BACKGROUND SUBTRACTION BASED ON DEEP PIXEL DISTRIBUTION LEARNING

Paper ID 1224

ABSTRACT

Previous approaches to background subtraction typically address the problem by formulating a representation of the background, and comparing the background to new frames. In this work, we focus on the essence of background subtraction, which is the classification of a pixel's current observation in comparison to historical observations, and propose a Deep Pixel Distribution Learning (DPDL) model for background subtraction. In the DPDL model, a novel pixel-based feature, called the Random Permutation of Temporal Pixels (RPoTP), is used to represent the distribution of past observations for a particular pixel, in which the temporal correlation between observations is deliberately obfuscated. Subsequently a convolutional neural network (CNN) is used to learn the distribution for determining whether the current observation is foreground or background, with the random permutation enabling the framework to focus primarily on the distribution of observations, rather than be misled by learning spurious temporal correlations. In addition, the pixel-wise representation allows for a large number of RPoTP features to be captured even with a limited number of groundtruth frames, with the DPDL model being effective even with only a single groundtruth frame. The proposed framework is able to achieve promising results in diverse natural scenes, and a comprehensive evaluation on standard benchmarks demonstrates the superiority of our work to state-of-the-art methods.

***Index Terms*—** Background subtraction, motion detection, deep learning, pixel distribution, random permutation

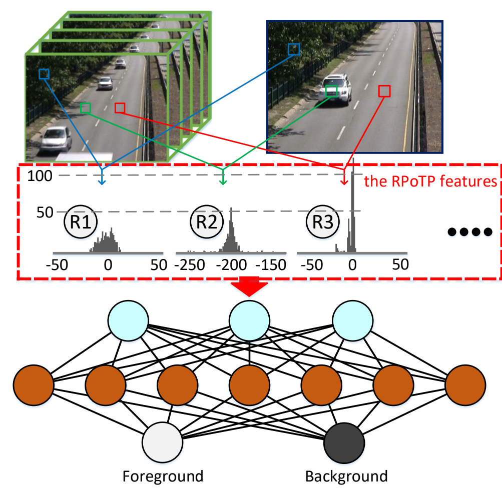

Fig. 1. The illustration of deep pixel distribution learning model. The RPoTP features encode the distributions of pixel observations that belong to dynamical background R1, moving objects R2 and static background R3 respectively. The RPoTP features from all pixels are then fed into the convolutional neural network to learn a classifier for background subtraction.

1. INTRODUCTION

Background subtraction is a fundamental problem of computer vision [1]. Existing algorithms have already achieved good performance in scenes with low complexity, such as indoor scenes [2]. However, there is still substantial scope for improvement when applying such algorithms to wider datasets with greater scene variability. Traditionally, background subtraction methods are hand-crafted based on the specific characteristics of the scenes to which they are applied, such as the multi-background models were applied to dynamic background [3], These domain-tailored models typically do not generalize very well to scenes with other characteristics. In addition, methods that depend on the availability of a large number of groundtruth frames are compromised in situations in which groundtruth annotation may not be obtained easily. It thus remains a challenge in developing a robust and effective background subtraction algorithm that can be applied widely without manual adaptation.

Background subtraction is essentially a classification of pixels in a sequence of image frames, typically under the assumption of a static camera [4], wherein each pixel in a particular frame is classified as foreground or background by comparing its current measurement with historical observations. When dealing with diverse and complex scenes, this is a difficult problem as the pixel measurements can take the form of complex distributions for both foreground and background pixels. For example, as shown in the R1, R2 and R3 of Fig. 1, the distributions of the pixel values in the pixels belonging to a dynamical background, moving objects and a static background are completely different, and manually-tailored models have limited ability to cope with these contrasting distributions. To solve this problem, we propose a new deep pixel distribution learning (DPDL) model for background subtraction, which learns to automatically adapt to the disparate distributions encountered in different scenes. An important component of this model is the proposed random permutation of temporal pixels (RPoTP) feature for encoding the distributions of pixel observations, which are fed into our deep network to learn the optimal classifier for background subtraction, as shown in Fig. 1.

