A Visual SLAM Algorithm for Indoor Dynamic Scenes Based on Semantic Feature Screening

Yao Wang* 1323425796@QQ.COM
School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan 411201, China

Changzhong Pan
School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan 411201, China

Hao Huang
School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan 411201, China

Editors: Nianyin Zeng, Ram Bilas Pachori and Dongshu Wang

Abstract

This study innovatively proposes a dynamic scene SLAM al- gorithm that integrates semantic feature filtering mechanism to address the problems of large positioning deviation and dense map artifacts in visual SLAM systems in indoor dynamic environments. In the track- ing module of the ORB-SLAM3 framework, this algorithm introduces a lightweight YOLOv11-seg neural network for scene semantic analysis and target area calibration. Through semantic information, depth data, and geometric relationships, feature point motion state discrimination is achieved, and a high-precision dynamic feature filtering algorithm is developed. To verify the performance of the algorithm, benchmark tests were conducted on the TUM dataset of the Technical University of Munich. Comparative experimental data showed that in the high dynamic conditions of the TUM testing scenario, this scheme achieved significant improvement in trajectory tracking accuracy, and its positioning per- formance was significantly better than the current mainstream dynamic SLAM technology.

Keywords: Complex dynamic environment, semantic segmentation, geometric constraints, SLAM

1. Introduction

SLAM technology serves as the core solution for autonomous navigation, enabling unmanned aerial vehicles, mobile robots, and autonomous cars to simultaneously determine their position and map unknown surroundings.

In indoor scenes, objects are usually divided into dynamic objects, semi static objects, and static objects. Dynamic objects are objects that move in real-time or briefly stay, such as humans and animals; A semi static object is an object that remains stationary within the keyframe interval of the SLAM system but may move due to external forces, such as a book or a pushed chair; Static objects, such as tables or walls, are objects whose positions remain essentially unchanged during the operation of the SLAM system. Current visual SLAM methodologies, designed primarily for static environments, exhibit notable performance deteri- oration when applied to scenes containing dynamic objects. Such environmental variability leads to increased pose estimation inaccuracies and mapping artifacts. This paper presents methodological improvements specifically targeting complex dynamic scenarios, aiming to achieve more reliable trajectory estimation. Our core contributions involve:

- Lightweight YOLOv11-seg network, then use the lightweight network to de- tect and output semantic labels of objects, and perform prior dynamic point filtering on dynamic objects.
- Combining depth geometric constraints for secondary filtering of semi static objects to reduce mapping errors in dynamic environments.
- The experiment was conducted on the TUM dataset, and compared with mainstream dynamic SLAM algorithms, it shows that this method performs better in most scenarios.

2. Related Work

In complex dynamic environment, dynamic point filtering mainly depends on target detection and semantic segmentation technology. Xiao et al. (2019) developed a Dynamic SLAM framework incorporating SSD-based object detection for dynamic feature removal, while enhancing system robustness through a novel detection compensation mechanism. Long et al. (2023) improved BA by combining sparse and dense features, and completed dynamic object segmentation, camera tracking and map construction at the same time, which is suitable for the scene where dynamic objects block static background in a large area. SG-SLAM (Cheng et al., 2023) uses NCNN network to obtain two-dimensional semantic information, and combines epipolar constraints to design a fast dynamic feature elimination strategy, which improves the accuracy and robustness in dynamic scenes, and builds a three-dimensional semantic map. Amos SLAM (Zhuang et al., 2024) proposes a two-stage anti dynamic scene method, which first removes prior dynamic areas through instance segmentation, and then refines and eliminates dynamic areas. Nevertheless, existing SLAM systems exhibit limitations in directional segmentation of dynamic objects, leading to compromised accuracy and limited generalizability. Additionally, excessive feature point elimination often occurs during the filtering process.

3. System Description

This study presents a robust dense SLAM system specifically designed for challenging dynamic scenarios (Figure 1). Building upon the ORB-SLAM3 (Campos et al., 2021) architecture, our framework implements a parallel processing pipeline comprising three core modules: real-time tracking, local map optimization, and loop closure detection. Within the tracking module, the system takes RGB and depth images as input, uses lightweight YOLOv11-seg for semantic segmentation, prior filters dynamic points, and combines depth information and polar geometric constraints for secondary filtering. Finally, static points are used to complete pose estimation. The local mapping thread is responsible for processing key frames, optimizing map points and removing redundant frames. The loopback detection thread is used to correct the cumulative error and complete loopback detection and map merging.

