RCDFNN: ROBUST CHANGE DETECTION BASED ON CONVOLUTIONAL FUSION NEURAL NETWORK

Chunlei Cai[1], *Li Chen*[1*], *Lei Zhou*[2], *Xiaoyun Zhang*[1], *Zhiyong Gao*[1]

[1]Institute of Image Communication and Network Engineering,
Shanghai Jiao Tong University, Shanghai, China
[2]University of shanghai for science and technology, Shanghai, China
[1]{caichunlei,hilichen,xiaoyun.zhang,zhiyong.gao}@sjtu.edu.cn [2]zmbhou@163.com

ABSTRACT

Video change detection, which plays an important role in computer vision, is far from being well resolved due to the complexity of diverse scenes in real world. Most of the current methods are designed based on hand-crafted features and perform well in some certain scenes but may fail on others. This paper puts up forward a deep learning based method to automatically fuse multiple basic detections into an optimal one. Specifically, a convolutional fusion neural network is designed to obtain an adaptive fusion strategy based on features extracted from video content. Limited by the amount of available labeled dataset for change detection, this paper leverages an extractor that well trained on external dataset to improve generalization. Experiments show that the proposed method generates state-of-the-art result compared with nine recent outstanding algorithms and it performs well for diverse scenarios such as dynamic background, camera jitter and night videos.

Index Terms— Change detection, convolutional neural network, video features, background subtraction

1. INTRODUCTION

The detection of changes within video streams usually comes first in the queue of computer vision and video processing, including visual surveillance, video retrieval and smart environments. All the applications require robust change detection algorithms with high precision as a pre-processing step. To achieve a robust solution in practice, many challenges should be addressed such as illumination changes, dynamic background, camera jitter and night videos.

The last decade has witnessed many significant improvements on change detection which is also called background subtraction. These algorithms can be classified into diverse categories such as statistical models [1], cluster models [2], estimation models [3] and sparse models [4]. To the best of our knowledge, although there are numerous work that performs well in some types of videos, there is no single traditional algorithm which can simultaneously address all the key challenges in real-world videos. This is because most traditional algorithms depend on feature engineering [5] and parameter tuning [6]. Each of them was developed in several different contexts under different challenges.

Recently, a few convolutional neural network (CNN) based video changes detection methods are proposed. These learning based algorithms can learn network parameters from data without manual feature selection. Wang et al [7] proposed an interactive moving objects segmenting method based on CNN. This method can produce segmentation label with comparable accuracy to human beings. However, the method is not fully automatic and the model should be re-trained for new scene. M. Babaee et al [8] proposed a background subtraction system which combines traditional background modeling and CNN based foreground classification algorithm. Although the algorithm outperforms many previous traditional methods without deep learning, its performance still can be improved, especially generalization capability.

An obstacle to the development of learning based methods is the absence of a realistic large-scale dataset with accurate ground-truth. Therefore, impressive performance is hard to achieve by end-to-end network architecture without adequate training data. To address this problem, we attempt to combine the ability of extracting high-level features of CNNs and the well-directed performance of traditional algorithms for certain challenges. Based on the intuition, a robust change detection scheme based on convolutional fusion neural network (RCDFNN) is proposed to automatically integrate the traditional methods into a robust one with improved performance. In the proposed scheme, a well trained CNN is used to extract high-level features from video frames. Based on the features which describes the characteristics of the video content, a fusion network is designed to combine raw detection results of several traditional methods using optimal integration strategy. Moreover, careful fine-tuning for extractor network pre-

* Corresponding author.

Table 1. The performance of four recent change detection methods for different challenges in terms of f-measure. The challenges include dynamic background (dyn.bg), night videos (night), camera jitter (cam.jit), turbulence (turbu) and thermal (therm). Red color indicates the best performance and green color indicates the worst performance.

