2021 IEEE International Conference on Multimedia and Expo (ICME) | 978-1-6654-3864-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICME51207.2021.9428086

ROBUST CROSS-SCENE FOREGROUND SEGMENTATION IN SURVEILLANCE VIDEO

Dong Liang[1] *Zongqi Wei*[1] *Han Sun*[1] *Huiyu Zhou*[2]

[1] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics
MIIT Key Laboratory of Pattern Analysis and Machine Intelligence
Collaborative Innovation Center of Novel Software Technology and Industrialization
[2]School of Informatics, University of Leicester

{liangdong, weizongqi, sunhan}@nuaa.edu.cn hz143@leicester.ac.uk

ABSTRACT

[1]Training only one deep model for large-scale cross-scene video foreground segmentation is challenging due to the off-the-shelf deep learning based segmentor relies on scene-specific structural information. This results in deep models that are scene-biased and evaluations that are scene-influenced. In this paper, we integrate dual modalities (foregrounds' motion and appearance), and then eliminating features without representativeness of foreground through attention-module-guided selective-connection structures. It is in an end-to-end training manner and to achieve scene adaptation in the plug and play style. Experiments indicate the proposed method significantly outperforms the state-of-the-art deep models and background subtraction methods in untrained scenes – LIMU and LASIESTA. Source Code is available at: https://github.com/WeiZongqi/HOFAM

Index Terms— foreground segmentation, hierarchical optical flow, cross-scene, attention model

1. INTRODUCTION

Video foreground segmentation aims at discovering the visually distinctive moving foreground objects in a video, and identifying all pixels covering these objects from background. Video foreground segmentation model can serve as an important pre-processing component for many applications, for examples, image and video compression [1], visual tracking [2] and person re-identification [3]. However, in practice, training only one deep model for large-scale cross-scene video foreground segmentation is still challenging issue, since the off-the-shelf deep learning based segmentor relies on scene-specific structural information. Smoothly adapting to new scenes requires additional laborious annotation, training from scratch or fine-tuning the model, otherwise the foreground, especially the tiny ones will be false segmented.

[1]This work is supported by AI+ Project of NUAA (XZA20003), and National Science Foundation of China (61772268).
Corresponding Author: Dong Liang

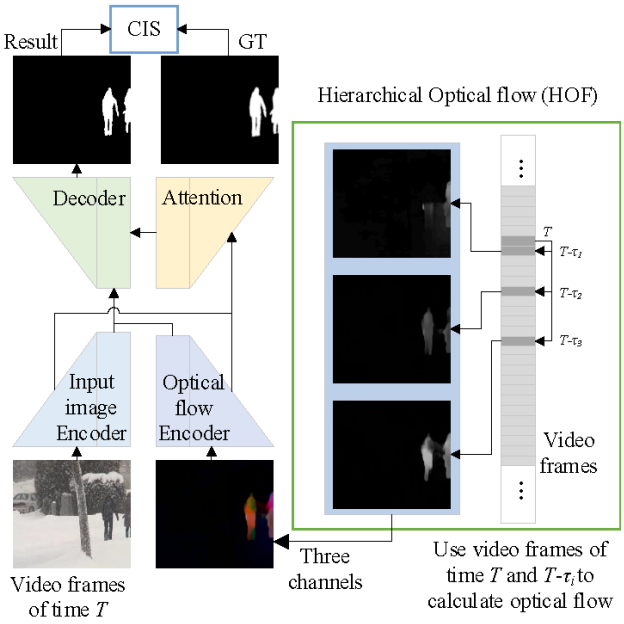

Fig. 1: The proposed foreground segmentation model. It combines both static appearance features and motion information, and integrates attention modules in the upsampling process to fuse the features of encoder and decoder.

