Remote Sensing of Environment 256 (2021) 112308

Contents lists available at ScienceDirect

Remote Sensing of Environment

journal homepage: www.elsevier.com/locate/rse

Change detection using deep learning approach with object-based image analysis

Tao Liu [a,b,*], Lexie Yang [a], Dalton Lunga [a]

[a] GeoAI, Geospatial Science and Human Dynamics Division, Oak Ridge National Laboratory, United States of America
[b] College of Forest Resources and Environmental Science, Michigan Technological University, United States of America

ARTICLE INFO

Keywords:
Change detection
OBIA
Deep learning
Pixel-based
Feature fusion

ABSTRACT

In their applications, both deep learning techniques and object-based image analysis (OBIA) have shown better performance separately than conventional methods on change detection tasks. However, efforts to investigate the effect of combining these two techniques for advancing change detection techniques are unexplored in current literature. This study proposes a novel change detection method implementing change feature extraction using convolutional neural networks under an OBIA framework. To demonstrate the effectiveness of our proposed method, we compare the proposed method against benchmark pixel-based counterparts on aerial images for the task of multi-class change detection. To thoroughly assess the performance of our proposed method, this study also for the first time compared three common feature fusion schemes for change detection architecture: concatenation, differencing, and Long Short-Term Memory (LSTM). The proposed method was also tested on simulated misregistered images to evaluate its robustness, a factor that plays an important role in compromising change detection accuracy but has not been investigated for supervised change detection methods in the literature. Finally, the proposed change detection method was also tested using very high resolution (VHR) satellite images for binary class change detection to map an impacted area caused by natural disaster and the result was evaluated using reference data from the Federal Emergency Management Agency (FEMA). With the experimental results from these two sets of experiments, we showed that (1) our proposed method achieved substantially higher accuracy and computational efficiency when compared to pixel-based methods, (2) three feature fusion schemes did not show a significant difference for overall accuracy, (3) our proposed method was robust in image misregistration in both testing and training data, (4) we demonstrate the potential impact of automation to decision making by deploying our method to map a large geographic area affected by a recent natural disaster.

1. Introduction

With the increasing availability of sensors deployed on spaceborne and airborne remote sensing platforms, massive volumes of multi-temporal remote sensing imagery targeting the same geographical locations have increased over time. Analyzing this multitemporal imagery for change detection is important to monitor land cover changes in urban areas and natural environments (Liu and Lathrop Jr, 2002; Ridd and Liu, 1998; Wang et al., 2018; Yang et al., 2003), assess disaster damages (Abuelgasim et al., 1999; Gong et al., 2012; Wang and Xu, 2010), and examine forest disturbance (Healey et al., 2018; Zhu et al., 2019). However, several issues related to multitemporal imagery, including the heterogeneity in space and time, the rarity of land-cover changes, the presence of data at multiple scales and multiple sources, the paucity of training data, as summarized by Karpatne et al. (2016), still pose serious challenges for advancing change detection techniques to support downstream applications as part of decision making at large.

Change detection methods can broadly be divided into two categories. The first one performs a map-to-map differencing to detect changes (Boldt et al., 2012; El-Kawy et al., 2011) and is commonly known as post-classification. Because the quality of the change detection results is heavily dependent on the maps separately generated from different time steps, errors in one map are propagated toward the final change detection results, making post-classification methods prone to error. Even if two accurate land cover maps are available, the common misregistration issue between maps would impose errors directly on the final change map product.

The second category of change detection methods does not require

* Corresponding author at: College of Forest Resources and Environmental Science, Michigan Technological University, United States of America.
E-mail address: taoliu@mtu.edu (T. Liu).

https://doi.org/10.1016/j.rse.2021.112308
Received 5 April 2020; Received in revised form 22 December 2020; Accepted 14 January 2021
Available online 23 January 2021
0034-4257/

map generation for multitemporal images. Instead, it directly generates a change detection map using bitemporal images through three steps: feature extraction, feature fusion, and change feature classification. For the feature extraction step, the features can be, for example, raw pixel values (Lyu et al., 2016), handcrafted features (Ehlers et al., 2014; Gong et al., 2017), or features that are automatically learned by convolutional neural network (CNN) (Mou et al., 2019). Feature fusion takes as input the two feature vectors extracted from bitemporal images to form a single feature vector or scalar that usually measures similarity between those two features, and the resulting single feature vector or scalar value is therefore applicable in the change feature classification step that follows the feature fusion step. Common fusion operations include Multivariate Alteration Detection (MAD) (Nielsen et al., 1998), Change Vector Analysis (CVA) (Lambin and Strahlers, 1994), Spectral Angle Mapper (SAM) (Yuhas et al., 1992), correlation coefficient (Ehlers et al., 2014), concatenation (Gong et al., 2017), and simple differencing (Zhang and Lu, 2019), i.e., subtraction of one feature vector from the other feature vector. Feature vectors can also be fused by concatenation to form another feature vector, which may or may not go through Principal Component Analysis (PCA) transformation (Wold et al., 1987) before the change feature classification step is performed. Change feature classification aims to assign change type labels to analysis unit (e.g., pixel level or image object level) using the change features derived from feature fusion step, resulting in potentially two products: binary change types (i.e., changed and unchanged types) and multi-class change types. In the latter, the choice of change types depends on the end user interests. Thresholding operation (Chen et al., 2014; Dai and Khorram, 1998; Liu et al., 2016; Vu et al., 2004), unsupervised clustering (Celik, 2009; Zhan et al., 2017), or supervised classification (Gong et al., 2017; Mou et al., 2019; Zhang and Lu, 2019) are common techniques for the feature classification task.

