frontiers | Frontiers in Signal Processing

TYPE Original Research
PUBLISHED 09 October 2024
DOI 10.3389/frsip.2024.1417363

OPEN ACCESS

EDITED BY
Raouf Hamzaoui,
De Montfort University, United Kingdom

REVIEWED BY
Serhan Cosar,
The Open University, United Kingdom
Gaochang Wu,
Northeastern University, China

*CORRESPONDENCE
Cong Yang,
cong.yang@suda.edu.cn

RECEIVED 14 April 2024
ACCEPTED 26 July 2024
PUBLISHED 09 October 2024

CITATION
Luan S, Yang C, Qin X, Chen D and Sui W (2024) Towards robust visual odometry by motion blur recovery.
Front. Sig. Proc. 4:1417363.
doi: 10.3389/frsip.2024.1417363

Towards robust visual odometry by motion blur recovery

Simin Luan[1], Cong Yang[1]*, Xue Qin[2], Dongfeng Chen[1] and Wei Sui[3]

[1]Ecology and Innovation Center of Intelligent Driving (BeeLab), Soochow University, Suzhou, China, [2]Harbin Institute of Technology, Harbin, China, [3]Horizon Robotics, Beijing, China

Introduction: Motion blur, primarily caused by rapid camera movements, significantly challenges the robustness of feature point tracking in visual odometry (VO).

Methods: This paper introduces a robust and efficient approach for motion blur detection and recovery in blur-prone environments (e.g., with rapid movements and uneven terrains). Notably, the Inertial Measurement Unit (IMU) is utilized for motion blur detection, followed by a blur selection and restoration strategy within the motion frame sequence. It marks a substantial improvement over traditional visual methods (typically slow and less effective, falling short in meeting VO's realtime performance demands). To address the scarcity of datasets catering to the image blurring challenge in VO, we also present the BlurVO dataset. This publicly available dataset is richly annotated and encompasses diverse blurred scenes, providing an ideal environment for motion blur evaluation.

KEYWORDS

SLAM (simultaneous localization and mapping), deblur, multimodal fusion, motion blur, IMU

1 Introduction

Visual Odometry (VO) is pivotal in visual synchronized localization and mapping (V-SLAM), robotic autonomous navigation, virtual/augmented/mixed reality, etc. (Nistér et al., 2004). However, motion blur, often occurring in dynamic environments such as when a wheeled robot traverses bumpy roads, poses significant challenges to VO by impairing the accuracy of feature point matching between frames, as presented in Figure 1. The primary challenge in VO under such conditions is the reduction of tracking failures due to motion blur. Moreover, there is a notable scarcity of public datasets explicitly representing motion blur scenarios in VO.

To enhance VO robustness, we introduce an efficient approach for filtering and restoring blurred frames in blur-prone environments, such as rapid movements and uneven terrains. Particularly, to assess motion blur in images, our approach leverages IMU (Inertial Measurement Unit) data to provide a direct and efficient means of blur estimation. To further restore the blurry sequence, an empirical debluring strategy is proposed based on adjacent frame filtering and fusion. It should be noted that involving IMU in VO is not unprecedented. For instance, VI-DSO (Von Stumberg et al., 2018) and ORB-SLAM3 Monocular Inertial (Campos et al., 2021) use IMU data for pose estimation. However, these methods fall short of reducing tracking failures in severe motion blur scenarios. Our approach, in contrast, effectively utilizes IMU data to detect and restore motion blur (aka. blurry sequences), thus improving the initialization speed and overall robustness of VO. Notably, our method is compatible with various IMU sensors, enhancing