Traditional unsupervised foreground subtraction methods [4, 5, 6] focus on building statistical model to suppress interference of dynamic background but they have bottleneck to achieve accurate background updating. Approach using CNN to replace background subtraction were proposed in [7, 8, 9, 10, 11]. All the aforementioned methods are scene-specific and needs to be trained from scratch for other scenes. DeepBS [12] and STAM [13] utilize a trained CNN to realize foreground segmentation across video scenes. For the training data, it randomly select 5% samples with corresponding ground truths of each subset from CDNet2014 dataset. The cross-scene segmentation is often coarse that the boundary of object and small object cannot be well preserved. Semantic

segmentation methods have enabled remarkable progress due to the development of convolutional neural networks. SOTA methods include PSPNet [14], DeepLabV3+ [15], BFP [16] and CCL [17]. Although semantic segmentation approaches could provide high-level semantic annotation for each frame, they ignore the temporal relevance and motion cues which are quite important for video foreground segmentation.

Essentially, foreground segmentation is an empirical task related to appearance, motion, and scene attributes. End-to-end feature descriptor provides a path for effective blending and fusion of appearance and motion features to filter multifarious foreground patterns across scene. Optical flow is an instantaneous motion cue which is less robust and inadequate to describe motions in pixel level. In this paper, we try to solve the following issues: 1) how to describe the foreground more comprehensively in the scene 2) Can we realize a plug and play foreground segmentation model without extra training when use it even for a new scene. We solve these issues by integrating more features from different modalities (foregrounds' motion and appearance), and then eliminating features without representativeness of foreground through attention-module-guided selective-connection structures. The proposed method is shown in Figure 1.

2. OUR WORK

2.1. Model structure

As shown in Figure 1, the proposed model combines both static appearance features and motion information, and integrates attention modules in the upsampling process to fuse the features of encoder and decoder.

2.2. Hierarchical Optical Flow

As a instantaneous motion field, optical flow lacks stability and sufficiency in representing motion. Optical flow from long interval video frames has the long term motion cues of the object but the outline of object is imprecise. Optical flow calculated by short interval video frames has accurate motion cues of the current frame, but sometimes it is insufficient to describe the whole moving object, such as the first optical flow in Figure 1. Hierarchical Optical Flow (HOF), illustrated in Figure 1 right, uses the current video frame and interval frames with different lengths to calculate 3 optical flows. Hierarchical frame interval to complement each other. The specific steps are as follows: The frame position at the current time T, and frames at the time of $T-\tau_1$, $T-\tau_2$ and $T-\tau_3$ by setting the interval frame length parameters τ_1, τ_2 and τ_3. Lastly calculate the optical flow information at time T, which is denoted as $Op(\tau_1)$, $Op(\tau_2)$ and $Op(\tau_3)$. We merge three optical flows with different frame interval into three channels as hierarchical optical flow $Hop(T)$. We use a state of the art deep model Selflow [18] to calculate optical flow.

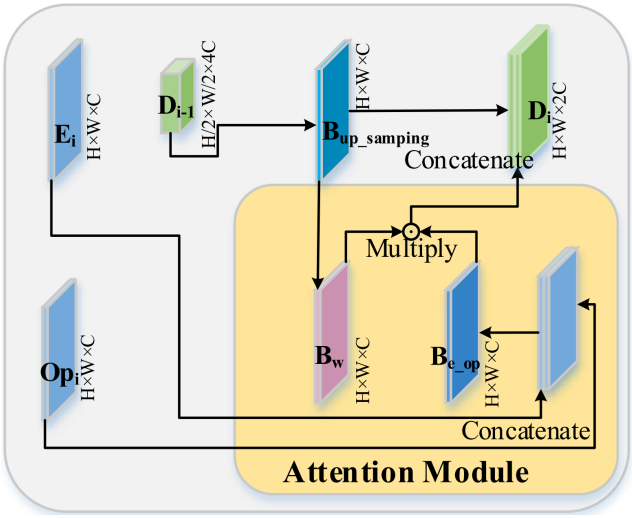

Fig. 2: Attention module in proposed network. The decoding is from the previous decoding layer D_{i-1} to the next layer D_i. The input parts are E_i and Op_i and the previous layer D_{i-1} in the decoder. The output part is decoder layer D_i.

2.3. Attention Module

The proposed model merges the decoder and encoder features through a dense attention processes during the decoder phase. In detail, high-level features provide global information to guide attention modules to weight proper low-level features contribute to prediction in the inputting image that encoder features are re-weighted by the decoder layers in pixel-level and concatenated with the latter.

