Background Subtraction based on Deep Feature Transformation Learning Background Subtraction

Xinyu Ren, Chenqiu Zhao, Yongxin Ge

***Abstract*—Background subtraction is a fundamental problem of computer vision. The main challenges of background subtraction comes from the diversity and the complexity of natural scenes. Plenty of previous works are proposed to handle this problem by artificial model. In this paper, we propose a self-adaption solution with the utilization of convolutional neural networks, and a novel background subtraction model named Deep Feature Transformation Learning is proposed. With the motivation of finding a new representation of pixels historical observations in a new feature space. The videos are divided into fixed length image blocks, which are later transformed into the pixels observation patches as the input of a FCN for learning the variation of pixels. We conduct our training and prediction processes on block level. The architecture of the FCN is devised from the semantic segmentation problem. Comparisons with several state-of-the-art algorithms in well-known benchmark show the superiority of proposed approach.**

***Index Terms*—Background Subtraction, Freely Moving Camera, Foreground and Background Cues,**

I. Introduction

Background subtraction or foreground object detection, as a fundamental problem in computer vision, has been much discussed with the increasing number of outdoor cameras over the last few decades. It is widely used as the pre-processing step of video processing, which can help us efficiently mark the region of interest, e. vehicles and humans, thus saving us huge amount of computing resources. Typically, Background subtraction is a binary classification task that assigns each pixel in a video sequence with a label, for either belonging to the background or foreground scene.

With a huge number of works, existed algorithms have already achieved well performances in the scenes of low diversity or complexity, such as the indoor scenes. However, background Subtraction is still unsolved because of the diversity in background scenes and the changes originated from the camera itself. Scene variations can be in many forms such as, camera jitter, dynamic background, bad weather, illumination changes, intermittent object motion. This is principally because the major methods, in most cases, try to find a universal solution by generating some universal background models for later linear classification; for instance, the earlier Frame Difference Methods, trying to get a fixed image as background to classify each pixels. The fundamental problem lies in that linear classifier might not be powerful enough for the job, due to the complexity and the diversity of natural scenes.

In order to present a better and universal solution, we proposed the Deep Pixels Variation Learning (DPVL) model for Background Subtraction in diversely natural scenes. In our method, the main job is to transform the sequence of pixels into another sequence which is easy for classification.

$$sq_1 = \{p_1, p_2, \cdots, p_n\} -> sq_2 = \{f_1, f_2, \cdots, f_n\} \qquad (1)$$

For sq_1, it is a sequence of pixels' intensity, which is hard to classify which entry is foreground or background. But in sq_2, it is more easy to classified, since the output of

In this paper, pixels observation patches are proposed to describe the historical observation of pixels. And a random initialized FCN network is trained to learn the historical observation of pixels and generate a new representation in a new feature space. We take advantage of the strong learning ability of FCN to learn a new representation of the Pixels observation patches in a new space where the pixels can be easily classified to background and foreground.

II. Related Work

Over the last few decades, Background subtraction has been well studied. Meanwhile, a huge number of methods were proposed. These methods can be broadly categorized into pixels-based, region-based, frame-based and Deep Learning.

A. A. pixels-based methods

The most widely used algorithms in Background subtraction are pixels-based methods. And one of the famous method is Gaussian Mixture Model (GMM), in which a GMM is used to model the history over time of pixels intensity values. It is assumed that pixels are independent from their neighbors. Incoming pixels are labeled as background if there exists a Gaussian in the GMM, where the distance between its mean and the pixel lies within a certain bound. For learning the parameters, that maximize the likelihood, the authors proposed an online method that approximates the Expectation Maximization (EM) algorithm.

In XXX , Mingliang Chen et al. propose a background subtraction algorithm using hierarchical superpixels segmentation, spanning trees and optical flow. Their Background model combine the GMM with constrains of temporal and spatial from optical flow and superpixels.

Kim et al used a codebook to record the sampling background values at each pixel, which can be seemed as a compressed representation of background model. This allows them efficient in memory and speed compared with other background modeling techniques. The final foreground is detected by a distance measurement in a cylindrical color model.

In XXX, Zhi Zeng et al. proposed an equal-qualification updating strategy to replace the maximum-negative-run-length-based filtering strategy. Their experiments show that, the proposed method outperforms well, despite using only color information.

Elgammal et al. introduced a probabilistic non-parametric method to model the background. It is assumed that each background pixel is drawn from a PDF. The PDF for each pixel is estimated with Kernel Density Estimation (KDE).

B. region-based approach

region-based approach assumpt that the neighbouring pixels have a similar variation as the pixel itself.

In xxx, Sriram Varadarajan et al. propose a region-based MoG to takes in which the updated mixtures represent the scene distribution in a neighbourhood region.

In (PCA), classification is done by comparing a block in current frame to its reconstruction from PCA coefficients and declaring it as background if the reconstruction is close.

