IEEE Open Journal of
Intelligent Transportation
Systems

Received 3 October 2021; revised 5 January 2022, 25 February 2022, and 13 April 2022; accepted 19 April 2022.
Date of publication 25 April 2022; date of current version 5 May 2022.

Digital Object Identifier 10.1109/OJITS.2022.3170075

Cascaded Feature-Mask Fusion for Foreground Segmentation

CHUANYUN XU[1,2], HUAN LIU[1], TENGHUI LI[1], YANG ZHANG[2], TIAN LI[3], AND GANG LI[1]

[1]School of Artificial Intelligence, Chongqing University of Technology, Chongqing 401135, China

[2]College of Computer and Information Science, Chongqing Normal University, Chongqing 401331, China

[3]Computer Science Department, RWTH Aachen University, 52074 Aachen, Germany

CORRESPONDING AUTHOR: H. LIU (e-mail: with.him777@gmail.com)

This work was supported in part by the China Chongqing Science and Technology Commission under Grant cstc2020jscx-msxmX0086; in part by the China Chongqing Banan District Science and Technology Commission Project under Grant 2020QC413; and in part by the China Chongqing Municipal Education Commission under Grant KJQN202001137.

ABSTRACT Foreground segmentation aims at extracting moving objects from the background in a robust manner under various challenging scenarios. The deep learning-based methods have achieved remarkable improvement in this field. These methods produce semantically correct predictions based on extracted rich semantic features yet perform poorly on segmentation of edge details. The main reason is that the high-level features extracted by the deep network lose the high-frequency information for the successful edge segmentation. On this basis, we propose a novel segmentation network with a cascade architecture to refine segmentation results step by step by introducing detailed information into high-level features. The network recorrects and optimizes the segmentation maps in each step so that more accurate segmentation results are obtained. Furthermore, we evaluate our approach on the challenging CDnet2014 dataset and achieve an F-measure of 0.9868. Our approach thus outperforms previous methods, such as FgSegNet_v2, FgSegNet, BSPVGan, Cascade CNN, IUTIS-5, WeSamBE, DeepBS, and GMM-Stauffer.

INDEX TERMS Deep learning, feature-mask fusion, foreground segmentation, high-level features, video surveillance.

I. INTRODUCTION

THE EXTRACTION of moving objects from video sequences plays an important role in visual applications, such as video surveillance [1], human tracking [2], action recognition [3], traffic monitoring [4], [5], [6], motion estimation and anomaly detection [7]. Various extraction methods have been proposed for foreground segmentation. Conventional approaches perform well only in a certain type of scenario and poorly for complex scenes. Compared with traditional methods, deep learning-based approaches have superior segmentation performance owing to their powerful capability to extract feature representations from images. Nevertheless, they have limited in terms of edge detailed segmentation, mainly because (1) the successive convolution and pooling operations lead to the decline of the final resolution when upsampling to the original size and (2) The features in the different stages of the network have different recognition abilities. In the high stage, the high-level features are rich in semantic information due to the large field of view, but the prediction space is coarse owing to the lack of spatial details. In the low stage, low-level features have finer spatial information but poor semantic information because of their narrow field of view. In a word, the high-level features extracted by the deep convolutional neural network (CNN) lack detailed information for edge segmentation, whereas the low-level features are essential for accurately predicting boundary details. In addition, for scenes with difficult foreground segmentation, such as scenes with small moving objects or short moving distances, the segmentation performance of the existing models are not good; i.e., the segmented edges are not accurate if only high-level features

The review of this article was arranged by Associate Editor Chi-Hua Chen.

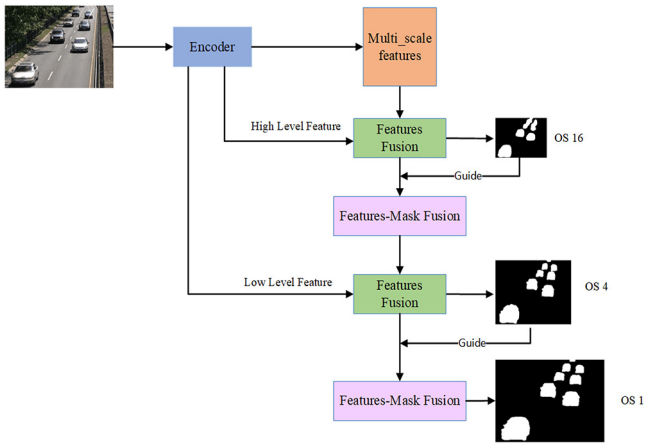


FIGURE 1. Cascaded feature-mask fusion network. One of the decoder inputs comes from the different-level features extracted by the encoder, and the features and the corresponding prediction mask are taken together to regress the segmentation performance. The cascaded design allows the capture of more edge details to gradually refine the mask.

are used whereas the segmented targets are easily missed if only low-level features are used. How to effectively use features at different levels is thus an issue that deserves more attention.

A cascaded feature-mask fusion network (CFMFN) is proposed in this research to fuse features at multiple levels and thus refine edge segmentation (Fig. 1). The proposed method uses high-level features to predict the basic contour of the foreground and low-level features to optimize the details. A mask based on feature fusion is used to fix regions of large error. The CFMFN recursively fuses the high-level features, low-level features, and the masks generated from the two types of feature. The input of each layer comprises both the initial segmentation and all outputs from previous levels. Through a multi-level cascade, the model focuses on the details presented in the initial segmentation to refine the boundary details. This design allows the CFMFN to learn to adaptively fuse the features of different scales in refining the segmentation at the finest level. In addition, the CFMFN provides highly accurate segmentation results when only a few training examples are used for training without the consideration of temporal data.

The main contributions of this research are summarized as follows.

1) We propose a foreground segmentation method based on multi-level feature-mask fusion, which gradually refines and corrects the local boundaries to achieve accurate segmentation.
2) We propose a novel cascaded encoder-decoder network, which is a decoder level cascade with better performance, rather than the reuse of the model of the previous methods.
3) The algorithm is evaluated on the CDnet2014 dataset [8] and found to perform better than many existing methods, especially in the difficult segmentation scenarios.

II. RELATED WORKS

A. FOREGROUND SEGMENTATION

Segmenting foreground objects from the video imagery is an active research topic in the field of computer vision. Traditional algorithms, namely unsupervised learning algorithms, mainly rely on background modeling. Stauffer and Grimson [1] and Zivkovic [2] first proposed a foreground segmentation method based on the Gaussian Mixture Model (GMM) to model each pixel as a background or foreground pixel, but their method cannot handle a rapidly changing background and its parametric nature is computationally inefficient. Various non-parametric methods [9], [10], [11], [12], [13] have been proposed for improved computational efficiency. In recent years, studies on the application of neural networks to foreground segmentation [14], [15], [16] have achieved impressive results. References [17], [18], [19] adopted the Generative Adversarial Networks (GAN), whereby the generator learns the mapping from the background and the current image for the foreground mask, and the discriminator then learns a loss function for the training of this mapping by comparing the groundtruth and predicted output through observing the input image and background. References [20], [21], [22], [23] trained the network by combining image frames with the generated background model. The Multi-scale and Cascaded CNNs [24], [25], [26], [27] improve the segmentation quality by acquiring multi-scale information, resulting in improved segmentation performance. References [28], [29], [30], [31], [32] considered the temporal data in a video sequence by designing different types of end-to-end three-dimensional CNN to track the temporal changes in the video sequence and avoid using background models for training.

