JAISCR, 2023, Vol. 13, No. 3, pp. 151 – 163
10.2478/jaiscr-2023-0012

MOVING OBJECT DETECTION FOR COMPLEX SCENES BY MERGING BG MODELING AND DEEP LEARNING METHOD

Chih-Yang Lin[1], Han-Yi Huang[2], Wei-Yang Lin[2,3], Hui-Fuang Ng[4],
Kahlil Muchtar[5,6,*], Nadhila Nurdin[5]

[1] *Department of Mechanical Engineering, National Central University,*
Taoyuan City 320317, Taiwan

[2] *Department of Computer Science and Information Engineering, National Chung Cheng University,*
Chiayi 62102, Taiwan

[3] *Advanced Institute of Manufacturing with High-Tech Innovations, National Chung Cheng University,*
Chiayi 62102, Taiwan

[4] *Department of Computer Science, University Tunku Abdul Rahman,*
Kampar 31900, Malaysia

[5] *Department of Electrical and Computer Engineering, Universitas Syiah Kuala*
Banda Aceh, Indonesia

[6] *Telematics Research Center, Universitas Syiah Kuala*
Banda Aceh, Indonesia

[*] *E-mail: kahlil@usk.ac.id*

Submitted: 1st December 2022; Accepted: 7th May 2023

Abstract

In recent years, many studies have attempted to use deep learning for moving object detection. Some research also combines object detection methods with traditional background modeling. However, this approach may run into some problems with parameter settings and weight imbalances. In order to solve the aforementioned problems, this paper proposes a new way to combine ViBe and Faster-RCNN for moving object detection. To be more specific, our approach is to confine the candidate boxes to only retain the area containing moving objects through traditional background modeling. Furthermore, in order to make the detection able to more accurately filter out the static object, the probability of each region proposal then being retained. In this paper, we compare four famous methods, namely GMM and ViBe for the traditional methods, and DeepBS and SFEN for the deep learning-based methods. The result of the experiment shows that the proposed method has the best overall performance score among all methods. The proposed method is also robust to the dynamic background and environmental changes and is able to separate stationary objects from moving objects. Especially the overall F-measure with the CDNET 2014 dataset (like in the dynamic background and intermittent object motion cases) was 0,8572.

Keywords: Video surveillance, deep learning, Moving object detection

1 Introduction

Object detection and classification are two essential tasks in many applications [1, 2, 3, 4]. Especially in intelligent transportation system and surveillance challenges, moving object detection can be regarded as fundamental and important research field. A robust and real-time moving object detection algorithm is essential for video surveillance [5], anomaly event detection [6], and object tracking [7]. However, developing reliable moving object detection is challenging due to many factors, one of which is a cluttered background. The most straightforward approach to deal with the background is subtracting the input image from the model, and pixels with different values exceeding a threshold are classified as foreground. Background subtraction approach is suitable for static or slow varying backgrounds, which in reality, scene backgrounds are complex and non-static due to illumination changes, shadow, or dynamic background objects such as swaying leaves and water waves. Gaussian Mixture Model (GMM) [8] and Visual Background Extractor (ViBe) [9] are well-known methods for creating and maintaining adaptive background models. In GMM, multiple Gaussian distributions are used at each pixel location to model the change in background colors. If a new input pixel fits any of the Gaussian distributions, it is considered a background pixel. Otherwise, it is classified as a foreground pixel. Meanwhile, ViBe aggregates a set of previously observed pixel values to build the background model and then uses the distances between an input pixel and the samples in the model to determine moving objects. The samples in the model are randomly replaced with new background pixels so that the model can adapt to background changes. Nevertheless, when dealing with complex background environments, traditional background modeling approaches often misclassify dynamic background objects such as moving trees and produce fragmented segmentation results, as shown in Figure 1a and 1b.

Recently, there are various studies that have proposed deep-learning-based approaches for moving object detection using Convolutional Neural Networks (CNN). Some researchers treated background modeling as an image restoration task [10, 11, 12]. Besides, various methods have been proposed in order to incorporate traditional background modeling methods with deep-learning-based approaches [13, 14, 15, 16, 17]. In addition, some methods, for example, [18], have been utilized to solve the image segmentation problem. Current approaches, whether traditional background modeling or deep learning-based, produce a binary segmentation mask with white pixels representing moving foreground objects and black pixels indicating the background. These approaches do not provide information regarding the number of moving objects detected in the image and the category of each of the moving objects. Such information is essential for subsequence processing steps in video surveillance systems.

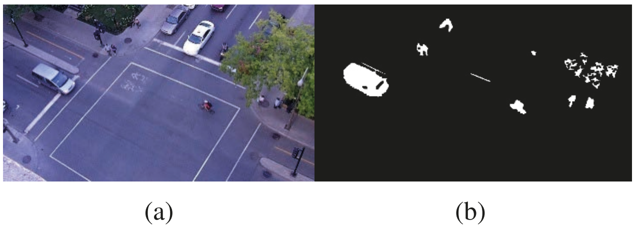

(a) (b)

Figure 1. Challenges encountered in traditional background modeling methods. (a) Original image (b) Result generated by traditional background modeling. In (b), the algorithm mistakenly classified the swaying leaves on the right side of the image as moving targets and produced fragmented segmentation results for the car and pedestrians on the left.

Well-known object detection methods, such as the two-stage region-based R-CNN series detectors [3], and the single-stage Yolo [1] and SSD [4] detectors, are able to detect and recognize objects in an image with high accuracy. However, these detectors cannot distinguish whether the detected objects are stationary or moving. Therefore, existing object detection models cannot be used directly for moving object detection.

In order to address the issues raised, this study proposes a new method for moving object detection and recognition for video surveillance by combining the object detection model with a traditional background modeling method. The proposed method first uses ViBe to identify potential regions of moving objects, and then applies Faster-R-CNN [3] to detect and classify moving objects in the proposed regions. Restricting object detection in regions of moving objects ensures that the detected objects are not stationary. Meanwhile, incorporating an object detection model in segmenting moving objects will improve the overall segmentation accuracy. In object detection, adjacent pixels that conform to the same object are processed as a whole instead of treating each pixel independently. As a result, noise caused by dynamic background can be eliminated and fragmented segmentation of large objects can also be avoided.