In our DPDL model, each RPoTP feature comprises a randomly permuted vector of past observations for a particular pixel, subtracted from the current pixel observation. The random permutation is to prevent the framework from learning spurious temporal correlations between observations when the number of training image frames is insufficient, and which also hinders the use of a common background subtraction model for different pixel positions. Through random permutation, the deep network is indirectly forced to rely solely on the statistics of the pixel intensity distribution for accurate classification.

The contributions of this paper are:

1. We propose a new feature, the *random permutation of temporal pixels* (RPoTP) feature, to encode the pixel intensity distribution in time, to be used as input to our subsequent model. The RPoTP feature is a randomly permuted vector of a pixel's historical observations, subtracted from the current observation. Since each feature depends only on a single pixel, a large number of features may be obtained for training, even with limited groundtruth frames.

2. We further propose a *deep pixel distribution learning model* (DPDL) for background subtraction that works for a diverse range of scenes. The DPDL model employs a deep convolutional neural network to learn the most discriminative representation of temporal intensity statistics of individual pixels for background subtraction. In particular, the success of the model suggests that a hierarchical representation of sample statistics is more effective than the direct representation as an intensity distribution. The DPDL model is able to achieve good performance in a large range of scenes in the dataset, without the requirement to manually tune for different scene categories. Our DPDL model is also able to segment moving objects effectively, even when depending only on one groundtruth frame.

2. RELATED WORK

Traditionally background subtraction is done by modeling the variation of pixel intensities over time (e.g. [3–10]). In particular, the Gaussian mixture model [3] is one of the more popular techniques for background subtraction [11], and numerous extensions of GMM have been proposed. For example, Varadarajan et al. [5] modeled regions rather than pixels using a Gaussian distribution to represent neighbouring relationships. Sriram et al. [12] extend this through the use of expectation maximization. In addition, there are also several algorithms that utilize kernel density estimation (e.g. [13]) as a substitute for Gaussians. Recently, Haines et al. [6] proposed the use of Dirichlet processes with Gaussian mixture models to analyze pixel distribution, while Chen et al. [14] used Gaussian mixture models representing the vertices of spanning trees.

Besides traditional algorithms, there are also several background subtraction methods that utilize machine learning, commonly involving support vector machines (SVM) and Bayesian methods. Han et al. [15] utilized density-based features as the input to an SVM classifier. Zhang et al. [16] proposed an imbalance compensation mechanism for use with bilayer modeling and Bayesian classification. The novel subspaces learning method proposed by Zhou et al. [17], in which a sequence of regular video bricks is extracted, poses the background modeling problem as pursuing a subspace representation for video bricks while adapting to scene variation.

Unsurprisingly, recent background subtraction methods have embraced deep learning. Wang et al. [18] used a cross-entropy loss function for training a multi-scale CNN to learn the background of a current scene, using the full image as input. Braham et al. [19] employed a deep network to evaluate the difference between current frames and a background image, the latter being computed as the temporal median across multiple frames. In both these works, a large number of frames is needed in the training phase to achieve accurate segmentation of moving objects. A more robust background model algorithm was proposed by Babaee et al. [20] for background extraction, with a network used for subtracting the background from the current image.

In contrast, our proposed method departs from the conventional approach of creating an explicit representation of the background scene, which we expect will face considerable difficulty in modeling a dynamic and complex background well. Instead we focus on the more fundamental concept underlying background subtraction, which is the classification of pixels in a time sequence [4], based on network-learned discriminative features that act directly on the distribution of pixels across time.

3. DEEP PIXEL DISTRIBUTION LEARNING

In this section, the details of the DPDL model are explained, including the process of capturing RPoTP features and the architecture of the DPDL model.