Depending on the analysis unit, the second category of change detection methods can be pixel or object based. Pixel-based methods extract land cover features from single pixels or the neighborhoods of single pixels and assign the classification labels to the pixels. In contrast, object-based methods operate with homogenous super-pixels as units and can be generated by unsupervised image segmentation algorithms. Because object-based methods treat objects as analysis units, all the pixels within a given object are assigned with the same classification label once the classification result for this object is obtained, substantially reducing the "salt-and-pepper" noise that is commonly seen in the results of pixel-based methods (Chen et al., 2012). Due to the potential advantages of object-based over pixel-based methods, object-based methods have been increasingly used for land cover mapping, have shown superior performance to pixel-based methods in many publications (Cleve et al., 2008; Fu et al., 2017; Gao et al., 2012; Pande-Chhetri et al., 2017), and are becoming a new paradigm for processing high or very high resolution images (Blaschke et al., 2014). The success of object-based methods in land cover mapping has led to its extensive adoption simultaneously in change detection applications (Chen et al., 2012; Comber et al., 2004; Desclée et al., 2006; Duro et al., 2013; Ehlers et al., 2014; Gong et al., 2017; Walter, 2004).

Since deep learning techniques made a breakthrough in the computer vision area for image classification tasks (Krizhevsky et al., 2012), they have seen an exponential increase of applications in the remote sensing community (Liu et al., 2018; Lunga et al., 2018; Ma et al., 2019; Yang et al., 2018; Yuan et al., 2016). While change detection using deep learning techniques is still in early development, several publications have described its superior performance over existing methods. An unsupervised deep convolutional coupling network for change detection was proposed (Liu et al., 2016) in which two CNN models were used to extract features on the pixel level and distance maps were generated by calculating the Euclidean distance between the bitemporal features. Then, a change map matrix was derived by updating CNN weights and change type labels alternately to minimize a cost function that involves distance maps and change maps. They compared different methods used to calculate the difference maps, showing that difference maps based on Euclidean distance produced the highest accuracy. Instead of training CNNs in an unsupervised manner as done by Liu et al. (2016), Zhan et al. (2017) employed a supervised method to train a CNN that aims to generate features which maximize the Euclidean distance of learned feature vectors between changed pixels while minimizing the distance between unchanged pixels. After the CNN was trained, it was used to extract features from bitemporal image patches on the pixel level, from which the Euclidean distance between the bitemporal features was then derived. The distance map was used to produce an initial change map using a thresholding method and was subsequently improved by using the K-NN algorithm. Their method was shown to be better or comparable with Liu et al. (2016) and Benedek and Szirányi (2009) for binary change detection on two datasets. In contrast with Liu et al. (2016) and Zhan et al. (2017), in which change detection was performed on a pixel level, Gong et al. (2017) applied image segmentation to bitemporal images and extracted handcrafted features from those segments to perform change detection on an object level. The handcrafted features were then stacked and fed into a neural network for binary change detection. They compared their method with conventional methods such as IRMAD, PCA, and SVM for binary change detection, concluding that their method had the best performance. Unlike Zhan et al. (2017), where the CNN was shared between bitemporal images, Zhang and Lu (2019) applied identical but separate CNNs to extract features from bitemporal images. The difference of the bitemporal features was then derived and used to label change type. They compared their method against several classic approaches(e.g., CVA, IRMAD) and also emerging methods proposed in Gong et al. (2017) and Liu et al. (2016), showing that their method presented the best accuracy. Like Zhang and Lu (2019), Mou et al. (2019) used separate CNNs to extract features from bitemporal images. Instead of calculating feature difference, Mou et al. (2019) employed Long Short- Term Memory (LSTM) (Greff et al., 2016; Hochreiter and Schmidhuber, 1997) to fuse the features. Their results showed that their method presented higher accuracy than the ones derived from using conventional classifiers such as random forest and support vector machine for multi-class change detection. They also demonstrated that their method was superior to classic methods such as CVA and IRMAD for binary change detection.

Despite the increasing popularity of object-based methods and the potential of deep learning techniques for change detection, investigations of the effect of combining these two techniques for change detection applications have not been published in literature to the best of our knowledge. Convolutional Neural Network (CNN), the workhorse behind many deep learning architectures, requires image patches as input, which makes the CNN compatible with object-based methods, given that image patches can be easily generated from objects. CNN makes it possible to automatically learn feature representations from objects in an end-to-end training procedure, averting the need for handcrafted features that have been commonly used in traditional object-based methods. Such a subtle but natural relationship between CNN and object-based methods has motivated researchers to combine them for land cover mapping. Liu et al. (2018) showed that utilizing CNN under an OBIA framework achieved substantially higher accuracy than conventional classifiers such as Random Forest and Support Vector Machine in mapping wetlands using Unmanned Aerial Vehicle (UAV) images. eCognition (http://www.ecognition.com/), a commercial software that was developed by Trimble Company, has been generally used by researchers for decades to conduct OBIA for land cover mapping and has recently integrated deep learning classifiers into their latest version. However, investigations of the integration of CNN and OBIA in the change detection domain are still missing in the literature, which accordingly makes it unclear whether such a combination could also benefit change detection applications compared with two recently published pixel-based change detection methods, both of which rely on CNN to extract features and show superior performance to existing change detection methods (Mou et al., 2019; Zhang and Lu, 2019).

