Increasing SLAM Pose Accuracy by Ground-to-Satellite Image Registration

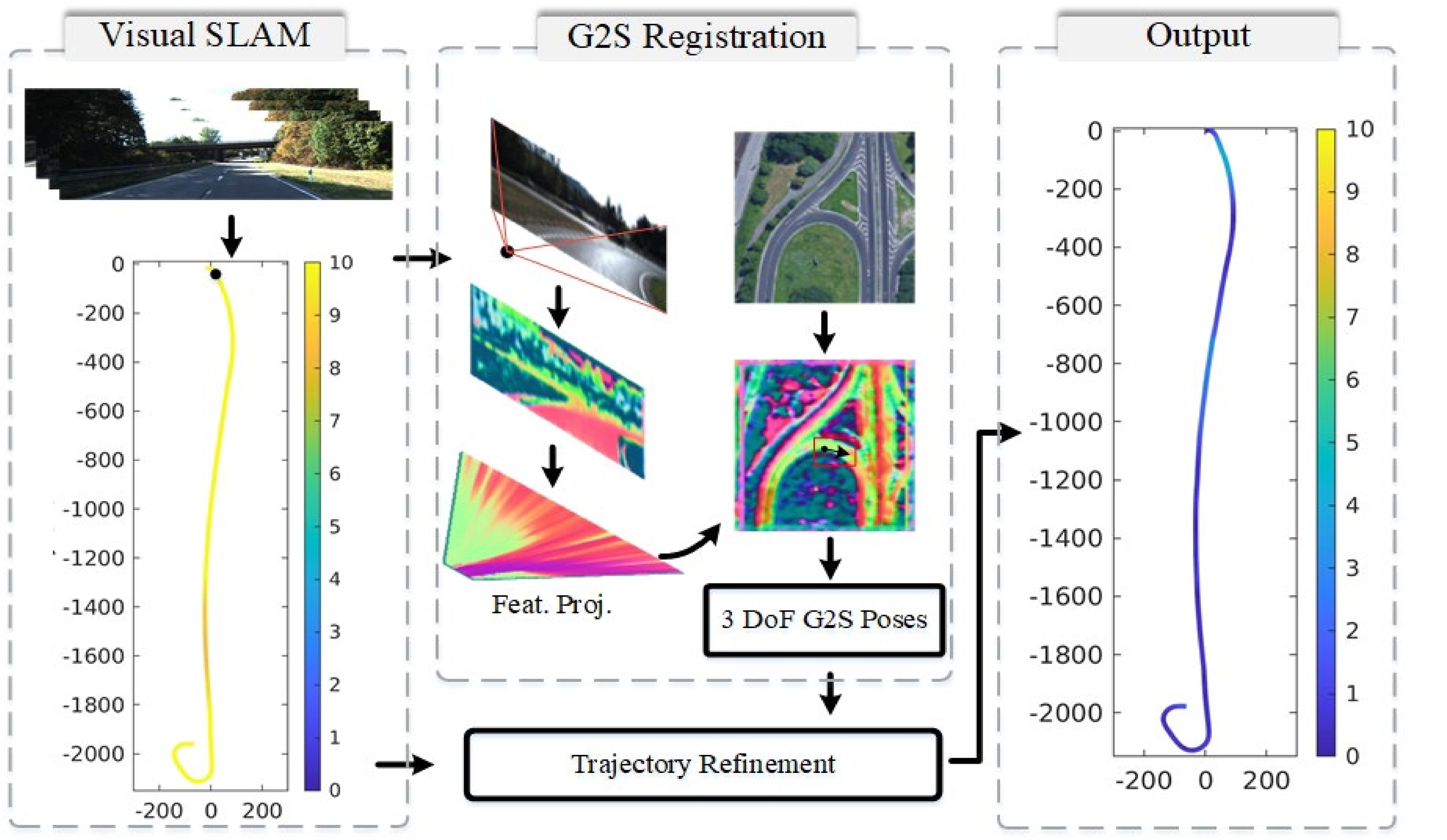
Yanhao Zhang[1], Yujiao Shi[2], Shan Wang[3,4], Ankit Vora[5], Akhil Perincherry[5], Yongbo Chen[3] , and Hongdong Li[3]

