Foreground Objects Detection Using a Fully Convolutional Network With a Background Model Image and Multiple Original Images

JAE-YEUL KIM¹ AND JONG-EUN HA²

¹Graduate School of Automotive Engineering, Seoul National University of Science and Technology, Seoul 01811, South Korea
²Department of Mechanics and Automotive Engineering, Seoul National University of Science and Technology, Seoul 01811, South Korea

Corresponding author: Jong-Eun Ha (jeha@seoultech.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government. (MSIT) (2020R1A2C1013335).

ABSTRACT

Visual surveillance aims to reliably extract foreground objects. Traditional algorithms usually use a background model image which is generated through the probabilistic modeling of changes over time and space. They detect foreground objects by comparing a background model image with a current image. Hard shadows, illumination changes, camouflage, camera jitter, and ghost object motion make the robust detection of foreground objects difficult in visual surveillance. Recently, various methods based on deep learning have been applied to visual surveillance. It has been shown that deep learning approaches can stably extract salient features, and they give a superior result compared to traditional algorithms. However, they show a good performance only for scenes that are similar to a scene used in training. Without retraining on a new scene, they give a worse result compared to traditional algorithms. In this paper, we propose a stable foreground object detection algorithm through the integration of a background model image used in traditional methods and deep learning methods. A background model image generated by SuBSENSE and multiple images are used as the input of a fully convolutional network. Also, it is shown that it is possible to improve a generalization power by training the proposed network using diverse scenes from an open dataset. We show that the proposed algorithm can have a superior result compared to deep learning-based and traditional algorithms in a new scene without retraining the network. The performance of the proposed algorithm is evaluated using various datasets such as the CDnet 2014, SBI, LASEISTA, and our own datasets. The proposed algorithm shows improvement of 17.5%, 8.9%, and 4.3%, respectively, in FM score compared to three deep learning-based algorithms.

INDEX TERMS

Foreground objects detection, visual surveillance, generalization power, deep learning, fully convolutional network.

I. INTRODUCTION

Visual surveillance aims to find foreground objects that are distinct from a static background. Traditional approaches in visual surveillance usually first get a background model image, and it is compared to a current image to detect foreground objects. They consist of many steps, including initialization, representation, maintenance of a background model image, and foreground detection operation [1]–[3]. Hard shadows, illumination changes, camouflage, camera jitter, and ghost object motion, make the robust detection of foreground objects difficult in visual surveillance. Many researches have been done to cope with these problems in diverse directions. Recently, deep learning-based approaches have also been adopted in visual surveillance. Most deep learning-based algorithms train a network in end-to-end which require ground truth label of foreground objects, while most traditional algorithms do not require training samples. Visual surveillance algorithms based on deep learning have shown a huge improvement like other domains of image classification, detection, and recognition. However, deep learning-based approaches in visual surveillance require further improvements in two directions. One is that they need an improved generalization power in scenes that are different...
from trained environments. Domain transfer algorithms [48] are a good candidate to cope with this problem. The other is that they should require much less ground truth labels than current deep learning-based algorithms. In visual surveillance, ground truth labels of foreground objects require a designation per pixel, which require a more time compared to the case of image classification and object detection. Zero-shot learning [49] is a good candidate for solving this problem.

In this paper, we propose an algorithm to partially solve these two problems. First, the proposed algorithm integrates the advantage of traditional and deep-learning based algorithms. Traditional algorithms provide a background model image that is the abstraction of input images through a spatio-temporal analysis. Deep learning-based algorithms can extract more diverse features than traditional algorithms on images. Proposed algorithm uses a background model image from traditional approaches and multiple images including a current image as the input of a fully convolutional network. Second, the proposed algorithm is trained using multiple ground truth labels across different scenes while most deep learning-based approaches separately train a network for each scene. The proposed algorithm gives an improved performance than deep learning-based and traditional approaches, which is shown by experiments using diverse datasets such as CDnet 2014 [25], SBI [45], and LASIESTA [46].

II. RELATED RESEARCH

Many approaches including background subtraction and foreground objects detection for visual surveillance have been proposed. Good surveys are available [5]–[8]. We divide them into two groups, namely, approaches that do not use deep learning and those that are based on deep learning.

A. EARLIER APPROACHES: NON DEEP LEARNING BASED

Stauffer and Grimson [9] proposed an algorithm called mixture of Gaussian (MOG) that represent the brightness value of each pixel as the combination of multiple Gaussian distributions. The expectation and maximization (EM) algorithm [10] is used to determine the number of the Gaussian mixture and each parameter of Gaussian distribution. No special initialization is required because it adapts their parameters as a sequence goes on. Pixels are considered as background when their brightness values belong to the Gaussian mixture model, otherwise, they are considered as foreground.

Elgammal et al. [11] proposed a probabilistic non-parametric method using kernel density estimation. Barnich et al. [1] introduced a sample based method in background modeling. Samples from previous predefined frames are used in background modeling. If there is a predefined number of samples that are close to the current pixel, then it is considered as background, otherwise, it is considered as foreground. Kim et al. [12] proposed a method that uses a codebook. At the initial stage, codewords from intensity, color and temporal features are constructed. They build up a codebook for later segmentation. The current frame’s pixel values of intensity and color are compared to those of the codewords in the code book. Finally, a foreground or background label is assigned to each pixel by comparing the distance with codewords in the codebook. In the case of a background pixel, the matching codeword is updated. Oliver et al. [13] proposed a method based on principal component analysis, which is called the eigenbackground. The mean and the covariance matrix are computed using a predefined number of images. Here, N eigenvectors are chosen corresponding to the N largest eigenvalues, and they are used as the background model. Incoming images are projected into those eigenvectors, and their distance in those spaces are used to identify the foreground and background.

Wang et al. [14] proposed an algorithm that uses a Gaussian mixture model for the background and uses single Gaussian for the foreground. They employed a flux tensor [15] that can explain variations of optic flow within a local 3D spatio-temporal volume, and it is used in detecting blob motion. Foreground and background models are integrated to find moving and static foreground objects with information from blob motion. Also, edge matching [16] is used to classify static foreground objects as ghosts or intermediate motions. Varadarajan et al. [3] proposed a method that applies a region-based mixture of Gaussians for foreground object segmentation to cope with the sensitivity of the dynamic background. Chen et al. [18] proposed an algorithm that uses a mixture of Gaussians in a local region. At each pixel level, the foreground and background are modeled using a mixture of Gaussians. Each pixel is determined to be foreground or background by finding the highest probability of the center pixel among N x N region.

