Computer Science and Information Systems 21(1):363–378 https://doi.org/10.2298/CSIS230129044T

3D Convolutional Long Short-Term Encoder-Decoder Network for Moving Object Segmentation *

Anil Turker[1] and Ender Mete Eksioglu[2]

[1] ASELSAN Inc.
Ankara, Turkey
anilturker17@gmail.com
[2] Electronics and Communication Engineering Department,
Istanbul Technical University, Istanbul, Turkey
eksioglue@itu.edu.tr

Abstract. Moving object segmentation (MOS) is one of the important and well-studied computer vision tasks that is used in a variety of applications, such as video surveillance systems, human tracking, self-driving cars, and video compression. While traditional approaches to MOS rely on hand-crafted features or background modeling, deep learning methods using Convolution Neural Networks (CNNs) have been shown to be more effective in extracting features and achieving better accuracy. However, most deep learning-based methods for MOS offer scene-dependent solutions, leading to reduced performance when tested on previously unseen video content. Because spatial features are insufficient to represent the motion information, the spatial and temporal features should be used together to succeed in unseen videos. To address this issue, we propose the MOS-Net deep framework, an encoder-decoder network that combines spatial and temporal features using the flux tensor algorithm, 3D CNNs, and ConvLSTM in its different variants. MOS-Net 2.0 is an enhanced version of the base MOS-Net structure, where additional ConvLSTM modules are added to 3D CNNs for extracting long-term spatiotemporal features. In the final stage of the framework the output of the encoder-decoder network, the foreground probability map, is thresholded for producing a binary mask where moving objects are in the foreground and the rest forms the background. In addition, an ablation study has been conducted to evaluate different combinations as inputs to the proposed network, using the ChangeDetection2014 (CDnet2014) which includes challenging videos such as those with dynamic backgrounds, bad weather, and illumination changes. In most approaches, the training and test strategy are not announced, making it difficult to compare the algorithm results. In addition, the proposed method can be evaluated differently as video-optimized or video-agnostic. In video-optimized approaches, the training and test set is obtained randomly and separated from the overall dataset. The results of the proposed method are compared with competitive methods from the literature using the same evaluation strategy. It has been observed that the introduced MOS networks give highly competitive results on the CDnet2014 dataset. The source code for the simulations provided in this work is available online.

Keywords: Moving object segmentation, flux tensor, deep learning, spatiotemporal, change detection, foreground segmentation, background subtraction.

*A preliminary version of this work has been published as a conference paper in INISTA 2022.

1. Introduction

Moving object detection and segmentation is a vital task in many systems, such as video surveillance, human tracking, action recognition, self-driving cars, and video compression. It is particularly important in video surveillance systems and is also used as a subsystem in applications like video compression and self-driving cars. Moving object segmentation, also known as background subtraction, involves modeling the background and then comparing it to the current frame in order to classify each pixel as foreground or background. The accuracy of the algorithm depends heavily on the background modeling and maintenance process. A background model is created using the initial background and updated using upcoming frames. There are different strategies used for updating the process of background. The update rate is one of the significant hyper-parameters of the algorithm that affect the accuracy.

Moving object segmentation (MOS) is considered a semantic segmentation problem, each pixel is classified as foreground or background. The encoder-decoder network is popularly used in semantic segmentation problems. The encoder part extracts features with convolutional filters and pooling operations, the decoder part uses high-level features to produce a binary map. In order to increase the performance of the decoder network, there is a skip connection for transferring low-level features to decoder layers that gives better localization results. However, spatial features alone are not sufficient for moving object detection or segmentation, so it is necessary to consider both spatial and temporal features.

Our first proposed network, called Moving Object Segmentation (MOS) Net, is designed for binary segmentation. It is based on the U-Net [1] network, which has been successful in many segmentation tasks and is frequently used in the literature. In this paper, we preferred to use a hybrid algorithm that contains a flux tensor and 3D CNN. Flux tensor method [2] is an efficient way to extract motion information without using eigenvalue decomposition. Also, the network contains a 3D CNN for extracting the spatiotemporal features by using temporal depth in the kernel. Combining the output of flux tensor and motion entropy maps extracted by convolutional filters makes the network more robust for unseen videos. The flux tensor output, the feature map extracted by the 3D CNN network, and the current frame are the inputs of our U-net model, and networks produce a foreground probability map as a result of the sigmoid activation in the last layer of the network. MOS-Net in its basic form was proposed in preliminary work [3].

This paper represents an extended version of the conference paper [3]. In this paper, we introduce the novel MOS-Net 2.0 network, which is an enhanced version of the vanilla MOS-Net. In MOS-Net 2.0, ConvLSTM modules are appended to 3D CNNs with the aim of extracting long-term spatiotemporal features. MOS-Net 2.0 greatly benefits from the utilization of these long-term spatiotemporal features. Simulation results indicate improved performance for the novel MOS-Net 2.0, especially in scenes that include dynamic background objects such as the shaking of moving tree leaves, snow, rain, waves, etc.

The rest of this paper is organized as follows. Section 2 presents a literature review that contains traditional and deep learning-based methods for moving object segmentation. Section 3 and Section 4 describe the architecture of our proposed encoder-decoder network variants which are called as MOS-Net and MOS-Net 2.0, respectively. Section 5 conducts a quantitative and qualitative comparison of our networks and the competing methods from the literature and also includes an ablation study. Section 6 provides the conclusions and a final discussion of the obtained results.