B. ENCODER-DECODER NETWORKS

Encoder-decoder networks comprise an encoder module and decoder module. The encoder reduces the resolution of the feature maps and extracts higher semantic information whereas the decoder gradually recovers the spatial details to produce sharp segmentation results. Encoder-decoder networks have been successfully applied to human pose estimation [33], object detection [34], semantic segmentation [35], [36], [37] and other computer vision tasks. The encoder-decoder network is therefore adopted in this study to obtain more semantic and boundary information through a refined cascade.

C. DILATED CONVOLUTION

Dilated convolution refers to injecting holes into the standard convolution map to increase the receptive field without losing too much detailed information. This kind of convolution has been widely used in semantic segmentation recently. Models such as DeepLab [36] apply several parallel dilated convolutions with different rates to capture multi-scale information. In this study, a multi-scale feature fusion module is added between the encoder and decoder to

capture contextual information of the image with different proportions.

D. CASCADE NETWORK

Multi-scale analysis of cascade networks leverages in many computer vision tasks, such as edge detection [38], object detection [34], [39], and segmentation [35]. In particular, many methods predict independent results for different stages and merge them to obtain multi-scale information. CascadePSP [40] is a high-resolution segmentation model that uses global and local refinement. It refines the downsampled image for global refinement using a cascaded design. The Local step then refines details in full resolution using image crops. On this basis, in the global refinement, the mask generated by fusing features from different layers is used as one of the inputs for the next finer level and the low-level features extracted by the encoder are fused instead of applying cropping to obtain more boundary details and contextual semantic information of the image for local refinement.

E. FEATURE FUSION

The use of a feature map that mainly or only comprises features of a layer may lead to poor detection performance. Because low-level features contain more detailed information and less semantic owing to fewer convolution operations, while high-level features have stronger semantic information and less perceptive of details. It is important to fuse features across feature layers [34], [39], [41] to improve feature effectiveness. On the basis of feature fusion, feature-mask fusion is added in this paper. The fusion of low-level features and high-level features takes into account the accuracy and robustness of network discrimination. By combining features at different levels and masks at different scales (especially matching of low-level features and fine masks), significant performance optimization is achieved not only at the shallow level of the network but also at the deep level.

III. METHOD

Foreground segmentation is a pixel-level classification task where the size of the predicted output is equal to the size of the input image. The overall structure of the network is therefore designed according to the encoder-decoder structure. The encoder is typically a model pretrained on classification tasks (e.g., VGG16 [42] and Resnet50 [43]). The encoder usually allows the whole network to converge more rapidly with the use of the pretrained model as the backbone network whereas the decoder conducts successive convolution and upsampling to generate fine segmentation results. In addition, a multi-scale feature fusion module is added at the end of the encoder to enlarge the field of view in the network. The CFMFN thus comprises an encoder, multi-scale feature fusion module, and decoder (Fig. 2).

A. ENCODER

The encoder of the CFMFN selects the pretrained ResNet50 as the backbone network and removes its final fully connected layer.

The output stride (OS) of the original ResNet50 in stage4 takes a value of 32. The OS is defined as the ratio of the spatial resolution of the input image to the final output resolution. A large OS leads to the loss of image sharpness, which makes it difficult for the decoder to restore pixel-level image details that are lost in the extraction process, and a small OS is thus usually used for segmentation tasks. The encoder of the CFMFN therefore adjusts the convolution stride in stage4 of the original ResNet50 to a value of 1 and the dilation to a value of 2.

B. MULTI-SCALE FEATURE FUSION MODULE

The multi-scale feature fusion module includes a 1 x 1 Conv layer, three 3 x 3 Conv layers, and a global average pooling (GAP) layer.

- GAP merges global context information into the multi-scale feature fusion module.
- The three 3 x 3 Conv layers are dilated convolutions with dilation rates of 4, 8, and 12 that capture multi-scale information and connect the output results of all branches in parallel.
- As an input of the decoder, the 1 x 1 Conv layer is used to obtain spatial dimensions consistent with the features of the encoder output.

C. DECODER

General methods usually directly upsample by a factor of 16 or 8 on the feature map, which is finally obtained by the decoder to generate a prediction equal in the size to the input image. This one-step decoding operation does not properly restore details lost in the pooling operation, and thus does not improve the segmentation accuracy.

The decoder of the CFMFN is therefore designed in a cascading fashion. Each step upsamples the fused features F', which comprise three parts: the feature map F, the high (low)-level features of the encoder, and the mask generated by the fusion. Thus, F' not only contains semantic information but also introduces certain detailed information, which effectively improves the accuracy of segmentation.

Before fusing the low-level information, we apply 1×1 convolution to low-level features to reduce the number of channels. This decoding process is expressed as in Equation (1).

$$F' = Conv\{Concat(F, features, mask)\}, \tag{1}$$

where the features are the high-level and low-level features from the encoder, which are the output of stage1 (OS of 4) and the output of stage4 (OS of 16) of the encoder.

CFMFN refines the image using a 3-Level cascade with output strides (OS) of 16, 4, and 1. Besides the final stride 1 output, our model also generates intermediate

