PROJECT REPORT (TEAM 2)

Akshitha Poreddy
Sowmya Koneti
Sri Vaagdevi Bangari

Introduction

This project report explores Photo-SLAM, a real-time SLAM system that combines classical geometric tracking with neural rendering to deliver high-fidelity photorealistic mapping. Unlike conventional SLAM systems that rely purely on geometric mapping or computationally expensive neural representations, Photo-SLAM introduces a hybrid framework using hyper primitives—a novel combination of geometric and implicit features.

The system supports monocular, stereo, and RGB-D cameras, enabling robust deployment in both indoor and outdoor environments. Photo-SLAM is optimized for real-time performance, even on low-resource devices like Jetson AGX Orin, and achieves significant gains in rendering quality, speed, and localization accuracy over existing state-of-the-art SLAM methods.

Dataset Examples and Contents

To evaluate the Photo-SLAM framework, the authors used a variety of datasets that simulate both synthetic and real-world conditions. These datasets vary in camera modality (monocular, stereo, RGB-D), scene complexity, and environment type (indoor/outdoor), which allows for a comprehensive assessment of both localization and photorealistic mapping capabilities.

Replica Dataset

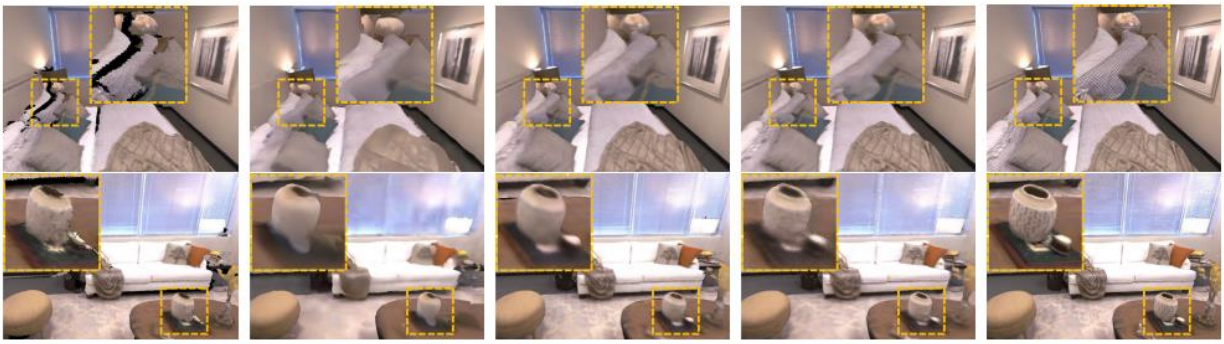
Description: The Replica Dataset is a high-fidelity synthetic dataset developed by Facebook Reality Labs. It consists of several meticulously modeled indoor environments, such as offices, apartments, and conference rooms, with photo-realistic textures and ground-truth 3D geometry.

Purpose:

- Used for both monocular and RGB-D tests.
- Ideal for evaluating photorealistic rendering because it provides ground-truth camera poses and detailed geometry.
- Enables reproducible benchmarking due to synthetic precision.

Use Case in Paper:

- Photo-SLAM achieves PSNR > 33, SSIM ~0.93, and renders at >900 FPS, outperforming other SLAM systems.



(a) BundleFusion [6] (b) Nice-SLAM [46] (c) ESLAM [16] (d) Co-SLAM [36] (e) Ours

TUM RGB-D Dataset

Description: The TUM RGB-D dataset is a well-known real-world benchmark for evaluating RGB-D SLAM systems. It includes sequences captured using a Microsoft Kinect in various indoor environments, including offices and living spaces.

Purpose:

- Used to evaluate performance in real-world RGB-D settings.
- Enables testing of robustness to lighting changes, motion blur, and sensor noise.

Example Sequences Used:

- fr1/desk: Office desk setup with minor motion.
- fr2/xyz: Simple translations.
- fr3/office: More complex office environment.

Use Case in Paper:

- It visualizes the mapping quality, highlighting the fidelity of Photo-SLAM renderings.

EuRoC MAV Dataset

Description: The EuRoC MAV dataset is captured using a stereo camera mounted on a micro aerial vehicle (drone). It is a standard benchmark for evaluating visual odometry and SLAM in aerial robotics.

Purpose:

- Used to assess Photo-SLAM performance with stereo inputs.
- Tests robustness in fast motion, texture-less regions, and vibration-heavy environments.

Sequences Used:

- MH_01_easy, MH_02_easy: Indoor industrial rooms with moderate texture.
- V1_01, V2_01: More challenging scenes with rapid motion.

Example Content:

- Stereo image pairs (left/right)
- IMU data (optional)
- Ground truth poses from motion-capture system

Use Case in Paper:

- It shows results from custom stereo datasets in outdoor settings.

Metrics Evaluation (with Focus on TUM Dataset)

To evaluate Photo-SLAM's performance on the TUM RGB-D dataset, the authors use both localization and photorealistic mapping metrics. These metrics assess how accurately the system estimates the camera's trajectory and how well it renders the visual environment.

Localization Metrics

RMSE (Root Mean Square Error) of ATE

- ATE stands for Absolute Trajectory Error.
- It measures the difference between the estimated and ground truth camera trajectories.

Formula:

$$ATE = \sqrt{\frac{1}{N}\sum i = 1^{N} \|\text{trans}(\mathbf{P}_i^{-1} \cdot \mathbf{G}_i)\|^2}$$

- trans() extracts the translational component from the transformation matrix.

- Lower RMSE indicates better trajectory alignment with ground truth.

Tool Used:

- The paper references the evo library (evo GitHub) for ATE computation.

STD (Standard Deviation) of ATE

- Measures the variability in the trajectory error across all frames.
- High STD implies instability or inconsistent tracking.