3.1. Random Permutation of Temporal Pixels

In videos of natural scenes, there are multiple factors that contribute to the variation of pixel values over time, such as changes in illumination, dynamic backgrounds and moving objects. These variations mean that each pixel is not simply represented as a single constant color, but rather a temporal distribution of colors. More importantly, the distribution

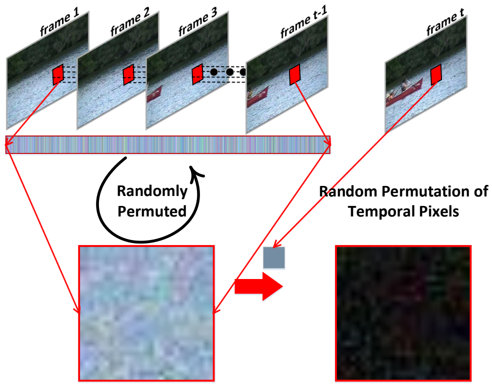

Fig. 2. The process of capturing Random Permutation of temporal feature in a particular pixel. The historical observations of pixels are permuted by a random permutation, and reshape into a square matrix. Then the RPoTP feature is captured from the subtraction between the matrix and the current observation of pixel.

of colors is different depending on the cause of the change. For example, the color variation that results from illumination changes due to changing positions of the sun typically follow a smooth unimodal distribution, similar to a Gaussian. Conversely, for dynamic backgrounds with cyclical repetition of patterns, such as leaves fluttering in the breeze, the variation often takes the form of a multi-peak distribution, due to the switching type behavior of these patterns (e.g. leaf vs no leaf at the pixel). However, when the color variation is caused by moving foreground objects, the pixel values are typically outliers compared to the distribution formed by past background observations.

In a background subtraction task, we want to discriminate distributions arising from foreground objects from those due to background change. The Random Permutation of Temporal Pixels (RPoTP) feature is proposed to do this effectively.

The process of extracting the RPoTP feature for a particular pixel is shown in the Fig. 2. Starting from an image sequence captured in a static camera, the temporal observations for a particular pixel are obtained as a list. The sequence is then randomly permuted and reshaped into a square matrix, which may be thought of to represent a fair sample set of past observations. Finally, the RPoTP feature is formed by subtracting the current pixel value from the sample matrix, in order to more explicitly describe differences from past observations. The RPoTP feature will eventually be used as input into a deep network that will classify the feature as foreground or background.

One key aspect of the RPoTP feature is the randomized permutation. Unless the training dataset is extremely large, a deep network with an extensive number of weights will learn spurious temporal correlations from the training data if the data is presented in the original temporal sequence. By randomly permuting the temporal arrangement of the pixel observations, the network is forced to learn discriminative weights that are focused on the pixel value distribution itself and relevant statistics, rather than on their temporal sequence. This is somewhat akin to the use of dropout layers in deep networks to prevent overfitting, but instead of forcing random input nodes to be zero, random swapping occurs instead. The random permutation enables the learned network to be much more generalizable to unseen test data.

While the procedure of extracting the RPoTP feature was presented only for a single pixel, it applies identically to all pixels. Mathematically we denote the sequence of past observations to be used for computing the RPoTP feature for a single pixel as $\{I_1^s, I_2^s, \dots I_{r^2}^s\}$, where the temporal window is of length r^2, as these observations will be reshaped into a $r \times r$ matrix. The subscript is in chronological order, with the current pixel value being I_{r^2+1}. Next, the sampled observations are rearranged according to a random permutation $\mathcal{X} = \{a_1, a_2, \dots, a_{r^2}\}$. The RPoTP feature may then be computed in the form a square matrix as:

$$RP^r(\lfloor \frac{a}{r} \rfloor, a \bmod r) = I_{r^2+1} - I^s_{\mathcal{X}(a)}. \tag{1}$$

In practical scenarios, a substantial labor cost may be incurred if many groundtruth frames are needed for training a background subtraction system. However, since the proposed RPoTP features are formed from individual pixels, a large number of features may be extracted from only a single groundtruth frame. Therefore although our DPDL method, as described in the next section, is based on a deep learning methodology which typically requires an extensive training dataset, we only require very limited groundtruth frames to produce strong results, and is effective even with only one groundtruth frame.

3.2. Network Architecture

As stated earlier, we consider background subtraction as fundamentally a classification problem, to determine if current pixels are foreground or background based on past observations. In our DPDL model, a RPoTP feature is extracted for each pixel and used as input to a convolutional neural network, which acts as a classifier labeling the pixel as either foreground or background.