As mentioned earlier, feature fusion is an indispensable step for change detection, responsible for merging features extracted from two time steps in a way that the fused features are expected to enhance change detection accuracy and are usable by a classifier to generate change type labels. Even though various feature fusions have been found in different publications, e.g., correlation coefficient (Ehlers et al., 2014), concatenation (Gong et al., 2017), differencing (Chen et al., 2014; Zhang and Lu, 2019), and LSTM (Mou et al., 2019), the statistical comparison of these feature fusion methods regarding their impact on change detection accuracy has not been found in the literature, making it difficult for the end user to decide which one to use for change detection applications.

In addition to various approaches aiming to improve change detection results, image misregistration has been shown to play an important role in compromising the accuracy for change detection of medium and low spatial resolution remote sensing imagery using pixel-based change detection methods (Dai and Khorram, 1998; Townshend et al., 1992). The impact of image misregistration on object-based change detection has also been evaluated in a study by Chen et al. (2014), which concluded that object-based change detection can alleviate the impact of image misregistration in comparison with pixel-based change detection methods. The change detection method used in that study is a conventional one, with handcrafted features used in the feature extraction step, a differencing operation for feature fusion, and a thresholding method for the change type labelling step. Therefore, it remains to be seen whether the power of an object-based framework combined with the ability of feature extraction by CNN would lead to a novel change detection approach that is robust to image misregistration.

Addressing the knowledge gaps identified above, the contributions of this study include the following: *(I) development of a novel change detection approach by utilizing CNN within an OBIA framework, (II) comparison between the proposed method and pixel-based method in terms of accuracy and computation efficiency, (III) comparison of three feature fusion schemes regarding their impact on the accuracy, (IV) evaluation of the robustness of the proposed approach to image misregistration, and (V) assessment of the performance of the proposed method in solving real-world problems.*

2. Study area and material

The experiments were conducted on three sites —two sites in Washington, DC, in the upper right corner of Fig. 1, and one site located in Joplin, MO, at the bottom of Fig. 1). With the availability of abundant reference data, study sites I and II were used to thoroughly evaluate the proposed method, including the investigation of how model performance can be affected by a different choice of change feature fusion methods, the misregistration errors in training samples, and the misregistration errors in testing samples. In addition, site I and site II were also used to compare the proposed method with their pixel-based counterparts regarding accuracy and computation efficiency. To further verify the effectiveness of the proposed method in real world applications, we applied our proposed method as a component in a fully automatic change detection approach (Liu and Yang, 2020) to map the impacted area caused by natural disaster.

2.1. Study site I and II

Washington, DC, provides accurate building footprints and their corresponding very high-resolution aerial remote sensing images to the public for multiple years ranging from 1999 to 2017 through the website https://opendata.dc.gov/. Two study sites occupied by dense human constructions in the urban area of Washington, DC, were selected for this study, with site I covering 29.16 km^2 with 27000 × 27000 pixels and site II 38.88 km^2 with 36000 × 27000 pixels (Fig. 1).

The paucity of changed area is a common problem for change detection studies (Karpatne et al., 2016). To alleviate this issue and

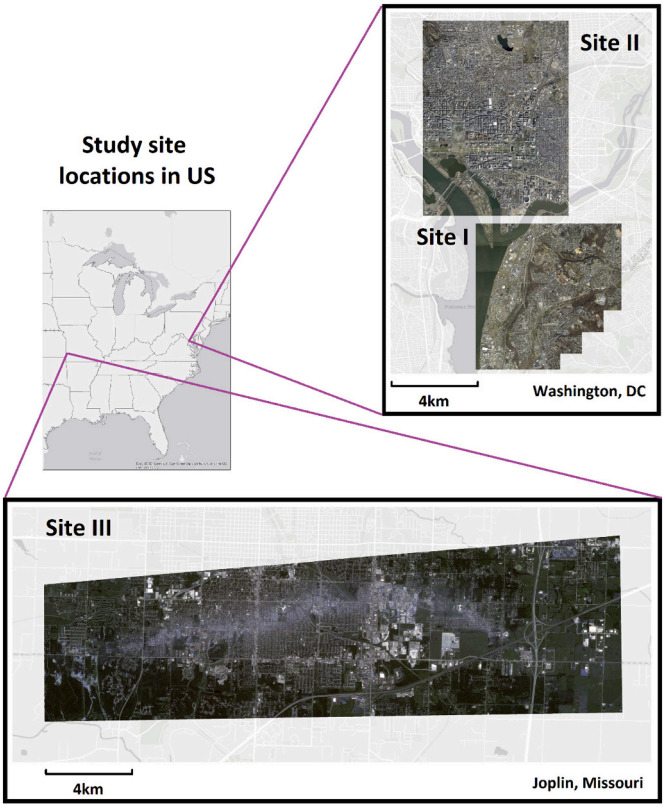

Fig. 1. Locations for three study sites and post-event images for each of them.

maximize the number of potential change events by selecting a longer period of time between bi-temporal images, we used building footprints from 1999 and 2017 in this study. The original spatial resolutions of the aerial orthophotos are 20 cm for the year 1999 and 8 cm for the year 2017. To maintain consistency, the 8 cm resolution of the 2017 aerial orthophoto was resampled to have a 20 cm resolution of the 1999 aerial orthophoto. The two orthophotos were used without conducting any radiometric and geometric corrections for this study (Fig. 2a, b, c, and d).