Sajid and Cheung [19] proposed an algorithm to cope with sudden illumination changes by using multiple background models through single Gaussians and different color representations. K-means clustering is used to classify the pixels of input images. For each pixel, K models are compared, and the group that shows the highest normalized cross-correlation is chosen. An RGB and YCbCr color frame is used, and segmentation is done for each color which yields six segmentation masks. Finally, background segmentation is performed by integrating all available segmentation masks. Hofmann et al. [20] proposed an algorithm that improves Barnich et al. [1]. They replace the global threshold R with an adaptive threshold R(x) that depends on the pixel location and a metric of the background model which is called background dynamics. The threshold R(x) and the model update rate are determined by a feedback loop using the additional information from the background dynamics. They showed that it can cope with a dynamic background and highly structured scenes. Tiefenbacher et al. [21] proposed an algorithm that improves the algorithm introduced by Hofmann et al. [20] by controlling the updates of the pixel-wise thresholds using a PID controller. St-Charles et al. [2] also proposed an improved algorithm by using local binary similarity patterns [23] as additional features of pixel intensities and slight
modification of the update mechanism of the thresholds and the background model.

B. DEEP LEARNING-BASED

Wang et al. [27] proposed multi-scale convolutional neural networks with cascade structure for background subtraction. Also, they trained a network for each video in the CDnet 2014 dataset. More recently, Lim et al. [28] proposed an encoder decoder type neural network for foreground segmentation called FgSegNet. It uses a pre-trained convolutional network of VGG-16 [29] as the encoding part with a triplet network structure. In the decoding part, a transposed convolutional network is used. Their network is trained by randomly selecting some training samples for each video in CDnet 2014.

Zeng et al. [30] proposed a multi-scale fully convolutional network architecture that takes advantage of various layer features for background subtraction. Zheng et al. [31] proposed an algorithm that combines traditional background subtraction and semantic segmentation [32]. The output of semantic segmentation is used to update the background model through feedback. Their result shows that it achieves the best performance among unsupervised algorithms in CDnet 2014. Sakkos et al. [33] presented a robust model that consists of a triple multi-task generative adversarial network (GAN) that can detect foreground even in exceptionally dark or bright scenes and in continuously varying illumination. They generate low and high brightness image pairs using the gamma function from a single image and use them in training by simultaneously minimizing GAN loss and segmentation loss.

The following algorithms are similar to the proposed algorithm in that they use multiple images as input to a convolutional neural network.

Varghese et al. [35] investigated visual change aiming to accurately identify variations between a reference image and a new test image. They proposed a parallel deep convolutional neural network for localizing and identifying the changes between image pairs. Patil et al. [34] proposed a motion saliency foreground network (MSFgNet) to estimate the background and to find the foreground in video frames. Original video frames are divided into a number of small video streams, and the background is estimated for each divided video stream. The saliency map is computed using the current video frame and the estimated background. Finally, an encoder decoder network is used to extract the foreground from the estimated saliency maps. Akilan et al. [36] proposed a 3D convolutional neural network with long short-term memory (LSTM) to include temporal information in a deep learning framework for background subtraction. Braham and Droogenbroeck [24] proposed the first scene specific convolutional neural network (CNN)-based algorithm for background subtraction. A fixed background model is generated by a temporal median operation over the first 150 video frames. Then, image patches centered on each pixel are extracted from both the current and background model images. The combined patches are used as the input of the trained CNN, and it outputs the probability of foreground. They evaluated their algorithm on the 2014 ChangeDetection.net dataset (CDnet 2014) [25]. The CNN requires training for each scene in the CDnet 2014 dataset. It requires a long computation time because patches from each pixel are required to pass the CNN, and it is similar to the sliding window approach in object detection. Babaee et al. [26] proposed a method that uses CNN to perform the segmentation of foreground objects, and they use a background model image which is generated by SuBSENSE [2] and Flux Tensor [14] algorithm. Spatial-median filtering is used for the post-processing of network outputs. The proposed algorithm use only SuBSENSE for the generation of a background model image, and it uses multiple images consisted of a background model image, a current image and past images as the input of a fully convolutional network. Yang et al. [22] proposed an algorithm that apply multiple images to fully convolutional networks (FCN). When selecting multiple input images, images close to a current image are selected more. Unlike Yang et al. [22], the proposed algorithm uses a background model image as well as multiple images while Yang et al. [22] only uses multiple images as the input of a network.

III. PROPOSED METHOD

Before explaining the proposed method, we investigate problems that occurs in traditional visual surveillance algorithms which use a background model image. Figure 1 shows generated background model images by SuBSENSE [2], PAWCS [47], and temporal median filtering using scenes in the CDnet 2014 dataset. For the temporal median filtering method, 150 input images were used. SuBSENSE and PAWCS are one of the best performing methods among traditional visual surveillance methods, and they automatically generate a background model image.

In most cases, SuBSENSE and PAWCS generate a background model image which is suitable for detecting foreground objects, as shown in Figure 1. In the case of the temporal median filtering method, the duration time of foreground objects directly affect the generation of a background model image. In the case of winterDriveway, all three algorithms gives a wrong background model image. If a generated background model image is perfect, detecting foreground objects can be solved without difficulty. However, it is difficult to generate a proper background model image due to difficulties caused by a change in lighting conditions, a varying moving speed of foreground objects, and a time dependence of changes. We can conclude that traditional background modeling algorithms cannot cope with all variations that occur during visual surveillance. The proposed algorithm solve these difficulties by integrating traditional background modeling approaches with recent deep learning methods. Deep learning-based methods have demonstrated that they are efficient in extracting meaningful spatial features on a given input image. In the case of visual surveillance, it is necessary to extract not only spatial features but also temporal...
features that are latent on changes over time. To this end, the proposed algorithm integrates the power of traditional background modeling algorithms and recent deep learning-based algorithms. We use a background model image and multiple original images as the input of a fully convolutional network. The modeling power of a background model image and the detecting ability of spatio-temporal variation on image sequence by deep learning are integrated in a.
unified fashion using the fully convolutional network. Also, we show the generalization power of the proposed algorithm by applying to scenes not used in the training.

