Platform of Collecting On-vehicle Video Images and Protection of Personal Information for Data Utilization

Yasutaka OKADA
Ryusuke SEKI
Kimitaka MURASHITA
Hiroshi YAMADA

Abstract

Recently, there have been many developments related to the Internet of Things (IoT) in the automotive sector. Cars are continuously connected to the cloud through on-vehicle devices, which in turn has led to the connected car sosiety which creates a new service and business. In the world of Cyber Physical System (CPS), huge data in the physical space is collected by connected cars, and converted into knowledge in the center, which creates new added values. The utilization of data collected by on-vehicle devices which are the source of this value creation is expected.

A video image around the vehicle, as one of the data, is captured by the on-vehicle cameras and may include personal information such as the face of pedestrians and number plates of vehicles. The General Data Protection Regulation (GDPR) is in force in the EU for the protection of such personal information.

As the video image is non-structured data, it is difficult to be utilized in its original form. Therfore video analysis requires tagging to extract information contained in the video. Furthermore, it is important to implement both of sophistication of the tagging process and the protection of personal information so that the video image can be utilized in a variety of services. In order to achive this objective, it is necessary to equip the on-vehicle devices with object detection technology, which can tag the collected images for identifying them and define the area of personal information protection by utilizing the tagged information.

This study reports the feasibility of object detection processing by on-vehicle devices and the application possibility of personal information protection.

1. Introduction

As the Internet of Things (IoT) becomes more popular in the automotive sector, we are shifting to a connected vehicle society in which new services and businesses are being created through the continual connection of on-vehicle devices with the cloud. In a cyber physical system (CPS) where connected vehicles are active, information on the physical space is collected by connected vehicles (on-vehicle devices), and converted into knowledge using a large-scale data processing technology and others in cyber space (center). Thus, useful newly discovered information and knowledge will create new services offered to users through on-vehicle devices (**Fig. 1**). To create new services, data collected by on-vehicle devices is expected to be utilized, which may create additional value as a significant source.

By contrast, although the utilization of video images around the vehicle is expected, video images may also contain personal information such as pedestrian faces and the license plates of other vehicles. The General Data Protection Regulation (GDPR) became effective in 2018 in the European Union (EU). Also in Japan to reflect the requirements specified in the GDPR based on the Private Information Protection Law, a revision of the law has been discussed and personal information protection has been socially enhanced.

Because a video image is made up of non-structured data, it is difficult to use in its original form. Therefore, the analysis of a video image requires tagging to extract information contained in the video. For application to various services, it is important to satisfy both sophisticated tagging and personal information protection. To do so, it is necessary to apply object detection technology through an on-vehicle device, which can attach tags to collected images for identification and determine areas of personal information protection based on the tagged information.

This article discusses and reports the feasibility of object detection processing using on-vehicle devices and the applicability of personal information protection.

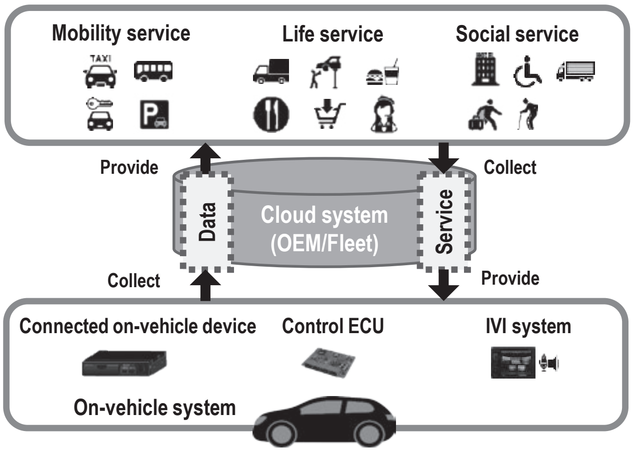


Fig. 1 Cyber Physical System (CPS)

2. Data Collection System in Connected Vehicles

To implement the system shown in **Fig. 1**, it is necessary to handle an enormous amount of on-vehicle data collected by a few hundred to tens of millions of connected vehicles. It causes problems as follows.

