance and counting stability.

5 Discussion

To address the problem of traditional visual coordinate mapping methods struggling to maintain stability under complex operating conditions in real-time video target tracking and counting applications, this study proposed a real-time geometric coordinate mapping method that integrates geometric constraints and distortion correction. The proposed method addresses the fundamental challenge of maintaining accurate geometric mappings from distorted video imagery by integrating geometric constraints with online distortion correction. This method was based on online visual observation, introduced planar geometric consistency constraints, and jointly updated mapping and distortion parameters, constructing an online optimizable real-time closed-loop mapping process.

In static experiments, the proposed method achieved RMSE and MAE of 1.16 and 0.77 mm, respectively, representing reductions of 11.2% and 17.3% compared to DL-CR. Planar consistency constraints reduced the median pixel error from 0.47 to 0.16 px. Under varying distortion conditions, the proposed method maintained an RMSE of 0.22 ± 0.02 px, with distortion parameter errors remaining at 2×10^{-3}. In dynamic perturbation experiments, when the translation increased from 5 to 50 px, the RMSE only increased from

0.72 to 1.32 px, and the precision remained at 84.5% even with a 20° rotation. Furthermore, when the feature point size increased to 1000, the system maintained a real-time processing capability of 36 FPS, with a corresponding counting accuracy of 96.9%. The ablation study on constraint strength (Fig. 8, G1–G4) provides indirect evidence regarding weight sensitivity: the median pixel error under weak constraints (0.23 px) and strong constraints (0.24 px) both remained within an acceptable range relative to the optimal setting (0.16 px), indicating that the method is not highly sensitive to moderate weight variations. The fixed weight design (λ = 0.5) was determined through empirical validation and avoids the risk of oscillatory behavior that adaptive schemes may introduce during abrupt scene changes. Regarding feature extraction, the Shi-Tomasi detector was selected for its computational efficiency, which is critical for maintaining real-time performance at 36 FPS. For scenarios with weak textures, learned keypoint detectors such as SuperPoint could serve as an alternative, though this would require careful evaluation of the resulting computational overhead.

Therefore, this method could simultaneously achieve real-time performance and statistical accuracy under continuous video streaming conditions, possessing engineering applicability to support long-term tracking and counting of dynamic targets in video target tracking applications. The findings demonstrate that the integration of geometric constraints and distortion correction within a unified optimization framework provides an effective approach for video target tracking in dynamic video environments. However, current research mainly relied on planar geometric constraints, limiting its adaptability to complex 3D structural scenes. Future research will extend validation to real-world surveillance data collected from multiple camera configurations and under varying illumination conditions, and will incorporate richer spatial geometric constraints and multi-source sensor information to further improve the method's generalization ability and application applicability in complex scenarios. The proposed method targets a different problem from visual SLAM (e.g., ORB-SLAM3) and PnP-based localization: it addresses real-time pixel-to-ground mapping for fixed-camera surveillance, whereas SLAM targets 6-DOF pose estimation and PnP requires known 3D coordinates. Potential failure cases include severe occlusion (<4 detectable feature points), extreme wide-angle lenses, and highly repetitive textures. The LM optimization step represents the computational bottleneck (27.9 ms/frame), which could be further accelerated through GPU parallelization. Regarding the distortion model, the camera used in the experiments (Computar M1214-MP2) is a standard industrial lens with a moderate field of view, for which tangential distortion is inherently minor compared to radial distortion—a well-documented characteristic in the camera calibration literature. Although the initialization includes both radial and tangential parameters ($\delta_0 = [k_1, k_2, p_1, p_2] = [0, 0, 0, 0]$), the online optimization naturally maintains the tangential parameters at values close to zero, confirming their negligible contribution for this lens type. For wide-angle or fish-eye lenses where tangential distortion becomes significant, the same framework can readily accommodate additional distortion parameters without structural modification.