Method	$\mathrm{FM}_{\mathrm{dyn.bg}}$	$\mathrm{FM}_{\mathrm{night}}$	$\mathrm{FM}_{\mathrm{cam.jit}}$	$\mathrm{FM}_{\mathrm{turbu}}$	$\mathrm{FM}_{\mathrm{therm}}$
PWACS [11]	**0.894**	**0.415**	0.813	**0.645**	0.828
EFIC [17]	**0.578**	**0.655**	**0.713**	0.671	**0.849**
SharedModel [13]	0.822	0.542	**0.814**	0.734	0.832
WeSamBE [16]	0.744	0.593	0.798	**0.774**	**0.796**

vents the algorithm from over-fitting on the limited training data. Thus, the generalization of the method is improved. Evaluations on the 2014 ChangeDetection.net (CDnet) dataset [9] indicate that the proposed method outperforms 9 recently proposed methods [8, 10–17] in terms of average ranking of commonly used metrics.

The rest of this paper is organized as follows. Section 2 will introduce the framework of the proposed integration scheme as well as the details of the CNN architecture and training scheme. Experimental results will be given in Section 3, and conclusions are made in Section 4.

2. THE PROPOSED CNN BASED CHANGE DETECTION SCHEME

Most traditional change detection algorithms depend on hand-crafted features and are designed for some types of scenes with specific challenges. **Table 1** shows the performance of 4 recent algorithms in terms of f-measure [9] for different challenge types. The results show that different algorithms have different application scenarios and some of the algorithms are complementary. For example, PWACS [11] performs the best for dynamic background but can not handle night videos well while EFIC [17] is just the reverse.

Instead of designing a new complex change detection algorithm for handling various challenges simultaneously, we attempt to design a new framework to integrate the complementary strength of existing methods, which perform well for some scenes individually, into a robuster one based on the features extracted from video content.

Considering the trade-off between efficiency and complexity, this paper utilizes the aforementioned 4 complementary detection algorithms as basic detectors and compensates for its shortcomings with each other in different scenes. The parameters used in these algorithms are set as the same as reported in CDnet website.

The proposed framework is shown as **Fig.1**. There are 3 components in this framework and they are described as follows. **1). Basic detectors:** several traditional change detection algorithms serve as weak detectors providing raw segmentation results. **2). Feature extraction network (NET^e):**

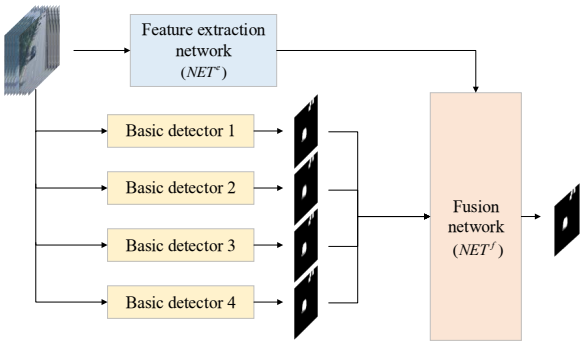

Fig. 1. Framework of the proposed change detection scheme.

A CNN extract features from video content as a descriptor of the video characteristics. The descriptor is used to guide the fusion network. **3). Fusion network (NET^f):** A CNN learns the fusion weights to integrate the raw results into an optimal one based on the characteristics of the video content.

In the procedure of the proposed method, basic detection methods work simultaneously to generate raw results for the current frame. Meanwhile, NET^e takes adjacent frames as input to extract features of the video content. Then, the features are fed into NET^f to decide how to integrate the basic results into an optimal one. The details of the two networks are described in the next sub-sections.

2.1. Feature extraction network

Convolutional neural networks have recently been very successful in a variety of computer vision tasks, such as large-scale image classification by Krizhevsky et al [18]. The main reason is that CNN is capable of learning its own features which is far better than hand-crafted features. Therefore, the scheme will utilize CNN to extract high-level features to describe the characteristics of video frames.

