Multiscale Fully Convolutional Network for Foreground Object Detection in Infrared Videos

Dongdong Zeng and Ming Zhu

***Abstract*— Accurate and fast infrared (IR) foreground object detection is one of the most significant issues to be solved due to its important meaning for IR target recognition, IR precise guidance, IR video surveillance, and so on. A common approach for such tasks is "background subtraction," which aims to detect foreground object through background modeling. Thus far, many background subtraction methods have been proposed and have achieved good performance. However, due to the special characteristics of IR images, a few algorithms are suitable for IR foreground object detection. Recently, features learned from convolutional neural networks (CNNs) have demonstrated great success in many vision tasks, such as classification and recognition. In this letter, we propose a novel multiscale fully convolutional network architecture for IR foreground object detection. Given a CNN model pretrained on a large-scale image data set, our method takes output features from different layers of the network. With features from multiple scales, our feature representation contains both category-level semantics and fine-grain details. The experimental results on IR image sequences show that the proposed method achieves the state-of-the-art performance while operating in real time.**

***Index Terms*— Background subtraction, infrared (IR) object detection, IR video surveillance, multiscale fully convolutional network (MFCN).**

I. Introduction

INFRARED (IR) foreground object detection is important due to its wide range of applications, such as IR target recognition, IR precise guidance, and IR video surveillance [1], [2]. Compared with visible images, foreground object detection in IR images can be more complex because of many special characteristics, such as low signal-to-noise ratios, low contrast, lack of structure, such as shape and texture information, high uncertainty, and high ambiguity of pixel values [3]. All these factors make the detection of IR foreground object more difficult and challenging.

Current IR foreground object detection methods are primarily based on state-of-the-art background subtraction algorithms. In the past few decades, a multitude of background subtraction methods for visible images has been proposed and has achieved good performance [4]. Most of these methods primarily manifest in the following two aspects: more advanced background models [5]–[8] and more complex feature representations [9]–[12]. However, due to the special characteristics of IR images and the differences between the features of IR and visible objects, a few background subtraction algorithms are suitable for IR foreground object detection.

Interestingly, deep neural networks have recently drawn much attention in the computer vision community, and deep features obtained from convolutional neural networks (CNNs) have been shown to be effective for many computer vision tasks, such as classification and recognition [13], [14]. Despite their popularity, only a few attempts have been made to employ CNNs for background subtraction. The first novel background subtraction method with the use of a CNN was proposed in [15]. In this method, a fixed background model image is first generated through a temporal median operation over several initialization frames. Then, for each pixel, small patches around the pixel extracted from the background image and the input frame combined with its ground-truth label are used to train the CNN model. After the network model is trained, to detect foreground object in a new frame, patches around a pixel are fed through the network and the foreground probability for that pixel is obtained. An improved CNN-based background subtraction was proposed in [16] with a cascade CNN architecture that achieves the state-of-the-art performance. However, these CNN-based methods have several drawbacks. First, all of them are patch-wise-based methods. Extracting the patches is not only time consuming but also results in very rough foreground masks. Second, to classify the foreground or background, only the output of the last layer features is considered in the network. Third, current methods using highly redundant data to train the network cause overfitting problems.

The fully convolutional network (FCN) architecture was first proposed in [17] for image segmentation. Compared with patch-wise methods, FCN-based models can capture more local and global context information, which produces more accurate and detailed segmentations. Recent research shows that considering features from different convolutional layers can improve results for different vision tasks [14], [18]. The lower layers contain low-level semantic information but retain a higher spatial resolution, while the deep layers capture more high-level semantic information but with less spatial detail.

With these considerations in mind, we propose a novel multiscale FCN (MFCN) architecture that takes advantage of different layer features for IR foreground object detection.

Manuscript received October 27, 2017; revised December 13, 2017 and January 12, 2018; accepted January 22, 2018. Date of publication February 9, 2018; date of current version March 23, 2018. This work was supported by the National Nature Science Foundation of China under Grant 61401425. *(Corresponding author: Dongdong Zeng.)*

D. Zeng is with the Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: zengdongdong13@mails.ucas.edu.cn).

M. Zhu is with the Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China (e-mail: zhu_mingca@163.com).

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2018.2797538

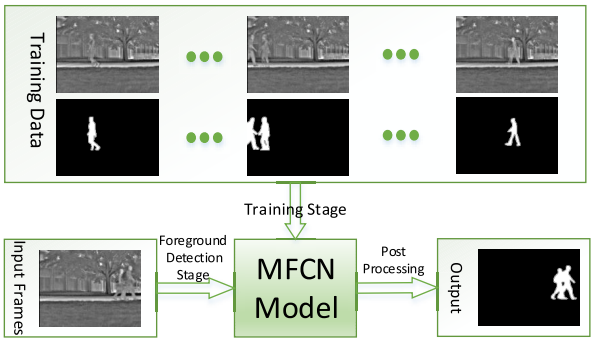

Fig. 1. Process of the proposed MFCN-based IR foreground object detection method.

