Deep Variation Transformation for Foreground Detection

Yongxin Ge, Xinyu Ren, Chenqiu Zhao, *Student Member, IEEE* Anup Basu, *Senior Member, IEEE*

***Abstract*—Previous approaches to foreground detection generally analyzed the variation of pixel observations. In this paper, we focus on transforming the variation into another space where the entry of variation is easily classified, using a novel foreground detection method called Deep Pixel Variation Transformation Learning (DPVTL). In particular, the pixel variation is represented by the sequence of pixel space, which is used as the input to a fully convolutional network space (FCN) to find a transformation of pixels' variation. Then, the FCN is trained to learn the pattern of pixel variations for the transformation, followed by a linear classifier for labeling the pixels as foreground or background. Resulting from the global analysis of pixel variations and the strong learning ability of deep learning networks, the proposed approach adaptively learns a good transformation from pixel variations to label and generate good performance in diverse natural scenes. Comprehensive experiments on several standard benchmarks demonstrate the superiority of the proposed approach compared to state-of-the-art deep learning and traditional methods.**

***Index Terms*—foreground detection, Feature Transformation, Deep Learning,**

I. Introduction

Foreground detection is a fundamental problem in computer vision [1] that has been approached over decades with an increasing number of cameras. It is widely used in applications involving video processing [2]. Typically, it is recognized as a binary classification task that assigns a label to each pixel in the video stream, for either belonging to the background or foreground scene. Traditional foreground detection algorithms focus on analyzing pixel variations, establishing background models with statistical methods such as GMM [3] [4] and KDE [5] [6]. However, due to the unpredictability and speed of pixel variations in natural scenes, the variation becomes unordered and is hard to analyze for foreground detection. Therefore, foreground detection is still a challenging problem in complex natural scenes.

In diversely natural scenes, it is possible that moving objects produce similar or even the same observations of pixels compared to the background. As shown in Fig. 1, observation C is closely related to the observations which belong to the background. However, it is actually produced by moving objects and should be classified as foreground. Unfortunately, in most cases, it is quite possible that C will be falsely classified due to the similarity with counterparts of the background. In this work, we focus on learning the pattern of pixels' variation and transforming the variations into a new space where the observations are easier to classify, rather than learning a classifier for individual pixels. As shown in the bottom part of Fig. 1, the pattern of a fragment consisting of observations A-D can be learned by the network and transformed into another fragment where these observations are easily and correctly classified as foreground. Based on this, the Deep Pixel Variation Transformation Learning model for foreground detection is proposed.

Yongxin Ge is with School of Software Engineering, Chongqing University, Chongqing (401331), PR China.(e-mail: yongxinge@cqu.edu.cn)

Xinyu Ren is with School of Software Engineering, Chongqing University, Chongqing (401331), PR China.(e-mail: hsinyujen@gmail.com)

Chenqiu Zhao is with Department of Computing Science, University of Alberta (T6G 2E1), Canada.(e-mail: zhao.chenqiu@ualberta.ca)

Anup Basu is with Department of Computing Science, University of Alberta (T6G 2E1), Canada.(e-mail: basu@ualberta.ca)

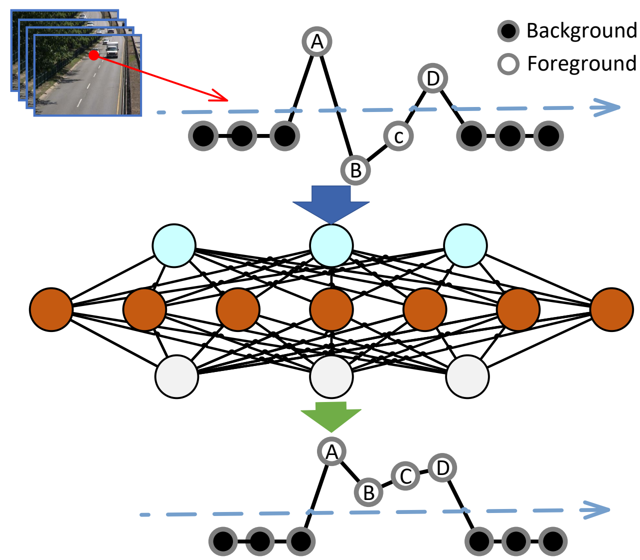

Fig. 1. Demonstration of deep variation transformation. Due to the complexity of natural scenes, the original pixels' variation is hard to classify correctly. After transforming by a deep learning network, the pixels in variation become easy to be correctly classified as foreground and background correctly.

In the DPVTL model, the sequence of pixel observations is used to represent the variations and input into the network for learning, which encodes both the intensity distribution and the sequential information. Then, a Fully Convolutional Network (FCN) [7] is applied to learn patterns of pixel variations and find a transformation which guarantees the linear separability of pixels in the transformed variation. In particular, we take advantage of the strong learning ability of deep neural networks to learn an end-to-end representation of the pixel variation in a new space where they can be easily classified as background or foreground. Utilizing the innovative framework of variation transformation, the proposed approach works well for a diverse range of complex scenes.

The rest part of this paper is organized as follows: In Section II, we briefly discuss both early and recent relevant literature. Details on the variation transformation are presented in Section III. The architecture of the network is introduced in Section IV, This is followed by experiments and comparison results in Section V, before concluding the paper in Section VI.

II. Related Work

Over the last few decades, a large number of foreground detection methods have been proposed, which are broadly categorized into pixel-based, region-based and learning-based methods.

A. Pixel-based methods

Pixel-based methods usually assume the independence between neighboring pixels, and utilize low-level features, such as color or gradients for background subtraction.