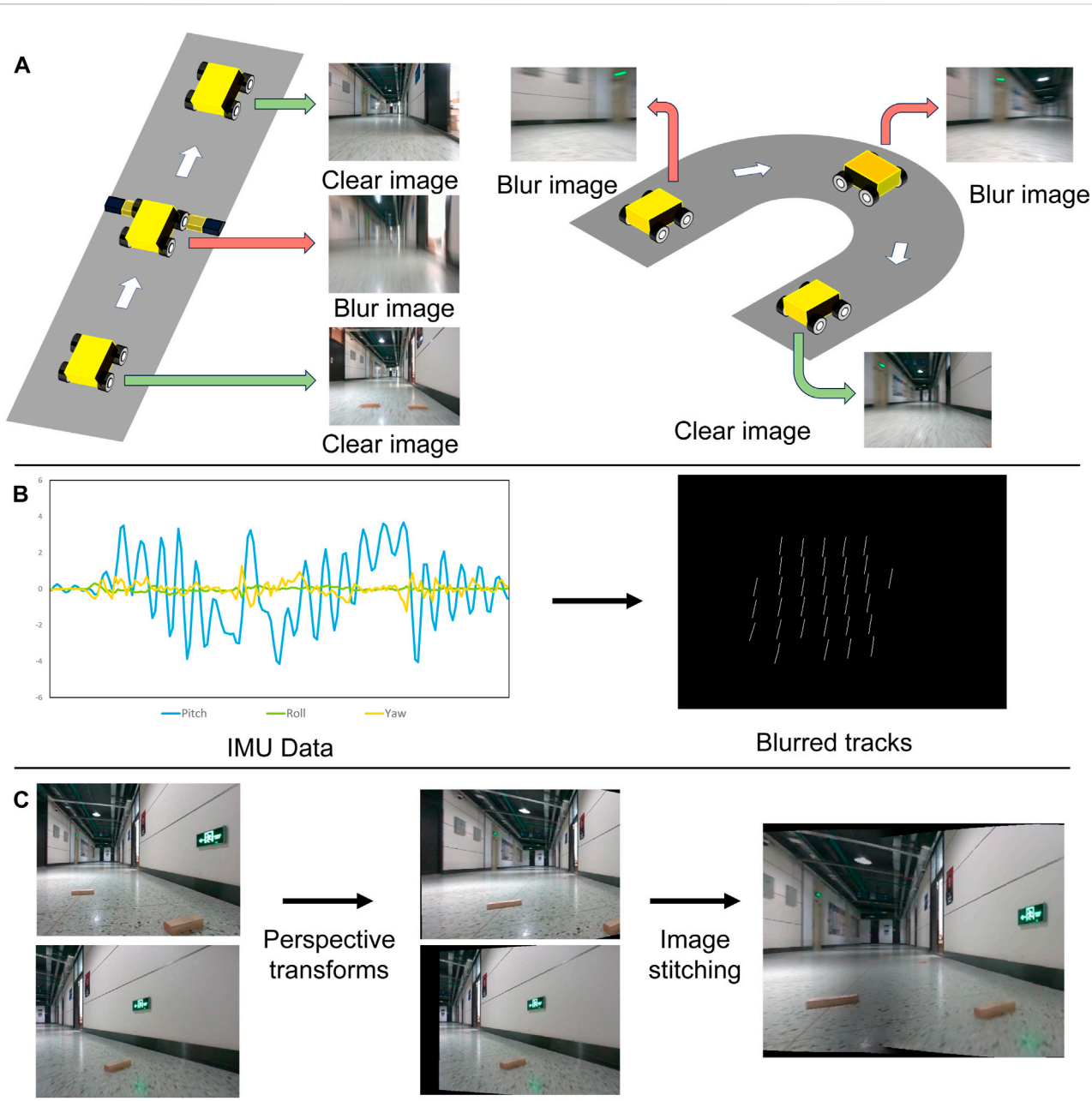

FIGURE 1
Illustration of blur occurring, detecting, and restoring: **(A)** Motion blur often occurs on obstacles and cornering. **(B)** The blur trajectories are derived from the IMU data, and the fuzzy images are screened according to the trajectory length. **(C)** The image perspective is transformed to the same viewing angle and stitched for blurry restoration.

its applicability. For the lack of dataset challenge, we introduce a new publicly available dataset, BlurVO, for indoor and outdoor motion blur evaluations in VO. Comprising 12 sequences from various real-world environments, BlurVO is equipped with data from pre-calibrated cameras and IMUs, fostering the development of more robust algorithms for VO in blur-prone scenarios.

Our main contributions are as follows: (1) We introduce a simple yet efficient approach for motion blur detection and restoration based on IMU. It marks a substantial improvement over traditional visual methods in terms of real-time performance, high-accurate blur estimation and recovery, and robustness of VO in challenging environments. (2) We introduce a new dataset, BlurVO, which contains a rich collection of blurry clips (both frames and corresponding IMU data) from various environments to promote further research tasks toward the robustness of VO.

2 Related work

We concisely survey existing VO and blur-related methods, including blur detection and datasets. For more detailed treatments of these topics, the compilation by He (He et al., 2020), and Vankawala (Vankawala et al., 2015) offer a sufficiently good review.

2.1 Visual odometry

Visual odometry methods can be broadly categorized into classical geometric, deep learning-based, and hybrid methods. Classical geometric methods: such as those by Davison (Davison, 2003) and Nister et al. (Nistér et al., 2004), and popular implementations like ORB-SLAM3 (Campos et al., 2021), rely on feature extraction and matching. Direct methods, including LSD-SLAM (Engel et al., 2014) and DSO (Von Stumberg et al., 2018), optimize camera pose based on photometric consistency. However, these methods often presuppose smooth camera motion and struggle with motion blur. Deep learning-based methods: initiated by Roberts (Roberts et al., 2008), aim for end-to-end pose estimation but face challenges in real-time processing. Hybrid methods: combining classical and deep learning approaches, seek to improve robustness but still grapple with motion blur. Our work diverges by leveraging sharp frames, employing a deblurring network selectively, and enhancing VO robustness in blur-prone environments. Note that VIO (Visual Inertial Odometry) combines image and IMU data, with notable examples like OKVIS (Leutenegger et al., 2013) and VINS-Mono (Qin et al., 2018). These methods achieve high accuracy but often require high-precision IMUs and do not specifically address motion blur. Our approach uniquely utilizes IMU data for detecting and recovering from motion blur, thereby enhancing VO performance with a broader range of IMU sensors.

2.2 Blur detection and restoration

Motion blur usually degrades the edges of objects in an image. Traditional methods for detecting blur usually extract features such as gradient and frequency to describe the changes in edges (Chen et al., 2013). Yi and Eramian (Yi and Eramian, 2016) designed a sharpness metric based on local binary patterns and used it to separate an image's in-focus and out-of-focus areas. Tang et al. (Tang et al., 2017) designed a logarithmic mean spectral residual metric to obtain a rough blur map. Then, they proposed an iterative update mechanism to refine the blur map from coarse to fine based on the intrinsic correlation of similar adjacent image regions. Although traditional methods have achieved great success in blur detection, they are only effective for images with simple structures and are not robust enough for complex scenes. Due to their high-level feature extraction and learning capabilities, deep CNN-based methods have refreshed the records of many computer vision tasks. Purohit et al. (Purohit et al., 2018) proposed to train two sub-networks to learn global context and local features, respectively, and then aggregate the pixel-level probabilities estimated by the two networks and feed them into the MRF-based blur region segmentation framework. Zhao et al. (Zhao et al., 2023) proposed a heterogeneous distillation mechanism to generate blur response maps by combining local feature representation and global content perception. However, these methods are highly complex and challenging to process in real-time. Our method uses IMU data as prior information to calculate the length of each image blur kernel. It uses these lengths to judge whether the image is blurry. This method balances efficiency, robustness, and applicability.

Recent CNN-based deblurring networks (e.g., Kupyn (Kupyn et al., 2019) and Cho (Cho et al., 2021)) focus only on single-frame restoration and have difficulty achieving real-time performance for motion blur over frame sequences. Thanks to the prior information (frame category) from the IMU, our proposed strategy can select appropriate operations for repair (deletion, retention, and restoration) for different blurry frames.

