Multiple Scale Aggregation with Patch Multiplexing for High-speed Inter-vehicle Distance Estimation

Masahiro Hirano[1], Yuji Yamakawa[1], Taku Senoo[2], Norimasa Kishi[1] and Masatoshi Ishikawa[3]

***Abstract*— We propose an accurate and robust inter-vehicle distance estimation method using high-speed stereo vision. The framework involves two phases: a tracking phase, wherein a preceding vehicle is accurately and stably tracked by a tracking algorithm optimized for stereo high-speed vision, and a distance estimation phase, wherein the inter-vehicle distance is estimated via a highly accurate scale estimation and aggregation method for multiple scale-based distance estimations to ensure that it is more accurate and robust without introducing a delay. Further, we propose patch multiplexing to realize accurate and efficient aggregation even in situations where the scale changes rapidly (e.g., emergency braking). Through comparative analysis using three real-world scenarios, we verify that the accuracy of inter-vehicle distance estimation using our approach is comparable to that of laser rangefinders. We also demonstrate that differential quantities, such as velocity and acceleration, could be accurately estimated using an adaptive Kalman filter. Our results will help develop safe and accurate truck platooning and adaptive cruise control systems.**

I. INTRODUCTION

In this study, we investigated the implications of high-speed cameras on intelligent vehicles. Cameras are a crucial sensor in autonomous driving and advanced driver-assistance systems; they help in following preceding vehicles, truck platooning, and lane keeping assist systems. Millimeter-wave radars and 3D LiDARs are generally employed for distance estimation in truck platooning owing to their high accuracy and robustness against environmental disturbances. However, these are bulky and expensive techniques, and the amount of information retrieved is limited. Thus, cameras are generally used in combination with radars and LiDARs [1]. To enhance the capability of a camera, hardware components such as short-wavelength infrared cameras have been actively developed to overcome the vulnerabilities of cameras to weather conditions. In such systems, an accurate and robust camera-based inter-vehicle distance estimation method is desired. Furthermore, highly accurate estimation of velocity and acceleration is required for advanced adaptive cruise control. Accordingly, significant research has been conducted to improve the accuracy of estimation. In this study, we hypothesize that high-speed cameras can solve these issues and propose a highly accurate and robust method of inter-vehicle distance estimation using high-speed stereo vision, as illustrated in Fig. 1.

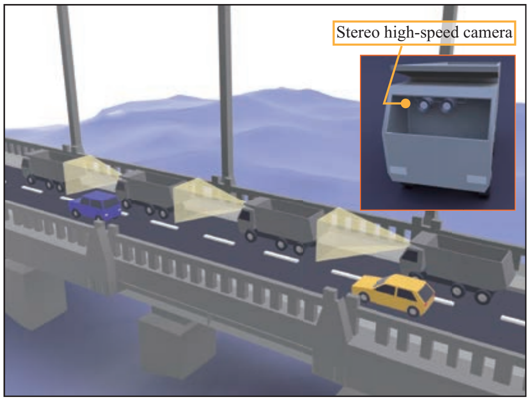

Fig. 1. On-board high-speed stereo vision system for truck platooning.

To measure the inter-vehicle distance using a stereo camera pair, it is necessary to accurately track the preceding vehicle with both cameras. However, sudden changes in the position of the preceding vehicle in camera images due to vibration or emergency braking can lead to inaccurate tracking and failure in establishing correspondence between the left and right cameras. We used a high-speed stereo vision system to alleviate such vibrations because it acquires images in short time intervals, which reduces the image motion between frames. The proposed method tracks preceding vehicles using the high-speed stereo vision system by alternately applying correlation filters to the left and right cameras with several customizations.

Our method can estimate distance via triangulation; however, it is not robust against estimation failures and not accurate enough to precisely estimate the velocity and acceleration. We address this issue by leveraging multiple distance estimation methods. A simple moving average for multiple estimation results is often used in such cases, but it causes non-negligible delay. Thus, we used a high-precision real-time scale estimation method based on the triangulated distance estimations of previous frames and performed aggregation without introducing delay. We also propose a method of multiplexing

[1] Masahiro Hirano, Yuji Yamakawa, and Norimasa Kishi are with the Institute of Industrial Science, The University of Tokyo, Tokyo, Japan. mhirano@iis.u-tokyo.ac.jp

[2] Taku Senoo is with the Graduate School of Advanced Science and Engineering, Hiroshima University, Hiroshima, Japan.

[3] Masatoshi Ishikawa is with the Information Technology Center, The University of Tokyo, Tokyo, Japan.

patches used for scale estimation to ensure accuracy even in situations where the scale changes rapidly, such as emergency braking. Through comparative analysis against a high-speed laser rangefinder (LRF), we validated the proposed method in various real-world highway scenarios, such as typical vehicle following, sudden throttle, braking, and intensive vibration.

The main contributions of this work can be summarized as follows.

- We proposed to aggregate multiple distance estimations without introducing delay via accurate scaling and patch multiplexing for efficient and stable aggregation.
- We built a stereo high-speed vision system, validated the proposed method via on-road experiments, and demonstrated that the accuracy of estimating the distance, velocity, and acceleration was comparable to that of LRFs.

