MFCNET: END-TO-END APPROACH FOR CHANGE DETECTION IN IMAGES

Ying Chen, Xu Ouyang, Gady Agam

Illinois Institute of Technology
Department of Computer Science
Chicago, IL 60616, USA
ychen245, xouyang3@hawk.iit.edu, agam@iit.edu

ABSTRACT

Change detection is an important task in computer vision and video processing. Due to unimportant or nuisance forms of change, traditional methods require sophisticated image preprocessing and possibly manual interaction. In this work, we propose an end-to-end approach for change detection to identify temporal changes in multiple images. Our approach feeds a pair of images into a deep convolutional neural network combining the model of MatchNet [1] and the Fully Convolutional Network [2] modified to reduce the number of parameters. We train and evaluate the proposed approach using a subset of frames from the Change Detection challenge 2014 dataset (CDnet 2014). Experimental evaluation comparing the performance of the proposed approach with several known approaches shows that the proposed approach outperforms existing methods.

Index Terms— Change detection, deep neural network, MatchNet, FCN, MFCNet

1. INTRODUCTION

Change detection is of great interest in various applications, such as medical diagnosis [3], video surveillance [4], and remote sensing [5]. Figure 1 shows an example of change detection between a pair of images of the same scene. The goal of change detection in images is to identify all regions with meaningful changes, such as an object moving, or an area with altered appearance, while ignoring irrelevant changes such as illumination changes. With the large amounts of images and videos recorded every day, detecting meaningful changes between images or video frames in an efficient way is an important task.

The two main steps in change detection are feature extraction and change classification. Various methods have been proposed to address change detection, including image differencing [6], principal component analysis [7], and change vector analysis [8]. These methods require sophisticated preprocessing and/or post-processing steps, and are hard to generalize to other application domains.

Deep learning [9] methods are commonly used due to their ability to generate good representations for raw input. Our approach to change detection in images is based on deep learning techniques. We design an end-to-end Convolutional Neural Network (CNN) [10] to extract high-level features of images and classify change detection. Using CNNs for change detection can be performed at pixel-level and object or image level [11]. In pixel-level approaches, a local neighborhood of a pixel provides a local context. A local context may result in misclassification and is prone to overfitting. In image-based methods [12], a global context is considered thus resulting in higher accuracy while avoiding overfitting.

(a) 1^{st} image (b) 2^{nd} image (c) Change

Fig. 1. Example of change detection between a pair of images.

We propose a new image-level method using state of the art deep learning architectures for change detection in images. Our approach is a combination of a modified MatchNet [1] and a Fully Convolutional Network (FCN) [2] which considers change detection in a similar way to semantic segmentation. We term our architecture MFCNet (Match Fully Convolutional Network). The main contributions of this paper are as follows: (1) we propose a new end-to-end deep learning model to address change detection using the CDnet2014 data [13]; (2) our approach uses image-level classification which is more accurate than pixel-level approaches and can help avoid overfitting. It uses global context and combines information from lower level layers; (3) our model achieves good performance and with fewer parameters. These contributions are demonstrated in the experimental results section.

2. RELATED WORK

Change detection in images is a fundamental step for image processing and understanding in many applications. Multiple methods have been developed for detecting scene changes [14], and using them to support other tasks.

Background subtraction can be considered as a type of change detection. Brutzer et al. [15] compared the performance of nine existing methods using a synthetic video surveillance dataset. Yi et al. [16] presented an end-to-end pixel-level approach for segmenting moving objects using a multi-resolution CNN with a cascaded architecture. The main difficulty in background subtraction algorithms is producing a clean background with minimal wrong detection.

Various traditional methods have been proposed to address change detection in images [6], [17]. The selection of a change threshold value is critical in most methods. For example, it is possible to compute the difference between a pair of images and detect change is detected above a threshold. Principal component analysis can be used to reduce data redundancy, but is scene dependent.