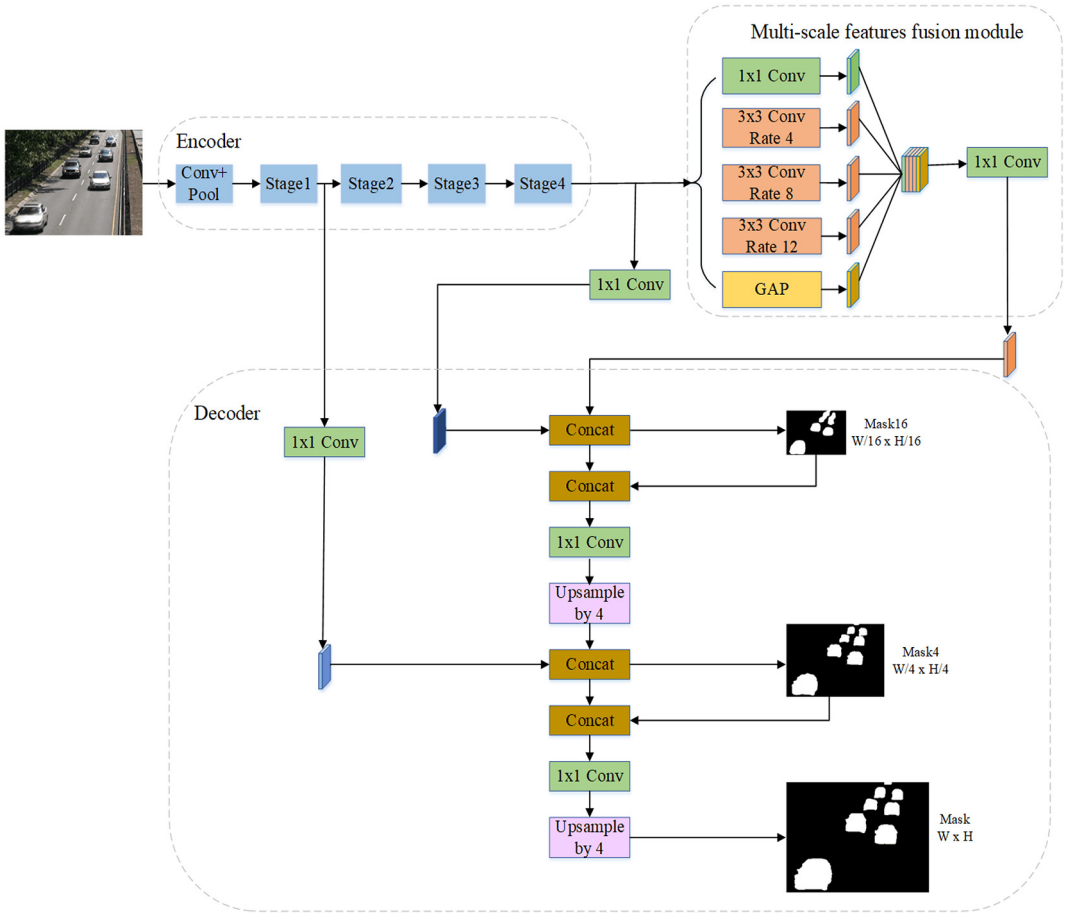


FIGURE 2. Flow of the CFMFN architecture.Our proposed CFMFN by employing an encoder-decoder structure. After encoding in the encoder module, The Multi-scale feature fusion module catches multi-scale contextual information by applying atrous convolution at multiple scales, while the decoder refines the segmentation results by cascaded fusion with different output strides(OS) and the corresponding mask. In this paper, we use output strides of 16,4,1.

stride 16 and stride segmentations which focus on fixing the overall structure of the input segmentation. the model takes the two outputs of the OS16 and OS4 denoted as $Mask_{16}$ and $Mask_4$. $Mask_{16}$ relates more to the global judgments of the image and not to local exploration. Because the introduced high-level features have stronger semantics information with the poor perception of details. $Mask_4$ introduces low-level features for edge detailed segmentation and is more refined than the coarse $Mask_{16}$. The network focuses on guiding the overall structure of the next input segmentation to provide the flexibility to correct local error boundaries. The CFMFN therefore gradually corrects segmentation errors while maintaining the initial segmentation details. With the combined effects of $Mask_{16}$ and $Mask_4$, the CFMFN roughly predicts the moving objects and corrects larger errors at the coarse level. At the fine level, the more robust features provided by the coarse level can be used to address the boundary details of the image to be processed.

Fig. 3 shows that the final *Mask* is much finer than $Mask_{16}$, which demonstrates the effectiveness of cascaded feature-mask fusion in edge detailed segmentation.

D. LOSS

It follows from the above that the gradual introduction of low-level features and the gradual refinement of the mask significantly improves the accuracy of segmenting moving objects. However, three predictions are made during the decoding. Therefore, we need to consider how to learn using these three predictions when designing the loss.

$Mask_{16}$, $Mask_4$, and *Mask* are generated in the CFMFN decoder with dimensions (W/16, H/16), (W/4, H/4), and (W, H) respectively. The core idea of the CFMFN is to gradually refine the mask and the design of the loss for $Mask_{16}$ and $Mask_4$ should thus focus more on how to roughly segment the moving objects, the CFMFN uses the cross-entropy loss with balanced weights. In our experiments, we observe that $Mask_4$ already provides sufficient accuracy, and the final *Mask* should pay more attention to addressing the local boundaries and segmentation details. Therefore, in the last layer, we use the Euclidean distance after L2 normalization instead of the cross-entropy loss. The idea of treating the last part as a regression task to correct the segmentation errors is similar to the idea of boundary regression for object detection. The results show the effective segmentation of the

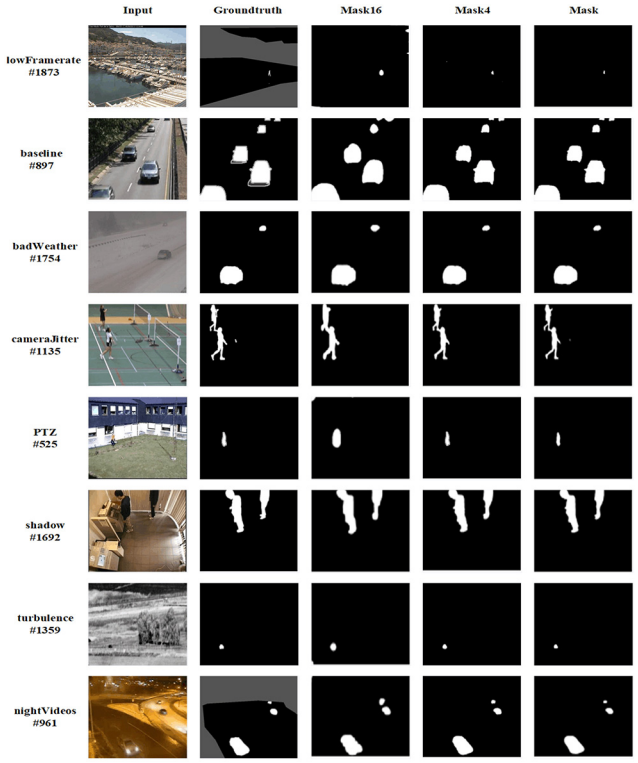


FIGURE 3. Differences between the masks generated by output strides of 16, 4 and 1. The 3-Level input model uses small-scale intermediates (mask16, mask4) that, though inaccurate, capture structural information to be refined at the later stage.

boundary regions. The loss for output stride 1 (L^1) can be written as in Equation (2).

$$L^1 = 2 * (1 - cos(Mask, Groundtruth)). \tag{2}$$

In short, different loss functions are applied to different strides because the coarse refinement focuses on the global structure while ignoring local details, whereas the finest refinement aims to achieve pixel-wise accuracy by relying on local cues. So the final loss of the CFMFN is the sum of three loss functions:

$$L = L^{16}_{CE} + L^{4}_{CE} + L^{1}, \tag{3}$$

where L^s, and L^s_{CE} stand for the Euclidean distance after L2 normalization, and cross-entropy loss for output stride s respectively.

IV. EXPERIMENTS

A. DATASET

CDnet2014 as the largest publicly available dataset for foreground segmentation, is widely used in foreground/background segmentation studies, with a total of 150,000 frames of pixel-level annotated data for 53 scenes in 11 categories, which are named badWeather, baseline, cameraJitter, dynamicBackground, intermittentObjectMotion, lowFramerate, nightVideos, PTZ, shadow, thermal, and turbulence. Each category has four to six video sequences, each containing 600 to 7999 frames, with spatial resolutions ranging from 320 × 240 to 720 × 576 pixels. The dataset covers a variety of challenging scenes involving illumination changes, hard shadows, highly dynamic background motion, and camera motion.

