Good Weights: Proactive, Adaptive Dead Reckoning Fusion for Continuous and Robust Visual SLAM

Yanwei Du, Jing-Chen Peng, Patricio A. Vela

***Abstract*— Given that Visual SLAM relies on appearance cues for localization and scene understanding, texture-less or visually degraded environments (*e.g.*, plain walls or low lighting) lead to poor pose estimation and track loss. However, robots are typically equipped with sensors that provide some form of dead reckoning odometry with reasonable short-time performance but unreliable long-time performance. The *Good Weights* (GW) algorithm described here provides a framework to adaptively integrate dead reckoning (DR) with passive visual SLAM for continuous and accurate frame-level pose estimation. Importantly, it describes how all modules in a comprehensive SLAM system must be modified to incorporate DR into its design. Adaptive weighting increases DR influence when visual tracking is unreliable and reduces when visual feature information is strong, maintaining pose track without overreliance on DR. *Good Weights* yields a practical solution for mobile navigation that improves visual SLAM performance and robustness. Experiments on collected datasets and in real-world deployment demonstrate the benefits of *Good Weights*.**

Keywords: Visual SLAM, dead reckoning, feature tracking, optimization

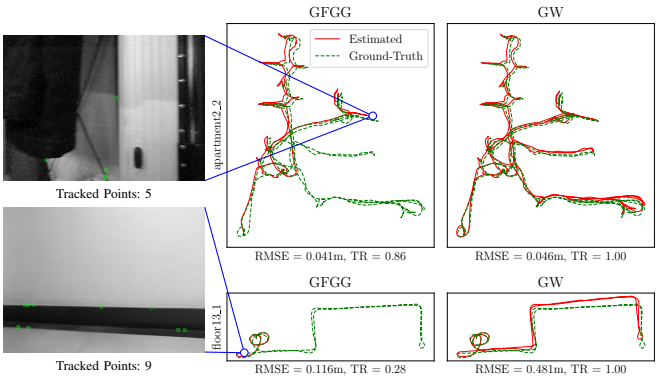

Fig. 1: Performance comparison between a visual SLAM method, GFGG, and its *Good Weights* augmentation, GW, on the CID dataset for the apartment (top) and floor (bottom) sequences. The left-most images show instances of poor visual signals with an indication of their location in the trajectory (blue circle). The GW augmentation supports full tracking and SLAM success in these scenarios where the standard approach fails.

I. Introduction

Visual Simultaneous Localization and Mapping (SLAM) is often formulated as a nonlinear least-squares problem, where camera poses and 3D landmarks are jointly estimated from visual observations [1]–[3]. Optimization accuracy and stability depends on the sufficiency and reliability of feature associations across frames, short-term and long-term. Incorrect motion priors, low-texture scenes, and other visually ambiguous situations result in poor or missing data association for which the least-squares problem diverges or fails [4], [5].

Dead reckoning, typically from IMU and wheel encoders [6], supplements visual odometry with continuous pose estimates [7]–[11]. These measurements are integrated either through pre-integration or direct pose factors in tightly- or loosely-coupled formulations. Running at higher frequency than cameras, they capture robot kinematics and supply short-term motion priors that help maintain localization when visual features are weak or unavailable. However, such systems fuse measurements with fixed weights, making them sensitive to special motion patterns and often dependent on online calibration to compensate for imperfect modeling [12], [13]. More importantly, these methods remain odometry-focused rather than integrated into a full visual SLAM pipeline, where map construction can reinforce visual constraints rather than relying predominantly on inertial cues.

Y. Du, J. Peng and P. A. Vela are with the School of Electrical and Computer Engineering, and Institute of Robotics and Intelligent Machines, Georgia Institute of Technology, Atlanta, GA 30308, USA. {yanwei.du, jpeng303, pvela}@gatech.edu

Recent studies [14]–[18] explore dynamic tuning of constraint weights to prevent one sensor from dominating the optimization. They highlight the benefit of treating sensor contributions as variable rather than fixed influences. An extreme form of adaptive weighting leads to a switching SLAM [19], [20], where the system selects one modality as the main source and switches to others during sensor degradation. Most switching designs focus on LiDAR-based SLAM, where degenerate environmental structure (e.g., corridors, tunnels) is detected via scan-matching residuals or Jacobian conditioning [21]–[24]. LiDAR-based pipelines, with direct geometric measurements, can apply post-optimization corrections such as reweighting or smoothing. In contrast, indirect visual SLAM maintains a multi-stage hierarchy that relies on appearance-based cues, not structural cues. Its design inherently involves multiple dependent modules: tracking, local mapping, and loop closing, each of which requires the previous stage to remain well-conditioned. When visual signals are degraded, failures in early tracking or local mapping propagate through the pipeline and compromise global consistency.

The impact of poor visual signals motivates the use of dead-reckoning (DR) motion priors as a structural component in visual SLAM for robots. DR provides pre-conditioning across modules to handle degenerate scenarios that cannot be addressed by post-hoc reweighting. This paper introduces a proactive, adaptive DR-aided visual SLAM framework. When feature association becomes poor or fails, the system incorporates DR as a motion prior to maintain pose con-

tinuity. Unlike tightly coupled fusion methods, DR is only used when needed and the system smoothly returns to vision once tracking recovers. This design uses lightweight visual-health indicators to regulate DR contributions and preserve long-term accuracy. The framework, denoted *Good Weights*, provides the following contributions:

- A proactive, adaptive weighting scheme using tracking quality to modulate dead reckoning as a motion prior, leading to improved robustness under poor visual tracking.
- A comprehensive strategy integrating dead reckoning priors into tracking, local mapping, and loop closing, while maintaining real-time operation.
- Experiments on indoor datasets and on a mobile robot demonstrate improved continuity and stability in low-texture conditions without sacrificing accuracy under normal conditions.