We use a background model image and multiple original images including a current image as input to the fully convolutional network. A background model image by SuBSENSE and PAWCS typically have multiple samples per pixel. In this paper, we adopt SuBSENSE to generate a background model image, and 50 samples are used in the generation of a background model image. Figure 2 shows the structure of the proposed algorithm. Three inputs with different characteristics are stacked, and they are used as the input of the fully convolutional network. Figure 3 shows the structure of a fully convolutional neural network for the detection of foreground objects, which adopt U-Net [40] structure. SuBSENSE is run in parallel to the proposed algorithm as shown in Figure 2. SuBSENSE alone can’t cope with the diverse variation which occurs during visual surveillance. The proposed algorithm makes it possible to cope with errors which might be included in a background model image generated by SuBSENSE using not only a background model image, but also a current image and past images as input to the network. A total of six images composed of one background model image and five original images are stacked, and they are used as the input of fully convolutional network. A background model image, a current image and past images are converted to gray image. The size of input image to the network is 320(W)×240(H)×6(C). The size of output image by the network is 320(W)×240(H)×1(C), which corresponds to a segmentation map. The kernel window size of 3×3 is used for all layers, and ReLU is used as an activation function except the last output. Batch normalization is applied before the activation function. Sigmoid is used as the activation function in the last output stage which generate an output between 0 and 1, and we interpret it as the probability of foreground objects.

IV. EXPERIMENTAL RESULTS

Experiments are done using various public dataset for visual surveillance with ground truth foreground maps, which makes it possible to quantitatively evaluate visual surveillance algorithms. We used the CDnet 2014 dataset, SBI dataset, and LASIESTA dataset in experiments. Also, we test the proposed algorithm using images acquired ourselves.

Binary cross entropy is used as loss and Adam optimizer [43] was used for finding a solution. He initialization [42] was used as a kernel initializer. The initial value of a learning rate was set to 0.001. If a validation loss does not decrease more than 5 times, a learning rate
is halved. Training is terminated when a learning rate does not decrease more than 10 times. The proposed algorithm is implemented using Keras [41]. The proposed model consists of 31,061,957 parameters, of which 31,048,261 are learnable parameters. The size of a model was chosen by considering the requirement that visual surveillance could operate in 30 fps. Computation time is 24ms in Intel i7-7820X, NVIDIA RTX 2080 Ti environment, and it takes 32ms in Intel i7-7700K, NVIDIA GTX 1080 environment.

A variety of metrics that are widely used in visual surveillance, namely, recall, precision, F-measure (FM), and percentage of wrong classification (PWC) are used to evaluate a performance. Precision, recall, FM, and PWC are defined as follows:

\[
\text{Precision} = \frac{TP}{TP + FP} \\
\text{Recall} = \frac{TP}{TP + FN} \\
\text{FM} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \\
\text{PWC} = \frac{FP + FN}{TP + TN + FP + FN} \times 100
\]

where TP, TN, FP, FN each means true positive, true negative, false positive and false negative. TP means that a ground truth foreground pixel is detected as a foreground one. TN means that a ground truth background pixel is detected as a background one. FP means that a ground truth background pixel is detected as a foreground one. FN means that a ground truth foreground pixel is detected as a background one.

The CDnet 2014 dataset consists of 53 scenes under 11 categories. In the CDnet 2014 dataset, images of the PTZ category were acquired during the movement of a camera, and it deviates from a fixed camera which is usual in visual surveillance. Therefore, in this paper, experiments were done except for the PTZ category. The proposed algorithm as well as other algorithms also show significantly a low performance for the PTZ category. The evaluation of the proposed algorithm is done by using scenes not used in training. After training using CDnet 2014 dataset, it was applied to the SBI dataset and LASIESTA dataset without retraining using those scenes, and it was done to see the generalization ability of the proposed algorithm.

The entire experiments can be categorized as follows.

1. Comparison with three deep learning-based visual surveillance algorithms
2. Comparison with two traditional visual surveillance algorithms
3. Investigation of generalization power: Training using one dataset and applying it to different datasets without retraining

### A. HYPER-PARAMETERS SELECTION

Experiments are done to choose a suitable background modeling algorithm for the proposed algorithm. First, we present experimental results only using a background model image and a current image as the input of the network. Three algorithms such as SuBSENSE [2], PAWCS [47], and temporal median filtering are considered as candidates of a background model image generation. For temporal median filtering method, 150 images were used for a background model image generation.

Training was done using some scenes in the CDnet 2014 dataset, and test statistics for comparison was obtained by applying them to scenes not used in training. Among total 53 scenes in the CDnet 2014 dataset, 25 scenes are used in training. We used 200 images from each scene, which amount to 5,000 images in total. Among the 5,000 images, 4,000 images were used as training images and remaining 1,000 images were used for validation during training. After training, test statistics are obtained using remaining 24 scenes in the CDnet 2014 dataset which are not used in training. Table 1 shows comparison result according to a different background model generation method. As shown in Table 1, SuBSENSE shows the best performance while temporal median filtering gives the worst performance. SuBSENSE algorithm was chosen for the generation of the background model image by reflecting experimental results.