II. Related Work

A. *Tracking*

Object tracking has attracted significant research interest in computer vision. Two primary approaches employed include correlation filter-based and deep learning-based methods. Correlation filter-based methods (e.g., MOSSE [2] and KCF [3]) track objects by continuously updating filters that output object locations by multiplying to the input images in the frequency domain. Recent deep learning-based approaches (e.g., SiamRPN [4] and DaSiamRPN [5]) utilize the Siamese network architecture, in which the same network is applied to both the input and target images, and the resulting feature maps are correlated to estimate object locations. Correlation filters learn the appearance of objects online and they are fast. This motivates us to optimize these filters for high-speed stereo vision to accurately and robustly track preceding vehicles.

B. *Distance Estimation*

Computer vision-based inter-vehicle distance estimation methods can be categorized into two groups, i.e., monocular and stereo. Several cost-effective monocular vision systems have recently been proposed. In [6], an accurate depth prediction method was proposed for dynamic scenarios, such as getting surrounded by moving vehicles. As a more vehicle-oriented method, a recent work [7] proposed a light-weight depth and motion estimation method that directly regresses the vehicle velocity from the corresponding trajectories using a multilayer perceptron. End-to-end learning has also been employed [8], [9] to improve the robustness and estimation accuracy. However, it has been shown that the difference between monocular and stereo depth estimation accuracy is significant [10] owing to scale ambiguity, which is a fundamental limitation of monocular vision.

Some stereo-based methods extract visual cues, such as edge information [11], LED [12], and UV disparity maps [13], but they only use one pair of images for distance estimation. Thus, the sequential time information is not well-utilized. By leveraging stable and accurate high-speed vision tracking, we propose a novel multiple aggregation architecture to significantly improve the estimation accuracy.

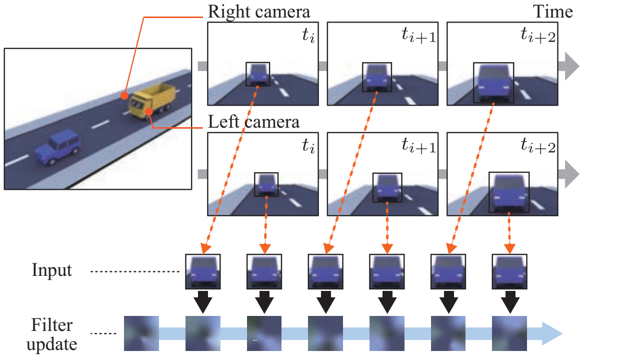

Fig. 2. Outline of alternating MOSSE for preceding vehicle tracking. The same correlation filter is alternately applied to high-speed stereo image sequences.

III. Alternating MOSSE for Stereo High-Speed Visual Tracking

A. *Preceding Vehicle Identification*

The first step in our method involves identifying the preceding vehicle with both cameras to estimate the distance accurately. Objects are recognized by deep learning-based object recognition methods, which provide the object location and labels. We employed the YOLOv3 [14] object recognition algorithm, wherein object locations are represented as bounding boxes enclosing objects. We assume that the preceding vehicle and ego-vehicle run in the same lane in the beginning of the estimation. Thus, in the first frames of both cameras, we identify a bounding box as the preceding vehicle whose center is located in the middle of the current lane.

B. *Alternating MOSSE*

1) MOSSE: Here, we briefly discuss the minimum output sum of squared error (MOSSE) filter [2]. MOSSE constructs correlation filters to track objects; it is designed to generate 2D Gaussian maps with peaks indicating object locations in images. One of the advantages of MOSSE is that it is computationally inexpensive. It learns the appearance of target online by updating the filter with a constant learning rate; thus, it is robust to changes in the appearance.

2) Alternating MOSSE for high-speed stereo vision: Based on MOSSE, we adopted a tracking method called "*alternating MOSSE (A-MOSSE)*," tailored for high-speed stereo vision (originally proposed in [15]). Fig. 2 shows an overview of the method. Three major changes were implemented, as listed below:

Filter sharing: The same MOSSE filter is applied to images from both cameras and updated alternately while developing the bounding box of each camera independently. This technique enables long-term robust stereo tracking because it preserves consistency between the bounding boxes of both cameras.

Bounding box scaling: The preceding vehicle is identified as a bounding box for each camera. After determining the geometric configuration of the camera, the centers of both bounding boxes can be triangulated to estimate the approximate inter-vehicle distance. The size of the bounding box is continuously scaled in response to the inter-vehicle distance. The portion of the preceding vehicle occupied in the bounding box is kept constant, thus ensuring robustness of tracking against varying distance ranges. To avoid additional computations due to scaling, we resize the image within the bounding box to a fixed size.

Learning rate adjustment: Because we use a high-speed camera, the changes in appearance of the tracking target are minimal in a short time interval between frame acquisition. We set a low value for the filter learning rate, η, which suppresses drift in tracking. In our experiments, we set the learning rate, η, to be 0.0004 for 500 [fps], whereas η is typically set to 0.02 for 30 [fps].

3) Bootstrapping: The bounding boxes in both cameras are not always well aligned when tracking starts because each bounding box is set independently. To suppress the initialization error induced by the misalignment in registration and improve robustness, we modified A-MOSSE to run twice for the first image pair.