II. Related Works

A. Dead Reckoning-aided Visual SLAM/Odometry

Inertial and wheel measurements show promising performance improvements when fused with visual observations [2], [8], [9], [25], typically within factor-graph or filtering frameworks. Fusion covariance parameters are derived from sensor specifications, feature detection and tracking statistics, or learned uncertainty models. These fusion techniques emphasize short-horizon odometry estimation and do not fully exploit the visual map constructed during robot navigation. Yet this map provides valuable visual constraints through landmark re-observations and loop closures, which improve accuracy and optimization conditioning [26], [27]. *Given the benefit of longer-term visual association, weighting strategies are needed for the stronger structural constraints provided by the visual map.*

B. Adaptive-Fusion SLAM

Adaptive fusion approaches for improved pose estimation in challenging scenarios broadly divide into two categories. The first follows the classical multi-sensor design [28], [29], in which all available modalities are fused and their weights are adjusted adaptively through heuristic rules or learning-based schemes [17]. The second category adopts switching-based fusion, where one primary modality drives estimation and backup modules are activated when degenerate conditions are detected [19], [20]. The strategy is effective in LiDAR SLAM, where degeneracy is identified during pose optimization by analyzing Jacobian conditioning [22]. Extending this idea to visual SLAM is non-trivial: due to its hierarchical design, conditioning analysis depends on reliable tracking and feature associations, which may already have failed prior to optimization in low-texture scenarios. Other LiDAR works exploit points measurements with motion profiles at earlier stages [23], [30]. These approaches rely on active depth sensing, which measures geometry directly. In contrast, (passive) visual SLAM must indirectly infer geometry from appearance cues, which are less consistently correlated with the underlying structure, making the problem inherently more difficult.

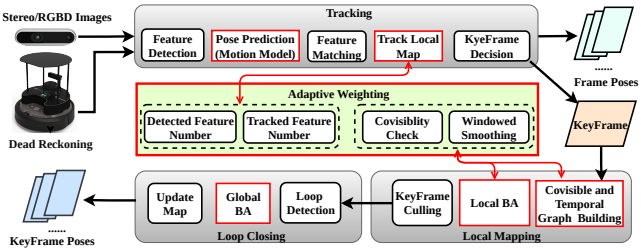

Fig. 2: System overview following the design of [1], [3], with newly added and modified modules highlighted in red rectangle blocks. The main contribution is the adaptive weighting scheme, where dead reckoning (DR) is selectively introduced as a motion prior within the system.

C. Optimization

Adaptive strategies exist in optimization [31]: Levenberg–Marquardt (LM) [32] and trust-region approaches [33] dynamically adjust a damping factor to trade off between gradient-descent-like steps and Gauss–Newton updates. Varying optimization weights based on the outcome of the previous optimization step leads to improved convergence. Similar ideas appear in robust estimation, where residuals are down-weighted online using kernels such as Huber or Cauchy [34]. *Good Weights* follows a similar design but anticipates unreliable measurements before optimization, scaling the contribution of dead reckoning according to visual tracking quality indicators. *It complements post-hoc adaptive schemes by addressing failure cases earlier in the pipeline, ensuring that the underlying Hessian remains well conditioned even when visual constraints are scarce.*

In summary, prior methods in optimization (LM, trust region) and sensor fusion (DR-aiding with adaptive weighting) share a reactive nature: they stabilize optimization only after evaluating progress. LiDAR systems can afford this because dense measurements ensure that residuals remain informative. Visual SLAM, however, cannot. *Good Weights* addresses this gap by differentially employing DR priors across the SLAM hierarchy to maintain well-conditioned subproblems throughout the pipeline. This proactive, hierarchical strategy makes *Good Weights* distinct from multi-sensor smoothing and classical damping techniques.

III. Methodology

A. System Overview

Our framework builds on a standard visual SLAM pipeline [1], [3], which is organized into three main threads: tracking, local mapping, and loop closing, as illustrated in Fig. 2. Frame poses are estimated in the tracking thread at the image-stream rate, providing real-time localization for online operation. Keyframes are initialized from selected frames, and their poses are further optimized in a local map or a global map depending on whether loop closure is triggered, reflecting post-processed mapping accuracy. Vision remains the primary source of pose estimation, ensuring accuracy and long-term consistency when features are reliably tracked. To handle situations where feature association becomes weak

or fails, the system incorporates dead reckoning (DR) as an auxiliary motion prior. The key contribution of this work is an adaptive weighting scheme that regulates how much influence DR has at each stage of the pipeline. When visual tracking is strong, DR plays a minor role; when tracking degrades, DR stabilizes the estimate; and when vision fails entirely, DR maintains continuity until recovery. The following subsections first introduce the adaptive weighting formulation, then describe its integration into the tracking, local mapping, and loop closing modules.

B. Adaptive Weighting Scheme

In what follows, all frame poses, including motion priors from DR, are expressed in the camera frame for consistency, assuming that sensor calibration and time synchronization have been properly handled.

1) Problem Formulation: The goal of visual SLAM is to estimate the sequence of frame poses $\mathbf{T}_{1:T} \in \mathrm{SE}(3)$ given image measurements and auxiliary motion priors. Without loss of generality, express the SLAM bundle adjustment form as:

$$\min_{\mathbf{T}_{1:T},\mathbf{X}} \underbrace{\sum_{(i,j)} \left\| r_{ij}^{\text{vis}}(\mathbf{T}_i, \mathbf{X}_j) \right\|^2_{\Sigma_v^{-1}}}_{\text{visual reprojection residuals}} + \underbrace{\sum_{k} \left\| r_k^{\text{DR}}(\mathbf{T}_k, \mathbf{T}_{k-1}) \right\|^2_{\Sigma_d^{-1}}}_{\text{dead-reckoning priors}}, \tag{1}$$

where r_{ij}^{vis} denotes the reprojection error of landmark j in frame i, and r_k^{DR} denotes the relative pose residual from DR between consecutive frames $k-1$ and k. The covariances Σ_v and Σ_d determine the weighting of each contribution.

In practice, visual information is not constant. When enough good features are tracked, visual residuals are reliable and dominate the optimization. When features are few or mismatched, the visual Jacobian J_v becomes weak, yielding an ill-posed optimization problem [34]. DR provides continuous motion estimates but suffers from drift over long horizons. If DR is always weighted the same, it either dominates and pulls the estimate off track, or becomes too weak to help when vision fails.