**Table 1. Comparison of FM score according to different background generation method.**

| CDnet 2014 Category | Current + SuBSENSE BG | Current + PAWCS BG | Current + Median BG |
|---------------------|-----------------------|-------------------|---------------------|
| FM                  | 0.9622                | 0.9712            |
| Camera Jitter       | 0.6330                | 0.5202            | 0.6012              |
| Bad Weather         | 0.9399                | 0.8796            | 0.8978              |
| Dynamic BG          | 0.7381                | 0.7101            | 0.6053              |
| Int. Obj. Motion    | 0.6498                | 0.6077            | 0.5484              |
| Low Frameate        | 0.9489                | 0.9237            | 0.9323              |
| Night               | 0.8292                | 0.8166            | 0.8372              |
| Shadow              | 0.9500                | 0.9044            | 0.7667              |
| Thermal             | 0.8676                | 0.7978            | 0.7264              |
| Turbulence          | 0.9402                | 0.9115            | 0.8034              |
| Average             | 0.8374                | 0.7961            | 0.7557              |

Next, we show experimental results according to different types of input images to the network. Table 2 shows comparison results by three cases. The first one is the proposed algorithm which uses stacked images of a background model image and current image as the input of the network.
image, a current image and past images as the input of the network. The second one uses a background model image and a current image as the input of the network. The third one uses a current image and multiple past images as the input of the network. In experiments, four past images which are selected under 25 frame intervals are used. Only using multiple images as the input of the network gives the worst result among three cases. The proposed algorithm gives the best result as shown in Table 2. Figure 4 shows the comparison result of loss of three cases. Figure 4(a) shows a training loss and Figure 4(b) shows a validation loss. All three methods show a similar tendency.

From experimental results of Table 1 and Table 2, we showed that the proposed algorithm which uses inputs from different domains could cope with diverse variations occur in visual surveillance. If we use only a background model image and a current image as the input of the network, experimental result shows that it has a margin for improvement. Finally, adding multiple previous images in addition to a background model image and a current image as the input of the network gives a dramatic improvement. We can conclude that spatio-temporal features are well reflected in the proposed algorithm.

The number of previous images and the spacing between images are hyper-parameters, and they are chosen through experiments. Table 3 shows FM values according to an interval between two consecutive images. When using the CDNet 2014 dataset, 25 frame intervals provided the best result. Table 4 shows experimental results according to the number of past images under the same interval of 25 frames. Setting the number of past images as 4 shows the best performance. Through these experiments, in the case of the proposed algorithm, the interval between input images was set as 25 frames.

TABLE 3. The comparison result by the interval between two images (FM score).

| CDNet 2014 Category       | 1     | 2     | 5     | 10    | 25    | 50    | 100   |
|---------------------------|-------|-------|-------|-------|-------|-------|-------|
| Baseline                  | 0.9439| 0.9474| 0.9511| 0.9565| 0.9625| 0.9685| 0.9647|
| Camera Jitter             | 0.7089| 0.7224| 0.7223| 0.6925| 0.6990| 0.6655| 0.6365|
| Bad Weather               | 0.8991| 0.8805| 0.8852| 0.9235| 0.9417| 0.9319| 0.9501|
| Dynamic BG                | 0.8280| 0.8616| 0.8423| 0.8613| 0.8767| 0.8584| 0.8312|
| Intermittent Obj Motion   | 0.6105| 0.6042| 0.6113| 0.6228| 0.6163| 0.6321| 0.6466|
| Low Framerate             | 0.8943| 0.9108| 0.9212| 0.9304| 0.9422| 0.9434| 0.9498|
| Night                     | 0.8333| 0.8446| 0.8457| 0.8409| 0.8400| 0.8249| 0.8535|
| Shadow                    | 0.9500| 0.9446| 0.9515| 0.9558| 0.9576| 0.9600| 0.9563|
| Thermal                   | 0.8297| 0.7902| 0.8069| 0.8511| 0.8775| 0.8651| 0.8736|
| Turbulence                | 0.9292| 0.9215| 0.9274| 0.8924| 0.9107| 0.9402| 0.9497|
| Average                   | 0.8365| 0.8380| 0.8409| 0.8473| 0.8558| 0.8523| 0.8508|

TABLE 4. The comparison result by number of PAST images to the network (FM score).

| CDNet 2014 Category       | 1       | 2       | 3       | 4       | 6       | 8       | 10      | 12      |
|---------------------------|---------|---------|---------|---------|---------|---------|---------|---------|
| Baseline                  | 0.9663  | 0.9652  | 0.9620  | 0.9625  | 0.9614  | 0.9665  | 0.9621  | 0.9608  |
| Camera Jitter             | 0.6765  | 0.6964  | 0.6882  | 0.6990  | 0.6955  | 0.7088  | 0.7122  | 0.7134  |
| Bad Weather               | 0.9481  | 0.9152  | 0.9148  | 0.9417  | 0.9297  | 0.9122  | 0.9301  | 0.9209  |
| Dynamic BG                | 0.7886  | 0.8256  | 0.8315  | 0.8767  | 0.8427  | 0.8600  | 0.8192  | 0.8168  |
| Intermittent Obj Motion   | 0.6426  | 0.6336  | 0.6204  | 0.6163  | 0.6270  | 0.6198  | 0.6372  | 0.6032  |
| Low Framerate             | 0.9506  | 0.9344  | 0.9350  | 0.9422  | 0.9377  | 0.9383  | 0.9431  | 0.9236  |
| Night                     | 0.8381  | 0.8377  | 0.8328  | 0.8400  | 0.8402  | 0.8427  | 0.8298  | 0.8404  |
| Shadow                    | 0.9548  | 0.9616  | 0.9557  | 0.9576  | 0.9562  | 0.9552  | 0.9572  | 0.9566  |
| Thermal                   | 0.8817  | 0.8485  | 0.8851  | 0.8775  | 0.8586  | 0.8582  | 0.8381  | 0.7797  |
| Turbulence                | 0.9428  | 0.9284  | 0.9357  | 0.9107  | 0.9295  | 0.9347  | 0.9231  | 0.8634  |
| Average                   | 0.8501  | 0.8480  | 0.8468  | 0.8558  | 0.8507  | 0.8529  | 0.8478  | 0.8323  |
and the total number of past images was set to be 4. Results in Table 3 and Table 4 are obtained by averaging scores after we train the network using different initial values in three times.

**B. COMPARISON WITH THREE DEEP LEARNING-BASED ALGORITHMS**

The proposed algorithm is compared to three deep learning-based visual surveillance algorithms [24], [26], [44]. All three methods trained networks using multiple scenes in the CDnet 2104 dataset. The proposed algorithm was trained using the same scenes specified in each paper for comparison, and test statistics described in papers were used.