Seven metrics provided by the CDnet2014 dataset are used to evaluate the performance of the CFMFN: Recall (Re), Precision (Pr), Specificity (Sp), False Negative Rate (FRN), False Positive Rate (FNR), Percentage of Wrong Classifications(PWC) and F-Measure (FM). Among them, FM is used as a comprehensive performance metric of the model performance ranking on the CDnet2014 dataset, so it is taken it as the main metric of the CFMFN performance evaluation in the present experiment. Given Re, Pr, and FM is defined by:

$$Pr = \frac{TP}{TP + FP}, \tag{4}$$

$$Re = \frac{TP}{TP + FN}, \tag{5}$$

$$FM = \frac{2 \times Pr \times Re}{Pr + Re}, \tag{6}$$

where TP, FP, and FN are stand for True Positive, False Positive, and False Negative respectively.

B. TRAINING PROTOCOL

The PyTorch [44] framework is used to implement the model in the experiments, following the same training procedure as used in a previous study [25] and keeping the pretrained weights of the original ResNet50 network. Two groups of experiments are carried out for each scene, whereby 50 and 200 frames are selected for the training set, and the remaining frames for the test set.

After shuffling the training set, 20% of the data are used for validation and 80% are used to train the model. Set the learning rate to 0.001, epochs to 100, the momentum of SGD (stochastic gradient descent) to 0.9, and the batch size to 32.

We use decreasing learning rate for training optimization and BatchNorm, Dropout to optimize the convolutional neural network. And the Fig. 4 shows that the difference between the training error and the validation error is small and in a stable state. We design an early stopping mechanism for up to 60 periods of training. The training ends ahead of time when the comprehensive performance metric FM of the verification set no longer improves over 20 epochs.

Owing to the high imbalance between the background and foreground pixels in the scene, the CFMFN uses balanced weights during training to reduce the problem of the imbalanced data classification.

In addition, because the output of the sigmoid function is within the range [0, 1], it is used as a probability value. A threshold of 0.5 is applied to the processing to obtain discrete binary labels of the foreground and background.

C. RESULTS

To reduce the burden of the ground-truth annotation, [17], [24], [25], [26] only used few frames for training

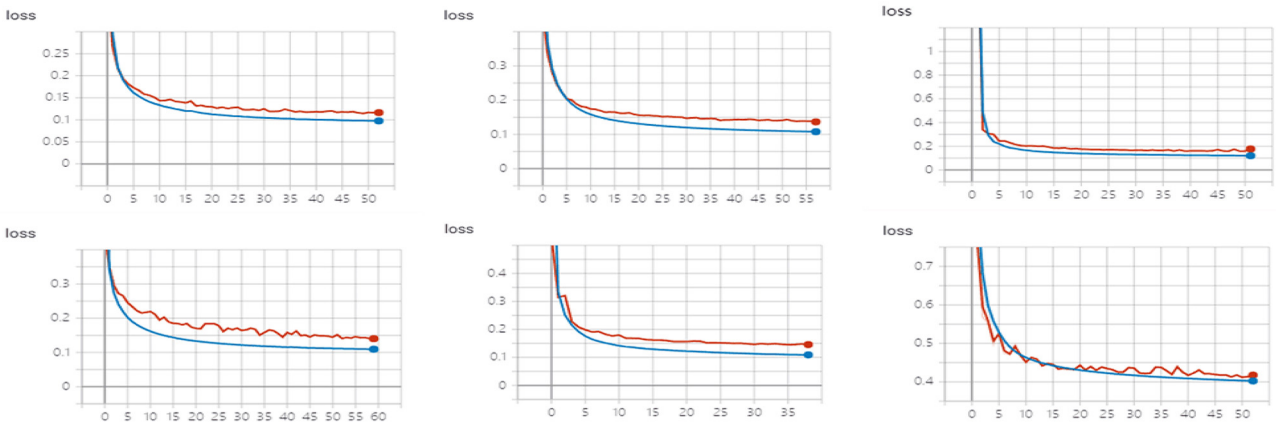


FIGURE 4. The loss function of the CFMFN while trained on some categories. The red line represents the validation loss, and the blue line represents the training loss. The horizontal axis is the number of epochs, and the vertical axis is the value of loss function. Using the early stopping mechanism, the training is stopped when the loss no longer decreases. The training scenes from left to right in the first row are turbulence2 (turbulence), office (baseline), sidewalk (cameraJitter), and the training scenes from left to right in the second row are boats (dynamicBackground), zoomInZoomOut (PTZ), backdoor (shadow).

TABLE 1. Test results obtained by manually and randomly selecting, 50 and 200 frames from CDnet2014 dataset across 11 categories.

Category	Re		Sp		Pr		FPR		FNR		PWC		FM	
	50f	*200f*	*50f*	*200f*	*50f*	*200f*	*50f*	*200f*	*50f*	*200f*	*50f*	*200f*	*50f*	*200f*
baseline	0.9871	0.9950	0.9998	0.9999	0.9924	0.9960	0.0002	0.0001	0.0129	0.0050	0.0589	0.0254	0.9897	0.9955
camJit	0.9797	0.9895	0.9997	0.9994	0.9672	0.9843	0.0013	0.0006	0.0203	0.0105	0.1905	0.0921	0.9734	0.9869
badWeat	0.9144	0.9704	0.9997	0.9998	0.9760	0.9827	0.0003	0.0002	0.0856	0.0296	0.1170	0.0497	0.9434	0.9765
dynaBg	0.9653	0.9857	0.9999	0.9999	0.9764	0.9892	0.0001	0.0001	0.0347	0.0143	0.0316	0.0136	0.9708	0.9875
intermit	0.9730	0.9871	0.9985	0.9997	0.9796	0.9960	0.0015	0.0003	0.0270	0.0129	0.3154	0.0931	0.9761	0.9914
lowFram	0.8782	0.9139	0.9996	0.9997	0.8658	0.8778	0.0004	0.0003	0.1218	0.0861	0.0992	0.0570	0.8714	0.8933
nightVid	0.9264	0.9661	0.9987	0.9994	0.8992	0.9659	0.0013	0.0006	0.0736	0.0339	0.2386	0.1113	0.9114	0.9660
PTZ	0.9216	0.9997	0.9997	0.9999	0.7946	0.9725	0.0003	0.0001	0.0784	0.0223	0.0592	0.0252	0.9311	0.9751
shadow	0.9742	0.9909	0.9994	0.9998	0.9797	0.9927	0.0006	0.0002	0.0258	0.0091	0.1423	0.0538	0.9768	0.9918
thermal	0.9495	0.9801	0.9985	0.9995	0.9720	0.9884	0.0015	0.0005	0.0505	0.0199	0.2860	0.1035	0.9605	0.9842
turbul	0.9538	0.9736	0.9998	0.9999	0.9588	0.9790	0.0002	0.0001	0.0462	0.0264	0.0476	0.0273	0.9562	0.9763
Overall	0.9476	0.9755	0.9993	0.9997	0.9553	0.9750	0.0007	0.0003	0.0524	0.0245	0.1442	0.0593	**0.9510**	**0.9750**

and validation. Therefore, the present experiments use the same training set as in [25], which is obtained by manually and randomly selecting 50 and 200 frames from the CDnet2014 dataset.