In particular, the Gaussian Mixture Model (GMM) proposed by Stauffer et al. [3] is the most popular approach among pixels-based methods [8]. It utilizes a mixture of weighted Gaussians to model the probability distribution of each pixel over a time sequence. Pixels are considered as background when the probability generated by Gaussians are higher than a user threshold. Zivkovic et al. [9] improved the GMM method by utilizing a recursive equation to automatically update the parameters and adjust the number of components of the mixture needed for each pixel. Kim et al. [10] presented the codebook method, which records the sampling background values on codewords for each pixel position. The incoming pixels are compared with these codewords to see if their distances lie within a certain bound. In addition, a non-parametric background model is proposed by Elgammal et al. [5]. They assume that each background pixel is drawn from a probability distribution function, which is estimated in the Kernel Density Estimation space (KDE). Another non-parametric method is proposed by Barnich et al. [2], called the Visual Background Extractor (ViBe). The background model of ViBe consists of pixel samples from the video stream. Each pixel in the current frame is compared with sample pixels from the corresponding background model and labelled as the foreground when there exists sufficient samples with a distance to itself within a certain range. To adaptively update the parameters, Hofmann et al. [11] improved ViBe by presenting an adaptive threshold, which depends on the pixel position and a background dynamics metric.

Unfortunately, the pixel-based methods ignore the spatio-temporal information due to their assumption of independence between pixels. But there is, in fact, a strong coherence in image sequences that contain abundant hidden clues for the background model. To address this shortcoming, we introduce a framework of variation transformation learning, where a pixel's historical observations are embedded into a piece of pixel patch and sorted in chronological order as a whole for the training of our DPVTL model. Bringing together the historical observations ensures data integrity and preserves the temporal coherence of our training data. On the other hand, the application of a neural network guarantees strong learning ability of our model. Thus, the proposed approach is more capable of learning the patterns of pixel variation and exploiting the spatio-temporal context, compared to those classifiers based on the assumption of pixel independence.

B. Region-based approaches

Region-based approaches are usually performed at block-level resolution, in order to exploit the spatial context between neighboring pixels.

Varadarajan et al. [12] proposed a region-based GMM model, which is derived using the expectation-maximization (EM) theory considering neighboring pixels. In addition, Chen et al. [13] combined GMM with constraints on temporal and spatial information from the optical flow and hierarchical superpixels. Moreover, Sheikh et al. [14] introduced a framework based on the Markov Random Field modeling with Maximum A Posteriori probability (MAP-MRF) estimation, which incorporates the pixel location into background and foreground KDEs for the detection based on spatial context. Similarly, in [15], original images are divided into overlapping blocks. Each block is sequentially processed by an adaptive multi-stage classifier, which consists of a likelihood evaluation, an illumination invariant measure and a temporal correlation check. Hence, Izadi et al. [16] presented a robust region-based approach, which generates a pair of foreground maps based on gradient and color respectively. Any foreground region that does not exist in the first foreground map could be recovered from the other one.

In contrast, we accept the assumption that neighboring blocks of background pixels should follow similar variations over time, and combined the pixel variation with its spatial neighbors to revise our prediction. Due to the application of deep learning, the proposed approach is more powerful in capturing the structural background variation and achieves significant improvement compared to its competitors.

C. Machine Learning based Methods

The last category of background subtraction methods applies traditional machine learning and deep learning for foreground detection.

Traditional machine learning methods are commonly involved with support vector machines (SVM) [17] [18] and Bayesian methods [14] [19] [20]. For example, Han et al. [18] integrated gradient, color, and Haar-like features to address the spatio-temporal variations for each pixel. Their background model is obtained for each feature in a kernel density framework and a SVM is employed for classification.

In recent years, deep learning methods have begain began to flourish in several computer vision fields, and have demonstrated effective capabilities to handle time-series problems [21] [22]. A novel approach for foreground detection with the use of the Convolutional Neural Network (CNN) is proposed by Wang et al. [23]. They utilize a CNN with a cascade network architecture for segmentation in foreground detection, which performs very well with sufficient training data. Braham et al. [24] employed a scene-specific CNN, which is trained with corresponding image patches from the background image, video frames and groundtruth. In particular, the background image is obtained by temporal median filtering, and the groundtruth can be replaced by segmentation results from other foreground detection methods. A similar approach is presented by M. Babaee et al. [25]. Their background images combine the segmentation mask from the SuBSENSE [26] algorithm and the output of Flux Tensor algorithm [27], which is able to adaptively update the parameters used in the background model. They also utilize spatial-median filtering for post processing of the network predictions. In [28], a fully convolutional network with the skip architecture is proposed. The authors also utilize a temporal approach to sample training images from the given video, thereby providing the background model with limited temporal information.

Some breakthroughs and progress have been achieved by applying the methods above, especially deep learning based approaches with optimized network architectures. However, due to the diversity of natural scenes, pixel variations can be very complex, leading to difficulty in analyzing specific pixel variation patterns. Unlike existing ones, our method focuses on creating a transformation, which guarantees the linear separability of pixel variations in the mapping space, rather than building classifiers for individual observations. Besides, our network is trained to learn patterns of pixel variations, which results in a better performance in time series modeling on temporal coherence.

III. Variation Transformation

In this section, we will elaborate on the proposed approach and how the variation transformation works in foreground detection.

Foreground detection is essentially a binary classification in time sequence, where pixels in a video stream are classified into two categories: foreground (cars, people or animals) and background (roads, trees or other backdrops). In most cases, if we print out the historical observations of a single pixel, it is not hard to see that they usually keep their stability when belonging to the background. However, there are some exceptions: historical observations vary regularly when belonging to some dynamic background scenes (e.g. waves, swaying tree leaves). In general, pixels from the background share some common patterns of variation.

To give an intuitive understanding of the pixel variation, we plot the historical observations and the corresponding groundtruth of a single pixel in Fig. 2. As we can see, there are several lines in three colors: blue, red and green. The blue line consists of original observations, representing a piece of pixel variation. Accordingly, the red line is made of the groundtruth data, corresponding to the observations from the blue one. And the green line consists of the outputs of the proposed approach which represents the transformed variation.