IV. MULTIPLE SCALE AGGREGATION WITH PATCH MULTIPLEXING

A. Overview

We propose multiple scale aggregation to improve the estimation accuracy; in this approach, multiple distance estimations are aggregated via statistical treatment for robustness against outliers. Because a change in the apparent size (scale) of the preceding vehicle encodes a change in the inter-vehicle distance, the distance can be calculated through distance estimation via triangulation for a previously acquired frame pair and the scale change of the preceding vehicle between the previous and current frames. The scale change can be computed using a pair of square patches generated from bounding boxes. Here, the frame containing the reference patch for calculating the scale is called a keyframe. We propose to aggregate multiple distance estimations by scaling to improve the accuracy and robustness. We further propose to multiplex patches of different sizes at the keyframes to perform aggregation more effectively even when the patch size of the current frame dynamically changes. The estimated inter-vehicle distance is highly accurate to estimate the velocity and acceleration using an adaptive cubic Kalman filter.

This architecture enables asynchronous processing, in which the processing for each keyframe is completed before estimation. This reduces the computational cost in real-time processing. We discuss each key component in the following sections.

B. Scale Estimation

The inter-vehicle distance, $\tilde{d}$, is computed using the scale change, s, of the preceding vehicle from an arbitrary keyframe to the current frame as follows:

$$\tilde{d} = d/s, \tag{1}$$

where d is the distance obtained by triangulation at the keyframe containing the reference patch.

Running vehicles inevitably suffer from vibrations due to irregularities in the road surface and engine operation. Such vibrations result in translational motion on images. The power spectra for each image are calculated to remove the effect of this translation. Subsequently, the scale change is converted to the translation of the image by applying the Fourier-Mellin transform [16]. It is calculated with significantly high accuracy using the phase-only correlation (POC) method [17], in which the normalized cross spectrum of two images is calculated by eliminating amplitudes and focusing on the phase component through discrete Fourier transform (DFT). The inverse Fourier transform of the normalized cross spectrum exhibits a distinct peak corresponding to the scale change. We used the windowing technique [17] to mitigate the effect of discontinuities around boundaries for better estimation. In addition, we can ignore the inverse Fourier transform on pixels at a significant distance from the peak location because the scale change is approximate, resulting in the reduction of computational cost.

POC involves DFT; to perform DFT efficiently, a square patch enclosing the bounding box is set. Let $n_{b_X} \times n_{b_Y}$ be the size of the bounding box. Then, the patch size, n_p, is given by

$$n_p = \min \left\{ n | n \in N_d,\ n \geq n_{b_X},\ n \geq n_{b_Y} \right\}, \tag{2}$$

where $N_d = \{n_{d_i} | i = 1, 2, \ldots\}$ is a predetermined set of optimal DFT sizes. The center of the patch corresponds to that of the bounding box.

Because the scale calculated by this method and A-MOSSE is robust against vibrations, the calculated inter-vehicle distance is also robust against vibrations.

C. Multiple Scale Aggregation

Because scale estimation by the aforementioned method may fail owning to sudden changes in lighting, distance estimation by a single keyframe lacks reliability. To address this issue and reduce variance (noise) in a single estimation, we aggregate the distances by the scaling calculated for multiple keyframes via statistical treatment to obtain a more robust and accurate estimation. This aggregation does not introduce phase lag because all distance estimations are based on the same moment.

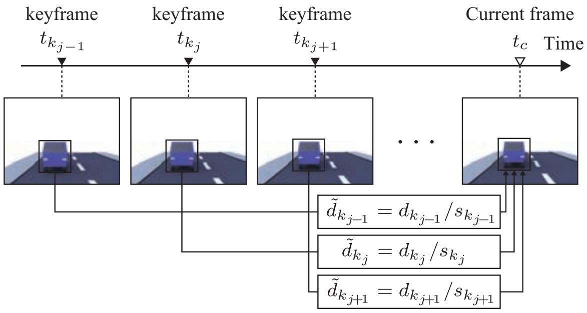
keyframe
$t_{k_{j-1}}$
keyframe
t_{k_j}
keyframe
$t_{k_{j+1}}$
Current frame
t_c
Time
· · ·
$\tilde{d}_{k_{j-1}} = d_{k_{j-1}}/s_{k_{j-1}}$
$\tilde{d}_{k_j} = d_{k_j}/s_{k_j}$
$\tilde{d}_{k_{j+1}} = d_{k_{j+1}}/s_{k_{j+1}}$

Fig. 3. Multiple scale aggregation. Distance estimation at current time, t_c, is derived by an aggregation of multiple scale-based distance estimations.

Let M be the patch size in the current frame and K be the set of keyframes with a patch size of M. Let d_{k_j} be the distance estimation by triangulation at keyframe $k_j \in K$ and s_{k_j} be the scale obtained from the patch in k_j and that in the current frame. We obtain a distance estimation, $\tilde{d}_{k_j} = d_{k_j}/s_{k_j}$, for each keyframe, as shown in Fig. 3.

To reject outliers differing by more than 10% in $\{\tilde{d}_{k_j}\}$, results that are significantly different from the distances predicted by the Kalman filter (Section IV-E) are rejected. Then, we obtain a set of keyframes, I, associated with the inliers. The estimated distance by multiple scale aggregation, $\tilde{d}$, is

$$\tilde{d} = \frac{1}{\mathcal{N}(\boldsymbol{I})} \sum_{k_j \in \boldsymbol{I}} \frac{d_{k_j}}{s_{k_j}}, \tag{3}$$

where $\mathcal{N}(I)$ is the number of estimations in I.

However, calculating the distances for multiple keyframes using the aforementioned method is computationally expensive. Therefore, we preprocess a part of scale estimation with keyframes in advance. Specifically, when a keyframe is selected, we preprocess an image retained after applying the Fourier-Mellin transform to the power spectrum of the patches, and then apply the DFT. This preprocessing for keyframe patches is conducted asynchronously and they are aggregated immediately after processing. It should be noted that the aggregation is highly parallelizable because each scale computation can be calculated independently, thus making the computation more efficient.