With the features gained from multiple scales, our feature representation contains both category-level semantics and fine-grain details. The experimental results on IR image sequences show that our method achieves both the state-of-the-art and the real-time performance during the detection process.

The remainder of this letter is organized as follows. Section II describes the framework of the proposed MFCN-based IR foreground object detection algorithm. Section III presents the results of the experiments conducted on various IR image sequences compared with other state-of-the-art methods. Finally, conclusions are given in Section IV.

II. MFCN-Based Background Subtraction

In this section, we give a detailed description of the framework for the MFCN-based IR foreground object detection method. Fig. 1 shows the process of the proposed method.

A. Training Data Preparation

To train the network, for each IR image sequence, we take a random subset of 150 input frames with their corresponding ground-truth frames as the training data. After all training input frames and label masks have been collected, a preprocessing operation is performed. As shown in Section II-B, since our network is based on the VGG-16 [14] network, the inputs have a size of $224 \times 224 \times 3$; thus, we must preresize all the training frames to the fixed size of $224 \times 224 \times 3$. Then, a mean subtraction is performed on each pixel. Since we regard the foreground object detection as a binary classification problem in our method, therefore, the corresponding training label frames are of the size of $224 \times 224 \times 2$. The pixel label value is given by the following:

$$y(p) = \begin{cases} 1, & \text{if class}(p) = \text{foreground} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where p denotes the pixel in the ground truth.

B. Network Architecture

The architecture of the proposed network is shown in Fig. 2. In contrast to previous work [15], the proposed method does not need to extract the background images. The input of this network is frames from different sequences, and the output is a probability map (one channel, the size of which is the same as the input).

Because a limited amount of training data is available; thus, a transfer learning method is adopted by pretraining a deep CNN on a large-scale image data set and then fine tuning the learned features for our task. In this letter, the model is fine-tuned on the VGG-16 [14] network (the dashed box in Fig. 2). We split the VGG-16 network into five blocks (V1, V2, V3, V4, and V5), with each block containing some convolution and max pooling operations. The sizes of the corresponding output feature maps are shown in Fig. 2. We can see that the lower blocks have a higher spatial resolution but contain more low-level local features, while the deeper blocks contain more high-level global features at a lower resolution.

Afterward, to gain multiscale features from the different layers, 3×3 convolution kernels are applied to the pretrained blocks. As shown in the second row of Fig. 2, the generated convolutional layers (C1, C2, C3, C4, and C5) are connected to the upper VGG blocks. The output feature maps maintain the same spatial resolution as the upper blocks with 128 channels.

Since foreground object detection is treated as a binary classification problem in this letter, the output masks show a great contrast between the foreground and the background, which means that the features of the foreground and the background in the input frames should also have a large difference. To extract this kind of contrast information, a contrast layer is added behind the output feature layer. The contrast layer is calculated as follows:

$$P_i = C_i - \text{AvgPool}(C_i). \tag{2}$$

Here, AvgPool is the average pooling operation with a kernel size of 3×3. In Fig. 3, we compare the detection results obtained by FCN architecture which without multiscale features and a new architecture (MFCN_) which without contrast layers. We can see that FCN results contain many holes and unconnected regions, results obtained by MFCN_ are also very coarse, and the boundaries of the foreground object are not well preserved, which makes the final segmented foreground masks much thinner and less accurate.