1 Robotics Institute, University of Technology Sydney, Sydney, Australia (yanhao.zhang@uts.edu.au);
2 ShanghaiTech University, Shanghai, China
3 College of Engineering and Computer Science, Australian National University.
4 Data61, CSIRO, Canberra, Australia.
5 Ford Motor Company, Dearborn, USA

Background & Motivation

- Visual SLAM suffers from long-term drift owing to the error accumulation.
- This is especially a problem for autonomous driving when a vehicle moves from one place to another without any loops.
- The ground-to-satellite (G2S) registration predicts the relative pose using satellite images. Pros: no error accumulation. Cons: not robust enough.

- We proposes a G2S-SLAM-Fusion method:
  - A coarse-to-fine method to select valid G2S poses.
  - An iterative refinement method to fuse the G2S poses using a scaled pose graph.

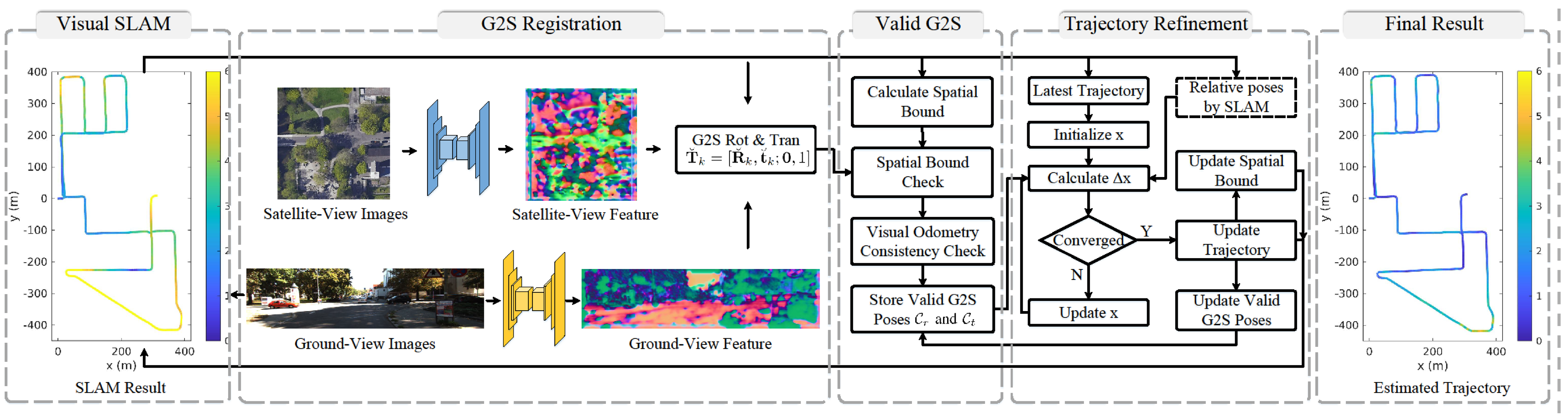


Algorithm 1: Iterative G2S-SLAM Fusion

Input: SLAM poses, ground and satellite images.
Output: Estimated vehicle trajectory.

```
Initialize poses T_k^0 = T̃_k;
Initialize spatial bound B_k^0;
Initialize G2S prediction for the first frame: Ť_0^0 = I_4;
Calculate visual odometry weights by Sec. IV-C.2;
for all other poses do
    Step 1: G2S Pose Prediction:
    Calculate G2S pose Ť_k^t using the trajectory pose
      T_k^t and the images {I_k^g, I_k^s};   ▷ 't' denotes the
      latest updated trajectory. t = 0 at the beginning.
    Step 2: Check validity:
    Calculate spatial bound B_{k-1}^t and B_k^t;
    if ť_{k-1}^t ∈ B_{k-1}^t & ť_k^t ∈ B_k^t then
        Calculate relative pose Ť_{k-1,k}^t and T_{k-1,k}^t;
        Visual odometry consistency check (4);
    end
    Step 3: Trajectory Refinement:
    if Ť_k^t is selected then
        Solve the nonlinear least squares problem (5);
        Update the spatial bound B_{t+1}^{t+1}, ···;   ▷ For
          checking the rest G2S predictions.
        Update all selected predictions C_r^{t+1}, C_t^{t+1}.  ▷
          Making the selected predictions w.r.t. the
          latest trajectory for the next trajectory
          refinement.
    end
end
```