Building footprints from 1999 and 2017 were overlaid to generate preliminary reference data, which contains four change/unchanged types, including nonbuilding to nonbuilding (NN), nonbuilding to building (NB), building to nonbuilding (BN) and building to building (BB). The geospatial misregistration between the 1999 and 2017 dataset is uneven across the whole area, with some area showing sub-pixel registration error and other areas presenting misregistration error ranging from 1 to 3 pixels. Such misregistration between images introduces errors into the preliminary reference data. An automatic procedure that considers the shape and number of pixels of regions was developed to clean the errors from preliminary reference data. Finally, a reference data based on geospatial location of land cover features shown in the 1999 orthophoto was generated. This reference data shows that NN, NB, BN and BB area types account for 90.9%, 1.6%, 0.9%, and 6.5% for site I and 75.6%, 3.4%, 1.3% and 19.6% for site II, respectively.

2.2. Study site III

Frequency of extreme weather events seem to show a upward trend in recent years according to the U.S. Climate Extremes Index (CEI), which is used to track extreme weather events from the year 1910 to the latest year 2019 (NOAA, 2020). To help humanitarian organizations and government agencies (e.g., FEMA) quickly assess the impacted area caused by natural disasters and deliver assistance to the regions where help is urgently needed, a rapid change detection approach needs to be

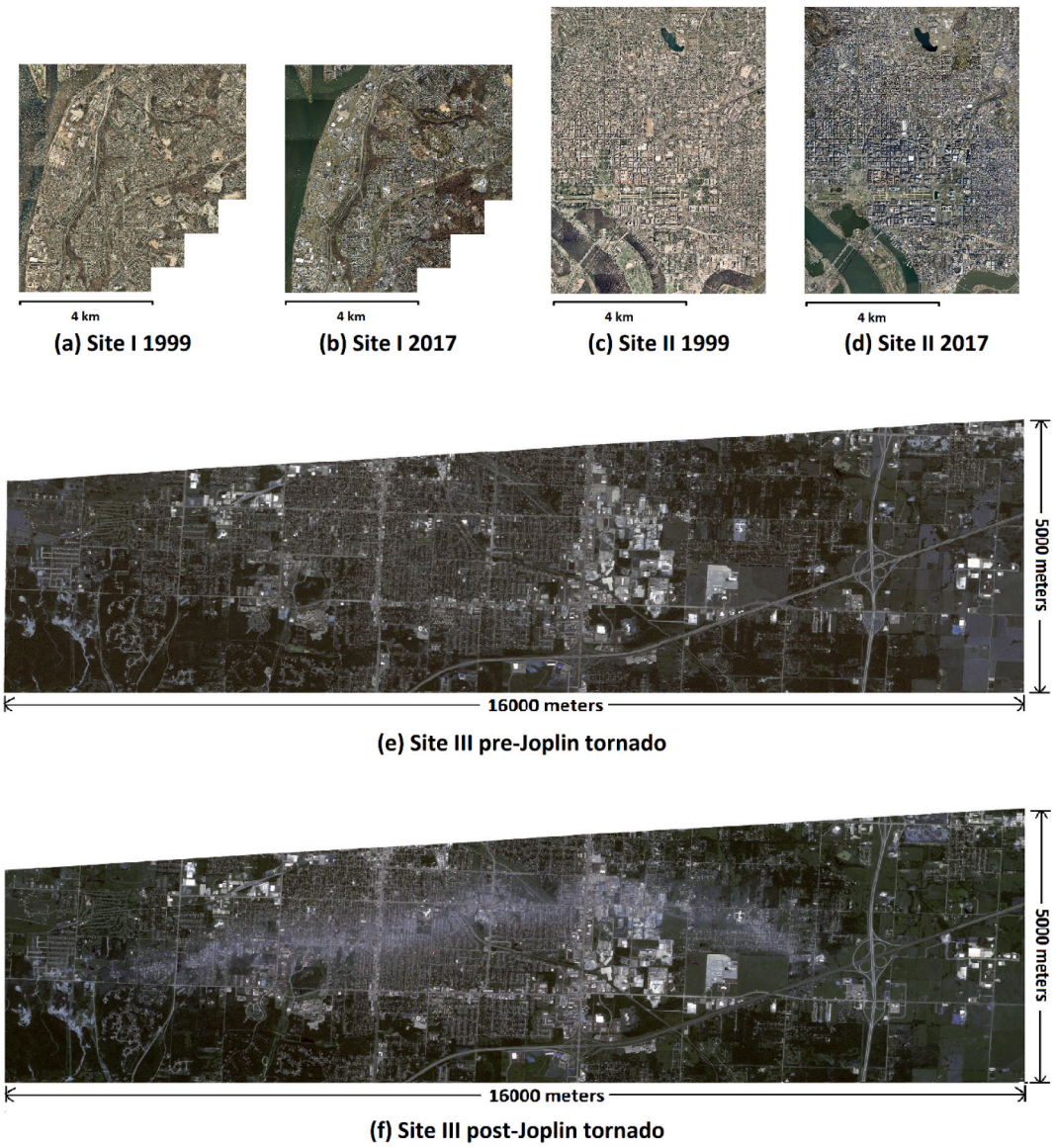

Fig. 2. Aerial orthophotos with 20 cm spatial resolution in sites I and II and WorldView-2 images with 60 cm spatial resolution for site III.

implemented. To demonstrate the usability of the change detection framework proposed in this study, we conducted a natural disaster assessment with bitemporal images on the 2011 Joplin tornado (bottom in Fig. 1). The purpose was to map the impacted area caused by a natural disaster with the proposed change analysis method.