In conclusion, the contributions of this paper are as follows:

- We proposed a new method that combines the strengths of a deep learning-based object detection model and traditional background modeling for moving object detection.
- Our method is able to output the number of moving objects detected in the image and the category of each of the moving objects.
- The proposed method is robust to the dynamic background and environmental changes and is able to separate stationary objects from moving objects.

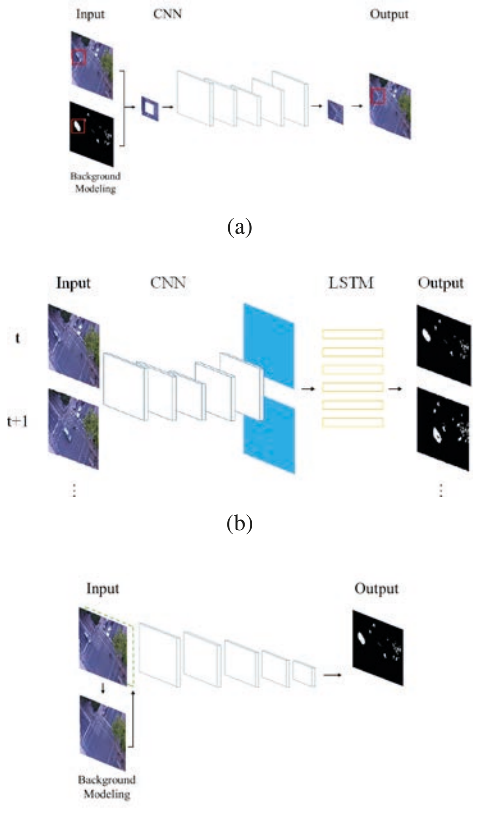
The rest of the paper is organized as follows: In Section 2, related works are reviewed and discussed. In Section 3, the proposed method is described in details. Section 4 presents the experiments and a discussion of the results. Finally, the main conclusions of the paper are summarized in Section 5.

2 Related Work

In this section, we briefly review existing traditional and deep learning-based moving object detection approaches related to this study and their respective challenges.

2.1 Moving objects with Background Restoration via traditional approach

One approach toward moving object detection is to treat background modeling as an image restoration task [10, 11, 12]. First, optical flow or traditional background modeling methods are used to mark areas in an image that may contain moving objects. These areas are then masked out from the image, and a deep neural network such as an autoencoder or GAN is trained to restore the masked background areas for the image. In other words, the network learns to repair and generate the background image after the moving foreground objects are removed. As shown in Figure 2a, the car is detected as a moving object by optical flow, and the area containing the car in the original image is masked out. Next, the masked image is input into the network to predict the background at the masked areas. Finally, the reconstructed background image will be used to compare with the original image to achieve moving object detection. The main challenge faced by this approach is that when the area covering the moving object is too large, this will cause the features available in the image too few to make a good prediction for the large masked area, thus detrimental to the detection performance. Another situation is that when the background is too complex to predict, the reconstructed background image might be incorrect and therefore causing false detection.



(a)

(b)

(c)

Figure 2. Architectural overview of various deep learning-based moving object detection approaches. (a) Background restoration, (b) Foreground background segmentation, (c) Mixing traditional background modeling with deep learning.

2.2 Moving object detection with Foreground Background Segmentation via deep learning method

Another line of work considers moving object detection as a binary image classification task where each pixel value is classified as foreground or background [16, 17, 18]. The original image is first input into an auto-encoder to extract the spatial features of the image. This stage is called the semantic feature extraction network (SFEN) in [16]. However, using only spatial features does not describe the temporality of the image. That is, a feature extraction network does not distinguish whether a segmented foreground object is static or in motion. To resolve this issue, recursive neural networks [16, 17], 3D-CNN [18], or kernel-induced possibilistic fuzzy associated BGS [19] are employed to model the temporal context of an image sequence. The induced kernel function in [19] is used to project low-dimensional data into higher-dimensional space and construct a robust background model based on the density of data in the temporal domain, avoiding noisy and outlier points. In [16], a convLSTM is used to extract the temporal features after the feature extraction network, and it is named the pixel-level sequence learning network (PSL). In PSL, current spatial features extracted via SFEN are input into convLSTM together with previously extracted features to predict a result, as shown in Figure 2b. Some post-processing such as conditional random fields (CRF) is performed to refine the boundary of the segmentation result and produce the final output. It is demonstrated that including temporality context greatly improves the segmentation accuracy of the model. The major drawback of explicitly modeling the temporal context is the long processing time, where the FPS is usually dropped below 10, and thus cannot be applied in a real-time setting. The Arithmetic Distribution Neural Network [20] was used to introduce another approach for learning the distribution of temporal pixels. They [20] used a Bayesian refinement model based on neighboring information and a graphics processing unit (GPU) to improve the model's robustness and accuracy.

2.3 Mixing Traditional with Deep Learning Methods

Combining traditional background modeling with deep learning methods is another common approach for moving object detection [13, 14, 15]. In this approach, a background model is first established using the traditional background modeling method. Next, a CNN is trained to perform background subtraction on the input image. Next, the original input image and the background model generated by the traditional method are concatenated and input to the neural network, and the neural network learns the differences between the input image and the background model and outputs the binary segmentation map, as shown in Figure 2c. For instance, deepBS [13] uses a temporal median operator to obtain the background model from n video frames. Next, a scene-specific CNN is trained on image patches extracted around a pixel from the input image, and from the corresponding background model and ground truth. These patches are concatenated and input to a shallow network, and the network will output a prediction indicating if the pixel is foreground or background. The main challenge with this approach is that the performance of the network prediction is affected by the effectiveness of the traditional background modeling. Another problem is that patch-based segmentation requires significant processing time.

In this study, we propose a new way to combine traditional background modeling with a deep learning-based object detection method to detect and classify moving objects in video surveillance. Through traditional background modeling, the proposed method first identifies potential regions of moving foreground objects and then applies a deep learning-based object detector to detect and classify moving objects in the regions. Our method is not only robust to the dynamic background and environmental changes; it is also capable of identifying the types of moving objects which is important for video surveillance systems.