Previous methods have good performance when background and foreground observations are significantly different and the backgrounds are normally static. While in fact, backgrounds can be rather turbulent and dynamic due to the complexity and diversity of natural scenes. Especially when illumination and camouflage are involved, background and foreground observations are easily confused and mixed up.

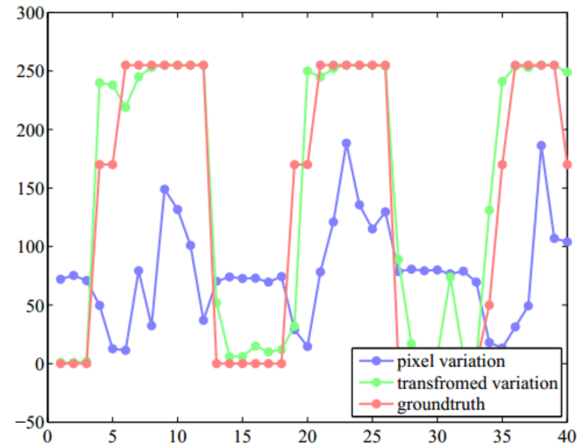

Fig. 2. Comparison of the original pixel variation, groundtruth and transformed variation. They are represented by the blue, red and green lines respectively.

For example, in Fig. 2, it is noticeable that some foreground observations on the blue line share the same or similar value with background ones. In this case, popular methods will inevitably yield some sticky moments when separating the pixels in the blue line. Statistical methods, for instance, are no longer valid, because they only focus on establishing a statistical model for the background, while having little or no concern for the temporal coherence of these observations. In other words, despite knowing that pixels from the background share some common patterns of variation in a temporal sequence, they still let the order information of sequential images be ignored.

Differing from existing methods, we decided to shift our attention from single pixel classification to variation transformation. More specifically, a FCN is employed to learn patterns of pixel variations and mapping them into a new space where they are close to the groundtruth data. The transformed variation is shown by the green line in Fig. 2. The benefits are evident and can be clearly seen. It is hard to distinguish a foreground observation when its value is similar to the background ones. Classification on the transformed variation, however, is much more convenient and intuitive. With the strong learning ability of the FCN, we are able to learn the diverse variation patterns of background and foreground over a long period of time. The variation patterns, in turn, contribute to the variation transformation where the value of each observation is adjusted according to its compatriot in time sequence.

IV. Foreground Detection via Deep Variation Transformation

In this section, details of the proposed approach are explained. The pipeline of the proposed approach is shown in Fig. 3, which can be divided into three parts. First, the procedure generating pixel patches, which are fed into the network for training as well as the description of pixel variation, is introduced. Then, the architecture of the network, which is devised for transforming the variation of pixels' observations, is demonstrated. Next, a reverse procedure for the generation of training data is used to reconstruct the foreground image from the transformed results.

Training Data Generation

Fundamentally, background subtraction is a classification of pixels' observations in time sequence. In this work, a deep learning network is utilized to learn the pattern of variations implied in the observations of pixels. Therefore, the training data fed into the network is a type of description of pixels' variation, which are actually patches consisting of pixels' temporal observations. In contrast, the output of the network is the probability of corresponding pixels belonging to foreground or background, and the network is devised with an end-to-end architecture to transform the intensity of pixels to the probability of the category the pixels belong to.

Let us denote the frames in a video sequence as $\mathcal{I} = \{I_1, I_2, \cdots, I_T\}$, where $\mathcal{I}$ denotes the set of frames and T is the numbers of frames. There are usually large numbers of frames in a particular video, and the number of frames for different videos are also different. In order to capture the global representation of pixel's variation, the training frames are sampled from the whole frame sequence uniformly. The uniform sampling procedure is shown as follows:

$$\mathcal{I}^d = \mathcal{D}(\mathcal{I}) = \{I_1, I_{[2\cdot\frac{T}{N}]}, \cdots, I_{[(N-1)\cdot\frac{T}{N}]}, \} \tag{1}$$

where $\mathcal{I}^d$ is the set of training frames, T the N represent the frame numbers of the entire video and the training set respectively, and $\mathcal{D}()$ denotes the sampling function.

After sampling training frames, the training patches, which are used as the input to the network, are captured. The training patch is the description of the pixels' variation, which is captured by reshaping the observations in a particular pixel into a square patch. Since the patch can be captured from each pixel, a large number of training patches can be extracted with the training frames; and each of these patches has $\frac{T}{N}$ pixels. In addition, for each pixel included in a patch, a label of the foreground or background is needed for training. Therefore, the labeling patches, which are captured by a similar procedure in the ground truth frames, are used as the output for training our network. Our network is actually an end-to-end transformation from pixels to labels. Mathematically, the generation procedure of training patches can be shown as follows:

$$S_{x,y}(m,n) = \mathcal{C}(\mathcal{I}^d, R) = \mathcal{I}_{m\times R+n}(x,y), \tag{2}$$

where $S_{x,y}(m,n)$ denotes the training patch extracted from the observations of pixel located at (x,y), R is the user argument to control the radius of training patches and $\mathcal{C}(\mathcal{I}^d, R)$ is the function to convert the observations of pixels to training patches.

It is generally accepted that a pixel is not independent but related to its neighborhood, and similar neighboring pixels share similar variations. Therefore, spatial information should benefit from learning the pattern of pixels' variation. With this motivation, several training patches extracted from neighboring pixels are combined to improve the efficiency of the proposed approach. In the experiments, three training patches are combined as follows:

$$S_{x,y}(m,n) = \mathcal{C}o(S_{x,y}(m,n), S_{x',y'}(m,n), S_{x'',y''}(m,n)), \tag{3}$$

where $\mathcal{C}o$ represents the concatenation process on the third dimension. And neighboring patches are at pixel position (x', y') and (x'', y'') respectively.