Change detection in images can be done using deep convolutional networks. Several deep learning models were proposed for change detection which avoid the need for handcrafted descriptors. These methods depend on two strategies to accurately delineate changes: scene segmentation and regularization. Sakurada et al. [18] present a two channel convolutional neural network to detect changes based on a pair of image patches. Alcantarilla et al. [19] adopt a deconvolution network approach [20], [21] and demonstrate its ability to learn an appropriate, spatially precise, similarity function for outdoor images. In contrast to these works, we propose a new deep neural network combining the ideas of an effective model used to find similarity between image patches [1] and a network for semantic segmentation [2]. We address change detection in both outdoor and indoor images using a public dataset (CDnet2014).

3. MFCNET FOR CHANGE DETECTION IN IMAGES

Observing that image-level change detection is similar in some aspects to semantic image segmentation, we use semantic image segmentation for change detection, and propose a new deep convolutional network for change detection between pairs of images. The proposed network employs a modified MatchNet [1] and a modified Fully Convolutional Network (FCN) [2]. MatchNet is used to yield high-level features from a pair of images, whereas FCN is used to perform encoding and decoding for the pair of images. The combined network produces an image-level change map which takes global information into account for the classification of each pixel. We denote the proposed network MFCNet (Match Fully Convolutional Networks). In the proposed network, we remove fully connected layers since our goal is to detect change in pixels. The removal of these layers have an added benefit of making simpler networks that run faster and can help avoid overfitting in small datasets.

3.1. Problem Statement

Change detection in images can be defined as follows: Given a sequence of images $I_1, I_2, \ldots, I_M$ taken at the same scene at different times, detect and localize all pixels with relevant changes. We denote pixel coordinates using (x, y) and image values using $I(x, y) \in R^c$ where $c = 1$ for grayscale images and $c = 3$ for RGB images.

In our proposed approach, we take as input a pair of images including a test image and a reference image. The proposed network outputs a binary image O called a change map or mask. A value of 1 in this map indicates change whereas 0 indicates no change:

$$O(x,y) = \begin{cases} 1, & \text{if there is relevant change at pixel (x,y).} \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

3.2. Network Architecture

The architecture of the proposed MFCNet network is shown in Figure 2. It is composed of two parts: a convolutional neural network and a deconvolutional neural network. The purpose of the convolutional neural network is to perform contraction and feature extraction for the image pair. The convolutional neural network borrows ideas from the feature network of MatchNet [1], and is composed of two channels of convolutional networks with shared weights used to encode the pair of input images. Table 1 provides the hyper-parameter settings for each layer of the convolutional network. The deconvolutional network which borrows ideas from a modified FCN [2], aims to improve the delineation and localization of changed regions, and produces a dense prediction map for all pixels at the same time. In order to improve classification results, we combine predictions from two lower layers which keep more detailed information. By doing so, the proposed network can predict finer details while retaining higher-level feature information. Finally, a softmax linear classifier is applied to each unit of the feature map outputted by the deconvolutional network to produce a final dense change map. The total number of parameters in the proposed network is about $700,000$ compared to 134 millions in the original FCN [2] alone.

The data of the proposed architecture in each layer is a three-dimensional $h \times w \times d$ array, where h and w represent spatial dimensions, and d is the channel or feature dimension. The input of the first layer is a pair of images, both with image size of $h \times w$ and d channels. Each layer is composed of convolution with a stride size of 2, followed by a batch normalization layer, a non-linear ReLU activation function [22], and a 2×2 max-pooling layer to reduce spatial dimension.

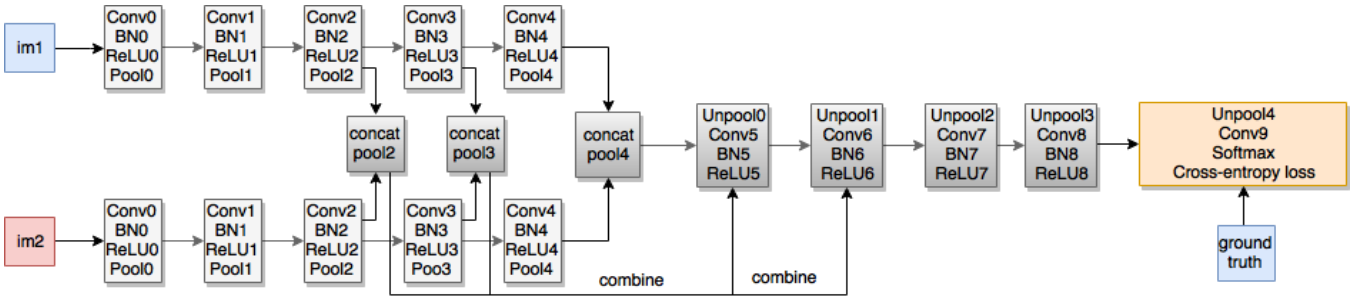


Fig. 2. The architecture of the proposed MFCNet for change detection.