Method



G2S Prediction:

- Given the input SLAM pose $\{\tilde{\mathbf{R}}_k, \tilde{\mathbf{t}}_k\}$, G2S aims to predict the pose change $\{\breve{\mathbf{R}}_k, \breve{\mathbf{t}}_k\}$:

$$\breve{\mathbf{R}}_k = \mathbf{R}_k^T \cdot \tilde{\mathbf{R}}_k, \quad \breve{\mathbf{t}} = \tilde{\mathbf{R}}_k^T \cdot (\mathbf{t}_k - \tilde{\mathbf{t}}_k)$$

Valid G2S Pose Selection:

- Spatial Bound: to check if a G2S prediction is within a boundary proportional to that by the trajectory covariance $\mathbf{\Phi}$. The boundary:

$$b(\alpha) = \frac{3}{n}\begin{bmatrix}\cos\Theta(\mathbf{R}_k) & \sin\Theta(\mathbf{R}_k)\\ \sin\Theta(\mathbf{R}_k) & \cos\Theta(\mathbf{R}_k)\end{bmatrix} \cdot \mathbf{\Phi}_k^{1/2} \cdot \begin{bmatrix}\cos(\alpha)\\ \sin(\alpha)\end{bmatrix}, \quad n = \mathrm{mean}(\lambda_1, \lambda_2)/r$$

- Odometry Consistency: to check if the relative pose by G2S $\breve{\mathbf{T}}_{k-1,k} = \breve{\mathbf{T}}_k^{-1}\mathbf{T}_{k-1,k}\breve{\mathbf{T}}_k$ and that by the current trajectory $\mathbf{T}_{k-1,k} = \mathbf{T}_{k-1}^{-1}\mathbf{T}_k$ is within a threshold.

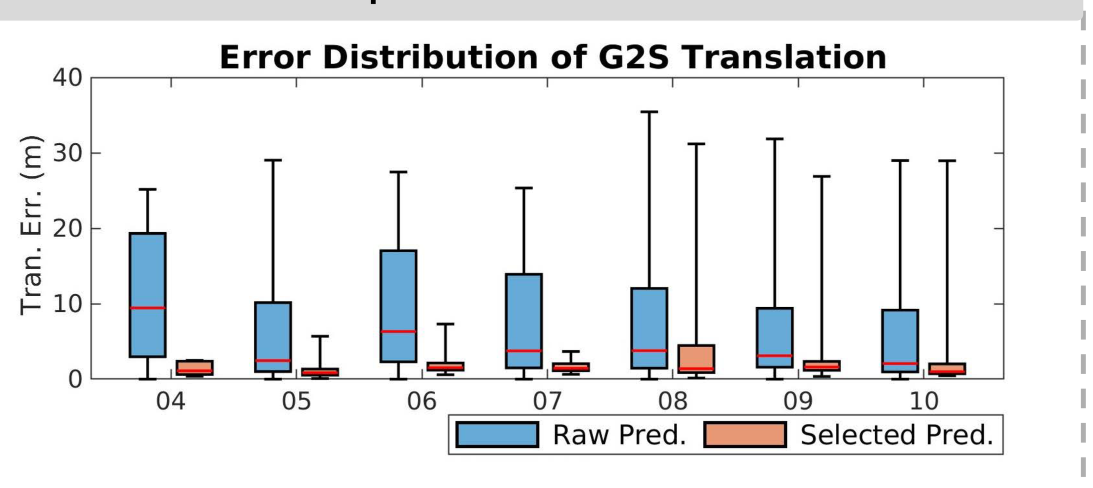
$$C_r = \{\breve{\mathbf{R}}_k | \Theta(\breve{\mathbf{R}}_{k-1,k} \cdot \mathbf{R}_{k-1,k}^T) < \mathrm{th}_\theta\}, \quad C_t = \{\breve{\mathbf{t}}_k | \mathbf{e}_i^T \cdot (\breve{\mathbf{t}}_{k-1,k} - \mathbf{t}_{k-1,k}) < \mathrm{th}_t\},$$