This motivates the use of an adaptive scheme. Rather than altering the visual covariances which would require complex per-feature uncertainty modeling [35] and extensive tuning as often seen in sensor fusion approaches [29], [36], we keep Σ_v fixed and instead regulate the contribution of DR through a global measure of visual tracking quality.

2) Adaptive Weighting Formulation: We introduce a score $Q_t \in [0,1]$ to represent the health of visual tracking at time t. A value of $Q_t = 1$ indicates strong and stable tracking, while $Q_t = 0$ means no visual constraints are available. The DR prior information is scaled as

$$\Sigma_d^{-1}(t) = \alpha(Q_t)\, \Sigma_{d0}^{-1}, \tag{2}$$

where Σ_{d0}^{-1} is the nominal DR information matrix. We interpolate the DR weight α between lower and upper bounds $\alpha_{\min}$ and $\alpha_{\max}$ as a function of the tracking quality Q_t. To handle the large dynamic range, interpolation is performed

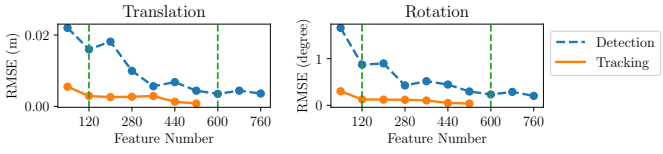

Fig. 3: Relationship Between Frame Pose RMSE and Tracking Statistics on CID Dataset. The detection variable is the number of features detected per frame (at most 800 in the settings). The tracking variable is the number of features matched between current frame and local map.

in log-space:

$$\alpha(Q_t) = \alpha_{\min} \left(\frac{\alpha_{\max}}{\alpha_{\min}} \right)^{1-Q_t} \tag{3}$$

From the linearized system of Eq. (1), the Hessian takes the form with two parts:

$$H(Q_t) = J_v^T \Sigma_v^{-1} J_v + \alpha(Q_t) J_d^T \Sigma_{d0}^{-1} J_d \tag{4}$$

The DR term $J_d^T \Sigma_{d0}^{-1} J_d$, introduces constraints on the full 6-DoF pose, producing dense block fill-ins in the Hessian. These constraints provide auxiliary support when visual measurements are sparse or unreliable, ensuring that the pose graph remains well connected and the optimization stable. The adaptive prior acts as a regularizing term to maintain optimization conditioning when visual tracking fails.

3) Scoring System for Q_t: Jacobian conditioning can be misleading in visual SLAM, where appearance-based measurements are only indirectly tied to geometry and poor tracking or incorrect data associations can distort measurement quality in low-texture scenarios. This contrasts with LiDAR SLAM, where depth measurements directly encode geometric structure.

Prior work has investigated using the number of tracked features as an indicator [19], providing an early warning of visual tracking failure and predicting degradation before pose estimation collapses. In our evaluation of a vision-only SLAM system [3] on the CID dataset [37] which contains challenging low-texture scenes, a clear negative correlation is observed: as the number of detected and tracked features decreases, the frame-level pose RMSE increases (Fig. 3). The effect is more pronounced for the number of detected features, where RMSE grows quadratically once the count drops below 600. In contrast, the feature tracking count shows lower sensitivity once it exceeds 120, where RMSE continues to decrease but at a diminishing rate. These results demonstrate that the detected feature count N_{det} and the tracked feature count N_{trk} serve as reliable indicators of visual tracking quality. To map these tracking variables to a quality score, define

$$Q_t = \omega_1 \frac{N_{det}}{N_{det}^r} + \omega_2 \frac{N_{trk}}{N_{trk}^r}, \tag{5}$$

where N_{det}^r and N_{trk}^r are preset target feature constants for reliable detection and tracking, and ω_1, ω_2 are nonnegative scalars such that $\omega_1 + \omega_2 = 1$. Each ratio is clipped to $[0,1]$, ensuring $Q_t \in [0,1]$. Since these two variables are computed during the initial tracking stage, recovering Q_t introduces minimal overhead.

4) DR Weighting Bounds α: The DR weighting limits $\alpha_{\min}$ and $\alpha_{\max}$ should be such that the DR contribution to optimization is appropriately regulated. DR should have little to no influence when tracking quality is high (Q large), and increasingly influential as tracking quality degrades (Q decreases), to dominate the optimization when tracking quality is low (Q small). Theoretically, finding the bounds can be tied to information-theoretic measures of the pose Hessian, such as *trace*, *log-determinant*, or *minimum eigenvalue* [3], [38], [39]. These metrics provide a principled way to regulate the balance between DR and visual constraints. However, such approaches require Hessian computations normally built during optimization, not available before, and their conditioning may not consistently reflect actual system performance.

Our objective is to achieve robust DR fusion across varying tracking conditions rather than precise weighting. We adopt a lightweight, data-driven strategy, using the *floor13_1* sequence of the CID dataset with nominal DR covariances Σ_{d0} set to 0.004 m/frame (translation) and 0.1°/frame (rotation). Safe α bounds are obtained through empirical sweeps, with each candidate evaluated by trajectory accuracy (frame-level RMSE) as shown in Fig. 4. Two segments with poor (case 0) and good (case 1) tracking were evaluated independently, with each test (a single α value) repeated five times. The results show that in both cases, DR begins to take effect at $\log(\alpha) = -1.0$ and gradually increases its influence, dominating the optimization at $\log(\alpha) = 3.0$. These values define the chosen operational bounds, set to capture the range where DR has a measurable effect on visual estimation rather than globally optimal weights. Avoiding fine-tuning prevents overfitting to specific sequences and preserves generality. Once established, the bounds are fixed across all experiments. Together with the adaptive rule $\alpha(Q_t)$ in Eq. (3), this ensures DR contributes only under visual tracking degradation, while vision remains dominant when reliable.

The next subsection describes how these adaptively weighted DR priors integrate into the different modules of the Visual SLAM pipeline.