2. Related Work

Traditional methods for detecting a moving object are also called background subtraction. The reason for this naming is that background modeled by the algorithm is compared with the current frame, and then each pixel is classified as a result of this comparison. We can examine the background subtraction algorithm in 3 parts: background initialization, background modeling and maintenance, and foreground segmentation. In the background initialization part, an initial background is initialized using historical video frames. The initialized background is a reference for segmentation in currently incoming video frames. An initial model can be created using the statistical properties of N video frames taken from the beginning of the video. Background modeling and its maintenance are the most critical part of the performance of the background subtraction algorithm. A background model is created using the initial background and updated using upcoming frames. Foreground segmentation is performed by comparing the background model and the current frame. There exist different comparison strategies. For instance, the absolute value of the difference between the current frame and the background model can be compared with a threshold value. If this difference is higher than the threshold value, the pixel is classified as foreground, otherwise as background.

There are many efficient and influential studies on moving object detection and segmentation in the literature. One important method is the Gaussian Mixture Model (GMM) [4], a parametric approach proposed by C. Stauffer et al. to model complex backgrounds such as moving background and brightness change. GMM is the weighted sum of N Gaussian distributions. In this method, the relationship between neighboring pixels is disregarded, and each pixel is evaluated independently. The mean value of the weighted Gaussian distribution is computed for each pixel and compared with the corresponding pixel value in the incoming frame. If a match is found, the pixel is classified as part of the background. GMM can tolerate the brightness changes because of the nature of Gaussian distribution and permit the different appearance of background objects with another Gaussian distribution for the pixel. ViBe [5], introduced by Barnich et al., is a method that models the background with a sample-based system. The last N values are kept in the library for each pixel. The classification process was carried out by comparing the pixels in this created library with newly arriving pixels. At least K matches are expected to classify a new pixel as background. The difference between the gray level values of the pixels has to be lower than a specified threshold value R to allow a match. Oliver et al [6] developed a learning-based algorithm in which they use subspace learning for modeling background, a method called as eigenbackground. The model first calculates the mean background image and covariance matrix from N sample images. Then the background is modeled using Principal Component Analysis (PCA) with the largest eigenvalues and corresponding M eigenvectors. After the new frames are projected on the eigenvectors, the picture and the projection are compared. Then the pixels are classified with a threshold value used in this comparison.

CNNs have been used by researchers to detect or segment moving objects, in addition to their extensive use in other fields of computer vision. Braham and Broeck proposed an early method that uses convolutional filters for the background subtraction [7]. In their work, the background and current images are obtained by using the temporal median filter, and then they are fed into LeNet-5 [8] network in the form of *N*N* patches. This method is significant for inspiring other researchers about the utilization of deep learning methods in this research area. Another popular method is DeepBs [9] which combines traditional and deep learning-based approaches and makes them more robust on unseen videos. It uses the Subsense algorithm [10] as a traditional approach. Then, the current frame and modeled background are given as input to the network in the form of *N*N* patches. There is also a pixel library that contains the historical values of each pixel. The length of the library changes depends on the motion information generated by the flux tensor algorithm [2]. In contrast to DeepBs, Gao and Cai proposed an end-to-end network [11] that uses 3D convolution to extract both spatial and temporal features. FgSegNet [12] network has state-of-the-art results on the CDNet-2014 dataset. It is a segmentation network consisting of an encoder and decoder network that extracts multi-scale features. It uses 50 or 200 frames for each video in training the network. This algorithm provides a video-specific result for foreground segmentation.

Another recent work is the BSUV-Net network [13], which consists of a fully convolutional network and is tested on unseen videos. BSUV-Net is a network that has a similar network structure to U-net [1]. The network uses the current frame and two temporal median filters with different numbers for the historical frames. In addition to the three channels, they exploit the segmentation network in their proposed method. In most of the approaches, the training and test strategy are not announced, which makes it difficult to compare the algorithm results. In addition, the proposed method can be evaluated differently as video-optimized or video-agnostic. In video-optimized approaches, the training and test set is obtained from randomly separated video frames. Even if the datasets are selected from different video frames, they may contain similar images. Therefore, the network's performance on unseen videos is not sufficient. The training and test videos are entirely different in the video-agnostic methods, so the network test on unseen videos. Consequently, video-optimized approaches have an unfair advantage over video-agnostic techniques.

3. Proposed Method I - MOS-Net

This section gives details about our proposed network MOS-Net [3] consisting of 3D CNN, the flux tensor algorithm, and the encoder-decoder network. A preliminary version of this work including the MOS-Net was provided in [3]. The proposed network is shown in Figure 1. In addition, we give the details of our training strategy and loss function.

3.1. 3D CNNs

Motion features are extracted through 3D CNNs using consecutive frames. The last *N* frames go into the network, the temporal dimension is reduced in each convolution layer, and the motion information is extracted as a result of the network. The temporal stride parameter is used as 5, 5, and 2 in each convolution layer, respectively. The different kernel sizes $5{\times}5$, 3×3, and 1×1 are used for extracting multi-scale features, and these features maps are added up. We used 50 last recent frames as input frames in our proposed 3D CNNs, and its architecture is shown in Figure 1a.

3.2. The flux tensor algorithm

There are different algorithms used to obtain motion information in the literature. One of the most well-known of these is the Lucas Kanade method [14] for the estimation of optical flow. This algorithm makes the optical flow estimation by assuming that the motion in neighboring pixels is similar. The least-square fit method is used for estimating the motion vectors.

