Development of Edge Camera System for Vehicle Detection System Using Local AI Optimizer Based on Minimum Network Resource

Yun Won Choi, Jang Woon Baek, Jin Hong Kim, Joon-Goo Lee
Electronics and Telecommunications Research Institute, Korea
yunwon.choi@etri.re.kr, jwbaek@etri.re.kr, jinhong@etri.re.kr, leejg01679@etri.re.kr

***Abstract*-- This paper proposes an edge camera system for a vehicle detection system using AI local optimization method utilizing minimal network transmission data. Currently, various AI CCTVs are installed, but if they are installed in an area without data network support, updates are slow and optimization is difficult. We improve traffic object recognition by remotely optimizing the detector with minimal data in a 3G or so communication environment, and use it to estimate the speed and location of the vehicle. Local AI optimizer utilizes optimized weight data using DBs using environmental data-based background images, and vehicle speed estimation utilizes warping data-based tracking data. We confirmed the high sensing performance and speed recognition rate through certification exam of the proposed edge camera system.**

I. INTRODUCTION

Recently, as AI CCTV performance increases, it is being used in various areas. With the rapid development of deep learning-based AI detection technology, the performance of image analysis systems has improved, and various detection performance such as forest fire monitoring, harmful tide monitoring, reverse driving prevention, tunnel safety monitoring, etc. [1]. Although AI performance has improved compared to the previous one, it takes a lot of time and additional data in the field to optimize the artificial intelligence object detection model generated based on general data. In particular, AI CCTV installed in remote mountains is difficult to install a wired network and is used as a terminal that provides only text information after manual optimization after installation due to cost problems to use LTE and 5G.

The object detection model used in early AI CCTV is generated as a result learned by utilizing a general dataset to fit the function. As shown in Fig. 1, a general data set is composed of an image with a view close to the front or a view only of a specific part, unlike a field view, so there are many problems with erroneous detection or non-detection. Therefore, it is common for detection models used in general artificial intelligence CCTVs to passively eliminate these problems through optimization by using additional datasets suitable for the site after installation [2,3].

Fig. 1. General dataset (ImageNet) vs Real CCTV screen

The main function of the vehicle detection system is to detect the vehicle in the camera using the vehicle speed estimation method and then extract the position and speed. Existing vehicle speed estimation is a method using an under-road sensor, which must be installed on the road every time and has disadvantages in terms of time, space, and cost. With the improvement of deep learning-based object detection technology, vehicle speed and position estimation technologies are widely used for intelligent traffic analysis, and various studies such as methods using single camera images or radar sensor fusion are being actively conducted. Vehicle position and speed estimation techniques basically use object detection and tracking, which uses a method of predicting the class after extracting the object candidate region [4] and a method of predicting the class and region at the same time [6-8], and object tracking uses a transformer and GNN to estimate the relationship of the object information [9-11]. There is also a method of detecting and tracking the position of an object through geometric coordinate transformation of multiple cameras [13,14].

In this paper, we propose an Edge Camera System for traffic object information detection based on a deep learning model installed outside of islands and mountainous areas and optimized detection performance by utilizing low-capacity wireless data. To improve the performance of object detection models, we synthesize background images of CCTV-installed regions and meta-information of detected objects on remote servers and update the re-learned model to improve performance and predict vehicle information using optimized detection models. We tested the detection performance and vehicle speed prediction performance through a certification test institution that demonstrates vehicle speed by implementing the proposed system, and confirmed the best performance indicators.

II. PROPOSED SYSTEM

A. System Overall

This paper consists of an edge camera system and an AI Sever, as shown in Figure 2. The edge camera system preprocesses the image input through an ip camera or a usb camera, and estimates the speed based on the accumulated information after detecting and tracking objects in the vehicle detection system (VDS).

The object detection module of the VDS performs optimization by updating the re-learned weight data by sending

background images and meta information of the field to the server using the Local optimizer.

The AI Server consists of a regional data generator and a re-learning module that synthesizes background images received from the edge camera system and sample images based on meta information, and retransmits the weight data generated after re-learning to the edge camera system

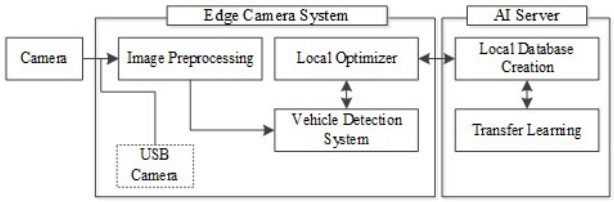


Fig. 2. System Overall

B. Hardware Implementation

The proposed system is configured to use both an embedded terminal, USB-camera, or IP-camera, and initially produced a prototype for NVIDIA Jetson TX2 as shown in Fig. 3, and currently has versions using NVIDIA Jetson Nano and Orin. Our edge camera for VDS is implemented in C++ language and consists of a camera input and preprocessing module, a VDS module that estimates speed after object detection and tracking, and a local optimization module that updates re-learned weight data by sending background image generation and meta information. When a vehicle is detected in the detection area specified in the camera image, the position and speed of the vehicle are estimated, and the result are stored as snapshot and event message.