Patil et al. [44] proposed MsEDNet consists of two steps of a background model image generation by a temporal histogram technique and applying an encoder-decoder network to detect foreground objects. They first generate a saliency map using a background model image, and it is fed into the encoder-decoder network to produce a foreground segmentation map, while the proposed algorithm directly uses a
background model image and multiple original images as the input of the network.

An algorithm proposed by Braham and Droogenbroeck [24] is one of earlier approaches which adopt deep learning in visual surveillance. They generate a background model image by a temporal median filtering, and image patches and corresponding background patches centered on each pixel are used as the input of a network to yield foreground objects probability per pixel.

Babaei et al. [26] used a background model image and a current image as the input of a network similar to the proposed algorithm. The proposed algorithm additionally use multiple past images. Also, the network structure of Braham and Droogenbroeck [24] and Babaei et al. [26] is similar to the LeNet-5 [17] which includes a fully connected layer, while the proposed algorithm adopts the network structure of a fully convolutional network.

First, the proposed algorithm is compared to Babaei et al. [26], which used 39 scenes in the CDNet 2014 dataset for training. For each scene, 150 and 20 images are used for training and validation, respectively. After training, test statistics are obtained using 53 scenes in the CDNet 2014 dataset. Table 5 shows the comparison result by the proposed method and Babaei et al. [26]. The proposed algorithm gives an improved result in all categories in the CDNet 2014 dataset. The proposed algorithm gives FM score of 0.8871 while Babaei et al. [26] gives FM score of 0.7548, which amounts to 17.5% improvement. Figure 5 shows comparison results by the proposed algorithm, Babaei et al [26],
and SuBSENSE [2]. The CDnet 2014 dataset provides the ground truth labels of foreground maps only for some images. In Figure 5, images which do not have the ground truth label are shown as a gray image.

Second, the proposed algorithm is compared to Patil et al. [44]. It trained a network using 5,500 training images and 12,384 images are used for validation from 11 categories in the CDnet 2014 dataset. The proposed algorithm was trained using different number of training images. One is trained using the same number of images like Patil et al. [44], and the other is trained using a fewer number of images compared to Patil et al. [44]. Table 6 shows the comparison result. Proposed algorithm shows a comparable FM score only using 980 images for training while Patil et al. [44] uses 5500 images for training. In the case of using a similar number of images for training, the proposed algorithm gives FM score of 0.9788 compared to FM score of 0.8988 by Patil et al. [44], which amounts to 8.9% improvement.

| CDnet 2014 Category | Proposed (980 Train, 245 Val) | Proposed (5,390 Train, 4,410 Val) | MeEDNET [44] (5,500 Train, 12,834 Val) |
|---------------------|-------------------------------|----------------------------------|---------------------------------------|
| FM                  | FM                            | FM                              | FM                                    |
| Baseline            | 0.9622                        | 0.9895                          | 0.9368                                |
| Camera Jitter       | 0.8588                        | 0.9716                          | 0.9173                                |
| Bad Weather         | 0.9050                        | 0.9727                          | 0.9055                                |
| Dynamic BG          | 0.8817                        | 0.9675                          | 0.9068                                |
| Intermittent Obj Motion | 0.7933                   | 0.9825                          | 0.8187                                |
| Shadow              | 0.9450                        | 0.9903                          | 0.9196                                |
| Thermal             | 0.8254                        | 0.9765                          | 0.8825                                |
| Average             | 0.8816                        | 0.9788                          | 0.8988                                |

Third, comparison was performed to Braham and Droogenbroeck [24]. It trained a network using the first half of the ground truth images provided by the CDnet.
2014 dataset, and evaluation was done using the other half of ground truth images. The proposed algorithm is trained and tested using the same scenes noted in Braham and Droogenbroeck [24]. Table 7 shows a comparison result by the proposed algorithm and Braham and Droogenbroeck [24]. For all scenes except the dynamic background, the proposed algorithm gives superior result than Braham and Droogenbroeck [24]. The FM score by the proposed algorithm and Braham and Droogenbroeck [24] is 0.9433 and 0.9046, which amounts to 4.3% improvement.

### C. COMPARISON WITH TRADITIONAL ALGORITHMS

We compared the proposed algorithm to SuBSENSE [2] and PAWCS [47], which show a top performance among traditional visual surveillance algorithms. Table 8 shows scenes used for training and scenes used for testing in the CDnet 2014 dataset. 160 and 40 images are used for training and validation per each scene. In total, 4000 images are used for training and 1000 images are used for validation. The test statistics are obtained using scenes not used in training. Table 9 shows comparison results by the proposed algorithm and two classic algorithms. The proposed algorithm was
TABLE 8. The scenes in CDnet 2014 dataset used for the training of the proposed algorithm.

| Category              | Scenes used in training | Scenes used in test |
|-----------------------|-------------------------|---------------------|
| Baseline              | Office, PETS2006        | Highway, Pedestrians |
| Camera Jitter         | Badminton, Sidewalk     | Traffic, Boulevard   |
| Bad Weather           | Wet snow, Blizzard     | Skating, Snowfall   |
| Dynamic Background    | Boats, Canoe, Fountain1 | Fountain2, Fall, Overpass |
| Intermittent Object Motion | Street light, Parking, Winter drive way | Abandoned box, Sofa, Train stop |
| Low Frame Rate        | Port_0_17 fps, Tunnel exit 0_35fps | Tram crossroad_1fps, Turnpike_0_5fps |
| Night Videos          | Bridge entry, Busy boulevard, Fluid highway | Street corner at night, Tram station, Winter street |
| Shadow                | Copy machine, Bungalows, Bus station | Back door, Cubicle, People in shade |
| Thermal               | Corridor, Library, Lakeside | Dining room, Park |
| Turbulence            | Turbulence0, Turbulence1 | Turbulence2, Turbulence3 |

trained using 25 scenes shown in Table 8, and test statistics in Table 9 was obtained by applying to 28 scenes not used for training. We can notice that the proposed algorithm gives a slightly improved result compared to two classic algorithms. The proposed algorithm can reach the performance of classical algorithms by training using only 5% data in the CDnet 2014 dataset.

