Toward an Online Continual Learning Architecture for Intrusion Detection of Video Surveillance

BEOM KWON1 AND TAEWAN KIM2

1Department of Artificial Intelligence, Dongyang Mirae University, Seoul 08221, South Korea
2Data Science Major, Dongduk Women’s University, Seoul 02748, South Korea

Corresponding author: Taewan Kim (kimtwan21@dongduk.ac.kr)

This work was supported by the Grant of the Korea Health Technology Research and Development Project through the Korea Health Industry Development Institute (KHIDI), Ministry of Health and Welfare, Republic of Korea, under Grant HI22C1884.

ABSTRACT With recent advances in deep learning technologies, many commercialized video surveillance systems have adopted Artificial Intelligence (AI)-powered video analytics technologies as a way to make our life smarter and safer. Nevertheless, there is no robust architecture with an appropriate network model for commercial services considering both high accuracy and low computational cost. Existing deep learning technologies would not be enough to model and represent the dynamics of the real-world scene, so it is difficult to satisfy all environments using a generic model. Appropriate training data from false-alarm and/or missed cases can address this limitation but is rarely available due to legal issues relating to the privacy of personal data and the unpredictability of new incoming data. In this paper, we propose a novel end-to-end hybrid video surveillance architecture for reliable object detection, consisting of front-end and back-end intelligence. For the intelligent front-end, we propose a new object detector with a Multi-scale ResBlock scheme to consider the scalability and flexibility of the system. We are also developing a new domain adaptation method to replace the generic model with each camera’s individual personal model by understanding real-time space and context information for intelligent back-end architecture. It is an iterative and continuous process in which new upcoming data and previous models are consistently engaged in a continuous improvement process. We conducted a series of experiments, including an interesting proof-of-concept tests called the Chameleon project, which demonstrated the high accuracy and versatility of the new architecture, while producing robust results that can be implemented in practice.

INDEX TERMS Intrusion detection, online learning, object detection, domain adaptation, continual learning.

I. INTRODUCTION Recent successful technological advances in deep learning-based image/video recognition haves led to an explosion in consumer demand for Artificial Intelligence (AI)-based products and services in many real-world applications such as autonomous driving [1], medical image processing [2], virtual training simulator [3] and video surveillance [4]. Among these applications, much attention has been paid to intelligent video surveillance based on AI to ensure rapid and precise exchange of safety information. This advanced technology enables us to solve various problems related to our lives and property. In this context, NVIDIA has been running for several years a workshop entitled AI City Challenge, which ranges from constructing training and test data to designing architecture between the edge and the cloud server to make our cities smarter and safer [5].

Much work has been focused on improving the performance of intelligent video analytics (IVAs) to detect, trace and recognize important or dangerous objects [6], [7]. Unlike conventional feature-based approaches, Convolutional Neural Network (CNN)-based object detection methods
rates large and small sub-networks to detect various sizes of pedestrian instances in an image. The authors in [13] designed an FPGA-based smart camera for object detection in video surveillance applications to solve the resource scarcity problem and analytical processing of live video based on the edges. Moreover, Sermanet et al. [17] applied an unsupervised approach based on convolutional sparse coding to pre-trained CNNs for pedestrian detection. However, as shown in Fig. 1(a), people appear in various poses, clothes and sizes, under various lighting conditions, so some problems may arise when designing a general intrusion detector. Unfortunately, animals, neon sign lamps, mannequins, and bugs in particular in the low-resolution video surveillance image can be recognized as a person with high confidence score, as shown in Fig. 1(b). Thus, many studies have tried to explain why this problem occurs and how it can be solved in terms of ‘data refinement’ [18], ‘domain adaptation’ [19], [20] and ‘new training techniques’ [21]. The authors in [22] developed a transfer learning framework to adapt a generic pedestrian detector to a specific domain with a minimum annotation requirement. Also, Kim et al. [21] proposed an unsupervised domain adaptation method for a single-stage object detector by using weak self-training and adversarial background score regularization method in order to reduce the false negatives and false positives. However, they have not been fully implemented and sufficiently validated for use in real-world video surveillance system as nearly every method requires relatively large computational resources.

Fig. 2(a) depicts a conceptual representation of the generic CNN-based method where the detector operates with a universal model in each domain, independent of domain information. Given the limited model capacity with a finite training dataset, it is difficult to satisfy all domains using a single model. Thus, unexpected problems such as false positives may appear in some cases, as shown in Fig. 1(b). To solve this problem, Khodabandeh et al. [18] described a robust method for unsupervised learning of object detection to reduce the inevitable problem of domain shift and privacy regulation in surveillance applications. The authors of [23] developed a framework for cross-domain detection of weakly supervised objects, assuming that they could access image-level annotated images in the target domain. However, these approaches have inherent limitations as they often focused solely on the training phase without considering the inference phase in the new domain. Hence, in order to deploy it on a real video surveillance platform, it is necessary to adapt the universal model to specific domains in the inference phase with real-time constraints, as depicted in Fig. 2(b).

To address these limitations, we have developed a novel end-to-end hybrid video surveillance architecture that we use to robustly detect people incorporating front-end and back-end intelligence. In the case of an intelligent front-end system, we propose a new object detector for detecting pedestrians in video monitoring utilizing Multi-Scale ResBlock (MSRB). A new domain adaptation method is then used to replace the general model with a person-specific model by automatically extract meaningful feature representations from the training data to simultaneously find the location and class information of objects [8]. Broadly speaking, it can be divided according to the number of object detection steps, one of each is a two-stage detector and the other is a one-stage detector. For the two-stage object detector, R-CNN [9] leverages CNN to classify the object after generating sparse propositions that could potentially include objects inside. Compared with the two-stage object detector, the one-stage detector such as SSD [10], YOLOv2 [11], and Pelee [12] directly predicts coordinates and class information with no region suggestion and refinement stages, making them widely used in many real-time video surveillance applications for their computational efficiency. The authors in [13] designed an FPRA-based smart camera for object detection in video surveillance applications to solve the resource scarcity problem and make edge-based live video analytic processing. However, a novel hybrid architecture between an intelligent camera and a cloud server that efficiently captures important objects in real-time has not yet been systematically developed.