The flux tensor algorithm [2] is an extended version of the 3D grayscale structure tensor [15] which has coherent motion segmentation results with respect to classical optical flow methods. In the flux tensor algorithm [2], motion information can be extracted with a lower computational cost compared to other optical flow algorithms without using the eigenvalue decomposition. The local 3D spatiotemporal volume contains the optical flow field, and the flux tensor describes the changes in the field over time. The matrix of flux tensor is shown in Eq. (1). It is possible to distinguish between static and moving objects by using the flux tensor matrix. The trace of the matrix can be used directly to find the moving object in the current frame. The trace of the matrix is shown in Eq. (2)

$$\mathbf{J} = \begin{bmatrix} \int_{\Omega}(\frac{d^2I}{dxdt})^2 dy & \int_{\Omega}\frac{d^2I}{dxdt}\frac{d^2I}{dydt}dy & \int_{\Omega}\frac{d^2I}{dxdt}\frac{d^2I}{dt^2}dy \\ \int_{\Omega}\frac{d^2I}{dydt}\frac{d^2I}{dxdt}dy & \int_{\Omega}(\frac{d^2I}{dydt})^2 dy & \int_{\Omega}\frac{d^2I}{dydt}\frac{d^2I}{dt^2}dy \\ \int_{\Omega}\frac{d^2I}{dt^2}\frac{d^2I}{dxdt}dy & \int_{\Omega}\frac{d^2I}{dt^2}\frac{d^2I}{dydt}dy & \int_{\Omega}(\frac{d^2I}{dt^2})^2 dy \end{bmatrix} \tag{1}$$

$$\mathbf{Trace}_J = \int_{\Omega)} \left\| \frac{d}{dt}(\nabla I) \right\|^2 dy \tag{2}$$

3.3. Encoder-Decoder network

This study uses a U-Net type encoder-decoder network for foreground segmentation. BSUV-Net [13], one of the recent methods, uses the U-net type neural network for the segmentation process. It also utilizes reference frames obtained from the temporal median filter at different intervals. The current frame and reference frames are given as input to a U-net network. In our proposed network, unlike BSUV-Net, the motion information is extracted by using 3D CNNs and flux tensor algorithm. The extracted motion feature maps are given as input to the encoder-decoder network, together with the current frame. Our proposed encoder-decoder network for MOS-Net is shown in Figure 1b.

The encoder module reduces the spatial resolution by using convolution and max pooling operations. The decoder module upsamples the reduced form to the original resolution. The decoder part maps the low-resolution image containing the features to the original resolution and utilizes the skip connection in the network. The first four convolutional layers of the encoder module are the same as the VGG-16 [16] network. All convolution filters have a 3×3 kernel size and max-pooling has a 2×2 kernel size as a stride rate of 2. Batch normalization [17] is used at the end of each convolution layer and standardizes the input for each mini-batch. It stabilizes the network, speeds up the training process, and has a regularization effect.

Motion information obtained by using a flux tensor and 3D convolutional neural network is input to the encoder-decoder network together with the current frame. The network gives an output of a binary image with the same spatial resolution as the current

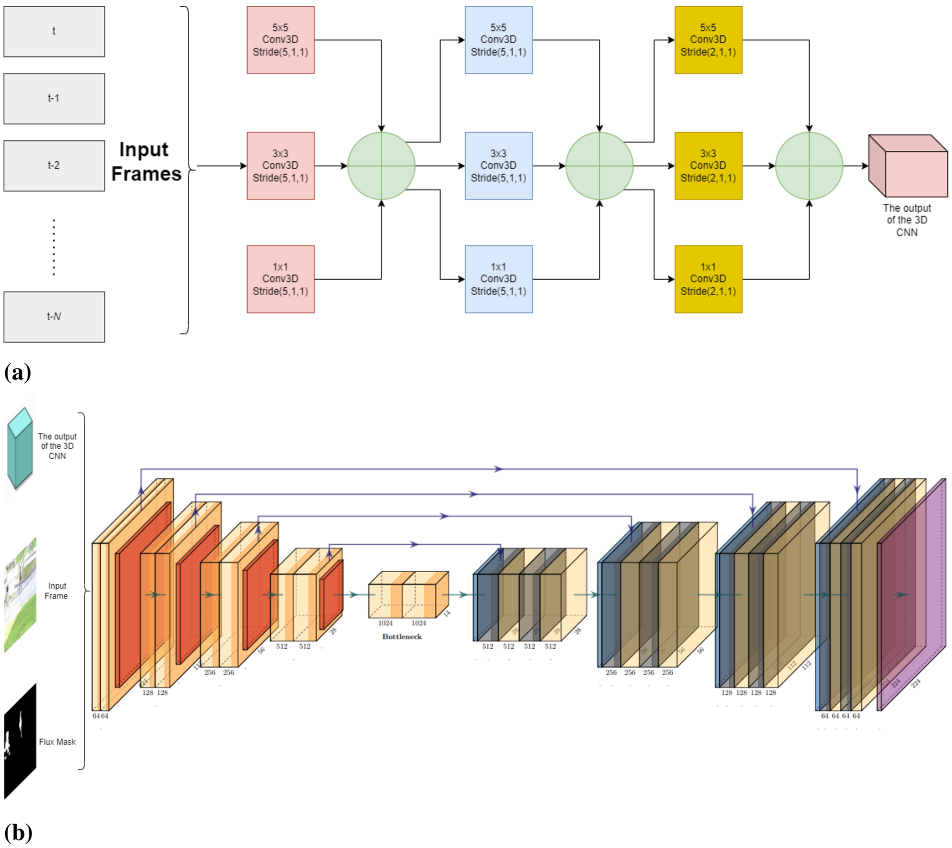


Fig. 1. Proposed foreground segmentation network, MOS-Net. (a) Initial 3D Convolutional Neural Network extracting the motion entropy maps from N historical frames, (b) Encoder-decoder network creating the final segmentation map output

frame. The binary image that contains the foreground probability map is obtained by using the sigmoid activation function. The foreground probability map consists of pixels value between 0 and 1, and pixels are threshold by a value; ones higher than the threshold value are classified as foreground, and others are the background.