3 Method

An overview of the architecture of the proposed model is shown in Figure 3. The input image is fed concurrently into two separate channels in the model. In the first channel, the image is preprocessed using ViBe [9] to detect moving foreground regions in the image. After that, a region proposal placement scheme is used to determine potential locations in the detected regions that are most likely to contain moving objects. In the meantime, in the second channel, the original input image is fed into a CNN to extract the feature map and then to the region proposal network (RPN) of Faster R-CNN [3] which generates region proposals for locations in the image that might contain objects. Next, the output of ViBe is used to filter out false region proposals proposed by the RPN that are not in the moving regions. Finally, the remaining regions' proposals are input into a classification network to classify the types of detected moving objects. The following sections give details of each of the processing steps.

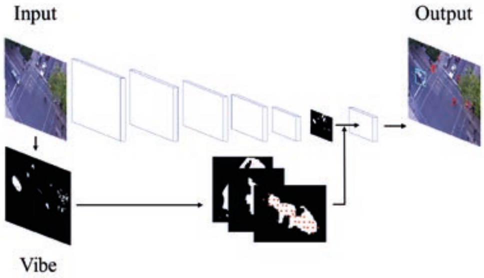


Figure 3. Overall architecture of the proposed model

3.1 Moving Foreground Detection with ViBe

ViBe [9] is a robust and commonly used pixel-wise background modeling algorithm for moving foreground detection. ViBe's result is a binary image with 1 indicating a moving object region and 0 indicating a background region. ViBe is highly efficient and produces satisfactory results. However, when dealing with complex and dynamic scenes, ViBe often produces fragmented segmentation results and misclassifies dynamic background objects, as shown in Figure 1. As a result, after foreground detection with ViBe, we applied morphological operations to remove the noise in the segmentation image. As shown in Figure 4, after three iterations of dilation, most holes are filled and broken fragments are re-connected, such as the car object on the left of the image. Then, followed by three iterations of erosion, noises smaller than the structural elements are filtered out and the extra boundaries caused by dilation are reduced, as shown in the bottom row of Figure 4. Finally, connected components with a number of pixels less than a threshold value of 200 are removed. There are still some noisy dynamic background objects that are not cleaned after the morphological processing step and they will be handled subsequently by the object detection network.

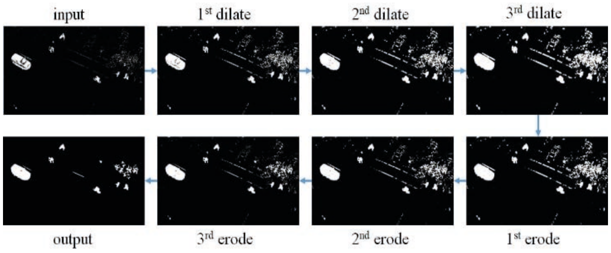


Figure 4. Overall architecture of the proposed model

3.2 Region Proposal Placement Scheme

The foreground segmentation result generated by ViBe will be used as a mask for guiding the object detector where to detect moving objects. The connected-component algorithm is first applied to the output of ViBe to demarcate moving objects. Note that the output of ViBe usually contains not only potential moving objects but also false detection such as the dynamic background. Therefore, performing object detection and classification on the output of ViBe can help determine which detected foreground regions are actually objects, and which are noise or dynamic backgrounds.

As mentioned earlier, Faster R-CNN, a popular two-stage object detector, will be used here to detect objects and classify objects in an image (will be discussed in a subsequent subsection). In the first stage, Faster R-CNN uses a region proposal network (RPN) to generate region proposals for locations in the image that might contain objects. In the second stage, the region proposals are passed to a classification network to determine whether the region actually contains an object, the object type, as well as parameters to refine the region shape to best fit the object. RPN usually produces a large number of region proposals and cannot distinguish whether the objects are stationary or moving. This is exactly the reason why we proposed to use the result of ViBe to guide the object detector to focus only on moving objects.

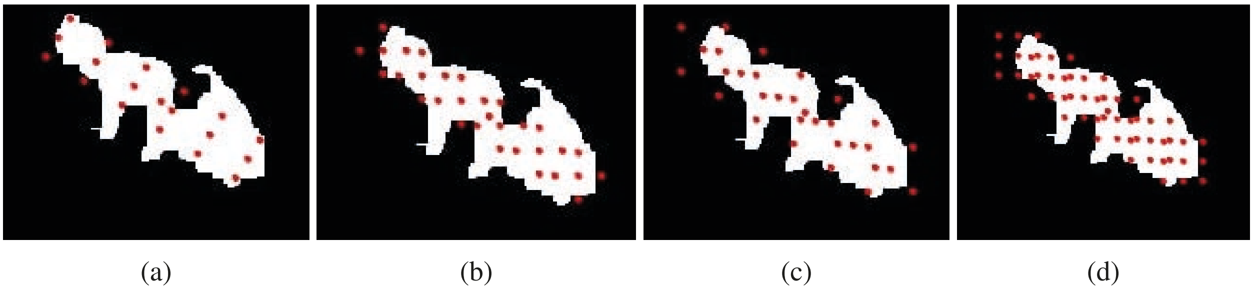


Figure 5. Different neighbor settings for region proposal placement scheme. (a) Two neighbors, (b) Four neighbors (0 degrees), (c) Four neighbors (45 degrees), (d) Eight neighbors.

However, not all pixel locations within a detected foreground region are good candidates for a potential object center position. For example, pixel locations near the centroid of the foreground region are more likely to be the object center, while pixel locations near the edges of the region are less likely. Based on this observation, we proposed a novel region proposal placement scheme to help select region proposals proposed by the RPN and to filter out unrelated proposals. Firstly, for each connected component, the minimum volume enclosing the ellipsoid (MVEE) is calculated and the major axis and minor axis of the ellipse are obtained. The idea is to select region proposals along or near the major axis. Secondly, large objects require longer intervals between proposals to avoid dense placement. On the contrary, if a connected component contains multiple small overlapping moving objects, it should have a shorter interval between region proposals. The interval can be determined by the semi-minor axis length of MVEE.

There are several possible ways to select the placement scheme on a connected foreground region. Figure 5 shows four different placement settings based on two neighbors, four neighbors, and eight neighbors. Our experimental results show that the setting with four neighbors with 0 degrees achieves the best performance. Therefore, unless otherwise stated, four neighbors with 0 degrees setting is used throughout the paper.