One of the most significant objects in video surveillance is a person since it is strongly associated with every event and activity like an intrusion, loiter, arson and abandonment. Therefore, many IVA functions have been extensively studied in terms of detecting and tracking a person [7], recognizing the appearance of his/her gender and age [14] and predicting upcoming future events [15]. In [16], the authors designed a new CNN model based on Fast R-CNN, which incorporates large and small sub-networks to detect various sizes of key objects, such as pedestrians, in real-time with high accuracy. However, the authors did not specifically focus on the challenges of detecting low-resolution images or highly occluded objects, which can significantly affect the performance of modern object detection systems. Therefore, further research is needed to address these issues and improve the overall accuracy and reliability of object detection systems.
FIGURE 2. Conceptual diagram of the conventional and proposed domain adaptation methods for a CNN-based object detector. (a) The detector operates with a universal model in all domains so it could not satisfactorily cover all areas. (b) Each camera has an individual model after adapting its specific environment according to the operation of each site.

FIGURE 3. Conventional and proposed video surveillance platform. (a) A ‘one-way’ framework where the front-end camera sends the detection results to the back-end server. (b) A mutual ‘two-way’ framework where the camera and server continuously exchange detection results and model performance data.

This paper is one of the first to study a life-long AI video surveillance architecture with a real-world platform that is primarily motivated by the near perfect accuracy and scalability of an IVA system. To demonstrate the usefulness and robustness of the proposed approach, we conducted extensive experiments on two public and one private datasets and show that the proposed method significantly outperforms some of the common baselines. This advantage enables this hybrid architecture to be deployed on a real-world commercial platform where we performed proof-of-concept (POC) tests with a Microsoft Azure cloud server to validate its applicability.

The main contributions of the proposed approach are summarized as follows:

- An end-to-end hybrid architecture between an intelligent camera and a cloud server was developed for detecting a person in a video surveillance system. It is an iterative and feedback-based framework that collects meaningful data through manual (user feedback) and automated operations for continuous understanding and learning.

- For the intelligent front-end, we propose a variant of the PeleeNet architecture called MSRB-PeleeNet for the CNN-based object detection model that performs competitive with other strong baselines.

- In order to precisely understand each space and contextual information, we are developing a new domain adaptation method in the inference phase, yielding an individual personal model in our intelligent back-end.

- To compare and verify the proposed method in a more practical way, we conducted POC tests under real conditions and laboratory experiments on the set of test data.

II. RELATED WORKS

A. SYSTEMATIC EDGE-TO-CLOUD APPROACHES

There are several approaches to improve CNNs performance using an online process to overcome the lab test result. One of the outstanding approaches is to collect as many photos as possible online to cover all possible cases in the real-world environment. The authors in [24] proposed to utilize world wide web resources to learn robust object category models for detecting objects in real cluttered scenes. In [25], they developed the discriminatory framework for active and online learning which showed better performance compared to the
study [24]. Both approaches did not focus on accelerating CNN’s performance with edge-to-cloud architecture, but utilized the same number of training samples for supervised learning.

The authors in [26] used many hardware resources, including a Central Processing Unit (CPU), Graphics Processing Unit (GPU), parallel image processing, video-based on distributed data analytics and machine learning framework. Its systematic optimization results in approximately 5 to 19 times acceleration compared to the off-the-shelf approach. Kang et al. conducted effective and efficient large-scale video analysis to accelerate large-scale image and video processing [27]. They proposed a search mechanism for the appropriate CNN model for a specific application based on a neural network analysis system, leading to highly accurate model reproduction. In addition, Han et al. developed a new CNN optimization system that produces a catalog of variants for each model and a runtime that schedules these variants on devices and in the cloud with regard to accuracy and resource bound [28]. To the best of our knowledge, this study could be one of the first papers to balance CNN-based intelligence between edge and cloud. Recently, Emmons et al. proposed preliminary work and described further research topics on advanced neural networks in a large-scale video surveillance application [29]. Since the general neural network is a black-box, they developed a split-brain architecture that balances the neural network activation between the edge and the cloud. They also introduced many communication and computation trade-off challenges to create a scalable and flexible architecture that delivers balanced edge to cloud computing.

B. DOMAIN ADAPTIVE OBJECT DETECTION FOR PERSON

Object detection is an important technology in a real-world video surveillance system, requiring very light computation and almost 100% accuracy. The authors in [30] proposed a person detector that is highly robust to occlusion and hard negative cases by applying average grid classifiers for post-refinement. Ren et al. presented a very simple but efficient method using a region proposal network and Boosted Forests for pedestrian detection based on Faster R-CNN [8].

Although the CNN-based object detection model can perform well in most cases, potentially negative results can arise in some video surveillance locations. To mitigate this problem, some studies have recently suggested a domain adaptation method for object detection. However, it has received less attention compared to classification and semantic segmentation applications. This is because detection tasks require additional consideration of class imbalance, intra-class variation of negative classes and require high dimensional and dense features compared to classification tasks [31].

For domain-adapted object detection, the authors of [32] introduced unsupervised multi-instance learning with Real AdaBoost [33] and developed a tracking-based effective unsupervised online sample collection mechanism to reduce false alarms caused by domain shifts. While it shows better performance, it depends on the performance of the classifiers, making it difficult to apply in real-world practical applications. To improve the performance degradation caused by false negatives, the authors of [34] proposed an automatic adaptation method that creates soft-labels for unlabeled data in the target domain based on data from existing detectors (e.g. mis-classified objects data). In addition, Kim et al. proposed weak self-training and adversarial background score regularization methods for domain adaptive one-stage object detection [21]. It can reduce false negatives and false positives by masking the gradients of hard negative examples and helps CNN to extract discriminative features from the target background.

III. PROPOSED METHOD

A. PROBLEM SETTING

To describe the problem more precisely, let’s denote the source data $(x^s, y^s)$ which is used in the training set for a given source domain $X_s$, and let the target data be denoted by $(x^t, y^t)$ which is drawn from the target domain $X_t$, where $x$ is an image and $y$ is a labeled data which consists of the coordinates of the person bounding box and class information $c$.