3.4. Loss function

Moving object segmentation is fundamentally the classification of each pixel as foreground or background. The number of background pixels on the dataset is much more than the number of foreground pixels. This imbalance problem degrades the performance of the proposed fully convolutional network. The cross-entropy loss function is prevalent in semantic segmentation tasks. When this loss function is used in an imbalanced dataset, the classifier favors the majority classes, and the network will be a biased model. The weighted cross-entropy loss is proposed as a solution to this imbalance problem. The effect of the samples with a minority in the dataset on the loss function gets increased in the weighted cross-entropy loss. The Jaccard index or IoU (Intersection over Union) is used widely for imbalanced dataset problems in object detection tasks. In addition to object detection, this loss function can also be used for semantic segmentation tasks. The Jaccard index is computed as follows.

$$\textit{Jaccard Index} = \frac{TP}{TP + FP + FN} \tag{3}$$

Here, the below given definitions are being used.

TP: the count of true positives
FP: the count of false positives
FN: the count of false negatives

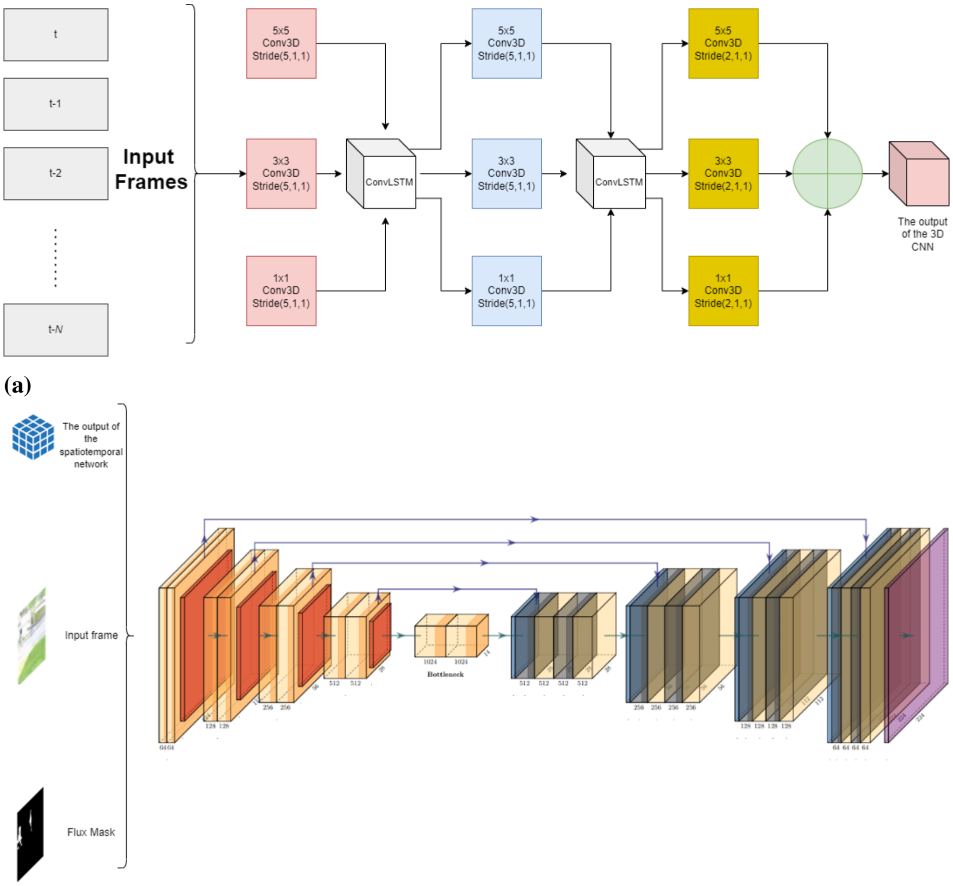
The Jaccard index measures the sensitivity of foreground pixel segmentation results. As the performance of the segmentation network improves, this value approaches one. On the other hand, as the performance decreases, the index approaches zero. The Jaccard index can be utilized to define a loss function with the following equation.

$$\textit{Jaccard Loss Function} = (1 - Jaccard\,Index) * \alpha \tag{4}$$

Here, α is a smoothing parameter.

4. Proposed Method II - MOS-Net 2.0

In the MOS-Net network introduced in Section 3, short-time spatiotemporal features are extracted from consecutive frames via a 3D Convolutional Neural Network. As stated in Chapter 2, the ConvLSTM time series can be used to extract long-term spatiotemporal features from an input. In order to increase the performance of the 3D CNN network used in this study, we add ConvLSTM modules to the 3D CNN spatiotemporal feature extractor. THis lead to the improved MOS-Net 2.0 network. The full network structure for MOS-Net 2.0 is shown in Figure 2. The encoder-decoder network responsible for



(a)

(b)

Fig. 2. Proposed foreground segmentation network, MOS-Net 2.0. a) Initial 3D CNN network with novel ConvLSTM modules generating the spatiotemporal features, b) Encoder-decoder network creating the final segmentation map

feature extraction and dense prediction, the flux tensor algorithm, and the utilized loss function are the same as in MOS-Net.

LSTM is one of the RNN methods used in time series problems. However, considering the large number of pixels in an image, too many parameters are required for an LSTM realization when imported to an image processing setting. At the same time, LSTM is insufficient to extract spatial features. Therefore, ConvLSTM is preferred where the matrix multiplication is exchanged with a convolution operation using a 2D kernel. ConvLSTM can extract spatiotemporal features from the 2D time series image, using both the convolutional layer and also the cell state included in LSTM. 3D CNNs transmit the necessary information for motion information to the encoder-decoder network. In the MOS-Net 2.0, the ConvLSTM module is applied to an element-wise collection of feature maps obtained with different sized filter in 3D convolutions. The updated architecture for the spatiotemporal feature extractor is shown in Figure 2a. The use of ConvLSTM is intended to improve the robustness of the network in extracting motion information.