Finally, to exploit multiscale features from multiple layers, a set of deconvolution operations is used to upsample these features, creating an output probability map the same size as the input, as shown in the last rows of Fig. 2. The deconvolution kernels have the size of 3×3 and the stride is 2. Instead of upsampling the feature maps with a fixed ratio of {8, 16, 32}, as done in [17] for semantic segmentation, we adopt a stepwise upsampling strategy that produces more refined feature maps. First, the feature layer C5 is concatenated with its contrast feature layer P5 in the last dimension. Then, the concatenated feature is upsampled by 2 with the deconvolution operation. After the new deconvolution layer D5 is obtained, it is concatenated with the feature layer C4 and its contrast feature layer P4. Then, we upsample the concatenated feature as before and get the new deconvolution layer D4. After five stages of deconvolution operations, the feature maps with different scales are integrated and upsampled to the input size. This process can be expressed as follows:

$$D_{i-1} = \text{Deconv}(\text{Concat}(C_i, P_i, D_i)). \tag{3}$$

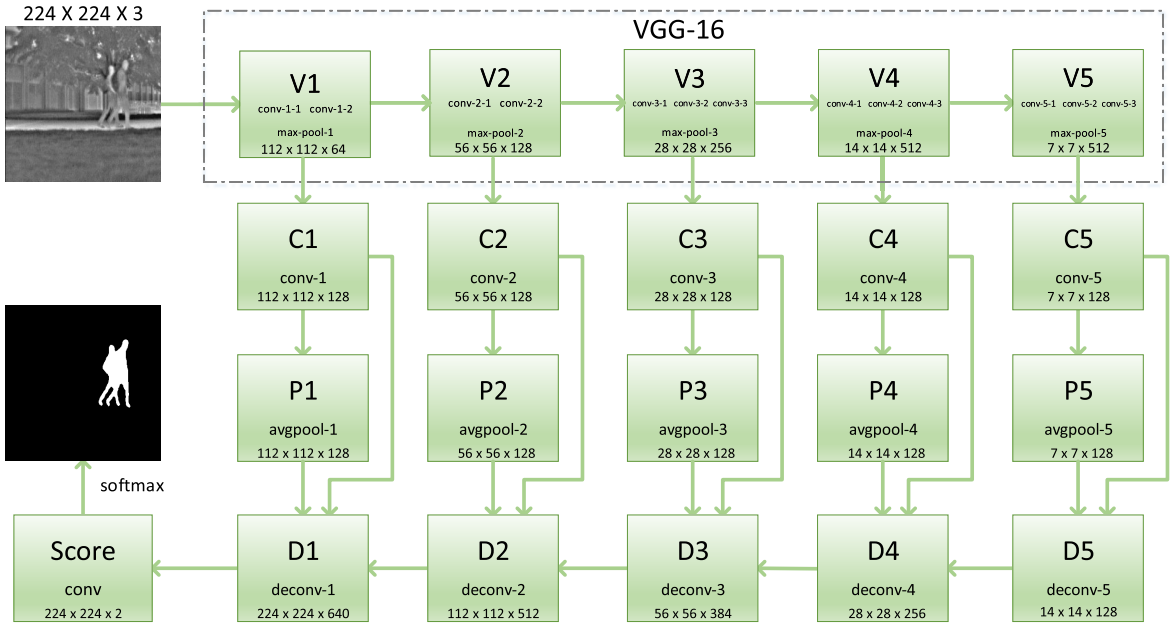

Fig. 2. Architecture of the proposed MFCN for IR foreground object detection. A FCN architecture covering multiscale convolution and deconvolution operations. With CNN features learned from multiple scales, the feature representation contains both category-level semantics and fine-grain details, which lead to more accurate foreground detection results.

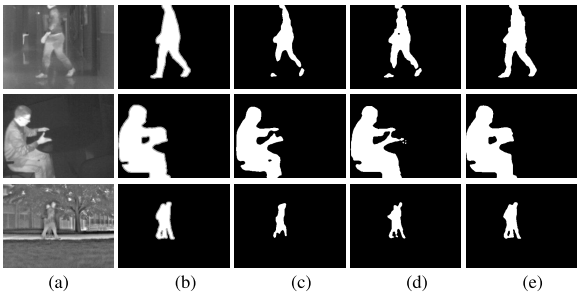

Fig. 3. Visual comparison of IR foreground object detection results obtained by different architectures. (a) Input frame. (b) Ground truth. (c) Detection results of FCN. (d) Detection results of MFCN_. (e) Detection results of MFCN.

In the end, a convolution kernel with a size of 1×1 is applied to the final deconvolution layer D1, and a score layer that contains two channels is obtained. Then, a softmax operation is used to produce the final foreground probability map.

For the loss function, we use the cross-entropy loss, which is defined as follows:

$$\text{Loss} = -\frac{1}{N}\sum_{p=1}^{N}[y_p \log(\hat{y}_p) + (1 - y_p)\log(1 - \hat{y}_p)]. \quad (4)$$

Here, N is the number of training image pixels, $y_p \in \{0, 1\}$ is the label of pixel p, and $\hat{y}_p$ is the predicted label.