2.3 SLAM dataset

In terms of blur-related datasets, as summarized in Table 1. Most existing SLAM datasets focus on autonomous driving (Geiger et al., 2013) (Wen et al., 2020) (Ligocki et al., 2020) or drones (Burri et al., 2016). Some datasets are targeted at ground robots. OpenLORIS (Shi et al., 2020) is collected in indoor environments by a wheeled robot designed for visual SLAM, where LiDAR SLAM is used to generate ground truth. In some cases, LiDAR SLAM can have larger errors than visual SLAM, making the ground truth unreliable. TUM RGBD (Sturm et al., 2012) partially uses robots as an acquisition platform but only contains RGB and depth cameras. Similar datasets include UTIAS MultiRobot (Leung et al., 2011), PanoraMIS (Benseddik et al., 2020), and M2DGR (Yin et al., 2021). Unfortunately, the above datasets do not specifically record scenes of violent robot movements. Even if some have relevant scenes, the number is minimal and insufficient to support dedicated motion blur research. Our BlurVO dataset is designed for comprehensive motion blur evaluation, covering different types and scenes of motion blur, filling a major gap in current research.

3 Our method

This part aims to detect and restore the motion blur sequence that affects VO robustness and V-SLAM operation. As presented in Figure 2, our method comprises two primary components: Motion Blur Detection and Motion Blur Recovery, aiming to detect and correct blurred frames in VO through IMU data analysis.

3.1 Motion blur detection

3.1.1 Blurry Categories

As detailed in Figure 2, we assess the blurry degree of each frame based on the IMU data corresponding to the image sequences. Built on that, the frames are divided into three categories: (1) Removable: Over-blurred frames, will be dropped directly. (2) Sharp: Qualified frames, will be kept without any changes. (3) Repairable: Slight-blurred frames, will be restored. The reason for dividing these categories is that the quality (640*480) and quantity (25 frames/second) of the blurred frames to be repaired are relatively large, which requires a lot of edge computing resources and makes it difficult to achieve real-time results. In addition, the effect of repairing overly blurred frames is limited, and it is impractical to repair every frame in the sequence.

3.1.2 Blurry Detection

In videos recorded by robots, image blur is mainly generated by camera shake. Our IMU-based method calculates the effect of camera shake on the spatial movement distance of image points

TABLE 1 Comparison of VO-related datasets.

Dataset	Environment	Platform	IMU	Motion blur
KITTI (Geiger et al., 2013)	Urban	Car	Yes	No
EUROC (Burri et al., 2016)	Indoors	UAV	Yes	No
UZH-FPV (Delmerico et al., 2019)	In/Outdoors	UAV	Yes	No
TUM VI (Schubert et al., 2018)	In/Outdoors	Hand-Held	Yes	No
NCLT (Carlevaris-Bianco et al., 2016)	In/Outdoors	Ground Robot	Yes	No
OPENLORIS (Shi et al., 2020)	Indoors	Ground Robot	No	No
M2DGR (Yin et al., 2021)	Indoors	Ground Robot	Yes	No
Our dataset	Indoors	Ground Robot	Yes	Yes

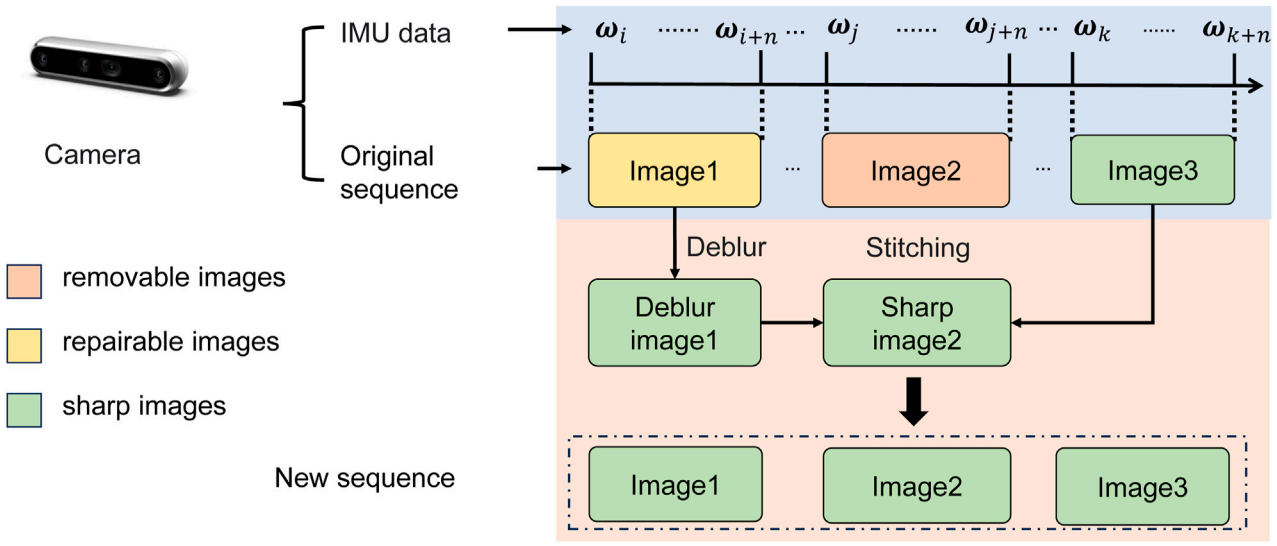

FIGURE 2
Our method consists of two main parts: Motion Blur Detection (top) and Motion Blur Recovery (down).

to determine the degree of image blur. The intuitive understanding is: let A' represent a point on the image, and the movement of A' affected by camera movement (yaw, roll, and pitch) can be routinely collected from the IMU. We call the A' movement trajectory the motion blur trajectory. Then, if the motion blur trajectory length is more significant than 1/50 of the frame width, the frame is roughly classified as "Blurred". Otherwise, we regard it as "Sharp". According to the trajectory deviation in the frame sequence, the "Blurred" frames are subdivided into "Removable" and "Repairable".