A recently region based method is presented in XXX which used the statistical circular shift moments (SCSM) in image regions for change detection.

subspace learning method in xxx, is used to compress the background into the eigenbackground. For each video, the mean and the covariance matrix are calculated. After a PCA of the covariance matrix, a projection matrix is set up with M eigenvectors. Then, incoming images are compared with their projection onto the eigenvectors. Foreground labels are assigned to pixels with large distances, after calculating the distances between the image and the projection and comparing them with the corresponding threshold value.

Marghes et al. used a mixed method that combines a reconstructive method (PCA) with a discriminative one (LDA) to robustly model the background.

single Gaussians are employed for foreground modeling. By computing flux tensors, which depict variations of optical flow within a local 3D spatio-temporal volume, blob motion is detected. With the combination of the different information from blob motion, foreground models and background models, moving and static foreground objects can be spotted. Also, by applying edge matching, static foreground objects can be classified as ghosts or intermittent motions.

Frame-based background modeling via Principal Component Analysis (PCA) and low-rank/sparse decomposition approaches is a popular alternative to pixel-level modeling xx. These approaches are however not ideal for surveillance applications as most rely on batch or offline processing or suffer from scaling problems.

In XXX, ss et al. addressed scaling problems by reformulating principal component analysis for 2D images. Meanwhile their method takes much lower memory consumption and computational cost than others. Some online approaches have also been proposed recently, but they are still very computationally expensive.

C. Algorithms based on Super-pixels

A novel approach for background subtraction with the use of CNN was proposed by Braham and Droogenbroeck. They used a fixed background model, which was generated from a temporal median operation over N video frames.

In XXX, Yi Wang et al. tried a CNN architecture combined with a Cascade model for segmentation in Background subtraction. Given 200 labelled images as training set, their model performed excellently in dataset2014.

Braham et al. present an Deep learning-based method. with the help of CNN, they generated a fixed background model from a temporal median operation over N video frames. Then, a scene-specific CNN is trained with corresponding image patches from the background image.

In xxx, M. Babaee et al. combine the segmentation mask from SuBSENSE algorithm and the output of Flux Tensor algorithm, which can dynamically change the parameters used in the background model based on the motion changes in the video frames. They also used spatial-median filtering as the post processing of the network outputs.

III. The IFB Framework

In this section, we introduce our DPVL model that consists of a pixels-based observation patches and a novel FCN networks for background subtraction. We explain the details of the procedures of capturing the observation patches and the architecture of network in our DPVL model. The complete system is illustrated in Figure XXX. We use a set of input and ground true images to generalize the Pixels Variation Permutations, and reshape them into fixed-size image patches as and fed into a FCN. After reassembling the patches into the complete output frame, it is post-processed, yielding the final segmentation of the respective video frame.

A. A. Image blocks to observation patches

The historical observations of pixels which belong to the background are usually share a common variation pattern. In more specific terms, the variations of pixels in background are generally keeping in some pattern. And we want to use a FCN network to learn a new representation of pixels observations in a new feature space.

In order to get enough pixels observations, we conduct our experiment on block-level. In this paper, we divide each video into smaller image blocks, which share a common frame length of t. For each image blocks, it contains M*N pixels historical observations in a temporal sequence of t frames. We need to take them out for later transformation as our network input.

Observation patch is purposed to take advantage of the strong learning ability of the FCNs. And what we need is an pixel-to-pixel transformation. More precisely, Every historical pixels observation in these observation patches has an corresponding transformation at the last layer of FCNs. Here is how we get these observation patches. First, we transform each image block into M*N pixels observation series which are the single pixels historical observation series in a frame length of t. Next, we reshape the observation sequence into a rectangle image which size is $\sqrt{t} \cdot \sqrt{t}$. That rectangle image is what we called observation patch. The observation patches are taken as network training sample set.

Another conception about observation patches is interval sampling. For a sequence like this:

$$p_t\{p_1, o_2, \cdots, p_n\} - > \{p_1, p_{11}, p_{21}, \cdots, p_n\} \qquad (2)$$

where t denote the sequence number of frames. By this way, we can get an new sequence contains more temporary information than the continuous pixel sequences. This works well when it comes to some situation where the moving objects keep stationary for a long time.

B. Fully Convolutional Network

The primary task of our network is to learn an optimal feature transform of pixels observation series, thus we can find a new representation which is easy for classification from raw RGB data. There needs to be a consistent one-to-one match between pixels observation sequence and our network output on the time series. In these circumstances, traditional convolutional network cannot meet our requirements. Different with those traditional ones, fully convolutional networks can take input of arbitrary size and produce correspondingly-sized output with efficient inference. FCNs have already been used in many areas like sematic segmentation and so on. Researchers found FCNs has a strong learning ability which wont lost to the traditional ones, meanwhile, it also has a high efficient computation ability. FCNs utilize 1*1 kernels to take the place of fully connected layers. Therefore, we take a fixed size of pixels observation patches as the input of our network. For the convenience of calculations, we transform observation sequence into observation patches.