While any good CNN-based object detector has been developed from a massive set of $X_t$ training datasets, it can confidently produce incorrect results when confronted with another new domain, that the detector has never seen before. In general, this is because that the source and target domain backgrounds share less common features compared to the foreground object backgrounds $(P(X_s) \neq P(X_t))$, where $P(X)$ is the marginal probability distribution of the $X$ domain. Understanding this problem involves a number of complex factors that can only be solved by understanding the new target domain by incorporating a sophisticated object detection algorithm into a robust architecture.

B. HYBRID ARCHITECTURE FOR ONLINE CONTINUOUS LEARNING

We propose a hybrid AI video surveillance architecture that combines front-end and back-end intelligence, as shown in Fig. 4. Unlike conventional cloud-based video surveillance platform, we deploy a CNN-based object detection model in the camera for intelligent front-end. Of course, it is really possible to obtain better performance by utilizing a model in the back-end with high computational power and deeper layers for the model. However, it is hard to handle multiple cameras simultaneously for a video surveillance system that operates 24/7 (24h a day and 7days a week) due to the limitations of processing complexity. Hence, if the performance does not differ significantly between the front- and the back-end model, it is more efficient to deploy the model at the front-end, considering the scalability, flexibility and expandability of the system.

In order to develop a robust, initial overall camera object detection model, we train our own CNN-based network model with a training dataset from numerous video

1We set the class information $c = 0$ for background and $c = 1$ for person.
Toward an Online Continual Learning Architecture for Intrusion Detection of Video Surveillance

FIGURE 4. Hybrid architecture for intelligent video surveillance where the data from the new environment and current model are engaged in a continuous improvement process.

surveillance sources. After extensive evaluation processes have been performed on various datasets, an output model is deployed to each object detection camera, as shown in Fig. 4. Although this initial model can perform well in most cases, potentially negative results can occur in some places. To mitigate this problem, we are developing a new domain adaptation method by understanding the individual environment in each camera. In other words, we replace the initial universal model with an individual specific model by carefully learning the relevant information about space and context. This is possible because the surveillance camera is usually installed in a fixed location, unlike other applications, such as autonomous vehicles and handheld cameras.

In our hybrid architecture, a short learning time (adaptation period about (∼1 sec.)) before the inference phase (service period) is needed to understand the environment more deeply. First, we fine-tune the initial model from the new target data to adapt to its environment during an adaptation period. This enables more accurate detection of the updated model by gaining knowledge of its space and contextual information over the service period. To deal with data heterogeneity for different purposes, we divide target images into two pieces of data: latent database and query database according to their confidence score of detection results, as shown in Fig. 4. If the image has a high confidence score from the previous model, we utilize it as the source of the verification dataset for future use to develop new IVA functions. On the other hand, if it has a low confidence score, we use it as a candidate for a new training dataset to overcome the uncertainty.

To ensure flexibility in the proposed architecture, we allow two approaches to sampling meaningful target data during the adaptation period, one manually (user feedback) and the other by automatic operation. For the user feedback method, the domain adaptation process is handled based on user feedback data, such as false positive and true negative cases. In other words, the adaptation procedure is carried out to deliver only a user-selected portion of the recorded video data to guarantee the high quality of the CCTV camera, as illustrated in Fig. 4. Almost every conventional video surveillance system employs this approach to solve the problem of low accuracy.

However, in general, people tend to avoid any kind of feedback related action, unless it is very disturbing. Therefore, we are also developing a new method for automatic domain adaptation by sampling new target domain data, which will be explained in the following Sec. III-D.

It is an iterative and lifelong process in which new up-coming data and previous model are engaged in a continuous process of improvement. However, attempting to assimilate new information into the previous model requires modifying its weights, which in turn significantly interrupts existing knowledge, yielding a problem known as catastrophic forgetting. Thus, to eliminate this problem and allow continuous learning, our architecture conducts the model evaluation process iteratively, allowing the current model to be updated in real-time and adapting to the individual environment. It makes it a hybrid more robust to the adaptation of new individual target data.

C. RELIABLE OBJECT DETECTOR FOR INTELLIGENT FRONT-END

In an edge device-based real-world video surveillance system, real-time operation with low computational cost must be possible, so the AI model must be fast and lightweight, and portability is a very important factor. Especially, to deploy a CNN-based model in the front-end camera, it is imperative
In PeleeNet only applies one size of the kernel (3 × 3), it is not easy to detect occluded or small objects which is a hard case in real-world video surveillance systems. Therefore, we aim to develop a robust object detector (especially for detecting people in video surveillance cameras) while minimizing computational resources. To solve this lack of information for pedestrian detection in a surveillance camera, the studies use a multi-scale feature map or multi-scale kernel that can cover a wide range of receptive fields from the kernel of various scales for visual information. Inspired by previous studies, we propose a variant of the PeleeNet architecture called Multi-Scale ResBlock (MSRB)-PeleeNet that utilizes a multi-scale kernel by redesigning the existing ResBlock residual structure on each additional feature map, as depicted in Fig. 5. The proposed MSRB consists of various scales of kernels (3 × 3, 5 × 5, and 7 × 7) in order to more specifically consider surrounding contextual information. It is possible to analyze the specific layers features using multi-scale kernels, yielding statistically better performance regardless of the image environment. It is also noted that the extracted contextual information is also utilized to maintain performance even with serious small objects.

Fig. 6 shows the qualitative results of two video surveillance scenes, according to which the proposed object detector enables more accurate detection of a person with occluded or small objects compared to the ResBlock of original PeleeNet. Moreover, we exclude the two last feature maps (3 × 3 and 1 × 1) in the original structure in order to reduce the high computational costs in PeleeNet. Finally, we can confirm that MSRB-PeleeNet is suitable for object detectors in front-end video surveillance cameras due to its superior accuracy and low computational complexity. Quantitative analysis will be discussed in Sec. IV-B.

D. CONTEXT-AWARE DOMAIN ADAPTATION FOR INTELLIGENT BACK-END

We propose a real-time domain adaptation scheme on MSRB-PeleeNet at each camera in order to learn about the corresponding environment of $X_t$. As we mentioned earlier, during the adaptation period, we refine the original model with new negative images from the target domain that do not contain any people with several iterations to get updated. This enables the updated model to detect a person more accurately, using knowledge of its space and contextual information through the service period.