3.3 Region Proposal Selection

For an input image, the RPN of Faster R-CNN [3] generates a feature map which predicts objects in the image, known as region proposals. A high value at a pixel location in the feature map indicates a high probability of containing an object. The first step to filter out stationary object predictions is by performing an AND operation between RPN feature map and ViBe's result.

The result of an AND operation contains several positive outcomes. First, stationary objects in the image will be masked out by ViBe's result while moving objects will be retained. Secondly, if an entity in the image is not the target object, the probability of containing an object is low, and therefore the value in the feature map will be small. Even if ViBe's result falsely detected it as a foreground, for instance, a dynamic background due to a water wave or waving tree, the low feature value from RPN will prevent it from being classified as a moving object. Thirdly, incomplete or fragmented segmentation of moving objects in ViBe's result due to noise can be recovered since RPN should produce a high probability value at the corresponding position.

After the AND operation between the RPN feature map and ViBe's results, we will select the top K candidate region proposals based on the input to the second stage for object classification and bounding box regression. One straightforward approach is to choose region proposals based on the probability values of the feature map. However, there are two issues with such an approach. First, as discussed in the previous section, region proposals near the major axis of a foreground region are more likely to enclose the object. Thus, the relative position of a region proposal is also an important factor for selection consideration. Secondly, it has been shown that one can fool a deep learning-based object detector by carefully altering a small part of an image. This can become a serious security issue in video surveillance. However, such an alteration will not affect the result of ViBe. As such, the proposed region proposal placement scheme discussed in the previous section will be used to assist region proposal selection and thus avoid such a security issue.

In summary, the top K candidate region proposals are selected based on the probability values of the feature map and the position of a region related to the region proposal placement scheme. If the number of proposals is less than K, then only the non-zero candidate proposals are selected and sent to the second stage. K is set at 300 in this paper. Figure 6 shows an example of the proposed region proposal selection scheme. As seen in Figure 6, stationary objects (two cars in the upper middle) are filtered out by ViBe. The dynamic background noise (waving tree in upper right) in the Vibe result images has a very low probability value in the feature map (6b) and is eliminated. So it will also be filtered out. The example shows that the proposed region proposal selection method can successfully achieve the purpose of moving object detection.

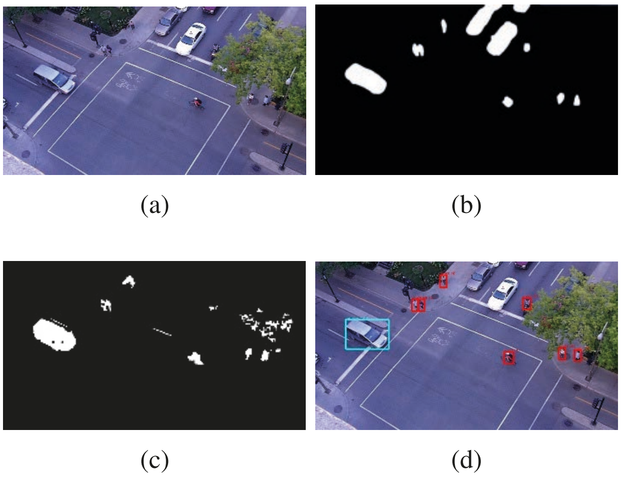


Figure 6. Example of region proposal selection. (a) Original image (b) RPN feature map (c) Vibe result (d) Moving object detection result after region proposal selection.

3.4 Object Detection Details

In this paper, Faster R-CNN [3] object detector is used for moving object detection and classification. ResNet-50 [17] with FPN [18] were used as the backbone network to extract spatial features from the input image. Figure 7 shows the architecture of our backbone network. ResNet can be divided into five parts, including conv1_x, conv2_x, conv3_x, conv4_x, and conv5_x. Multiple bottlenecks are used in ResNet with each bottleneck containing three convolution layers. The first layer is a 1x1 convolution to reduce the dimension of feature maps' depth channel. Next is a 3x3 convolution layer. And finally, another 1x1 convolution layer to expand the depth dimension. The goal of bottlenecks is to reduce the total number of parameters in the network. The conv2_x to conv5_x layers contain 3, 4, 6, and 3 bottlenecks, respectively. The five parts correspond to C1 to C5 in Figure 7.

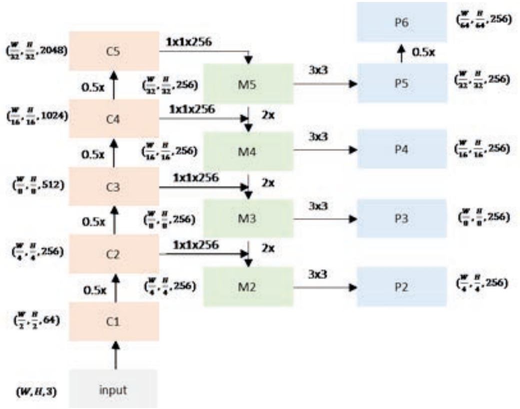

P6
$(\frac{W}{64},\frac{H}{64},256)$
0.5x
$(\frac{W}{32},\frac{H}{32},2048)$
C5
1x1x256
0.5x
$(\frac{W}{32},\frac{H}{32},256)$
M5
3x3
P5
$(\frac{W}{32},\frac{H}{32},256)$
$(\frac{W}{16},\frac{H}{16},1024)$
C4
1x1x256
2x
0.5x
$(\frac{W}{16},\frac{H}{16},256)$
M4
3x3
P4
$(\frac{W}{16},\frac{H}{16},256)$
$(\frac{W}{8},\frac{H}{8},512)$
C3
1x1x256
2x
0.5x
$(\frac{W}{8},\frac{H}{8},256)$
M3
3x3
P3
$(\frac{W}{8},\frac{H}{8},256)$
$(\frac{W}{4},\frac{H}{4},256)$
C2
1x1x256
2x
0.5x
$(\frac{W}{4},\frac{H}{4},256)$
M2
3x3
P2
$(\frac{W}{4},\frac{H}{4},256)$
$(\frac{W}{2},\frac{H}{2},64)$
C1
$(W,H,3)$
input

Figure 7. Architecture of the object detector backbone network.