The experiments only evaluate models using the test frames, in other words, the training frames are not included in the reported performances. The results are presented in Tab. 1, with the first 11 rows giving the average results of each category, and the last row giving the average results for the 11 categories. With the settings mentioned above, the CFMFN has an overall FM of 0.9510 with 50-frame experiments and 0.9750 with 200-frame experiments.

The CFMFN provides high accuracy in foreground segmentation when using 200 frames. The overall FM is highest for the baseline category (0.9955) and lowest for the lowFramrate category (0.8933). Only comparing the test frames, the best models reported on the official website, FgSegNet_v2 [26] and FgSegNet [25], have FM values of 0.8897 and 0.8816 on the lowFramrate respectively, and the CFMFN thus performs better.

The FM value inevitably decreases when the number of training samples is reduced from 200 to 50 frames. Especially for the nightVideos, the FM is 0.0546 lower than that when the training set has 200 frames. However, the CFMFN still generates acceptable results with an average overall FM of 0.9510 across 11 categories, indicating that the CFMFN works robustly in challenging scenarios.

D. COMPARION WITH THE STATE-OF-THE-ART

The CFMFN is compared with several methods described in related works and the best algorithms reported on the official website, namely FgSegNet_v2, FgSegNet, BSPVGan [17], Cascade CNN [24], IUTIS-5 [45], WeSamBE [46], DeepBS [47], and GMM-Stauffer [1]. Among them, IUTIS-5, WeSamBE, and GMM-Stauffer are unsupervised methods. The comparison results are shown in Tab. 2. The number of training frames is inconsistent across the deep learning algorithms and the traditional algorithm does not require a training set, we need to consider all ground-truth provided by the CDnet2014 dataset in comparing the results for our method the results for the previous methods. According to the tabulated data, the FM value is much higher for deep learning methods than for traditional models, especially in challenging categories such as PTZ (camera motion) and nightVideos (low light at night).

The FM value of the CFMFN is 0.71% and 0.13% higher than that for FgSegNet_v2 in the lowFramerate and turbulence categories respectively, and a little worse than FgSegNet_v2 in the other categories, with there being a slight

TABLE 2. A comparison among eight methods across 11 categories. Each row shows the results for each method.

Methods	F-measure											Overall
	baseline	*camJit*	*badWeat*	*dynaBg*	*intermit*	*lowFrame*	*nightVid*	*PTZ*	*shadow*	*thermal*	*turbul*	
Ours	0.9973	0.9895	0.9846	0.9947	0.9923	**0.9650**	0.9745	0.9906	0.9928	0.9907	**0.9828**	**0.9868**
FgSegNet_v2	0.9980	0.9961	0.9900	0.9950	0.9939	0.9579	0.9816	0.9936	0.9966	0.9942	0.9815	0.9890
FgSegNet	0.9975	0.9945	0.9838	0.9939	0.9933	0.9558	0.9779	0.9893	0.9954	0.9923	0.9776	0.9865
BSPVGan	0.9830	0.9890	0.9640	0.9780	0.9830	0.8630	0.9010	0.9490	0.9360	0.9760	0.9310	0.9501
Cascade CNN	0.9786	0.9758	0.9451	0.9658	0.8505	0.8804	0.8926	0.9344	0.9593	0.8958	0.9215	0.9272
WeSamBE	0.9310	0.7440	0.7970	0.7390	0.8690	0.7960	0.8610	0.6600	0.5930	0.3840	0.7540	0.7390
DeepBS	0.9580	0.8990	0.8647	0.8761	0.6097	0.5900	0.6359	0.3306	0.9304	0.7583	0.8993	0.7593
IUTIS-5	0.9567	0.8332	0.8289	0.8902	0.7296	0.7911	0.5132	0.4703	0.9084	0.8303	0.8507	0.7820
GMM-Stauffer	0.8320	0.6990	0.6040	0.7540	0.5810	0.6860	0.7430	0.6310	0.4630	0.2010	0.5560	0.6140

TABLE 3. Ablation study of the loss design.

Methods	F-measure				
	port_0_17fps	*tramCrossroad_1fps*	*tunnelExit_0_35fps*	*turnpike_0_5fps*	*lowFramerate*
Ours	**0.8834**	0.9908	**0.9915**	**0.9943**	**0.9650**
FgSegNet_v2	0.8356	0.9934	0.9903	0.9918	0.9528

difference of 0.22% in the overall performance. However, what we need to explain here is that since FgSegNet_v2 was proposed, its performance on the CDnet2014 official website remains the first. This is because it has achieved good performance on CDnet2014 dataset, with FM of most categories very close to 1. Thus the overall performance is relatively hard to improve, but our algorithm still achieves superior performance, outperforming many existing methods. Compared with FgSegNet, the CFMFN has more advantages on badWeather, dynamicBackground, lowFramerate, PTZ, and turbulence categories, and its overall performance is higher than that of FgSegNet by 0.03%. That is to say, the CFMFN has superior segmentation performance relative to the other advanced algorithms in most categories. In particular, the CFMFN has superior segmentation performance in the difficult scenes of the lowFramerate category. In conclusion, the CFMFN outperforms FgSegNet, BSPVGan, Cascade CNN, IUTIS-5, WeSamBE, DeepBS, and GMM-Stauffer in terms of the overall FM by 0.03%, 3.67%, 5.69%, 24.78%, 22.75%, 20.48%, and 37.28% respectively. In other words, the CFMFN outperforms not only traditional methods but also other deep learning based methods in terms of the overall performance (especially in terms of the robustness and effectiveness).

Existing methods performs better in almost all categories, except lowFrameRate category where it performs poorly compared to other categories. This low performance is primarily due to a challenging video sequence (port_0_17fps scenes in lowFrameRate category), where there are extremely small foreground objects in dynamic scenes with gradual illumination changes. In this case, the network may pay more attention to the major class (background) but less attention to the rare class (foreground), resulting in misclassifying very small foreground objects. However, the proposed method still improves over the best method by some margins in this category. It can be seen from the experimental results(Tab. 3) that our performance is improved by almost 5% over the optimal model, which can prove the superiority of our algorithm in this scenario.

Finally, the example results in Fig. 5 show the segmentation performance of several methods for typical complex scenarios. It is observed that the CFMFN can segment the local boundaries of large-scale objects (Fig. 5 traffic#1146) and small-scale objects (Fig. 5 blizzard#3108) more accurately than the other methods. Even when facing tiny foreground objects (Fig. 5 sidewalk#1155) and poor illumination (Fig. 5 street#2472), the CFMFN again makes a more accurate prediction. For some scenes with a great similarity between the foreground and the background (Fig. 5 fall#1527), the CFMFN can still make accurate predictions in terms of the ambiguity of segmentation.