In Figure 2, the decoding process is from the previous decoding layer D_{i-1} to the next layer D_i. The input parts are E_i and Op_i and the previous layer D_{i-1} in the decoder. The output part is decoder layer D_i. In order to explain the operation mechanism of attention module more clearly, we use $B_{up\_sampling}$, B_w and B_{e_op} as the process stages. The specific process is as follows: Suppose we have obtained two feature map tensors $E_i \in \mathbb{R}^{H\times W\times C}$ and $Op_i \in \mathbb{R}^{H\times W\times C}$ (H and W are the height and width of a single feature map, and C indicates the number of the feature map channels). In order to get D_i, firstly, we concatenate two kinds of corresponding feature map E_i and Op_i in two encoders. After concatenating, the channel becomes twice (2C) as much as the original channel (C), and then $B_{e\_op} \in \mathbb{R}^{H\times W\times C}$ is obtained by convolution.

$$B_{e\_op} = conv_0(Relu(E_i \| Op_i)) \tag{1}$$

where $conv_0$ means convolution of kernel 3×3 and step 1 used to extract appearance feature and reduce channels, $\|$ is the concatenation operator and $Relu$ is the ReLU active functions.

In decoding layer $D_{i-1} \in \mathbb{R}^{H/2\times W/2\times 4C}$, do up sampling convolution to get $B_{up\_sampling} \in \mathbb{R}^{H\times W\times C}$. Then,

the weighted coefficient tensor $B_w \in \mathbb{R}^{H\times W\times C}$ (between 0 and 1) is obtained by convolution and activation operation,

$$B_w = BN(\sigma(conv_1(Relu(B_{up\_sampling})))) \quad (2)$$

where σ is the Sigmoid active functions, $conv_1$ is convolution of kernel 3×3 and step 1 to learn weighted coefficient and BN is batch normalization (BN). Then B_w is combined with the feature map B_{e_op} by multiplying pixel by pixel to obtain the weighted feature map (Atten result). This step is the weighting operation of the decoder in attention module.

After batch normalization, we get original decoder feature from $B_{up\_sampling}$. We also add Dropout operation to the original decoder feature, and each node has a 50% probability of being suppressed during the training process, and removes this operation during the test process. The weighted encoder feature map and the original decoder feature are concatenated to get $D_i \in \mathbb{R}^{H\times W\times 2C}$ in current decoding layer i.

$$D_i = (B_w \odot B_{e\_op}) \| BN(Dropout(B_{up\_sampling})) \quad (3)$$

where $\odot$ is Hadamard product.

2.4. Loss Function

Focal Loss [19] are designed for solving the positive/negative unbalanced sample problem in RetinaNet for object detection which is based on the binary cross entropy function. We define an area ratio between the foreground and background in one frame $S(fg)$, and then define a balance coefficient inside class β, which is shown as follows:

$$\beta = t_3 \min(\frac{1}{S(fg)}, 50) \quad (4)$$

Where t_3 is a hyper-parametric. The reason for setting the minimum value of $\frac{1}{S(fg)}$ and 50 is to prevent the potential scene from infinity, where 50 is the value set after sampling the small object in the training scenes. The class-in scale focal (cisfocal) loss is,

$$\mathcal{L}_{cisfocal} = \begin{cases} -\beta\alpha(1-p)^\gamma \log(p) & y=1 \\ -(1-\alpha)p^\gamma \log(1-p) & y=0 \end{cases} \quad (5)$$

where p represents the probability of model prediction, with foreground label $y=1$ and background label $y=0$. α is the parameter matrix of foreground and background pixel samples. γ is the parameter regulating the contribution of hard and easy samples. For the hard sample case, it will get a lower p. In order to train model stably, Manhattan distance $l1$ *loss* is also used as regularization in the training process. It is measured between the predicted p and ground truth y, $\mathcal{L}_{l1} = ||p-y||_1$. The final loss function can be expressed as follows:

$$\mathcal{L} = t_1\mathcal{L}_{cisfocal} + t_2\mathcal{L}_{l1} \quad (6)$$

3. EXPERIMENT

In this section, we evaluate the proposed network for foreground segmentation on three benchmark datasets, namely CDNet 2014 [20], LIMU [21] and LASIESTA [22]. Quantitative results in terms of average F-measure and visual results are evaluated and verified with the state-of-the-art methods.