Table 1. Parameters of convolutional network. FS:filter size.

Name	Output Dim.	FS
conv0	$224 \times 224 \times 24$	7×7
pool0	$112 \times 112 \times 24$	3×3
conv1	$112 \times 112 \times 64$	5×5
pool1	$56 \times 56 \times 64$	3×3
conv2	$56 \times 56 \times 96$	3×3
pool2	$28 \times 28 \times 96$	3×3
conv3	$28 \times 28 \times 96$	3×3
pool3	$14 \times 14 \times 96$	3×3
conv4	$14 \times 14 \times 64$	3×3
pool4	$7 \times 7 \times 64$	3×3

By not using any fully connected layers in the architecture, we control the model size and performance.

3.3. Training Details

The weights of the proposed network are initialized randomly using a zero mean normal distribution. We use the Adam optimizer [23] with its original parameter settings to optimize the loss function and train the network. The hyper-parameters of the networks are set as follows. The initial learning rate is set to 10^{-4}, and is set to decay using a pre-defined decay function. The batch size is set to 4 and the number of epochs used is 22. Since video frames are largely redundant, we selected 500 frames from thousands of each available video frames, and the interval is defined as the total number of frames divided by 500. The number of batches per epoch is about 1000. We use cross-entropy as our cost function defined by:

$$E = -\frac{1}{M}\sum_{j=1}^{M}\sum_{i=1}^{N} I[y_j = i] \log p_i^{(j)} \quad (2)$$

where M and N represent the number of instances and labels respectively, I stands for the indicator function when $I[y_j = i]$ is equal to 1 where $y_j = i$, and 0 otherwise, and y_j indicates the ground truth label of each instance. $p_i^{(j)}$ is the probability of instance j for label i, and can be computed by the softmax function:

$$P(y_j|z_j) = \frac{e^{z_j}}{\sum_{k=1}^{N} e^{z_k}} \quad (3)$$

where z_j is the logit of the j^{th} unit in the output layer.

4. RESULTS AND EVALUATION

4.1. Dataset

We train and evaluate our model on the CDnet2014 [13] public video dataset used for change detection. This dataset contains 11 video categories with 4 to 6 videos sequences in each category, and is composed of various challenging situations including dynamic background, camera jitter, shadows, and night videos. This is a large and varied dataset that is suitable for training a deep learning model. We split the dataset into train/dev/test sets with predefined ratios after shuffling the dataset. The network is trained on the train set, and evaluated on the dev set to get the best model. The test set is used to measure the performance of the best model.

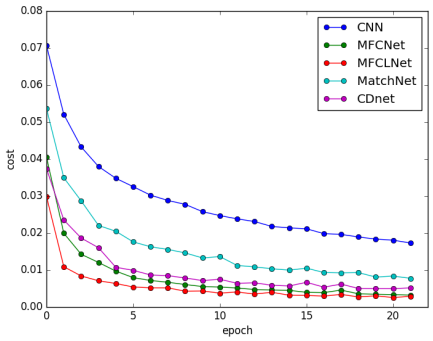
4.2. Evaluation Metrics

Due to the imbalanced nature of the dataset where most pixels do not have a relevant change, we employ the average of $F\text{-}measure$ [24] ($F\text{-}AVG$) of each class as an evaluation metric. It takes both precision and recall into account and combines them into a single metric.

