A Lightweight Sensor Fusion for Neural Visual Inertial Odometry

Yao Lu[1], Xiaoxu Yin[2], Feng Qin[3], Ke Huang[4], Menghua Zhang[1(✉)], and Weijie Huang[1(✉)]

[1] School of Electrical Engineering, University of Jinan, Jinan 250000, China
zhangmenghua@mail.sdu.edu.cn, cse_huangwj@ujn.edu.cn
[2] Qilu Aerospace Information Research Institute, Jinan, China
[3] Zaozhuang Vocational College of science and technology, Zaozhuang, China
[4] School of Information Science and Engineering,
Shandong Normal University, Jinan 250000, China

Abstract. In recent years, the performance of visual inertial odometry (VIO) based on deep learning has shown significant advantages over traditional geometric methods. However, all existing methods estimate each pose through visual and inertial measurements, which involves a large amount of computational redundancy, resulting in huge time costs and hardware damage when training and deploying on devices. In order to maintain accuracy while reducing the number of training parameters, an improved algorithm based on Visual-Selective-VIO is proposed. To reduce the number of network parameters and maintain the training accuracy, a unique attention mechanism is designed for the visual branch and a lightweight pose estimation module. By improving the visual branch, we serialize the information of attention feature maps, covering both channel and spatial dimensions. Then, we multiply these two feature maps with the original input feature maps for adaptive feature correction. This method improves the sensitivity of the model to channel features and enables more accurate image localization. Experimental results show that our algorithm maintains accuracy with a 10% reduction in network parameters compared to advanced VIO algorithm, making it more suitable for training large-scale datasets and deployment in practical applications.

Keywords: Visual inertial odometry · Gate recurrent unit · Adaptive learning

1 Introduction

Humans can perceive their own motion in space through a variety of multimodal fusion methods. Optic flow (vision) and proprioception (inertial sensors) are the two most important sensory information for humans to perceive their self-motion [1].

H. Zhang et al. (Eds.): NCAA 2023, CCIS 1870, pp. 46–59, 2023.
https://doi.org/10.1007/978-981-99-5847-4_4

Estimating six degrees of freedom (6-DOF) motion is one of the important technical challenges faced by robots and autonomous driving. The advantages of visual cameras, such as low cost and ease of operation, have made them widely used in these fields. Over the past decade, with the development of visual odometers and visual synchronous localization and mapping (VSLAM) [7], this challenge has gradually gained attention and exploration. These technologies provide an important background and foundation for visual based 6-DOF motion estimation. 6-DOF motion estimation has shown impressive results. However, while methods such as DSO [8] and ORB-SLAM [9] have achieved high precision and real-time positioning in large-scale environments, there is still much room for improvement in positioning accuracy under non-textured environments, image blurring, and extreme lighting conditions. In the fields of computer vision, robotics, and autonomous driving, visual-inertial odometry based on the fusion of visual information and inertial sensor information is currently a topic of strong research interest [2–6]. Compared with traditional visual odometry, the visual-inertial odometry system includes additional IMU information, which can improve the motion tracking performance of mobile agents in non-textured environments or under extreme lighting conditions, and provide more accurate and robust attitude information. At the same time, the low cost, high performance, and all-time domain advantages of camera and inertial sensor fusion are widely used in the fields of robotics, drones, and smart phones. However, traditional visual-inertial odometry methods (not based on deep learning) heavily rely on manual intervention for fault case analysis and system initialization selection, and require careful parameter tuning for various specific environments. Deploying such a system with rapid calibration in fast-moving scenarios still faces significant challenges.

In recent years, with the continuous development and successful application of deep learning methods in various computer vision tasks [15–17], deep learning and data-driven VIO methods [7,10–14] have attracted widespread attention and demonstrated strong competitiveness in some complex and specific scenarios.

Compared to traditional geometric based methods, deep learning based VIO solutions utilize deep neural networks (DNNs) to extract higher quality features. These solutions are trained on large-scale datasets to learn better fusion of visual and inertial features, and to filter out abnormal sensor data. However, training large-scale datasets requires a significant amount of time and resource costs. In order to reduce the number of network parameters and maintain the accuracy of training, we propose an architecture that combines GRU and CBAM.