In a frame sequence, adjacent blurry frames record the same scene, so the contents of these frames are very similar. It will take much time if all these frames are deblurred using a neural network. So, we selectively process some keyframes through the blur network to improve repair efficiency. These keyframes are marked as "Repairable", and the intermediate frames between these key frames are generated by perspective transformation and image stitching. The blurred frames between these keyframes are considered "Removable" and deleted uniformly. Our method uses IMU data to track the position of point A' on each image and uses position differences to determine frame similarity. We define the similarity criterion based on a displacement distance of less than 50 pixels. Specifically, the first blurry frame after a clear frame is marked as "Repairable". Subsequent "Repairable" frames are identified based on their relative displacement with the frame. If no frame within a range of ten frames has a displacement distance of more than 50 pixels, the 10th frame is designated as "Repairable". This section further details calculating the position of point A' using IMU data and studies the impact of camera translation and rotation on the transformation of A'. In the case of camera translation only, the Equation 1 applies.

$$\begin{aligned} P_{t2} &= P_{t1} + T \\ Z_{t1}A'_{t1} &= KP_{t1} \\ Z_{t2}A'_{t2} &= KP_{t2} \end{aligned} \quad . \tag{1}$$

where P_t is the posture of the camera at time t, A'_t is the pixel position of the projection of point A on the image in space at time t, and Z_t is the depth distance between point A and the camera at time t, T and K are the motion translation and camera intrinsic matrix,

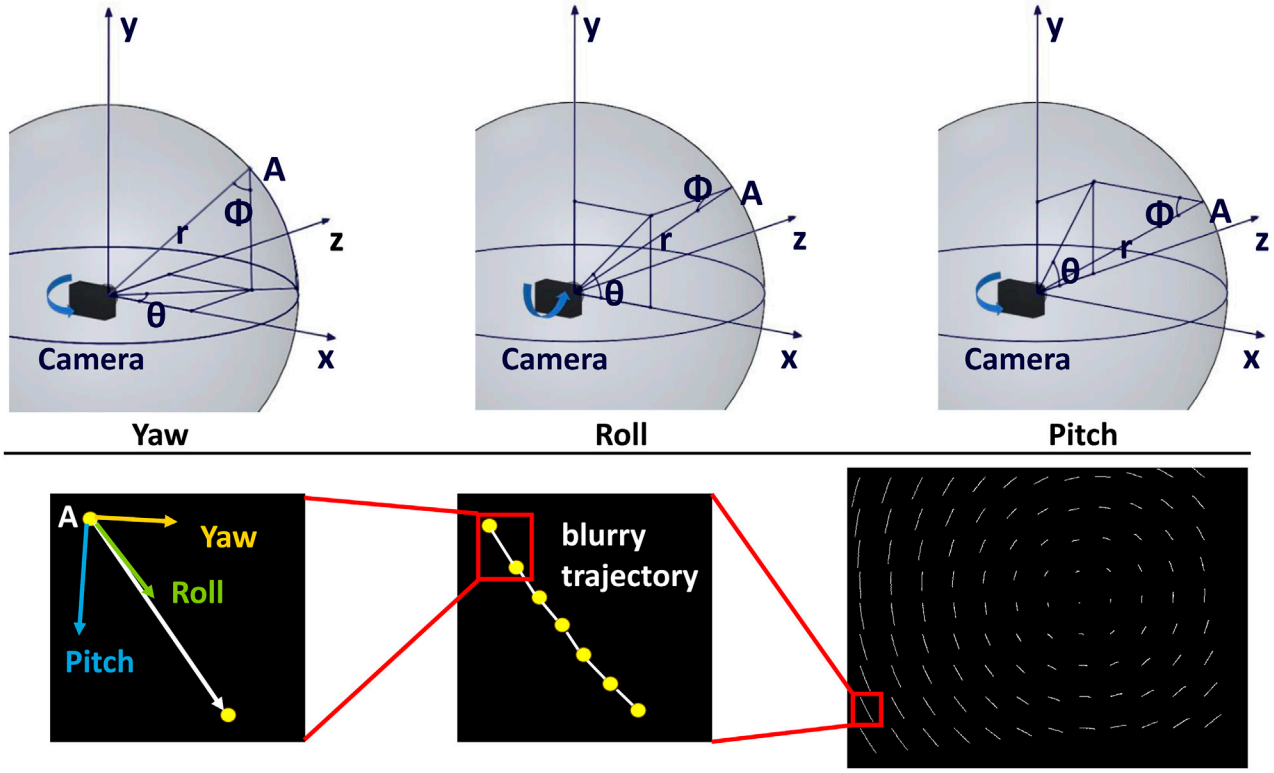

FIGURE 3
Blurry degree estimation based on IMU. Top: *A* shifting in three camera motions estimated from IMU: Yaw, Roll, and Pitch. Bottom: Estimating *A* shifting trajectory (aka. motion blur trajectory) in blurred Yaw, Roll, and Pitch.

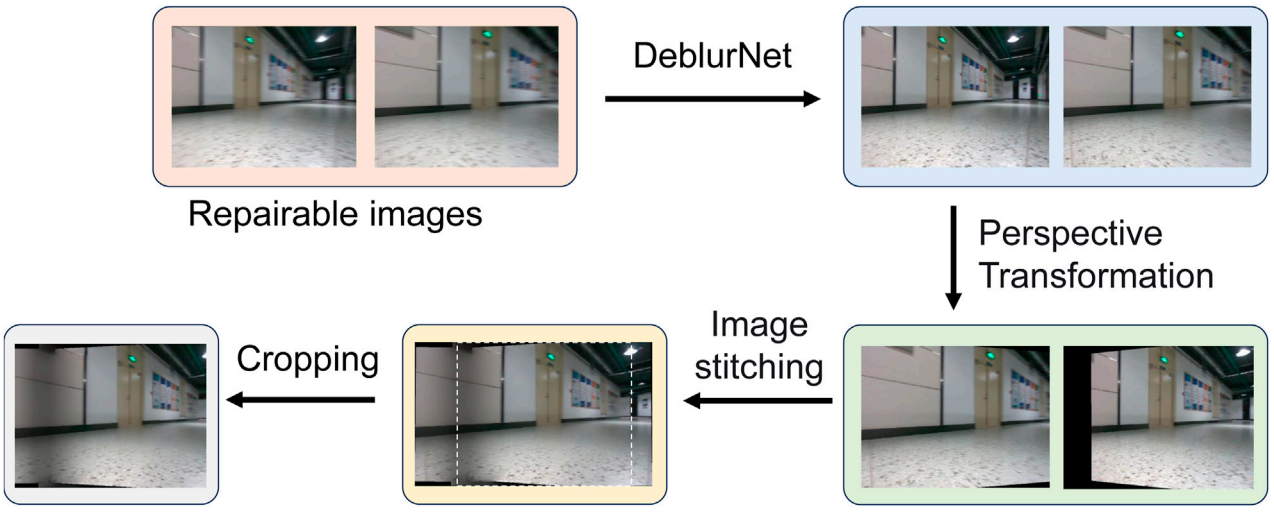

FIGURE 4
Image restoration process. The deblurring network processes the "Repairable" image to generate a clear image, performs perspective transformation, and finally stitches the transformed image and crops the sharp image from it.