C. Integration with Visual SLAM

1) Motion Model for Feature Tracking: As presented in Fig .2 (top block), DR first provides a Motion Model for feature matching. Visual pose estimation depends on establishing correct feature correspondences, which are typically obtained by projecting landmarks from the local map into the current frame using a predicted pose. This is fundamentally different from LiDAR-SLAM with geometric association. In vision-only SLAM systems [1]–[3], prediction often employs a constant-velocity (CV) motion model with correspondences limited to a 2D search window centered on predicted projections. The CV model only extrapolates from past states and does not capture recent motion, making it sluggish and unreliable during rapid maneuvers. In contrast, DR measures robot kinematic information to generate motion estimates that more closely follow the robot's short-term trajectory, often at a higher rate than visual estimation, thereby providing stronger motion priors for feature tracking.

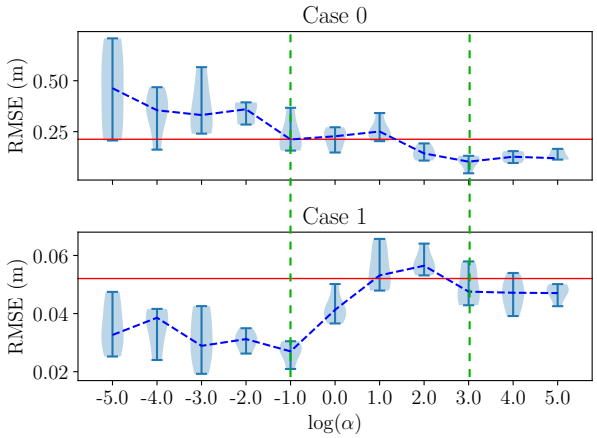

Fig. 4: DR weight sweeping study on sequence *floor13_1*. Two segments with poor (case 0) and good (case 1) tracking were evaluated independently, each repeated five times. Median RMSE values are connected by the blue dashed line. Dotted green lines show the values of α_{min} and α_{max} chosen for DR weighting bounds. The solid horizontal red line represents the RMSE achieved by pure dead reckoning (no visual feedback).

2) Map-to-Frame Pose Estimation: **Track-Local-Map** happens after feature tracking is finished. A motion-only bundle adjustment (BA) is used for pose estimation, which is a reduced form of the same optimization problem as in Eq. (1). Only the current camera pose T_t is optimized while landmark positions X_j are fixed:

$$\min_{\mathbf{T}_t} \sum_j \left\| r_{tj}^{\text{vis}}(\mathbf{T}_t, \mathbf{X}_j) \right\|^2_{\Sigma_v^{-1}} + \left\| r_t^{\text{DR}}(\mathbf{T}_t, \mathbf{T}_{t-1}) \right\|^2_{\Sigma_d^{-1}} \quad (6)$$

At this stage, robust estimation is typically used to suppress outliers, yet performance still depends on having sufficient inliers. In indoor environments, this assumption often fails: plain walls and narrow spaces can leave the estimator with few or no detected landmarks. The problem then becomes under-constrained, and pose estimation quickly diverges. Since feature detection and tracking statistics are available beforehand, they are used to trigger adaptive DR weighting per Eq. 2, keeping the Hessian well-conditioned and the optimization stable. DR serves as a short-term constraint and motion prior, maintaining state connectivity and ensuring continuous pose output when vision cannot sustain tracking.

3) Local Bundle Adjustment: A keyframe is created based on tracking statistics and motion profiles, then refined through local bundle adjustment (LBA) by linking to its covisible neighbors. While this process works under reliable visual tracking, texture-less environments often yield poor covisibility and weak or missing constraints. To address this, we incorporate DR as additional relative constraints to temporally adjacent keyframes, thereby reinforcing graph connectivity and keeping the LBA problem well conditioned.

The weight of DR constraints at this stage is scaled according to the number of covisibility connections. Let C_{ij} denote the connection count (number of visual edges) between keyframes i and j. We define C_{ref} as a refer-

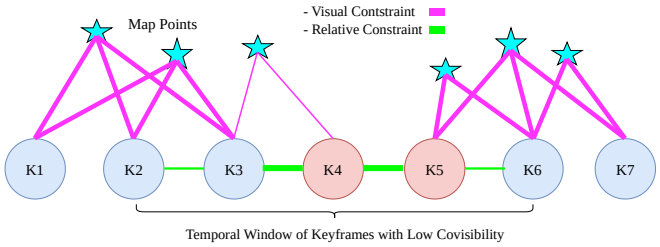

Fig. 5: Adaptive DR weighting in bundle adjustment. Stronger DR constraints (wider lines) are applied to low-covisibility keyframes ($K4$, $K5$) and their neighbors, then fade out as tracking recovers.

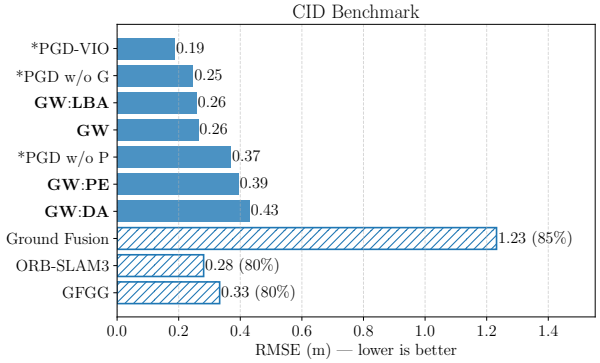

Fig. 6: Performance comparison of SLAM methods on the CID dataset. Methods are sorted from top to bottom by sequence completeness and then by RMSE. Results marked with * are reported from [42]. Our methods are shown in bold. Hashed lines indicate incomplete sequence results, with the value given in parentheses next to the RMSE.

ence connection value, estimated online during local bundle adjustment (LBA). Specifically, C_{ref} is computed as the median connection count from well-tracked keyframes in the most recent covisibility graph, representing the typical connection strength under reliable tracking conditions. The keyframe quality score for LBA is then defined as,

$$Q_{ij} = \frac{C_{ij}}{C_{ref}} \tag{7}$$

The ratio is saturated so that Q_{ij} remains bounded within $[0, 1]$. An initial weight value is estimated as per Eq. 3, such that poorly connected keyframes receive stronger DR influence. Applying a windowed smoothing strategy distributes DR weights to adjacent keyframes. The weighting decreases with covisibility distance, giving closer neighbors higher weights and ensuring a gradual, consistent influence of DR across the local window, as illustrated in Fig. 5.