D. Patch Multiplexing

In the method discussed previously, scale aggregation can be conducted only for keyframe patches with the same size as that of the patch in the current frame. However, if the patch size in the current frame changes owing to the preceding vehicle approaching or leaving, a risk may arise wherein few keyframes may attain the same patch size, thereby limiting the effect of scale aggregation. Therefore, we multiplex patches one size larger and smaller for each keyframe based on the reference patch.

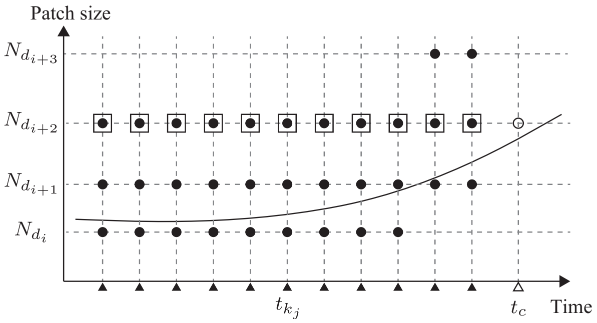

Fig. 4. Multiple scale aggregation with patch multiplexing. The curve denotes the size of the bounding box. The current frame is indicated by △ and the current patch is indicated by ○. The black circles, ●, represent the patches created in patch multiplexing at keyframes ▲, and patches marked by □ are aggregated for distance estimation at t_c.

Fig. 4 summarizes multiple scale aggregation with patch multiplexing. For a keyframe at time t_{k_j}, three patches with different sizes, $N_{d_i}, N_{d_{i+1}}$, and $N_{d_{i+2}}$, are prepared, where $N_{d_{i+1}}$ is the reference patch size. We assume that in the current frame at time t_c, a patch of size $N_{d_{i+2}}$ is calculated. Owing to patch multiplexing, patches with size $N_{d_{i+2}}$ at the keyframes can be aggregated even if the reference patches in the keyframes are different.

This method ensures that keyframes can be used for aggregation, even when the inter-vehicle distance changes. Although this multiplexing increases the computation time required for the keyframes, it does not affect the throughput performance. This is because keyframe processing can be executed asynchronously. In addition, the scale estimation method introduced in the previous section loses accuracy when the scale exceeds a certain value, but multiplexing avoids this problem by ensuring that patch pairs with significantly different scales are rejected.

E. Filtering

The relative velocity and acceleration are estimated from the inter-vehicle distance using a cubic Kalman filter [18]. Let x_k be the distance estimated at time k and dt be the frame acquisition interval. The cubic Kalman filter is then formulated as

$$\begin{bmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \end{bmatrix} = \begin{bmatrix} 1 & \mathrm{dt} & \mathrm{dt}^2/2 \\ 0 & 1 & \mathrm{dt} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k-1} \\ \dot{x}_{k-1} \\ \ddot{x}_{k-1} \end{bmatrix} + \begin{bmatrix} \mathrm{dt}^3/6 \\ \mathrm{dt}^2/2 \\ \mathrm{dt} \end{bmatrix} w_k, \tag{4}$$

$$z_k = x_k + v_k, \tag{5}$$

where the process noise, w_k, and observation noise, v_k, follow the Gaussian distributions, $w_k \sim \mathcal{N}(0, Q)$ and $v_k \sim \mathcal{N}(0, R)$, respectively. The covariance matrix Q is modeled as

$$Q = q \begin{bmatrix} \mathrm{dt}^5/20 & \mathrm{dt}^4/8 & \mathrm{dt}^3/6 \\ \mathrm{dt}^4/8 & \mathrm{dt}^3/3 & \mathrm{dt}^2/2 \\ \mathrm{dt}^3/6 & \mathrm{dt}^2/2 & \mathrm{dt} \end{bmatrix}. \tag{6}$$

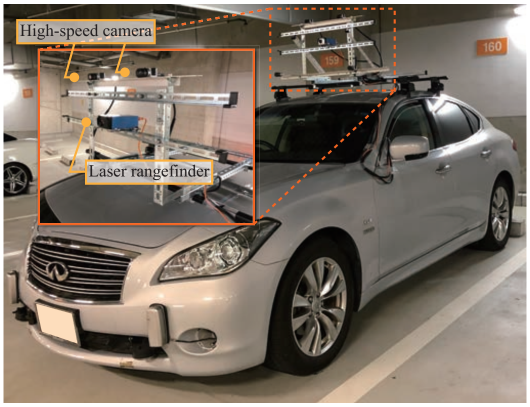

Fig. 5. System setup.

Because the distance estimation accuracy is inversely proportional to the inter-vehicle distance, the covariance of the observation noise, R, depends quadratically on the estimated distance, $\tilde{d}$:

$$R = \begin{cases} R_{\min} & d_{\min} \leq \tilde{d} \\ \frac{R_{\max}-R_{\min}}{(d_{\max}-d_{\min})^2}(\tilde{d}-d_{\min})^2 + R_{\min} & d_{\min} \leq \tilde{d} \leq d_{\max} \\ R_{\max} & d_{\max} \leq \tilde{d}, \end{cases} \tag{7}$$

where $R_{\min}$ and $R_{\max}$ are the lower and upper limits of R, respectively. R is clipped between a minimum distance, $d_{\min}$, and maximum distance, $d_{\max}$.

