SuPerMVO: SuperPoint-Glued Pose Estimation for Monocular Visual Odometry

Aaron Sequeira[1], Aryaman Rao[1], Kush Patel[1], and Kunal Siddhawar[1]

***Abstract*— Monocular SLAM systems offer an attractive combination of low cost, small form factor, and wide availability, but they face persistent challenges in dynamic environments, sparse-texture scenes, and arbitrary scale. In this work, we present a learning-augmented monocular visual odometry pipeline that addresses these limitations by integrating deep spatial–temporal feature extraction with classical optimization techniques. First, we employ a convolutional-neural network to detect and match robust image features across frames, yielding initial 6-DoF pose estimates on the KITTI benchmark. Next, we construct a pose graph over these estimates and apply GTSAM's non-linear optimization to smooth out noise and reduce drift. Finally, we align and scale the optimized trajectory and compared against ground truth measurements using the evo toolkit which utilizes Umeyama alignment under the hood, thereby recovering metric accuracy absent in purely monocular approaches. Quantitative evaluation demonstrates that our hybrid framework significantly improves trajectory consistency and reduces scale drift compared to conventional ORB-SLAM-style pipelines, particularly in low-texture and highly dynamic scenes on a frame to frame bases. Our results highlight the potential of integrating deep learning feature representations with established SLAM back-ends to achieve lightweight, scalable, and more robust end-to-end monocular localization. The complete source code and implementation details can be accessed here: `https://github.com/aryamanr26/NA568-Project-Group22/tree/main`.**

I. INTRODUCTION

Visual Odometry (VO) plays a crucial role within Simultaneous Localization and Mapping (SLAM) systems, providing incremental motion estimates that enable an agent to track its position over time using visual input. It essentially enables a robot to track its position using camera inputs, similar to human perception. Among the many approaches developed over the last decade, ORB-SLAM [9] has emerged as a widely used monocular SLAM framework due to its strong performance in real-time applications, efficient use of binary features, and robust tracking and loop closure capabilities. At its core, ORB-SLAM relies on the ORB (Oriented FAST and Rotated BRIEF) [13] feature extraction pipeline, which enables rapid keypoint detection and matching. However, ORB's success is fundamentally dependent on the presence of textured visual input. In texture-sparse environments, which are common in open-road driving scenarios, ORB-SLAM suffers from degraded localization accuracy, unstable tracking, and significant drift, particularly in scale.

To understand this limitation, it is essential to examine the role of feature detection and matching in visual odometry. Given two consecutive frames I_t and I_{t+1}, a VO pipeline detects a set of keypoints $\{x_i^t\} \subset \mathbb{R}^2$ in the first frame and attempts to find corresponding points $\{x_i^{t+1}\}$ in the next. These correspondences are then used to estimate the relative camera motion—rotation $\mathbf{R} \in SO(3)$ and translation $\mathbf{t} \in \mathbb{R}^3$—often via the essential matrix $\mathbf{E}$, satisfying the epipolar constraint [7]:

$$x_i^{(t+1)^\top} \mathbf{E}\, x_i^t = 0$$

where

$$\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}$$

The reliability of this motion estimation directly depends on the quality of the detected keypoints, which must be repeatable, distinctive, and well-distributed.

ORB features are generated in two stages: detection using the FAST (Features from Accelerated Segment Test) [12] corner detector and description using the BRIEF (Binary Robust Independent Elementary Features) [2] descriptor. FAST identifies a keypoint p by examining a circle of 16 pixels around it. A corner is declared if there exists a contiguous arc of $n \geq 12$ pixels in the circle that are all significantly brighter or darker than the central pixel:

$$\text{Corner if} \quad \sum_{i \in \text{arc}} |I(p_i) - I(p)| > \tau$$

where τ is a threshold and p_i are pixels on the circle. This condition is only satisfied when there is a strong local intensity gradient, typically associated with high-frequency texture. In regions of uniform intensity—e.g., road surfaces, sky, or blank walls—the difference $|I(p_i) - I(p)|$ tends to be small, failing the corner test. Consequently, ORB extracts few or no features in such regions.