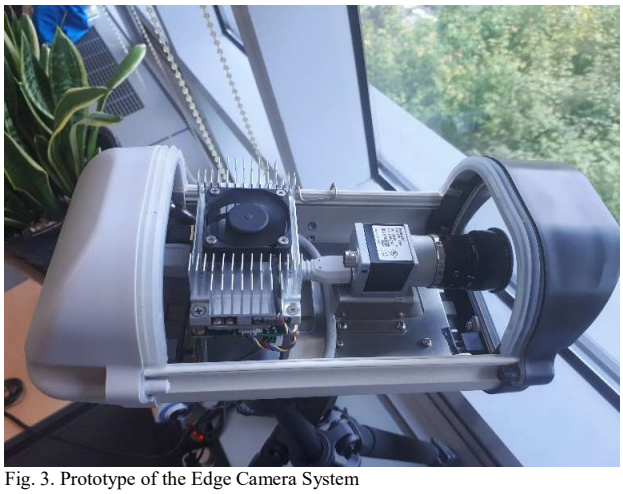
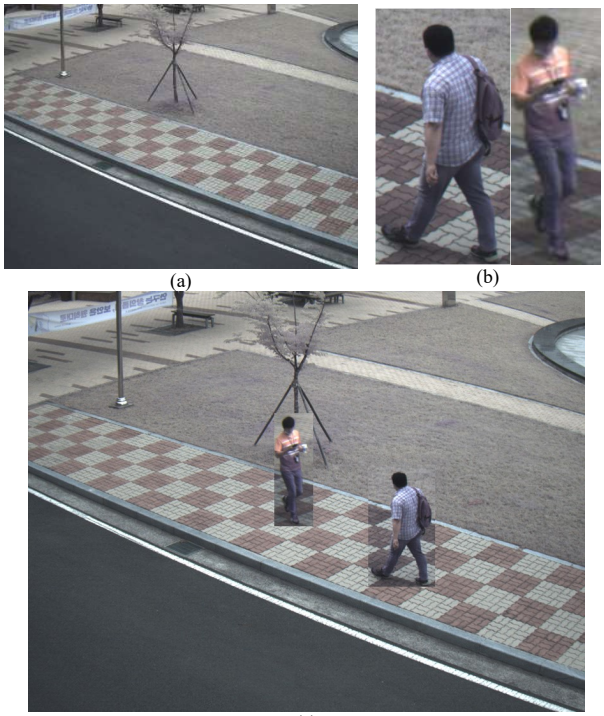


Fig. 3. Prototype of the Edge Camera System

C. Local AI Optimizer Based on Minimum Network Resource

We propose a method to optimize object detection model performance for AI CCTV, which is installed outside of islands and mountainous areas and utilizes a data network that is difficult or slow to connect to the network in real time. Deep learning-based object detection models initially used in the proposed system have very high generalization performance because they were trained based on general data. For academic systems, generalization performance is important in general performance indicators, but characterization performance such as overfitting is very important when manufactured as a commercial product. Therefore, commercial products used in certain environments require an optimization process after installation. A high generalization performance has high object detection performance, but false detection is also high, so it is necessary to reduce erroneous detection and improve detection performance through re-learning by entering negative information.

In this paper, the extracted background image based on the accumulated image at the beginning of the installation is first sent to the AI server, and meta information (x, y, width, height) detected by the existing object model is continuously stored in the server for a certain period of time. As shown in Fig. 4(a), a new database is created by attaching sample data (Fig. (b)) using object meta information to background images classified based on environmental information such as current time and weather. A database as shown in Fig. 4(c) is created and re-learned using the generated image and coordinate information to generate detection weight information optimized for the environment in which the camera is installed. This detection model is transmitted to AI CCTV to update the information file and re-run the detection model. Unlike the existing method of creating an optimization DB by receiving both detection images and meta information, we propose a method of reducing erroneous detection and improving detection performance by utilizing minimal data (background image & meta information text).