3.1. Lightweight of Semantic Segmentation Network

To enhance the operational efficiency of ORB-SLAM3 in dynamic environments, this paper designs a lightweight semantic segmentation network EfficientNet- YOLOv11-seg. As shown in Figure 2. By replacing the backbone network of YOLOv11-seg, the computing load of the network is reduced and the computing speed of the whole network is improved.

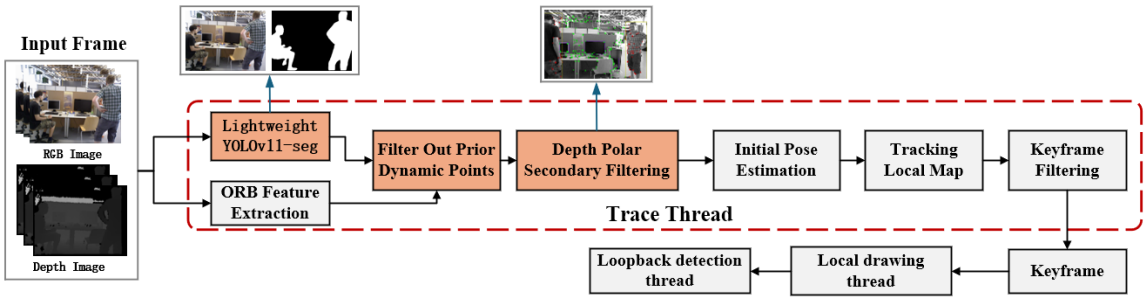


Figure 1: System framework.

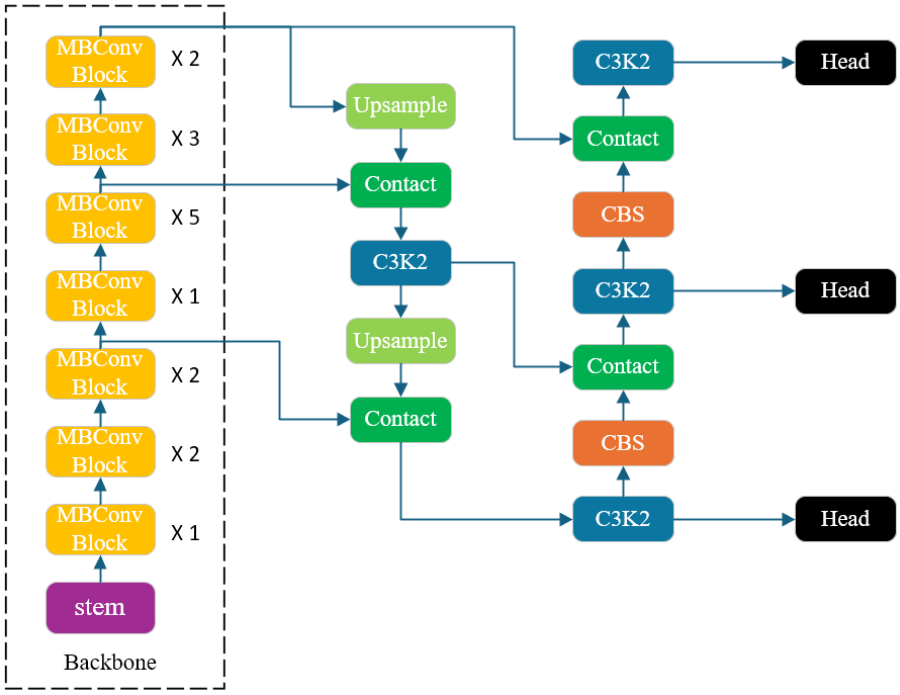


Figure 2: EfficientNet-YOLOv11-seg's network framework.

EfficientNet (Tan and Le, 2019) is an efficient convolutional neural network architecture. Its core idea is to achieve the best balance between model accuracy and computational efficiency through compound scaling strategy and depth separable convolution. EfficientNet proposes to restrict the depth d, width ω and resolution r of the input image of the network in order to achieve higher performance when resources are limited, as shown in the following formula

$$d = \alpha^{\phi}, \omega = \beta^{\phi}, r = \gamma^{\phi} \tag{1}$$

Constraints:

$$\alpha \cdot \beta^2 \cdot \gamma^2 \approx 2(\alpha \geq 1, \beta \geq 1, \gamma \geq 1) \tag{2}$$

Among them α, β, γ are scaling factors for depth, width, and resolution, respectively, and ϕ are global scaling factors.