From an engineering perspective, the main computational cost comes from the per-frame joint optimization step, especially the Jacobian computation in the Levenberg-Marquardt update. The five-iteration limit keeps the method real-time in the tested range, but more complex distortion models or learned feature extractors may increase latency. In stable frames, the update frequency can be reduced, and the Jacobian computation can be parallelized on GPU to improve throughput.

6 Conclusion

This study presented a real-time geometric coordinate mapping method for video target tracking. By introducing planar geometric consistency constraints and online distortion-parameter updates into one optimization framework, the method converts distorted pixel observations into stable target-frame coordinates. Experiments on public calibration and RGB-D sequences show improved mapping accuracy, robust behavior under distortion variation, and real-time performance under dynamic perturbation and scale expansion. Overall, the method provides an interpretable coordinate-mapping module for tracking and counting pipelines. Future work will extend quantitative validation to privacy-compliant surveillance and industrial videos with physical-coordinate annotations, and will further study piecewise-planar or full 3D extensions for more complex scenes.

Acknowledgement: None.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: In this study, Na Li contributed to conceptualization, methodology development, software implementation for real-time coordinate mapping, and validation of geometric constraint integration in video streams; Yashu Zhang undertook the collection and preprocessing of real-world surveillance video datasets and annotated dynamic target trajectories; Fengpu Lin performed formal analysis of spatial distortion patterns, optimized online calibration parameters, and implemented distortion correction modules; Liutao Zhao, as the corresponding author, managed the project, designed the overall methodology, integrated the framework of geometric constraints and distortion correction, and led the writing, review, and editing of the original manuscript; Zhongshan Zhu provided resource support through access to the surveillance infrastructure of the Beijing Computing Center and assisted in field deployment testing under real environmental conditions; Chen Tom developed the theoretical model of planar consistency constraints, derived the mathematical formulation of the joint optimization model, and validated it in simulated environments at the University of Massachusetts Amherst; Tengfei Tu conducted experimental evaluations of tracking accuracy under conditions of vibration and temperature drift, performed comparative analyses against baseline methods, and visualized spatiotemporal mapping results. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the Corresponding Author, Liutao Zhao, upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zhang Z. A flexible new technique for camera calibration. IEEE Trans Pattern Anal Mach Intell. 2000;22(11):1330–4. doi:10.1109/34.888718.
2. He KJ, Zhang CL, Xie S, Li ZX, Wang ZW, Qin RG. Target-aware tracking with spatial-temporal context attention. IEEE Trans Circuits Syst Video Technol. 2024;34(8):7176–89. doi:10.1109/tcsvt.2024.3370981.
3. Hartley R, Zisserman A. Multiple view geometry in computer vision. 2nd ed. Cambridge, UK: Cambridge University Press; 2004. doi:10.1017/cbo9780511811685.
4. Peng K, Wei J, Hao X. CSPR-Net: self-supervised curved surface projection rectification network for geometric distortion correction in non-planar projections. arXiv:2602.08433. 2026.
5. Brown DC. Decentering distortion of lenses. Photogramm Eng. 1966;32:444–62.
6. Tsai R. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. IEEE J Robot Automat. 1987;3(4):323–44. doi:10.1109/jra.1987.1087109.
7. Guo H, Du X, Wang W. Motion deep association for spatio-temporal object tracking. Pattern Recognit. 2025;168(11):111787. doi:10.1016/j.patcog.2025.111787.
8. Li X, Wang S, Zhao Y, Verbeek J, Kannala J. Hierarchical scene coordinate classification and regression for visual localization. In: Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun 13–19; Seattle, WA, USA. p. 11980–9. doi:10.1109/cvpr42600.2020.01200.