In addition to the general ResNet network, we also use the Feature Pyramid Network (FPN). The purpose of the feature pyramid is to improve the detection accuracy of different object scales, especially for small objects. Some approaches directly resize the original image into several different scales and input the rescaled images into different convolutional neural networks to extract features. Finally, predictions are made on the different feature maps. However, input rescaled images to multiple convolutional neural networks increased the overall processing time. In CNN, the deep networks can extract richer semantic information and it is more suitable for detecting large objects. On the other hand, shallow networks have more location information and are suitable for detecting small objects. In FPN, instead of resizing the image before the network, information on the deep and shallow feature maps is merged together to improve detection accuracy. More detailed information on the backbone network is shown in Table 1. In the beginning, the original input image is passed through a five-stage convolution that generates feature maps C1, C2, C3, C4, and C5, respectively. The feature maps extracted by the backbone network are further convolved with a 1x1x256 convolution which reshapes all the feature maps to 256 channels. Next, the feature maps are up-sampled and added to the previous feature map and produce intermediate feature maps M2, M3, M4, and M5, which fuse deep information together with shallow information. Finally, the intermediate feature maps are convolved with a 3x3 convolution to get the final feature maps. As shown in Figure 7 and Table 1, after the backbone network, we obtained five feature maps P2 to P6 in which shallow and deep information are combined.

3.5 Model Training and Testing Output

During the training phase, we use the training images to train an original two-stage object detection network to detect the target objects without using the ViBe results. The results output at this phase does not recognize an object's motion state. During the testing phase, the proposed moving object detection framework will incorporate ViBe results in the RPN to filter out stationary objects and output the detected moving objects.

Note that compared to existing moving object detection methods, which output only a binary segmentation mask of moving foreground objects, our method additionally outputs an object's category and its associated bounding box. It is straightforward to simultaneously output a pixel-level segmentation mask for each detected object. To this end, we adopted the Mask R-CNN [21] approach, which extends Faster R-CNN by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition. In the experiments described below, Mask R-CNN will be used in our model to generate the binary segmentation mask.

3.6 Dataset and Evaluation Metrics

In literature, CDNET 2014 is one of the most comprehensive change detection dataset that is provided publicly. This dataset provides the evaluation of state-of-the-art approaches, not only unsupervised, but also supervised methods (which mostly involving deep learning approach). The CDNET [22] background subtraction benchmark dataset was used in this study to evaluate the proposed method. The CDNET dataset contains videos in 11 categories that represent different challenges that may be encountered in moving object detection. Since the main objective of this study is to improve original object detection methods so that an object's motion state can be recognized, they can be used for effective moving object detection in video surveillance. As such, we have selected the Dynamic Background and Intermittent Object Motion categories for evaluation purposes. Each category contains 6 video sequences.

The performance metrics used in this study to evaluate the proposed method are Precision, Recall, and F-measure, define as:

$$Precision = \frac{TP}{TP+FP} \tag{1}$$

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

$$F-Measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{3}$$

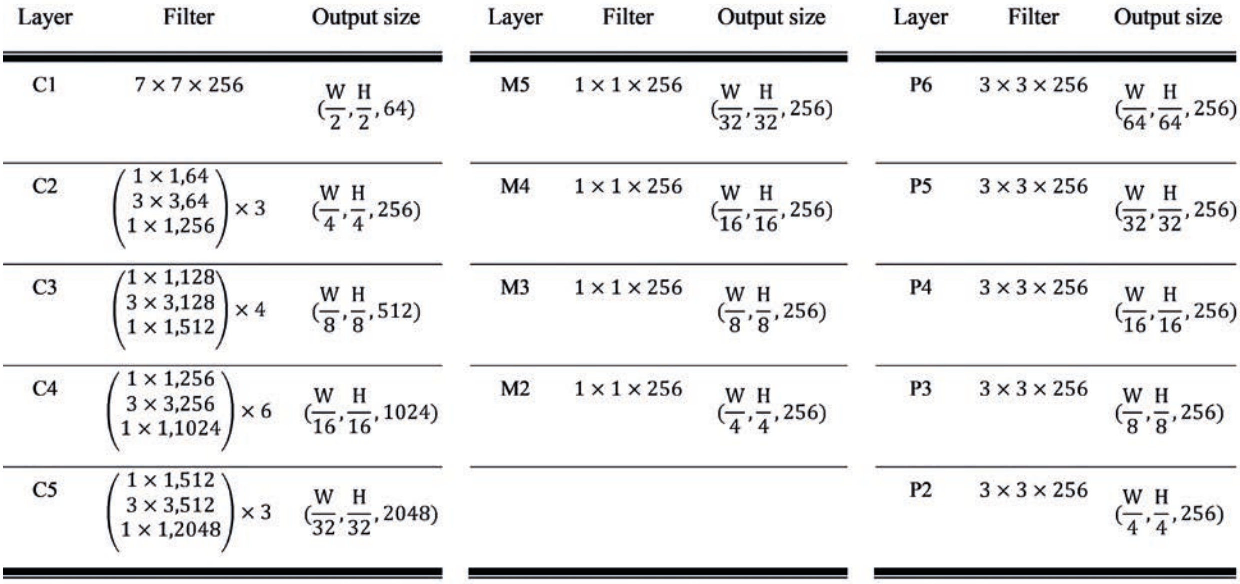
where TP refers to a true positive, the number of foreground pixels that are correctly classified as foreground and FP is a false positive, which is the number of background pixels that are incorrectly classified as foreground. FN denotes a false negative, the number of foreground pixels that are misclassified as background.

3.7 Implementation Details

After performing data preprocessing, such as removing video sequences that do not have corresponding ground truth, the dataset is divided into a 50% training set and a 50% test set. The training set is used to train the object detector. Since most images in a video sequence are similar and highly correlated, to avoid data redundancy, we sample the images in a video sequence with a fixed time interval. In the end, about 50 images per video were selected for training. The weights of the object detector were pre-trained using the COCO dataset [23]. In many experiments, the pre-trained network can be used directly without the need to retrain it using training data. In cases where the target object category is not covered in the COCO dataset, we retrain the object detection network with the training set.

Two traditional methods, GMM [8] and ViBe [9], were selected for comparison with the proposed

Table 1. Detailed information of the object detector backbone network.