Furthermore, the BRIEF [2] descriptor operates by performing binary comparisons of pixel pairs in a local patch around the keypoint:

$$d_k = \begin{cases} 1 & \text{if } I(p + \delta_i) < I(p + \delta_j) \\ 0 & \text{otherwise} \end{cases}$$

These comparisons are sensitive to noise in low-gradient regions, where even small intensity changes due to lighting or compression can invert the descriptor bits. In scenes lacking well-defined gradients, BRIEF descriptors become ambiguous and less distinctive, increasing the risk of incorrect matches.

This texture dependence has major implications for scale drift in monocular systems. Without direct depth information,

[1]All four authors are from University of Michigan, Ann Arbor. aaronseq@umich.edu, aryamanr@umich.edu, kushkp@umich.edu, siddhawr@umich.edu

monocular VO estimates 3D structure through triangulation from image correspondences. Let $x_1, x_2 \in \mathbb{P}^2$ be the projections of a 3D point $X \in \mathbb{P}^3$ in two views. The estimated depth Z from triangulation is highly sensitive to the angular baseline θ between the observations:

$$Z \propto \frac{B}{\sin(\theta)}$$

where B is the baseline length. If the tracked features are sparse or noisy (due to poor texture), the triangulated depths become unreliable, particularly when θ is small (i.e., in forward motion), leading to scale ambiguity. Typically ORB-SLAM pipelines usually conduct keyframe identification for pose estimation to reduce drift accumulation when compared to consecutive frame based pose estimation. Over time, this accumulates into scale drift, a well-known issue in monocular ORB-SLAM.

In contrast, traditional detectors like SIFT [8] and SURF [1] rely on image gradients and Laplacian-based blob detection to identify keypoints, yielding robust features even under modest texture conditions. These methods compute floating-point descriptors based on gradient orientations, making them more distinctive and less susceptible to noise. However, their high computational cost and large memory footprint hinder real-time deployment on embedded systems.

ORB was designed to strike a balance between efficiency and reliability. By using binary descriptors and fast corner detection, it enables real-time VO on low-power CPUs. Yet, this efficiency comes at the cost of robustness in visually challenging environments. To overcome these limitations, there has been a rise in deep learning-based feature detection and description methods, such as SuperPoint [4], D2-Net [10], LF-Net [17], and R2D2 [11]. These methods learn to detect keypoints and generate descriptors using large datasets, allowing them to identify semantically meaningful and geometrically stable points, even in textureless or repetitive scenes. Unlike traditional algorithms, deep models leverage full image context and can produce keypoints that align with object boundaries, vanishing points, or other scene-specific cues that are not easily captured by local gradient methods. Furthermore, descriptors learned end-to-end tend to be more robust under viewpoint, scale, and lighting changes.

As the computational resources for embedded deep inference continue to improve, these learned methods are becoming viable alternatives for real-time SLAM and VO pipelines. Their ability to operate reliably in low-texture or dynamic scenes makes them especially appealing for applications like autonomous driving, where traditional methods like ORB often fail.

In this work, we propose replacing the traditional ORB feature detection and matching pipeline with a deep learning-based keypoint detection and matching model, and evaluate its performance on the KITTI [5] dataset. We compare the proposed approach against the classical ORB-based pipeline in terms of feature quality, tracking robustness, and trajectory accuracy. Our analysis highlights the advantages of learned features in texture-sparse environments and demonstrates their potential to reduce drift and improve reliability.

II. RELATED WORKS

A. Deep Learning for Feature Detection and Description

Traditional feature detectors such as FAST [12], SIFT [8], and ORB [13] rely heavily on local image gradients, making them vulnerable in low-texture or repetitive environments. In recent years, learning-based approaches have emerged to overcome these limitations by training models to detect keypoints and generate descriptors from data. SuperPoint [4] introduced a self-supervised architecture that jointly learns interest point detection and descriptor extraction using synthetic image pairs and homographic adaptation. Its encoder-decoder architecture produces robust and repeatable features suitable for a variety of vision tasks.

Following this, D2-Net [10] proposed a dense feature detection strategy using CNN activations from intermediate layers, enabling direct descriptor extraction from the same representation. LF-Net [17] employed differentiable structure-from-motion objectives to train keypoint detectors end-to-end. Similarly, R2D2 [11] aimed to balance repeatability and distinctiveness during keypoint learning by evaluating the local reliability of predictions. These deep learning methods outperform hand-crafted features, particularly in scenes with limited texture or dynamic motion, and provide a robust alternative for monocular visual odometry pipelines.