V. EVALUATION

A. Experimental Setup

Because our method is based on high-speed cameras, the proposed model cannot be evaluated on datasets with standard frame rate videos. Therefore, we constructed a dedicated experimental setup and collected data for evaluation. Fig. 5 shows the experimental setup. We used two industrial high-speed cameras, acA800-510uc, manufactured by Basler Inc. (resolution: 800×600 px, frame rate: 500 fps) with a wide lens (F1.8/6 mm). These cameras were synchronized and mounted in alignment with both optical axes parallel to the vehicle rooftop (height: 2060 mm, baseline: 550 mm). They were carefully calibrated using MATLAB's computer vision toolbox. The pitch and roll angles were 4.6° and 1.3°, respectively. For a comparative study, we mounted an LRF, LDM71, manufactured by JENOPTIK AG (precision: 20 cm at 20 m, frequency: 500 Hz, max: 270 m) with the cameras. The LRF was attached 95 mm ahead of the camera image plane.

We set keyframes every 0.05 s and the maximum number of scale aggregation was 100, implying that keyframes set at time t were disregarded from aggregation after time $t+5$. This also indicates that multiple scale aggregation is fully set after 5 s. We compared the proposed system with the LRF. The cubic Kalman filter presented in IV-E was applied for distance estimation using the proposed system and LRF to compare the velocity and acceleration. In the experiments, we set $R_{\min} = 10^6$, $R_{\max} = 10^8$, $d_{\min} = 15$, and $d_{\max} = 120$. Additionally, we set $q = 10^7$; an offline finite impulse response (FIR) differentiator filter was applied to LRF measurements to determine lag-free velocity and acceleration estimation. To compare our estimation $x(t)$ with $y(t)$ corresponding to the LRF, we evaluated the mean absolute error (MAE) as follows:

$$\text{MAE} = \frac{1}{T}\sum_{t=1}^{T} |x(t) - y(t)|. \tag{8}$$

All calculations were executed on an Ubuntu computer with Intel® Xeon® processor (6 cores, 2 CPU, 3.4 GHz) and 48 GB RAM. We used OpenMP® for parallelization.

We conducted experiments in three different highway scenarios; the results are presented in the supplemental video.

B. Evaluated Scenarios and Results

1) Approaching and leaving at curve: We conducted an experiment considering truck platooning on a highway. We recorded a scene of a truck slowly approaching and leaving from the ego-vehicle, while maintaining an almost constant distance for approximately 40 s. In the first part of the scene, the car moves straight, then enters a curve toward the middle of the scene. The tracking result is shown in Fig. 6-(a). After the preceding truck enters the lane 20 m ahead of the ego-vehicle from the neighboring lane, tracking was considered accurate and successful, even though the preceding truck steered left and right in the lane and tilted slightly to the right in the middle of the curve.

Compared with the LRF, the proposed method estimates the distance, velocity, and acceleration with high accuracy, as shown in Fig. 7 (detailed analysis can be found in Section V-B.4). Before completely setting multiple scale aggregation at 5 s, the estimations were not sufficiently accurate. Note that the proposed method measures the distance from the camera to the vehicle center, whereas LRF measures the distance from the laser source to the illuminated point. This leads to a small difference between the two estimations.

In addition, Fig. 8 shows the comparison between the case where the triangulation by A-MOSSE is directly used as the distance measurement result and that where multiple scale aggregation is used to improve the accuracy. Because distance estimation by triangulation involved more noise compared to that with multiple scale aggregation, the velocity and acceleration estimated by triangulation were significantly noisier. This is because it is difficult to estimate the velocity and acceleration accurately if distance estimation is not highly accurate because these are differential values. This shows that the proposed multiple scale aggregation method significantly improves the estimation accuracy.