5. Experiments and Performance Evaluation

The proposed method is trained and tested using the ChangeDetection2014 (CDnet2014) dataset [18], which is frequently used in moving object segmentation tasks. ChangeDetection is often employed in assessing the performance of moving object segmentation and change detection algorithms. CDnet2014 was announced as an extended version of the 2012 CDnet dataset, with five new categories and 22 videos (Figure 3).

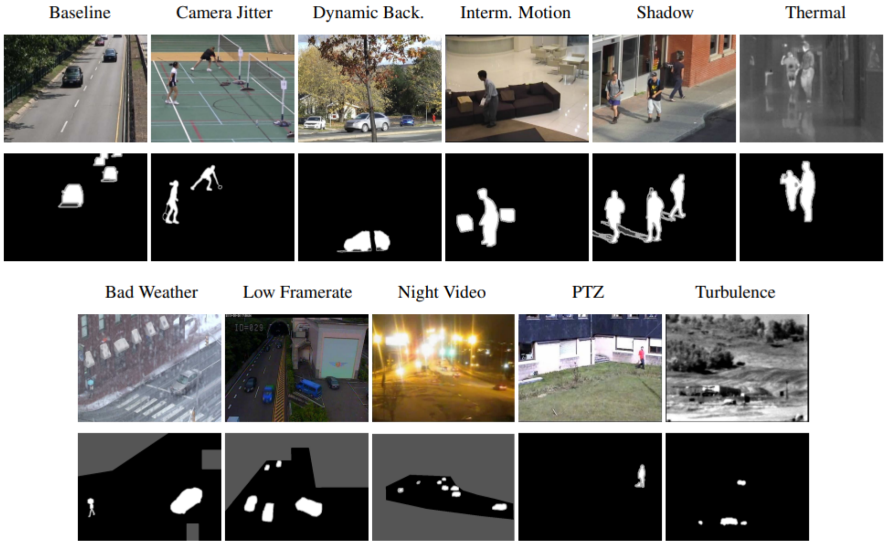


Fig. 3. The sample video frames from CDnet2014 dataset [18]

We implemented proposed methods using the Pytorch framework and a Tesla P100-PCIE-16GB GPU, utilizing a batch size of 8. Adam was used as the optimization algorithm during the training, and the learning rate started at 0.0001. Afterwards the learning rate was gradually reduced every 20 epochs. The maximum number of epochs is 60 for each network configuration. The networks are trained by randomly selecting 200 video frames for each video listed in Table 1.

Fully convolutional networks are independent of the particular spatial resolution of the input. The spatial resolutions of the videos in the CD2014 dataset differ from each other. We used the fixed size input as *224x224* to utilize the parallel processing power of GPUs. The inputs can be made same size by resizing operations or cropping fixed-sized patches from the images. We used the randomly cropping strategy for the fixed-size input in the training process, since randomly cropped images give the network an extra augmentation and regularization effect. In addition, random noise has been added to the input image as another augmentation technique that makes the network robust to sudden changes. On the other hand, we used the original spatial resolution of the input frame without any scaling or cropping operation in the inference process.

Table 1. Change Detection 2014 Dataset Data Division [19]

Category	Train Data	Test Data
Baseline	highway, office, PETS2006	pedestrians
badWeather	skating, snowFall, wetSnow	blizzard
cameraJitter	badminton, boulevard, sidewalk	traffic
dynamicBackground	canoe, fall, fountain01, fountain02, overpass	boats
intermittentObjectMotion	abandonedBox, sofa, streetLight, tramstop, winterDriveway	parking
lowFramerate	port_0_17fps, tramCrossroad_1fps, tunnelExit_0_35fps	turnpike_00_05fps
nightVideos	bridgeEntry, busyBoulvard, fluidHighway, streetCornerAtNight, winterStreet	tramStation
shadow	backdoor, bungalows, cubicle, peopleInShade	busStation
thermal	diningRoom, lakeSide, library, park	corridor
turbulence	turbulence1, turbulence2, turbulence3	turbulence0

As a result of the encoder-decoder network, a foreground probability map (FPM) with values between 0 and 1 is obtained. Binary images can be produced using different threshold values such as 0.1, 0.3, 0.5, 0.7, and 0.9. Figure 4 shows the average F1 score of MOS-Net and MOS-Net 2.0 networks for these threshold values on the unseen videos in Table 1. It indicates that MOS-Net and MOS-Net 2.0 reach the best average F1 score with a 0.1 threshold value. Using these results, the fixed threshold value was determined as 0.1, and this threshold value was used in the experiments.

The network gives a foreground probability map resulting from the last sigmoid activation function. Then, pixel values greater than 0.1 classify as foreground and others as background. Our proposed methods, MOS-Net and MOS-Net 2.0 outperform competing methods as listed in Table 2. They have F1-scores 0.83 and 0.85 on the CDnet2014 dataset, respectively. As shown in Table 2, FgSegNet [12], a state of the art method for CDnet2014 dataset, has a performance of 0.22 in terms of F1 score. It has degraded performance on unseen videos, because FgSegNet [12] offers a video-specific solution. Another recent work is BSUV-Net [13]. Its performance is 0.80 in terms of F1-score on the unseen videos, as listed in Table 1. BSUV-Net [13] utilizes temporal median filters, which suppresses the performance of the network for dynamic scenes. As can be seen from the results, our proposed network MOS-Net 2.0 gives the best results among the competing algorithms. In Table 3, visual results obtained in different categories of the CDnet2014 dataset are given.