D. EVALUATION OF GENERALIZATION POWER

We present experimental results by applying proposed algorithm trained using the CDnet 2014 dataset to the SBI [45] and LASIESTA [46] datasets without retraining. Throughout this experiment, we want to investigate the generalization power of the proposed algorithm. Since the introduction of deep learning methods in visual surveillance, it is generally known that they provide a superior result compared to traditional methods. However, many training images having ground truth labels are required in order to guarantee such an improved performance. In the case of visual surveillance, ground truth label requires to be specified per pixel whether it is a foreground or a background, which is a time consuming operation. The advantage of traditional non-deep learning algorithms is that they do not require these training images. In the case of the proposed algorithm, after training using CDnet 2014 dataset, it is applied to other environments such as the SBI, LASIESTA, and self-obtained dataset without retraining to check a generalization power.

The proposed algorithm was trained using 36 scenes in 7 categories, excluding camera jitter, PTZ, thermal, and turbulence categories in the CDnet 2014 dataset. The total number of images used for training was 61593. Among them, 49261 images were used for training and 12332 images were used for validation. In the case of the Toscana scene of the SBI dataset, it consists of only six images that are not continuous in time, so it is very different from general visual surveillance dataset composed of continuous images over time. Therefore, the Toscana scene was excluded from the evaluation of the SBI dataset. Also, we do not consider a situation where a camera is moving. Therefore, in the LASIESTA dataset, results were compared only in 20 scenes where a camera is fixed, excluding scenes where a camera is moving.

TABLE 10. The comparison result by the proposed algorithm and traditional algorithms using SBI dataset [45].

| SBI Scene | Proposed | PAWCS [47] | SuBSENSE [2] |
|-----------|----------|------------|--------------|
| Board     | 0.9291   | 0.7798     | 0.5777       |
| CIVIAR1   | 0.9656   | 0.8589     | 0.9144       |
| CIVIAR2   | 0.9268   | 0.6772     | 0.8714       |
| CaVignal  | 0.4489   | 0.3697     | 0.3980       |
| Candela   | 0.7374   | 0.8725     | 0.5356       |
| Hall & Monitor | 0.9741 | 0.7411     | 0.7758       |
| Highway1  | 0.9266   | 0.7015     | 0.5523       |
| Highway2  | 0.9633   | 0.9031     | 0.8937       |
| Human Body2 | 0.9396 | 0.7013     | 0.8346       |
| IBM TEST2 | 0.9789   | 0.9386     | 0.9390       |
| People & Foliage | 0.6191 | 0.3162     | 0.2660       |
| Snellen   | 0.4699   | 0.0344     | 0.1403       |
| Foliage   | 0.5198   | 0.1153     | 0.1435       |
| Average   | 0.7999   | 0.6161     | 0.6033       |

Table 10 and Table 11 show a comparison result by the proposed algorithm and two classical algorithms using the SBI dataset and LASIESTA dataset, respectively. The SBI dataset and LASIESTA dataset were not used in the training of the proposed algorithm. We can notice that the proposed algorithm provides a better performance than two classical algorithms even without retraining using new scenes in the SBI and LASIESTA dataset.

Figure 6 and Figure 7 show a comparison result on the SBI dataset and LASIESTA dataset, respectively. From Figure 6, we can notice that SuBSENSE [2] generates a background model image having partial errors in Board #180, Hall & Monitor #220, and Highway1 # 122. Nevertheless, the proposed algorithm gives an improved foreground map. Also, traditional algorithms detect hard shadow regions as foreground objects as shown in highway1 #122 while the proposed algorithm gives an improved foreground map. But, the proposed algorithm cannot cope with ghost objects as
in the case of cavignal #200 where SuBSENSE [2] gives a wrong background model image. From Figure 7, we can notice that the proposed algorithm gives an improved result although SuBSENSE [2] gives background model images having partial foreground objects in I_BS_01 #78, I_CA_01 #307, and I_CA_02 #248. The proposed algorithm gives a more accurate detection than traditional algorithms using the same background model image in I_CA_01 and I_CA_02 of Figure 7 which correspond to camouflage. Also, the proposed algorithm gives a more stable detection that traditional algorithms using the same background model image in I_IL_01 of Figure 7 which has illumination changes. In O_SU_01, O_SU_02 of Figure 7, the proposed algorithm can cope with shadow more stable way than traditional algorithms. For LASIESTA dataset, the proposed algorithm cannot cope with ghost objects when SuBSENSE gives a wrong background model image as in the case of SBI dataset. The proposed algorithm gives a more stable result in hard shadow, illumination changes, and camouflage than traditional algorithms, as shown in Figure 6 and Figure 7.

Lastly, the performance of the proposed algorithm was compared using images taken ourselves. Two outdoor scenes and three indoor scenes are acquired, and they are used for comparison. An indoor scene is acquired from a laboratory and a corridor. The manual adjustment of lighting inside a laboratory was done to vary illumination. Figure 8 and Figure 9 show a comparison result by the proposed algorithm, SuBSENSE, and PAWCS using indoor and outdoor video sequences acquired ourselves. The ground truth foreground map are manually acquired for some images to compare

### TABLE 11. The comparison result by the proposed algorithm and traditional algorithms using LASIESTA dataset [46].

| LASIESTA scene | Proposed | PAWCS [47] | SuBSENSE [2] |
|----------------|----------|------------|--------------|
| I BS 01        | 0.8360   | 0.6568     | 0.6140       |
| I BS 01        | 0.7688   | 0.7062     | 0.6819       |
| I CA 01        | 0.9006   | 0.9600     | 0.9412       |
| I CA 02        | 0.9056   | 0.7596     | 0.7992       |
| I_SI 01        | 0.9673   | 0.9413     | 0.9238       |
| I_SI 02        | 0.9796   | 0.9554     | 0.8934       |
| I IL 01        | 0.9538   | 0.6649     | 0.2413       |
| I IL 02        | 0.9645   | 0.7334     | 0.6957       |
| I MB 01        | 0.9392   | 0.9627     | 0.9607       |
| I MB 02        | 0.9233   | 0.9011     | 0.8799       |
| I OC 01        | 0.9881   | 0.9593     | 0.9176       |
| I OC 02        | 0.9696   | 0.9528     | 0.9321       |
| O_RA 01        | 0.9449   | 0.9296     | 0.8718       |
| O_RA 02        | 0.9803   | 0.9384     | 0.8794       |
| O_CL 01        | 0.9858   | 0.9821     | 0.9555       |
| O_CL 02        | 0.9877   | 0.9620     | 0.8754       |
| O_SN 01        | 0.9885   | 0.9161     | 0.8813       |
| O_SN 02        | 0.8822   | 0.4544     | 0.7036       |
| O_SU 01        | 0.9683   | 0.8397     | 0.7947       |
| O_SU 02        | 0.9805   | 0.8506     | 0.7890       |
| Average        | 0.9429   | 0.8513     | 0.8113       |
results in qualitative manner. The result by the proposed algorithm was obtained using training result on the CDNet 2014 dataset without retraining it using new sequences. The proposed algorithm gives stable detection of foreground objects, in particular, in the case of indoor illumination change, as shown in Figure 8, while SuBSENSE and PAWCS algorithms fail in detecting foreground objects.