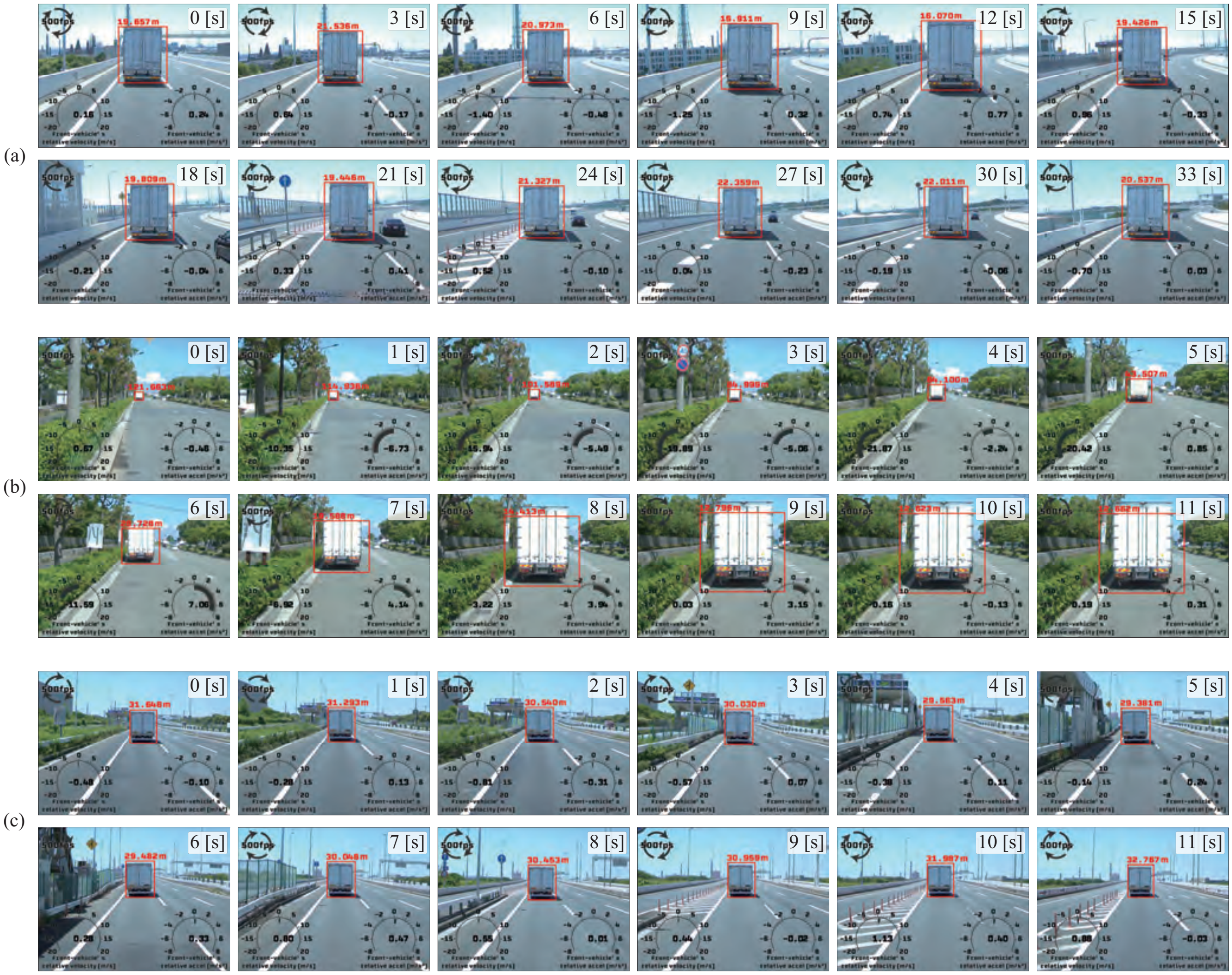

Fig. 6. Tracking results of (a) approaching and leaving, (b) sudden throttle and braking, and (c) intensive vibration. The red rectangle denotes the bounding box in tracking using A-MOSSE. The bounding box is labeled with the estimated distance. The relative velocity and acceleration are shown at the bottom left and right, respectively. All results can be found in the supplemental video.

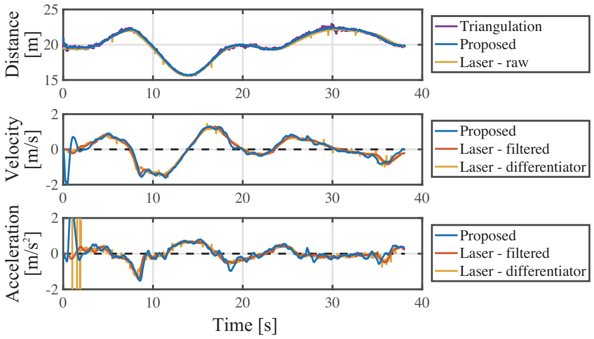

Fig. 7. Estimated results for "approaching and leaving."

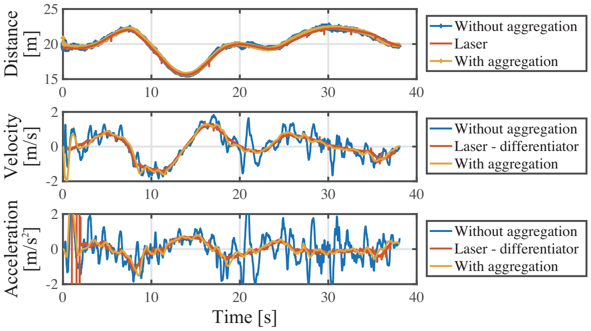

Fig. 8. Estimated results with and without multiple scale aggregation.

2) Sudden throttle and braking: Another experiment was conducted in a scene where the vehicle first accelerated and braked sharply to a stop (Fig. 6-(b)). We started recording the 13 s-scene when the distance to the stationary truck in the sight exceeded 120 m, and stopped when the distance approached 12 m in approximately 10 s. Although the position of the preceding truck changed significantly in the image owing to the up and down motion of the ego-vehicle when accelerating and

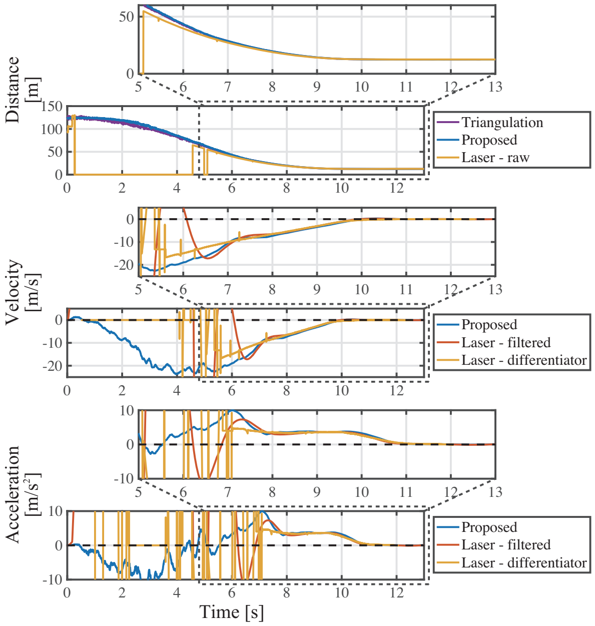

Fig. 9. Estimated results for “sudden throttle and braking.”

braking, accurate vehicle tracking was realized.