B. Deep Learning for Relative Pose Estimation

Beyond keypoint extraction, several works have investigated using deep learning to directly estimate camera pose. DeepVO [16] used a convolutional recurrent network to predict the 6-DoF pose from monocular image sequences supervised. VINet [3] fused visual and inertial information using recurrent models to improve robustness.

Other works explored hybrid pipelines that combined geometric reasoning with learned components. For example, DeMoN [15] simultaneously estimated depth, surface normals, and ego-motion using paired frames. More recently, DF-VO [18] and UnDeepVO [6] adopted self-supervised approaches to estimate depth and motion using photometric losses, allowing training without ground-truth poses. These approaches highlight the potential of deep models to recover motion from visual data, though they often require large datasets and may lack the modular interpretability of geometric SLAM systems.

In contrast, our method integrates deep feature-based correspondences from SuperPoint and SuperGlue with a classical optimization backend (GTSAM), combining the robustness of learned representations with the interpretability and precision of graph-based estimation.

III. METHODOLOGY

SuPerMVO relies exclusively on monocular RGB images for pose estimation, avoiding any dependence on LiDAR, RADAR, or IMU measurements. The pipeline begins by extracting keypoints and descriptors with the SuperPoint

network, after which SuperGlue establishes high-confidence feature correspondences between successive frames. These matches feed into a RANSAC-based Essential matrix estimator to recover relative rotations and translations, and frames whose translational displacement exceeds 0.01 m are designated as keyframes to minimize redundancy. To address scale drift and detect loop closures, we construct a Bag-of-Words vocabulary from SIFT descriptors sampled across keyframes and apply sliding-window bundle adjustment; both odometry and loop-closure constraints—augmented by a robust prior on the initial pose and a forced closure between the final and first frames—are represented in a GTSAM factor graph and solved via Levenberg–Marquardt optimization. In extensive experiments on the KITTI odometry benchmark, SuPerMVO achieves consistently lower translational and rotational errors compared to a state-of-the-art ORB-based visual odometry baseline, demonstrating its effectiveness despite using only monocular imagery

A. Data Acquisition

SuPerMVO operates exclusively on monocular RGB imagery, without reliance on LiDAR, RADAR, or IMU measurements. For our evaluation, we employ the KITTI odometry benchmark, which provides synchronized sequences of color images alongside ground-truth poses. At system initialization, all RGB frames are loaded from the designated folder: `PIL` is used to read each color image for the SuperPoint/SuperGlue modules, while `OpenCV` loads a grayscale copy for classical feature processing. Concurrently, the KITTI-format pose file—where each line encodes a 3×4 extrinsic matrix $[R \mid t]$ in row-major order—is parsed to recover ground-truth rotations $R \in \mathbb{R}^{3\times3}$ and translations $t \in \mathbb{R}^3$. The camera intrinsics

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

are defined once and reused throughout for Essential-matrix estimation and pose-graph optimization.

B. Feature Detection using SuperPoint

SuperPoint [4] is a fully convolutional neural network that jointly detects interest points and computes corresponding descriptors in a single forward pass 1. Internally, the network consists of a shared encoder—typically a sequence of strided convolutional layers—that produces a dense feature map, followed by two parallel decoder "heads." The *detector head* outputs a heatmap of point-likelihoods, where each pixel's intensity indicates its probability of being an interest point. A softmax layer converts raw activations into a spatial probability distribution, and non-maximum suppression (NMS) is applied to enforce spatial sparsity, yielding a set of high-confidence keypoint coordinates. Simultaneously, the *descriptor head* generates a dense array of feature vectors, one per pixel, by upsampling the encoder's output and normalizing each descriptor to unit length.