Here, we introduce a new method for automatically sampling negative background images during the adaptation period as shown in Fig. 7. For ease of understanding, let’s denote the target data during adaptation period of frame duration $N$ by $x'_t = \{x'_0, x'_1, \ldots, x'_n\}$, where $n$ is the frame number in the range $0 \leq n \leq N$. Although there is no corresponding labeled data with the coordinates of the bounding box for training, we can use it as a negative set when there is no person in the image. First, we choose the first frame $x'_0$ as an element of the finally selected set $x'_S$ ($x'_S \subset x'_t$). At this stage, even
if there are some foregrounds in the scene, we can also get good domain adaptation results when we conducted lots of experiments. The second step is to obtain an image of the gray-scale frame difference:

$$\hat{x}_n^t = \hat{x}_n^t - \hat{x}_{n-1}^t,$$

(1)

where $\hat{x}_n$ is the gray-scale image of the $n$th frame image. Here, it is highly possible that if the normalized mean value of the frame difference over the last few frames (five frames in this paper) is greater than a certain threshold $\epsilon$, more than one moving object appears on the $n$th frame. To avoid the duplication problem,\(^2\) we choose $(n - 1)\text{th}$ frame, $x_{n-1}^t$, as a candidate image for the next step.

In the third step, we define the dissimilarity metric (DSM) of $x_{n-1}^t$ with all the frames in the selected $x_S^t$ sample set to avoid duplication in that set, considering the differences between luminance, contrast, structure and texture:

$$\text{DSM}(x_{n-1}^t) = S(x_{n-1}^t) * \frac{1}{1 + \exp(-T(x_{n-1}^t))},$$

(2)

where $S(x_{n-1}^t)$ is a metric of the structural similarity index (SSIM) [36], which is a remarkable method of assessing image quality taking into account the human visual system, which is calculated for all images in $x_S^t$:

$$S(x_{n-1}^t) = \frac{(2\mu_{n-1}^t \mu_x + C_1) + (2\sigma_{n-1}^t \sigma_x + C_2)}{(2\mu_{n-1}^t + \mu_x^2 + C_1)(2\sigma_{n-1}^t + \sigma_x^2 + C_2)},$$

(3)

where $x_t$ is the selected images before the $(n - 1)\text{th}$ frame, $\{x_t\} \in x_S^t$, and $\mu_x$ and $\sigma_x$ are the mean intensities and standard deviations of the frame $x$, respectively:

$$\mu_x = \frac{1}{L} \sum_{i=1}^{L} x_i, \quad \sigma_x = \sqrt{\frac{1}{L-1} \sum_{i=1}^{L} (x_i - \mu_x)^2},$$

(4)

and $T(x_{n-1}^t)$ is the absolute difference in texture entropy on the $(n - 1)\text{th}$ frame, $E(x_{n-1}^t)$, which represents the difference in visual entropy in terms of texture information based on information theory [37]:

$$T(x_{n-1}^t) = |E(x_{n-1}^t) - E(x_t)|,$$

(5)

$$E(x_{n-1}^t) = \sum_{i} \left( \frac{1}{\gamma T,\hat{x}_{n-1}} \log \left( \frac{s \cdot \gamma T,\hat{x}_{n-1}}{2 \Gamma(1/\gamma T,\hat{x}_{n-1})} \right) \right),$$

(6)

where $p_{T,\hat{x}_{n-1}}$ is the probability mass function of the $i$th sub-band of wavelet decomposition output of $\hat{x}_{n-1}$, $\gamma$ is the shape parameter of the generalized Gaussian distribution, $s$ is the positive constants of $s = \frac{1}{\chi} \sqrt{\Gamma(3/\gamma)/\Gamma(1/\gamma)}$. Thus, the final negative sample set $x_S^t$ is determined by the thresholding of the value of $\delta$.\(^3\) The structure of the above method is summarized in Algorithm 1.

\(^2\)Since the consecutive frames of a surveillance video are almost identical, it is better to extract just one frame as a representative sample than to sample all the frames.

\(^3\)More detailed explanation is given in the previous study of [37].

For training, all background samples are sorted by their predicted confidence scores in ascending order and use some of them with positive samples to build a CNN-based object detector in general [10]. However, we only tune the original object detector using the negative set $x_S^t$ with no positive samples to detect. Therefore, if the original detector finds a person bounding box to be considered a false positive because there is no object present at this location in the image, we punish the loss according to the predicted box’s confidence score as follows:

$$L(c, l) = \frac{1}{N_p} \left( L_{conf}(c) + L_{loc}(c, l) \right),$$

(7)

$$L_{conf}(c) = -\sum_i \log(c_i), \quad L_{loc}(c, l) = \sum_i \text{smooth}_{L1}(l_i),$$

(8)

where $N_p$ is the number of predicted bounding boxes $l$ with a confidence score $c$.

**IV. EXPERIMENTS**

In order to verify the accuracy and suitability of the proposed approach, we performed two types of experiments; one is the laboratory tests and the other is POC tests. In the first part of the experiment (Sec. IV-B), we show the results of the proposed object detector with domain adaptation on two public and one private datasets for intelligent video surveillance. In the second part of the experiment (Sec. IV-C), we validate the effectiveness and unique strengths of the proposed approach in various POC tests. It is noted that we use all of the object detector models including MSRB to detect a person in video surveillance scenes as an example application of the MSRB. Thus, all datasets used in this study are video surveillance sequences captured in a fixed location with limited fields of view.