respectively, t_1 and t_2 are the time when the camera exposure starts and ends respectively. Since the camera's exposure time is generally negligible (tens of milliseconds), the translational displacement of the camera is minimal during this period. We assume that $Z_{t1} = Z_{t2}$, and calculate the pixel displacement $\|A'_{t2} - A'_{t1}\|$ as KT/Z. This equation shows that as the point A is located at a deeper position (Z becomes larger, K and T remain constant, the effect of camera translation on pixel motion becomes smaller. A real-world analogy

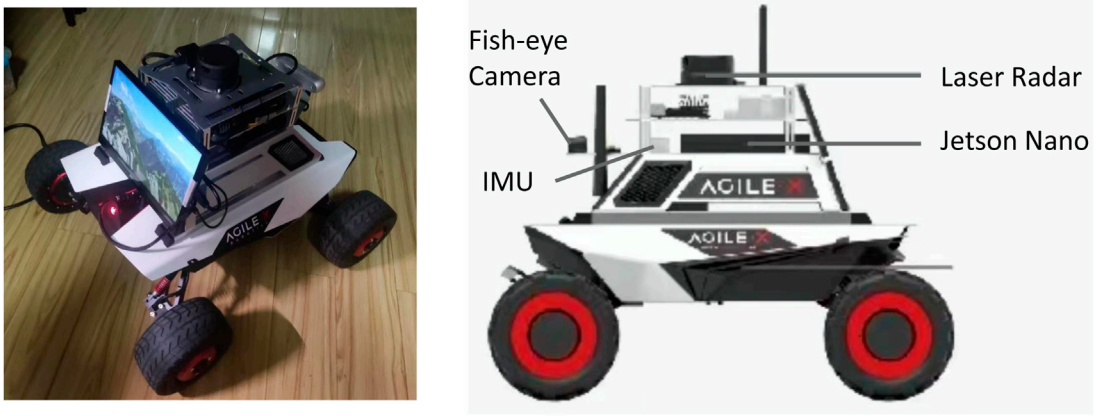

FIGURE 5
Our ground robot for data collection.

is when driving a car and looking at an obstacle on the side of the road - the closer this obstacle is to the vehicle, the faster it moves. Therefore, camera rotation is the main factor that causes blurry movement.

For rotational motion, as detailed in Figure 3, we use a spherical coordinate system to redefine A by parameters (r, θ, ϕ). Then, the projected coordinate (U, V) of A is calculated by Equation 2.

$$\begin{aligned} U &= \frac{f_x X}{Z} = \frac{f_x}{\tan\theta} \\ V &= \frac{f_y Y}{Z} = \frac{f_y}{\sin\theta \tan\phi} \end{aligned}. \tag{2}$$

where (f_x, f_y) is the camera's focal length.

As shown in Figure 3(Bottom), let L denote the long of the blurred track, $\Delta\theta$ and $\Delta\phi$ denote the camera rotation angle, the A shifting (aka. blurry trajectory) is calculated by Equation 3.

$$\begin{aligned} \Delta U &= f_x\left(\frac{1}{\tan\theta} - \frac{1}{\tan(\theta+\Delta\theta)}\right) \\ \Delta V &= \frac{f_y}{\tan\phi}\left(\frac{1}{\sin\theta} - \frac{1}{\sin(\theta+\Delta\theta)}\right) \\ L &= \sum_{i=0}^{n}\left(\sqrt{(\Delta V_i)^2 + (\Delta U_i)^2}\right) \end{aligned}. \tag{3}$$

where n is the number of line segments that make up the blur trajectory and we introduce a threshold λ of L based on the image's resolution. In Section 4, we set $\lambda = \frac{1}{50}h$, h is the image height in pixels.

3.2 Blurry recovery

We select some keyframes ("Repairable" frames) from the blurry frames and use SRNDeblurNet (Tao et al., 2018) to repair these frames. These repaired clear keyframes can be used to generate other intermediate frames. The generation process is shown in Figure 4. First, we perform a perspective transformation on these "Repairable" frames so that two adjacent keyframes can have the same perspective to facilitate image stitching. Then, the stitched images are cropped to generate clear intermediate frames, and the blurry frames originally at these positions in the video are deleted as "Removable" frames.

Here, we detail the restoration and stitching process on "Repairable" frames. SRNDeblurNet (Tao et al., 2018) is constructued by Encoder-decoder ResBlock Network. After deblurring with SRNDeblurNet, we utilize IMU data to determine the position of identical pixels across different frames in the image stitching process. Subsequently, we iteratively calculate the homography matrix to perform a perspective transformation on adjacent frames. To ensure that the restored frames are continuous with the context, we predict the pixel position of the center point corresponding to the frame and use these pixels to determine the cropping position. The perspective-transformed images are then spliced and cropped.

This approach offers two advantages over traditional methods, which rely solely on feature point matching to calculate homography matrices: (1) By basing the process on IMU data, our approach substantially reduces errors caused by inaccuracies in feature point matching during image splicing. (2) Utilizing IMU data also allows us to infer the position of feature points at any given moment between two frames. This capability enables the generation of intermediate frames, resulting in a continuous and accurate frame sequence, something traditional methods cannot achieve.

4 Experiment

In this part, we first introduce the datasets in our experiments and then evaluate our motion blur detection approach. The effectiveness of our motion blur restoration method is assessed with mainstream VIO and VO methods. Limitations of our approach are finally discussed.