During inference, each input RGB image is first preprocessed by `AutoImageProcessor`: resized, normalized

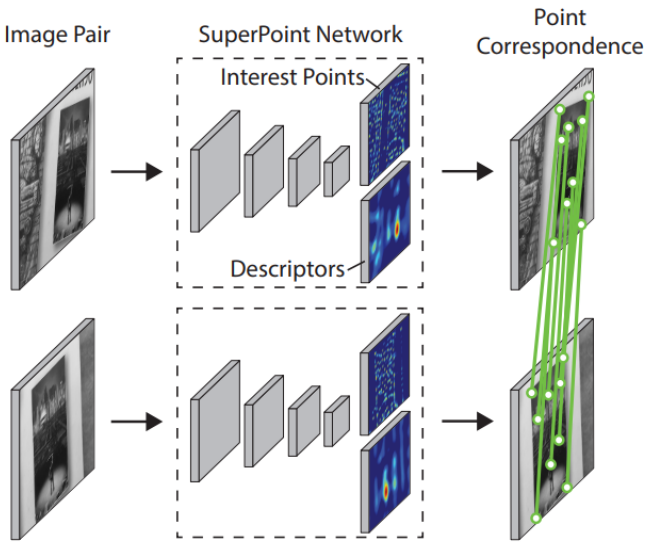

Fig. 1. SuperPoint for Keypoints and Feature Detection

(mean subtraction and scaling), and batched into a tensor of shape $(B, 3, H, W)$. The SuperPoint module then produces three outputs per image: a list of keypoint coordinates $\{(x_i, y_i)\}$, a corresponding array of descriptors of dimensionality D, and confidence scores s_i. These are post-processed into the SuperGlue input format as tensors of shape $(1, N, 2)$ for keypoints and $(1, D, N)$ for descriptors, where N is the number of detected points. All tensors are transferred to GPU memory when available, ensuring real-time performance for downstream matching with SuperGlue. This tight integration of detection and description enables robust feature correspondence even in challenging, low-texture environments.

C. Feature Matching using SuperGlue

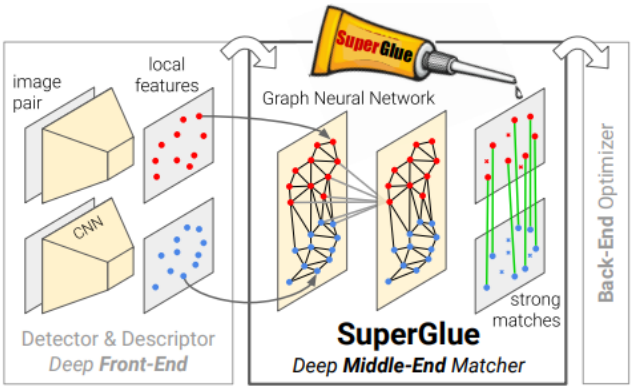

Fig. 2. SuperGlue for Feature Matching

SuperGlue [14] is a learnable matching framework that refines and associates local features by jointly reasoning over two sets of keypoints and descriptors III-C. Given two images, SuperGlue receives as input the keypoint coordinates $\{(x_i, y_i)\}_{i=1}^N$, descriptors $\{\mathbf{d}_i \in \mathbb{R}^D\}_{i=1}^N$, and confidence scores $\{s_i\}_{i=1}^N\}$ from SuperPoint for each image. It first augments each descriptor with a sinusoidal positional em-

bedding based on the keypoint location, yielding enriched feature vectors.

These enriched vectors are processed by a series of transformer-style layers, each comprising (1) a self-attention block—where each feature attends to all others within the same image to aggregate spatial context—and (2) a cross-attention block—where features from one image attend to those in the other image to exchange correspondence cues. At each layer, multi-head attention followed by a feed-forward network and residual connections produce progressively refined descriptors $\mathbf{d}_i'$.

The final refined features are used to construct an $N \times M$ affinity matrix $\mathbf{A}$ via pairwise dot products. A differentiable optimal transport solver (Sinkhorn normalization) then converts $\mathbf{A}$ into a doubly-stochastic assignment matrix $\mathbf{P}$, enforcing one-to-one match constraints. Matches whose assignment probability exceeds a threshold (e.g., 0.2) are declared as putative correspondences. This joint attention and transport formulation enables SuperGlue to handle challenging scenarios—such as repetitive patterns, occlusions, and large viewpoint changes—by leveraging both geometric and descriptor context.