3.1. Data Preparation and Experiment Setting

Following the training setting in DeepBS [12], for the training data, we randomly select 5% samples with their ground truths of each subset from CDNet 2014 to train HOFAM. The left 95% of samples in CDNet 2014 are used to test the model, without any overlap of the training set. Segmented foreground is obtained without any post-processing.

We have done a lot of experiments for hyper parameters tuning in advance and compare many different settings. For hierarchical optical flow in the experiment, we set $\tau_1 = 1$, $\tau_2 = 5$ and $\tau_3 = 10$. In the loss function, and finally set $t_1 = 0.8$, $t_2 = 0.2$, $t_3 = 0.25$, $\alpha = 0.75$, $\gamma = 0$. The training batch size is 16, and we train 16000 epochs in all. Adam is used as the optimizer and its beta1 = 0.95, beta2 = 0.999. Learning rate is set to a small value of 5×10^{-5}.

For methods for comparison, we divide them in to three folds: (1) cross-scene deep models (single model), (2) specific-scence models (including deep model and background subtraction methods), and (3) semantic segmentation models. For cross-scene deep models, STAM [13] and DeepBS [12] trained as the same way as HOFAM. We also compare the model without Attention ($HOFAM_{noAtt}$) or Optical flow ($HOFAM_{noOp}$). For semantic segmentation models, DeepLabV3+ [15] and PSPNet [14] train in ADE20K [23], because there is no semantic annotation in CDNet2014. We define some classes as foreground according to the protocol recommended in [24], including {person, car, cushion, box, book, boat, bus, truck, bottle, van, bag and bicycle}.

Precision, Recall and F-measure for segmentation are pixel-level evaluation that accumulate all the positive and negative pixels in all tested frames, but ignore the scale of foreground, which is unfair to small foreground evaluation. In order to more fairly evaluate the results of foreground segmentation of small sizes, we additionally supplement an metric based on dice coefficient as follows:

$$Mean\ Dice = \frac{2}{N}\sum_{i=1}^{N}\frac{(TP+FN)_i \cap (TP+FP)_i}{(TP+FN)_i \cup (TP+FP)_i} \quad (7)$$

Where N is the number of frames that contains foreground, $(TP+FN)_i$ is the truth label in frame i, $(TP+FP)_i$ is the prediction result in frame i.

	$Op[\tau_1]$	$Op[\tau_2]$	$Op[\tau_3]$	Attention	$Loss_{cisfocal}$	$Loss_{focal}$	$Loss_{l1}$	***F-measure***	***Mean Dice***
1	✓	✓	✓	✓	✓		✓	***0.9776***	***0.9466***
2	✓	✓		✓	✓		✓	*0.9704*	*0.9416*
3	✓			✓	✓		✓	*0.9642*	*0.9368*
4	✓	✓	✓	✓		✓	✓	*0.9730*	*0.9408*
5	✓	✓	✓	✓	✓			*0.9735*	*0.9423*
6	✓	✓	✓	✓		✓		*0.9706*	*0.9385*
7	✓	✓	✓	✓			✓	*0.9661*	*0.9334*
8				✓	✓		✓	*0.9030*	*0.8705*
9	✓	✓	✓		✓		✓	*0.8791*	*0.8502*

Table 1: ABLATION EXPERIMENT ON CDNet 2014

3.2. Ablation Experiments on CDNet 2014

In the ablation experiments, we verify the hierarchical optical flow, attention module, and loss function $class\text{-}in\ scale\ focal\ loss$ to $focal\ loss$ with related combinations.