The pre- and post-tornado images are WordView-2 satellite images

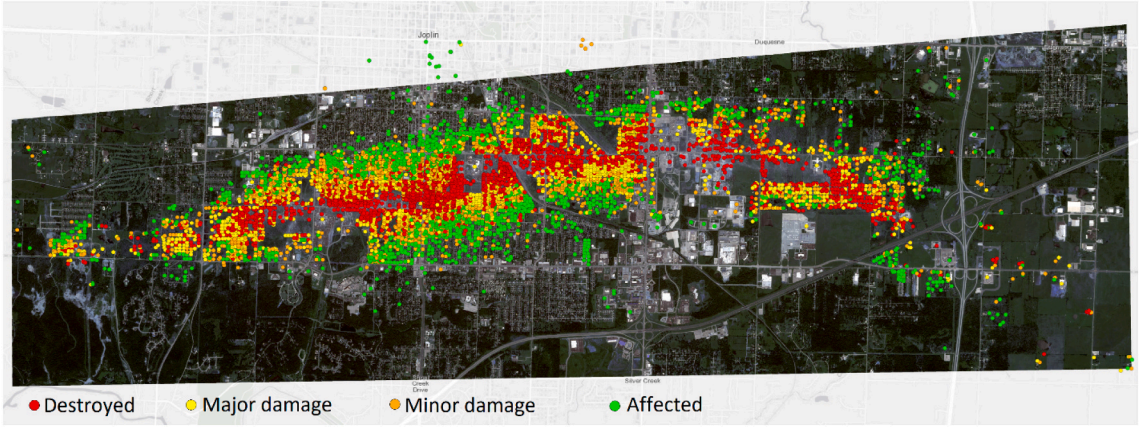

Fig. 3. Reference data prepared by FEMA contains four types of damaged structures.

with 0.6 m resolution. The pre-tornado image scene was collected on August 8, 2009, and the post-tornado image scene was obtained on May 29, 2011, 7 days after the tornado. The original image scenes were cropped to the common extent, and the cropped post-event image scene was resampled to have the same pixel dimensions as the cropped pre-event scene. Finally, this pre-processing operation resulted in pre- and post-tornado images, with both having 0.62 m resolution and covering 80 km^2 with 26000 × 8000 pixels for each of them (Fig. 2e and f).

In the historical damage assessment database (FEMA, 2019), the structures are divided into four categories to describe the amount of damage to structures. The corresponding number of assessed structures in our study area for this tornado event are listed as follows: *Destroyed*: 2497, *Major*: 1729, *Minor*: 1699, and *Affected*: 2197. In Fig. 3, the most significantly impacted areas, where destroyed (yellow dots) and major damaged (blue dots) structures exist, are mainly on the tornado path. According to the Damage Assessment Operations Manual (FEMA, 2016), it is challenging even for humans to observe minor damage from nadir-view 60 cm resolution imagery, such as cracks on the exterior walls. Therefore, we selected data points that are categorized as Major and Destroyed in this database as our reference. It should be noted that the reference data collected by FEMA was only used for evaluating the performance of the proposed change detection model rather than providing the training samples to the model, as they were provided by an automatic sample generation procedure in Liu and Yang (2020). With this test site, we focused on exploiting the proposed change detection approach to map the locations of changes caused by the tornado.

3. Methodology

The proposed change detection method is demonstrated in Fig. 4, in which the site I data are used as the example. It starts from the segmentation of the orthophoto from the year 1999 to create objects, followed by the projection of the boundaries of those objects onto the orthophoto from 2017. By doing this, for any given object in year 1999, its corresponding object in year 2017 can be retrieved based on its location. Given one pair of objects, as shown in the center of Fig. 4, the bounding boxes of those objects are used to generate image patches, which are the inputs of two separate CNNs with identical architectures to extract features. The feature vectors extracted from the year 1999 image patch and the 2017 image patch are represented by blue and purple colors in Fig. 4. The extracted feature vectors need to be fused before they can be used by a classifier to produce classification labels. We tested three types of feature fusion methods in our study, including concatenation (Gong et al., 2017), differencing (Chen et al., 2014; Zhang and Lu, 2019), and LSTM (Mou et al., 2019). The resulting features from the feature fusion step are used for training a classifier, which consists of a fully connected layer and Softmax function (Softmax is not shown in Fig. 4 for simplicity).

The following subsections describe the key components in detail, including segmentation for generating objects, the CNN module, and the adapted feature fusion methods.