- Enormous cost for sending data to a data center
- A long time and much cost for data processing at the data center

In addition, the collected on-vehicle data also contain data unnecessary for service providers (called servicers) who use the data. For this reason, as a system for effectively collecting data from on-vehicle devices, we have been developing a system called On-demand Data Collection **(Fig. 2)**.

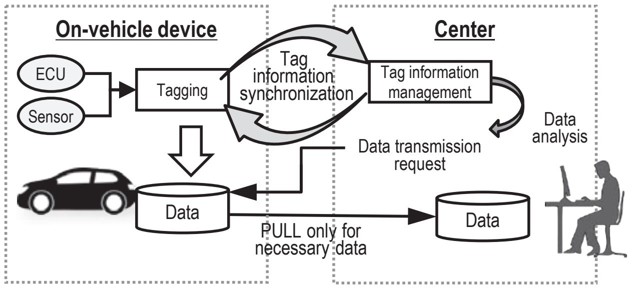


Fig. 2 On-demand Data Collection

With On-demand Data Collection, the system detects events from vehicle data, such as the vehicle speed, steering angle, vehicle location, and time, and then attaches tags to them (e.g., a "sudden breaking" or "abrupt steering"). The system sends only the tag data to the data center, and the servicer searches for only necessary data from the tag data collected in the center and receives the data from on-vehicle devices. With this mechanism, data transmission to and data processing at the center are minimized, and only the data required for the servicer can be effectively collected.

Presently, we are developing the system aiming for commercial application to a business use driving recorder [1] with a communication function and a data logger [2] for telematics services supporting the management of vehicle operation. In services using these products,

- locations to be visualized where a sudden breaking or abrupt steering may lead to an accident,
- historical driving records of taxicab and bus drivers, and others are expected. Furthermore, those are needed to be utilized to the following services.

<Service Examples>

①The development of level-3 self-driving vehicles

②An information distribution service for subsequent vehicles

③Dynamic map creation and marketing

To implement the collection of data for such services, it is necessary to detect objects (things) in a physical space using an on-vehicle camera, and to then attach tags to them. **Fig. 3** shows examples of objects to be detected.

According to the current increase in personal information protection, it is also required to protect personal information contained in the collected data. To take measures to protect personal information, the areas are masked where it is judged that personal information (pedestrian faces, license plates, and others) is contained in a video image by applying personal information tags.

To satisfy both the implementing of the service described above ① to ③ and personal information protection, it is necessary to operate a multi-class object recognition system (hereafter, image AI) through on-vehicle devices, and to detect objects for tagging and areas for protecting personal information with high accuracy. We will develop a lightweight and high-performance image AI that can be applied to an on-vehicle edge device. **(Fig. 4)**.

