Visual Odometry Offloading in Internet of Vehicles with Compression at the Edge of the Network

L. Qingqing[1,2], J. Peña Queralta[2], T. N. Gia[2], H. Tenhunen[3], Z. Zou[1] and T. Westerlund[2]

[1] School of Information Science and Technology, Fudan Universtiy, China
[2] Department of Future Technologies, University of Turku, Finland
[3] Department of Electronics, KTH Royal Institute of Technology, Sweden
Emails: [1]{qingqingli16, zhuo}@fudan.edu.cn, [1]{jopequ, tunggi, tovewe}@utu.fi, [3]hannu@kth.se

***Abstract*—A recent trend in the IoT is to shift from traditional cloud-centric applications towards more distributed approaches embracing the fog and edge computing paradigms. In autonomous robots and vehicles, much research has been put into the potential of offloading computationally intensive tasks to cloud computing. Visual odometry is a common example, as real-time analysis of one or multiple video feeds requires significant on-board computation. If this operations are offloaded, then the on-board hardware can be simplified, and the battery life extended. In the case of self-driving cars, efficient offloading can significantly decrease the price of the hardware. Nonetheless, offloading to cloud computing compromises the system's latency and poses serious reliability issues. Visual odometry offloading requires streaming of video-feeds in real-time. In a multi-vehicle scenario, enabling efficient data compression without compromising performance can help save bandwidth and increase reliability.**

***Index Terms*—Odometry; VSLAM; Visual Odometry; Visual SLAM; Internet of Vehicles; IoV; Edge Computing;**

I. Introduction

Accurate localization is one of the key pillars behind full autonomy. It is also essential for wider types of advanced intelligent systems, including those related to human-robot interaction. In terms of self-driving cars, the future of autonomous vehicles is also the future of connected vehicles [1]. This will come under the umbrella of Internet of Everything (IoE) and, more concretely, the Internet of Vehicles (IoV) paradigms [2]. In these paradigms, all vehicles share data with each other in vehicle-to-vehicle (V2V) communication, and any entity with information that might affect its operation in vehicle-to-everything (V2X) communication.

In GNSS-denied environments, or in those applications where high accuracy is necessary, localization often relied on odometry information. Typical ways of obtaining odometry information is through lidars or cameras. Visual odometry with mono or stereo vision has been extensively studied and current state-of-the-art methods provide robust solutions for accurate localization in both indoors and outdoors scenarios.

As the first cars with self-driving capabilities are entering the market, a significant part of the vehicle production cost is in the hardware required to provide robust and reliable autonomous operation. V2V and V2X will be one of the key factors in obtaining the target in terms of reliability, road safety, traffic efficiency, and energy savings. If we combine this with intensive computational offloading to near infrastructure, there is potential for important savings in the on-board complexity of both hardware and software [3]. This can be implemented in vehicles with human-in-the-loop, where the operation can change to manual if required. Nonetheless, in any case, strict control must be maintained over the network load in order to ensure that the bandwidth available for each unit is enough to keep latency and delays within safe limits.

II. Related Work

To the best of our knowledge, previous works that have considered offloading visual odometry calculations have utilized cloud-centric architectures. Yun *et al.* proposed a cloud robotics platform named RSE-PF for distributed visual SLAM [4]. The authors reported round-trip latency of around 150ms. This, compared to state-of-the-art methods able of processing at 30 frames/second or more, might result in delays that limit the potential application scenarios. While the authors utilized websockets in order to save bandwidth compared to HTTP, they did not report on the maximum number of concurrent units that could be handled. Dey *et al.*, while still relying on cloud servers, also proposed offloading in a multi-tier edge+cloud setup [5]. The authors put a focus on finding the optimal offloading strategy to make best use of the different network layers. They formulated an integer linear programming problem and provided an initial approach for dynamically deciding on the best offloading decision, in which the network bandwidth was a variable. In contrast, we put the focus on studying what is the maximum bandwidth savings that we can obtain without sacrificing performance, while maintaining a reliable service with minimal latency.

III. Image Compression for Visual Odometry

Visual odometry (VO) is an estimation of camera motion method based on a series of sequential images. VO can be applied in a verity applications. The general idea is to calculate the position correspondences between the two views by finding some invariant features. In this work we utilize an approach to visual odometry consisting of 3D-2D correspondences: In this method, the transformation matrix is calculated using the Perspective-n-Point(PnP) method. Firstly, the features across two neighbor frames obtained by the camera are detected and matched. The best matching points will be obtained

TABLE I
COMPRESSION RATE AND PERFORMANCE IMPACT WITH DIFFERENT JPEG COMPRESSION TECHNIQUES.

	Bandwidth Savings	Accuracy loss
JPEG50	22%	0.1%
JPEG10	71%	1.2%

after incorrect matches are discarded. Then 3D points are obtained by triangulating. After that, we eliminate inaccurate 3D points twice and combined the optical flow method and feature matching method to find more accurate 3D-to-2D correspondences. Finally, camera pose will be solved through these correspondences by PnP algorithm.

A major challenge in offloading visual odometry is the amount of data that needs to be streamed over the network. Compared to 2D lidar data or IMU data, a continuous stream of images consumes significantly higher bandwidth. Therefore, if images can be compressed without compromising the algorithm's performance, we can increment the efficiency of the edge offloading scheme.