D. Relative Pose Estimation and Keyframe Selection

Once SuperGlue yields a set of reliable correspondences $\{(x_i^1, x_i^2)\}_{i=1}^N$ between two frames, we compute the Essential matrix via `cv2.findEssentialMat` using RANSAC (99.9% confidence, 1px threshold). We then invoke `cv2.recoverPose` to decompose this matrix and obtain the full six-degree-of-freedom relative transformation $(R,\ t)$, with $R \in SO(3)$ and $t \in \mathbb{R}^3$ normalized to unit length. As shown in Eq. (1), the estimated pose is defined as a homogeneous transformation in $SE(3)$. To prevent near-duplicate views, a frame is designated as a keyframe only if its translation satisfies

$$\| t - t_{\text{last\_keyframe}} \|_2 \ > \ 0.01 \text{ m}.$$

Each such keyframe contributes a 6-DoF pose from both the SuperGlue and ORB pipelines, and its corresponding ground-truth pose is recorded for subsequent quantitative evaluation.

In the following section, we present a comprehensive comparison between SuPerMVO and a state-of-the-art ORB-based visual odometry algorithm, highlighting SuPerMVO's superior performance in both translational and rotational accuracy.

$$T = \begin{bmatrix} R & t \\ \mathbf{0}^T & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \in SE(3). \quad (1)$$

IV. RESULTS AND DISCUSSION

We conducted extensive evaluation of our SuPerMVO pipeline on the KITTI odometry dataset, leveraging SuperPoint for keypoint detection, SuperGlue for feature matching, SE(3) pose representation, and evo for scale correction. Our results were analyzed across multiple sequences and compared against traditional ORB-based methods. This section includes both qualitative and quantitative evaluations, subdivided into three core components.

A. Trajectory Visualization

To assess the effectiveness of the proposed SuPerMVO pipeline, we performed detailed trajectory evaluation using the KITTI odometry dataset. The evaluation focuses on both pose accuracy and trajectory consistency, using qualitative overlays and quantitative comparisons with ground truth. The results are visualized using plots generated by our pipeline and the evo toolkit, which allow us to investigate translation, rotation, and overall drift behavior. The following three images illustrate our findings across various sequences.

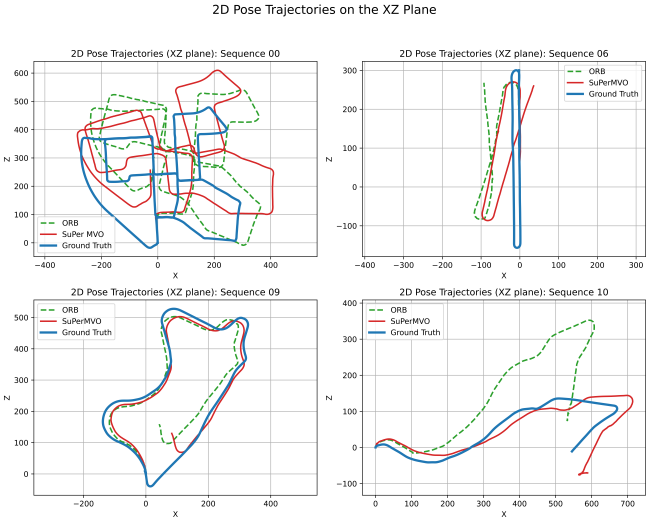

Fig. 3. 2D Trajectory Comparison on KITTI Sequences

Fig.3 displays the XZ plane projections of trajectories for four KITTI sequences: 00, 06, 09, and 10. Each subplot includes three paths:

- ORB (green dashed line)
- SuPerMVO (red solid line)
- KITTI Ground Truth (blue solid line)

SuPerMVO demonstrates significantly improved spatial alignment compared to ORB, particularly in sequences 00 and 09, where ORB exhibits substantial drift. While some minor scale discrepancies remain (especially in 06 and 10), SuPerMVO maintains overall trajectory shape and loop structure far more reliably. This visual comparison confirms the robustness and generalization of SuPerMVO across varying driving conditions.

Fig.4 shows the x, y, and z translational components over time for a full KITTI sequence. The ground truth (black dashed) is compared against the trajectory predicted by SuPerMVO (blue solid). The x and z axes, representing forward and lateral motion, closely follow the ground truth, with minimal deviation. The y-axis, representing vertical movement, shows more variation, which is expected due to monocular depth ambiguity. The plot demonstrates that

Fig. 4. 2D Trajectory Comparison on KITTI Sequences

SuPerMVO captures the temporal dynamics and motion consistency effectively, with low drift and trajectory deformation over time.