The advantage of high-speed vision is demonstrated when the tracked object moves significantly in the image in a short period. We compared the time of continuous successful tracking with a simulated low frame rate camera by thinning out the frames in the same scene (Table I). When the frame rate was less than 80 FPS, tracking failed at an early stage. However, continuous tracking was successful when the frame rate was higher than 100 FPS. In real-world scenarios, the effect of vehicular vibration on the measurement using the onboard camera is significant. This result shows the importance of using a high-speed camera as an onboard camera.

Fig. 9 shows the estimation results for this scene. Owing to patch multiplexing, multiple scale aggregation can be beneficial even when the inter-vehicle distance dynamically changes. The proposed method could estimate the distance in all ranges; however, the LRF failed to measure the distance because it was difficult to continuously irradiate the laser on the truck owing to car movements. This is also observed in LiDAR, which makes it difficult to dynamically track targets.

Because LRF could measure the distance after 8 s, we compared the estimated values during that period. A systematic error was observed in distance estimation; however, this effect was limited for the velocity and acceleration. This result demonstrates that the proposed method is effective for emergency brake assist.

TABLE I

Tracking suspended time in different frame rates (FPS). The faster the FPS, the more stable the tracking.

Frame rate [fps]	50	60	80	100	250	500
Time [s]	0.46	2.38	1.32	13.0	13.0	13.0

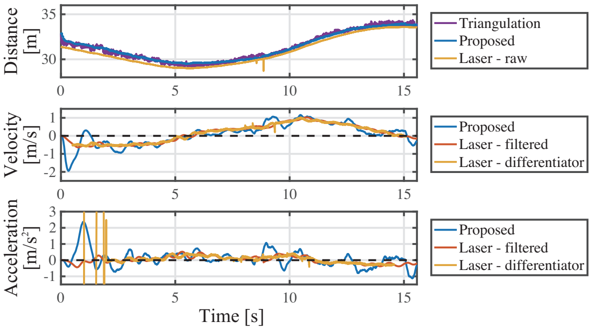

Fig. 10. Estimated results for “intensive vibration.”

3) Intensive vibration with slope: We conducted a third experiment on a highway where the preceding and ego-vehicles were vibrating intensively owing to irregularities in the road (Fig. 6-(c)). The road was levelled, but an incline was introduced toward the middle of the scene. Even though the vehicles were vibrating and the slope was changing in the middle of the scene, stable tracking was realized.

The estimated result is presented in Fig. 10. Although the image exhibits severe vibration, we estimated the distance with an accuracy of less than 50 cm at a distance of 30 m including systematic errors, compared to LRF. However, the distance estimation result changed relatively steeply around the 9 s mark when the vehicle entered the uphill slope; moreover, a large difference in velocity and acceleration was observed in comparison to LRF measurements. At this instance, the vehicle approached the uphill slope. This suggests that the changes in vehicle posture with respect to road surface caused by the vehicle entering the uphill slope may have influenced the results, which is an issue to be addressed in future studies.

4) Parameter Study: We examined the estimation results for different numbers of patches in multiple scale aggregation. We used the approaching and leaving data described in Section V-B.1 for evaluation. The number of patches affects the accuracy of estimation and computation time. The estimation results for different numbers of patches (2, 20, and 100) are shown in Fig. 11. The computation time was 4.7 and 9.96 ms when the number of patches was 2 and 100, respectively. The computation was aided by parallelization in aggregation. Because a large portion of computation involves DFT, we can further reduce the computation time by leveraging a GPU. The difference between the distance estimates and LRF measurements was 0.16 and 0.14 m for patch number 2 and 100, respectively, and both were small. In addition, the difference in velocity was 0.3 and 0.1 m/s for patch number 2 and 100, respectively. The difference was significantly smaller when the number of patches was 100. Similarly, for the acceleration, the higher the

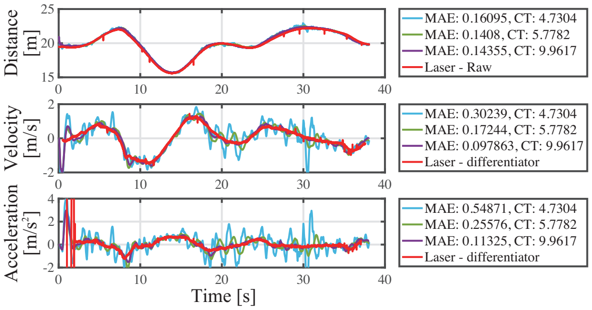

Fig. 11. Estimation results for different numbers of patches: 2 (light blue), 20 (green), and 100 (purple). Mean absolute error (MAE) and computation time (CT) are shown.

number of patches, the longer the computation time, with smaller differences with LRF measurements.

Fig. 12 shows a more detailed analysis of this relationship. For distance estimation, increasing the number of patches from 10 had minimum effect on the error. For velocity and acceleration, the estimation error continued to decrease until the number of patches was increased to 60. These results show that there is a trade-off between the computation time and estimation accuracy. The number of patches can be tuned according to the computational resources and required accuracy. Note that the proposed method can further be accelerated with GPU.