Our DPDL model is divided into three blocks as shown in Fig. 3. The first part involves extracting RPoTP features as the input to the network, while the second and the third parts form the learning $\mathcal{L}$ and decision $\mathcal{D}$ parts respectively in the network architecture. Mathematically the overall classification may be expressed as:

$$\ell_{x,y} = D(\mathcal{L}^{h(\theta,r)}(RP^r_{x,y})), \tag{2}$$

where $\ell_{x,y}$ is binary label of the pixel at location (x, y) identifying it as foreground or background, $\mathcal{L}$ is the learning block and $\mathcal{D}$ is the decision block.

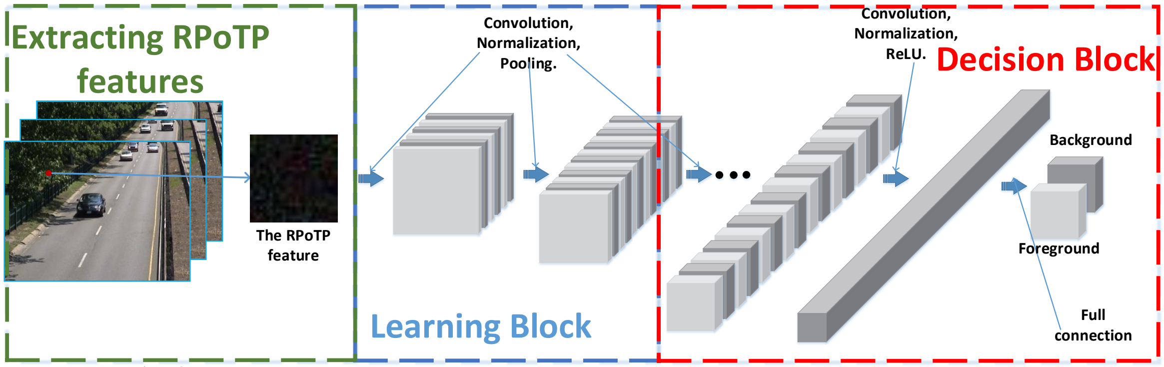

Fig. 3. The framework of the DPDL model as well as the architecture of the network.

The learning block $\mathcal{L}$ consists of several stacks of convolutional, normalization and pooling layers, with $h(\theta, r)$ determining the number of stacks based on r and various hyperparameters θ of the network according to:

$$h(r, \theta) = \frac{r - N_D}{N_{LC} + N_{LP} - 2}, \quad \theta = \{N_D, N_{LC}, N_{LP}\} \quad (3)$$

where $N_{LC} \times N_{LC}$ and $N_{LP} \times N_{LP}$ are respectively the sizes of kernels used in the convolutional layers and pooling layers of the learning block, while $N_D \times N_D$ is the size of the input to the decision block. The decision block $\mathcal{D}$ is a set of convolutional, normalization and ReLU layers, followed by a fully connected layer that implements the softmax operator. During training, logistic loss on the final output node is minimized.

4. EXPERIMENTS

4.1. Evaluation on the CDNet dataset

In this section, the performances of our DPDL model are evaluated for diverse natural scenes using CDnet2014 [25] dataset, which are currently the most extensive datasets available for background subtraction benchmarking. In the experiments, the Re (Recall), Pr (Precision) and Fm (F-measure) metrics are used for evaluation.

Since the DPDL model is effective with limited (or only one) groundtruth training frames as previously mentioned, some cases may arise in which the groundtruth training frames do not include any foreground moving objects, which means that there would not be any training data with foreground labels. To solve this problem, the training frames are extracted from two parts of the full dataset. In order to get enough foreground labels for training, one half of the training dataset comprises frames with moving objects, with the remainder sampled from whole video sequences. These two halves of the training dataset are mixed prior to training the DPDL model. Note that this improvisation is only relevant for evaluating the DPDL model on the benchmark datasets, as during practical use we would expect the user to select appropriate frames that contain at least one moving object for manual labeling, prior to training.