C. Training Details

The proposed MFCN model is implemented in TensorFlow [19]. The layers from VGG-16 are initialized with pretrained weights [14], while other weights are randomly initialized with a truncated normal distribution $\mathcal{N}(0, 0.01)$. The AdamOptimizer method is used for updating the model parameters with a learning rate of 10^{-4}. During the training stage, the training data are augmented with horizontal flipping. Each sequence is trained for 20 epochs with a batch size of 5 frames.

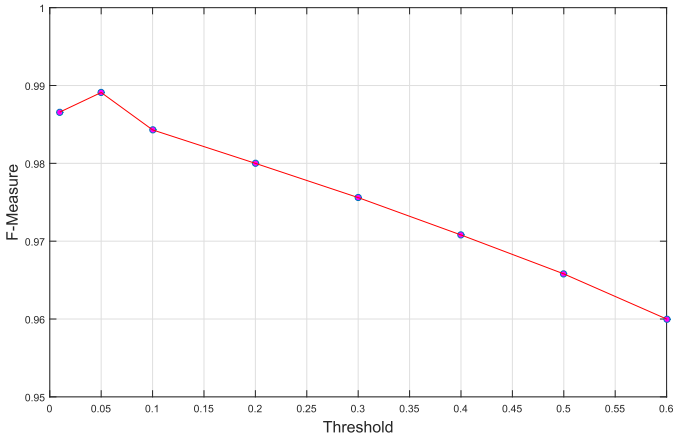

Fig. 4. FM scores of the proposed method evaluated with different threshold values.

D. Foreground Object Detection

During the foreground object detection stage, the foreground probability map is obtained with a softmax operation on the score layer. Then, a threshold is applied to the map and gets the final binary mask. Fig. 4 shows how the F-Measure (FM) scores vary for different threshold values. We can see that the value of 0.05 gives the best performance. Finally, a median filter with a size of 3×3 is applied to enhance the spatial coherency and to reduce the noise.

III. Experimental Results

A. Data Set and Evaluation Metrics

We evaluate the proposed MFCN-based IR foreground object detection method using image sequences from the

TABLE I
OVERALL FM SCORES ON ALL IR IMAGE SEQUENCES FOR DIFFERENT METHODS

Method	Recall	Specificity	FPR	FNR	PWC	Precision	F-Measure
MFCN	**0.9802**	**0.9998**	**0.0001**	**0.0197**	**0.0872**	**0.9941**	**0.9870**
CascadeCNN [16]	0.9461	0.9931	0.0069	0.0539	1.0478	0.8577	0.8958
IUTIS-5 [20]	0.7990	0.9952	0.0048	0.2010	1.1484	0.8969	0.8303
SubSense [12]	0.8161	0.9908	0.0092	0.1839	2.0125	0.8328	0.8171
KDE [7]	0.6725	0.9955	0.0045	0.3275	1.6795	0.8974	0.7423
SOBS [21]	0.6003	0.9957	0.0043	0.3997	1.9841	0.8857	0.6923
ViBe [8]	0.5704	0.9980	0.0019	0.4295	2.5253	0.9586	0.6877
GMM [5]	0.5691	0.9946	0.0054	0.4309	4.2642	0.8652	0.6621