The computational and parameter complexities of the baseline model are denoted as F_b and P_b respectively, while those of the scaled model are represented by F_s and P_s:

$$\begin{aligned} F_s &\approx \alpha \cdot \beta^2 \cdot \gamma^2 \cdot F_b \\ P_s &\approx \alpha \cdot \beta^2 \cdot P_b \end{aligned} \tag{3}$$

The composite scaling strategy improves model performance while increasing computational and parameter complexity. However, compared with traditional models ResNet and VGG, EfficientNet can achieve higher accuracy under the same computational load.

3.2. Dynamic Point Filtering Based on Semantic and Geometric Constraints

Dynamic Object Prior Filtering. For indoor environmental applications, in order to eliminate interference caused by moving objects, this paper takes people as a priori dynamic target, adopting a lightweight EfficientNet YOLOv11 seg network for robust detection and segmentation of dynamic entities in complex scenes, and sets the semantic segmentation network to output the semantic label $L(p)$ of each pixel, The variable p denotes the spatial coordinates of a pixel within the image plane. The mask $M(p)$ of dynamic objects can be expressed as:

$$M(p) = \begin{cases} 1, L(p) \in \text{ Prior dynamic target} \\ 0, \text{ other} \end{cases} \tag{4}$$

According to the semantic segmentation of the network mask information, after removing a priori dynamic point with a mask value of 1.But semi static objects such as books or chairs are preserved, To address the inherent limitations of semantic segmentation in distinguishing dynamic and static features in semi static objects, depth perception and epipolar geometry constraints are required to enhance feature filtering.

Secondary Filtering of Semi Static Objects Since semi static objects may move or rest, it is necessary to judge their motion state in combination with depth polar geometric constraints. The polar geometric constraint analysis is as follows, as shown in Figure 3, for two consecutive frames, represented as I_1 and I_2 respectively, P point is a point in space, and P point moves to P' point in the dynamic scene. Let O_1 and O_2 denote the optical centers of the dual-camera system, while P_1 and P_2 represent the projected feature points of spatial point P in consecutive frames. The coplanar configuration formed by points O_1, O_2 and P defines the polar plane in vision geometry. The line O_1O_2 is the baseline, and the intersection points of the baseline and the image plane e1 and e2 are respectively I_1 and I_2, which are called poles. the intersecting lines L_1 and L_2 are called polar lines. Since feature point P_2 lies on the epipolar line L_2 by geometric constraint, According to the basic matrix F (Ranftl and Koltun, 2018), the expression of the polar line constraint is obtained:

$$P_2^T L_2 = P_2^T F P_1 = 0 \tag{5}$$

Define the offset distance E_i as the perpendicular distance between feature point $P_i (i = 1, 2, 3)$ and its corresponding epipolar line, computed using the point-to-line distance formula:

$$E_i = \frac{\left| P_i^T F P_1 \right|}{\sqrt{X^2 + Y^2}} \tag{6}$$

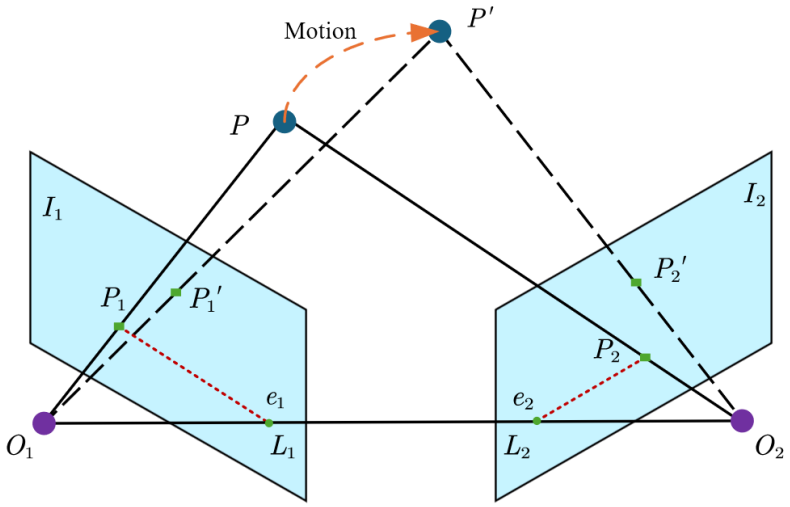


Figure 3: Schematic diagram of polar geometry principle.