The quantitative evaluation results for the DPDL model and other methods on the CDnet 2014 are shown in Table 1. Due to the length of paper, more results are available on the supplementary materials. All results for prior work were obtained from corresponding publications by the authors. Different results were obtained for the DPDL model by using 1, 20 and 40 groundtruth frames, and are labeled as DPDL_1, DPDL_{20} and DPDL_{40} respectively. It should be noted that 40 training groundtruth frames is a tiny fraction of available groundtruth frames in the dataset, with each video containing thousands of groundtruth frames. We also present results in Table 2 demonstrating the effectiveness of DPDL_1, the DPDL model using only one groundtruth frame, on several videos sampled from the CDnet dataset.

As shown in the Table 1, the DPDL model performed strongly on both datasets. In particular, DPDL_{40} achieved the highest results for the Fm metric. Promising results were also obtained for DPDL_{20}, when the number of groundtruth training frames were reduced by half. We consider some reasons why the DPDL model outperformed existing methods. A number of these methods involved simplistic pixel distribution models or hand-crafted features, and as a result these models do not have sufficient complexity or accuracy to discriminate foreground objects from background imagery for natural scenes, when compared to the DPDL model. Others methods, such as DeepBS [20], are based on the concept of a background image or model. This differs from our approach of focusing solely on the mechanism of foreground-background classification, rather than requiring coherent models of a background or a foreground. Therefore, although DeepBS [20] is based on a deep learning methodology, its performance is still limited by the requirement of a background image.

Even in the extreme case of using only one groundtruth frame for training, the DPDL model was still effective in comparison with several existing algorithms such as GMM [3], RMoG [5], and AAPSA [7], as demonstrated in Table 2. The reason why this was possible was that the features of the different moving objects in these sample videos were similar, and consequently so were the distributions of pixel observations. For example, cars were the dominant moving objects in the highway video in the Baseline category. The features of the cars, such as color and texture, had limited variability, and thus the distribution of pixel values were similar as well. In

Table 1. Quantitative evalution of proposed approach using Fm metric in CDnet2014 dataset.

Approach	Baseline	Dyn. Bg.	Cam. Jitt.	Int. Mot.	Shadow	Ther.	Bad Wea.	Low Fr.	Nig. Vid.	PTZ	Turbul.	Overall
DeepBS [20]	0.9580	0.8761	**0.8990**	0.6098	0.9304	0.7583	0.8301	0.6002	0.5835	0.3133	**0.8455**	0.7458
AAPSA [7]	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 [21]	0.9145	0.8274	0.7886	0.5753	0.8581	0.7866	0.6837	0.6406	0.3735	0.3218	0.7227	0.6812
MBS [22]	0.9287	0.7915	0.8367	0.7568	0.7968	0.8194	0.7980	0.6350	0.5158	0.5520	0.5858	0.7288
PAWCS [23]	0.9397	**0.8938**	0.8137	0.7764	0.8913	0.8324	0.8152	0.6588	0.4152	0.4615	0.6450	0.7403
ShareM [8]	0.9522	0.8222	0.8141	0.6727	0.8898	0.8319	0.8480	**0.7286**	0.5419	0.3860	0.7339	0.7474
SuBSENSE [9]	0.9503	0.8177	0.8152	0.6569	0.8986	0.8171	0.8619	0.6445	0.5599	0.3476	0.7792	0.7408
WeSamBE [24]	0.9413	0.7440	0.7976	0.7392	0.8999	0.7962	0.8608	0.6602	0.5929	0.3844	0.7737	0.7446
GMM [3]	0.8245	0.6330	0.5969	0.5207	0.7370	0.6621	0.7380	0.5373	0.4097	0.1522	0.4663	0.5707
RMoG [5]	0.7848	0.7352	0.7010	0.5431	0.7212	0.4788	0.6826	0.5312	0.4265	0.2470	0.4578	0.5735
$DPDL_1$	0.7886	0.6566	0.5456	0.5115	0.6957	0.6697	0.6036	0.5966	0.3953	0.2942	0.6301	0.5807
$DPDL_{20}$	0.9620	0.8369	0.8627	0.8174	0.8763	0.8311	0.8107	0.6646	0.5866	0.4654	0.7173	0.7665
$DPDL_{40}$	**0.9692**	0.8692	0.8661	**0.8759**	**0.9361**	**0.8379**	**0.8688**	0.7078	**0.6110**	**0.6087**	0.7636	**0.8106**

Table 2. Quantitative evaluation of $DPDL_1$, which is trained with single a groundtruth frame, using Re, Pr, Fm metric in several videos of the CDnet dataset.