Network Architecture and Variation Transformation

As we mentioned above, the main purpose of our DPVTL algorithm is the transformation between the sequence of pixels' observations with possibility of pixels belonging to foreground or background. Therefore, our network entails learning a mapping from the input observations of pixels $s \in \mathcal{S}$ to the probability $p \in \mathcal{P}$ of the corresponding pixel belonging to foreground or background. An end-to-end architecture network is proposed to transform the observations of pixels to a new space which is close to labels, as shown in Fig. 3. Mathematically, the training procedure is as follows:

$$\begin{aligned}\mathcal{P}(m,n) &= f_{\theta_N}(f_{\theta_{N-1}}(\cdots f_{\theta_1}(\mathcal{S}(m,n))\cdots)) \\ &= f^n_{\prod_{n=1}^{N}\theta_n}(\mathcal{S}(m,n)),\end{aligned} \tag{4}$$

where $\mathcal{P}(x,y)$ is the probability, $\mathcal{S}(m,n)$ is the input patch of the network and f_{θ_i} represents different layers of the network.

For the loss function, Sigmoid Cross Entropy (SCE) loss is used, which helps address the learning slowdown. The formulation is as follows:

$$\begin{aligned}\mathcal{L}(S'(x,y), S^{gt}(x,y)) = \sum_{x=1}^{m}\sum_{y=1}^{n} S^{gt}(x,y)log(Sig(S'(x,y))) \\ + (1 - S^{gt}(x,y))log(1 - Sig(S'(x,y))),\end{aligned} \tag{5}$$

$$Sig(x) = \frac{1}{1+e^{-x}}, \tag{6}$$

where $S^{gt}(x,y)$ denotes the groundtruth patch which is given by the corresponding groundtruth stack, and Sig represents the sigmoid function. SCE is calculated between the transformed patches and the corresponding groundtruth patches. Boundaries of moving objects and pixels that are out of the region of interest are not taken into account in the cost function. Finally, we get the transformed variation through the FCN, which is much easier for classification.

The architecture of our FCN for foreground detection is illustrated in Fig. 3. The proposed FCN contains 6 convolutional layers without pooling, and the filter size of the first five layers is set to 3×3, compared to the last convolutional layer which has a filter size of 1×1. We use the Rectified Linear Unit (ReLU) as the activation function after first three convolutional layers and the Sigmoid Cross Entropy loss function is applied to measures the performance of our network after the last convolutional layer. We do not use any other adjustment in our network training and the experimental results prove that the proposed approach is feasible and very effective. After the thresholding calculations, our experiments show that

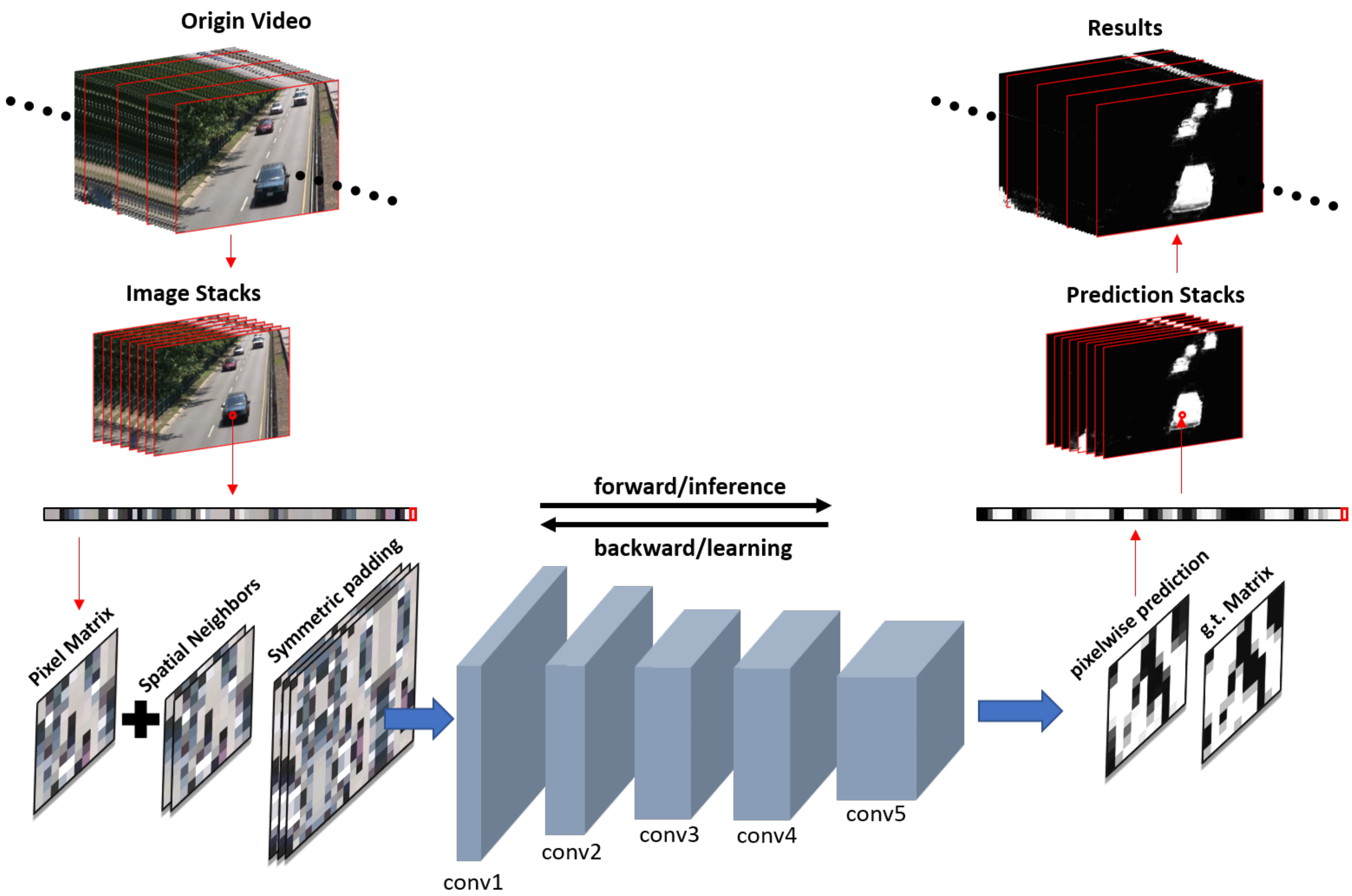

Fig. 3. Original videos broken down into pixel patches for training of the FCN, which produces an efficient machine for end-to-end transformation learning.

a randomly initialized FCNs, trained end-to-end on feature learning can achieve state-of-the-art performance without any extra optimization mechanism.