Training a CNN from scratch as a feature extractor for the proposed scheme is almost impossible because of the lack of training data. There have been many successful precedents for using a trained network as feature extractor from some tasks for another new one, such as object tracking [19], image style transforming [20], etc. Inspired by those work, the scheme applies a VGG-16 [21], which is already well trained for image classification, as the feature extractor. However, VGG-16 is originally designed for processing a single image rather than video frames. In this paper we design a network for video feature extraction as shown in **Fig.2**. A sliding window with N frames centered around the current one are fed into VGG-16 in sequence. The feature maps generated by VGG-16 of these frames are stacked into deeper feature maps as a descriptor for the video content.

More specifically, feature maps of 'conv5_2', which reflect high-level semantic of the video content, are stacked as

Fig. 2. VGG-16 is used as video feature extractor. Feature maps of 'conv5_2' are stacked and sent to integration decision maker and those from 'conv2_2' are chosen for post-processor.

feature $\mathbf{f}$ and sent to NET^f. It is worth noting that the VGG-16 has been already trained well on a large-scale image set [22]. Thus, to avoid over-fitting on the limited training videos, the network should be fine-tuned carefully by new training data for change detection.

2.2. Fusion network

For a given video with some types of challenges, the fusion network is designed to obtain an optimal fusion strategy to combine these raw results of basic detectors into a better one. The strategy uses a linear combination of raw results to produce an optimal one as follows,

$$\mathbf{s} = \sum_{c=1}^{C} \mathbf{p}(c) \odot \mathbf{r}(c) \tag{1}$$

where $\mathbf{r}(c) \in \mathbb{B}^{H \times W}$ is the cth basic detector's result which is a 2 dimensional binary map. C is the number of raw detections and H, W are the height and width of the video frame respectively. $\mathbf{s} \in \mathbb{R}^{H \times W}$ is the fusion result which is a weighted sum of the ones of raw results at pixel level. $\mathbf{p} \in \mathbb{R}^{H \times W \times C}$ is the optimal weights maps which reflects the integration strategy. This paper defines a **fusion layer** implementing the operation described in **Eq.1**. Intuitively, if one of the basic detector performs better for the current video with a certain challenge than others, the weight for the result of this detector should be larger relatively. Therefore, the fusion weights $\mathbf{p}$ is related to the video content and is computed as follows,

$$\mathbf{p}(c) = \frac{e^{\mathbf{m}(c)}}{\sum_{k=1}^{C} e^{\mathbf{m}(k)}} \tag{2}$$

$$\mathbf{m} = F(\mathbf{f}; \omega), \mathbf{m} \in \mathbb{R}^{H \times W \times C} \tag{3}$$

Where $\mathbf{f} \in \mathbb{R}^{\frac{H}{16} \times \frac{W}{16} \times 512N}$ represents the feature maps from NET^e mentioned in Section 2.1. $\mathbf{m}$ is the output of a complex nonlinear function with parameters ω and input $\mathbf{f}$. $\mathbf{p}(c)$ is the output of soft-max layer with $\mathbf{m}$ as input. It is also explained as the posterior probability that the cth basic result is the best one given the descriptor of the current video.

Obviously, function F for m is the key of obtaining the optimal prediction of $\mathbf{p}$. In this paper, a CNN is trained to approximate F for its great nonlinear expression ability. The network architecture is shown as **Fig.3**. Video feature $\mathbf{f}$ together with raw results $\mathbf{b}$ are fed into the network and $\mathbf{s}$ is the output. Within NET^f, deconvolution layer [23] is used to decode the high-level video features into fusion weights for the raw results. Sub-pixel layer [24] is used to re-order the weights into a pixel-level fusion maps. Activation layers enhance the nonlinear ability. The network is supposed to learn that given video frames with a certain type of challenge, which basic change detection algorithm performs the best.