C. Network Architecture

The architectures of our proposed FCN is illustrated in Figure xx, our network contains 5 convolutional layers, 2 pool layers and a convolutional layer which have a filter size of 1*1. A short calculation revealed that the network output will less 10 pixels after forward calculating. In order to make output the same size as input, we borrowed ideas from Image semantic segmentation, which is doing zero-padding before the training. After the forward computing, we can get the new representation of input observation values in a new feature space. Then, after some thresholding calculations. Our experiment results show that a random initialized FCNs, trained end-to-end on feature learning can achieve the state-of-the-art without further machinery.

IV. Experiments

Our experiments are conducted on the computer with Nvidia tasela K20c GPU. And the DeepLearning tool we use is the matconvnet-23. 4 beta.

Our FCN network is random initialized, and the training epoch is set to 20. For each video, we divide it into image blocks, which have a fixed frame length of 144. For each image block, M*N pixels historical observations were taken out, where the M and N denoted the size of images. After a simple transformation, we can get M*N pixels observation patches. Each of them contains the historical observation of a pixels in a frame length of 144.

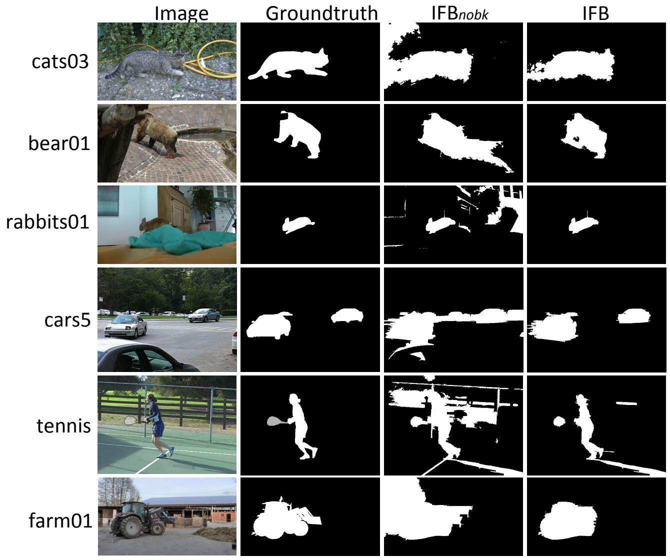

Fig. 1. The example of latex figure.

TABLE I
The example of latex table

Videos	IFB$_{nobk}$			IFB		
	Re	Pr	Fm	Re	Pr	Fm
cats03	(**0.9472** ,	0.5428 ,	0.6901)	(0.9063 ,	**0.7955** ,	**0.8473**)
bear01	(**0.9854** ,	0.4806 ,	0.6461)	(0.8695 ,	**0.8673** ,	**0.8684**)
rabbits01	(**0.9530** ,	0.4433 ,	0.6051)	(0.9510 ,	**0.8730** ,	**0.9103**)
cars5	(**0.9887** ,	0.3693 ,	0.5378)	(0.9437 ,	**0.7108** ,	**0.8109**)
tennis	(**0.9364** ,	0.4558 ,	0.6132)	(0.8808 ,	**0.7345** ,	**0.8010**)
farm01	(**0.9887** ,	0.3897 ,	0.5591)	(0.8954 ,	**0.7462** ,	**0.8140**)
Average	(**0.9666** ,	0.4469 ,	0.6086)	(0.9078 ,	**0.7879** ,	**0.8420**)

For each video, we take only one image block to generate the observation patches as our training sample set. The other image blocks are saved for testing.

In dataset2014, most of the videos have a number of frames over 1000, thus our training data only accounts for 5But we can still get plenty of training data cause the number of the training set based on the size of images. For example, in the video Highway of dataset2014, the image size is 320*240, so the number of observation patches for training is 76800. In our experiment, the finally result would be better if we slightly magnify the observation patches. We do the same to label data, but with no 0 padding.

A. Evaluation Metric and Dataset

The subjective results of background modeling approaches based on the foreground detection binary maps. And the general international standards is F-Measure. F-Measure is defined as:

$$F = \frac{precision\dot{r}ecall}{precision - recall} \qquad (3)$$

V. Conclusions

In this paper, we proposed the IFB framework for background subtraction for the case of a freely moving camera.

Unlike previous work in which attempts were made to improve the accuracy of the estimation of motion, our IFB focuses on integrating rough foreground and background cues for foreground segmentation. In particular, foreground cues are detected by a GMM model with the estimation of background motion, while background cues are captured from spatio-temporal features filtered by homography transformation, where the SURF [1], KAZE [2], SIFT [3] features are used as examples. Then, super-pixels under multiple levels are utilized to integrate these cues. The efficiency of IFB results from the complementarity between foreground and background cues. The accuracy of the proposed approach is improved though the utilization of super-pixels under multiple levels. A comprehensive experiment to compare our results with the state-of-the-art shows the efficiency of our framework and points to its potential for use in practical applications.

References

[1] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Eur. Conf. on Comput. Vis. (ECCV)*, 2006, pp. 404–417.

[2] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "Kaze features," in *Eur. Conf. on Comput. Vis. (ECCV)*, 2012, pp. 214–227.

[3] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.