4.1 Datasets

Existing datasets are mainly collected for evaluating VO methods, and a dedicated dataset for motion blur scenarios is lacking. While datasets like EuRoC (Burri et al., 2016) contain partially blurred sequences, the variety and annotation of blur types are insufficient for comprehensive testing. Thus, we present BlurVO, a dataset featuring 12 motion blur sequences with four

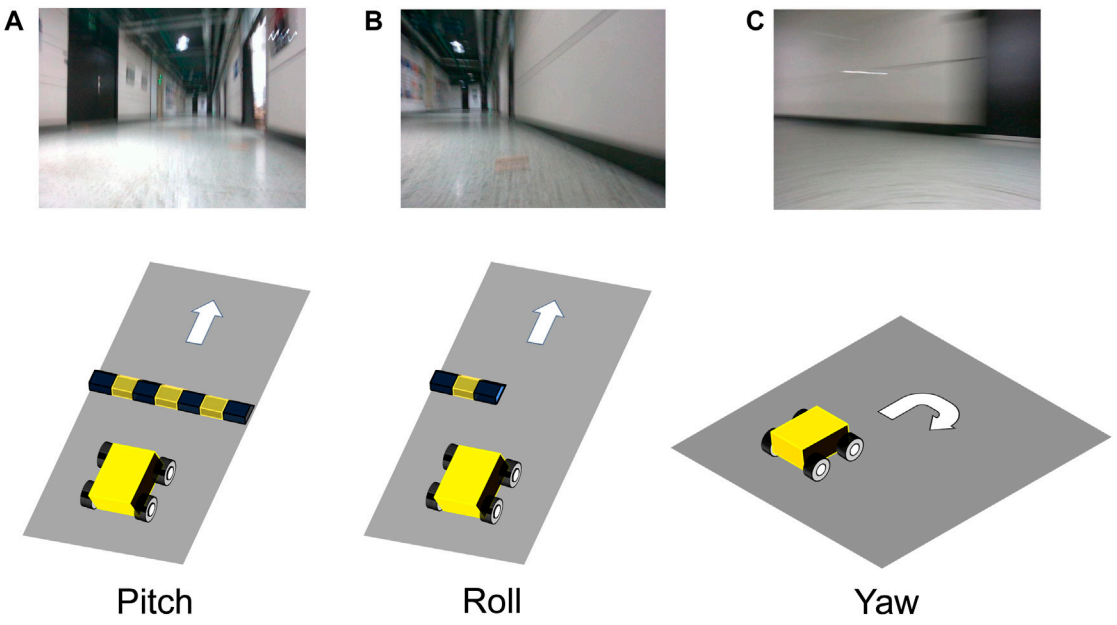

FIGURE 6
(A) When both robot wheels pass an obstacle simultaneously, pitch blur will occur. **(B)** A roll blur will occur when only one robot wheel passes an obstacle. **(C)** When the robot turns quickly, a yaw blur will occur.

TABLE 2 Statistics of datasets used in our experiment.

	EuRoC (Burri et al., 2016)	BlurVO
Total Scenes	2	**5**
Motion Blur Sequences	6	**12**
Motion Blur Types	Shake	**Pitch**, **Roll** **Yaw**, Shake
Camera Resolution	752*480	640*480
Frame Rate (FPS)	20	**25**
IMU Sampling Rate (Hz)	200	200
IMU Data	Acceleration Angular Velocity	Acceleration Angular Velocity

distinct types (jitter, pitch, roll, and yaw) at varying levels. The extrinsic parameters between the IMU and the camera were calibrated using the Kalibr tool (Furgale et al., 2013). The positions of all the installed equipment are shown in the Figure 5. A stereo camera and a fisheye camera are used. We use lidar to scan the surrounding environment to obtain infrared images, and a consumer-grade IMU is also installed. We use a laser scanner to track the robot.

To obtain the ground truth of the trajectory, we placed some small wooden bars as obstacles in the room for data collection. As shown in the Figure 6, we manipulated the robot through these obstacles to generate a dataset containing different motion blurs. It should be noted that it is difficult for a wheeled robot to simulate a scene with only jitter blur (i.e., the camera only moves without rotation), so we did not deliberately collect jitter blur scenes. We generate different scene sequences by permuting and combining various types of blur. The comparison between BlurVO and EUROC (Burri et al., 2016) datasets is shown in Table 2.

4.2 Metrics

We benchmarked our method against ORB-SLAM3 and DSO. For evaluation, we used absolute trajectory error (ATE) and frame loss percentage (FD) to measure accuracy and robustness. It takes 1 s for the SRN network to process a 640*480 blurry image. At the same time, our method cleverly combines traditional techniques and neural network methods, using SRN to process a small number of keyframes (Repairable frames) to guide contextual image repair. So, on average, we only need to use the SRN network to repair one of the ten blurry images, so our method dramatically improves the repair efficiency, with an average processing speed of 200 ms. It should be noted that the time here is only calculated as the time to process the blurry image. The average speed of processing the entire sequence on Jetson Nano can reach 30 ms, which fully meets the robot's real-time requirements.

4.3 Blur detection comparison

Since it is difficult to find a suitable indicator to judge the blurriness of an image, in this paper, we define the frames that make SLAM tracking fail as blurry images and manually annotate the blurry photos in three sequences. Using these annotations, we tested the recall and accuracy of different blur detection methods. The results are shown in Table 3. Our method can identify blurry frames

TABLE 3 Results of different methods for blur detection on BlurVO.

Dataset:BlurVO	Pitch			Roll				Yaw		
	Pitch_1	Pitch_2	Pitch_3	Roll_1	Roll_2	Roll_3	Roll_4	Yaw_1	Yaw_2	Yaw_3
ORB-SLAM3 Monocular (Campos et al., 2021)	-	29.5%	**0%**	69.7%	55.0%	**0%**	-	47.5%	**0%**	88.0%
DSO (Engel et al., 2017)	-	90.5%	-	**10.6%**	23.9%	17.0%	**30.0%**	89.5%	-	-
ORB-SLAM3 (Campos et al., 2021) Monocular Inertial (Campos et al., 2021)	74.5%	**0%**	14.5%	49.5%	38.0%	**0%**	40.0%	49.5%	20.0%	-
Proposed	**10.9%**	**0%**	**0%**	19.5%	**12.0%**	**0%**	33.5%	**11.1%**	**0%**	**0%**

TABLE 4 Results of different methods on different blur segments in BlurVO dataset.