A. Foreground Detection

After the transformation from observations to the probability of belonging to foreground or background, the foreground is detected from the final binary mask. During the transformation, the pixel patches are fed into the network and transformed into the probability patches, which are compared with a threshold T_r for classification and then used to reconstruct the foreground to obtain the final results. Comparison with a threshold can be mathematically shown as:

$$\mathcal{M}(x,y) = g(T_f, \mathcal{C}^{-1}(S'(x,y), R)), \tag{7}$$

where the $\mathcal{M}(x,y)$ is the final binary mask. $S'(x,y)$ is the probability of patches output by the network, $\mathcal{C}^{-1}$ is the reverse of the reshape procedure shown in Eq. 2 and R is the parameter for the reversing reshape procedure. T_f is the threshold given by a user and $g()$ is the thresholding function shown in Eq. 8.

$$g(x,y) = \begin{cases} 1, & x < y \\ 0, & otherwise \end{cases}, \tag{8}$$

V. Experiments

In this section, we describe comprehensive experiments to evaluate the performance of the proposed approach on the CDnet 2014 benchmark [29] and CAMO-UOW [30]. CDnet is the largest dataset for foreground detection as far as we know, containing 11 categories with several complex challenging scenes, such as Dynamic Background, Camera Jitter, Shadow, Night Videos, PTZ and so on. The CAMO-UOW is a challenging benchmark for camouflaged foreground detection, containing 10 high resolution videos. For each video, one or two persons appear in the scene with clothes having color similar to the background.

The proposed approach is compared with several existing traditional state-of-the-art foreground detection algorithms, including IUTIS-5 [31], SuBSENSE [26], WeSamBE [32], sharable GMM the SharedModel [33], word-dictionaries-based method like PAWCS [34], SemanticBGS [35] and AAPSA [36]. Moreover, two deep learning based algorithms are also compared with the proposed approach, including DeepBS [25], and DBMF [28]. The results of the compared algorithms are captured by the implementation provided by the authors.

For comparison, the F-measure Space (Fm) is used for evaluation. Fm is a general international standard in foreground detection which measures the segmentation accuracy by considering both recall and precision. The definition of Fm is as follows:

$$Fm = \frac{2 \times precision \times recall}{precision + recall} = \frac{2TP}{2TP + FN + FP}, \quad (9)$$

where TP, FP, and FN are true positives, false positives, and false negatives respectively, computed on the pixels of all the test frames for each video.

Quantitative and qualitative comparisons are shown in Table I and Fig. 4 respectively. To keep the length of this paper reasonable, only several typical videos are selected for the qualitative comparisons and discussion. In the dynamic background scene, the video "canoe" is a typical challenging video which includes a large area of rippling water. The main challenge comes from the dynamic background, in which it is hard to describe the background by a single image. For this condition, since traditional foreground detection methods such as the SharedModel and the WeSamBE cannot describe the complex dynamical background, they fail to detect the people on the boat, as shown in Fig. 4. Besides, the detected moving objects of the SharedModel are not accurate in the boundary due to the utilization of texture features. In contrast, benefiting from the strong learning ability of Deep Learning networks, DeepBS successfully detects the people. Unfortunately, since the DeepBS ignores the fact that a single background is not enough to describe the dynamic background, even this algorithm suffers from poor detection of the boat shape. In contrast, the proposed approach performs better than others in this scene, since the essence of foreground detection is considered as a binary classification of pixels' observation over a time sequence. Based on this insight, the FCN network focuses on learning the patterns of pixel variation rather than a static background image. Thus the proposed approach achieves promising performance for the canoe video.

As for the case of the shadow scene, "peopleInShade" is a typical example with prevalent hard and soft shadows. For traditional approaches, these shadow regions are usually segmented as foreground since they also movie with the objects. Therefore, traditional methods like PAWCS, WeSamBE and SharedModel falsely segment part of shadows as moving objects. In addition, the foreground provided by IUTIS-5 is incomplete on part of the pedestrian's body due to the interference of shades, because of the severe dependency on texture features. On the other hand, DeepBS performs well for this video benefiting from the utilization of CNN. However, the shape of pedestrians are slightly deformed as a result of the patch-wise processing in CNN. In contrast, since our DPVTL focusses on learning the pattern of pixels' variation in the shadow regions, the proposed approach successfully segments the shadow part as the background and achieves the highest performance in the category of shadow scene.

In the video "corridor" among the Thermal scene, there is no color information since the videos are obtained through a Thermal camera. Moreover, moving objects in these videos are exceedingly fuzzy and unclear, which is the main challenge for this category. WeSamBE, SharedModel and PAWCS successfully detect the target objects, because of a stable background in this indoor video. However, they fail to remove the reflections since their modeling ability has already reached a limit under the extreme condition of the thermal map. DeepBS, by contrast, succeeds in eliminating most of the reflections. Meanwhile, moving objects are also clearly divided from the background thanks to the strong modeling ability of CNN. However, due to the dependency of the edge feature, a small object was missed in the detection result. Fortunately, the proposed approach utilizes the pattern of pixels' variation, which is theoretically effective even in observations without color information. Consequently, our DPVTL performs much better than other algorithms, with most parts of the shadow being removed and the segmentation results become more accurate.

