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 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-
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 cannot 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 robust 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** ($NET^e$): 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 \( f \) 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,

\[
s = \sum_{c=1}^{C} p(c) \odot r(c)
\]  

(1)

where \( r(c) \in \mathbb{B}^{H \times W} \) is the \( c \)th 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. \( s \in \mathbb{R}^{H \times W} \) is the fusion result which is a weighted sum of the ones of raw results at pixel level. \( p \in \mathbb{R}^{H \times W \times C} \) is the optimal weights map 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 \( p \) is related to the video content and is computed as follows,

\[
p(c) = \frac{e^{m(c)}}{\sum_{k=1}^{C} e^{m(k)}}
\]  

(2)

\[
m = F(f; \omega), \quad m \in \mathbb{R}^{H \times W \times C}
\]  

(3)

Where \( f \in \mathbb{R}^{H \times W \times 512N} \) represents the feature maps from \( NET^e \) mentioned in Section 2.1. \( m \) is the output of a complex nonlinear function with parameters \( \omega \) and input \( f \). \( p(c) \) is the output of soft-max layer with \( m \) as input. It is also explained as the posterior probability that the \( c \)th 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 \( p \). 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 \( f \) together with raw results \( b \) are fed into the network and \( s \) 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.

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.

\[
L = \frac{1}{H \times W} \sum_{i=1}^{H \times W} (g(i) - s(i))^2
\]  

(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 Eq.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

\[
s^*(i) = \begin{cases} 
1, & s(i) > 0 \\
0, & s(i) \leq 0
\end{cases}
\]  

(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.

<table>
<thead>
<tr>
<th>Method</th>
<th>FPR ↓</th>
<th>FNR ↓</th>
<th>PWC ↓</th>
<th>Re ↑</th>
<th>Pr ↑</th>
<th>FM ↑</th>
<th>Average rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>RCDFNN</td>
<td>0.0054</td>
<td><strong>0.2432</strong></td>
<td>1.4955</td>
<td><strong>0.7568</strong></td>
<td>0.8543</td>
<td><strong>0.8026</strong></td>
<td><strong>2.33</strong></td>
</tr>
<tr>
<td>IUTIS-5 [10]</td>
<td>0.0040</td>
<td>0.2764</td>
<td>1.4951%</td>
<td>0.7236</td>
<td>0.8832</td>
<td>0.7954</td>
<td>2.50</td>
</tr>
<tr>
<td>PW ACS [11]</td>
<td>0.0037</td>
<td>0.3009</td>
<td>1.5673%</td>
<td>0.6991</td>
<td><strong>0.8868</strong></td>
<td>0.7819</td>
<td>3.17</td>
</tr>
<tr>
<td>FTSC [12]</td>
<td>0.0054</td>
<td>0.2830</td>
<td>1.6555%</td>
<td>0.7170</td>
<td>0.8475</td>
<td>0.7768</td>
<td>4.17</td>
</tr>
<tr>
<td>SharedModel [13]</td>
<td>0.0080</td>
<td>0.2698</td>
<td>1.8105%</td>
<td>0.7402</td>
<td>0.7950</td>
<td>0.7666</td>
<td>5.17</td>
</tr>
<tr>
<td>CwiarDRP [14]</td>
<td><strong>0.0032</strong></td>
<td>0.4124</td>
<td>1.9642%</td>
<td>0.5876</td>
<td>0.8848</td>
<td>0.7062</td>
<td>6.50</td>
</tr>
<tr>
<td>WeSanBE [16]</td>
<td>0.0059</td>
<td>0.3418</td>
<td>1.9423%</td>
<td>0.6582</td>
<td>0.8228</td>
<td>0.7314</td>
<td>6.83</td>
</tr>
<tr>
<td>SubSENSE [15]</td>
<td>0.0076</td>
<td>0.3026</td>
<td>1.9450%</td>
<td>0.6974</td>
<td>0.7935</td>
<td>0.7423</td>
<td>7.00</td>
</tr>
<tr>
<td>DeepBS [8]</td>
<td>0.0060</td>
<td>0.4047</td>
<td>2.1983%</td>
<td>0.5953</td>
<td>0.8068</td>
<td>0.6851</td>
<td>8.33</td>
</tr>
<tr>
<td>EFIC [17]</td>
<td>0.0134</td>
<td>0.3181</td>
<td>2.5639%</td>
<td>0.6819</td>
<td>0.6812</td>
<td>0.6812</td>
<td>9.00</td>
</tr>
</tbody>
</table>

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 negative 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 $128 \times 128$ 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.

**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 a robust 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 (STC-SM15DZ2270400).
References