By using this combination architecture, we can reduce the number of network parameters while maintaining training accuracy. Experiments have shown that our designed method can effectively improve the accuracy of training. The advantage of this method is that it can better capture the correlation between images and inertial data, and can automatically filter out interference from abnormal sensors.

Our research results indicate that VIO solutions based on deep learning have better performance and adaptability compared to traditional methods. By

training on large-scale datasets, we can enable the model to learn richer feature representations and reduce training costs by optimizing the network structure. This will provide higher accuracy and efficiency for the application of VIO technology, and provide more reliable solutions for future visual navigation and positioning tasks.

In this paper, our main contributions are summarized as follows:

- A novel framework to reduce the parameter size of the training network is proposed, which can improve the efficiency of deployment on devices, and reduce computational costs. The method has been fully compared with other advanced algorithms and provides a new solution for training VIO large-scale datasets.
- The complementary advantages between Gate Recurrent Unit and Convolutional Block Attention Module are discovered in VIO field in this article.
- Our method is extensively tested on the KITTI Odometry dataset, and achieves good performance in terms of adaptability.

2 Relate Work

2.1 VO

The VO algorithm estimates the incremental self motion of the camera. A traditional VO algorithm, involves extracting features from an image, matching features between the current image and subsequent images, and then calculating optical flow. Motion can be calculated using optical flow. The fast semi direct monocular visual odometer (SVO) algorithm (Forster, Pizzoli, and Scaramuzza 2014) is an example of the most advanced VO algorithm. Its design is to directly operate on image patches without relying on slow feature extraction, thus achieving fast and robust performance. On the contrary, it uses a probability depth filter on the patch of the image itself. Then update the depth filter by aligning the entire image. This algorithm runs in real-time on embedded platforms and has high computational efficiency. However, its probability formula makes it difficult to tune, and it also requires a bootstrap process to initiate this process. As expected, its performance largely depends on the hardware used to prevent tracking failures - typically requiring the use of a global shutter camera above 50 fps to ensure accurate mileage estimation [24,25].

2.2 Traditional VIO Methods

In recent years, VIO has become a highly focused method that integrates camera and IMU data into a pose estimator, with the ability to provide higher robustness and accuracy in complex and dynamic environments. In the past few decades, tightly coupled VIO systems can be mainly divided into two categories: filter based methods and optimization based methods. Among the filter based methods, representative ones include MSCKF [19] and ROVIO [20]. MSCKF combines geometric constraints and IMU measurements in a multi-state constrained

extended Kalman filter (EKF), which has low computational complexity and provides accurate attitude estimation in large scale real world environments. ROVIO uses EKF to fuse IMU data and photometric errors, which is another popular filter based VIO method. Among the optimization based methods, representative ones include OKVIS [21] and VINS [5]. OKVIS is a keyframe based VIO system, while VINS is a tightly coupled method based on nonlinear optimization that achieves high precision mileage measurement by integrating pre integrated IMU measurements and feature observations. These VIO methods have made significant progress in combining camera and IMU data, and have been widely applied in the field of attitude estimation. They exhibit excellent performance and robustness in different environments and application scenarios.

2.3 Deep Learning-Based VIO

With the development of hardware, deep learning based methods have achieved significant success in computer vision applications, including VIO. VINet [7] is the first end-to-end trainable depth learning VIO method, which learns attitude regression from image sequences and IMU measurements through supervised learning. In this method, the long short memory (LSTM) network is introduced to model the correlation of temporal motion. Subsequently, Chen et al. [10] proposed two different masking techniques to selectively fuse visual and inertial features. ATVIO [11] adopts an attention based fusion function and applies adaptive loss for attitude regression. Recent research has also proposed a self supervised learning framework to learn 6-DoF self motion without ground annotation during training. Shamwell et al. [12] proposed VIOLearner, which estimates attitude through a multi-level error correction view synthesis method. DeepVIO [13] improves VIO's attitude estimation through additional optical flow self supervision. In addition, Almalioglu et al. [14] demonstrated a self supervised VIO method based on depth estimation. Mingyu Yang and Yu Chen [18] proposed an adaptive method for disabling dynamic visual modality for visual selective VIO. This method can effectively fuse visual and IMU information in specific environments, thereby improving the accuracy and efficiency of localization. This study provides valuable ideas and methods for the further development of the VIO field. These deep learning based methods have made significant progress in the field of VIO and provide new ideas and technical means for achieving more accurate and efficient attitude estimation. They are of great significance for solving complex visual navigation and positioning problems, and are expected to provide more reliable solutions for future robots and autonomous navigation systems.