V. DISCUSSION

A. ABLATION STUDY OF THE NETWORK STRUCTURE

In an ablation study, we make different design choices of the decoder to explore the effect of the decoder structure on the segmentation performance. There are several design choices of the decoder as shown in Fig. 7; some decoders directly decode the input of the decoder without introducing additional features (general methods), others introduce high-level features into the decoding process to obtain more global information, and others (such as deeplabv3+) introduce low-level features into the decoding process to obtain more detailed information. The high-level features and low-level features are effectively fused in the cascade design to generate fine masks. The experimental results are given in Tab. 4. It is seen that the introduction of additional features at different levels improves the segmentation performance, but the improvement from effectively fusing features at multiple levels is more pronounced. Fig. 8 shows that cascaded feature-mask fusion captures object details better in multi-level detail optimization and makes better use of features to generate finer segmentation than previous decoder designs.

B. ABLATION STUDY OF THE LOSS DESIGN

An experiment is conducted to show that learning both the global refinement (L_{CE}^{16}) and local refinement (L_{CE}^{4}) is essential; see Tab. 5. The FM value rises more when L^{16} and L^{1}

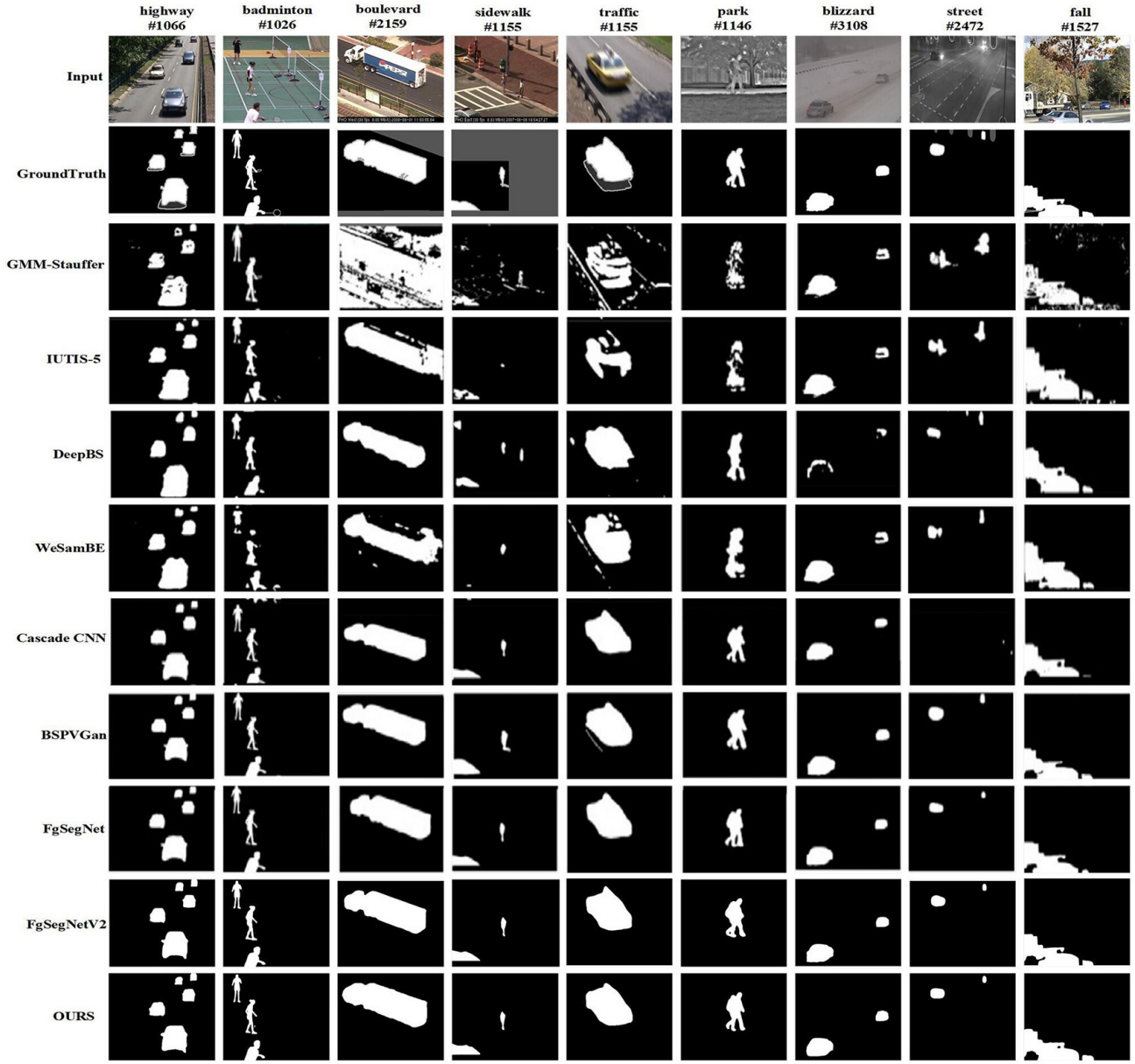


FIGURE 5. Qualitative comparison among different foreground segmentation methods.

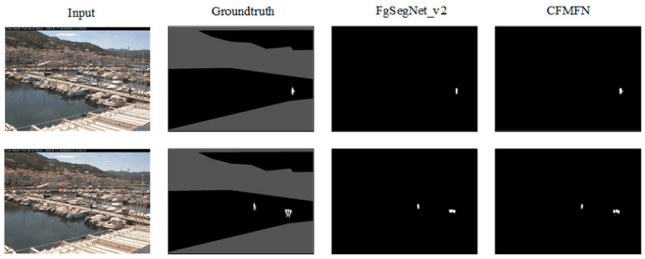


FIGURE 6. Our method segments video sequences in the lowFrameRate category more finely than FgSegNet_v2.

are chosen, corresponding to the ablation of the network structure. The low-level features have higher resolution and contain more location and detailed information. However, the

TABLE 4. Ablation study of the network structure.

Methods	Precision	Recall	PWC	F-measure
With OS_1 only	0.9436	0.9613	0.2587	0.9493
With OS_1 & OS_{16}	0.9581	0.9648	0.1764	0.9611
With OS_1 & OS_4	0.9796	0.9768	0.0842	0.9782
OS_1 & OS_4& OS_{16}	0.9845	0.9892	0.0599	0.9868

results show that single-layer feature learning alone leads to sub-optimal detection, and effectively combining high-level features and low-level features for learning obviously optimizes the performance.