3.1. Segmentation

Object characterization is one of the first steps when applying object-

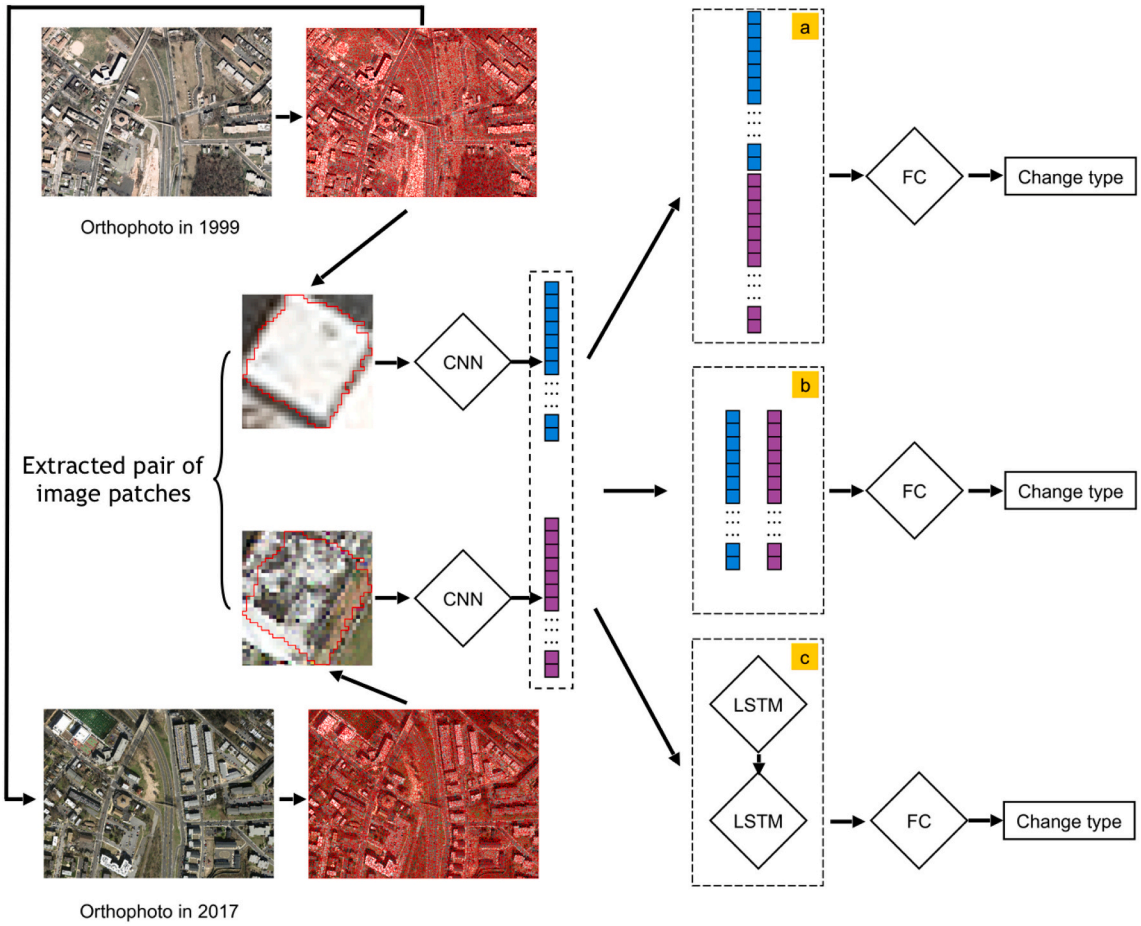

Fig. 4. Flowchart of the proposed method. Three feature fusion methods were tested including (a) concatenation, (b) differencing, and (c) LSTM. CNN = Convolution Neural Network; FC = Fully Connected layer.

based change detection methods. Depending on whether the final objects draw the information from only one of the bitemporal images or both images, the approaches used to generate objects for change detection applications can be divided into two categories. The first category of methods performs segmentation only once using one of the bitemporal images to generate objects, which is the same as what is done for object-based land cover mapping. The change detection algorithm then determines the change type for each object using information extracted from bitemporal images. In contrast, the second category of methods utilizes bitemporal images to generate the final objects. One way to realize the second type of approach is to conduct image segmentation separately on each image and then stack the segmentation results to make a union operation of all the segment boundaries. Another way is to stack the bitemporal images first and then conduct the image segmentation based on the stacked images. The second category of methods has a strong assumption about the geospatial consistency between bitemporal images; therefore, the methods tend to generate unwanted segments for unchanged land cover features (e.g., building footprints) if the images are misregistered. Given that the images used in our study are not strictly registered, in addition to the simplicity associated with the first category of methods, we adopted the first category of methods to generate objects in our study, as illustrated in Fig. 4, where the segmentation map was generated using the 1999 orthophoto only.

Many off-the-shelf algorithms are available to choose from for image segmentation. In the remote sensing community, we opted to use free open-source algorithms for promoting easy research reproducibility, including simple linear iterative clustering (SLIC) (Achanta et al., 2012) and Quickshift (Vedaldi and Soatto, 2008). It is beyond the scope of this study to investigate which one is the best for our change detection application, including those commercial packages. Instead, we conducted image segmentation using both SLIC and Quickshift implemented by scikit-image with Python (Van der Walt et al., 2014) in our preliminary experiments, and the results generated by Quickshift were selected for the experiments in this study, sites I and II, because of its slightly better results than SLIC based on visual inspection of segmentation results. However, we found one advantage of SLIC is that one of its important parameters, called "n_segments", can be set automatically according to the image tile size. This feature of SLIC makes it easier to use in practice. Therefore, SLIC was adopted in the site III experiment when we demonstrated the application of our proposed method in solving real-world problem.

3.2. Convolutional neural network

We used a customized CNN module to extract features from image patches, as shown in Fig. 5. In our preliminary experiments, we tested two strategies for utilizing CNN for feature extraction. Strategy I employs a single CNN to extract features from both bitemporal image patches. In other words, the two image patches share one single CNN module. In contrast, strategy II requires two separate CNN modules, with each of them exclusively working on one of the image patches to extract features. Both strategy I (Zhan et al., 2017) and strategy II (Liu et al., 2016; Mou et al., 2019; Zhang and Lu, 2019) have been used in previous studies, but they were not compared against each other in publications. Our preliminary results indicate that strategy II outperformed strategy I. As such, we adopted strategy II in this study.