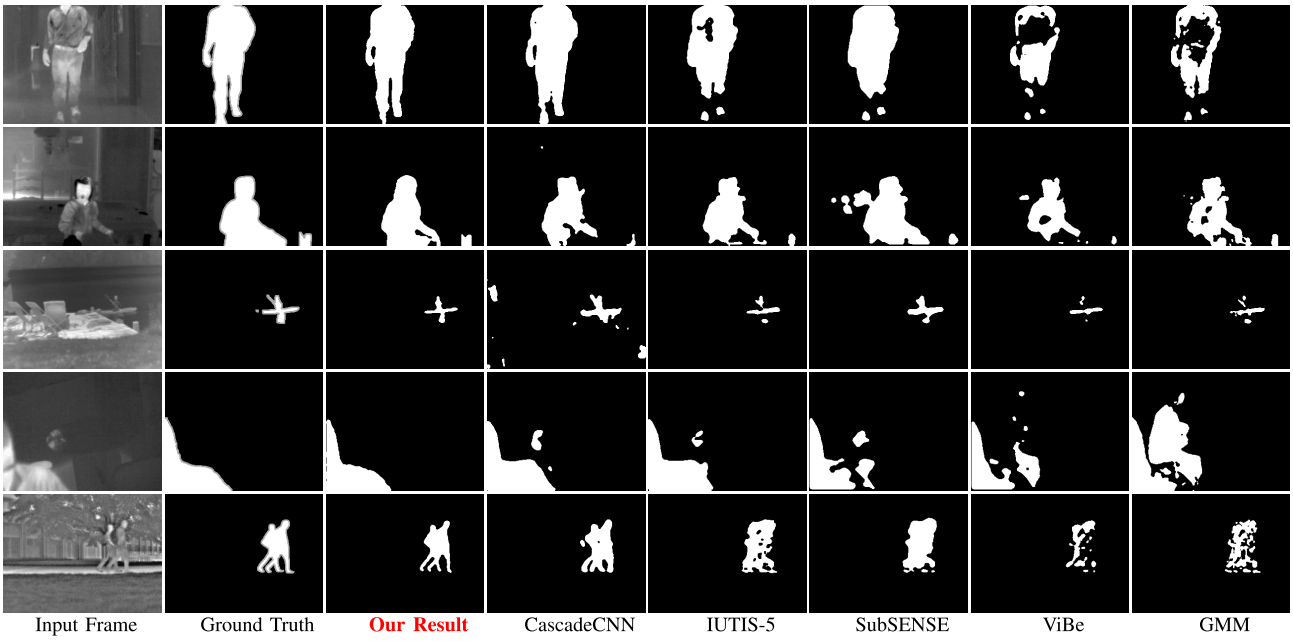

Fig. 5. Qualitative performance comparison for various sequences. (From top to bottom) *Corridor, dinningRoom, lakeSide, library*, and *park*. First column to the last column: input frame, ground truth, our result, CascadeCNN [16], IUTIS-5 [20], SubSENSE [12], ViBe [8], and GMM [5] detection results.

change detection challenge [22] thermal category data set, which contains five IR sequences, namely, "*corridor* (size: 320 × 240 and length: 5400 frames)," "*diningRoom* (size: 320×240 and length: 3700 frames)," "*lakeSide* (size: 320×240 and length: 6500 frames)," "*library* (size: 320 × 240 and length: 4900 frames)," and "*park* (size: 352 × 288 and length: 600 frames)." All sequences were obtained from realistic scenarios, and accurate human-constructed ground truths are available.

In order to make an exhaustive competitive comparison between different foreground object detection methods, seven different metrics were defined in [22]: recall (Re), specificity (Sp), false positive rate (FPR), false negative rate (FNR), percentage of wrong classifications (PWC), precision (Pr), and FM. Among these metrics, we are especially interested in the FM metric, which is commonly accepted as a good indicator of the overall performance of a background subtraction method. This metric is defined as follows:

$$\mathrm{FM} = 2 \cdot \frac{\mathrm{Re} \cdot \mathrm{Pr}}{\mathrm{Re} + \mathrm{Pr}}. \tag{5}$$

The FM metric represents a balance between recall and precision. As shown in [22], most state-of-the-art foreground object detection methods typically exhibit higher FM scores than these worse performing methods.

B. Performance Evaluation

1) Quantitative Evaluation: We compare the proposed method with some other classical and state-of-the-art IR foreground object detection methods, including the following: CascadeCNN [16], IUTIS-5 [20], SubSENSE [12], KDE [7], SOBS [21], ViBe [8], and GMM [5]. In Table I, we present a detailed performance comparison. For a specific metric, if a method obtains the best scores, the corresponding value is highlighted in bold. Using the standardized evaluation tool provided in [22], seven metric scores are reported. For the PWC, FNR, and FPR metrics, lower values indicate higher accuracy, while for the recall, recall, specificity, and FM metrics, higher values indicate a better performance. The results of other methods are from the website.[1] We can see that the proposed method obtains the best performance in all metrics, especially with its FM score of 0.9870, and outperforms the second best method with a considerable margin. As demonstrated in [16], a method with an FM score

[1] www.changedetection.net.

above 0.94 and a PWC score below 0.9, the detection results may be considered almost as good as the ground truth, since a simple dilation (or erosion) of one or two pixels of the ground truth may result in an FM score drops from 1.0 to about 0.94. This again shows the efficiency of the proposed method.