VI. Conclusion

Herein, we proposed a method for estimating highly accurate inter-vehicle distances using high-speed stereo vision cameras. We used A-MOSSE, a visual tracking algorithm optimized for high-speed stereo cameras, and showed that it realizes stable and accurate tracking even under severe vibrations. In addition, we proposed multiple scale aggregation that refined distance estimation without introducing delay through accurate scale estimation in combination with a specially designed method called patch multiplexing to perform stable aggregation even in situations where the scale changes rapidly. Through comparative analysis, we showed that the accuracy of inter-vehicle distance estimation using our approach is comparable to that of LRFs by evaluating differential quantities such as velocity and acceleration using an adaptive Kalman filter.

The velocity and acceleration estimated in this study were relative quantities with respect to the ego-vehicle. Future work directions include the estimation of absolute velocity and acceleration by combining them with the movement information of the ego-vehicle. This system is thus capable of making advanced decisions on acceleration and braking according to the movement of the preceding vehicle; therefore, it contributes to advanced camera-based driver-assistance systems and autonomous driving. We thus demonstrated the advantages of replacing on-board stereo cameras with high-speed stereo cameras, thereby potentially enabling the implementation of stereo on-board high-speed vision.

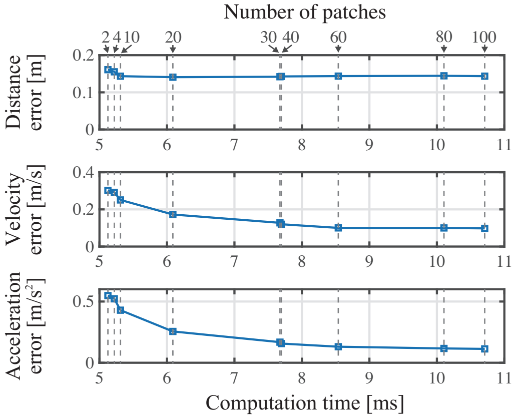

Fig. 12. Mean absolute error and computation time for different number of patches (2, 4, 10, 20, 30, 40, 60, 80, and 100).

References

[1] Z. Liu, *et al.*, "Robust target recognition and tracking of self-driving cars with radar and camera information fusion under severe weather conditions," *IEEE T-ITS*, pp. 1–14, 2021.

[2] D. S. Bolme, *et al.*, "Visual object tracking using adaptive correlation filters," in *IEEE CVPR*, 2010, pp. 2544–2550.

[3] J. F. Henriques, *et al.*, "High-speed tracking with kernelized correlation filters," *IEEE T-PAMI*, vol. 37, no. 3, pp. 583–596, 2015.

[4] B. Li, *et al.*, "High performance visual tracking with siamese region proposal network," in *IEEE CVPR*, 2018, pp. 8971–8980.

[5] Z. Zhu, *et al.*, "Distractor-aware siamese networks for visual object tracking," in *ECCV*, vol. 11213, 2018, pp. 103–119.

[6] H. Li, *et al.*, "Unsupervised monocular depth learning in dynamic scenes," CoRR, abs/2010.16404, 2020.

[7] M. Kampelmühler, *et al.*, "Camera-based vehicle velocity estimation from monocular video," CoRR, abs/1802.07094, 2018.

[8] T. Zhe, *et al.*, "Inter-vehicle distance estimation method based on monocular vision using 3d detection," *IEEE T-VT*, vol. 69, no. 5, pp. 4907–4919, 2020.

[9] Z. Song, *et al.*, "End-to-end learning for inter-vehicle distance and relative velocity estimation in ADAS with a monocular camera," in *IEEE ICRA*, 2020, pp. 11 081–11 087.

[10] N. Smolyanskiy, *et al.*, "On the importance of stereo for accurate depth estimation: An efficient semi-supervised deep neural network approach," in *IEEE CVPRW*, 2018, pp. 1007–1015.

[11] V. D. Nguyen, *et al.*, "A fast evolutionary algorithm for real-time vehicle detection," *IEEE T-VT*, vol. 62, no. 6, pp. 2453–2468, 2013.

[12] V. T. B. Tram and M. Yoo, "Vehicle-to-vehicle distance estimation using a low-resolution camera based on visible light communications," *IEEE Access*, vol. 6, pp. 4521–4527, 2018.

[13] J. Leng, *et al.*, "Robust obstacle detection and recognition for driver assistance systems," *IEEE T-ITS*, vol. 21, no. 4, pp. 1560–1571, 2020.

[14] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," CoRR, abs/1804.02767, 2018.

[15] T. Senoo, *et al.*, "Measurement of distance and velocity for overlapped vehicles using high-speed stereo vision," in *SICE SI*, 2019, pp. 2200–2202.

[16] Q. Chen, *et al.*, "Symmetric phase-only matched filtering of Fourier-Mellin transforms for image registration and recognition," *IEEE T-PAMI*, vol. 16, no. 12, pp. 1156–1168, 1994.

[17] K. Takita, *et al.*, "High-accuracy subpixel image registration based on phase-only correlation," *IEICE Trans. Fundamentals*, vol. 86, no. 8, pp. 1925–1934, 2003.

[18] L. Puglisi, *et al.*, "On the velocity and acceleration estimation from discrete time-position sensors," *CEAI*, vol. 17, pp. 30–40, 10 2015.