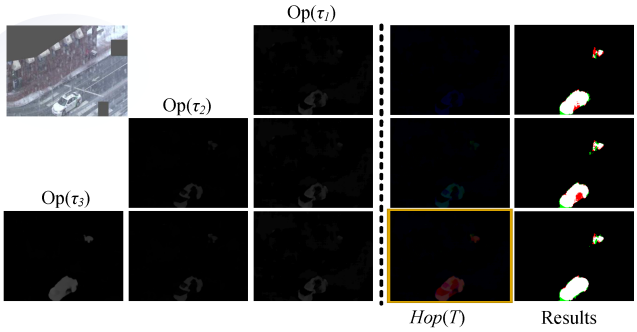

Fig. 3: Hierarchical optical flow and foreground segmentation results.

From Table 1, compared with model just using adjacent optical flow, hierarchical optical flow have obvious improvement in F-measure and Mean Dice. From Figure 3, hierarchical optical flow (orange border) fusing 3 different optical flow provides more sufficient motion cues to guide the foreground segmentation.

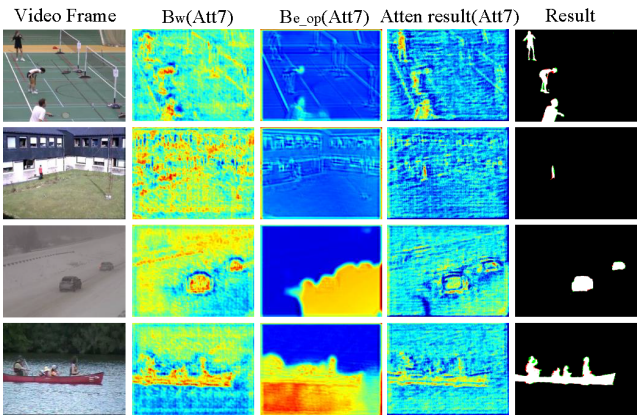

Fig. 4: Visualization of attention module.

From Table 1, compared with model without using attention module, attention module brings obvious improvement in F-measure and Mean Dice. From Figure 4, we visualize the process results of the seventh attention module (Att7) in decoder. Because the proposed attention module involves multi-layer and multi-channel processes, it is difficult to visualize the process of attention directly and accurately through two-dimensional images. We average the results of one layer to reveal this trend roughly. From comparison between Atten result(Att7), attention module highlights the area of foreground object. B_w and B_{e_op} are the intermediate steps to get Atten result. In the result of B_w(Att7) and B_{e_op}(Att7), it seem that B_w and B_{e_op} present more original feature distributions of decoder and encoder with uncertainties and biases on appearance and optical flow.

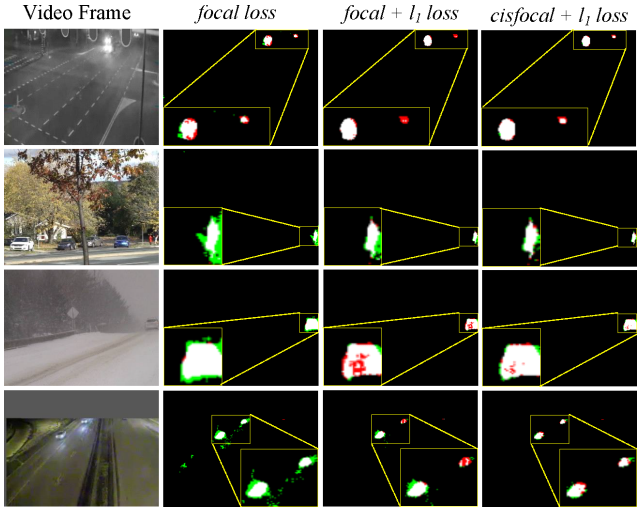

Fig. 5: Comparison results of foreground segmentation of small objects with different losses.

We also compare the results of three different loss function singly. Compared with $focal + l1\ loss$, $cisfocal + l1\ loss$ has obvious improvement and get the best scores in F-measure and Mean Dice. In particular, the improvement of Mean Dice is more obvious. From Figure 5, the proposed loss have better performance of small objects. We use color green and red to mark the false positive and false negative.