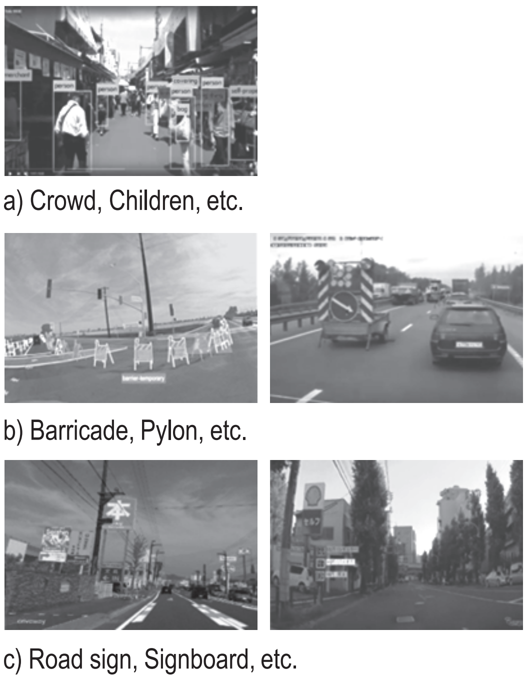


Fig. 3 Examples of Objects Necessary to Implement Services

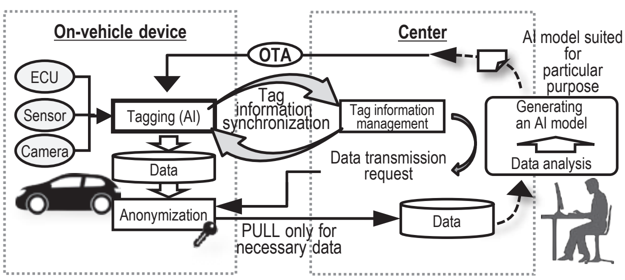


Fig. 4 Application of Image AI

3. Method for Detecting Objects Using DNN

As a method for identifying collected images and attaching tags to determine the area for personal information protection, an image AI using the Deep Neural Network (DNN) is a major candidate.

Since the DNN is applied to detect objects and has significantly higher detection performance than other conventional methods, it has attracted considerable attention in recent years. Typical end-to-end models using the DNN for object detection include Faster R-CNN [3], SSD [4], and YOLOv3 [5]. **Table 1** shows the performance of each model using the MS-COCO [6] dataset.

Table 1 Performance of Major Object Detection Model

Object detection model	DNN input size [pixel]	Inference speed [FPS]	Accuracy [mAP*(1)]
Faster R-CNN	224	5.8	59.1
SSD	321	16.4	45.4
YOLOv3	320	45.5	51.5

Table 1 shows that Faster R-CNN achieves the highest accuracy among the three models, although its inference speed is slow compared to that of SSD and YOLOv3. This means that there is a trade-off between the inference speed and the accuracy. In this study as the result of verification of the assumed application, we selected the YOLOv3 model, which achieves high inference speed and high accuracy, as a candidate DNN for operation through on-vehicle devices.

---

*(1) The mean average precision (mAP):
This is an indicator used to evaluate the object detection accuracy. It is calculated as the mean of the average precision (AP), which is the average of precision until the point in time when the information of a certain object is given. It shows that the higher the numeric value is, the higher the accuracy of the object detection is.

First, we evaluated the performance of the selected YOLOv3 model. **Tables 2** and **3** show the evaluation results of the inference speed, precision*(2), and model sizes under a condition in which the faces and license plates have been learned. For the faces, we used 190,000 pieces of data disclosed in WIDER FACE [7], and for the license plates, we used 12,000 pieces of data prepared by our company.

Table 2 Face (YOLOv3)

DNN input size	Inference speed [FPS]	Precision	Model size [MB]
224 × 224	30.8	0.87	246.3
416 × 416	14.3	0.94	246.3
608 × 608	7.8	0.95	246.3

Table 3 Number Plate (YOLOv3)

DNN input size	Inference speed [FPS]	Precision	Model size [MB]
224 × 224	23.7	0.85	246.3
416 × 416	13.6	0.94	246.3
608 × 608	8.5	0.94	246.3

Next, we investigated the cause of a low precision when the DNN input size is 224 × 224. To clarify the relationship between the object size in a video image and the detection performance, we classified objects in the drive recorder's video images (1,920 × 1,080 pixels) into three types (Small, Medium, and Large) according to the object size. For the three types of object sizes, we followed the classification used in MS-COCO (**Table 4**).

Table 4 Classifying Object in Video Images by Size

Classification by size	Object area size S [pixel]
Small	S ≦ 32 × 32
Medium	32 × 32 ＜ S ≦ 96 × 96
Large	96 × 96 ＜ S

Tables 5 and **6** show the precision and recall rate*(3) of the faces and license plates, which are classified into the object sizes shown in **Table 4**, for the three types of DNN input sizes (224 × 224, 416 × 416, and 608 × 608). From **Tables 5** and **6**, it can be seen that for a small 224 × 224 input size, the precision and recall rate are significantly low. Because it is possible to visually identify faces and license plates that are classified into a small size, a high precision is required for this size. If the object size is 416 × 416 or larger, there is no significant difference in performance. Therefore, if the model is applied to personal information protection, it is necessary that the DNN input sizes be 416 × 416 or larger.