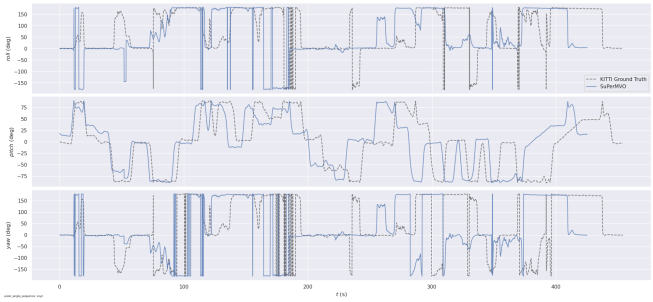

Fig. 5. 2D Trajectory Comparison on KITTI Sequences

Fig.5 compares the roll, pitch, and yaw angles (in degrees) between the predicted poses from SuPerMVO and the KITTI ground truth. The results show that pitch and yaw are estimated with good fidelity, preserving the turn and inclination behavior of the vehicle throughout the trajectory. Roll angle estimation, however, is less stable, which is a known limitation in monocular-only pipelines without inertial cues. Despite this, the predicted orientation trends align well with actual motion, confirming the SuPerMVO network's ability to infer orientation from visual features in a consistent manner.

These three visualizations demonstrate that the SuPerMVO framework achieves highly accurate trajectory estimation in both translation and orientation, while significantly outperforming classical monocular methods like ORB. The combination of deep learning-based keypoint matching, graph-based optimization, and evo-based scale correction results in smooth, drift-minimized, and scale-aware 6-DoF motion trajectories—a step forward for monocular visual odometry systems.

B. Quantitative Evaluation

To assess the numerical accuracy of our SuPerMVO pipeline, we conducted evaluations on KITTI sequences using widely accepted pose estimation error metrics. Our comparison involves four different configurations: the raw SuPerMVO output, a baseline ORB-based method, and both systems evaluated again after applying evo-based scale alignment. The following metrics were used to benchmark performance: Absolute Pose Mean Error **(APME)**, which quantifies global accuracy; Absolute Root Mean Square Error **(Abs RMSE)** in translation, indicating cumulative deviation; Relative Pose Mean Error **(RPME)**, measuring short-term consistency between consecutive poses; and Relative Root Mean Square Error **(Rel RMSE)**, which captures local drift tendencies. These metrics provide a comprehensive view of both the global and local performance of the visual odometry systems under evaluation.

Method (Sequence 00)	APME (↓)	Abs RMSE (↓)	RPME (↓)	Rel RMSE (↓)
SuPerMVO (Ours)	**125.74**	**141.99**	**0.272**	**0.337**
ORB-3	128.90	143.97	0.284	0.351
SuPerMVO-scaled (Evo)	92.41	**108.55**	**0.245**	**0.285**
ORB-3 scaled (Evo)	**87.81**	109.76	0.255	0.2982

Method (Sequence 06)	APME (↓)	Abs RMSE (↓)	RPME (↓)	Rel RMSE (↓)
SuPerMVO (Ours)	**47.687**	**51.337**	**0.305**	**0.333**
ORB-3	51.283	54.641	0.320	0.354
SuPerMVO-scaled (Evo)	**45.455**	**47.866**	**0.250**	**0.307**
ORB-3 scaled (Evo)	49.094	52.336	0.274	0.331

Method (Sequence 10)	APME (↓)	Abs RMSE (↓)	RPME (↓)	Rel RMSE (↓)
SuPerMVO (Ours)	**80.529**	**90.709**	**0.297**	**0.397**
ORB-3	124.031	139.635	0.303	0.408
SuPerMVO-scaled (Evo)	**51.086**	**58.392**	**0.260**	**0.363**
ORB-3 scaled (Evo)	51.865	60.004	0.273	0.381

Method (Sequence 09)	APME (↓)	Abs RMSE (↓)	RPME (↓)	Rel RMSE (↓)
SuPerMVO (Ours)	**36.819**	**42.435**	**0.218**	**0.272**
ORB-3	42.632	48.028	0.231	0.288
SuPerMVO-scaled (Evo)	**34.002**	**39.678**	**0.210**	**0.257**
ORB-3 scaled (Evo)	39.205	43.752	0.227	0.275