The CNN used in this study is based on a modification of ResNet (He et al., 2016). Instead of using deep ResNet directly (e.g., 18 layers, 34 layers, 50 layers etc.), we reduced the ResNet to a relatively shallow one, altering it to include only seven convolutional layers (Fig. 5a). This is due to the concern that the original ResNet was proposed to deal with large-scale image datasets like ImageNet, while our change detection application has only four classes. We tested different depths of CNN in

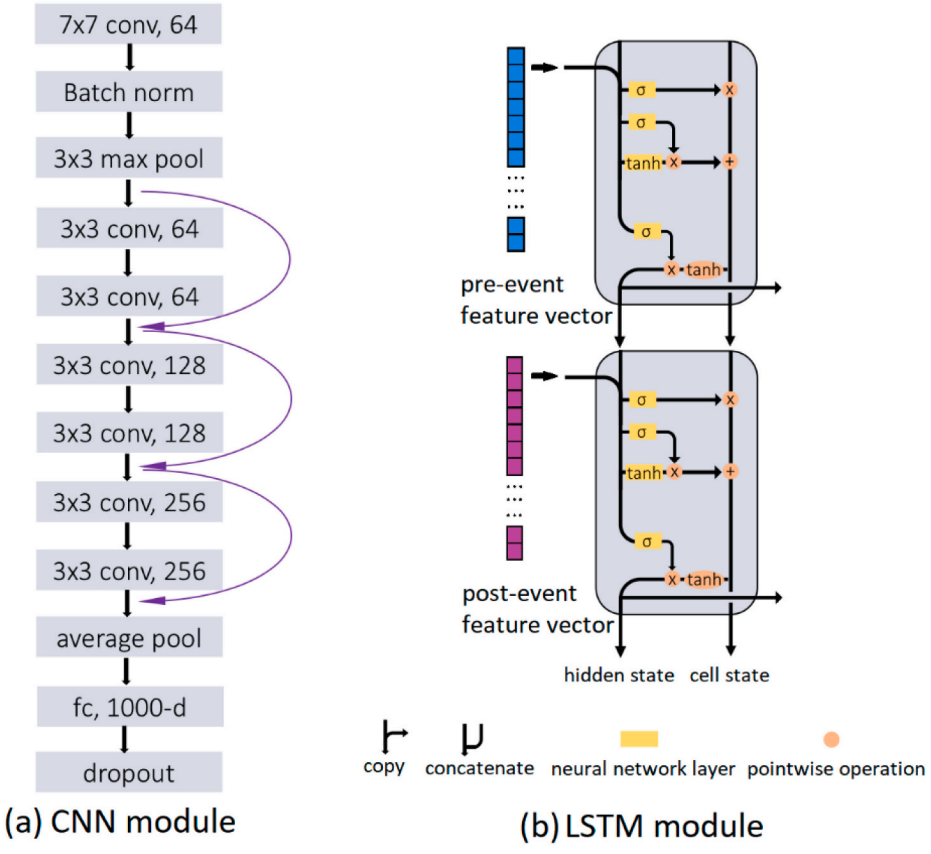

Fig. 5. (a) CNN architecture and (b) LSTM.

our preliminary experiments, and it turned out that with a 50 layer CNN, the training accuracy reached 100% quickly while the testing accuracy fluctuated around a low level, indicating the characteristics of overfitting. Therefore, we modified the network to make it deep enough to be capable of learning useful features but not too deep to introduce overfitting issues for our change detection application. To improve generalization, we also inserted the dropout layer (Srivastava et al., 2014) and the batch normalization layer (Ioffe and Szegedy, 2015) into the network. With the above considerations, the CNN module used in our experiments has 1,489,704 trainable parameters.

We choose 64 × 64 × 3 as the input dimension for our CNN module, simply because such a patch size covers the area of most extracted objects. All the image patches need to be resized to have the same dimension (64 × 64 × 3) before they are input into the CNN. The resizing operation is not shown in Fig. 4 for simplicity. Similar to the other hyperparameters (e.g., number of layers) for all the applications of deep learning models, the dimension of the input layer also has an optimal value that gives the best performance. Here, 64 was selected based on empirical knowledge shown in (Liu and Abd-Elrahman, 2018), which indicates too much enlargement of object patches would decrease the performance of deep learning models.

3.3. Feature fusion

Given one pair of image patches, a feature vector with a length of 1000 can be generated for each patch by the CNN module, as represented by blue and purple bars in Fig. 4. These two feature vectors need to be fused so that they can be used by a classifier to determine the change type label. Three feature fusion methods were tested in our study: (1) concatenation, (2) differencing, and (3) LSTM methods.

The concatenation approach simply concatenates two feature vectors of length 1000 to form a long vector with length 2000. The differencing method subtracts the feature vector of year 2017 from the feature vector of year 1999 to produce a vector of length 1000. LSTM treats the feature vectors extracted from bitemporal images as the records of two events that happened in chronological order and tries to determine the change type. As described in Fig. 5b, the feature vector extracted from the image patch in year 1999 has gone through a series of operations (e.g., sigmoid, tanh, pointwise addition and multiplication) before the extracted information, represented in the form of hidden state and cell state, is transferred into the next time step to interact with the feature vector year 2017 with the same series of operations. The hidden state generated from the year 2017 time step was treated as the final feature fusion result and participated in the next step for classification. The hidden size of LSTM is set as 64 based on change detection overall accuracy in our preliminary experiments.