Table 5 Detection Accuracy of Face

DNN input size	Object size	Precision	Recall rate
224 × 224	Small	0.77	0.46
	Medium	0.94	0.78
	Large	0.98	0.92
416 × 416	Small	0.92	0.70
	Medium	0.96	0.87
	Large	0.98	0.94
608 × 608	Small	0.93	0.74
	Medium	0.97	0.88
	Large	0.97	0.93

* (2) The precision:
This is an indicator used to evaluate the accuracy. It indicates the ratio of actually correct objects to objects that were predicted to be correct. It shows that the higher the precision is, the higher the accuracy is.
* (3) The recall rate:
This is an indicator used to evaluate a misjudgment (waste of right recognition), and indicates the ratio of the number of objects predicted to be correct to the number of objects that are actually correct. The higher the recall rate is, the lower the number of objects that are misjudged is.

Table 6 Detection Accuracy of License Plate

DNN input size	Object size	Precision	Recall rate
224 × 224	Small	0.74	0.68
	Medium	0.95	0.90
	Large	0.89	1.00
416 × 416	Small	0.90	0.92
	Medium	0.98	0.95
	Large	0.89	0.98
608 × 608	Small	0.92	0.95
	Medium	0.98	0.96
	Large	0.89	0.98

As shown above, with the existing YOLOv3 model, it is possible to secure high detection accuracy if the input sizes are 416 × 416 or larger. However, from **Tables 2** and **3**, it has been found that the reduction of model size is necessary regardless of the DNN input size since it is not small as applied to our assumed on-vehicle device, and speed-up of the inference for DNN input size 416 × 416 or greater is also necessary.

4. Issues toward Application in On-vehicle Device

To apply the image AI using a DNN to on-vehicle devices, we need a lightweight model that can detect objects at a high speed and with high accuracy.

<Issues>

(1)Realtime inference processing on the on-vehicle devices

(2)Size of the object detection model that can be implemented in an on-vehicle platform

(3)Satisfying both (1) and (2) above and achieving a high detection accuracy

5. Approach to the above Issues

We verified the effect and influences on the detection accuracy for the following two lightweight technologies to speed up the inference and reduce the model size while maintaining a high detection accuracy.

(1)Speeding up the inference

The following quantization [8] is proposed: convert the DNN parameters (convolution factor, bias factor, etc.), which are generally calculated within the range of 32-bit floating point numbers, into parameters with small bits. After applying the quantization calculating within the range of 16-bit floating-point numbers to YOLOv3, verification of the effect of the increased speed is conducted.

(2)Reducing the model size

As a method for reducing the weight of models, the following methods are proposed: quantization of the DNN (binary conversion [9], ternary conversion [10], etc.), sparsification [11], and weight pruning [12] and others. Although it is possible to greatly reduce the model size of the DNN by using these methods, there is a problem that the on-vehicle device cost increase because dedicated hardware is needed to speed up the inference.

For a method in which dedicated hardware is not needed, channel pruning [13] is proposed. Channel pruning is a method for realizing the speed-up and reduction of the number of model parameters while maintaining the accuracy by removing channels that have no effect on the detection results from the calculation of the convolution layer, which is a dominant factor in the number of calculations conducted in a DNN. By applying channel pruning to YOLOv3 after the quantization described above, we have verified the effects of the speed-up and the model size reduction.

6. Verification Method and Results

6.1 Verification method

Fig. 5 shows the verification system used. We applied a Logicool USB camera as an on-vehicle camera and a NVIDIA Jetson Xavier, which is a built-in device that includes an AI processor as an on-vehicle edge device. For the cloud-side, we developed and verified trial software operating on a PC.

Because the weight reduction method does not depend on the detection objects (faces and license plates), we use a dataset of license plates in this study and conduct the verification by applying a model whose DNN input is 416 × 416 pixel, which is a well-balanced size for both the inference speed and the detection performance.