9. Brachmann E, Krull A, Nowozin S, Shotton J, Michel F, Gumhold S, et al. DSAC—differentiable RANSAC for camera localization. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017 Jul 21–26; Honolulu, HI, USA. p. 2492–500. doi:10.1109/cvpr.2017.267.
10. Deng T, Xie H, Wang J, Chen W. Long-term visual simultaneous localization and mapping: using a Bayesian persistence filter-based global map prediction. IEEE Robot Automat Mag. 2023;30(1):36–49. doi:10.1109/mra.2022.3228492.
11. Islam R, Habibullah H, Hossain T. AGRI-SLAM: a real-time stereo visual SLAM for agricultural environment. Auton Rob. 2023;47(6):649–68. doi:10.1007/s10514-023-10110-y.
12. Campos C, Elvira R, Rodriguez JJG, Montiel JMM, Tardos JD. ORB-SLAM3: an accurate open-source library for visual, visual-inertial, and multimap SLAM. IEEE Trans Robot. 2021;37(6):1874–90. doi:10.1109/tro.2021.3075644.
13. Ruan C, Zang Q, Zhang K, Huang K. DN-SLAM: a visual SLAM with ORB features and NeRF mapping in dynamic environments. IEEE Sens J. 2024;24(4):5279–87. doi:10.1109/jsen.2023.3345877.
14. Zhuo G, Lu S, Zhou H, Zheng L, Zhou M, Xiong L. 4DRVO-net: deep 4D radar-visual odometry using multi-modal and multi-scale adaptive fusion. IEEE Trans Intell Veh. 2024;9(6):5065–79. doi:10.1109/tiv.2023.3330956.
15. Qin T, Li P, Shen S. VINS-mono: a robust and versatile monocular visual-inertial state estimator. IEEE Trans Robot. 2018;34(4):1004–20. doi:10.1109/tro.2018.2853729.
16. Szeliski R. Computer vision: algorithms and applications. Cham, Switzerland: Springer International Publishing; 2022. doi:10.1007/978-3-030-34372-9.
17. Fitzgibbon AW. Simultaneous linear estimation of multiple view geometry and lens distortion. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2001; 2001 Dec 8–14; Kauai, HI, USA.
18. Li S, Chu Y, Zhao Y, Zhao P. An efficient deep learning-based framework for image distortion correction. Visual Comput. 2024;40(10):6955–67. doi:10.1007/s00371-024-03580-3.
19. Wang Y, Li H, Hou S. Joint distortion estimation and removal network for versatile hybrid-distorted image restoration. IEEE Trans Instrum Meas. 2025;74:1–14. doi:10.1109/tim.2025.3541646.
20. Sun H, Ding Y, Fan R, Zhang Y, Zhang T, Huang X, et al. Deep learning-driven digital twin system for pedestrian tracking and evacuation load assessment in public spaces. Eng Appl Artif Intell. 2026;174:114440. doi:10.1016/j.engappai.2026.114440.
21. Tan Z, Chen X, Feng L, Ge Y, Zhi S, Liu J, et al. TVG-SLAM: robust Gaussian splatting SLAM with tri-view geometric constraints. IEEE Robot Autom Lett. 2026;11(2):1314–21. doi:10.1109/lra.2025.3641103.
22. Triggs B, McLauchlan PF, Hartley RI, Fitzgibbon AW. Bundle adjustment—a modern synthesis. In: Vision algorithms: theory and practice. Berlin/Heidelberg, Germany: Springer; 2000. p. 298–372. doi:10.1007/3-540-44480-7_21.
23. Ponce J. Computer vision: a modern approach. Upper Saddle River, NJ, USA: Prentice Hall; 2002.
24. Malis E, Vargas M. Deeper understanding of the homography decomposition for vision-based control [dissertation]. Paris, France: INRIA; 2007.
25. Kannala J, Brandt SS. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. IEEE Trans Pattern Anal Mach Intell. 2006;28(8):1335–40. doi:10.1109/tpami.2006.153.
26. Heikkila J, Silvén O. A four-step camera calibration procedure with implicit image correction. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition; 1997 Jun 17–19; San Juan, Puerto Rico. p. 1106–12.