C. ABLATION STUDY OF THE CHOICES OF LOSS

Simple ablation experiments on the choices of loss are conducted to demonstrate the effectiveness of the loss design

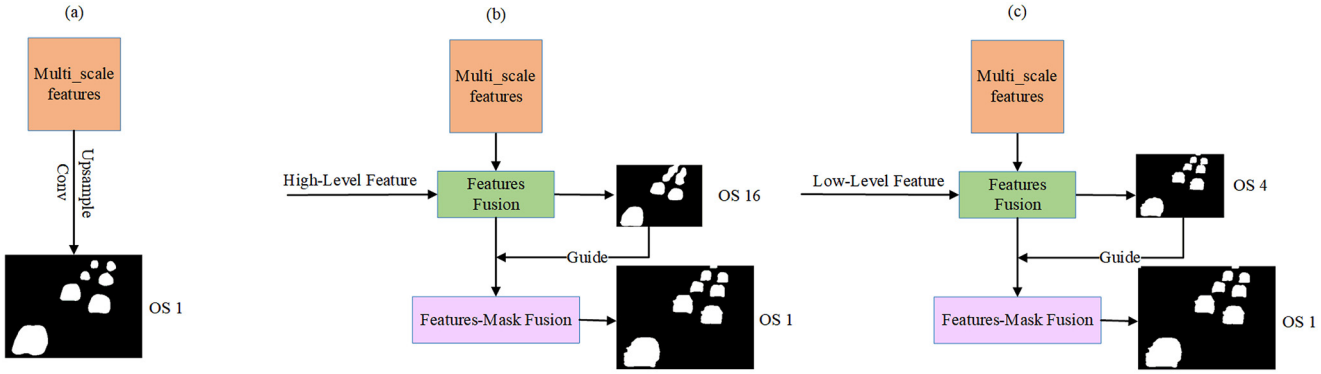


FIGURE 7. Different design choices of the decoder.

TABLE 5. Ablation study of the loss design.

Methods	Precision	Recall	PWC	F-measure
L^1 only	0.9440	0.9779	0.1370	0.9580
L^{16}_{CE} & L^1	0.9669	0.9773	0.1228	0.9683
L^{4}_{CE} & L^1	0.9785	0.9784	0.0901	0.9784
L^{16}_{CE}& L^{4}_{CE}& L^1	0.9845	0.9892	0.0599	0.9868

TABLE 6. Ablation study of the choices of loss.

Methods	Precision	Recall	PWC	F-measure
L^1 only	0.9614	0.9799	0.1186	0.9677
L_{CE} only	0.9541	0.9801	0.1043	0.9650
L_{CE} and L^1	0.9845	0.9892	0.0599	0.9868

TABLE 7. Ablation study of design choices of the decoder.

Methods	Precision	Recall	PWC	F-measure
Concat(F,features)	0.9621	0.9776	0.2542	0.9602
Concat(F,features)*mask	0.9592	0.9521	0.1071	0.9547
Concat(F,features,mask)	0.9845	0.9892	0.0599	0.9868

of the CFMFN. The choices of loss are that the loss functions of the three masks are all set to the Euclidean distance after L2 normalization, the loss functions of the three masks are all set to the cross-entropy loss, the loss functions of $Mask_{16}$ and $Mask_4$ are set to the cross-entropy loss, and the *Mask* is set to the Euclidean distance after L2 normalization. Tab. 6 shows the results of applying the different loss functions for different strides. In the rough segmentation of $Mask_{16}$ and $Mask_4$, more attention is paid to the global information, while the final *Mask* focuses on the details to achieve the accurate segmentation of the local boundaries, which indicates the effectiveness of the CFMFN choices of loss.

D. ABLATION STUDY OF THE DESIGN CHOICES OF THE DECODER

To evaluate the effectiveness of our proposed module, we compare feature-mask fusion with two other mask learning methods, namely multiplication and no-operation methods. Experimental results are given in Tab. 7. The fusion of the extracted features with the corresponding mask can focus on key features for refining boundary details.

Tab. 7 shows that feature-mask fusion obtains the best result among the different ways of combining masks and

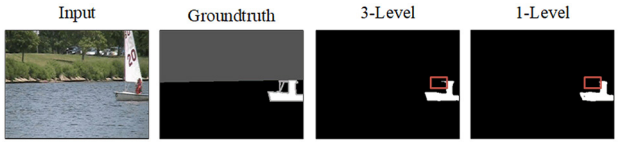


FIGURE 8. Difference between a 3-Level input model and a 1-Level input model.

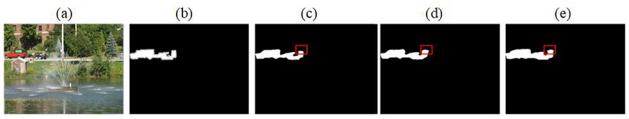


FIGURE 9. (*a*) input (*b*) Groundtruth (*c*) Concat(*F*,features,mask) (*d*) Concat(*F*, features) (*e*) Concat(*F*, features)∗mask.

features. Because the value range of the mask is 0 to 1, the repeated multiplication gradually reduces the eigenvalue, if no processing is done, more useless information is introduced and the segmentation performance is sub-optimal. However, features that concatenate the corresponding mask can better guide the fine segmentation, which not only retains the high-level information transmitted from the coarser mask but also keeps the fine local information provided by the lower-level mask. Additionally, Fig. 9 shows that mask concatenates features provide more precise results, which demonstrates the effectiveness of mask learning.

VI. CONCLUSION

This paper proposed the CFMFN as an encoder-decoder model that is capable of end-to-end training in a supervised manner. We improved by feature-mask fusion and adopted a cascade design to accurately segment moving objects from coarse to fine levels. Moreover, the CFMFN learns foreground objects from isolated frames, and fine foreground segmentation can be learned using a small number of frames. Experimental results show that the overall F-measure of the CFMFN on the CDnet2014 dataset is 0.9868, and the segmentation performance is superior to that of many existing methods. However, owing to the high computational cost of the multi-scale input, future research will aim to explore a cascaded multi-scale feature extraction network fused with

attention mechanisms to improve the performance of segmenting moving objects. In addition, We have considered proving the effectiveness of our approach in other areas as future research work.

ACKNOWLEDGMENT

The authors would like to thank CDnet2014 benchmark for making the segmentation masks of all methods publicly available.

REFERENCES

[1] 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.*, vol. 2, 1999, pp. 246–252.

[2] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Proc. 17th Int. Conf. Pattern Recognit. (ICPR)*, vol. 2, 2004, pp. 28–31.

[3] S. Zhu and L. Xia, "Human action recognition based on fusion features extraction of adaptive background subtraction and optical flow model," *Math. Problems Eng.*, vol. 2015, pp. 387–464, Sep. 2015.

[4] K. Wang, Y. Liu, C. Gou, and F.-Y. Wang, "A multi-view learning approach to foreground detection for traffic surveillance applications," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 4144–4158, Jun. 2016.

[5] Y. Liu, K. Wang, and D. Shen, "Visual tracking based on dynamic coupled conditional random field model," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 3, pp. 822–833, Mar. 2016.

[6] C. Gou, K. Wang, B. Li, and F.-Y. Wang, "Vehicle license plate recognition based on class-specific ERs and SaE-ELM," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, 2014, pp. 2956–2961.

[7] W. Li, V. Mahadevan, and N. Vasconcelos, "Anomaly detection and localization in crowded scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 18–32, Jan. 2014.

[8] 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*, 2014, pp. 387–394.

[9] O. Barnich and M. Van Droogenbroeck, "ViBe: A universal background subtraction algorithm for video sequences," *IEEE Trans. Image Process.*, vol. 20, pp. 1709–1724, 2010.

[10] 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, pp. 359–373, 2015.

[11] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, "A self-adjusting approach to change detection based on background word consensus," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2015, pp. 990–997.