Layer	Filter	Output size	Layer	Filter	Output size	Layer	Filter	Output size
C1	$7\times7\times256$	$(\frac{W}{2},\frac{H}{2},64)$	M5	$1\times1\times256$	$(\frac{W}{32},\frac{H}{32},256)$	P6	$3\times3\times256$	$(\frac{W}{64},\frac{H}{64},256)$
C2	$\begin{pmatrix}1\times1,64\\3\times3,64\\1\times1,256\end{pmatrix}\times3$	$(\frac{W}{4},\frac{H}{4},256)$	M4	$1\times1\times256$	$(\frac{W}{16},\frac{H}{16},256)$	P5	$3\times3\times256$	$(\frac{W}{32},\frac{H}{32},256)$
C3	$\begin{pmatrix}1\times1,128\\3\times3,128\\1\times1,512\end{pmatrix}\times4$	$(\frac{W}{8},\frac{H}{8},512)$	M3	$1\times1\times256$	$(\frac{W}{8},\frac{H}{8},256)$	P4	$3\times3\times256$	$(\frac{W}{16},\frac{H}{16},256)$
C4	$\begin{pmatrix}1\times1,256\\3\times3,256\\1\times1,1024\end{pmatrix}\times6$	$(\frac{W}{16},\frac{H}{16},1024)$	M2	$1\times1\times256$	$(\frac{W}{4},\frac{H}{4},256)$	P3	$3\times3\times256$	$(\frac{W}{8},\frac{H}{8},256)$
C5	$\begin{pmatrix}1\times1,512\\3\times3,512\\1\times1,2048\end{pmatrix}\times3$	$(\frac{W}{32},\frac{H}{32},2048)$				P2	$3\times3\times256$	$(\frac{W}{4},\frac{H}{4},256)$

Table 2. F-measure comparisons of different methods on the CDNET dataset. Bold entries indicate the best result in the given column.

Method	Dynamic Background	Intermittent ObjectMotion	Overall
GMM	0.2024	0.3332	0.2678
ViBe	0.3796	0.5409	0.4602
SFEN(Vgg)	0.6030	0.5775	0.5902
SFEN(Vgg)+CRF	0.6207	0.6058	0.6132
SFEN(Vgg)+PSL+CRF	0.7538	0.6175	0.6856
SFEN(ResNet)+PSL+CRF	0.8220	0.8453	0.8336
DeepBS	0.8761	0.6098	0.7429
Proposed method	0.8521	0.8624	0.8572

method. GMM is written in MATLAB, and ViBE is written in the C programming language. Two deep learning-based methods, DeepBS [11] and SFEN [13], are also included here for comparison. The results of the two methods are extracted from their respective papers.

4 Result and Discussion

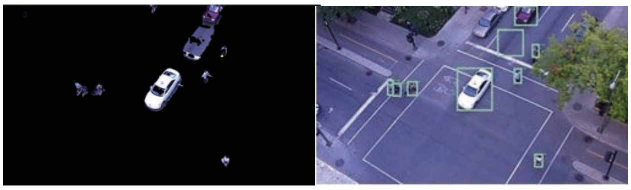
The *F*-measure of five different methods on the CDNET dataset is shown in Table 2. It can be seen that our method has the best overall performance among the five methods. There is a significant improvement between the proposed method and the traditional methods. Furthermore, when compared to deep learning-based approaches, our methods' *F*-measures are slightly lower than DeepBS in the dynamic background category but substantially higher in the intermittent object motion category. The overall score of our method is 0.1143 points higher than DeepBS's. Similarly, the proposed method outperformed SFEN in terms of the overall score. LSTM slows down the speed performance of SFEN, which, despite being able to produce reasonably good segmentation results, drops the processing speed to less than 10 fps [13]. SFEN can reach about 33 fps without LSTM, but its *F*-measure will drop to 0.6132. Figure 8 shows sample segmentation results of our method compared to the two traditional methods. The proposed method is able to maintain the integrity of the foreground objects and is not sensitive to the dynamic background, as can be seen with the help of the object detector.

Our model contains two components, the traditional background modeling algorithm ViBe and the deep learning-based object detector. ViBe is executed on the CPU. The deep learning-based object detector is performed on the GPU NVIDIA GTX 1080 ti. The size of the image has a significant impact on processing speed. When the image size is set to 320x240 for CDNET, real-time performance can be achieved at about 34 fps. Processing speed drops to 15 frames per second when 720x480 images are used.

4.1 Timing of Mixing ViBe Result

In the proposed model, ViBe results are incorporated with the results of the region proposal network (RPN) before being sent to the classification network. In this experiment, we investigate additional options for adding the ViBe result to the network. In particular, we perform an AND operation between the ViBe result and the original image before inputting it into the network for object detection. If a pixel is classified as background in ViBe, the same pixel in the original image will be changed to black. An example is shown in Figure 9a, where all static pixels are set to black while foreground pixels are unchanged. The object detection network is then trained using the final image, which does not contain any static background information.

For such a setup, we discover that the network will learn the information around object edges. Such information can mislead the network into misclassifying a static object as a moving object by only observing the boundary of the object. An illustration is shown in Figure 9a, where a stopped car begins to move after the traffic light turns green. Since the ViBe background model has not yet been updated completely, a small portion of the car (e.g., the front end) is classified as moving foreground by the ViBe at the location where the car was previously stopped. Because the object detection network has been trained to detect objects based on the boundary of an object, it will report that a car object has been detected even though there is no car at that location, as shown in Figure 9b. Therefore, the experimental results suggest that ViBe's results should be incorporated into the RPN feature map instead of the original image.



(a) (b)

Figure 9. Example result of performing AND operation between ViBe result and the original image before inputting to the neural network for object detection.