4) Global Bundle Adjustment: Global bundle adjustment (GBA) is triggered upon loop closure and jointly refines all keyframe poses and landmarks to correct long-term drift and achieve a globally consistent map. The same connectivity issues observed in LBA also appear at the global level, where low-covisibility keyframes contribute little to the optimization. To address this, DR-stabilized weights from LBA are preserved, keeping the global optimization in Eq. 1 well posed. This stabilization is particularly important for robotic applications like inspection and surveying [40], [41], where repeat traversals are frequent. In such scenarios, a corrected and consistent map allows frame-level poses to align with keyframe-level accuracy, providing reliable localization for robust long-term navigation.

Employing DR throughout the hierarchical design preserves system conditioning for all optimizations, which delivers continuous and accurate pose estimation even in challenging scenarios. This is most effective during short-term visual tracking losses, where DR maintains connectivity and prevents failure. For extended visual outages, the accumulated error is limited by DR accuracy.

IV. Experiments

Validation of the proposed method involved integrating the adaptive weighting modules into a feature-based visual SLAM system [3]. Dead reckoning (DR) is implemented using an EKF-based framework following [28], fusing in robot odometry information. The system supports stereo and RGB-D modes, where for map point initialization the former uses stereo matching and the latter uses the depth measurement.

Denote the full Good Weights method as GW (DR in all modules) and GW:*X* to indicate variants with DR applied incrementally to each module. *DA* refers to feature data association (§III-C.1), *PE* to pose estimation in tracking (§III-C.2), and LBA to local bundle adjustment (§III-C.3). GW parameters are fixed for all evaluations as follows: $w_1 = 0.5$, $w_2 = 0.5$, $N^r_{det} = 600$, and $N^r_{trk} = 120$ (§III-B.3, Fig. 3); $\alpha_{min} = 10^{-1}$ and $\alpha_{max} = 10^3$ (§III-B.4, Fig. 4); and $C_{ref} = 20$ for initialization then adjusted online during local bundle adjustment (§III-C.3).

A. CID Open-Loop Benchmarking

The evaluation is first conducted on the public CID benchmark sequences [37], an indoor dataset with synchronized RGB-D images, IMU, and wheel odometry measurements, with Ground-truth generated using a high-accuracy LiDAR SLAM system. It contains 22 challenging sequences with diverse motion profiles and low-texture scenes. Two multi-floor sequences were excluded given that the data capture device was lifted to change floors and the method currently assumes available wheel odometry. The baseline methods are: ORB_SLAM3 [2] in RGBD–Inertial mode, Ground-Fusion [16] as an RGBD-Inertial–Wheel odometry system, and PGD-VIO [42], a plane-enhanced RGBD-Inertial system, together with its variants: *w/o P* with only point features, and *w/o G* with point and planar features. Since the PGD-VIO code is not public, reported performance is from [42].

Evaluation follows the protocol in [42]: Tracking accuracy is evaluated using APE RMSE over five runs per sequence. A sequence is marked as failed if any run has RMSE above 10 m or a tracking ratio below 50%. Fig. 6 depicts the sorted completeness (ratio of completed sequences to total) with secondary sorting by average RMSE. All experiments are performed on an Intel i7-12700K processor.

1) Result Analysis:

a) Frame Tracking Performance: From bottom to top, GFGG, ORB_SLAM3 and Ground-Fusion fail to achieve full

TABLE I: Repeat-Run Evaluation on CID.

Method	Frame RMSE (m)		R(F/KF)	
	Loop-2	Loop-3	Loop-2	Loop-3
GW:DA	0.274	0.333	2.920	4.312
GW:PE	0.246	0.234	1.092	1.032
GW:LBA	0.243	0.242	1.092	1.074
GW	**0.176**	**0.169**	1.092	1.053

completeness (80%, 80% and 85%, respectively), suggesting that vision-only systems and tightly coupled sensor-fusion approaches lack robustness under these conditions. In low-texture environments, poor visual tracking forces inertial or wheel-odometry cues to dominate the optimization; however without sufficient visual correction they accumulate drift, leading to degraded accuracy. For GW, full completeness is first achieved when DR is applied to feature data association (DA), with further accuracy gained when extended to pose estimation (PE). In both cases, DR stabilizes the front-end tracking thread, strengthening feature association and ensuring continuous and robust pose estimation. When integrated into the back-end, both GW:LBA and GW achieve an RMSE of 0.26 m, representing a 35% improvement over the front-end variants.

The performance of GW closely matches the planar variant of PGD-VIO *w/o G* (RMSE: 0.25 m) and outperforms the point-only baseline (*w/o P*) by 29% (RMSE: 0.37 m). The full PGD-VIO with graph-based drift suppression achieves a lower error of 0.19 m, reflecting its explicit geometric registration and drift correction in highly planar indoor environments. In contrast, our approach relies only on point reprojection error, reinforced with adaptively weighted DR according to tracking quality, yet still reaches accuracy on-par with plane-based methods without the added complexity of plane extraction or graph maintenance for plane matching. This shows that a lightweight, adaptively DR-augmented point-based visual SLAM can deliver competitive accuracy while remaining practical for real-robot deployment.

b) Global-BA and Map-based SLAM Performance: To further assess the effect of global bundle adjustment with DR (§III-C.4) and to present how the visual map contributes to SLAM accuracy, we adopt the repeat-run protocol from [27]. In this setup, each sequence is executed for three continuous loops without resetting the system, so the visual map constructed is retained and reused for the subsequent runs, and the frame-level RMSE is then computed over these additional loops. Table. I summarizes the average RMSE across CID sequences, along with the ratio between frame and keyframe RMSE, defined as $R(F/KF) = \frac{Frame\ RMSE}{KeyFrame\ RMSE}$, This ratio reflects how closely frame poses align with the optimized keyframe poses maintained in the SLAM map; values closer to 1.0 indicate that frame accuracy is nearly as good as keyframe accuracy, demonstrating stronger map consistency.