3 Method

The main network structure proposed in this article mainly consists of Inertial Encoder, Visual Encoder, fusion module, pose estimation module, and Decision Module. The Inertial Encoder consists of Conv1d, BatchNorm1d, and LeakyReLU, while the Visual Encoder consists of Conv2d, CBAM, BatchNorm2d, and LeakyReLU, with CBAM connected after downsampling at each

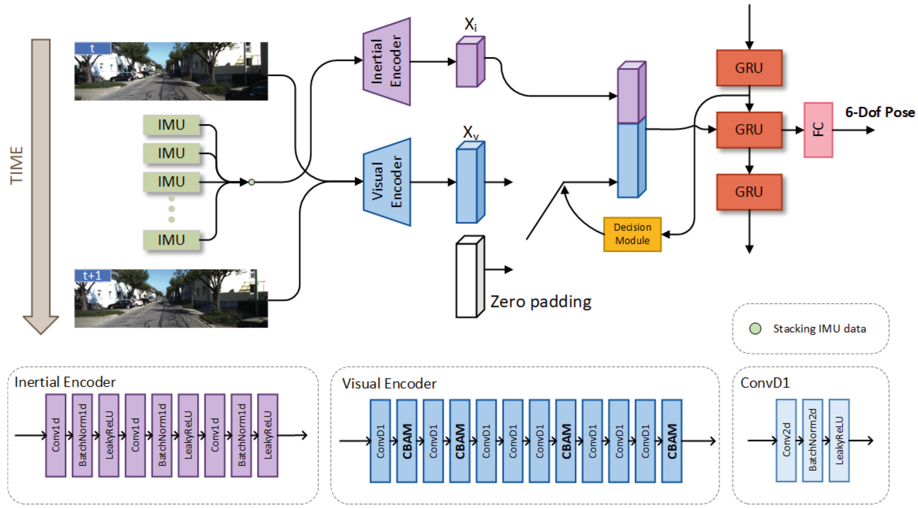

Fig. 1. Proposed architecture for pose estimation

layer. The pose estimation module consists of GRU and FC. The imu features and visual features output by the Encoder are input into the fusion network for fusion output, and then input into the pose estimation module to output 6-DoF. As shown in Fig. 1. Here, this article draws inspiration from the decision network approach proposed by Mingyu Yang, Yu Chen et al. [18] in Visual Selective VIO and designs the Decision Module. During the training process, we use Gumbel Softmax distribution to sample decisions from the decision module to ensure that the entire system is end-to-end differentiable. In the reasoning process, the decision is sampled through the Bernoulli distribution controlled by the policy network.Once the decision module determines the use of visual modality, the current image will be processed by a visual encoder and the obtained visual features, along with inertial features, will be provided to the attitude estimation module for regression GRU attitude estimation. However, if the decision module decides to disable the visual encoder, the input of zero padding will be passed to the GRU to ensure the continuity of the calculation process. This design enables the system to perform flexible input processing according to the instructions of the decision module, while maintaining the trainability and effectiveness of the algorithm.

3.1 Attention Mechanism for the Visual Branch

In order to improve the representation ability and performance of the CNN network, we introduce the Convolutional Block Attention Module(CBAM) module in our algorithm, as shown in Fig. 2. Traditional convolutional networks only focus on local information and often ignore global information, which leads to poor performance. Therefore, the CBAM module can better focus on the global

information of the monocular camera image. The channel attention module compresses the spatial dimensions while keeping the channel dimension unchanged, focusing on meaningful information in the image. The spatial attention module compresses the channel dimensions while keeping the spatial dimensions unchanged, focusing on the position information of the target. By using these two modules, the computational performance of the model is significantly improved with a small increase in computational and parameter complexity. The formula of Channelx Attention Module as (1):

$$\begin{aligned} \mathrm{M_c(F)} &= \sigma(MLP(AvgPool(\mathrm{F}))+MLP(MaxPool(\mathrm{F}))) \\ &= \sigma(\mathrm{W_1(W_0(F^c_{avg}))+W_1(W_0(F^c_{max}))}) \end{aligned} \tag{1}$$