The quantitative evaluation of the proposed approach on CDnet 2014 is shown in Table I. It can be seen that the proposed approach significantly outperforms all of the compared state-of-the-art algorithms in most of the complex scenes and achieves 11.37% gain in FM over the second one on the whole dataset. Moreover, since DBMF only published their results in several specific videos instead of the whole dataset, the proposed approach was also ran in these videos to compare with DBMF with the results shown in Table 2. This also demonstrates the superiority of the proposed approach.

As shown in Table I and Table II, previous deep learning based methods like DeepBS and DBMF achieve good performance. From our perspective, this good performance can be attributed to the stronger modeling ability and learning adaptation of CNN and FCN. However, the proposed approach utilizes pixel variations in a temporal sequence rather than low-level static features such as color, edges and textures, which give us the ability to avoid the shortcomings of the background models. Consequently, the proposed approach produces considerably better results, with improvement over 17.85% in FM metrics compared to DeepBS and over 9.50% compared to DBMF.

The evaluation of the proposed approach in CAMO-UOW dataset is shown in Table III. Unlike the CDnet dataset, the videos of CAMO-UOW dataset are specially proposed for moving objects with camouflage, which is the main challenge of this dataset. As shown in Table III, the proposed approach achieves better performance compared to its competitions, with an average F-measure of 0.97, compared to values between 0.77 and 0.94 for other methods. Therefore, it is fair to say that the proposed approach performs better compared to the state-of-the-art.

In the CAMO-UOW dataset, target objects have similar color and textures as the background. This creates many difficulties and obstacles for traditional methods. However, our FCN is a powerful Neural Network model which is good at capturing the non-linearities of the manifold of pixel variations.

All of the experiments of the proposed method were implemented in Matlab and run on a computer with Nvidia Tasla K80 GPU with all images kept at their original resolution. For each video in CDnet 2014, 100 training frames are extracted to produce the image patch. It should be noted that the 100 frames only account for less than 10% of total Groundtruth in CDnet 2014. In contrast, 90% of data was used as training

Fig. 4. The qualitative evaluation of the proposed method. All the results is followed in the CDnet 2014.

TABLE I
PERFORMANCE COMPARISON OF THE PROPOSED APPROACH AND SOME STATE-OF-THE-ART ALGORITHMS ON THE VIDEO SEQUENCES FROM DIFFERENT CATEGORIES IN CDNET 2014. FOR EACH VIDEO, 100 FRAMES WERE TAKEN FOR TRAINING OUR FCN.

Videos	baseline	dyna.bg	cam.jitter	int.obj.m	shadow	thermal	bad.weat	low f.rate	night vid.	PTZ	turbul.	overall
DeepBS [25]	0.9580	0.8761	0.8990	0.6097	0.9304	0.7583	0.8647	0.5900	0.6359	0.3306	0.8993	0.7458
IUTIS-5 [31]	0.9567	0.8902	0.8332	0.7296	0.9084	0.8303	0.8289	**0.7911**	0.5132	0.4703	0.8507	0.7717
FTSG [27]	0.9330	0.8792	0.7513	0.7891	0.8832	0.7768	0.8228	0.6259	0.5130	0.3241	0.7127	0.7283
AAPSA [36]	0.9183	0.6706	0.7207	0.5098	0.7953	0.7030	0.7742	0.4942	0.4161	0.3302	0.4643	0.6179
CwisarDH [37]	0.9145	0.8274	0.7886	0.5753	0.8581	0.7866	0.6837	0.6406	0.3735	0.3218	0.7227	0.6812
PAWCS [34]	0.9397	0.8938	0.8137	0.7764	0.8934	0.8324	0.8059	0.6433	0.4171	0.4450	0.7667	0.7403
SuBSENSE [26]	0.9503	0.8177	0.8152	0.6569	0.8986	0.8171	0.8594	0.6594	0.4918	0.3894	0.8423	0.7408
SemanticBGS [35]	0.9604	**0.9489**	0.8388	0.7878	0.9244	0.8219	0.8260	0.7888	0.5014	0.5673	0.6921	0.7892
MBS [38]	0.9287	0.7915	0.8367	0.7568	0.8262	0.8194	0.7980	0.6350	0.5158	0.5520	0.5858	0.7288
WeSamBE [39]	0.9413	0.7440	0.7976	0.7392	0.8999	0.7962	0.8608	0.6602	0.5929	0.3844	0.7737	0.7446
ShareM [40]	0.9522	0.8222	0.8141	0.6727	0.8898	0.8319	0.8480	0.7286	0.5419	0.3860	0.7339	0.7474
GMM [9]	0.8245	0.633	0.5969	0.5207	0.7370	0.6621	0.7380	0.5373	0.4097	0.1522	0.4663	0.5707
RMoG [41]	0.7848	0.7352	0.7010	0.5431	0.7212	0.4788	0.6826	0.5312	0.4265	0.2470	0.4578	0.5735
DPVTL	**0.9811**	0.9329	**0.9014**	**0.9595**	**0.9467**	**0.9479**	**0.8780**	0.7818	**0.7737**	**0.5957**	**0.9034**	**0.8789**

samples in DBMF, which suggests that the proposed approach achieves better performance with limited training frames. Considering that videos in CAMO-UOW have fewer frames, we reduced the number of training frames to 64 for each video. During the experiments, the training and testing sets are completely separated. More specifically, our FCN network is randomly initialized. We train the network with mini-batches of size 200, and a learning rate of $\alpha = 1 \times 10^3$ over 20 epochs. The last threshold R is set to 0.6.

VI. CONCLUSION

In this paper, we proposed a novel foreground detection approach based on our proposed variation transformation learning. Transformation learning focused on learning a transformation from pixel variations to the probability of pixels belonging to the foreground or background. The global variation patterns of pixel observation is learned for better classification of pixels. In particular, a FCN network and a

TABLE II
PERFORMANCE COMPARISON OF THE PROPOSED APPROACH AND SOME CLASSICAL METHODS AND DEEP-BASED METHOD DBMF. FOR EACH VIDEO, 64 FRAMES WERE TAKEN FOR TRAINING OUR FCN.