The first observation is that GW achieves RMSEs of 0.176 m and 0.169 m in the second and third loops, respectively. Frame-level tracking errors converge toward keyframe accuracy, with the error ratio approaching 1.0. This demonstrates the benefit of reusing the SLAM map in passive visual SLAM, allowing frame poses to be refined to near keyframe-level accuracy. DR in GW:PE and GW:LBA shows a similar trend but with reduced accuracy compared to GW, largely because weak visual constraints yield poor keyframe connectivity and a less stable BA problem. The GW:DA variant, which applies DR only at the front-end for feature tracking, provides little benefit in low-texture scenarios with no visual measurements and can lead to incorrect estimates. These observations highlight the importance of integrating DR information consistently across the hierarchical modules of visual SLAM to keep the system well-conditioned. Of note, repeat-run, map reuse GW performance matches that of first run PGD-VIO (recall that PGD-VIO explicitly uses depth information). Overall, the results show that with DR assistance, visual SLAM can reliably complete trajectory tracking on the first pass and then exploit the constructed map by adaptively balancing DR and visual constraints, which is a capacity that conventional odometry-only fusion systems generally lack.

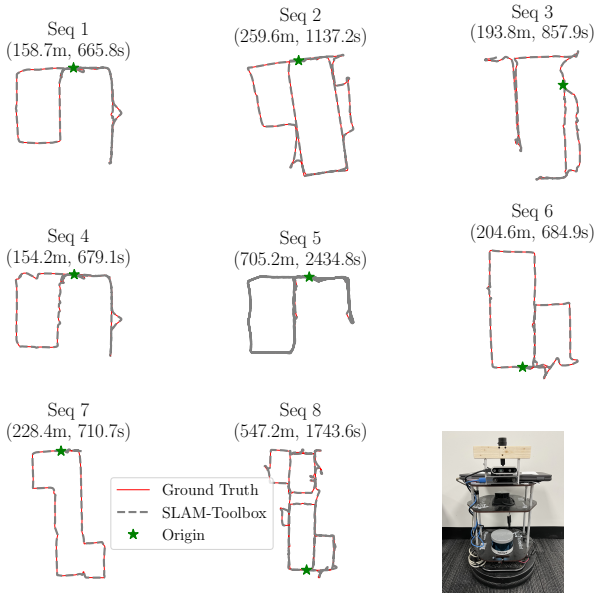

Fig. 7: Trajectories of sequences collected in an office environment, with the robotic platform shown in the bottom-right corner.

B. Real Robot Navigation Sequences

We collected navigation sequences using a TurtleBot2 equipped with a RealSense D435i (stereo and depth at 30 Hz, IMU at 400 Hz) and onboard robot odometry at 20 Hz from a 3-axis gyroscope and wheel encoders. Ground-truth trajectories were obtained with a 2D LiDAR (RPLIDAR-S2, 10 Hz) using SLAM-Toolbox [43] and refined through spline-based smoothing [44].

In contrast to the CID dataset, our sequences were gathered under fully autonomous navigation with SLAM-Toolbox and the complete ROS navigation stack [45], following the methodology in [27]. Waypoints were either transmitted remotely or predefined on a floor map. In total, eight sequences were recorded, covering both single-round exploration and multiple re-visits, as illustrated in Fig. 7.

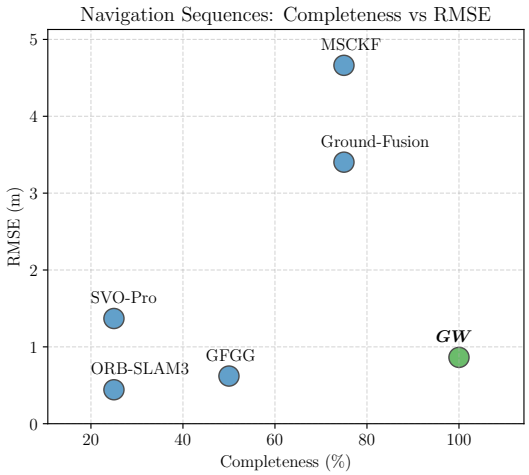

Fig. 8: Performance comparison of SLAM methods on navigation sequences.

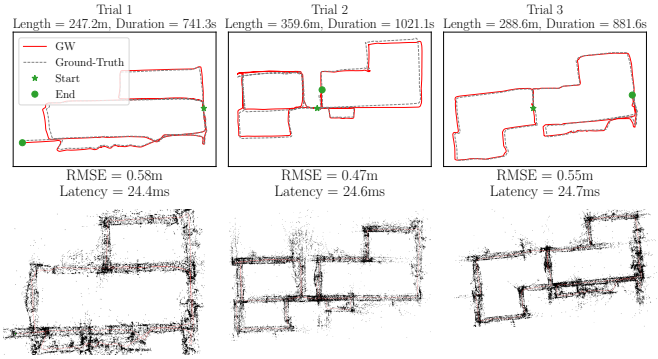

Fig. 9: Closed-loop navigation evaluation in an office environment. Top row: estimated trajectories (red solid) against ground truth (gray dashed) for three distinct navigation trials, with trajectory RMSE (m) and median tracking latency (ms) below. Bottom row: visual maps reconstructed during the corresponding trials.

Baseline methods include established Stereo Visual(-Inertial) SLAM/Odometry systems: ORB_SLAM3 [2], GFGG [3], MSCKF [8], SVO-Pro [46], DSOL [46] and Ground-Fusion (RGBD-Inertial-Wheel) [16]. Performance measures are sequence average RMSE and Completeness in Fig. 8. PGD-VIO is not available for benchmarking.

1) Result Analysis: GW is the only method to achieve full trajectory completeness, with an RMSE of 0.87 m. Compared to the vision-only baseline GFGG, completeness more than doubles while maintaining similar accuracy, confirming the effectiveness of incorporating DR. Focusing on completeness, the next best results are MSCKF and Ground-Fusion at 75%, with RMSE values of 4.6 m and 3.4 m, respectively. These results strengthen the benefits of fusing inertial and wheel odometry with visual odometry, yet they still fall short of GW due to the lack of adaptive weighting and insufficient exploitation of visual maps when reliable. SVO-Pro and ORB_SLAM3 perform the worst, failing frequently in long or low-texture sequences, demonstrating limited robustness for real robot navigation. DSOL consistently fails across all sequences, likely due to drift accumulation inherent in direct odometry methods; it is excluded from comparison.