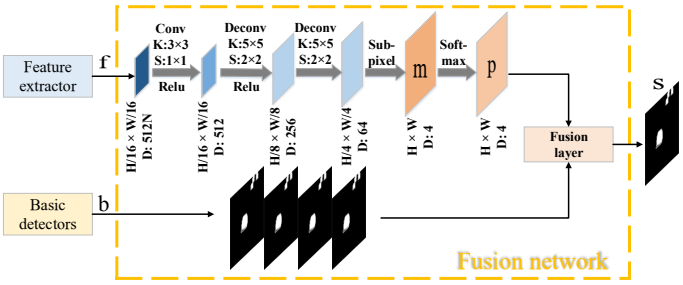

Fig. 3. Architecture of the fusion network. **'Conv'** and **'Deconv'** represent convolution and deconvolution layer respectively. **'Relu'** means the activation is applied following **'Conv'** or **'Deconv'**. **K** is the size of convolution kernel. **S** represents stride size and **D** is the depth of feature maps.

2.3. Training the networks

In the proposed method, NET^e and NET^f are involved in the training process. The parameters of these CNNs should be optimized by back propagation [25] for the change detection task. To train the networks effectively, the loss function is defined as the mean square error between the fusion result and the ground truth map shown as **Eq.4**.

$$L = \frac{1}{H \times W} \sum_{i=1}^{H \times W} (\mathbf{g}(i) - \mathbf{s}(i))^2 \tag{4}$$

The results of change detection are binary maps with 0 and 1 representing background and foreground respectively. In this paper, basic results and the ground truth are transformed by replacing 0 with -1 for back propagation in training stage. Therefore, the fusion result is a continuous number ranging between -1 and 1 while training. In testing stage, the output is turned into binary map finally shown as **Eq.5**.

$$\mathbf{s}^*(i) = \begin{cases} 1, \mathbf{s}(i) > 0 \\ 0, \mathbf{s}(i) <= 0 \end{cases} \tag{5}$$

3. EXPERIMENT RESULTS

The CDnet 2014 [9] is the largest realistic change detection data set consisting of 53 videos with pixel-wise ground-truth.

Table 2. The performance of the proposed RCDFNN method and 9 recent outstanding change detection methods in terms of 6 metrics and the average ranking. **Red** color indicates the best performance. ↓ indicates the smaller the better for the metric and ↑ is the opposite.

Method	$FPR\downarrow$	$FNR\downarrow$	$PWC\downarrow$	$Re\uparrow$	$Pr\uparrow$	$FM\uparrow$	Average rank
RCDFNN	0.0054	**0.2432**	1.4955	**0.7568**	0.8543	**0.8026**	**2.33**
IUTIS-5 [10]	0.0040	0.2764	**1.4951%**	0.7236	0.8832	0.7954	2.50
PWACS [11]	0.0037	0.3009	1.5673%	0.6991	**0.8868**	0.7819	3.17
FTSG [12]	0.0054	0.2830	1.6555%	0.7170	0.8475	0.7768	4.17
SharedModel [13]	0.0080	0.2698	1.8105%	0.7402	0.7950	0.7666	5.17
CwisarDRP [14]	**0.0032**	0.4124	1.9642%	0.5876	0.8848	0.7062	6.50
WeSamBE [16]	0.0059	0.3418	1.9423%	0.6582	0.8228	0.7314	6.83
SubSENSE [15]	0.0076	0.3026	1.9450%	0.6974	0.7935	0.7423	7.00
DeepBS [8]	0.0060	0.4047	2.1983%	0.5953	0.8068	0.6851	8.33
EFIC [17]	0.0134	0.3181	2.5639%	0.6819	0.6812	0.6812	9.00

It is used for performing experiments in this paper. More specifically, each video is separated into two equal-sized non-overlapping sub sequences, where the first one is for training and the second one is for testing.