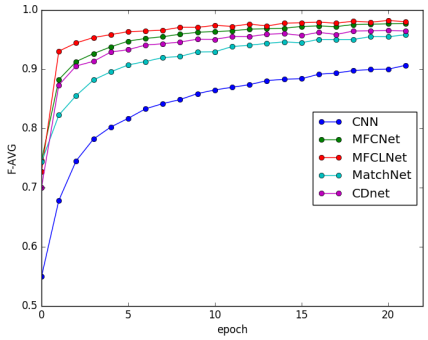
4.3. Experimental Results

4.3.1. Training methodology

We compare our approach with several known techniques, including CNN using a difference patch around each pixel, MatchNet [1] for measuring similarity of patches around each pixel, and a deconvolutional network for measuring the change between images (CDnet) [19]. To have a fair comparison, we set the number of parameters of these methods to be similar to that of the proposed MFCNet. In addition, we compare the performance obtained by our network with a combination of MatchNet and the original FCN, which we



(a) Cost



(b) F-AVG

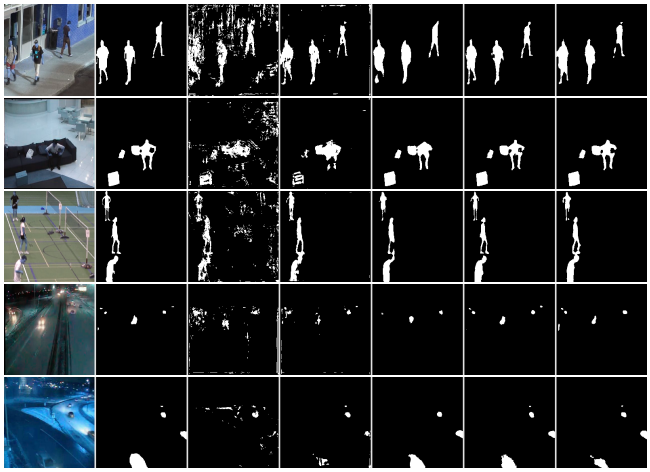
Fig. 3. Cost and F-AVG curves for network training.

term MFCLNet (Match Fully Convolutional Large Network). Note that the proposed MFCNet has about 200 times fewer parameters compared with the proposed MFCLNet which has 13 million parameters. The first two methods extract a small patch around each pixel as the input to the network (thus considering local features), while the remaining methods use the entire image as the input to the networks (thus considering a global context). All images were resized to 224×224 to make the training and evaluation simpler.

Figure 3 shows the smoothed training results obtained on the CDnet2014 dataset. As shown in the figure, our methods (MFCNet and MFCLNet) and CDnet perform better than the other two approaches in most cases, due to the fact that

Table 2. Evaluation performance

Name	F-AVG	F-measure	Accuracy
CNN	0.707	0.454	0.928
MatchNet	0.922	0.852	0.989
CDnet	0.968	0.939	0.994
MFCLNet	0.976	0.953	0.996
MFCNet	0.975	0.950	0.996



(a) (b) (c) (d) (e) (f) (g)

Fig. 4. Visual test results on CDnet2014. (a) Image; (b) Ground truth; (c) CNN; (d) MatchNet; (e) CDnet; (f) MFCLNet; (g) MFCNet.

they take global context into account and so can handle noisy pixels. In addition, taking an entire image rather than a local patch as an input to the network saves time and resources as well as reduce redundant information. As can be observed, our proposed methods have the best performance in most epochs during training, which demonstrates that combining the modified MatchNet and FCN works well for change detection. From the figures, we also observe that the proposed MFCNet has close performance to the proposed MFCLNet, while using substantially fewer parameters.

4.3.2. Evaluation methodology

We evaluated the performance of the trained model on the separate test set. The results in Table 2 show that our methods outperform other methods in both $F-measure$ and accuracy, thus demonstrating better generalization of the proposed methods.

Figure 4 shows some visual results of the compared methods. As can be seen, the proposed model detects more regions compared with other methods and is also able to localize sharp corners which are generally hard to localize.

5. CONCLUSION AND DISCUSSION

We propose a new approach to solve change detection in images using deep learning. Our proposed network combines ideas from MatchNet and FCN, and is demonstrated to yield better performance on a challenging change detection dataset when compared with several existing methods. In future, we plan to focus on detecting the exact kind of change, such as, human, car, or road, instead of just classifying change or no change.