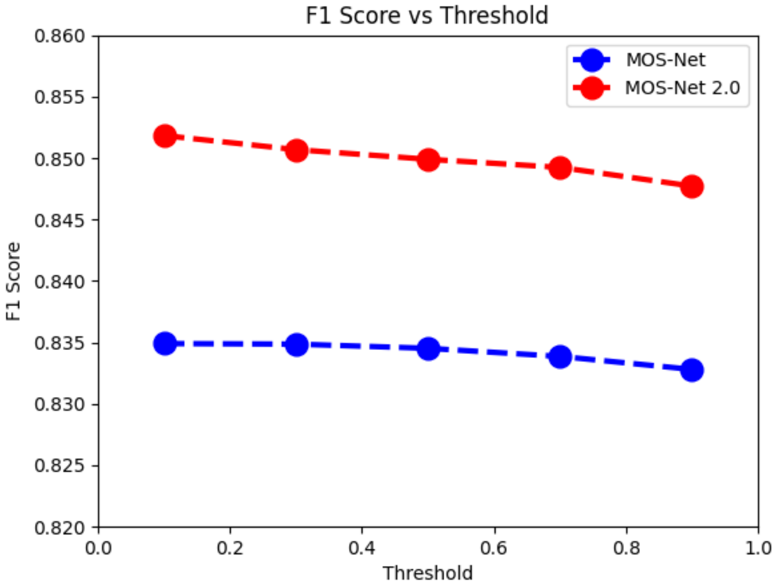


Fig. 4. The illustration of average F1 score on unseen videos versus thresholds

The comparison of the network parameter numbers of the proposed methods with FgSegNet and BSUV-Net, which are studies conducted in recent years, are given in Table 4. In the FgSegNet method, the parameters of the first three layers in the encoder block are frozen in the training process. These weights are the original weights of the VGG16 network. It is seen that FgSegnet has the lowest number of parameters since it uses convolutional filters and pooling operations for extracting only spatial features on the current frame. It has been observed that the performance on the unseen dataset is low due to the absence of spatiotemporal features. As seen in Table 4, the MOS-Net 2.0 with the highest F1 score also has the highest number of parameters. MOS-Net 2.0 has more parameters than MOS-Net, because of the inclusion of the ConvLSTM module.

One important contribution of this work is the evaluation of the effects of traditional and deep learning based features on the moving object segmentation performance. Our proposed networks, MOS-Net and MOS-NET 2.0 have multiple inputs at the encoder module, which extracts further spatiotemporal features using these inputs. We retrained the encoder-decoder network by trying out different combinations of these inputs as an ablation study. We investigate the impact of the choice of the inputs on the segmentation performance in previously unseen videos with regards to the recall, precision, and F1

Table 2. Comparison of method in terms of F-Measure on Change Detection 2014 Dataset

Method	Baseline	Bad weather	Camera Jitter	Dynamic Background	Int.Obj. motion	Low Framerate	Night	Shadow	Thermal	Turbulence	Overall
ViBe [5]	0.90	0.53	0.66	0.22	0.26	0.60	0.62	0.67	0.75	0.58	0.58
SubSense [10]	0.92	0.81	0.80	0.69	0.48	,0.81	0.76	0.82	0.86	0.79	0.77
PAWCS [20]	0.93	0.66	0.83	**0.88**	0.21	0.91	0.63	0.86	0.65	0.68	0.73
IUTIS-5 [21]	0.96	0.80	0.83	0.75	0.65	0.85	0.75	0.82	0.88	0.63	0.79
DeepBS [9]	0.95	0.61	**0.88**	0.81	0.60	0.49	0.16	**0.94**	0.89	0.77	0.71
FgSegNet [12]	0.06	0.12	0.36	0.12	0.46	0.60	0.27	0.27	0.18	0.02	0.22
BSUV-Net [13]	0.97	0.88	0.76	0.77	0.59	**0.96**	**0.82**	0.92	0.87	0.41	0.80
MOS-Net(ours)	**0.97**	0.85	0.73	0.57	**0.76**	0.91	0.81	0.81	**0.92**	0.92	0.83
MOS-Net 2.0(ours)	0.95	**0.89**	0.76	0.78	0.75	0.89	0.80	0.85	0.90	**0.94**	**0.85**

score. The results of this ablation study are shown in Table 5, where the training parameters are the same for all combinations. Current Fr. is the current frame of the video, and Flux Tensor refers to the traditional solution, the flux tensor algorithm. 3D CNN refers to the 3D CNN network used in the MOS-Net architecture. On the other hand, 3D CNN + ConvLSTM indicates the spatiotemporal feature extractor in the MOS-Net 2.0 variant, which utilizes a combination of 3D CNNs and ConvLSTM.

As seen in Table 5, the current frame gives poor results when it is used alone at the input of the encoder-decoder network. Because spatial features are insufficient for moving object segmentation, temporal information must also be given to the classifier. When the current frame and flux tensor algorithm are used together, it is seen that there is a severe increase in the F1 score. The motion information obtained from the flux tensor algorithm significantly increased the performance of the segmentation network. When the 3D CNNs network output is added as an additional input to this network, an average F1 score of 0.83 is obtained. With this result, the impact of 3D CNN features on segmentation performance has been validated versus the combination of current frame and flux tensor algorithm. After 3D CNN features are added to the input of the segmentation network in addition to current frame and flux tensor, there is a massive increase of 0.29 in the F1 score.