Methods	highway	office	Pedestrians	PETS2006	Fall	sofa	overall
GMM [3]	0.5788	0.2338	0.5202	0.6011	0.8026	0.5225	0.5432
CodeBook [42]	0.8356	0.5939	0.7293	0.7808	0.3921	0.8149	0.6911
ViBe [2]	0.7535	0.6676	0.8367	0.6668	0.6829	0.4298	0.6729
PBAS [11]	0.8071	0.6839	0.7902	0.7280	0.3420	0.5768	0.6547
P2M [43]	0.9160	0.3849	0.9121	0.7322	0.5819	0.4352	0.6604
DBMF [28]	0.9412	0.9236	0.8394	0.9059	0.8203	0.8645	0.8824
DPVTL	**0.9888**	**0.9819**	**0.9728**	**0.9808**	**0.9394**	**0.9333**	**0.9662**

TABLE III
PERFORMANCE COMPARISON OF THE PROPOSED APPROACH AND SOME STATE-OF-THE-ART ALGORITHMS ON VIDEO SEQUENCES FROM DIFFERENT CATEGORIES IN CAMO-UOW.

Methods	MOG2 [44]	FCI [45]	LBA-SOM [46]	PBAS	SuBSENSE	ML-BGS [47]	DECOLOR [48]	COROLA [49]	FWFC [50]	DPVTL
Video 1	0.79	0.88	0.8	0.9	0.89	0.89	0.92	0.8	0.94	**0.96**
Video 2	0.82	0.79	0.8	0.82	0.88	0.8	0.83	0.58	0.96	**0.98**
Video 3	0.88	0.86	0.85	0.91	0.9	0.8	0.9	0.82	0.94	**0.95**
Video 4	0.89	0.9	0.76	0.93	0.78	0.88	0.95	0.87	0.94	**0.98**
Video 5	0.84	0.86	0.82	0.83	0.82	0.8	0.82	0.75	0.91	**0.98**
Video 6	0.93	0.87	0.77	0.95	0.92	0.95	0.97	0.72	0.94	**0.98**
Video 7	0.76	0.83	0.88	0.91	0.87	0.79	0.91	0.83	0.96	**0.99**
Video 8	0.83	0.87	0.85	0.87	0.93	0.86	0.86	0.68	**0.96**	**0.96**
Video 9	0.89	0.9	0.87	0.84	0.92	0.87	0.86	0.78	0.88	**0.99**
Video 10	0.89	0.86	0.89	0.91	0.92	0.9	0.94	0.85	0.96	**0.97**
average	0.85	0.86	0.83	0.89	0.88	0.85	0.90	0.77	0.94	**0.97**

novel variation transformation learning framework is proposed to efficiently combine the temporal coherence and distribution information of pixels over a long period of time. Comparisons with other traditional and deep learning methods shows that our DPVTL algorithm significantly improves performance on several challenging scenes.

REFERENCES

[1] B. T, "Traditional and recent approaches in background modeling for foreground detection: An overview," *Comput. Sci. Rev.*, vol. 1112, pp. 31 – 66, 2014.

[2] O. Barnich and M. Van Droogenbroeck, "Vibe: A universal background subtraction algorithm for video sequences," *IEEE Trans. on Image. Process.*, vol. 20, no. 6, pp. 1709–1724, June 2011.

[3] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Conf. on Comput. Vis. and Pattern Recognit.*, vol. 2, 1999, pp. –252 Vol. 2.

[4] D.-S. Lee, "Effective gaussian mixture learning for video background subtraction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 827–832, May 2005.

[5] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," *Eccv*, vol. 1843, pp. 751–767, 2000.

[6] A. Mittal and N. Paragios, "Motion-based background subtraction using adaptive kernel density estimation," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 2, June 2004, pp. II–II.

[7] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, April 2017.

[8] K. Goyal and J. Singhai, "Review of background subtraction methods using gaussian mixture model for video surveillance systems," *Artificial Intelligence Review*, vol. 50, no. 2, pp. 241–259, Aug 2018.

[9] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Proc. Int. Conf. on Pattern Recognit. (ICPR)*, vol. 2, 2004.

[10] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, "Real-time foreground-background segmentation using codebook model," *Real-Time Imaging*, vol. 11, pp. 172–185, 2005.

[11] M. Hofmann, P. Tiefenbacher, and G. Rigoll, "Background segmentation with feedback: The pixel-based adaptive segmenter," in *Computer Vision and Pattern Recognition Workshops*, 2012, pp. 38–43.

[12] S. Varadarajan, P. Miller, and H. Zhou, "Region-based mixture of gaussians modelling for foreground detection in dynamic scenes," *Pattern Recognit.*, vol. 48, no. 11, pp. 3488 – 3503, 2015.

[13] M. Chen, X. Wei, Q. Yang, Q. Li, G. Wang, and M. H. Yang, "Spatiotemporal gmm for background subtraction with superpixel hierarchy," *IEEE Trans. on Pattern Anal. and Mach. Intell.*, vol. PP, no. 99, pp. 1–1, 2018.

[14] Y. Sheikh and M. Shah, "Bayesian modeling of dynamic scenes for object detection." *IEEE Trans Pattern Anal Mach Intell*, vol. 27, no. 11, pp. 1778–1792, 2005.

[15] V. Reddy, C. Sanderson, and B. C. Lovell, "Robust foreground object segmentation via adaptive region-based background modelling," in *International Conference on Pattern Recognition*, 2010, pp. 3939–3942.

[16] M. Izadi and P. Saeedi, "Robust region-based background subtraction and shadow removing using color and gradient information," pp. 1–5, 01 2008.

[17] L. Cheng, M. Gong, D. Schuurmans, and T. Caelli, "Real-time discriminative background subtraction," *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1401–1414, May 2011.