(a) (b)

(c)

Fig. 4. DB creation method for local optimization

As shown in Fig. 5, we conducted a test by applying the regional optimization method to the highway CCTV image. As shown in Fig. 5 (a), some vehicles could not be detected with the existing detection model, and erroneous detection occurred on the outskirts of the road. We generate a background image as shown in Fig. 5(b) using a background image generation method that includes a classifier based on environmental information from the original image where the detection results are not shown in Fig. 5(a). By attaching a traffic object to the currently detected meta-information-based location to the background image, the newly created database has the form shown in Fig. 5(c). When re-learning the weight of the detection model using this database and re-detection is performed using it, the results shown in Fig. 5(d) can be obtained and it can be confirmed that the optimization has progressed well.

(a)

(b)

(c)

(d)

Fig. 5. Local optimization result

D. Vehicle Detection System Based on Speed Estimation

In this paper, traffic objects are detected every frame based on the optimized yolo-detector and changes in objects are tracked using similarity between characteristic information such as size, color, and shape of detected objects in front and rear frames. Although the probability of missing is low thanks to the already optimized detector, tracking can determine trends such as positions, intervals, etc. that change since the first detection.

The speed estimation module shown in Figure 6 has a configuration, and first obtains a warping matrix that converts the coordinates of the initial input image into warping target coordinates. Using the matrix obtained in this way, the coordinates of the detected object are converted to obtain the location in the warping image, and the actual location is estimated by applying the dotted reference distance (8m per 1 line set) for each road. The speed estimates the actual speed by comparing the actual coordinates of the object detected in images at regular frame intervals.

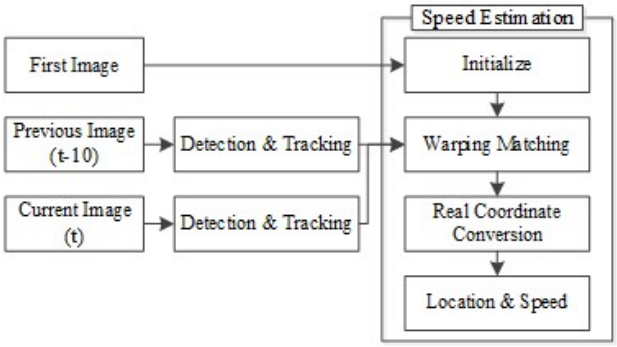


Fig. 6. Vehicle Detection System Overall

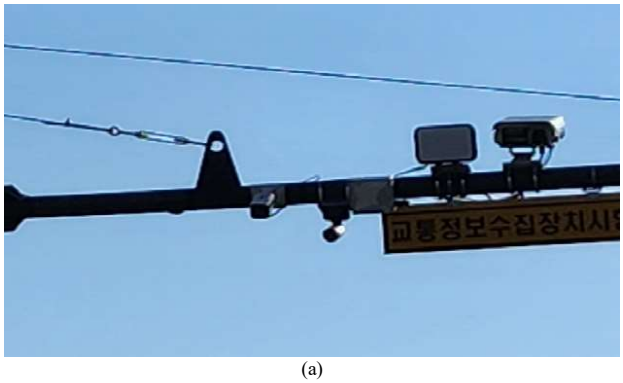
We conducted the test by applying the vehicle detection system to the actual road CCTV. From the image obtained from CCTV, when the right central two lanes are warped in a straight line, the results shown in Fig. 7 top can be obtained. Using the warping matrix and coordinate conversion information, the speed when passing through the cyan area was estimated as shown in Fig 7 bottom, and the performance was confirmed through an experiment in which a vehicle driving at a constant speed was put in.

Fig. 7. Warping image and VDS result

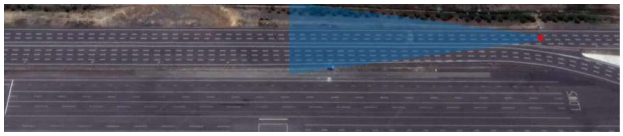
III. EXPERIMENTS

A. Experiment Environment

We used a VDS certification test installed in the high-speed circuit at the Korea Intelligent Automotive Parts Promotion Institute (KIAPI) to verify the object detection performance of AI CCTV and the speed estimation performance of VDS. This certification test verifies and grades the traffic object detection performance, speed estimation performance, and occupancy performance of the proposed system. In particular, as shown in Fig. 8(b), a camera is installed at a red dot position to detect information on vehicles coming from the driving road, and the state in which the camera is installed is shown in Fig. 8(a). The screen obtained from the actual camera is shown in Figure 8(c), and the experiment was conducted using this image information on AI CCTV.