**A. EXPERIMENTAL SET-UP**

In our experiments, we used two public datasets provided by Performance Evaluation in Tracking for Surveillance (PETS).
TABLE 1. Results and comparison of various CNN network models with public and private training datasets in terms of mAP and fps on the GPU chip (NVIDIA Quadro P6000).

| Method         | Training DB | Input Size | fps | mAP @0.5 |
|----------------|-------------|------------|-----|----------|
|                |             |            |     | PETS     | Oxford | Private P |
| YOLOv4         | MS COCO     | 416 x 416  | 35  | 0.8801   | 0.6124 | 0.8180    |
| EfficientDet   | Private P   | 416 x 416  | 35  | 0.8937   | 0.7627 | 0.8398    |
| MobileNet v3   | Private P   | 512 x 512  | 26  | 0.8823   | 0.7715 | 0.8453    |
| PeleeNet       | MS COCO     | 304 x 304  | 66  | 0.3796   | 0.2114 | 0.7232    |
| PeleeNet       | Private P   | 512 x 384  | 62  | 0.8304   | 0.6965 | 0.7755    |
| Refinedet      | MS COCO     | 300 x 300  | 36  | 0.8504   | 0.7024 | 0.7335    |
| M2det          | MS COCO     | 320 x 320  | 33.5| 0.8216   | 0.6960 | 0.7455    |
| RetinaNet      | MS COCO     | 320 x 320  | 24  | 0.8402   | 0.7846 | 0.8025    |
| HSD            | MS COCO     | 320 x 320  | 39.6| 0.8523   | 0.7505 | 0.8013    |
| EPGRNet        | MS COCO     | 512 x 512  | 47  | 0.8427   | 0.7989 | 0.7513    |
| Ours (MSRB)    | MS COCO     | 512 x 384  | 55  | 0.8892   | 0.8185 | 0.8678    |
| Ours (MSRB)    | Private P   | 512 x 384  | 55  | 0.9074   | 0.8200 | 0.8710    |

TABLE 2. Results and comparison of various CNN network models with public and private training datasets in terms of mAP and fps on the DSP chip (Qualcomm QCS603).

| Method         | Training DB | Input Size | fps | mAP @0.5 |
|----------------|-------------|------------|-----|----------|
|                |             |            |     | PETS     | Oxford | Private P |
| YOLOv4         | MS COCO     | 416 x 416  | 4.2 | 0.8002   | 0.5824 | 0.7923    |
| EfficientDet   | Private P   | 416 x 416  | 4.2 | 0.8341   | 0.7214 | 0.8088    |
| MobileNet v3   | Private P   | 512 x 512  | 4.3 | 0.8105   | 0.7524 | 0.7998    |
| PeleeNet       | MS COCO     | 304 x 304  | 31  | 0.3168   | 0.1042 | 0.7045    |
| PeleeNet       | Private P   | 512 x 384  | 31  | 0.7945   | 0.6008 | 0.7524    |
| Refinedet      | MS COCO     | 300 x 300  | 7   | 0.8125   | 0.6995 | 0.7198    |
| M2det          | MS COCO     | 320 x 320  | 6.3 | 0.8154   | 0.6659 | 0.7104    |
| RetinaNet      | MS COCO     | 320 x 320  | 3.2 | 0.8256   | 0.7801 | 0.7565    |
| HSD            | MS COCO     | 320 x 320  | 4.4 | 0.8213   | 0.7045 | 0.7362    |
| EPGRNet        | MS COCO     | 512 x 512  | 4.7 | 0.8115   | 0.7390 | 0.7034    |
| Ours (MSRB)    | MS COCO     | 512 x 384  | 29  | 0.8673   | 0.8055 | 0.8239    |
| Ours (MSRB)    | Private P   | 512 x 384  | 29  | 0.8895   | 0.8074 | 0.8155    |

Algorithm 1 Negative Sample Selection

Result: $x^f_{3S}$

1. Initialization: $x^f_{0S} \in x^f_S$
2. while $n \leq N$ do
3. Obtain the frame difference by (1)
4. if $\frac{1}{n} \sum_{n=1}^{n-1} x^f_{Dn} \geq \epsilon$ then
5. Calculate DSM($x^f_{n-1}$) by (1)
6. if DSM($x^f_{n-1}$) $\geq \delta$ then
7. Add ($x^f_{n-1}$) into the set of $x^f_S$
8. end if
9. end if
10. end while

2009 [38] and the Oxford Town Centre [39]. For PETS in the 2009 dataset, we used an ‘S2-L1’ video showing an outdoors scene with several pedestrians crossing each other, for a total of 795 frames. We also used the Oxford Town Centre dataset showing pedestrians walking down a busy street, recorded at 25 fps with a resolution of 1920 x 1080 that pedestrians paths are mostly smooth at a constant speed. However, due to privacy regulations there is very small quantity of real video surveillance data. Thus, we used a private dataset which is composed of 130,561 training images (Private P) and 5,787 hard level evaluation images (Private E) with pedestrian bounding boxes for research purposes. Opening these two data to the public is difficult because of some legal issues.

For object detector training and domain adaptation, we set the batch size to 64 and the learning rate to 0.001. When we trained Private P’s private dataset, the network was trained for 150k iterations with an initial learning rate of $1 \times 10^{-3}$, weight decay of $5 \times 10^{-4}$, learning rate decay of 0.1 at 90k and 120k. In the case of benchmark methods, the inputs were resized to a specific size and trained using the default settings in the official code of each algorithm, which we adopt SSD [10] as the basic framework for MobileNet and PeleeNet. The anchor size was extracted using k-means clustering for each PETS 2009, Oxford Town Centre and Private P training set.

Experiments on the server were conducted on the NVIDIA Quadro P6000 with CUDA 9.0 and cuDNN v7.1 and the Qualcomm Vision Intelligence Platform with QCS603 designed for efficient machine learning on IoT devices was used for the AI camera. For accuracy comparison, the mean Average Precision (mAP) was selected, which has been used widely in previous object detection studies, with the Intersection over Union (IoU) threshold of 0.5.
B. EXPERIMENTAL RESULTS AND DISCUSSION

To evaluate the accuracy of MSRB-PeleeNet, we compared the performance with previous object detection models YOLOv4 [40], EfficientDet [41], MobileNet v3 [35], PeleeNet [12], RefineDet [42], M2det [43], RetinaNet [44], HSD [45] and EFGRNet [46]. For a fair comparison, we measure mAP with processing time on two different NVIDIA Quadro P6000 and Qualcomm QCS603 devices after training with the same PrivateT dataset for person detection in video surveillance videos.

Table 1 shows the summarized performance in terms of mAP and frame per second (fps) for two public and one private datasets on GPU. As shown in the table, the proposed model delivered the highest mAP values and reasonable processing time with a substantial margin on the GPU. When PrivateT was used as a training dataset, the fps of PeleeNet outperformed those of other methods. However, the mAP performance of the proposed method is superior to that of other methods, including PeleeNet and MobileNet, for all test datasets (i.e. PETS, Oxford, PrivateE). When MS COCO was used as a training dataset, PeleeNet achieved the best fps performance of all methods. However, PeleeNet exhibits the worst mAP performance for all test datasets. On the other hand, the proposed method, achieving the second-best fps performance, exhibits the best performance in terms of mAP.