[18] B. Han and L. Davis, "Density-based multifeature background subtraction with support vector machine," *IEEE Trans. on Pattern Anal. and Mach. Intell.*, vol. 34, no. 5, pp. 1017–1023, May 2012.

[19] P. Heas, C. Herzet, and E. Memin, "Bayesian inference of models and hyperparameters for robust optical-flow estimation," *IEEE Trans. on Image. Process.*, vol. 21, no. 4, pp. 1437–1451, April 2012.

[20] X. Zhang, Z. Liu, H. Li, X. Zhao, and P. Zhang, "Statistical background subtraction based on imbalanced learning," in *2014 IEEE International Conference on Multimedia and Expo (ICME)*, vol. 00, July 2014, pp. 1–6.

[21] H. Mobahi, R. Collobert, and J. Weston, "Deep learning from temporal coherence in video," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML '09, 2009, pp. 737–744.

[22] M. Lngkvist, L. Karlsson, and A. Loutfi, "A review of unsupervised feature learning and deep learning for time-series modeling," *Pattern Recognition Letters*, vol. 42, pp. 11 – 24, 2014.

[23] Y. Wang, Z. Luo, and P.-M. Jodoin, "Interactive deep learning method for segmenting moving objects," 09 2016.

[24] M. Braham and M. Droogenbroeck, "Deep background subtraction with scene-specific convolutional neural networks," pp. 1–4, 05 2016.

[25] "A deep convolutional neural network for video sequence background subtraction," *Pattern Recognition*, vol. 76, pp. 635 – 649, 2018.

[26] P. L. St-Charles, G. A. Bilodeau, and R. Bergevin, "Subsense: A universal change detection method with local adaptive sensitivity," *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 359–373, Jan 2015.

[27] R. Wang, F. Bunyak, G. Seetharaman, and K. Palaniappan, "Static and moving object detection using flux tensor with split gaussian models," in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, June 2014, pp. 420–424.

[28] L. Yang, J. Li, Y. Luo, Y. Zhao, H. Cheng, and J. Li, "Deep background modeling using fully convolutional network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 254–262, Jan 2018.

[29] Y. Wang, P. M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar, "Cdnet 2014: An expanded change detection benchmark dataset," in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, June 2014, pp. 393–400.

[30] S. Li, D. Florencio, W. Li, Y. Zhao, and C. Cook, "A fusion framework for camouflaged moving foreground detection in the wavelet domain," *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3918–3930, Aug 2018.

[31] S. Bianco, G. Ciocca, and R. Schettini, "Combination of video change detection algorithms by genetic programming," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 6, pp. 914–928, Dec 2017.

[32] S. Jiang and X. Lu, "Wesambe: A weight-sample-based method for background subtraction," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2017.

[33] Y. Chen, J. Wang, and H. Lu, "Learning sharable models for robust background subtraction," in *2015 IEEE International Conference on Multimedia and Expo (ICME)*, June 2015, pp. 1–6.

[34] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, "A self-adjusting approach to change detection based on background word consensus," in *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*, Jan 2015, pp. 990–997.

[35] M. Braham, S. Pierard, and M. V. Droogenbroeck, "Semantic background subtraction," in *2017 IEEE International Conference on Image Processing (ICIP)*, Sept 2017, pp. 4552–4556.

[36] G. Ramłrez-Alonso and M. I. Chacn-Murguła, "Auto-adaptive parallel som architecture with a modular analysis for dynamic object segmentation in videos," *Neurocomputing*, vol. 175, pp. 990 – 1000, 2016.

[37] M. D. Gregorio and M. Giordano, "Change detection with weightless neural networks," in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, June 2014, pp. 409–413.

[38] H. Sajid and S.-C. S. Cheung, "Universal multimode background subtraction," *Submitted to Image Processing, IEEE Transactions on*, 2015.

[39] S. Jiang and X. Lu, "Wesambe: A weight-sample-based method for background subtraction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. PP, no. 99, pp. 1–1, 2017.

[40] Y. Chen, J. Wang, and H. Lu, "Learning sharable models for robust background subtraction," in *Multimedia and Expo (ICME), 2015 IEEE International Conference on*, June 2015, pp. 1–6.

[41] S. Varadarajan, P. Miller, and H. Zhou, "Spatial mixture of Gaussians for dynamic background modelling," in *Proc. IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2013, pp. 63–68.

[42] M. Wu and X. Peng, "Spatio-temporal context for codebook-based dynamic background subtraction," *AEU - International Journal of Electronics and Communications*, vol. 64, no. 8, pp. 739 – 747, 2010.

[43] L. Yang, H. Cheng, J. Su, and X. Li, "Pixel-to-model distance for robust background reconstruction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 5, pp. 903–916, May 2016.

[44] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern Recognition Letters*, vol. 27, no. 7, pp. 773 – 780, 2006.

[45] F. E. Baf, T. Bouwmans, and B. Vachon, "Fuzzy integral for moving object detection," in *2008 IEEE International Conference on Fuzzy Systems (IEEE World Congress on Computational Intelligence)*, June 2008, pp. 1729–1736.

[46] L. Maddalena and A. Petrosino, "A self-organizing approach to background subtraction for visual surveillance applications," *IEEE Transactions on Image Processing*, vol. 17, no. 7, pp. 1168–1177, July 2008.

[47] J. Yao and J. M. Odobez, "Multi-layer background subtraction based on color and texture," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–8.

[48] X. Zhou, C. Yang, and W. Yu, "Moving object detection by detecting contiguous outliers in the low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 3, pp. 597–610, March 2013.

[49] M. Shakeri and H. Zhang, "Corola: A sequential solution to moving object detection using low-rank approximation," *Computer Vision and Image Understanding*, vol. 146, pp. 27 – 39, 2016.

[50] S. Li, D. Florencio, W. Li, Y. Zhao, and C. Cook, "A fusion framework for camouflaged moving foreground detection in the wavelet domain," *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3918–3930, Aug 2018.