MOS-Net 2.0 is an improved version of MOS-Net. ConvLSTM is added to the spatiotemporal feature extractor network consisting of 3D CNNs for extracting long-term spatiotemporal features. As can be seen in Table 5, the ConvLstm model brings a slight performance increase over MOS-Net. In order to show the effect of motion information formed as a result of the flux tensor algorithm, the flux tensor algorithm was removed from the inputs of the MOS-Net and MOS-Net 2.0 studies, and training was conducted. It has been observed that there is a decrease in the F1 score obtained on the test videos when the flux tensor algorithm is not present. It has been demonstrated that extracting the motion information with the flux tensor algorithm is an important input for the deep network's segmentation success on unseen videos.

6. Conclusions

In recent years, there has been a growing interest in developing methods for motion analysis and object segmentation in video sequences. These tasks are crucial for a wide range of applications, including video surveillance, traffic monitoring, and autonomous vehicle

Table 3. The qualitative results of the proposed method and comparison with the other methods

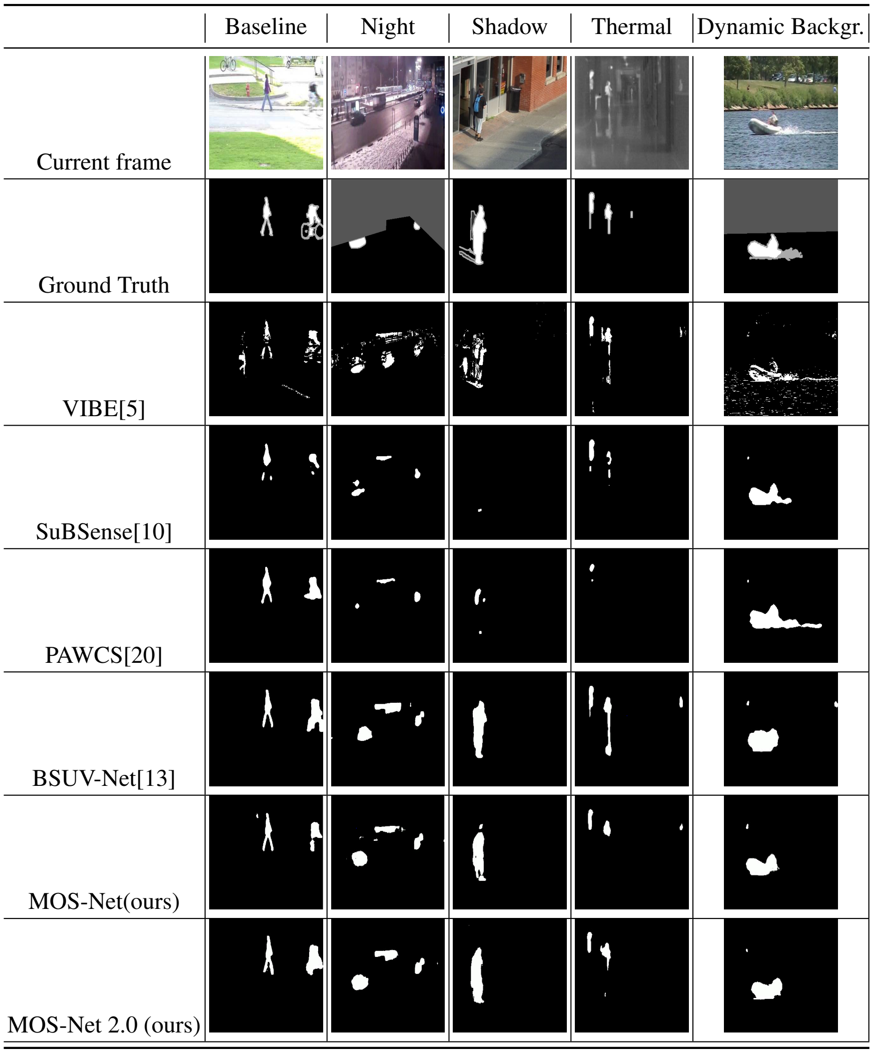


Table 4. The comparison of network parameter size for MOS-Net, MOS-Net 2.0, FgSegNet, and BSUV-Net

Methods	**Network Size (# parameters)**
FgSegNet	9,229,313
BSUV-Net	30,365,825
MOS-Net	30,382,265
MOS-Net 2.0	30,391,545

Table 5. The illustration of the impact of studies

Current Fr.	Flux Tensor	3D CNN	3D CNN+ConvLSTM	Rec	Prec	F1
✓				0.57	0.47	0.43
✓	✓			0.62	0.64	0.54
✓	✓	✓		0.82	0.85	0.83
✓	✓		✓	0.82	0.88	0.85
✓		✓		0.75	0.75	0.75
✓			✓	0.74	0.79	0.76

navigation. Traditional approaches to motion analysis and object segmentation often rely on hand-crafted features and shallow models, which can be sensitive to noise and variations in the data. To address these limitations, we have proposed MOS-Net and MOS-Net 2.0, which are based on fully convolutional encoder-decoder networks. These networks exploit the power of deep learning to learn rich, discriminative segmentation features from the data.

MOS-Net combines the flux tensor algorithm, which is a fast and efficient method for extracting motion information, with 3D convolutional filters to extract spatiotemporal features. The flux tensor algorithm operates on the principle of conservation of mass, and it has been shown to be effective for motion analysis in a variety of scenarios. By combining the flux tensor algorithm with 3D convolutional filters, MOS-Net is able to capture both motion and appearance information, which is essential for accurate object segmentation. MOS-Net 2.0 builds upon the basic architecture of MOS-Net by adding ConvLSTM layers, which are designed to capture long-term temporal dependencies in the data. By incorporating ConvLSTM layers, MOS-Net 2.0 is able to make more informed decisions based on the context of the past and future frames. Simulation experiments have shown that MOS-Net 2.0 outperforms MOS-Net in terms of both accuracy and speed.