C. Real-Robot Closed-loop Navigation Performance

To validate that the open-loop outcomes translate to actual deployment, GW is tested on real robot closed-loop navigation tasks. Using the navigation stack in Sec. IV-B, we replace the SLAM component with our method and store the LiDAR scans to later generate ground-truth trajectories offline using SLAM-Toolbox. Baseline methods are not included, as they failed to complete open-loop trajectories. Testing involved three navigation trials with different navigation paths in an office environment. Fig. 9 shows the trajectories (top row) and the corresponding reconstructed maps (bottom row). Across all trials, GW achieved stable navigation performance, with an average trajectory RMSE of 0.54 m and a tracking latency of 24.6 ms, while operating in real time on 30 Hz image streams. The consistent visual maps and low trajectory errors confirm that the proposed approach is reliable for practical navigation tasks.

V. Conclusion

This paper describes *Good Weights*, a visual SLAM framework for proactive, adaptive integration of dead-reckoning priors into the tracking, local mapping, and global bundle adjustment modules. Unlike LiDAR-based systems, which benefit from dense geometric constraints and rely on post-optimization corrections such as reweighting or smoothing, visual SLAM depends on sparse feature associations and becomes ill-conditioned when those associations fail. *Good Weights* addresses this structural limitation by regulating the influence of dead reckoning (DR) based on visual tracking quality to ensure well-conditioned optimization throughout the pipeline. Experiments on public benchmarks and real robot datasets show that this design surpasses existing DR-aided visual SLAM approaches and achieves repeat-run performance comparable to depth-based SLAM.

References

[1] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras," *IEEE T-RO*, vol. 33, no. 5, pp. 1255–1262, 2017.

[2] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM," *IEEE T-RO*, vol. 37, no. 6, pp. 1874–1890, 2021.

[3] Y. Zhao, J. S. Smith, and P. A. Vela, "Good Graph to Optimize: Cost-Effective, Budget-Aware Bundle Adjustment in Visual SLAM," *ArXiv*, vol. abs/2008.10123, 2020.

[4] W. Wang, Q. Zhang, Y. Hu, M. Gallay, W. Zheng, and J. Guo, "Recent advances in slam for degraded environments a review," *IEEE Sensors Journal*, 2025.

[5] M. Bujanca, X. Shi, M. Spear, P. Zhao, B. Lennox, and M. Luján, "Robust SLAM Systems: Are We There Yet?" in *IEEE/RSJ IROS*, 2021.

[6] Y. Wu, J. Kuang, X. Niu, J. Behley, L. Klingbeil, and H. Kuhlmann, "Wheel-SLAM: Simultaneous localization and terrain mapping using one wheel-mounted imu," *IEEE Robotics and Automation Letters*, 2022.

[7] T. Qin, P. Li, and S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[8] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *IEEE ICRA*, 2007, pp. 3565–3572.

[9] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. P. Huang, "OpenVINS: A Research Platform for Visual-Inertial Estimation," *IEEE International Conference on Robotics and Automation*, pp. 4666–4672, 2020.

[10] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: an open-source library for real-time metric-semantic localization and mapping," in *IEEE ICRA*, 2020.

[11] J. Zheng, K. Zhou, and J. Li, "Visual-inertial-wheel slam with high-accuracy localization measurement for wheeled robots on complex terrain," *Measurement*, vol. 243, p. 116356, 2025.

[12] K. Wu, C. X. Guo, G. A. Georgiou, and S. I. Roumeliotis, "VINS on wheels," *IEEE International Conference on Robotics and Automation*, pp. 5155–5162, 2017.

[13] W. Lee, K. Eckenhoff, Y. Yang, P. Geneva, and G. P. Huang, "Visual-Inertial-Wheel Odometry with Online Calibration," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4559–4566, 2020.

[14] V. Pritzl, M. Vrba, C. Tortorici, R. Ashour, and M. Saska, "Adaptive estimation of uav altitude in complex indoor environments using degraded and time-delayed measurements with time-varying uncertainties," *Robotics and Autonomous Systems*, vol. 160, p. 104315, 2023.

[15] V. Hulchuk, J. Bayer, and J. Faigl, "LiDAR-Visual-Inertial Tightly-coupled Odometry with Adaptive Learnable Fusion Weights," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2024, pp. 5641–5647.

[16] J. Yin, A. Li, W. Xi, W. Yu, and D. Zou, "Ground-fusion: A low-cost ground slam system robust to corner cases," in *IEEE International Conference on Robotics and Automation*, 2024, pp. 8603–8609.

[17] Y. Jia, H. Luo, F. Zhao, G. Jiang, Y. Li, J. Yan, and Z. Jiang, "Lvio-Fusion: A Self-adaptive Multi-sensor Fusion SLAM Framework Using Actor-critic Method," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 286–293, 2021.

[18] J. Mahmoud, A. Penkovskiy, H. T. Long Vuong, A. Burkov, and S. Kolyubin, "Rvwo: A robust visual-wheel slam system for mobile robots in dynamic environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2023, pp. 3468–3474.

[19] J. Lee, R. Komatsu, M. Shinozaki, T. Kitajima, H. Asama, Q. An, and A. Yamashita, "Switch-SLAM: Switching-Based LiDAR-Inertial-Visual SLAM for Degenerate Environments," *IEEE Robotics and Automation Letters*, vol. 9, pp. 7270–7277, 2024.

[20] J. Xu, G. Huang, W. Yu, X. Zhang, L. Zhao, R. Li, S. Yuan, and L. Xie, "Selective Kalman Filter: When and How to Fuse Multi-Sensor Information to Overcome Degeneracy in SLAM," *ArXiv*, vol. abs/2412.17235, 2024.

[21] T. Tuna, J. Nubert, P. Pfreundschuh, C. Cadena, S. Khattak, and M. Hutter, "Informed, Constrained, Aligned: A Field Analysis on Degeneracy-Aware Point Cloud Registration in the Wild," *IEEE Transactions on Field Robotics*, vol. 2, pp. 485–515, 2024.