The performance metrics are computed using the framework of the CDNET 2014 challenge. The framework is evaluated according to the following 6 different measures [9]: false positive rate (FPR), false negtive rate (FNR), percentage of wrong classifications (PWC), recall (Re), precision (Pr) and f-measure (FM). An average ranking of the tested algorithms is also computed based on the partial ranks on these metrics. The training details and test results are shown as follows.

Training. The sliding window centered around a training frame together with the related ground truth map form a training sample. The width of window N is set as 7. Each mini-batch uses an 128×128 patch randomly cropped from these samples. The Adam optimizer [26] is used with $\epsilon = 1.0$, $\beta_1 = 0.9$ and $\beta_2 = 0.999$ and the system is trained with a total of 4.0M steps. For learning rates, we set $lr^f = 10^{-4}$, $lr^e = 10^{-7}$ for NET^e and NET^f respectively.

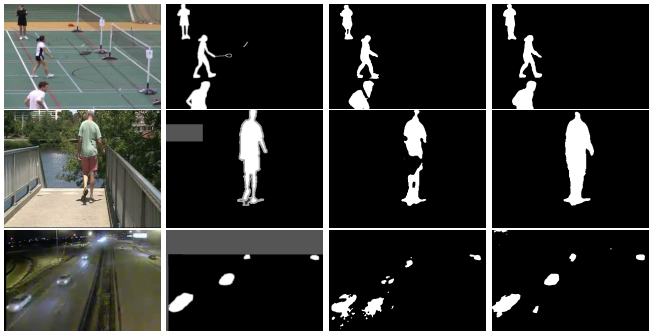

Fig. 4. Examples of binary masks created by IUTIS-5 [10] and the proposed RCDFNN. Images in the first column are the original frames extracted from category cam.jit, dyn.bg and night respectively. The second column is the ground truth. The results of IUTIS-5 and RCDFNN are shown in the third and the fourth columns.

Test results. In addition to the basic methods, 5 other recent change detection algorithms with good performance are chosen for comparison. The overall result of performance comparisons is shown in **Table 2**. In this table, the result of RCDFNN outperforms all the other methods in the terms of the average ranking. Compared with IUTIS-5 which also combines several basic methods, the proposed RCDFNN uses high-level video features extracted from CNN to obtain robuster fusion result. And compared with deep learning based method DeepBS, the proposed method also shows better generalization ability, benefiting from the fusion strategy and the training scheme. **Fig.4** shows some examples of binary result created by the proposed RCDFNN and the best method IUTIS-5 among the 9 compared ones. As we can see, for different scenarios, RCDFNN produced detection results with higher precision and lower false alarm. Moreover, the prediction map is more homogeneous visually. The testing results have proven the adaptability and robustness of the proposed method.

4. CONCLUSION

In this paper, an integration scheme for change detection is proposed. Although no single change detection method can address all challenges, they are complementary for different scenes. Thus the proposed method fuses the results of several existent methods into a robust one which is more adaptive to different challenges. The fusion weights is obtained based on the features extracted from the video content. Two CNNs are applied as the feature extractor and fusion network. The proposed networks can be trained in an end-to-end way. Comparison with recent methods shows that the proposed method has achieved state-of-the-art performance.

Acknowledgment. This work was supported in part by National Natural Science Foundation of China (61771306, 61527804, 61521062, 61133009, 61301116), Chinese National Key S&T Special Program (2013ZX01033001-002-002), the 111 Project (B07022), the Shanghai Key Laboratory of Digital Media Processing and Transmissions (STCSM15DZ2270400)

References

[1] Chris Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *cvpr*, 1999, p. 2246.

[2] Kyungnam Kim, T. H Chalidabhongse, D Harwood, and L Davis, "Background modeling and subtraction by codebook construction," in *International Conference on Image Processing*, 2004, pp. 3061–3064 Vol. 5.

[3] Remy Chang, T Gandhi, and M. M Trivedi, "Vision modules for a multi-sensory bridge monitoring approach," in *The International IEEE Conference on Intelligent Transportation Systems, 2004. Proceedings*, 2004, pp. 971–976.