Overall, the results of our experiments indicate that MOS-Net and MOS-Net 2.0 are effective methods for motion analysis and object segmentation. They outperform traditional approaches and recent methods such as FgSegnet and BSUV-Net, particularly in terms of F1 score when applied to unseen data. Our main contribution is the successful fusion of spatiotemporal features extracted by deep learning methods and the flux tensor algorithm, which leads to significant performance improvements. We believe that MOS-Net and MOS-Net 2.0 have the potential to make a significant impact on a wide range of applications. The code for the simulations of the proposed deep networks is available at `https://github.com/anilturker/MovingObjectSegmentation`.

References

1. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. CoRR abs/1505.04597 (2015), `http://arxiv.org/abs/1505.04597`
2. Bunyak, F., Palaniappan, K., Nath, S.K., Seetharaman, G.: Flux tensor constrained geodesic active contours with sensor fusion for persistent object tracking. J. Multimed. 2(4) (Aug 2007)

3. Turker, A., Eksioglu, E.M.: A fully convolutional encoder-decoder network for moving object segmentation. In: 2022 International Conference on INnovations in Intelligent SysTems and Applications (INISTA). pp. 1–6 (2022)
4. Stauffer, C., Grimson, W.: 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, pp. 246–252 Vol. 2 (1999)
5. Barnich, O., Van Droogenbroeck, M.: Vibe: A universal background subtraction algorithm for video sequences. IEEE Transactions on Image Processing 20(6), 1709–1724 (2011)
6. Oliver, N., Rosario, B., Pentland, A.: A bayesian computer vision system for modeling human interactions. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8), 831–843 (2000)
7. Braham, M., Piérard, S., Van Droogenbroeck, M.: Semantic background subtraction. In: 2017 IEEE International Conference on Image Processing (ICIP). pp. 4552–4556 (2017)
8. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural Computation 1(4), 541–551 (1989)
9. Babaee, M., Dinh, D.T., Rigoll, G.: A deep convolutional neural network for video sequence background subtraction. Pattern Recognition 76, 635–649 (2018), https://www.sciencedirect.com/science/article/pii/S0031320317303928
10. St-Charles, P.L., Bilodeau, G.A., Bergevin, R.: Subsense: A universal change detection method with local adaptive sensitivity. IEEE Transactions on Image Processing 24(1), 359–373 (2015)
11. Gao, Y., Cai, H., Zhang, X., Lan, L., Luo, Z.: Background subtraction via 3d convolutional neural networks. In: 2018 24th International Conference on Pattern Recognition (ICPR). pp. 1271–1276 (2018)
12. Lim, L.A., Keles, H.Y.: Learning multi-scale features for foreground segmentation. CoRR abs/1808.01477 (2018), http://arxiv.org/abs/1808.01477
13. Tezcan, M.O., Ishwar, P., Konrad, J.: Bsuv-net 2.0: Spatio-temporal data augmentations for video-agnosticsupervised background subtraction. CoRR abs/2101.09585 (2021), https://arxiv.org/abs/2101.09585
14. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proceedings of the 7th international joint conference on Artificial intelligence (IJCAI), pp. 674–679. Vancouver, British Columbia (1981)
15. Nath, S.K., Palaniappan, K.: Adaptive robust structure tensors for orientation estimation and image segmentation. In: Bebis, G., Boyle, R., Koracin, D., Parvin, B. (eds.) Advances in Visual Computing. pp. 445–453. Springer Berlin Heidelberg, Berlin, Heidelberg (2005)
16. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR abs/1409.1556 (2015)
17. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. CoRR abs/1502.03167 (2015), http://arxiv.org/abs/1502.03167
18. Wang, Y., Jodoin, P.M., Porikli, F., Konrad, J., Benezeth, Y., Ishwar, P.: Cdnet 2014: An expanded change detection benchmark dataset pp. 393–400 (2014)
19. Mandal, M., Dhar, V., Mishra, A., Vipparthi, S.K.: 3dfr: A swift 3d feature reductionist framework for scene independent change detection. IEEE Signal Processing Letters 26(12), 1882–1886 (dec 2019), https://doi.org/10.1109%2Flsp.2019.2952253
20. St-Charles, P.L., Bilodeau, G.A., Bergevin, R.: A self-adjusting approach to change detection based on background word consensus. In: 2015 IEEE Winter Conference on Applications of Computer Vision. pp. 990–997 (2015)
21. Bianco, S., Ciocca, G., Schettini, R.: Combination of video change detection algorithms by genetic programming. IEEE Transactions on Evolutionary Computation 21(6), 914–928 (2017)

Anil Turker is an accomplished engineer specializing in the field of Electronics and Communication Engineering. He was born on August 14, 1996, in Amasya, Turkey. Anıl completed his B.Sc. degree in Electric and Electronic Engineering at Ankara Yıldırım Beyazıt University in 2019, where he achieved the top rank in his faculty. He further pursued his academic journey and obtained his M.Sc. degree in Electronics and Communication Engineering from Istanbul Technical University in 2022. Anil's research interests lie in the area of computer vision and deep learning, with a particular focus on moving object segmentation.

Ender Mete Eksioglu received the B.Sc. and M.Sc. degrees in Electrical Engineering from the University of Michigan, Ann Arbor, in 1997 and 1999, respectively. He completed the Ph.D degree at the Istanbul Technical University (ITU) in 2005. He is currently a Professor at the Electronics and Communication Engineering department in ITU. His current research interests include deep learning for imaging problems, sparsity-based signal processing and their applications.

Received: January 29, 2023; Accepted: May 10, 2023.