6. REFERENCES

[1] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. Berg, "Matchnet: Unifying feature and metric learning for patch-based matching," in *Proc. CVPR*. IEEE, 2015, pp. 3279–3286.

[2] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. CVPR*, 2015, pp. 3431–3440.

[3] M. Bosc, F. Heitz, J.P. Armspach, I. Namer, D. Gounot, and L. Rumbach, "Automatic change detection in multimodal serial mri: application to multiple sclerosis lesion evolution," *Neuroimage*, vol. 20, no. 2, pp. 643–656, 2003.

[4] C. Stauffer and E. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. PAMI*, vol. 22, no. 8, pp. 747–757, 2000.

[5] L. Bruzzone and F. Prieto, "An adaptive semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images," *IEEE Trans. IP*, vol. 11, no. 4, pp. 452–466, 2002.

[6] M. Hussain, D. Chen, A. Cheng, H. Wei, and D. Stanley, "Change detection from remotely sensed images: From pixel-based to object-based approaches," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 80, pp. 91–106, 2013.

[7] GF Byrne, PF Crapper, and KK Mayo, "Monitoring land-cover change by principal component analysis of multitemporal landsat data," *Remote sensing of Environment*, vol. 10, no. 3, pp. 175–184, 1980.

[8] R D Johnson and ES Kasischke, "Change vector analysis: a technique for the multispectral monitoring of land cover and condition," *Intl. J. Remote Sensing*, vol. 19, no. 3, pp. 411–426, 1998.

[9] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*, vol. 1, MIT press Cambridge, 2016.

[10] L. Chua and T. Roska, "The cnn paradigm," *IEEE Trans. Circuits and Systems I: Fundamental Theory and Applications*, vol. 40, no. 3, pp. 147–156, 1993.

[11] YH Araya and C. Hergarten, "A comparison of pixel and object-based land cover classification: a case study of the asmara region, eritrea," *WIT Transactions on The Built Environment*, vol. 100, pp. 233–243, 2008.

[12] G. Chen, G. Hay, and M. Carvalho, L.and Wulder, "Object-based change detection," *Intl. J. Remote Sensing*, vol. 33, no. 14, pp. 4434–4457, 2012.

[13] Y. Wang, P. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar, "Cdnet 2014: An expanded change detection benchmark dataset," in *Proc. CVPR Workshop*. IEEE, 2014, pp. 393–400.

[14] J. Košecka, "Detecting changes in images of street scenes," in *Proc. ACCV*. Springer, 2012, pp. 590–601.

[15] S. Brutzer, B. Höferlin, and G. Heidemann, "Evaluation of background subtraction techniques for video surveillance," in *Proc. CVPR*. IEEE, 2011, pp. 1937–1944.

[16] Y. Wang, Z. Luo, and P.M. Jodoin, "Interactive deep learning method for segmenting moving objects," *Pattern Recognition Letters*, vol. 96, pp. 66–75, 2017.

[17] S Minu and A. Shetty, "A comparative study of image change detection algorithms in matlab," *Aquatic Procedia*, vol. 4, pp. 1366–1373, 2015.

[18] K. Sakurada and T. Okatani, "Change detection from a street image pair using cnn features and superpixel segmentation.," in *Proc. BMVC*, 2015, pp. 61–68.

[19] P. Alcantarilla, S. Stent, G. Ros, R. Arroyo, and R. Gherardi, "Street-view change detection with deconvolutional networks.," in *Robotics: Science and Systems*, 2016.

[20] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. ICCV*, 2015, pp. 1520–1528.

[21] G. Ros, S. Stent, P. Alcantarilla, and T. Watanabe, "Training constrained deconvolutional networks for road scene semantic segmentation," *arXiv preprint arXiv:1604.01545*, 2016.

[22] A. Glorot, X.and Bordes and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. Intl. Conf. AI and Statistics*, 2011, pp. 315–323.

[23] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[24] Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan, *Introduction to information retrieval*, vol. 39, Cambridge University Press, 2008.