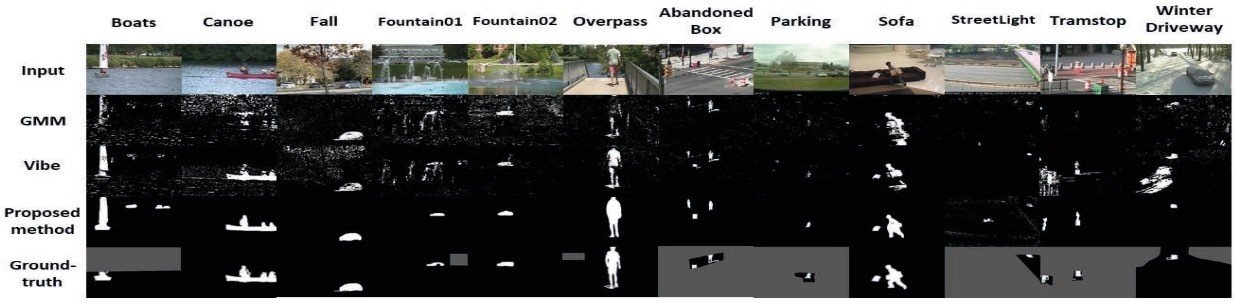


Figure 8. Segmentation results on the CDNET dataset. Starting from the first row is the original image, followed by results of GMM, ViBe, the proposed method, and Ground-truth.

4.2 Failure Cases

Our model performs object detection within moving foreground areas based on the ViBe result. False detection (false positive) can occur when ViBE misclassifies a dynamic background as foreground and there are static objects that overlap or are near the misclassified area. When this occurs, the object detector will detect static objects and classify them as moving objects.

The quantitative findings from the six videos in CDNET's dynamic background category are displayed in Table 3. It can be seen that the performance of the proposed method is much lower for the fountain01 video compared with other videos. Figure 10 displays a sample frame from the fountain01 video along with the results of its detection. It is visible that two parked cars overlap with the fountain, and both cars are misclassified as moving objects. To avoid such failure cases, a more carefully designed post-processing operation on the ViBe result can be considered to reduce dynamic background noise.

Table 3. Quantitative results of the videos in the Dynamic Background category of CDNET.

Video	Precision	Recall	F-measure
Boats	0.8818	0.9745	0.9258
Canoe	0.9816	0.9275	0.9538
Fall	0.8836	0.9097	0.8965
Fountain01	0.3707	0.7279	0.4912
Fountain02	0.9743	0.8055	0.8819
Overpass	0.9423	0.9851	0.9632

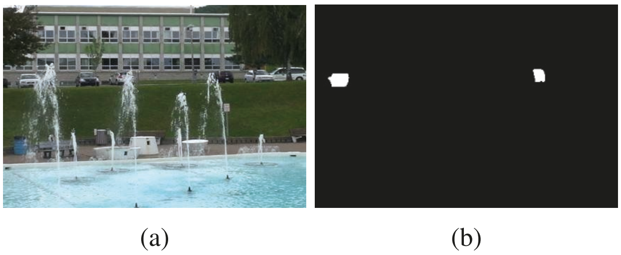

(a) (b)

Figure 10. Failure cases happen when static objects overlap with a dynamic background such as fountains.

5 Conclusion

Moving object detection is an important component in video surveillance and in general computer vision applications. Traditional methods are often challenged by the establishment, updating, and comparison of background models. Deep learning-based approaches tackle some of the challenges, but they are usually complex and are not suitable for real-time video surveillance systems.

In this paper, we use the results of ViBe as guidance to a Faster RCNN to perform object detection in areas that may contain moving objects. Our method combines the strengths of deep learning-based object detection models and traditional background modeling such that our method can output the number of moving objects and their object types, which is essential for video surveillance systems. Experimental results show that the proposed method works well under challenging dynamic backgrounds and changing conditions.

References

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

[2] R. Grycuk, R. Scherer, A. Marchlewska, and C. Napoli, Semantic hashing for fast solar magnetogram retrieval, Journal of Artificial Intelligence and Soft Computing Research,vol. 12, 2022.

[3] S. Ren, K. He, R. Girshick, and J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, Advances in neural information processing systems, vol. 28, 2015.

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

[5] K. Muchtar, A. Bahri, M. Fitria, T. W. Cenggoro, B. Pardamean, A. Mahendra, M. R. Munggaran, and C.-Y. Lin, Moving pedestrian localization and detection with guided filtering, IEEE Access, vol. 10, pp. 89 181–89 196, 2022.

[6] M.-I. Georgescu, A. Barbalau, R. T. Ionescu, F. S. Khan, M. Popescu, and M. Shah, Anomaly detection in video via self-supervised and multi-task learning, in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 12 742–12 752.

[7] F. R. Valverde, J. V. Hurtado, and A. Valada, There is more than meets the eye: Self-supervised multi-object detection and tracking with sound by distilling multimodal knowledge, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 11 612–11 621.

[8] C. Stauffer and W. E. L. Grimson, Adaptive background mixture models for real-time tracking, in Proceedings. 1999 IEEE computer society conference on computer vision and pattern recognition (Cat. No PR00149), vol. 2. IEEE, 1999, pp. 246–252

[9] O. Barnich and M. Van Droogenbroeck, Vibe: a powerful random technique to estimate the background in video sequences, in 2009 IEEE international conference on acoustics, speech and signal processing. IEEE, 2009, pp. 945– 948.

[10] Z. Qu, S. Yu, and M. Fu, Motion background modeling based on context-encoder, in 2016 Third International Conference on Artificial Intelligence and Pattern Recognition (AIPR). IEEE, 2016, pp. 1–5.

[11] M. Sultana, A. Mahmood, S. Javed, and S. K. Jung, Unsupervised deep context prediction for background estimation and foreground segmentation, Machine Vision and Applications, vol. 30, no. 3, pp. 375–395, 2019.

[12] Y. Tao, P. Palasek, Z. Ling, and I. Patras, Background modelling based on generative unet, in 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). IEEE, 2017, pp. 1–6.

[13] M. Babaee, D. T. Dinh, and G. Rigoll, A deep convolutional neural network for video sequence background subtraction, Pattern Recognition, vol. 76, pp. 635–649, 2018.

[14] M. Braham and M. Van Droogenbroeck, Deep background subtraction with scene-specific convolutional neural networks, in 2016 international conference on systems, signals and image processing (IWSSIP). IEEE, 2016, pp. 1–4.

[15] Y. Wang, Z. Luo, and P.-M. Jodoin, Interactive deep learning method for segmenting moving objects, Pattern Recognition Letters, vol. 96, pp. 66–75, 2017.

[16] Y. Chen, J. Wang, B. Zhu, M. Tang, and H. Lu, Pixelwise deep sequence learning for moving object detection, IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 9, pp. 2567–2579, 2017.