### TABLE 3. Results and comparison of the proposed domain adaptation method with various CNN network models in terms of mAP and fps on the GPU chip (NVIDIA Quadro P6000).

| Method          | Training DB | Input Size | fps  | mAP w/ PETS 2009 | mAP w/Oxford | mAP w/PrivateE |
|-----------------|-------------|------------|------|------------------|--------------|----------------|
| YOLOv4 + D.A.   |             | 416 x 416  | 32   | 0.8801           | 0.6124       | 0.8180         |
| MobileNet v3 + D.A. |         | 512 x 512  | 49   | 0.4097           | 0.3335       | 0.5024         |
| PeleeNet + D.A.  | PrivateT & xS | 512 x 384  | 59   | 0.8304           | 0.6965       | 0.7755         |
| RefineDet + D.A. |             | 300 x 300  | 34   | 0.8504           | 0.7024       | 0.7335         |
| M2det + D.A.    |             | 320 x 320  | 30   | 0.8107           | 0.7867       | 0.7732         |
| RetinaNet + D.A. |            | 320 x 320  | 22   | 0.8788           | 0.8168       | 0.8729         |
| Ours (MSRB) + D.A. |            | 512 x 384  | 53   | 0.9074           | 0.8200       | 0.8710         |

FIGURE 8. Results of various CNN network models and our method (MSRB). The first, second and third rows are the results of example images of PETS 2009, Oxford and Private datasets, respectively.
To sum up, compared to the original PeleeNet, the proposed method shows greater accuracy in the overall test datasets, but with a slightly higher computational effort.

Table 2 shows the summarized performance in terms of mAP and fps for two public and one private datasets on a Digital Signal Processor (DSP). Almost all CNN-based object detection models have real-time performance on the GPU, but their accuracy is significantly reduced on the DSP chip because of the high computational complexity as shown in the table. From the table, it is seen that PeleeNet exhibits the best fps performance among all the methods. However, the mAP of PeleeNet was the worst for all test datasets. Although MobileNet v3 also exhibits a fast processing time, it is highly dependent on the hardware types of the mobile devices. In other words, it has so far been difficult to deploy on other chipsets for mobile devices as it uses a new customer layer called a depth-wise separable convolutional layer. On the other hand, the mAP of the proposed method is superior to those of the other methods. In addition, the fps value of the proposed method is greater than for other methods except for PeleeNet. This result is consistent with the mAP and fps performance shown in Table 1.

In order to demonstrate the superiority of the proposed domain adaptation method, its performance (i.e., accuracy and computational complexity) was also compared with the results of other studies. As shown in Table 3, it can be concluded that the effects of the proposed domain adaptation method are very significant when comparing the results with the results of the original object detector. Since a short adaptation period is required for domain adaptation, the value of fps is reduced in all object detectors, but is irrelevant to real-world inference. The main computational burden is due to tuning with $\chi^2$ to update the original model.

Fig. 8 shows the detector results of the previous two object detector models and MSRB with the domain adaptation method on six representative video surveillance images from each test dataset. As shown in the first row of the figure, when two people overlapped each other, the existing Mask R-CNN and PeleeNet methods recognized them as one person. In addition, both detectors often fail to detect a person who is occluded by other objects or appears partially, resulting in a low recall rate. On the other hand, there are some false positives in either foreground objects or backgrounds as shown in Fig. 8. Compared to both object detectors, the proposed MSRB shows a significant increase in performance in accuracy, but there may also be some false positives when they are person-like as shown in the third columns of this figure. Therefore, we can reduce these problems by using the proposed real-time domain adaptation method as shown in the fourth column in Fig. 8.

C. CHAMELEON PROJECT ON COMMERCIAL VIDEO SURVEILLANCE PLATFORM

In order to verify the proposed approach in a real-world video surveillance system, we performed POC tests on a real commercial $T$ view platform, which is a large-scale cloud-based video surveillance service platform with real-time streaming, IVA and dispatch services in South Korea. Three hundred surveillance video cameras were employed out of 130k cameras during the week to validate our method. We call this experiment the Chameleon project, the aim of which is to verify the robust performance of the proposed method in practice.

There are two main components to implementation the system in this project, as shown in Fig. 9. One is Communication for Front-end Intelligence which requires the exchange of data and object detector models between each camera and the cloud server. For the front-end device, we used Microsoft’s Vision AI Developer Kit with Qualcomm QCS603 based on the Azure cloud platform. The second is Understanding at Back-end Intelligence where we train the current model and iteratively evaluate the updated model to improve and identify performance. This makes the hybrid architecture more robust to adaptation to new, individual target data.

All implementations were based on Microsoft’s Azure services in which we used GPU instances, web application server and message transmission protocol among cameras, cloud server, database and storage using Infrastructure as a Service, Platform as a Service and Software as a Service, respectively. More detailed explanations of a system implementation for both components are as follows.

1) COMMUNICATION FOR FRONT-END INTELLIGENCE

In this project, each camera had a container-based client to communicate with the server, which captured images and corresponding metadata were transferred to the storage server, as well as their identification data as shown in Fig. 9. The server then performed a sampling process to obtain a negative sample set in the new target domain during the adaptation period for each camera based on its metadata.

As it is an iterative and lifelong process in which data is continuously accumulated while learning a better model, it is necessary to manage the history of updating models. After training with the new target data, we evaluated the accuracy of the new model before sending it to each camera, comparing the previous one. In this project, this cycle takes approximately one minute for three hundred camera channels.

2) UNDERSTANDING IN BACK-END INTELLIGENCE

We used the Snapdragon Neural Processing Engine-based GPU server and the Caffe framework for a series of steps...
B. Kwon, T. Kim: Toward an Online Continual Learning Architecture for Intrusion Detection of Video Surveillance

FIGURE 10. Results of the proposed approach on four different camera channels of Chameleon project where the first column shows the results of the MSRB, the second column shows the results of the proposed domain adaptation with the MSRB. The last column depicts the PR curve; the y-axis is precision and the x-axis is recall.

to learn, test and quantize all front-end cameras in parallel. Thus, to heterogeneously handle huge amounts of data requests from different cameras, we utilized the load balancing and queuing process on the Microsoft Azure GPU instance as shown in Fig. 9. More specifically, when the back-end server received the images and the corresponding metadata, the training and quantization processes were performed on the container-based server within the Microsoft Azure GPU instance. All these processes were operating simultaneously in the proposed hybrid architecture in parallel and asynchronously.