The formula of Spatial Attention Module as (2):

$$\begin{aligned} \mathrm{M_s(F)} &= \sigma(f^{7*7}[AvgPool(\mathrm{F});MaxPool(\mathrm{F})]) \\ &= \sigma(f^{7*7}[\mathrm{F^s_{avg}};\mathrm{F^s_{max}}]) \end{aligned} \tag{2}$$

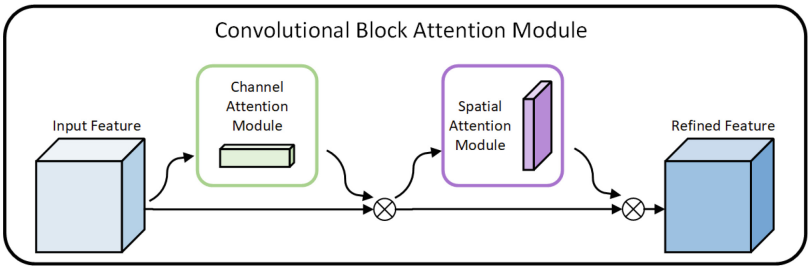

Fig. 2. Convolution Block Attention Module

3.2 Lightweight Pose Estimation Module

To address the challenge of reducing the number of parameters in a model while maintaining its computational performance, we have introduced the Gate Recurrent Unit (GRU) module, as illustrated in Fig. 3. This module has fewer parameters, which greatly improves hardware computation and time costs, providing a significant advantage for engineering practical applications. By incorporating a two-layer GRU-based pose estimation Recurrent Neural Network (RNN) in our study, we aim to enhance the accuracy and efficiency of our model.

The GRU module is a type of RNN that utilizes gating mechanisms to control the flow of information. Specifically, it employs two gates, namely the update gate and reset gate, which work together to regulate the memory content of the cell. The update gate determines the proportion of new information that should be retained in the cell state, while the reset gate controls the amount of old information that should be discarded.

By utilizing the GRU module, we can effectively reduce the number of parameters in our model without sacrificing its performance. This is particularly useful

in practical applications where computational resources are limited, and efficiency is crucial. Our two-layer GRU-based pose estimation RNN leverages the advantages of the GRU module to improve the accuracy and robustness of our model, making it suitable for a wide range of applications, including autonomous driving, robotics, and augmented reality.

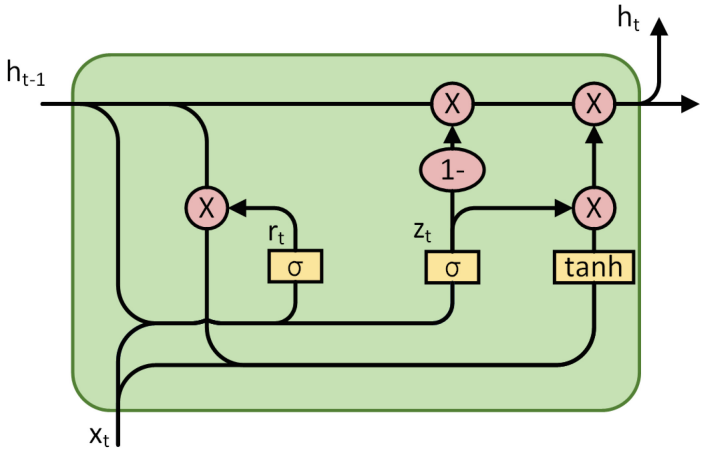

Fig. 3. Gate Recurrent Unit

3.3 Loss Function

In our training process, we use the mean square error (MSE) loss function to minimize the attitude estimation error, which is defined by formula (3). This loss function plays a key role in the training process, helping us measure the accuracy of attitude estimation. By applying MSE loss to attitude estimation error, our goal is to reduce the model's error in this task and improve its performance:

$$\mathcal{L}_{\text{pose}}=\frac{1}{3(\mathrm{T}-1)}\sum_{\mathrm{t}=1}^{\mathrm{t}-1}(||\widehat{\mathrm{v}}_{\mathrm{t}}-\mathrm{v}_{\mathrm{t}}||_2^2+\alpha||\widehat{\varphi}_{\mathrm{t}}-\varphi_{\mathrm{t}}||_2^2) \tag{3}$$