Category	Videos	(Re,Pr,Fm)
Baseline	highway	(0.9213, 0.9444, 0.9327)
	PETS2006	(0.8577, 0.8863, 0.8718)
Bad Weath.	blizzard	(0.7614, 0.9244, 0.8350)
	skating	(0.9257, 0.9078, 0.9167)
Cam.Jitt.	badminton	(0.9131, 0.8348, 0.8722)
	traffic	(0.8091, 0.6471, 0.7191)
Dyn. Bg	fountain02	(0.7076, 0.9584, 0.8142)
	canoe	(0.9172, 0.9684, 0.9421)
Int. Mot.	abandonedBox	(0.8063, 0.6515, 0.7207)
	streetLight	(0.7870, 0.9935, 0.8783)
Low Fr.	tramCrossroad	(0.8488, 0.9628, 0.9022)
	turnpike	(0.7888, 0.8220, 0.8050)
Shadow	busStation	(0.7316, 0.9464, 0.8253)
	copyMachine	(0.8633, 0.8780, 0.8706)
	peopleInShade	(0.8023, 0.9575, 0.8730)
Thermal	corridor	(0.8172, 0.7946, 0.8057)
	library	(0.6076, 0.9977, 0.7553)
Turbul.	turbulence2	(0.7234, 0.9817, 0.8330)

this situation, one frame had sufficient foreground and background samples to capture the appropriate pixel distributions, especially since each pixel represented a separate training instance for the network, and there were many pixels in a single groundtruth frame. Some other examples in Table 2 had similar characteristics of consistent foreground objects, which explains why $DPDL_1$ performed effectively for these videos. In contrast, in the traffic video of the Camera Jitter category, there were small variations in pixel positions due to random camera jitter. This meant that the pixel distribution captured from a single groundtruth frame would not generalize well to other frames, since it was only representative of a single instance of jitter displacement. This led to relatively poorer performance of $DPDL_1$ on this video. However when the number of training frames was increased, the DPDL model had sufficient information for generalizing, which therefore resulted in an improved performance.

There are also some limitations in the DPDL model. As shown in the Table 2, although the scores of final Fm metric is high, there is an obvious difference between the scores of Re and Pr. This situation arises from a large imbalance in number of labeled foregrond and background pixels. In addition, as mentioned in the Section 3.2, the depth of the network is dependent on the temporal window size r used. Therefore when a large r is used, the network will require a substantially longer amount of time for training, especially since there is under-sampling is not used within the network.

5. CONCLUSION

In this paper, we proposed the DPDL method for effective background subtraction in diverse natural scenes. Unlike previous work that compared a background model with new frames, we recast background subtraction as a classification of the current pixel value as compared to past observations. In our DPDL model, the convolutional neural network (CNN) is utilized to learn the distribution of past pixel values for classifying the current observation self-adaptively. In particular, the Random Permutation of Temporal Pixels (RPoTP) features are proposed to represent the distribution of the past observations, in which the temporal correlations among past observations are deliberately obfuscated to force the network to focus on the distribution. Since the hierarchical representation in CNN better describes complex distributions than direct representation, the proposed approach performed well in diverse natural scenes. Moreover, due to the pixel-wise representation of RPoTP features, a large number of RPoTP features can be captured with only a limited number of groundtruth frames, and our DPDL method is effective even with only a single groundtruth frame.

6. REFERENCES

[1] 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. Mach. Intell.*, , no. 99, 2017.

[2] Thierry Bouwmans, "Traditional and recent approaches in background modeling for foreground detection: An overview," *Computer Science Review*, 2014.

[3] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Proc. Int. Conf. Pattern Recog. (ICPR)*, 2004.

[4] O. Barnich and M. Van Droogenbroeck, "Vibe: A universal background subtraction algorithm for video sequences," *IEEE Trans. on Image. Process.*, 2011.