[4] Ciprian David, Vasile Gui, and Florin Alexa, "Foreground/background segmentation with learned dictionary," in *International Conference on Applied Mathematics, Simulation, Modelling, Circuits, Systems and Signals*, 2009, pp. 197–201.

[5] Nizar Bouguila, Nizar Bouguila, and Djemel Ziou, "A robust video foreground segmentation by using generalized gaussian mixture modeling," in *Computer and Robot Vision, 2007. CRV '07. Fourth Canadian Conference on*, 2007, pp. 503–509.

[6] 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.

[7] Yi Wang, Zhiming Luo, and Pierre Marc Jodoin, "Interactive deep learning method for segmenting moving objects," *Pattern Recognition Letters*, 2016.

[8] Mohammadreza Babaee, Duc Tung Dinh, and Gerhard Rigoll, "A deep convolutional neural network for background subtraction," 2017.

[9] Nil Goyette, Pierre Marc Jodoin, Fatih Porikli, Janusz Konrad, and Prakash Ishwar, "Changedetection.net: A new change detection benchmark dataset," in *Computer Vision and Pattern Recognition Workshops*, 2012, pp. 1–8.

[10] Simone Bianco, Gianluigi Ciocca, and Raimondo Schettini, "Combination of video change detection algorithms by genetic programming," *IEEE Transactions on Evolutionary Computation*, vol. PP, no. 99, pp. 1–1, 2017.

[11] Pierre Luc Stcharles, Guillaume Alexandre Bilodeau, and Robert Bergevin, "A self-adjusting approach to change detection based on background word consensus," in *Applications of Computer Vision*, 2015, pp. 990–997.

[12] Rui Wang, Filiz Bunyak, Guna Seetharaman, and Kannappan Palaniappan, "Static and moving object detection using flux tensor with split gaussian models," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 420–424.

[13] Yingying Chen, Jinqiao Wang, and Hanqing Lu, "Learning sharable models for robust background subtraction," in *IEEE International Conference on Multimedia and Expo*, 2015, pp. 1–6.

[14] M. De Gregorio and M. Giordano, "Wisardrp for change detection in video sequences," in *25th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, 26-28 April 2017, Bruges, Belgium*, 2017, vol. PP, pp. 453–458.

[15] Pierre Luc St-Charles, Guillaume Alexandre Bilodeau, and Robert Bergevin, "Subsense: A universal change detection method with local adaptive sensitivity," *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, vol. 24, no. 1, pp. 359–73, 2015.

[16] Shengqin Jiang and Xiaobo Lu, "Wesambe: A weight-sample-based method for background subtraction," *IEEE Transactions on Circuits & Systems for Video Technology*, vol. PP, no. 99, pp. 1–1, 2017.

[17] Gianni Allebosch, Francis Deboeverie, Peter Veelaert, and Wilfried Philips, "Efic: Edge based foreground background segmentation and interior classification for dynamic camera viewpoints," pp. 130–141, 2015.

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, "Imagenet classification with deep convolutional neural networks," in *International Conference on Neural Information Processing Systems*, 2012, pp. 1097–1105.

[19] Lijun Wang, Wanli Ouyang, Xiaogang Wang, and Huchuan Lu, "Visual tracking with fully convolutional networks," in *IEEE International Conference on Computer Vision*, 2015, pp. 3119–3127.

[20] Leon A Gatys, Alexander S Ecker, and Matthias Bethge, "A neural algorithm of artistic style," *Computer Science*, 2015.

[21] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *Computer Science*, 2014.

[22] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[23] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre Antoine Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion.," *Journal of Machine Learning Research*, vol. 11, no. 12, pp. 3371–3408, 2010.

[24] Wenzhe Shi, Jose Caballero, Ferenc Huszar, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," pp. 1874–1883, 2016.

[25] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[26] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *Computer Science*, 2014.