2) Qualitative Evaluation: To make a better visual comparison of the detection results under different scenarios, we select the following frames: the 1959th frame from the *corridor* sequence, the 3166th frame from the *diningRoom* sequence, the 6067th frame from the *lakeSide* sequence, the 4470th frame from the *library* sequence, and the 362th frame from the *park* sequence. As shown in Fig. 5, the first column displays the input frames and the second column shows the corresponding ground truth. From the third column to the eighth column, the foreground object detection results are given for the following methods: our method (MFCN), CascadeCNN, IUTIS-5, SubSENSE, ViBe, and GMM. Visually, we can see that our results appear superior to those of the other methods and are closest to the ground truth, which is in good agreement with the quantitative evaluation results.

3) Generalization to Other Spectra: The input of our network architecture is about the size of $224 \times 224 \times 3$, which inspires us to consider whether the proposed method can be generalized to visible videos. We also evaluated the MFCN network on various RGB sequences from the change detection challenge data set [22]. We achieved an average FM score of 0.96, which is slightly worse than the results for the IR images, but better than many other foreground detection algorithms. So we can see that the proposed method has a strong universality. It can be applied not only to IR images but also to other visible spectral images.

4) Real-Time Performance: Processing speed is a critical factor to be considered before selecting an IR foreground object detection method. During the detection stage, the proposed MFCN model is run on a 4.0-GHz Intel Core-i7 7700 CPU with an NVIDIA GTX 1060 GPU and an Ubuntu 16.04 operating system. The average processing time per frame is 0.0372 s (nearly 27 frames/s), which shows real-time potential.

IV. Conclusion

In this letter, an MFCN architecture for IR foreground object detection is presented. Benefitting from the hierarchical convolutional features learned from multiple scales, the proposed method achieves a much higher foreground detection accuracy, which shows the effectiveness of deep features compared with conventional hand-crafted features. Experiments were performed on various IR image sequences, showing that this method outperforms recent state-of-the-art methods and has the potential for real-time applications. Currently, the main limitation of the MFCN is that it is a supervised method and human-constructed ground truths are needed to train the model. However, the MFCN can also be trained with the results produced by other unsupervised methods, but it may reduce the accuracy. As a future work, we plan to combine traditional unsupervised methods and the proposed CNN-based method to compensate each other for better performance.

References

[1] J. Han, Y. Ma, J. Huang, X. Mei, and J. Ma, "An infrared small target detecting algorithm based on human visual system," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 3, pp. 452–456, Mar. 2016.

[2] Y. He, M. Li, J. Zhang, and J. Yao, "Infrared target tracking based on robust low-rank sparse learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 2, pp. 232–236, Feb. 2016.

[3] C. L. P. Chen, H. Li, Y. Wei, T. Xia, and Y. Y. Tang, "A local contrast method for small infrared target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 574–581, Jan. 2014.

[4] T. Bouwmans, "Traditional and recent approaches in background modeling for foreground detection: An overview," *Comput. Sci. Rev.*, vol. 11, pp. 31–66, May 2014.

[5] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 1999, pp. 246–252.

[6] M. S. Allili, N. Bouguila, and D. Ziou, "A robust video foreground segmentation by using generalized Gaussian mixture modeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, May 2007, pp. 503–509.

[7] 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.

[8] 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.

[9] M. Heikkila and M. Pietikäinen, "A texture-based method for modeling the background and detecting moving objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 657–662, Apr. 2006.

[10] S. Liao, G. Zhao, V. Kellokumpu, M. Pietikäinen, and S. Z. Li, "Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 1301–1306.

[11] P.-L. St-Charles and G.-A. Bilodeau, "Improving background subtraction using local binary similarity patterns," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Mar. 2014, pp. 509–515.

[12] 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. 359–373, Jan. 2015.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[14] K. Simonyan and A. Zisserman. (Sep. 2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556

[15] M. Braham and M. Van Droogenbroeck, "Deep background subtraction with scene-specific convolutional neural networks," in *Proc. IEEE Int. Conf. Syst., Signals Image Process.*, May 2016, pp. 1–4.

[16] 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. 2016.

[17] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3431–3440.

[18] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Dec. 2015, pp. 58–66.

[19] M. Abadi *et al.* (Mar. 2016). "TensorFlow: Large-scale machine learning on heterogeneous distributed systems." [Online]. Available: https://arxiv.org/abs/1603.04467

[20] S. Bianco, G. Ciocca, and R. Schettini. (May 2015). "How far can you get by combining change detection algorithms?" [Online]. Available: https://arxiv.org/abs/1505.02921

[21] L. Maddalena and A. Petrosino, "The SOBS algorithm: What are the limits?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2012, pp. 21–26.

[22] 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.