Let d_i and d_{i+1} represent the depth for temporally matched feature points in consecutive frames,then the depth change value D_i. Because the depth value is affected by the scale, D_i and E_i are normalized:

$$D_i' = \left| \frac{d_{i+1} - d_i}{\max(d_i, d_{i+1})} \right|, E_i' = \frac{E_i}{\max(E_i)} \tag{7}$$

Set the weights of the two to w_D and w_E to meet $w_D + w_E = 1$, and adjust the weights according to the needs of specific tasks. For characteristic point P_i, calculate the dynamic point score:

$$S(i) = \omega_D \cdot D_i' + \omega_E \cdot E_i' \tag{8}$$

Establish dynamic feature criteria: when the motion evaluation value $S(i)$ of a feature point exceeds the preset threshold T, the point is determined to belong to a dynamic point. According to reference (Bescos et al., 2018), for semi static objects, because their motion is slow, the depth value may not change significantly, while polar constraints are more sensitive to geometric inconsistency and can better capture the motion characteristics of semi static objects, so the weight value of both should be set to $w_D < w_E$.

4. Experiments

This algorithm has been systematically experimentally validated on the TUM public dataset, using a comparative analysis method to compare its performance with the ORB-SLAM3 benchmark algorithm and other cutting-edge SLAM schemes. The experimental platform is configured with a hardware environment of Intel i5-12600K processor, and 8GB of DDR4 memory. The software environment is Ubuntu 20.04 LTS operating system.

To evaluate the tracking results, absolute trajectory error (ATE) and relative attitude error (RPE) are used as measurement indicators. Through multiple experimental verifications, the threshold $T = 0.25$ and weight $w_D = 0.3$, $w_E = 0.7$ for geometric constraints are set. As shown in

the figure 4, The experimental results show that the algorithm achieves centimeter level trajectory accuracy and highly matches the real trajectory, which proves that the algorithm can better handle high dynamic scenes.

As shown in Table 1, the experimental results show that in the four dynamic scenario tests of Fr3/walking sequences in the TUM dataset, our system exhibits significant advantages compared to ORB-SLAM3: in the high dynamic walking xyz scenario, RMSE and standard deviation are reduced by 94.6% and 93.6%, respectively; In low dynamic sitting scenarios, RMSE and standard deviation also achieved reductions of 46.1% and 43.1%, respectively. These data fully validate the positioning accuracy advantage of our algorithm in high dynamic environments.

Table 1: Comparison of absolute trajectory error between ORB-SLAM3 and improved algorithm

Sequences	ORBSLAM3				OURS			
	RMSE	Mean	Median	S.D.	RMSE	Mean	Median	S.D.
walking_xyz	0.699	0.561	0.411	0.417	0.020	0.018	0.015	0.010
walking_static	0.359	0.318	0.294	0.166	0.007	0.006	0.005	0.004
walking_rpy	0.586	0.534	0.518	0.239	0.028	0.023	0.019	0.016
walking_half	0.331	0.307	0.294	0.123	0.026	0.021	0.018	0.016

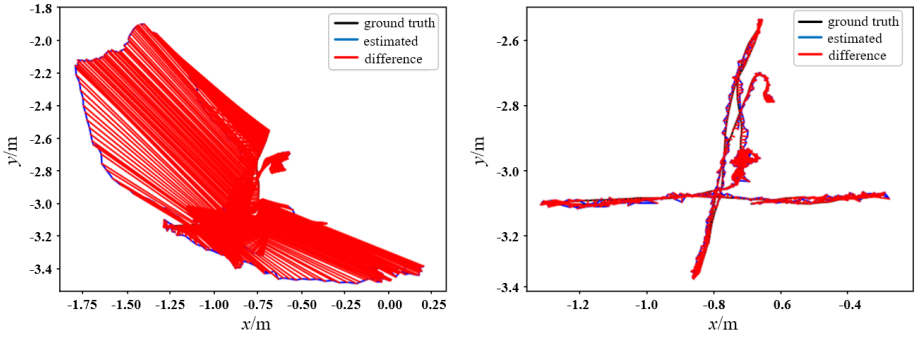
The algorithm proposed in this article was benchmarked against three advanced visual SLAM systems: Det-SLAM (Schütz et al., 2022), RDS-SLAM (Eslamian and Ahmadzadeh, 2022) and LC-SLAM (Liu and Miura, 2021), and the experimental results in the original paper are used. The experimental data (Tables 2 and 3) shows that in the comparison of RMSE and S.D. indicators of absolute and relative trajectory errors, our algorithm performs the best in the dynamic sequences of walking-xyz and walkingrpy. its adaptability and improvement effect are better than other methods, and it runs more stably in dynamic environment.