3) POC RESULTS AND DISCUSSION

Sixty test sequences out of three hundred camera channels were employed to identify the performance of this project on a commercial video surveillance platform. These sequences were annotated manually to ensure high precision of people marking. Each sequence has a frame rate of 15 fps and a length of 2 hours, so 108,000 images were used to evaluate the performance of each camera channel.

Fig. 10 shows the exemplary result frames for four representative sequences from test sequences that were used for validation. As shown in the first column of these figures, some false positives were detected either with the person or in the backgrounds. In addition, false positives were more frequent than true negatives. These errors were reported more often with high confidence results in the infrared mode. The second column of Fig. 10 shows the results of the domain adaptation using the MSRB that the false positives were significantly reduced while maintaining the ability to detect a person. In addition, a precision-recall (PR) curve was used to evaluate the overall performance of each camera, which represents the precision (positive predictive value) versus recall (true positive rate). In the PR curve, the y-axis is precision and the x-axis is a recall that performance increased when we applied domain adaptation during the adaptation period as shown in the last column in Fig. 10.

To investigate the performance depending on the camera environment, we divided the sixty test sequences into three groups of “indoor”, “outdoor during the day”, and “outdoor at night”. Broadly, as shown in Fig. 11(a), the mAP performance for all groups increased over time and was significant for indoor compared to outdoor environment. This is because there was a complex pattern in the indoor scene leading to low precision and accuracy as shown in Fig. 11(b). However, some false positives were also detected in high-frequency image areas, such as moving tree leaves and the reflected light region in outdoor sequences. There was relatively little improvement in the outdoor night scene because there were hardly any moving objects, including pedestrians during the dawn period as depicted in Fig. 11(c).
V. CONCLUSION

Although many state-of-the-art CNN-based object detectors have shown remarkable performance, there are many unexpected errors that can occur during round-the-clock operation, leading to severe problems in video surveillance applications. Here, we propose a novel, end-to-end hybrid video surveillance architecture for reliable real-time object detection, consisting of front-end and back-end intelligence. Unlike conventional approaches, the proposed front-end utilizes a new object detector, called MSRB-PeleeNet to accommodate high scalability, flexibility and cost-effectiveness to meet the requirements of commercial video surveillance platforms. In addition, we develop a new domain adaptation method to replace the generic model with each camera’s individual personal model by understanding the space and context information in real-time for intelligent back-end architecture. It is an iterative and continuous process in which new upcoming data and the previous model are consistently engaged in a continuous improvement process. As the advanced techniques of CNN-based computer vision and the demand for CNN-based intelligent video analysis continue to develop, we believe the proposed method will play an important role in applying the intelligent model to a real commercial platform.