In the formula, T is the length of the training sequence. $\mathrm{v}_t$and φ_t represent the ground truth translation vector and rotation vector, respectively. α is the weight that balances the translation loss and rotation loss, and it is set to 100 according to the previous supervised VO/VIO methods. Additionally, we apply an extra penalty factor C to the use of each visual encoder to encourage disabling of visual feature computation. During the training process, we compute the average penalty and define it as the efficiency loss:

$$\mathcal{L}_{eff}=\frac{1}{T-1}\sum_{t}^{t-1}\mathrm{C}\mathrm{d}_t \tag{4}$$

Finally, we train the end-to-end system to comprehensively consider the sum of attitude estimation loss and efficiency loss (Eq. 5), in order to achieve a balance of computational efficiency while maintaining good accuracy.

$$\mathcal{L} = \mathcal{L}_{pose} + \mathcal{L}_{eff} \tag{5}$$

During the training process, we not only focus on the accuracy of attitude estimation, but also consider the computational efficiency of the system. We combine attitude estimation loss with efficiency loss to find a balance point. This balance point enables our system to perform efficiently while providing accurate attitude estimation.

4 Experiment

4.1 Dataset

We tested our method on the KITTI Odometry dataset [23], which is a highly influential VO/VIO dataset in the field. The dataset includes 22 stereo video sequences, out of which sequences 00–10 provide ground truth trajectories, while sequences 11–22 are used for evaluation without ground truth. To follow the procedure described in [22], sequence 03 was excluded as it lacked raw data. We trained our model on sequences 00, 01, 02, 04, 06, 08, and 09, and tested it on sequences 05, 07, and 10. The dataset's left monocular images were used for this purpose, with the frequency of image and ground truth poses is 10 Hz and the frequency of IMU data is 100 Hz.

4.2 Experimental Setup and Details

This architecture was implemented using PyTorch and trained on NVIDIA 3090Ti GPU. During the training process, we adjusted all images to a size of 512×256. The sequence length of training was set as 11. We inserted 11 frames of IMU data between every two images, resulting in an input dimension of 6×11 for IMU data. The visual encoder used a pre trained FlowNet-S network for optical flow estimation, as detailed in reference [22]. CBAM was added after each downsampling layer, and a fully connected (FC) layer was added at the end to generate 512 dimensional visual features. We used three one-dimensional convolutions for the IMU data branch and one FC layer, generating 256 dimensional inertial features. The attitude estimation network consisted of two GRU layers, which has 1024 gate units. The hidden state of the last GRU layer was used to estimate a 6-degree of freedom attitude through two layers of MLP at every time step. The training process was divided into two major stages which included warm-up and joint-training stage. When it was at the warm-up stage, a random strategy was used to train the visual encoder, inertial encoder, and attitude estimation network for 40 epoches, and the output of the visual encoder was used with a 50% probability. The learning rate was set to 7×10^{-5} during

this stage. When it was at the warm-up stage the joint-training stage, all end-to-end components (including the policy network) were trained for 40 epoches with a learning rate of 7×10^{-6}, followed by another 20 epoches with a learning rate set at 1×10^{-7}. The batch size was set as 32. During the training, the visual information of the first frame was used to ensure effective initial pose estimation. The implementation and training settings of this architecture ensured effective processing and feature extraction of images and IMU data, and optimized the performance of each component through joint training to achieve accurate attitude estimation.

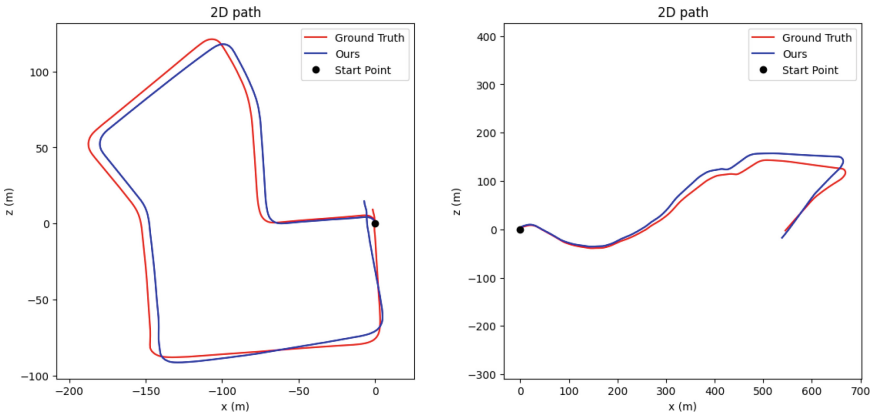

Fig. 4. Ground truth trajectories and motion trajectories on KITTI sequences 07 and 10.