V. CONCLUSION

In this paper, a visual surveillance algorithm is proposed which combines a traditional and a recent deep learning-based algorithm. A fully convolutional network which uses a background model image, a current image, and multiple past images as the input of the network is proposed. A background model image is updated using SuBSENSE algorithm. We could have an improved performance by combining images having different characteristics as the input of a network. Also, training was done using diverse scenes from public dataset, while most conventional deep learning-based algorithms train a network for each scene. From these two factors, the proposed algorithm could have an improved performance in unknown scenes compared to deep learning-based and traditional algorithms without retraining the network. Experimental results using public dataset such as the CDNet 2014, SBI, LASIESTA and self-obtained images support the improvement by the proposed algorithm. The proposed algorithm gives a more stable detection in hard shadow, illumination change, and camouflage than traditional algorithms. But, the proposed algorithm gives a poor detection in ghost objects, which is caused by a wrong background model by SuBSENSE. For further study, we are planning to generate a background model image by deep learning.

REFERENCES

[1] O. Barnich and M. Van Droogenbroeck, “ViBe: A universal background subtraction algorithm for video sequences,” IEEE Trans. Image Process., vol. 20, no. 6, pp. 1709–1724, Jun. 2011.
[2] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, “SuBSENSE: A universal change detection method with local adaptive sensitivity,” IEEE Trans. Image Process., vol. 24, no. 1, pp. 359–375, Jan. 2015.
[3] S. Varadarajan, P. Miller, and H. Zhou, “Region-based mixture of gaussians modelling for foreground detection in dynamic scenes,” Pattern Recognit., vol. 48, no. 11, pp. 3488–3503, Nov. 2015.
[4] Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar, “CDNet 2014: An expanded change detection benchmark dataset,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, Jun. 2014, pp. 387–394.
[5] T. Bouwmans, “Traditional and recent approaches in background modeling for foreground detection: An overview,” Comput. Sci. Rev., vol. 11−12, pp. 31−66, May 2014.
[6] A. Sobral and A. Vacavant, “A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos,” Comput. Vis. Image Understanding., vol. 122, pp. 4−21, May 2014.
[7] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, “A review on deep learning techniques applied to semantic segmentation,” 2017, arXiv:1704.06857. [Online]. Available: http://arxiv.org/abs/1704.06857
[8] T. Bouwmans, S. Javed, M. Sultana, and S. Ki Jung, “Deep neural network concepts for background subtraction: A systematic review and comparative evaluation,” 2018, arXiv:1811.05255. [Online]. Available: http://arxiv.org/abs/1811.05255
[9] C. Stauffer and W. E. L. Grimson, “Adaptive background mixture models for real-time tracking,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Jun. 1999, pp. 246–252.
[10] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” J. Roy. Statist. Soc., B, Methodol., vol. 39, no. 1, pp. 1–38, 1977.
[11] A. Elgammal, R. Duraiswami, D. Harwood, and L. S. Davis, “Background and foreground modeling using nonparametric kernel density estimation for visual surveillance,” Proc. IEEE, vol. 90, no. 7, pp. 1151–1163, Jul. 2002.
[12] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, “Real-time foreground-background segmentation using codebook model,” Real-time Imag., vol. 3, pp. 172–185, 2005.
[13] N. M. Oliver, B. Rosario, and A. P. Pentland, “A Bayesian computer vision system for modeling human interactions,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 8, pp. 831–843, Aug. 2000.
[14] R. Wang, F. Bunyak, G. Seetharaman, and K. Palaniyappan, “Static and moving object detection using flux tensor with split Gaussian models,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, Jun. 2014, pp. 414–418.
[15] F. Bunyak, K. Palaniyappan, S. K. Nath, and G. Seetharaman, “Flux tensor constrained geodesic active contours with sensor fusion for persistent object tracking,” J. Multimedia, vol. 2, no. 4, pp. 20–33, Aug. 2007.
[16] R. Heras Evangelio and T. Sikora, “Complementary background models for the detection of static and moving objects in crowded environments,” in Proc. 8th IEEE Int. Conf. Adv. Video Signal Based Survell. (AVSS), Aug. 2011, pp. 71–76.
[17] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[18] Y. Chen, J. Wang, and H. Lu, “Learning sharable models for robust background subtraction,” in Proc. IEEE Int. Conf. Multimedia Expo (ICME), Jun. 2015, pp. 1–6.
[19] H. Sajid and S.-C. Samson Cheung, “Background subtraction for static & moving camera,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Sep. 2015, pp. 4530–4534.
[20] M. Hofmann, P. Tiefenbacher, and G. Rigoll, “Background segmentation with feedback: The pixel-based adaptive segmenter,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops, Jun. 2012, pp. 38–43.
[21] P. Tiefenbacher, M. Hofmann, D. Merget, and G. Rigoll, “PID-based regulation of background dynamics for foreground segmentation,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Oct. 2014, pp. 3282–3286.
[22] L. Yang, J. Li, Y. Luo, Y. Zhao, H. Cheng, and J. Li, “Deep background modeling using fully convolutional network,” IEEE Trans. Intell. Transp. Syst., vol. 19, no. 1, pp. 254–262, Jan. 2018.
[23] G.-A. Bilodeau, J.-P. Jodoin, and N. Saumier, “Change detection in feature space using local binary similarity patterns,” in Proc. Int. Conf. Comput. Robot Vis., May 2013, pp. 106–112.