(a)


(b)

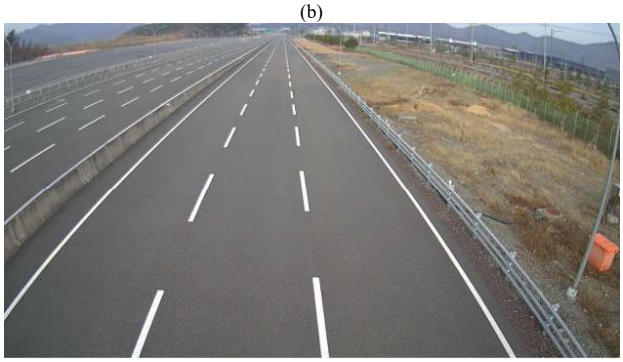

(c)

Fig. 8. Experiment Environment

Our proposed system verified performance based on 201 scenarios by vehicle speed, lane, and vehicle type, and compared the information obtained from the evaluation reference equipment with the result information of the proposed system. Using the comparative information, the Mean Absolute Percentage Error (MAPE) is calculated, and the accuracy for the entire scenario is calculated, and the ratings shown in Table 1 are assigned.

$$\text{Accuracy} = 100 - \text{MAPE}(\%) = 100 - \frac{100}{\text{n}} \sum_{i=1}^{n} \left| \frac{Y_i - X_i}{Y_i} \right|$$

$\text{Y}_\text{i} =$ *Reference value of the* i^{th} *unit of analysis*
$X_i =$ *Measurement value of the evaluation equipment at the ith analysis unit time*

TABLE 1
TABLE OF PERFORMANCE VERIFICATION CRITERIA

Grade	Traffic Accuracy (%)	Speed Accuracy (%)
A	≥95%	≥95%
B	95>, ≥90	95>, ≥90
C	90>, ≥80	90>, ≥80
D	< 80%	< 80%

B. Experiment Results

Experimental results of the proposed system are shown in Table.2. Object detection results for a total of 201 scenarios are 100%, and the MAPE of the speed estimation results is 4% (8.26/201), which has 96% accuracy, so it received an A grade for both fields. For the actual 201 scenarios, each information can be confirmed through table.3, and the actual object detection and speed estimation results can be confirmed through Fig. 9. It was confirmed that the proposed system had high performance through grade A in the certification test.

TABLE 2
THE RESULT OF PERFORMANCE VERIFICATION

Item	Result	
Traffic Accuracy (%)	100%	A
Speed Accuracy (%)	96%	A

TABLE 3
THE DETAILED RESULT OF PERFORMANCE VERIFICATION

NO	TIME	REFE. SYSTEM	OUR SYSTEM	AE	MAPE
1	14:27:40	61	67	6	0.089552239
2	14:28:49	72	75	3	0.040000000
3	14:29:27	84	86	2	0.023255814
4	14:37:51	69	67	2	0.029850746
⋮	⋮	⋮	⋮	⋮	⋮
96	16:29:48	87	85	2	0.023529412
97	16:30:49	97	91	6	0.065934066
98	16:31:08	93	102	9	0.088235294
99	16:31:20	96	94	2	0.021276596
100	16:31:40	90	86	4	0.046511628
101	16:32:57	85	83	2	0.024096386
102	16:33:08	88	92	4	0.043478261
103	16:33:17	92	89	3	0.033707865
104	16:33:46	96	92	4	0.043478261
⋮	⋮	⋮	⋮	⋮	⋮
196	17:50:10	83	81	2	0.024691358
197	17:51:19	72	75	3	0.040000000
198	17:51:32	89	83	6	0.072289157
199	17:53:06	94	89	5	0.056179775
200	17:53:26	77	81	4	0.049382716
201	17:54:06	95	99	4	0.040404040