In order to thoroughly evaluate the accuracy of our odometry estimates, we have computed the root mean square error (RMSE) of the estimated translation and rotation vectors for the entire trajectory. This is a widely used metric for evaluating the overall performance of odometry systems. In addition, we have also assessed the relative translation and rotation errors, denoted by t_{rel} and r_{rel}, respectively. These metrics are used to evaluate the accuracy of odometry estimates for various subsequence path lengths, as described in [23].

The RMSE metric provides a comprehensive assessment of the performance of our odometry system by considering the error of both the translation and rotation vectors. By computing the RMSE for the entire trajectory, we can obtain an overall measure of the system's accuracy. This allows us to compare the performance of our system against other state-of-the-art methods.

In addition to the RMSE, we have also evaluated the relative translation and rotation errors. These metrics are particularly useful when assessing the performance of odometry systems for specific subsequence path lengths. By analyzing the t_{rel} and r_{rel} metrics for various subsequence path lengths, we can gain a better understanding of the performance of our system in different scenarios. This

helps us to identify any potential weaknesses in our system and devise strategies to improve its accuracy and robustness.

Overall, the combination of the RMSE and relative translation and rotation errors provides a comprehensive and detailed evaluation of the accuracy of our odometry system. By thoroughly analyzing these metrics, we can identify areas for improvement and further optimize our system to meet the requirements of various practical applications.

4.3 Main Result

We evaluated our method on the KITTI dataset and compared it with the full modal baseline, GRU-only, and CBAM-only approaches. To ensure a fair comparison, we trained our proposed model and the other three models using the same optimizer and common hyperparameters, including the number of epochs and learning rate. We tested the models on the KITTI dataset and calculated the average usage of the visual encoder and the average root mean square error (RMSE) of translation and rotation. Table 1 summarizes the results.

Table 1. The relative translational t_{rel} & rotationalr$_{rel}$ error, and visual encoder usage of the baseline model and the overall parameter quantity of the network

	Seq.05			Seq.07			Seq.10			The amount of parameters
	t_{rel}	r_{rel}	usage	t_{rel}	r_{rel}	usage	t_{rel}	r_{rel}	usage	
Baseline	2.8431	1.0804	27.1475	2.7688	2.2087	29.4813	3.6016	1.7061	31.7765	48.454376M
Only GRU	8.2991	4.1528	**22.037**	11.2387	7.7288	23.5669	14.1928	6.8264	**22.6856**	44.518120M
Only CBAM	3.7812	1.6506	31.1707	**3.280**	3.2786	33.6670	**3.5754**	1.8408	37.1143	48.664988M
Ours	**3.6371**	**1.6365**	27.8724	3.3949	**2.8249**	**21.929**	5.6794	**1.7744**	29.8582	**44.728732M**

We conducted multiple experiments and found that with the addition of GRU, the usage of visual encoder and total parameter count of the network both decreased as expected. When CBAM was introduced, The incorporation of CBAM has resulted in more comprehensive features covering the object to be recognized, leading to improved object recognition probability. This suggests that the attention mechanism has effectively trained the network to prioritize key information for improved recognition. But in this paper, adding a separate attention mechanism to the visual side can increase the dominance of visual information, resulting in a decrease in accuracy. Therefore, the GRU was added to suppress overly strong visual features to achieve a balanced effect. Moreover, through comparison, we found that our method increased by 0.6–1% in terms of relative translation/rotation error, but the overall parameter count of the network decreased by 7%.