Fig. 6. 2D Trajectory Comparison on KITTI Sequences

Across all four evaluated KITTI sequences, SuPerMVO consistently outperforms the classical ORB3 pipeline in both absolute and relative pose estimation metrics. In Sequence 00, SuPerMVO shows a clear improvement over ORB-3 with reductions in both APME (125.74 vs. 128.90) and Abs RMSE (141.99 vs. 143.97). After applying evo-based scale alignment, SuPerMVO-scaled achieves even greater gains with a 26.8% reduction in APME and a 24.6% drop in Abs RMSE compared to its unscaled version, and also performs better than ORB-3-scaled in every metric. In Sequence 06, the differences are even more pronounced: SuPerMVO-scaled achieves the lowest error across all metrics, including the lowest RPME (0.250) and Rel RMSE (0.307), confirming its superior local consistency and scale awareness.

Sequence 09 highlights SuPerMVO's capability in maintaining accurate pose estimates even under challenging scenarios, with an unscaled APME of 36.819 and Abs RMSE of 42.435—substantially better than ORB-3. After scaling, SuPerMVO achieves the best overall performance with the lowest APME (34.002), RPME (0.210), and Rel RMSE (0.257), indicating a highly robust trajectory reconstruction. Similarly, in Sequence 10, SuPerMVO again outperforms ORB-3 by a wide margin. The scaled SuPerMVO results show a 37% reduction in both APME and Abs RMSE over the unscaled version and lead across all metrics against ORB-

3-scaled, with particularly strong improvements in local consistency (RPME = 0.260 vs. 0.273) and scale accuracy (Rel RMSE = 0.363 vs. 0.381). The consistent improvements across both global and local error metrics—especially after scale alignment—highlight SuPerMVO's strong generalization across varying sequences and conditions, making it a compelling and reliable solution for monocular visual odometry.

C. Loop Closure Experiment

We attempted loop closure using a Bag-of-Words (BoW) approach. Although our BoW vocabulary construction and descriptor matching pipeline (using SIFT + KMeans) detected potential loops, the corrections were not consistent across sequences. The failure can be attributed to:

- KITTI's texture-less, repetitive environments.
- Weak distinctiveness of visual words from SIFT features.
- Difficulty in setting a robust threshold for visual similarity.

Ultimately, loop closure was omitted from the final results, though the framework supports its re-introduction.

D. Sliding Window Bundle Adjustment Trial

We also implemented a sliding window bundle adjustment module, but its performance was suboptimal. The method required abundant and reliable feature correspondences, which were sparse in our sequences. Moreover, planar road environments limited landmark diversity. Hence, we opted for GTSAM for pose refinement.

V. CONCLUSION

In this work, we proposed SuPerMVO, a monocular visual odometry pipeline combining deep learning-based feature extraction with classical optimization techniques. Our integration of SuperPoint and SuperGlue enabled robust feature matching even in low-texture scenes, and GTSAM effectively refined pose estimates via factor graph optimization. Finally, evo allowed us to recover absolute scale and produce metric-consistent trajectories. Experimental results on the KITTI dataset demonstrate that SuPerMVO achieves superior performance in terms of ATE, RPE, and scale drift, compared to traditional ORB approaches. This confirms the viability of learning-based VO frameworks when fused with geometric optimization. A visual demonstration of our SuPerMVO results can be found at: YouTube.

VI. FUTURE WORKS

Future work on this project could proceed in the following directions:

- **Transformer-Based Loop Closure:** Integrate Vision Transformers for semantic-aware loop detection to handle perceptual aliasing and dynamic environments.
- **Multi-Sensor Fusion:** Fuse IMU or stereo data with visual input for better scale estimation and robustness under motion blur.
- **Real-Time Deployment:** Optimize the model for embedded platforms to enable real-time SLAM on resource-constrained systems.
- **Full SLAM Integration:** Extend SuPerMVO into a state-of-the-art SLAM pipeline with global mapping and loop closure correction.