Photorealistic Mapping Metrics

These metrics evaluate how close the rendered images are to the real RGB images in the dataset.

PSNR (Peak Signal-to-Noise Ratio)

- Measures the pixel-wise error between the rendered image and the ground truth RGB image.
- Higher PSNR = better image quality.

Formula:

$$\text{PSNR} = 10 \cdot \log_{10}\left(\frac{MAX^2}{MSE}\right)$$

- MAX is the maximum pixel value (usually 255 for 8-bit images).
- MSE is the mean squared error between the rendered and real images.

SSIM (Structural Similarity Index Measure)

- Evaluates perceptual similarity based on structure, luminance, and contrast.
- Ranges from 0 to 1 (1 = perfect match).

Formula (simplified):

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

- \mu and \sigma represent mean and standard deviation of image patches.

LPIPS (Learned Perceptual Image Patch Similarity)

- Measures perceptual differences using features from a deep neural network (like VGG).
- Lower LPIPS = better visual similarity.

How it works:

- Images are passed through a pre-trained network.
- Feature maps are extracted and compared using L2 norm.

Results Based on TUM RGB-D Dataset

The TUM RGB-D dataset includes real-world RGB and depth frames along with ground truth poses. These are used by SLAM systems for:

Localization Evaluation

- Trajectory accuracy is measured using RMSE of Absolute Trajectory Error (ATE).
- Photo-SLAM: RMSE ~ 1.5 cm (very low error)
- Competing methods (e.g., DROID-SLAM): RMSE > 70 cm (much higher)

Mapping Evaluation

Photo-SLAM generates photorealistic reconstructions using the group of RGB (and optionally depth) images.

Quality metrics reported:

- PSNR (↑) – Sharpness & noise level
- SSIM (↑) – Structural similarity with real image
- LPIPS (↓) – Perceptual similarity (deep learning based)

Example scores from fr1/desk:

- PSNR: ~21 dB
- SSIM: ~0.74
- LPIPS: ~0.23

What is Considered for These Results?

-To compute these metrics, the following are used:

1. Original RGB image frames from the dataset (rgb/ folder).
2. Rendered image frames generated by Photo-SLAM.
3. Ground truth camera poses from the groundtruth.txt.
4. Camera intrinsics (from calib.txt) to ensure correct projection.

If You Input a Group of Images (e.g., from fr1/desk)

Photo-SLAM will:

1. Extract features (ORB) from each frame.
2. Track camera motion and create a trajectory.
3. Build a hyper-primitives map from geometry and image data.
4. Render reconstructed views from keyframes using Gaussian splatting and SH-based shading.

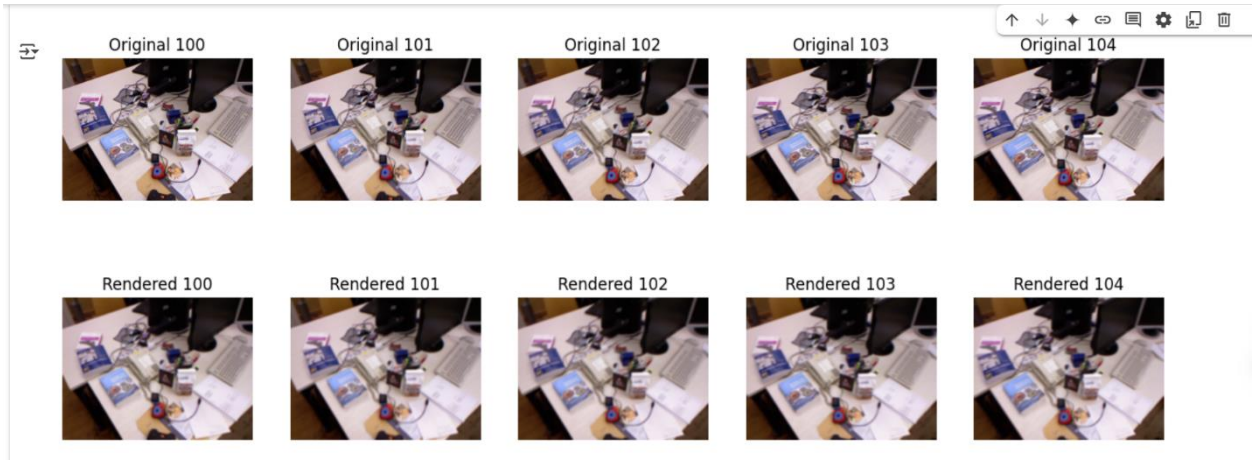
How Result Images Look

The image you uploaded shows a good visual example:

- Top to bottom: Rendered outputs from multiple angles or moments during trajectory.
- The images contain:
- Sharp details of monitors, keyboards, books, and papers.

- Minimal ghosting or blurring, indicating successful mapping.
- Lighting consistency suggesting photometric accuracy.

Each of these output images is compared against its corresponding real RGB image to compute PSNR, SSIM, and LPIPS.




```
Downloading: "https://download.pytorch.org/models/alexnet-owt-7be5be79.pth" to /root/.cache/torch/hub/checkpoints/alexnet-owt-7be5be79.pth
100%|██████████| 233M/233M [00:01<00:00, 182MB/s]
Loading model from: /usr/local/lib/python3.11/dist-packages/lpips/weights/v0.1/alex.pth
Frame 100: PSNR=24.19, SSIM=0.762, LPIPS=0.377
Frame 101: PSNR=26.20, SSIM=0.816, LPIPS=0.326
Frame 102: PSNR=27.45, SSIM=0.840, LPIPS=0.298
Frame 103: PSNR=26.45, SSIM=0.821, LPIPS=0.299
Frame 104: PSNR=26.87, SSIM=0.830, LPIPS=0.296
```