In Fig. 5, we present a visual explanation of the frequency and vehicle speed used by the visual encoder on sequence 07. The description in the upper left corner uses color coding, with darker colors indicating lower usage and lighter colors indicating higher usage, to demonstrate the use of visual encoders in

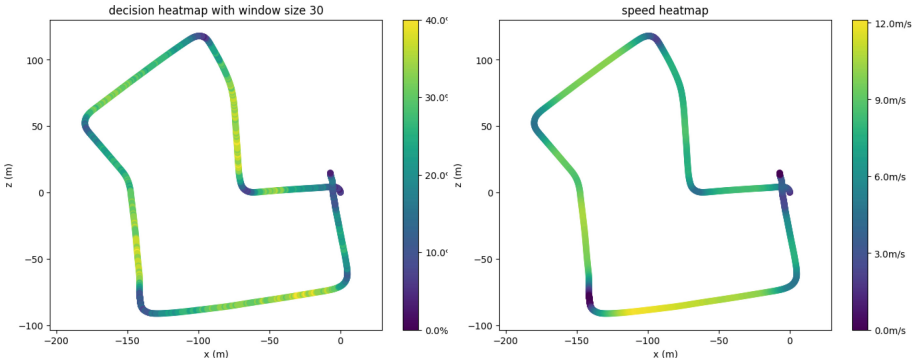

Fig. 5. The visualization of the learning strategy for sequence 07. The image in the upper left corner shows the mapping of visual encoder usage, showing the local usage of each time step. The top right corner displays a proxy vehicle speed chart. Strategic networks tend to activate visual encoders more frequently during fast linear movements, while reducing their use during slow movements and turns.

local areas. At the same time, in the upper right corner, we show the speed changes of the agent at each time step, with darker colors indicating lower speeds. Through this strategy, the system can dynamically adjust the utilization of visual encoders based on different motion states to more effectively handle different scenes and actions. The visualization of this learning strategy helps us understand the decision-making process of the system under different conditions and its adaptability to visual encoders.

Observing the diagram, it can be observed that there is a clear correlation between the use of visual modality and vehicle speed and turning angle. When the vehicle is moving slowly or making turns, the strategy network uses less visual mode. This may be because the perception of environmental details during slow driving and turning is not as critical, resulting in a relatively low level of activation of the visual encoder. However, we observed that the visual encoder was activated more frequently when the proxy vehicle was traveling rapidly in a straight line.

This behavior can be explained by the inherent property of direct measurement of angular velocity in IMU. Compared to angle estimation based on visual features, using IMU to estimate turning angles is relatively easy. This is because the angular velocity can be calculated through a simple first-order integration. However, for the estimation of the translation process, additional IMU measurement is required. Because IMU can only measure the acceleration, which is the second derivative of the translation, the velocity constraint needs to be initialized. Therefore, relying solely on IMU for estimation usually results in significant errors when the vehicle is moving rapidly. To reduce this error, the strategy network frequently uses visual modalities to provide additional information.

In summary, the visual explanation in Fig. 5 reveals the relationship between the frequency of visual encoder usage and vehicle speed. It displays the changes

in the activation level of visual modes under different driving states, as well as the role of visual encoders in vehicle control. This is very valuable for a deep understanding of the decision-making process and perception ability of autonomous driving systems.

5 Conclusion

In this paper, a novel method is proposed to address the challenge of integrating VIO algorithms into devices more easily. Our method reduces model parameters by introducing GRU and improves accuracy by incorporating CBAM. Additionally, the visual modality can be opportunistically disabled when visual information is not critical, reducing computational cost and power consumption. Our experiments demonstrate that our method provides approximately 10% reduction in parameter computation with no significant performance degradation. Moreover, the learned policy is interpretable and exhibits scene-dependent adaptive behavior. Our adaptive learning strategy is model independent, so it can be easily applied in other deep VIO systems. The universality of this strategy enables it to quickly migrate to different systems and frameworks without requiring significant modifications and adaptation. This provides researchers and developers with a flexible and efficient method to utilize adaptive learning strategies in their own deep VIO systems, thereby improving the performance and robustness of attitude estimation. This portability and ease of use make our learning strategy a valuable tool that can promote further research and application in the field of deep VIO.

Acknowledgement. This work was supported by the Youth Foundations of Shandong Province under Grant Nos. ZR202102230323 and ZR2021QF130, the National Natural Science Foundation of China under Grant No. 62273163, and the Key R & D Project of Shandong Province under Grant No. 2022CXGC010503.