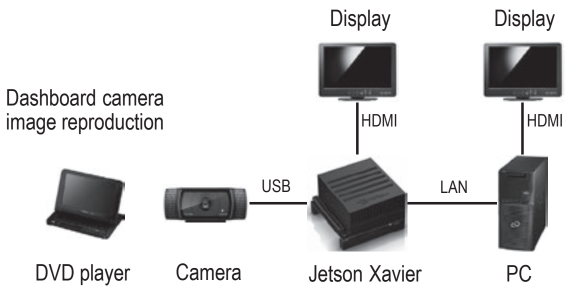


Fig. 5 Verification System

6.2 Results

For the model in which license plates have been learned, we conducted the evaluation described in the following.

(1)Speed-up based on quantization

Table 7 shows the inference speeds, precision, and model sizes when the DNN parameters are quantized to 16 bit. The inference speeds are the average values of ten-thousand measurements. We confirmed that even when the DNN parameters are quantized, it is possible to speed up the inference while maintaining the precision.

Table 7 License Plate (Quantization)

	Inference speed [FPS]	Precision	Model size [MB]
Without quantization	13.6	0.94	246.3
With quantization	18.6	0.94	246.3

(2)Reducing the model size through pruning

Table 8 shows the results when channel pruning is applied to the YOLOv3 model after 16-bit quantization. The model size of the DNN depends on the number of DNN parameters. It can be confirmed that approximately 61.3 million parameters are gradually decreased by repeatedly applying the pruning process. From **Table 9**, we confirmed that it is possible to speed up the inference by 1.4 times and reduce the model size by 1/7.5 for the model in which the pruning is repeatedly applied eight times while maintaining the precision ratio.

Table 8 Application Result of Channel Pruning

Number of pruning applications	Number of parameters [M]	Model size [MB]	Inference speed [FPS]	Precision
Zero	61.3	246.3	18.6	0.94
One	33.3	134.1	19.7	0.91
Two	20.0	80.4	22.2	0.93
Three	13.5	54.6	23.8	0.94
Four	10.6	42.9	22.7	0.94
Five	9.5	38.2	24.5	0.95
Six	8.9	36.0	23.1	0.94
Seven	8.6	34.5	24.2	0.95
Eight	8.2	32.9	25.3	0.94
Nine	7.7	31.1	25.2	0.84

Table 9 License Plate (Quantization + Pruning)

	Inference speed [FPS]	Precision	Model size [MB]
Quantization	18.6	0.94	246.3
Quantization + Pruning	25.3	0.94	32.9

6.3 Consideration

We conducted a weight reduction of YOLOv3 by applying the quantization in which the DNN parameters, calculated within the range of 32-bit floating-point numbers, were limited to the range of small bits, and the pruning method for reducing the parameters that have a small effect on the inference results. As a result, we confirmed that it was possible to speed up the inference by 1.9 times and reduce the model size to 1/7.5 while maintaining the accuracy (**Table 10**). As further weight-reduction methods, additional approaches such as small-bit quantization (to 8 bit, 4 bit, etc.) or a replacement of the backbone network used for extracting the characteristics of YOLOv3 with a more lightweight network can be applied. In the future, we will apply a further weight reduction, and aim at the implementation of DNN weight-reduction technology that can be executed even in inexpensive on-vehicle devices.

Table 10 Comparison of YOLOv3 and YOLOv3 after Weight Reduction

	Inference speed [FPS]	Precision	Model size [MB]
YOLOv3	13.6	0.94	246.3
YOLOv3 after weight reduction	25.3	0.94	32.9

The following chapter describes the application to the anonymization of information by using personal tag information.

7. Application to Personal Information Protection

In the protection processing of personal information, tag information is used for identifying the personal information of the target. Tag information is attached according to the type, number of objects, and others detected by YOLOv3. If faces or license plates are detected, personal information tags are attached, and the areas of the objects in the images are masked.

Fig. 6 shows the result of face detection for video images of a dashboard camera in a cabin using a model with DNN input size of 416 × 416 pixel. In addition, **Fig. 7** shows the result of license plate detection.