Method	Mean Dice ↑	Recall↑	Precision ↑	F-measure↑	Model Types
HOFAM	**0.9466**	**0.9661**	**0.9893**	**0.9776**	
$HOFAM_{noAtt}$	0.8502	0.8369	0.9268	0.8795	
$HOFAM_{noOp}$	0.87055	0.9297	0.8789	0.9036	Cross-scene
DeepBS [12]	0.7041	0.7545	0.8332	0.7548	
STAM [13]	0.9452	0.9458	0.9851	0.9651	
Cascade CNN [8]	0.8947	0.9506	0.8997	0.9209	Specific-scene
FgSegNet [25]	0.5738	0.6073	0.6235	0.6094	deep model
GMM [26]	0.5361	0.6846	0.6025	0.5707	Specific-scene
CPB [27]	0.6157	0.7049	0.6223	0.6325	background subtraction
SuBSENSE [28]	0.6843	0.8124	0.7509	0.7408	

Table 2: AVERAGE PERFORMANCE COMPARISON OF DIFFERENT METHODS ON CDNET 2014.

3.3. Results and Evaluation on CDNet 2014

Since the method proposed in this paper are trained on this dataset, the purpose of this experiment is not to test the capability of cross-scene segmentation, but to test the proposed single model compared with specific scenes. From Table 2, it can be seen from this result that even a single model trained using only 5% of the training data of all scenes, the performance of this method still exceeds deep models and background subtraction models with specific-scene training.

3.4. Cross-Scene Segmentation Results on LIMU and LASIESTA

For cross-scene testing, LIMU [21] and LASIESTA [22] dataset are used to verify foreground segmentation in cross scene. On LIMU, from Table 3, HOFAM presents a better performance on two subsets than other models. On subset of CameraParameter, PSPNet has better results on person segmentation, and HOFAM ranks second with 0.7979. In overall, HOFAM gains the best performance of F-measure 0.7981 while PSPNet ranks second with 0.7506, and STAM ranks third with 0.7344. We visualize the results in Figure 6.

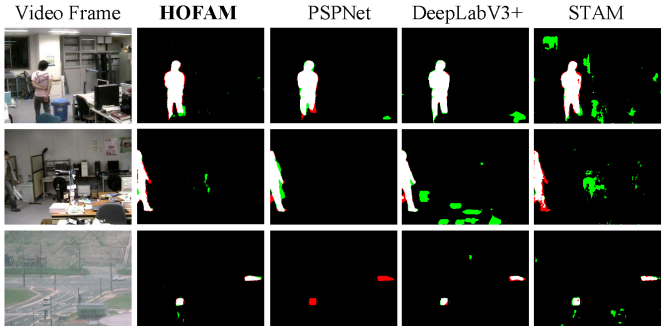

Fig. 6: Comparison on cross-scene dataset LIMU. Each column has five images and there are video frame, segmented results of HOFAM, PSPNet, DeepLabV3+ and STAM, from left to right. Green: False Positive, Red: False Negative.

On LASIESTA, from Table 4, outdoor Moving camera (O_MC), outdoor Cloudy conditions (O_CL), indoor Occlusions (I_OC) and indoor Moving camera (I_MC), are showed. In overall, HOFAM gains the best performance of F-measure

Method	CameraParameter	Intersection	LightSwitch	Overall	Model Types
HOFAM	0.7979	**0.7851**	**0.8493**	**0.7981**	
$HOFAM_{noAtt}$	0.6998	0.7364	0.7965	0.7291	
$HOFAM_{noOp}$	0.7055	0.7294	0.6981	0.7130	Cross-scene
DeepBS [12]	0.6705	0.5545	0.6332	0.6073	training
STAM [13]	0.7742	0.6749	0.7163	0.7344	on CDnet 2014
Cascade CNN [8]	0.1025	0.0453	0.0277	0.0585	
FgSegNet [25]	0.2668	0.1428	0.0414	0.1503	
GMM [26]	0.6372	0.6423	0.6743	0.6519	Specific-scene
CPB [27]	0.6545	0.6778	0.6633	0.6652	background subtraction
SuBSENSE [28]	0.6744	0.6530	0.6934	0.6753	
PSPNet [15]	**0.8656**	0.1303	0.6510	0.7506	Semantic
DeepLabV3+ [14]	0.7739	0.6766	0.3330	0.6986	training on ADE20k

Table 3: F-MEASURE ON LIMU.