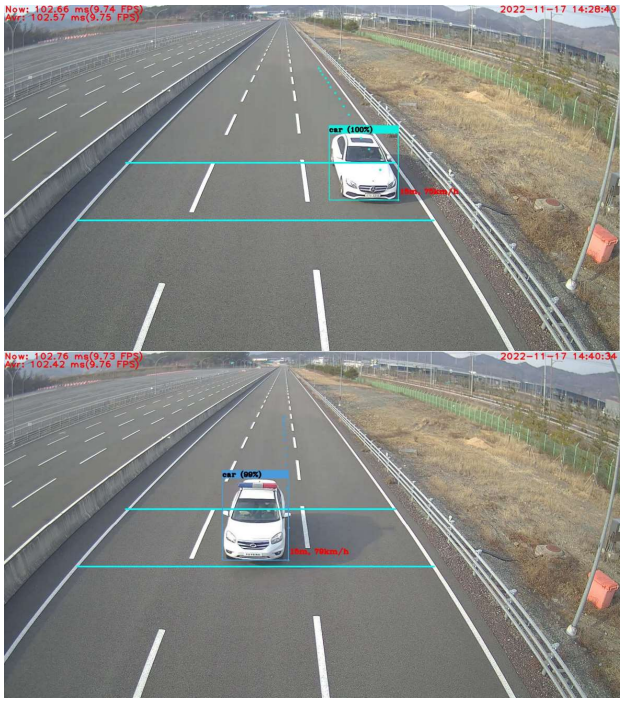


Fig. 9. VDS performance verification result

IV. CONCLUSION

In this paper, we propose an edge camera system for vehicle detection system using local ai optimizer based on minimum network resource. Existing edge cameras for AI CCTV were difficult to connect to real-time data networks, making it difficult to support remote optimization, and in the existing central intelligent control system, speed has been estimated through servers in the control center. We propose a system that optimizes the object detection model using minimal data resources and includes a vehicle speed estimation model capable of real-time processing to detect vehicles from edge cameras installed in fixed environments and estimate speed estimation. The proposed system evaluated its performance through KIAPI's VDS certification test and received grade A to confirm its high performance, but it was detected as it was related to system stability in multiple ways, but it verified its low performance in event management. In the future, we will build an edge camera system using low-level platform with NVIDIA Jetson Nano.

We will develop an event management module suitable for multi-lane environment and various event detection, and we will develop a system suitable not only for VDS but also for Automatic Incident Detection Systems (AIDS).

ACKNOWLEDGMENTS

This work was supported by Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government [23ZD1120, Development of ICT Convergence Technology for Daegu-GyeongBuk Regional Industry].

REFERENCES

[1] S. Hong, B. Min, and D. Han, "HD-CCTV system with extended transmission distance for smart surveillance system", IEEE International Conference on Consumer Electronics, pp. 207-208, Jan. 2016.

[2] B. Blanco-Filgueira, D. García-Lesta, M. Fernández-Sanjurjo, V. M. Brea, and P. López. (2018). ''Deep learning-based multiple object visual tracking on embedded system for IoT and mobile edge computing applications.'' [Online]. Available: https://arxiv.org/abs/1808.01356

[3] Y. Ma, Y. Cao, S. Vrudhula, and J.-s. Seo, "Optimizing the Convolution Operation to Accelerate Deep Neural Networks on FPGA," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, pp. 1–14, 2018

[4] S. Ren, K. He, R. Girshick, J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", Advances in Neural Information Processing Systems 28, NIPS 2015

[5] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779-788, 2016

[6] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement", arXiv preprint arXiv:1804.02767, 2018

[7] A. Bochkovskiy, C. Wang, and H. M. Liao, "Yolov4: Optimal speed and accuracy of object detection", arXiv preprint arXiv:2004.10934, 2020

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, and A. C Berg. "Ssd: Single shot multibox detector", In European conference on computer vision, pp. 21–37, 2016

[9] P. Chu, J. Wang, Q. You, H. Ling and Z. Liu, "Transmot: Spatial-temporal graph transformer for multiple object tracking", arXiv preprint arXiv:2104.00194, 2021

[10] P. Dendorfer, H. Rezatofighi, A. Milan, J. Shi, D. Cremers, I. Reid, S. Roth, K. Schindler, and L. Leal-Taixe, "Mot20: A benchmark for multi object tracking in crowded scenes", arXiv:2003.09003[cs], Mar. 2020. arXiv: 2003.09003.

[11] F. Zeng, B. Dong, T. Wang, C. Chen, X. Zhang, and Y. Wei, "Motr: End-to-end multiple-object tracking with transformer", arXiv preprint arXiv:2105.03247, 2021
[12] J. Revaud and M. Humenberger, "Robust automatic monocular vehicle speed estimation for traffic surveillance," in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 4551–4561, Oct. 2021
[13] E. Ristani and C. Tomasi, "Features for multi-target multi-camera tracking and re-identification," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops, pp. 6036–6046, 2018
[14] P. Li, J. Zhang, Z. Zhu, Y. Li, L. Jiang, and G. Huang, "State-aware re-identification feature for multi target multi-camera tracking", In IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2019