Table 2: Comparison of RMSE values of different algorithms

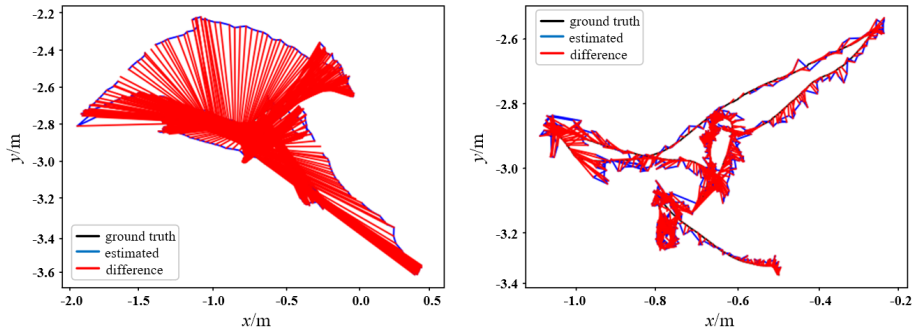
Sequences	ATE				RPE			
	Det-SLAM	RDS-SLAM	LC-SLAM	OURS	Det-SLAM	RDS-SLAM	LC-SLAM	OURS
walking_xyz	0.0553	0.021	0.024	0.0207	0.0653	0.026	0.036	0.0119
walking_static	0.0049	0.081	0.016	0.0078	0.0100	0.022	0.016	0.0061
walking_rpy	0.0386	0.146	0.055	0.0289	0.0680	0.024	0.086	0.0174
walking_half	0.0045	0.025	0.041	0.0269	0.0167	0.027	0.051	0.0145

Table 3: Comparison of S.D. values of different algorithms

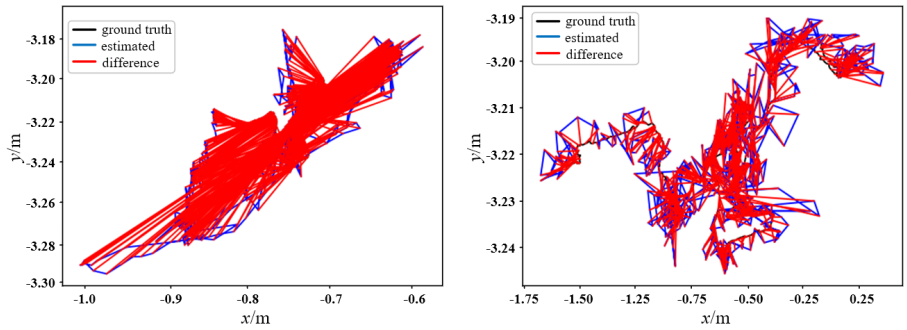
Sequences	ATE				RPE			
	Det-SLAM	RDS-SLAM	LC-SLAM	OURS	Det-SLAM	RDS-SLAM	LC-SLAM	OURS
walking_xyz	0.0553	0.021	0.024	0.0207	0.0653	0.026	0.036	0.0119
walking_static	0.004	0.081	0.016	0.007	0.010	0.022	0.016	0.006
walking_rpy	0.0386	0.146	0.055	0.0289	0.0680	0.024	0.086	0.0174
walking_half	0.0045	0.025	0.041	0.0269	0.0167	0.027	0.051	0.0145