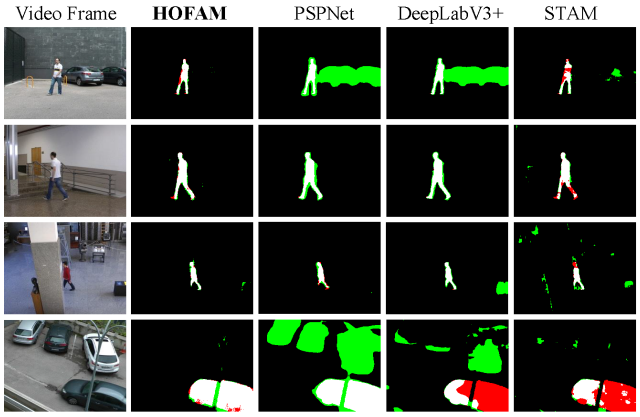

Fig. 7: Comparison of on cross-scene dataset LASIESTA.

Method	O_MC	O_CL	I_OC	I_MC	Overall	Model Types
HOFAM	**0.6919**	**0.8602**	0.8456	0.7895	**0.8072**	
$HOFAM_{noAtt}$	0.5518	0.6364	0.7067	0.5683	0.6148	
$HOFAM_{noOp}$	0.5656	0.6637	0.6883	0.6030	0.6312	Cross-scene
DeepBS [12]	0.7020	0.7673	0.6758	0.5911	0.6774	training
STAM [13]	0.6365	0.7624	0.7362	0.6735	0.6807	on CDnet2014
Cascade CNN [8]	0.1028	0.1414	0.1155	0.1799	0.1288	
FgSegNet [25]	0.1539	0.1687	0.4923	0.4306	0.2447	
GMM [26]	0.3125	0.8027	0.7746	0.2513	0.4527	Specific-scene
CPB [27]	0.2910	0.8407	0.8095	0.0641	0.4304	background subtraction
SuBSENSE [28]	0.3029	0.8327	0.7412	0.1164	0.4425	
PSPNet [15]	0.1652	0.3533	**0.9281**	0.7086	0.3723	Semantic
DeepLabV3+ [14]	0.1675	0.2319	0.8294	**0.8276**	0.3395	training on ADE20k

Table 4: F-MEASURE ON LASIESTA.

0.8072 while STAM ranks second with 0.6807. On outdoor subsets, HOFAM has much higher F-measure than PSPNet and DeepLabV3+. Figure 7 demonstrates the visualized results. The test speed of HOFAM is 5.33 fps for the frame size 256 by 256 on two GTX2080TI with 32 GB RAM, i9 CPU and Ubuntu 16.04 LTS operating system. The entire network uses deep learning framework of Tensorflow 1.13 version.

Conclusions

We propose a Hierarchical Optical Flow Attention Model for cross-scene foreground segmentation to realize cross-scene foreground segmentation task with practical significance. Comparing with the state-of-the-art cross-scene deep

models, specific-scence deep model, background subtraction methods and semantic segmentation models on LIMU and LASIESTA benchmarks indicates its promising generalization capability of the scene without any additional training. Although with dual input, the framework realizes single model and end-to-end training. Future work would be to use self-supervised learning to explore the attention models for specific training scenarios.

4. REFERENCES

[1] L. Itti, "Automatic foveation for video compression using a neurobiological model of visual attention," *IEEE Transactions on Image Processing*, vol. 13, no. 10, pp. 1304–1318, 2004.

[2] W. Hu, X. Li, X. Zhang, X. Shi, S. Maybank, and Z. Zhang, "Incremental tensor subspace learning and its applications to foreground segmentation and tracking.," *Int. J. Comput. Vis.*, pp. 303–327, 2011.

[3] X. Chen, C. Fu, Y. Zhao, F. Zheng, J. Song, R. Ji, and Y. Yang, "Salience-guided cascaded suppression network for person re-identification," in *CVPR*, 2020, pp. 3297–3307.

[4] C.R. Wren, A. Azarbayejani, T. Darrell, and A.P. Pentland, "Pfinder: Real-time tracking of the human body.," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 780–785, 1997.