Trajectory Refinement:

- The valid G2S pose and the visual odometry are fused together via a scaled pose graph.

$$\underset{\{\cdots \mathbf{R}_k, \mathbf{t}_k, s_k \cdots\}}{\operatorname{argmin}} \quad L_{\mathrm{G2S}} + L_{\mathrm{VO}} + L_s$$

Comparison with G2S



Meth.	Azimuth (°)		Longitudinal (m)		Lateral (m)	
	mean↓	1° (%) ↑	mean↓	1m (%) ↑	mean↓	1m (%) ↑
[7]	0.163	99.89%	7.651	20.39%	0.746	79.62%
Ours	0.231	97.97%	0.544	84.09%	0.485	89.94%

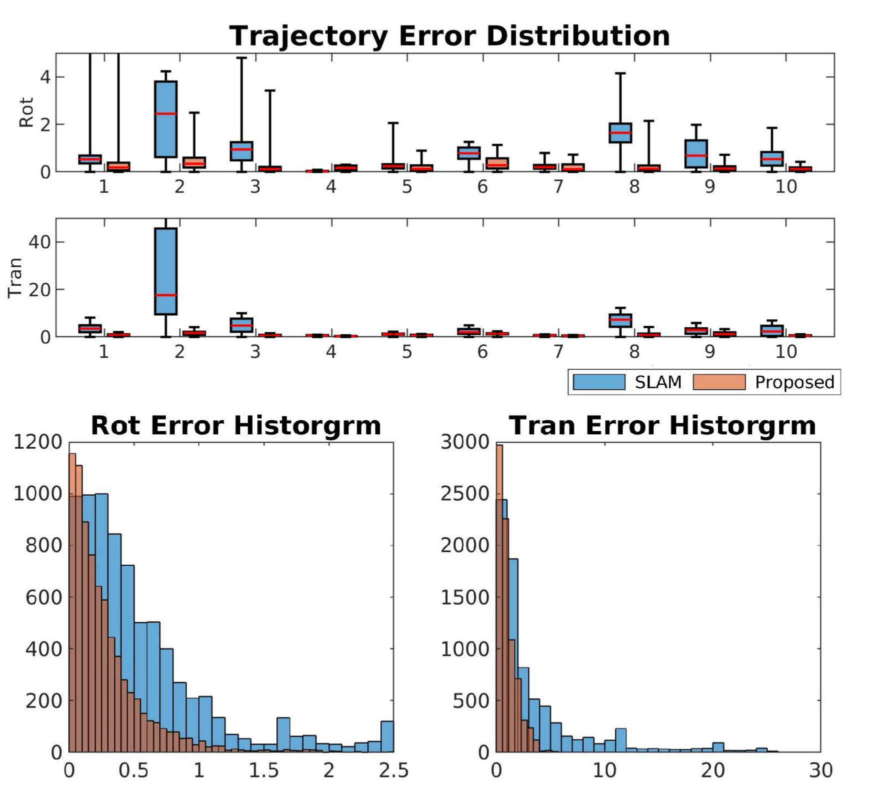
The translation accuracy is improved.