[12] N. Vaswani, T. Bouwmans, S. Javed, and P. Narayanamurthy, "Robust subspace learning: Robust PCA, robust subspace tracking, and robust subspace recovery," *IEEE signal Process. Mag.*, vol. 35, no. 4, pp. 32–55, Jul. 2018.

[13] P. Narayanamurthy and N. Vaswani, "A fast and memory-efficient algorithm for robust PCA (MEROP)," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2018, pp. 4684–4688.

[14] T. Bouwmans, S. Javed, M. Sultana, and S. K. Jung, "Deep neural network concepts for background subtraction:A systematic review and comparative evaluation," *Neural Netw.*, vol. 117, pp. 8–66, Sep. 2019.

[15] B. Garcia-Garcia, T. Bouwmans, and A. J. R. Silva, "Background subtraction in real applications: Challenges, current models and future directions," *Comput. Sci. Rev.*, vol. 35, Feb. 2020, Art. no. 100204.

[16] M. Mandal and S. K. Vipparthi, "An empirical review of deep learning frameworks for change detection: Model design, experimental frameworks, challenges and research needs," *IEEE Trans. Intell. Transp. Syst.*, early access, May 19, 2021, doi: 10.1109/TITS.2021.3077883.

[17] W. Zheng, K. Wang, and F.-Y. Wang, "A novel background subtraction algorithm based on parallel vision and Bayesian GANs," *Neurocomputing*, vol. 394, pp. 178–200, Jun. 2020.

[18] S. Murala and P. Patil, "FgGAN: A cascaded unpaired learning for background estimation and foreground segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, 2021, pp. 1770–1778.

[19] W. Zheng, K. Wang, and F. Wang, "Background subtraction algorithm based on Bayesian generative adversarial networks," *Acta Automatica Sinica*, vol. 44, no. 5, pp. 878–890, 2018.

[20] J.-Y. Kim and J.-E. Ha, "Foreground objects detection using a fully convolutional network with a background model image and multiple original images," *IEEE Access*, vol. 8, pp. 159864–159878, 2020.

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

[22] L. P. Cinelli, L. A. Thomaz, A. F. da Silva, E. A. da Silva, and S. L. Netto, "Foreground segmentation for anomaly detection in surveillance videos using deep residual networks," in *Proc. 35th Simpósio Brasileiro De Telecomunicações E Processamento De Sinais*, Sao Pedro, Brazil, 2017, pp. 3–6.

[23] K. Lim, W.-D. Jang, and C.-S. Kim, "Background subtraction using encoder-decoder structured convolutional neural network," in *Proc. 14th IEEE Int. Conf. Adv. Video Signal Based Surveillance (AVSS)*, 2017, pp. 1–6.

[24] 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.

[25] L. A. Lim and H. Y. Keles, "Foreground segmentation using convolutional neural networks for multiscale feature encoding," *Pattern Recognit. Lett.*, vol. 112, pp. 256–262, Sep. 2018.

[26] L. A. Lim and H. Y. Keles, "Learning multi-scale features for foreground segmentation," *Pattern Anal. Appl.*, vol. 23, no. 3, pp. 1369–1380, 2020.

[27] 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)*, 2018, pp. 1670–1675.

[28] D. Sakkos, H. Liu, J. Han, and L. Shao, "End-to-end video background subtraction with 3D convolutional neural networks," *Multimedia Tools Appl.*, vol. 77, no. 17, pp. 23023–23041, 2018.

[29] T. Akilan, "Video foreground localization from traditional methods to deep learning," Ph.D. dissertation, Dept. Electr. Comput. Eng., Univ. Windsor, Windsor, ON, Canada, 2018.

[30] Z. Hu, T. Turki, N. Phan, and J. T. L. Wang, "A 3D atrous convolutional long short-term memory network for background subtraction," *IEEE Access*, vol. 6, pp. 43450–43459, 2018.

[31] 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.

[32] M. Mandal, V. Dhar, A. Mishra, S. K. Vipparthi, and M. Abdel-Mottaleb, "3DCD: Scene independent end-to-end spatiotemporal feature learning framework for change detection in unseen videos," *IEEE Trans. Image Process.*, vol. 30, pp. 546–558, 2020.

[33] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 483–499.

[34] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2117–2125.

[35] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1925–1934.

[36] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 833–851.

[37] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, 2015, pp. 234–241.

[38] J. He, S. Zhang, M. Yang, Y. Shan, and T. Huang, "Bi-directional cascade network for perceptual edge detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3828–3837.

[39] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.

[40] H. K. Cheng, J. Chung, Y.-W. Tai, and C.-K. Tang, "CascadePSP: Toward class-agnostic and very high-resolution segmentation via global and local refinement," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8890–8899.

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

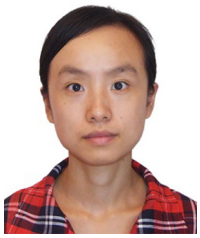
[42] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[44] A. Paszke *et al.*, "Automatic differentiation in PyTorch," in *Proc. NIPS Autodiff Workshop*, 2017.
[45] S. Bianco, G. Ciocca, and R. Schettini, "How far can you get by combining change detection algorithms?" in *Proc. Int. Conf. Image Anal. Process.*, 2017, pp. 96–107.
[46] S. Jiang and X. Lu, "WeSamBE: A weight-sample-based method for background subtraction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 9, pp. 2105–2115, Sep. 2018.
[47] M. Babaee, D. T. Dinh, and G. Rigoll, "A deep convolutional neural network for background subtraction," 2017, *arXiv:1702.01731*.

CHUANYUN XU received the M.S. degree in software engineering and the Ph.D. degree in computer science from Chongqing University in 2006 and 2014, respectively. He is currently an Associate Professor with the School of Artificial Intelligence, Chongqing University of Technology and the College of Computer and Information Science, Chongqing Normal University. He worked as a Project Scientist with the University of California at Riverside from 2016 to 2018. His research interests include artificial intelligence, machine learning, and image processing.

HUAN LIU received the bachelor's degree from the Harbin University of Commerce in 2019. She is currently pursuing the graduate degree with the School of Artificial Intelligence, Chongqing University of Technology. Her research interests mainly focus on computer vision and machine learning.

TENGHUI LI received the B.Sc. degree in computer science from the Southwest University of Science and Technology in 2019. He is currently pursuing the M.Sc. degree with the School of Artificial Intelligence, Chongqing University of Technology. His masters focused on research in computer vision which included the image segmentation, image classification, and moving object detection.



YANG ZHANG received the Ph.D. degree in computer science from Chongqing University. She is currently an Associate Professor with the College of Computer and Information Science, Chongqing Normal University. Her research interests include software measurement, services computing, and trusted computing.

TIAN LI is currently pursuing the postgraduate degree in computer science with RWTH Aachen. He is passionate about artificial intelligence and machine learning, with strong technical, business and interpersonal skills for working in a team and successfully completing several projects. He currently focuses on theory and practice of data science in process mining and data mining.

GANG LI received the Ph.D. degree in computer science from Chongqing University. He is currently a Professor with the School of Artificial Intelligence, Chongqing University of Technology. His research interests include pattern recognition, image processing, and computer vision. He is a member of China Computer Federation and Chinese Association for Artificial Intelligence.