[5] A. Elgammal, R. Duraiswami, D. Harwood, and L.S. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance.," *Proc. IEEE*, pp. 1151–1163, 2002.

[6] O. Barnich and M. Van Droogenbroeck, "Vibe: A universal background subtraction algorithm for video sequences," *IEEE Trans. Image Process.*, pp. 1709–1724, 2011.

[7] M. Braham and M.V. Droogenbroeck, "Deep background subtraction with scene-specific convolutional neural networks.," *IWSSIP*, 2016.

[8] Y. Wang, Z. Luo, and P. Jodoin, "Interactive deep learning method for segmenting moving objects.," *Pattern Recognit.*, pp. 66–75, 2017.

[9] Y. Wang, L. Zhu, and Z. Yu, "Foreground detection for infrared videos with multiscale 3-d fully convolutional network.," *IEEE Geoscience and Remote Sensing Letters*, pp. 712–716, 2018.

[10] P. W. Patil and S. Murala, "Msfgnet: A novel compact end-to-end deep network for moving object detection.," *IEEE Transactions on Intelligent Transportation Systems*, pp. 4066–4077, 2018.

[11] J. Garca-Gonzlez, J.M.O. de Lazcano-Lobato, R.M. Luque-Baena, M.A. Molina-Cabello, and E. Lpez-Rubio, "Foreground detection by probabilistic modeling of the features discovered by stacked denoising autoencoders in noisy video sequences.," *Pattern Recognit.*, pp. 481–487, 2019.

[12] M. Babaee, D.T. Dinh, and G. Rigoll, "A deep convolutional neural network for video sequence background subtraction.," *Pattern Recognit.*, pp. 635–649, 2018.

[13] Dong Liang and Jiaxing Pan, "Spatio-temporal attention model for foreground detection in cross-scene surveillance videos," *Sensors*, vol. 19, no. 23, pp. 5142, 2019.

[14] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network.," *CVPR*, pp. 2881–2890, 2017.

[15] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation.," *ECCV*, pp. 801–818, 2018.

[16] Henghui Ding, Xudong Jiang, Ai Qun Liu, Nadia Magnenat Thalmann, and Gang Wang, "Boundary-aware feature propagation for scene segmentation," *ICCV*, pp. 6819–6829, 2019.

[17] Henghui Ding, "Context contrasted feature and gated multi-scale aggregation for scene segmentation," in *CVPR*, 2018.

[18] P. Liu, M. Lyu, I. King, and J. Xu, "Selflow: Self-supervised learning of optical flow.," *CVPR*, pp. 4571–4580, 2019.

[19] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollr, "Focal loss for dense object detection.," *ICCV*, pp. 2980–2988, 2017.

[20] N. Goyette, P.M. Jodoin, F. Porikli, J. Konrad, and P. Ishwar, "Changedetection.net: A new change detection benchmark dataset.," *CVPRW*, pp. 1–8, 2012.

[21] University Kyushu, "Limu," 2008.

[22] C. Cuevas, E. M. Yez, and N. Garca., "Labeled dataset for integral evaluation of moving object detection algorithms: Lasiesta," *CVIU*, pp. 103–117, 2016.

[23] B. Zhou, H. Zhao, X. Puig, and S. Fidler, "Scene parsing through ade20k dataset.," *CVPR*, pp. 633–641, 2017.

[24] Marc Braham, Sebastien Pierard, and M Van Droogenbroeck, "Semantic background subtraction," *ICIP*, pp. 4552–4556, 2017.

[25] L. A. Lim and H. Y. Keles., "Foreground segmentation using convolutional neural networks for multiscale feature encoding.," *Pattern Recognition*, p. 256262, 2018.

[26] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking.," *CVPR*, pp. 246–252, 1999.

[27] W. Zhou, K. Shun'ichi, H. Manabu, S. Yutaka, and D. Liang, "Foreground detection based on co-occurrence background model with hypothesis on degradation modification in dynamic scenes.," *Signal Process.*, pp. 66–79, 2019.

[28] P. L. St-Charles, G. A. Bilodeau, and R. Bergevin, "Subsense: A universal change detection method with local adaptive sensitivity," *IEEE Trans. Image Process.*, p. 359373, 2014.