IV. Experiment and Results

We have analyzed how traditional JPEG compression affects the performance of visual offloading algorithms for different levels of image quality levels. We have utilized a subset of the TUM dataset [6]. The experiments have been run with a set of 3682 acquired over 186 seconds. We compare the errors produced as a consequence of compressing the images at 10% quality and 50% quality, which results in compressed sizes of 0.78 and 0.29 times the original image size, respectively, as shown in Table 1. In terms of localization accuracy, the cumulative error is around 10 times smaller in the case of the 50% quality compression. The total and cumulative errors, and the errors in each direction are shown in Figure 1. We can see that, in this case, the performance varies across dimensions. The most clear difference appears in the error in the z axis, where the difference between the two compression methods is evident. In general, we can conclude that we can utilize images with 50% quality without compromising accuracy in most applications, while a 10% quality can be utilized in situations where the the accuracy requirements are not so tight. It is worth mentioning that the drift in the localization error is continuous in both cases, with a mostly constant average error. Therefore, it these method should be combined with loop-closure techniques that ensure that the localization error can be reduced to near zero within certain intervals of time.

V. Conclusion and Future Work

In this paper, we present preliminary results on the study of how different degrees of image compression affect the performance of visual odometry algorithms. We have concluded that around 20% of the network bandwidth can be saved without compromising accuracy, while a slight reduction in accuracy can bring over 70% of network load reduction, enabling a more flexible scaling of the computational offloading scheme.

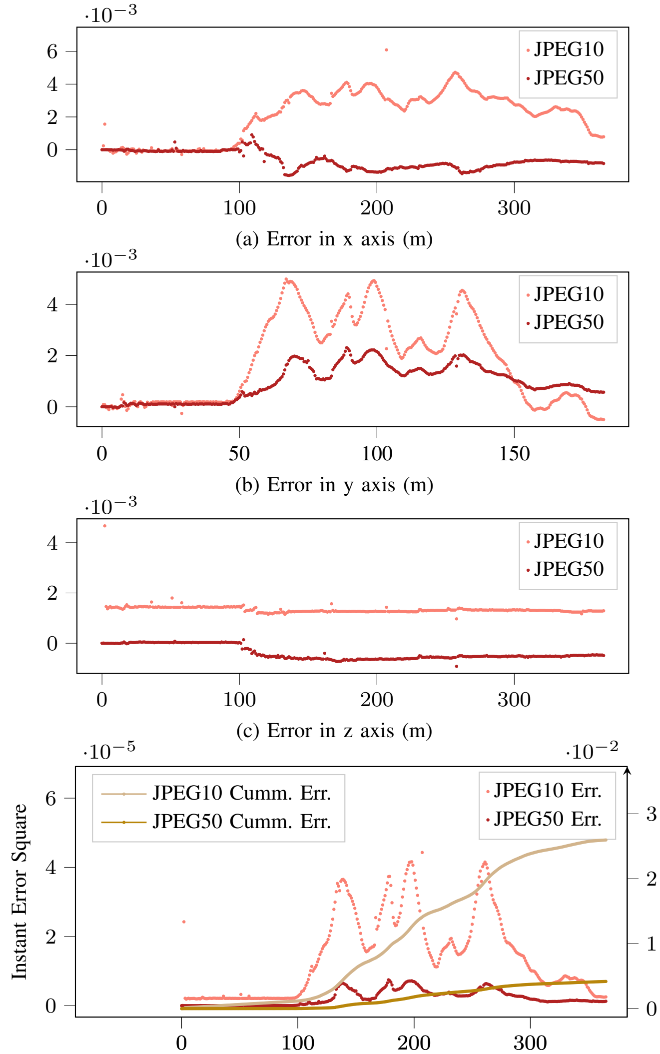

Fig. 1. Performance comparison of different compression rates.

In future work, we will study a wider range of image compression techniques, including ML-powered lossless compression techniques and measure network conditions.

Acknowledgment

This work has been supported by NFSC grant No. 61876039, and the Shanghai Platform for Neuromorphic and AI Chip (NeuHeilium).

References

[1] A. Bazzi *et al.* On the performance of ieee 802.11 p and lte-v2v for the cooperative awareness of connected vehicles. *IEEE Transactions on Vehicular Technology*, 66(11):10419–10432, 2017.
[2] O. Kaiwartya *et al.* Internet of vehicles: Motivation, layered architecture, network model, challenges, and future aspects. *IEEE Access*, 2016.
[3] V. K. Sarker *et al.* Offloading slam for indoor mobile robots with edge-fog-cloud computing. In *ICASERT, IEEE*, 2019.
[4] P. Yun *et al.* Towards a cloud robotics platform for distributed visual slam. In *Computer Vision Systems*. Springer, 2017.
[5] S. Dey *et al.* Robotic slam: A review from fog computing and mobile edge computing perspective. In *MOBIQUITOUS*. ACM, 2016.
[6] J. Sturm *et al.* Evaluating egomotion and structure-from-motion approaches using the TUM RGB-D benchmark. In *CDCFR, IROS*, 2012.