REFERENCES

[1] B. Bogdoll, M. Nitsche, and J. M. Zollner, “Anomaly detection in autonomous driving: A survey,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2022, pp. 4488–4499.
[2] S. Suganayadevi, V. Seethalakshmi, and K. Balasamy, “A review on deep learning in medical image analysis,” Int. J. Multimedia Inf. Retr., vol. 11, no. 1, pp. 19–38, Mar. 2022.
[3] B. Kwon, J. Kim, K. Lee, Y. K. Lee, S. Park, and S. Lee, “Implementation of a virtual training simulator based on 360° multi-view human action recognition,” IEEE Access, vol. 5, pp. 12496–12511, 2017.
[4] K. Yan, H. Shan, T. Sun, H. Hu, Y. Wu, L. Yu, Z. Zhang, and T. Q. S. Quek, “Reinforcement learning-based mobile edge computing and transmission scheduling for video surveillance,” IEEE Trans. Emerg. Topics Comput., vol. 10, no. 2, pp. 1142–1156, Apr./Jun. 2022.
[5] M. Naphade, M. C. Chang, A. Sharma, D. C. Anastasius, V. Jagarlamudi, P. Chakraborty, T. Huang, S. Wang, M. Y. Liu, R. Chellappa, J. N. Hwang, and S. Lyu, “The 2018 NVIDIA AI city challenge,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, Jun. 2018, pp. 53–60.
[6] T. Wang and D. J. Cook, “Multi-person activity recognition in continuously monitored smart homes,” IEEE Trans. Emerg. Topics Comput., vol. 10, no. 2, pp. 1130–1141, Jun. 2022.
[7] Y. Wang, S. Velipasalar, and M. Casares, “Cooperative object tracking and composite event detection with wireless embedded smart cameras,” IEEE Trans. Image Process., vol. 19, no. 10, pp. 2614–2633, Oct. 2010.
[8] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in Proc. Conf. Neural Inf. Process. Syst. (NeurIPS), Dec. 2015, pp. 1–9.
[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 580–587.
[10] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, “SSD: Single shot MultiBox detector,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Oct. 2016, pp. 21–37.
[11] J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 7263–7271.
[12] R. J. Wang, X. Li, and C. X. Ling, “Pelee: A real-time object detection system on mobile devices,” in Proc. Conf. Neural Inf. Process. Syst. (NeurIPS), Dec. 2018, pp. 1–10.
[13] S. Wang, C. Zhang, Y. Shu, and Y. Liu, “Live video analytics with FPGA-based smart cameras,” in Proc. Workshop Hot Topics Video Anal. Intell. Edges, Oct. 2019, pp. 9–14.
[14] B. Kwon and S. Lee, “Joint swing energy for skeleton-based gender classification,” IEEE Access, vol. 9, pp. 28334–28348, 2021.
[15] H. Liu, S. Chen, and N. Kubota, “Intelligent video systems and analytics: A survey,” IEEE Trans. Ind. Informat., vol. 9, no. 3, pp. 1222–1233, Aug. 2013.
[16] J. Li, X. Liang, S. Shen, T. Xu, J. Feng, and S. Yan, “Scale-aware fast R-CNN for pedestrian detection,” IEEE Trans. Multimedia, vol. 20, no. 4, pp. 985–996, Oct. 2018.
[17] P. Seremanet, K. Kavukcuoglu, S. Chintala, and Y. Lecun, “Pedestrian detection with unsupervised multi-stage feature learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2013, pp. 3626–3633.
[18] M. Khodabandeh, A. Vahdat, M. Ranjarb, and W. Macready, “A robust learning approach to domain adaptive object detection,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 480–490.
[19] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, “Adversarial discriminative domain adaptation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 7167–7176.
[20] M. Long, Z. Cao, J. Wang, and M. I. Jordan, “Conditional adversarial domain adaptation,” in Proc. Conf. Neural Inf. Process. Syst. (NeurIPS), Dec. 2018, pp. 1–11.
[21] S. Kim, J. Choi, T. Kim, and C. Kim, “Self-training and adversarial background regularization for unsupervised domain adaptive one-stage object detection,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 6092–6101.
[22] X. Wang, M. Wang, and W. Li, “Scene-specific pedestrian detection for static video surveillance,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 2, pp. 361–374, Feb. 2014.
[23] N. Irroue, R. Furuta, T. Yamasaki, and K. Aizawa, “Cross-domain weakly-supervised object detection through progressive domain adaptation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 5001–5009.
[24] L.-J. Li and L. Fei-Fei, “OPTIMOL: Automatic online picture collection via incremental model learning,” Int. J. Comput. Vis., vol. 88, no. 2, pp. 147–168, Jun. 2010.
[25] B. Collins, J. Deng, K. Li, and L. Fei-Fei, “Towards scalable dataset construction: An active learning approach,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Oct. 2008, pp. 86–98.
[26] A. Poms, W. Crichton, P. Hanrahan, and K. Fatahalian, “Scanner: Efficient video analysis at scale,” ACM Trans. Graph., vol. 37, no. 4, pp. 1–13, Aug. 2018.
[27] D. Kang, J. Emmons, F. Abuzaid, P. Bailis, and M. Zaharia, “NoScope: Optimizing neural network queries over video at scale,” 2017, arXiv:1703.02529.
[28] S. Han, H. Shen, M. Philipose, S. Agarwal, A. Wolman, and A. Krishnamurthy, “MCDNN: An approximation-based execution framework for deep stream processing under resource constraints,” in Proc. 14th Annu. Int. Conf. Mobile Syst., Appl., Services, Jun. 2016, pp. 123–136.
[29] J. Emmons, S. Fouladi, G. Ananthanarayanan, S. Venkataraman, S. Savarese, and K. Winston, “Cracking open the DNN black-box: Video analytics with DNNs across the camera-cloud boundary,” in Proc. Workshop Hot Topics Video Anal. Intell. Edges, Oct. 2019, pp. 27–32.
[30] J. Noh, S. Lee, B. Kim, and G. Kim, “Improving occlusion and hard negative handling for single-stage pedestrian detectors,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 966–974.
[31] K. K. Hikse and D. Hogg, “Adapting pedestrian detectors to new domains: A comprehensive review,” Eng. Appl. Artif. Intell., vol. 50, pp. 142–158, Apr. 2016.
[32] P. Sharma, C. Huang, and R. Nevatia, “Unsupervised incremental learning for improved object detection in a video,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 3298–3305.
[33] J. Friedman, T. Hastie, and R. Tibshirani, “Additive logistic regression: A statistical view of boosting (with discussion and a rejoinder by the authors),” Ann. Statist., vol. 28, no. 2, pp. 337–407, Apr. 2000.
[34] A. RoyChowdhury, P. Chakrabarty, A. Singh, S. Jin, H. Jiang, L. Cao, and E. Learned-Miller, “Automatic adaptation of object detectors to new domains using self-training,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 780–790.
[35] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, “Searching for MobileNetV3,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 1314–1324.
B. Kwon, T. Kim: Toward an Online Continual Learning Architecture for Intrusion Detection of Video Surveillance

[36] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no. 4, pp. 600–612, Apr. 2004.

[37] T. Kim, J. Kim, S. Kim, S. Cho, and S. Lee, “Perceptual crosstalk prediction on autostereoscopic 3D display,” IEEE Trans. Circuits Syst. Video Technol., vol. 27, no. 7, pp. 1450–1463, Jul. 2017.

[38] J. Ferryman and A. Shahroki, “PETS2009: Dataset and challenge,” in Proc. 12th IEEE Int. Workshop Perform. Eval. Tracking Surveill., Dec. 2009, pp. 1–6.

[39] B. Benfold and I. Reid, “Stable multi-target tracking in real-time surveillance video,” in Proc. CVPR, Jun. 2011, pp. 3457–3464.

[40] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal speed and accuracy of object detection,” 2020, arXiv:2004.10934.

[41] M. Tan, R. Pang, and Q. V. Le, “EfficientDet: Scalable and efficient object detection,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 10781–10790.

[42] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 2, pp. 318–327, Feb. 2020.

[43] J. Nie, R. M. Anwer, H. Cholakkal, F. S. Khan, Y. Pang, and L. Shao, “Enriched feature guided refinement network for object detection,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 9537–9546.

BEOM KWON was born in Seoul, South Korea, in 1989. He received the B.S. degree in electrical and electronic engineering from Soongsil University, Seoul, in 2012, and the M.S. and Ph.D. degrees in electrical and electronic engineering from Yonsei University, Seoul, in 2018. From March 2018 to September 2019, he was a Senior Researcher with the Agency for Defense Development (ADD), Daejeon, South Korea. From October 2019 to August 2021, he was a Staff Engineer with Samsung Electronics Company Ltd., Suwon, South Korea. Since September 2021, he has been an Assistant Professor with the Department of Artificial Intelligence, Dongyang Mirae University, Seoul. His research interests include artificial intelligence and it applications.

TAEWAN KIM received the B.S., M.S., and Ph.D. degrees in electrical and electronic engineering from Yonsei University, Seoul, South Korea, in 2008, 2010, and 2015, respectively. From 2015 to 2021, he was with the Vision AI Laboratory, SK Telecom, Seoul. In 2022, he joined as a faculty with the Division of Future Convergence (Data Science Major), Dongduk Women’s University, Seoul, where he is currently an Assistant Professor. His research interests include computer vision and machine learning including continual and online learning.