[24] M. Braham and M. Van Droogenbroeck, “Deep background subtraction with scene-specific convolutional neural networks,” in Proc. Int. Conf. Syst. Signals Image Process. (IWSSIP), May 2016, pp. 1–4.
[25] N. Goyette, P.-M. Jodoin, F. Porikli, J. Konrad, and P. Ishwar, “A novel video dataset for change detection benchmarking,” IEEE Trans. Image Process., vol. 23, no. 11, pp. 4663–4679, Nov. 2014.
[26] M. Babaei, D. T. Dinh, and G. Rigoll, “A deep convolutional neural network for video sequence background subtraction,” Pattern Recognit., vol. 76, pp. 635–649, Apr. 2018.
[27] Y. Wang, Z. Luo, and P.-M. Jodoin, “Interactive deep learning method for segmenting moving objects,” Pattern Recognit. Lett., vol. 96, pp. 66–75, Sep. 2017.
[28] L. Ang Lim and H. Yalim Keles, “Foreground segmentation using a triplet convolutional neural network for multiscale feature encoding,” 2018, arXiv:1801.02225. [Online]. Available: http://arxiv.org/abs/1801.02225
[29] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, arXiv:1409.1556. [Online]. Available: http://arxiv.org/abs/1409.1556
[30] D. Zeng and M. Zhu, “Background subtraction using multiscale fully convolutional network,” IEEE Access, vol. 6, pp. 16010–16021, 2018.
[31] D. Zeng, X. Chen, M. Zhu, M. Goesele, and A. Kuijper, “Background subtraction with real-time semantic segmentation,” IEEE Access, vol. 7, pp. 153869–153884, 2019.
[32] M. Braham, S. Pierard, and M. Van Droogenbroeck, “Semantic background subtraction,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Sep. 2017, pp. 4552–4556.
[33] D. Sakkos, E. S. L. Ho, and H. P. H. Shum, “Illumination-aware multi-task GANs for foreground segmentation,” IEEE Access, vol. 7, pp. 10976–10986, 2019.
[34] P. W. Patil and S. Murala, “MSFgNet: A novel compact End-to-End deep network for moving object detection,” IEEE Trans. Intell. Transp. Syst., vol. 20, no. 11, pp. 4066–4077, Nov. 2019.
[35] A. Varghese, J. Gubbi, A. Ramaswamy, and P. Balamuralidhar, “ChangeNet: A deep learning architecture for visual change detection,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 120–145.
[36] T. Akilan, Q. J. Wu, A. Safaei, J. Huo, and Y. Yang, “A 3D CNN-LSTM-Based Image-to-Image foreground segmentation,” IEEE Trans. Intell. Transp. Syst., vol. 21, no. 3, pp. 959–971, Mar. 2020, doi: 10.1109/TITS.2019.2900426.
[37] L. Ang Lim and H. Yalim Keles, “Learning multi-scale features for foreground segmentation,” 2018, arXiv:1808.01477. [Online]. Available: http://arxiv.org/abs/1808.01477.
[38] M. D. Gregorio and M. Giordano, “A WiSARD-based approach to CDnet,” in Proc. BRICS Congr. Comput. Intell. 11th Brazilian Congr. Comput. Intell., Sep. 2013, pp. 172–177.
[39] M. Sedky, M. Moniri, and C. C. Chibelushi, “Spectral-360: A physics-based technique for change detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, Jun. 2014, pp. 405–408.
[40] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent., Sep. 2015, pp. 234–241.
[41] F. Chollet. (2015). Keras. [Online]. Available: https://github.com/fchollet/keras
[42] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification,” 2015, arXiv:1502.01852. [Online]. Available: http://arxiv.org/abs/1502.01852.
[43] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980. [Online]. Available: http://arxiv.org/abs/1412.6980.
[44] P. W. Patil, S. Murala, A. Dhall, and S. Chaudhary, “MsEDNet: Multi-scale deep saliency learning for moving object detection,” in Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC), Oct. 2018, pp. 1670–1675.
[45] L. Maddalena and A. Petrosino, “Towards benchmarking scene background initialization,” in Proc. Int. Conf. Image Anal. Process., 2015, pp. 469–476.
[46] C. Cuevas, E. M. Yañez, and N. García, “Labeled dataset for integral evaluation of moving object detection algorithms: LASIESTA,” Comput. Vis. Image Understand., vol. 152, pp. 103–117, Nov. 2016.
[47] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, “Universal background subtraction using word consensus models,” IEEE Trans. Image Process., vol. 25, no. 10, pp. 4768–4781, Oct. 2016.
[48] X. Peng, B. Usman, N. Kaushik, J. Hoffman, D. Wang, and K. Saenko, “VisDA: The visual domain adaptation challenge,” 2017, arXiv:1710.06924. [Online]. Available: http://arxiv.org/abs/1710.06924.
[49] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, “Zero-shot learning—A comprehensive evaluation of the good, the bad and the ugly,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 9, pp. 2251–2265, Jul. 2019.

JAE-YEUL KIM received the B.S. degree in mechanical and automotive engineering from the Seoul National University of Science and Technology, Seoul, South Korea, in 2018, where he is currently pursuing the M.S. degree with the Graduate School of Automotive Engineering. His current research interests include visual surveillance using deep learning and scene understanding for autonomous navigation.

JONG-EUN HA received the B.S. and M.E. degrees in mechanical engineering from Seoul National University, Seoul, South Korea, in 1992 and 1994, respectively, and the Ph.D. degree in mechanical engineering from KAIST, Daejeon, South Korea, in 2000. From February 2000 to August 2002, he worked with Samsung Corning, where he developed an algorithm for a machine vision system. From 2002 to 2005, he worked with Multimedia Engineering, Tongmyong University. Since 2005, he has been a Professor with the Department of Mechanical and Automotive Engineering, Seoul National University of Science and Technology. His current research interests include deep learning and intelligent robots and vehicles.

---