Dataset BlurVO	Scene_1		Scene_2		Scene_3	
	Precision	Recall	Precision	Recall	Precision	Recall
Image-based (Kim et al., 2018)	92.3%	88.9%	**95.7%**	73.9%	92.0%	81.8%
Proposed	**96.2%**	**96.2%**	91.9%	**86.8%**	**97.0%**	**97.4%**

Values in bold in the table indicate that the data corresponds to the best method.

TABLE 5 Results of different VO and VIO methods on BlurVO dataset.

Dataset:BlurVO	Scene_1		Scene_2		Scene_3	
	ATE	FD	ATE	FD	ATE	FD
ORB-SLAM3 (Campos et al., 2021) Monocular (Campos et al., 2021)	2.32	45.9%	2.74	48.7%	3.05	66.7%
DSO (Engel et al., 2017)	-	80.5%	-	75.6%	-	82.1%
ORB-SLAM3 (Campos et al., 2021) Monocular Inertial (Campos et al., 2021)	2.41	52.3%	2.35	51.3%	2.74	50.7%
Proposed	**0.12**	**21.7%**	**0.19**	**5.7%**	**0.11**	**22.7%**

Values in bold in the table indicate that the data corresponds to the best method.

TABLE 6 Results of different methods on EuRoC (Burri et al., 2016) dataset.

Dataset:EuRoC (Burri et al., 2016)	MH 04		MH 05	
	ATE	FD	ATE	FD
ORB-SLAM3 (Campos et al., 2021) Monocular (Campos et al., 2021)	0.1494	0.20%	0.0656	**0%**
DSO (Engel et al., 2017)	0.1810	**0%**	0.1064	**0%**
ORB-SLAM3 (Campos et al., 2021) Monocular Inertial (Campos et al., 2021)	**0.1276**	8.62%	0.0643	13.57%
Proposed	0.1441	**0%**	**0.0466**	0.45%

Values in bold in the table indicate that the data corresponds to the best method.

in the video well and perform targeted repairs. We compared the IMU-based detection method with the image-based method on the BlurVO dataset. As shown in Table 4, our method achieved an accuracy of more than 90% and a recall of 85% in detecting blurry images, which is superior to the image-based method in terms of efficiency and effectiveness.

4.4 Overall performance

Our approach significantly improves all metrics on the BlurVO dataset, especially in ATE and FD reduction. In challenging blur scenes, our method consistently outperforms ORB-SLAM3 (Campos et al., 2021) and DSO (Engel et al., 2017),

demonstrating its effectiveness in handling motion blur. The detailed FD results of different methods under different BlurVO scenarios (Roll, Pitch, Yaw) are shown in Table 4. On the BlurVO data set, the initialization success rate of ORB-SLAM3 (Campos et al., 2021) and DSO (Campos et al., 2021) is meager, causing them to fail to initialize successfully. Although ORB-SLAM3 Monocular Inertial (Campos et al., 2021) combines IMU data to improve accuracy, it still does not reduce its dependence on images, so they do not improve the robustness much. Unlike this, our method makes good use of IMU data to restore blurred images and successfully enhances the robustness of VO. Excellent results are achieved in Pitch and Yaw, the two most common types of motion blur. An overall comparison of these methods on BlurVO and EuRoC (Burri et al., 2016) is shown in Table 5, 6. In BlurVO, DSO (Campos et al., 2021) has the worst robustness, so we cannot effectively measure the accuracy of DSO (Campos et al., 2021) on BlurVO through experiments, and the accuracy of ORB-SLAM3 (Campos et al., 2021) is also interfered with by blurred images. Our method effectively eliminates these interferences, thereby improving the accuracy. On the EuRoC (Burri et al., 2016) dataset, since there is not much motion blur, the improvement of our method is not apparent. ORB-SLAM3 monocular inertial (Campos et al., 2021) has high requirements for the quality of IMU data and images, which may lead to tracking failure at some nodes.

5 Conclusion

This paper presents a novel and practical approach to enhance VO in motion blur scenarios. Our method uniquely leverages an inertial neural network to analyze IMU data, enabling the detection of various types of motion blur with high precision. This approach facilitates the intelligent recovery of blurred sequences, thereby significantly improving the robustness and reliability of VO. Furthermore, we introduce BlurVO, a comprehensive motion blur dataset designed for VO research. Future research focuses on optimizing the inertial neural network and exploring the integration of our method with VO and VIO systems, expanding the scope and applicability of robust VO solutions.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

SL: Conceptualization, Data curation, Formal Analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing–original draft, Writing–review and editing. CY: Conceptualization, Data curation, Formal Analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing–original draft, Writing–review and editing. XQ: Writing–original draft, Writing–review and editing. DC: Writing–original draft, Writing–review and editing. WS: Writing–original draft, Writing–review and editing.

Funding

The author(s) declare that no financial support was received for the research, authorship, and/or publication of this article.

Conflict of interest

Author WS was employed by Horizon Robotics.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

Benseddik, H.-E., Morbidi, F., and Caron, G. (2020). Panoramis: an ultra-wide field of view image dataset for vision-based robot-motion estimation. *Int. J. Robotics Res.* 39 (9), 1037–1051. doi:10.1177/0278364920915248

Burri, M., Nikolic, J., Gohl, P., Schneider, T., Rehder, J., Omari, S., et al. (2016). The euroc micro aerial vehicle datasets. *Int. J. Robotics Res.* 35 (10), 1157–1163. doi:10.1177/0278364915620033

Campos, C., Elvira, R., Rodríguez, J. J. G., Montiel, J. M., and Tardós, J. D. (2021). Orb-slam3: an accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Trans. Robotics* 37 (6), 1874–1890. doi:10.1109/tro.2021.3075644

Carlevaris-Bianco, N., Ushani, A. K., and Eustice, R. M. (2016). University of Michigan north campus long-term vision and lidar dataset. *Int. J. Robotics Res.* 35 (9), 1023–1035. doi:10.1177/0278364915614638

Chen, L., Han, M., and Wan, H. (2013). "The fast iris image clarity evaluation based on brenner," in *International Symposium on instrumentation and measurement, sensor network and automation (IMSNA)* (IEEE), 300–302.

Cho, S.-J., Ji, S.-W., Hong, J.-P., Jung, S.-W., and Ko, S.-J. (2021). "Rethinking coarse-to-fine approach in single image deblurring," in IEEE International Conference on Computer Vision, 4641–4650.