[17] Z. Hu, T. Turki, N. Phan, and J. T. Wang, A 3d atrous convolutional long short-term memory network for background subtraction, IEEE Access, vol. 6, pp. 43 450–43 459, 2018.

[18] D. Sakkos, H. Liu, J. Han, and L. Shao, End-to-end video background subtraction with 3d convolutional neural networks, Multimedia Tools and Applications, vol. 77, no. 17, pp. 23 023–23 041, 2018.

[19] B. N. Subudhi, M. K. Panda, T. Veerakumar, V. Jakhetiya, and S. Esakkirajan, Kernel-induced possibilistic fuzzy associate background subtraction for video scene, IEEE Transactions on Computational Social Systems, 2022.

[20] C. Zhao, K. Hu, and A. Basu, Universal background subtraction based on arithmetic distribution neural network, IEEE Transactions on Image Processing, vol. 31, pp. 2934–2949,2022.

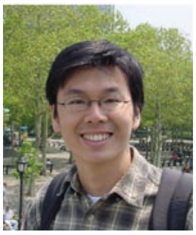
[21] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[22] N. Goyette, P.-M. Jodoin, F. Porikli, J. Konrad, and P. Ishwar, Changedetection. net: A new change detection benchmark dataset, in 2012 IEEE computer society conference on computer vision and pattern recognition workshops. IEEE, 2012, pp. 1–8.

[23] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, Microsoft coco: Common objects in context, in European conference on computer vision. Springer, 2014, pp. 740–755.

Chih-Yang Lin (Senior Member, IEEE) is a professor with the Department of Mechanical Engineering, National Central University, Taoyuan, Taiwan. He is an IET Fellow and has authored or co-authored over 200 papers in international conferences and journals, and received Best Paper Awards from Pacific-Rim Conference on Multimedia (PCM) in 2008, Best Paper Awards, and Excellent Paper Award from IPPR Conference on Computer Vision, Graphics and Image Processing Conference in 2009, 2013, and 2019, Best Paper Award from 6th International Visual Informatics Conference 2019 (IVIC'19), and Best Paper Award from 2nd International Conference on Broadband Communications, Wireless Sensors, and Powering in 2020. He has served as a program chair, session chair, publication chair, publicity chair, or workshop organizer at many international conferences, including AHFE, ICCE, ACCV, IEEE Multimedia Big Data, ACM IH and MMSec, APSIPA, and CVGIP. He is also a regular Reviewer of the IEEE Transactions on Image Processing, the IEEE Transactions on Circuits and Systems for Video Technology, the IEEE Transactions on Multimedia, IEEE Access, and many other prestigious Elsevier journals. His research fields include computer vision, machine learning, deep learning, image processing, big data analysis, and the design of surveillance systems.
https://orcid.org/0000-0002-0401-8473

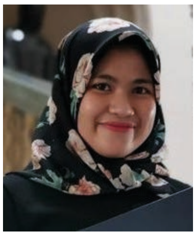
Han-Yi Huang received a master's degree from CCU Tai- wan. The ongoing research is related to background modeling, deep learning, integrated surveillance system, and so on. During the study, the advisors are Prof. Wei-Yang Lin and Prof. Chih-Yang Lin (Senior Member, IEEE), respectively.



Wei-Yang Lin received BSEE from National Sun Yat-sen University, Taiwan, in 1994. He received MSEE and PhD degrees from University of Wisconsin- Madison in 2004, and 2006, respectively. Since 2006, he has been with the Department of Computer Science and Information Engineering, National Chung Cheng University, Taiwan, where he is currently an assistant professor. His research interests include computer vision, biometric authentication, and multimedia signal processing.
https://orcid.org/0000-0003-0895-2498

Hui-Fuang Ng received a Ph.D. degree in biosystems and agricultural engineering from the University of Minnesota, USA, in 1996. He was a Software Engineer with PPT Vision Inc., Minnesota, USA, from 1996 to 2003. He joined the Department of Computer and Information Science, at Asia University, Wufong, Taiwan, from 2003 to 2013. He is currently with the Department of Computer Science, University Tunku Abdul Rah- man (UTAR), Malaysia. He is also the Chairperson of the Center for IoT and Big Data, UTAR. His re- search interests include image processing, computer vision, and machine learning.
https://orcid.org/0000-0003-4394-2770

Kahlil Muchtar (Senior Member, IEEE) received the B.S. degree in informatics from the School for Engineering of PLN's Foundation (STT-PLN), Jakarta, Indonesia, in 2007, the M.S. degree in computer science and in- formation engineering from Asia University, Taichung, Taiwan, in 2012, and the Ph.D. degree in electrical engineering from the National Sun Yat-sen University (NSYSU), Kaohsiung City, Taiwan. From 2018 to 2020, he was involved in a startup company as an AI Re- search Scientist at Nodeflux, Jakarta. From 2019 to 2021, he was appointed as the Chairperson of the Telematics Research Center (TRC), Universitas Syiah Kuala, Banda Aceh, Indonesia. Since October 2021, he has been appointed as the Head of the Computer Engineering Bachelor Program. He is currently an Assistant Professor with the Department of Electrical, and Computer Engineering, Universitas Syiah Kuala. His research interests include computer vision and image processing. He received the 2014 IEEE GCCE Outstanding Poster Award and IICM Taiwan 2017 The Best of Ph.D. Dissertation Award. He served as the Publication Chair for IEEE ICELTICs 2017 and 2018. In 2021, he served as the General Chair for the IEEE IC-COSITE. He is also a member of ACM (Association for Computing Machinery) and APSIPA (Asia-Pacific Signal and Information Processing Association).
https://orcid.org/0000-0001-5740-1938



Nadhila Nurdin received a bachelor's degree in computer science from the Universitas Syiah Kuala (USK), in 2017. She continued her study in computer science majoring in system modeling and data analysis at AGH University of Science and Technology in Poland and completed her master's degree in 2020 with support from Narodowa Agencja Wymiany Akademickiej (NAWA) under Ignacy Łukasiewicz Scholarship program. Since 2022, she has been a Lecturer and a member of the Electrical and Computer Engineering Department, at Universitas Syiah Kuala. Her research interests are Ma- chine Learning, Computer Vision, Natural Language processing, and Explainable Artificial Intelligence.
https://orcid.org/0009-0008-0240-8639