VII. ACKNOWLEDGMENT

We would like to sincerely thank Prof. Maani Ghaffari and Dr. Minghan Zhu for their invaluable guidance, insightful lectures, and continuous support throughout the course of this project. Their expertise in robotics and perception played a pivotal role in shaping our understanding and approach. We also extend our gratitude to our GSI, Junzhe Wu, for his timely feedback, debugging help, and clarifications during office hours, which greatly aided the implementation and refinement of our SuPerMVO pipeline. Lastly, we acknowledge the contributions of the open-source community behind SuperPoint, SuperGlue, GTSAM, and evo, whose tools formed the backbone of our development and experimentation efforts.

REFERENCES

[1] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. "SURF: Speeded Up Robust Features". In: *Computer Vision – ECCV 2006*. Ed. by Aleš Leonardis, Horst Bischof, and Axel Pinz. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 404–417.

[2] Michael Calonder et al. "BRIEF: Computing a Local Binary Descriptor Very Fast". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.7 (2012), pp. 1281–1298. DOI: 10.1109/TPAMI.2011.222.

[3] Ronald Clark et al. "VINet: Visual-Inertial Odometry as a Sequence-to-Sequence Learning Problem". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. 1. 2017. DOI: 10.1609/aaai.v31i1.11215. URL: https://ojs.aaai.org/index.php/AAAI/article/view/11215.

[4] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. "SuperPoint: Self-Supervised Interest Point Detection and Description". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2018, pp. 337–33712. DOI: 10.1109/CVPRW.2018.00060.

[5] Andreas Geiger, Philip Lenz, and Raquel Urtasun. "Are we ready for autonomous driving? The KITTI vision benchmark suite". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. 2012, pp. 3354–3361. DOI: 10.1109/CVPR.2012.6248074.

[6] Ruihao Li et al. “UnDeepVO: Monocular Visual Odometry Through Unsupervised Deep Learning”. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 2018, pp. 7286–7291. DOI: 10.1109/ICRA.2018.8461251.

[7] HC Longuet-Higgins. “A computer algorithm for reconstructing a scene from two projections”. In: *Nature* 293.5828 (1981), pp. 133–135.

[8] David G Lowe. “Object recognition from local scale-invariant features”. In: *Proceedings of the seventh IEEE international conference on computer vision*. Vol. 2. Ieee. 1999, pp. 1150–1157.

[9] Raúl Mur-Artal, J. M. M. Montiel, and Juan D. Tardós. “ORB-SLAM: A Versatile and Accurate Monocular SLAM System”. In: *IEEE Transactions on Robotics* 31.5 (2015), pp. 1147–1163. DOI: 10.1109/TRO.2015.2463671.

[10] Sanath Narayan et al. “D2-Net: Weakly-Supervised Action Localization via Discriminative Embeddings and Denoised Activations”. In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 13588–13597. DOI: 10.1109/ICCV48922.2021.01335.

[11] Jerome Revaud et al. “R2D2: repeatable and reliable detector and descriptor”. In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019.

[12] E. Rosten and T. Drummond. “Fusing points and lines for high performance tracking”. In: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Vol. 2. 2005, 1508–1515 Vol. 2. DOI: 10.1109/ICCV.2005.104.

[13] Ethan Rublee et al. “ORB: An efficient alternative to SIFT or SURF”. In: *2011 International Conference on Computer Vision*. 2011, pp. 2564–2571. DOI: 10.1109/ICCV.2011.6126544.

[14] Paul-Edouard Sarlin et al. “Superglue: Learning feature matching with graph neural networks”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 4938–4947.

[15] Benjamin Ummenhofer et al. “DeMoN: Depth and Motion Network for Learning Monocular Stereo”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. July 2017.

[16] Sen Wang et al. “DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks”. In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. 2017, pp. 2043–2050. DOI: 10.1109/ICRA.2017.7989236.

[17] Zihan Yu et al. “LF-Net: A Learning-Based Frenet Planning Approach for Urban Autonomous Driving”. In: *IEEE Transactions on Intelligent Vehicles* 9.1 (2024), pp. 1175–1188. DOI: 10.1109/TIV.2023.3332885.

[18] Huangying Zhan et al. *DF-VO: What Should Be Learnt for Visual Odometry?* 2021. arXiv: 2103.00933 [cs.CV]. URL: https://arxiv.org/abs/2103.00933.