References

1. Fetsch, C.R., Turner, A.H., DeAngelis, G.C., Angelaki, D.E.: Dynamic reweighting of visual and vestibular cues during self-motion perception. J. Neurosci. **29**(49), 15601–15612 (2009)
2. Forster, C., Carlone, L., Dellaert, F., Scaramuzza, D.: Onmanifold preintegration for real-time visual Cinertial odometry. IEEE Trans. Rob. **33**(1), 1–21 (2017)
3. Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R., Furgale, P.: Keyframe-based visual Cinertial odometry using nonlinear optimization. Int. J. Robot. Res. **34**(3), 314–334 (2015)
4. Li, M., Mourikis, A.I.: High-precision, consistent EKF based visual-inertial odometry. Int. J. Robot. Res. **32**(6), 690–711 (2013)

5. Qin, T., Li, P., Shen, S.: VINS-MONO: a robust and versatile monocular visual-inertial state estimator. IEEE Trans. Rob. **34**(4), 1004–1020 (2018)
6. Clark, R., Wang, S., Wen, H., Markham, A., Trigoni, N.: ViNet: visual-inertial odometry as a sequence-to-sequence learning problem. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31 (2017)
7. Cadena, C., et al.: Past, present, and future of simultaneous localization and mapping: toward the robust-perception age. IEEE Trans. Rob. **32**(6), 1309–1332 (2016)
8. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry. IEEE Trans. Pattern Anal. Mach. Intell. **40**(3), 611–625 (2017)
9. Mur-Artal, R., Tard®s, J.D.: Orb-slam2: an open-source slam system for monocular, stereo, and RGB-D cameras. IEEE Trans. Robot. **33**(5), 1255–1262 (2017)
10. Chen, C., Rosa, S., Miao, Y., et al.: Selective sensor fusion for neural visual-inertial odometry. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10542–10551 (2019)
11. Liu, L., Li, G., Li, T.H.: AtVio: attention guided visual-inertial odometry. In ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4125–4129. IEEE (2021)
12. Shamwell, E.J., Leung, S., Nothwang, W.D.: Vision-aided absolute trajectory estimation using an unsupervised deep network with online error correction. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2524–2531. IEEE (2018)
13. Han, L., Lin, Y., Du, G., Lian, S.: Deepvio: self-supervised deep learning of monocular visual inertial odometry using 3D geometric constraints. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 6906–6913. IEEE (2019)
14. Almalioglu, Yasin, et al.: SelfVIO: self-supervised deep monocular Visual CInertial Odometry and depth estimation, pp. 119–136. Neural Networks, 150 (2022)
15. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. Commun. ACM **60**(6), 84–90 (2017)
16. Simonyan K, Zisserman A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
17. Ren, S., et al.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, vol. 28 (2015)
18. Yang, M., Chen, Y., Kim, H.S.: Efficient deep visual and inertial odometry with adaptive visual modality selection. In: Computer Vision CECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, pp. 233–250. Proceedings, Part XXXVIII (2022)
19. Mourikis, A.I., Roumeliotis, S.I.: A multi-state constraint Kalman filter for vision-aided inertial navigation. In: Proceedings 2007 IEEE International Conference on Robotics and Automation, pp. 3565–3572. IEEE (2007)
20. Bloesch, M., Omari, S., Hutter, M., Siegwart, R.: Robust visual inertial odometry using a direct EKF-based approach. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 298–304. IEEE (2015)
21. Leutenegger, S., Furgale, P., Rabaud, V., et al.: Keyframe-based visual-inertial slam using nonlinear optimization. In: Proceedings of Robotis Science and Systems (RSS) 2013 (2013)
22. Chen, C., et al.: Selective sensor fusion for neural visual-inertial odometry. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10542–10551 (2019)

23. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The kitti vision benchmark suite. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3354–3361. IEEE (2012)
24. Forster, C., Carlone, L., Dellaert, F., Scaramuzza, D.: IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation. In: Robotics: Science and Systems XI (2015)
25. Forster, C., Pizzoli, M., Scaramuzza, D.: SVO: fast semi-direct monocular visual odometry. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 15–22. IEEE (2014)