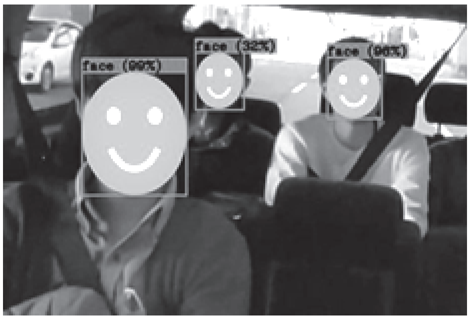


Fig. 6 Anonymization Result of Faces

Fig. 7 Anonymization Result of License Plate

We confirmed that there are no large errors in the position and size of the detection frame for the faces and license plates, and that the accuracy is appropriate enough to apply to the protection of the personal information.

In addition, if a misalignment occurs in the detection position and size, it can be responded by enlarging the size of the mask. We found that utilizing tag information for identifying personal information of the target enables the protection process of it to be applied.

8. Conclusion

In this study, we described the results of a weight reduction model and its application to personal information protection in our DNN-based image AI for on-vehicle devices.

First, we adopted YOLOv3 as an object detection model and reduced the weight of the model by applying quantization and pruning. We then confirmed that it is possible to reduce the number of parameters and speed up the inference while maintaining the detection performance, and showed the effectiveness of such a weight reduction approach.

Next, we demonstrated that it was possible to mask the pedestrians' faces and license plates using personal information tags.

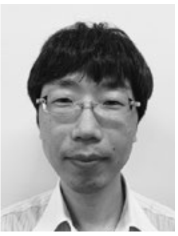
Based on these results, we confirmed that, with use of the DNN-based image AI, it is possible to determine the areas in which personal information should be protected by applying tagging and tag information to identifying collected images.

In the future, we plan to proceed with implementation to a built-in evaluation board in on-vehicle devices for extending the application to dashboard cameras with a communication function and vehicles in the fleet contract, as well as to promote further weight reduction.

References

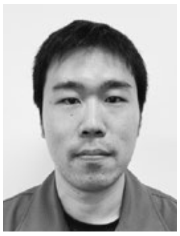
1) G500 Series Commercial Use Drive Recorder, https://www.denso-ten.com/jp/biz-recorder/g500/products/index.html
2) Data logger, https://www.denso-ten.com/jp/biz-recorder/g500/products/function03/index.html
3) Ren, Shaoqing et al., "Faster R-CNN: Towards real-time object detection with region proposal networks," Advances in Neural Information Processing Systems (2015).
4) Liu, Wei et al., "SSD: Single Shot MultiBox Detector," arXiv preprint arXiv:1512.02325 (2015).
5) Redmon, Joseph et al., "YOLOv3: An Incremental Improvement," arXiv:1804.02767 (2018).
6) Chen, Xinlei et al., "Microsoft coco captions: Data collection and evaluation server," arXiv preprint arXiv:1504.00325 (2015).
7) Yang, Shuo et al., "WIDER FACE: A Face Detection Benchmark," arXiv:1511.06523 (2015).
8) Krishnamoorthi, Raghuraman, "Quantizing Deep Convolutional Networks for Efficient Inference: A Whitepaper," arXiv:1806.08342 (2018).
9) Matthieu, Courbariaux et al., "Binarynet: Training Deep Neural Networks with Weights and Activations Constrained to +1 or -1," arXiv preprint arXiv:1602.02830 (2016).
10) Hwang, Kyuyeon et al., "Fixed-point Feedforward Deep Neural Network Design Using Weights +1, 0, and -1," Signal Processing Systems (2014).
11) Liu, Baoyuan et al., "Sparse Convolutional Neural Networks," In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2015).
12) Han, Song et al., "Learning both Weights and Connections for Efficient Neural Networks," arXiv:1506.02626 (2015).
13) Liu, Zhuang et al., "Learning Efficient Convolutional Networks through Network Slimming," arXiv:1708.06519 (2017).

Profiles of Writers



Yasutaka
OKADA

Engineering
Management Div
Advanced R&D
Planning Dept



Ryusuke
SEKI

Engineering
Management Div
Advanced R&D
Planning Dept

Kimitaka
MURASHITA

Engineering
Management Div
Advanced R&D
Planning Dept

Hiroshi
YAMADA

Engineering
Management Div