[5] Sriram Varadarajan, Paul Miller, and Huiyu Zhou, "Spatial mixture of Gaussians for dynamic background modelling," in *Proc. IEEE Int. Conf. Advanced Video and Signal Based Surveillance*, 2013.

[6] T. S. F. Haines and T. Xiang, "Background subtraction with dirichletprocess mixture models," *IEEE Trans. on Pattern Anal. Mach. Intell.*, , no. 4, 2014.

[7] Graciela Ramrez-Alonso and Mario I. Chacn-Murgua, "Auto-adaptive parallel som architecture with a modular analysis for dynamic object segmentation in videos," *Neurocomputing*, 2016.

[8] Yingying Chen, Jinqiao Wang, and Hanqing Lu, "Learning sharable models for robust background subtraction," in *IEEE Int. Conf. Multimedia and Expo (ICME)*, 2015.

[9] P. L. St-Charles, G. A. Bilodeau, and R. Bergevin, "Subsense: A universal change detection method with local adaptive sensitivity," *IEEE Trans. on Image. Process.*, , no. 1, 2015.

[10] X. Liu, J. Yao, X. HONG, X. Huang, Z. Zhou, C. Qi, and G. Zhao, "Background subtraction using spatio-temporal group sparsity recovery," *IEEE Trans. on Circuits Syst. Video Technol.*, , no. 99, 2017.

[11] Kalpana Goyal and Jyoti Singhai, "Review of background subtraction methods using gaussian mixture model for video surveillance systems," *Artificial Intelligence Review*, 2017.

[12] Sriram Varadarajan, Paul Miller, and Huiyu Zhou, "Region-based mixture of gaussians modelling for foreground detection in dynamic scenes," *Pattern Recog.*, , no. 11, 2015.

[13] Z. Liu, K. Huang, and T. Tan, "Foreground object detection using top-down information based on em framework," *IEEE Trans. on Image. Process.*, , no. 9, 2012.

[14] 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. Mach. Intell.*, , no. 99, 2017.

[15] Bohyung Han and L.S. Davis, "Density-based multifeature background subtraction with support vector machine," *IEEE Trans. on Pattern Anal. Mach. Intell.*, 2012.

[16] X. Zhang, Z. Liu, H. Li, X. Zhao, and P. Zhang, "Statistical background subtraction based on imbalanced learning," in *IEEE Int. Conf. Multimedia and Expo (ICME)*, 2014.

[17] X. Zhou, C. Yang, and W. Yu, "Moving object detection by detecting contiguous outliers in the low-rank representation," *IEEE Trans. on Pattern Anal. Mach. Intell.*, , no. 3, 2013.

[18] Yi Wang, Zhiming Luo, and Pierre-Marc Jodoin, "Interactive deep learning method for segmenting moving objects," *Pattern Recog. Letters*, , no. Supplement C, 2016, Scene Background Modeling and Initialization.

[19] M. Braham and M. Van Droogenbroeck, "Deep background subtraction with scene-specific convolutional neural networks," in *Int. Conf. Systems, Signals and Image Processing (IWSSIP)*, 2016.

[20] M. Babaee, D. T. Dinh, and G. Rigoll, "A Deep Convolutional Neural Network for Background Subtraction," *ArXiv e-prints*, 2017.

[21] M. D. Gregorio and M. Giordano, "Change detection with weightless neural networks," in *IEEE Conf. Comput. Vis. Pattern Recog. Workshops*, 2014.

[22] H. Sajid and S. C. S. Cheung, "Universal multimode background subtraction," *IEEE Trans. on Image. Process.*, , no. 7, 2017.

[23] P. L. St-Charles, G. A. Bilodeau, and R. Bergevin, "Universal background subtraction using word consensus models," *IEEE Trans. on Image. Process.*, , no. 10, 2016.

[24] S. Jiang and X. Lu, "Wesambe: A weight-sample-based method for background subtraction," *IEEE Trans. on Circuits Syst. Video Technol.*, , no. 99, 2017.

[25] Y. Wang, P. M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar, "Cdnet 2014: An expanded change detection benchmark dataset," in *IEEE Conf. Comput. Vis. Pattern Recog. Workshops*, 2014.