Davison, A. J. (2003). "Real-time simultaneous localisation and mapping with a single camera," in IEEE International Conference on Computer Vision (IEEE Computer Society), 1403. doi:10.1109/iccv.2003.12386543

Delmerico, J., Cieslewski, T., Rebecq, H., Faessler, M., and Scaramuzza, D. (2019). "Are we ready for autonomous drone racing? the uzh-fpv drone racing dataset," in IEEE International Conference on Robotics and Automation (IEEE), 6713–6719.

Engel, J., Koltun, V., and Cremers, D. (2017). Direct sparse odometry. *IEEE Trans. Pattern Analysis Mach. Intell.* 40 (3), 611–625. doi:10.1109/tpami.2017.2658577

Engel, J., Schöps, T., and Cremers, D. (2014). "Lsd-slam: large-scale direct monocular slam," in European Conference on Computer Vision (Springer), 834–849.

Furgale, P., Rehder, J., and Siegwart, R. (2013). "Unified temporal and spatial calibration for multi-sensor systems," in IEEE International Conference on Intelligent Robots and Systems (IEEE), 1280–1286.

Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics: the kitti dataset. *Int. J. Robotics Res.* 32 (11), 1231–1237. doi:10.1177/0278364913491297

He, M., Zhu, C., Huang, Q., Ren, B., and Liu, J. (2020). A review of monocular visual odometry. *Vis. Comput.* 36 (5), 1053–1065. doi:10.1007/s00371-019-01714-6

Kim, B., Son, H., Park, S.-J., Cho, S., and Lee, S. (2018). Defocus and motion blur detection with deep contextual features. *Comput. Graph. Forum* 37 (7), 277–288. doi:10.1111/cgf.13567

Kupyn, O., Martyniuk, T., Wu, J., and Wang, Z. (2019). "Deblurgan-v2: deblurring (orders-of-magnitude) faster and better," in IEEE International Conference on Computer Vision, 8878–8887.

Leung, K. Y., Halpern, Y., Barfoot, T. D., and Liu, H. H. (2011). The utias multi-robot cooperative localization and mapping dataset. *Int. J. Robotics Res.* 30 (8), 969–974. doi:10.1177/0278364911398404

Leutenegger, S., Furgale, P. T., Rabaud, V., Chli, M., Konolige, K., and Siegwart, R. Y. (2013). "Keyframe-based visual-inertial slam using nonlinear optimization," in *Robotics: science and systems*.

Ligocki, A., Jelinek, A., and Zalud, L. (2020). "Brno urban dataset-the new data for self-driving agents and mapping tasks," in 2020 IEEE International Conference on Robotics and Automation (ICRA) (IEEE), 3284–3290.

Nistér, D., Naroditsky, O., and Bergen, J. (2004). "Visual odometry," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition. doi:10.1109/cvpr.2004.1315094

Purohit, K., Shah, A. B., and Rajagopalan, A. (2018). "Learning based single image blur detection and segmentation," in 2018 25th IEEE International Conference on Image Processing (ICIP) (IEEE), 2202–2206.

Qin, T., Li, P., and Shen, S. (2018). Vins-mono: a robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robotics* 34 (4), 1004–1020. doi:10.1109/tro.2018.2853729

Roberts, R., Nguyen, H., Krishnamurthi, N., and Balch, T. (2008). "Memory-based learning for visual odometry," in IEEE International Conference on Robotics and Automation (IEEE), 47–52.

Schubert, D., Goll, T., Demmel, N., Usenko, V., Stückler, J., and Cremers, D. (2018). "The tum vi benchmark for evaluating visual-inertial odometry," in IEEE International Conference on Intelligent Robots and Systems (IEEE), 1680–1687.

Shi, X., Li, D., Zhao, P., Tian, Q., Tian, Y., Long, Q., et al. (2020). "Are we ready for service robots? the openloris-scene datasets for lifelong slam," in IEEE International Conference on Robotics and Automation (IEEE), 3139–3145.

Sturm, J., Engelhard, N., Endres, F., Burgard, W., and Cremers, D. (2012). "A benchmark for the evaluation of rgb-d slam systems," in IEEE International Conference on Intelligent Robots and Systems (IEEE), 573–580.

Tang, C., Hou, C., Hou, Y., Wang, P., and Li, W. (2017). An effective edge-preserving smoothing method for image manipulation. *Digit. Signal Process.* 63, 10–24. doi:10.1016/j.dsp.2016.10.009

Tao, X., Gao, H., Shen, X., Wang, J., and Jia, J. (2018). "Scale-recurrent network for deep image deblurring," in IEEE International Conference on Computer Vision, 8174–8182.

Vankawala, F., Ganatra, A., and Patel, A. (2015). A survey on different image deblurring techniques. *Int. J. Comput. Appl.* 116 (13), 15–18. doi:10.5120/20396-2697

Von Stumberg, L., Usenko, V., and Cremers, D. (2018). "Direct sparse visual-inertial odometry using dynamic marginalization," in IEEE International Conference on Robotics and Automation (IEEE), 2510–2517.

Wen, W., Zhou, Y., Zhang, G., Fahandezh-Saadi, S., Bai, X., Zhan, W., et al. (2020). "Urbanloco: a full sensor suite dataset for mapping and localization in urban scenes," in 2020 IEEE international conference on robotics and automation (ICRA) (IEEE), 2310–2316.

Yi, X., and Eramian, M. (2016). Lbp-based segmentation of defocus blur. *IEEE Trans. image Process.* 25 (4), 1626–1638. doi:10.1109/tip.2016.2528042

Yin, J., Li, A., Li, T., Yu, W., and Zou, D. (2021). M2dgr: a multi-sensor and multi-scenario slam dataset for ground robots. *IEEE Robotics Automation Lett.* 7 (2), 2266–2273. doi:10.1109/lra.2021.3138527

Zhao, W., Wei, F., Wang, H., He, Y., and Lu, H. (2023). Full-scene defocus blur detection with defbd+ via multi-level distillation learning. *IEEE Trans. Multimedia* 25, 9228–9240. doi:10.1109/tmm.2023.3248162