[22] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," *IEEE International Conference on Robotics and Automation*, pp. 809–816, 2016.

[23] S. Zhao, H. Zhu, Y. Gao, B. Kim, Y. Qiu, A. M. Johnson, and S. Scherer, "SuperLoc: The Key to Robust Lidar-Inertial Localization Lies in Predicting Alignment Risks Superodometry.Com/SuperLoc," *IEEE International Conference on Robotics and Automation*, pp. 14 080–14 086, 2024.

[24] F. Han, H. Zheng, W. Huang, R. Xiong, Y. Wang, and Y. Jiao, "DAMS-LIO: A Degeneration-Aware and Modular Sensor-Fusion LiDAR-inertial Odometry," *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2745–2751, 2023.

[25] S. Chen, Y. Feng, C.-Y. Wen, Y. Zou, and W. Chen, "Stereo visual inertial pose estimation based on feedforward and feedbacks," *IEEE/ASME Transactions on Mechatronics*, vol. 28, no. 6, pp. 3562–3572, 2023.

[26] R. Mur-Artal and J. D. Tardós, "Visual-Inertial Monocular SLAM With Map Reuse," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 796–803, 2017.

[27] Y. Du, S. Feng, C. G. Cort, and P. A. Vela, "Task-driven SLAM Benchmarking For Robot Navigation," *arXiv preprint arXiv:2409.16573*, 2024.

[28] S. Lynen, M. W. Achtelik, S. Weiss, M. Chli, and R. Siegwart, "A robust and modular multi-sensor fusion approach applied to MAV navigation," in *IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2013, pp. 3923–3929.

[29] J. Nubert, T. Tuna, J. Frey, C. Cadena, K. J. Kuchenbecker, S. Khattak, and M. Hutter, "Holistic Fusion: Task- and Setup-Agnostic Robot Localization and State Estimation with Factor Graphs," *ArXiv*, vol. abs/2504.06479, 2025.

[30] G. P. C. Júnior, A. M. C. Rezende, V. R. F. Miranda, R. Fernandes, H. Azpúrua, A. A. Neto, G. Pessin, and G. M. Freitas, "EKF-LOAM: An Adaptive Fusion of LiDAR SLAM With Wheel Odometry and Inertial Data for Confined Spaces With Few Geometric Features," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 3, pp. 1458–1471, 2022.

[31] J. Nocedal and S. J. Wright, "Numerical optimization," in *Fundamental Statistical Inference*, 2018.

[32] A. Ranganathan, "The levenberg-marquardt algorithm," *Tutoral on LM algorithm*, vol. 11, no. 1, pp. 101–110, 2004.

[33] A. R. Conn, N. I. Gould, and P. L. Toint, *Trust region methods*. SIAM, 2000.

[34] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment—a modern synthesis," in *International workshop on vision algorithms*. Springer, 1999, pp. 298–372.

[35] Y. Qiu, Y. Chen, Z. Zhang, W. Wang, and S. A. Scherer, "MAC-VO: Metrics-Aware Covariance for Learning-Based Stereo Visual Odometry," *IEEE International Conference on Robotics and Automation*, pp. 3803–3814, 2024.

[36] K. Ebadi, L. Bernreiter, H. Biggie, G. Catt, Y. Chang, A. Chatterjee, C. E. Denniston, S.-P. Deschênes, K. Harlow, S. Khattak *et al.*, "Present and future of slam in extreme environments: The darpa subt challenge," *IEEE Transactions on Robotics*, vol. 40, pp. 936–959, 2023.

[37] Y. Zhang, N. An, C. Shi, S. Wang, H. Wei, P. Zhang, X. Meng, Z. Sun, J. Wang, W. Liang *et al.*, "CID-SIMS: Complex indoor dataset with semantic information and multi-sensor data from a ground wheeled robot viewpoint," *The International Journal of Robotics Research*, 2023.

[38] L. Carlone and S. Karaman, "Attention and Anticipation in Fast Visual-Inertial Navigation," *IEEE Transactions on Robotics*, vol. 35, pp. 1–20, 2019.

[39] Y. Zhao and P. A. Vela, "Good feature matching: Towards accurate, robust VO/VSLAM with low latency," *IEEE T-RO*, vol. 36, no. 3, pp. 657–675, 2020.

[40] M. Staniaszek, T. Flatscher, J. Rowell, H. Niu, W. Liu, Y. You, M. Gadd, M. Mattamala, A. Schutz, D. de Martini, L. Pitt, R. Skilton, M. Fallon, and N. Hawes, "AutoInspect: Toward Long-Term Autonomous Inspection and Monitoring," *IEEE Transactions on Field Robotics*, vol. 2, pp. 529–548, 2025.

[41] D. Lattanzi and G. Miller, "Review of robotic infrastructure inspection systems," *Journal of Infrastructure Systems*, vol. 23, no. 3, p. 04017004, 2017.

[42] Y. Zhang, F. Tang, Z. Xu, Y. Wu, and P. Ma, "PGD-VIO: A Plane-Aided RGB-D Inertial Odometry With Graph-Based Drift Suppression," *IEEE Robotics and Automation Letters*, vol. 10, no. 5, pp. 4276–4283, 2025.

[43] S. Macenski and I. Jambrecic, "SLAM Toolbox: SLAM for the dynamic world," *Journal of Open Source Software*, vol. 6, no. 61, p. 2783, 2021.

[44] E. Mueggler, G. Gallego, H. Rebecq, and D. Scaramuzza, "Continuous-Time Visual-Inertial Odometry for Event Cameras," *IEEE Transactions on Robotics*, vol. 34, pp. 1425–1440, 2017.

[45] ROS Navigation Tutorial. [Online]. Available: https://wiki.ros.org/navigation/Tutorials/RobotSetup

[46] C. Qu, S. S. Shivakumar, I. D. Miller, and C. J. Taylor, "DSOL: A Fast Direct Sparse Odometry Scheme," in *IEEE/RSJ IROS*, 2022, pp. 10 587–10 594.