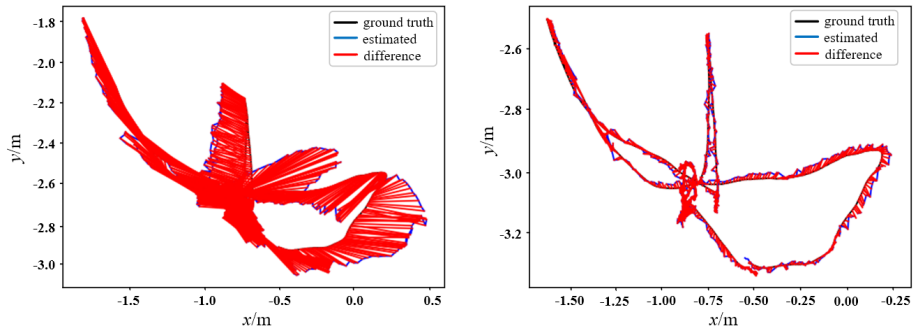
(*a*) fr3_walking_xyz



(*b*) fr3_walking_rpy



(*c*) fr3_walking_static



(*d*) fr3_walking_halfsphere

Figure 4: ORB-SLAM3 (the first column) and the algorithm in this paper (the second column) in the high dynamic scene estimated trajectory error diagram.

5. Conclusion

This article proposes a new visual SLAM method that combines a lightweight YOLOv11-seg semantic segmentation network, depth information, and geometric constraints based on ORB-SLAM3 to remove dynamic feature points in dynamic scenes, thereby achieving inter frame matching and pose estimation of static object feature points. Experimental analysis shows that the algorithm proposed in this article has good positioning accuracy. However, when the proportion of dynamic objects in the visual scene exceeds 60%, the system will experience a decrease in localization performance. The future direction of work is to use tracking of moving objects to achieve attitude estimation and solve this problem.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant No. 621731138), the Hunan Provincial Innovation Foundation for Post- graduate (Grant No. CX20231041).

References

Berta Bescos, José M. Fácil, Javier Civera, and José Neira. Dynaslam: Tracking, mapping, and inpainting in dynamic scenes. *IEEE Robotics and Automation Letters*, 3(4):4076–4083, 2018. doi: 10.1109/LRA.2018.2860039.

Carlos Campos, Richard Elvira, Juan J. Gómez Rodríguez, and et al. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Transactions on Robotics*, 37 (6):1874–1890, 2021. doi: 10.1109/TRO.2021.3075644.

Shuhong Cheng, Changhe Sun, Shijun Zhang, and Dianfan Zhang. Sg-slam: A real-time rgb-d visual slam toward dynamic scenes with semantic and geometric information. *IEEE Transactions on Instrumentation and Measurement*, 72:1–12, 2023. doi: 10.1109/TIM.2022.3228006.

Ali Eslamian and Mohammad Reza Ahmadzadeh. Det-slam: A semantic visual slam for highly dynamic scenes using detectron2. In *2022 8th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*, page 1–5. IEEE, December 2022. doi: 10.1109/icspis56952.2022.10043931.

Yubao Liu and Jun Miura. Rds-slam: Real-time dynamic slam using semantic segmentation methods. *IEEE Access*, 9:23772–23785, 2021. doi: 10.1109/ACCESS.2021.3050617.

Ran Long, Christian Rauch, Vladimir Ivan, and et al. Rgb-d-inertial slam in indoor dynamic environments with long-term large occlusion, 2023.

René Ranftl and Vladlen Koltun. Deep fundamental matrix estimation. In *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part I*, page 292–309. Springer-Verlag, 2018. doi: 10.1007/978-3-030-01246-5_18.

Markus Schütz, Bernhard Kerbl, and Michael Wimmer. Software rasterization of 2 billion points in real time. *Proc. ACM Comput. Graph. Interact. Tech.*, 5(3), July 2022. doi: 10.1145/3543863.

Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946, 2019.

Linhui Xiao, Jinge Wang, Xiaosong Qiu, and et al. Dynamic-slam: Semantic monocular visual localization and mapping based on deep learning in dynamic environment. *Robotics and Autonomous Systems*, 117:1–16, 2019. doi: https://doi.org/10.1016/j.robot.2019.03.012.

Yaoming Zhuang, Pengrun Jia, Zheng Liu, and et al. Amos-slam: An anti-dynamics two-stage rgb-d slam approach. *IEEE Transactions on Instrumentation and Measurement*, 73:1–10, 2024. doi: 10.1109/TIM.2023.3332395.