Comparison with SLAM

ACCURACY COMPARISON WITH SLAM RESULT USING KITTI DATASET

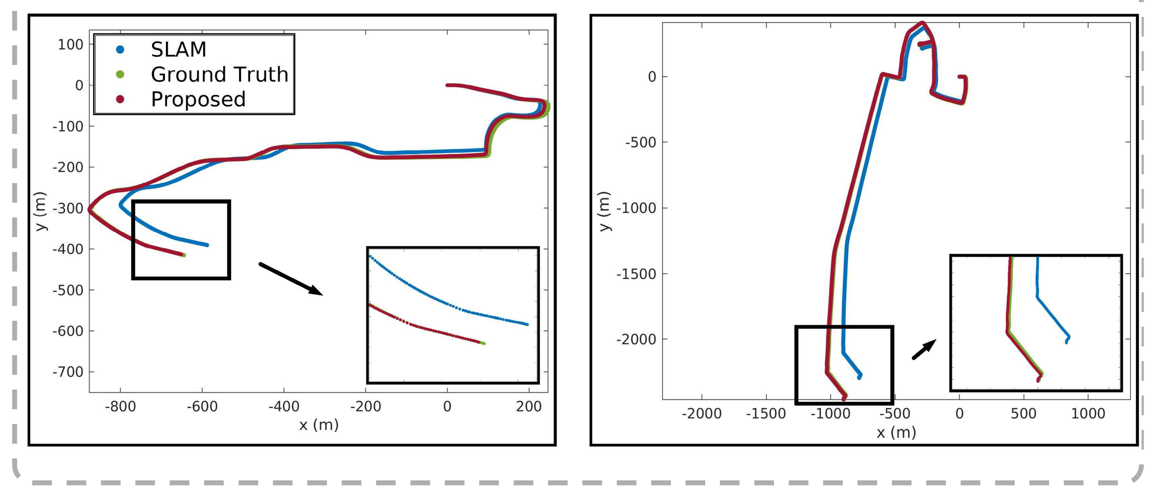
Sequence	S by Trajectory Origin						S by Multiple Ground Truth					
	$\theta^\dagger$ ↓	$\theta^\S$ ↓	$\theta^£$% ↑	$\mathbf{t}^\dagger$ ↓	$\mathbf{t}^\S$ ↓	$\mathbf{t}^£$% ↑	$\theta^\dagger$ ↓	$\theta^\S$ ↓	$\theta^£$% ↑	$\mathbf{t}^\dagger$ ↓	$\mathbf{t}^\S$ ↓	$\mathbf{t}^£$% ↑
00	0.731	0.491	32.73 %	4.144	0.946	77.17 %	0.545	0.545	-0.16%	1.081	1.306	-20.80%
01	2.644	0.726	72.56 %	31.978	1.516	95.26%	1.733	0.717	58.60%	15.00	1.529	89.80%
02	1.001	0.211	78.93 %	5.650	0.823	85.44%	0.547	0.214	60.97%	3.661	0.802	78.09%
04	0.037	0.188	-409.75%	0.625	0.363	41.94%	0.413	0.088	78.62%	0.193	0.344	-78.79%
05	0.304	0.231	23.90%	1.292	0.543	57.98%	0.257	0.288	-12.08%	0.922	0.435	52.77%
06	0.832	0.430	48.29%	2.657	1.311	50.64%	0.426	0.367	13.96%	0.912	0.768	15.73%
07	0.291	0.203	30.44%	0.640	0.512	19.99%	0.296	0.278	6.19%	0.419	0.355	15.25%
08	1.990	0.466	76.56%	7.460	1.168	84.35%	1.038	0.523	49.61%	4.323	2.073	52.06%
09	0.949	0.205	78.37%	3.080	1.553	49.57%	0.779	0.286	63.33%	1.903	1.254	34.11%
10	0.784	0.146	81.33%	3.538	0.646	81.73%	0.461	0.190	58.69%	0.906	0.509	43.82%
Avg.	0.956	**0.330**	65.52%	6.106	**0.938**	84.64%	0.650	**0.350**	46.17%	2.932	**0.938**	68.03%

θ: RMSE of absolute azimuth rotation (unit: °); $\mathbf{t}$ RMSE of absolute 2D translation (unit: m);
†: The stereo SLAM result using [8]; §: the result by the proposed method;
£: the accuracy improvement using $\frac{\text{SLAM error} - \text{our method error}}{\text{SLAM error}}$;
'↓': smaller error represents higher accuracy; '↑': higher percentage represents larger improvement.



- Overall, the proposed framework achieves higher accuracy.
- On average, the translation error reduces 64% (from 0.96° to 0.43°) and the rotation error reduces 83% (from $6m$ to $1m$).

Trajectory by FordAV



Trajectory by KITTI