4. Experiments

4.1. Multi-class change detection for site I and II

4.1.1. Training and testing sample preparation

Image segmentation procedure resulted in 2,498,703 segments in total for site I and site II. Each segment was overlaid onto the reference image and was assigned the majority type of changes within the segment boundary. The distribution of the number of samples for four change types is shown in column 1 of Table 1. This reveals a common obstacle in change detection studies: the number of changed samples are too small compared with unchanged samples. The two changed types, which are NB and BN in Table 1, account for only 2.4% and 1.1% of the total segments, respectively. We split site I into 9 grids and site II into 12 grids, with each grid containing 9000 × 9000 pixels. From each of the 21 grids, we randomly selected 5000 samples for NN, 5000 samples for BB, and all the samples for NB and BN in order to alleviate the imbalance issue. After this first round of sample selection, 2,498,703 samples were reduced to 284,681 and the samples became less unbalanced, as indicated by the comparisons of percentage numbers between column 2 and 3 in Table 1. To reduce the computation cost for the large number of experiments conducted in this study, we further reduced the sample size using a stratified sampling method. To that end, we first used a classifier to extract features for each of the 284,681 samples, where the classifier was trained with one-fifth of 284,681 samples selected with a simple random sampling method. Using the extracted features, we performed unsupervised clustering classification with a K-means algorithm for each of the four change types separately, resulting in five clusters for each change type. After that, we randomly selected 20% samples from each cluster, forming the final number of samples shown in the last column in Table 1. These two rounds of sample selection allow a small set of samples to be randomly selected to represent the whole population while easing the sample unbalance issue. With such two rounds of sample selection, we finally selected 56,936 samples, constituting 2.3% out of 2,498,703 total segments. This set of samples is referred to as the base sample set hereafter.

Table 1
Number of samples before and after the First and Second Round of Sample Selection.

Change type	# Total segments	# Samples after round 1 selection	# Samples after round 2 selection
NN	2,084,692 (83.4%)	105,000(36.9%)	21,000(36.9%)
NB	59,366(2.4%)	59,366(20.9%)	11,872(20.9%)
BN	26,405(1.1%)	26,405(9.3%)	5281(9.3%)
BB	93,910(3.8%)	93,910(33.0%)	18,782(33.0%)
Summary	2,498,703	284,681	56,936

Fivefold cross validation samples were generated using 56,936 samples, as shown the last column of Table 1. This leads to 75% of 56,936 samples used for training and 25% kept for testing for each of the fivefold cross validation experiments.

4.1.2. Misregistration tolerance experiments

4.1.2.1. Misregistration simulation dataset. To determine whether the proposed method is robust to image registration, we simulated misregistration in the direction of 0 degrees (horizontal direction), 45 degrees, and 90 degrees (vertical direction) with translation distances of 2, 4, 6, and 8 pixels, following the practice of previous studies (Chen et al., 2014; Dai and Khorram, 1998; Townshend et al., 1992). During the simulation procedure, the orthophoto from year 1999 was fixed, while the orthophoto from year 2017 was shifted in the given direction and distance. This resulted in 13 groups of datasets, including the original dataset (dataset group #1 in Table 2), and the 12 simulated datasets (dataset group #2 to #13 in Table 2). It should be mentioned that translation is a simplified simulation of image misregistration in the real world, which may also involve scaling and rotation among other types of transformations. However, according to our observation, translation is the most common one.

4.1.2.2. Experiment design. Investigating the tolerance of the proposed method toward image misregistration is one of the primary objectives in this study. We studied the tolerance from two perspectives. On one hand, we are interested to see whether the trained CNN classifier is tolerant of misregistration in testing data. To this end, we trained the classifier using the original data that does not contain intentionally added misregistration (data group#1 in Table 2). Then we applied the trained classifier to testing data in all the 13 data groups (Fig. 6a). On the other hand, we also want to know the impacts of misregistration of training data on classifier performance. To that purpose, we trained the classifier using data group #3 and #5 separately and applied trained classifiers to the testing data in group #1, #2, #3, #4, and #5, as illustrated in Fig. 6b. Data groups #3 and #5, respectively, have misregistration of 4 and 8 pixels in horizontal directions. The

Table 2
Original data and misregistration simulation data.

Dataset group #	1	2	3	4	5	6	7	8	9	10	11	12	13
Direction (degree)	Null	0	0	0	0	45	45	45	45	90	90	90	90
Distance (pixels)	0	2	4	6	8	2	4	6	8	2	4	6	8

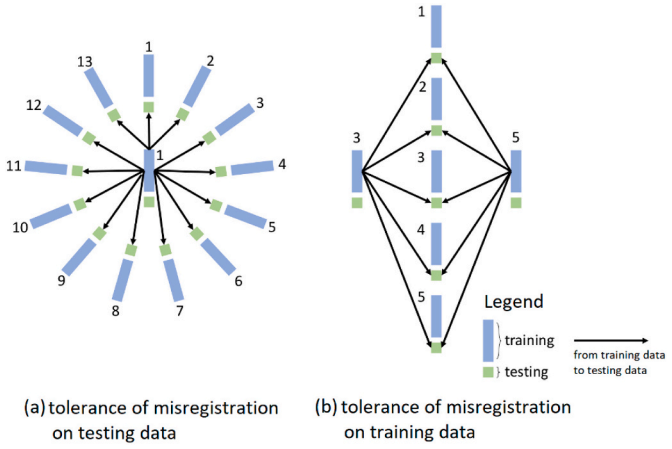

Fig. 6. Left: classifier trained with training data in original dataset (data group #1) was tested on testing data in all the data groups. Right: classifier trained with training data in groups #3 and #5 was tested on testing data in groups #1, #2, #3, #4 and #5. The data group number is consistent with Table 2.

experiment design shown in Fig. 6b will reveal whether such 4 or 8 pixels of misregistration in training datasets would compromise the performance of our proposed change detection methods.

4.1.3. Unbalanced training sample experiment

Change is a rare event, which leads to a seriously unbalanced training dataset where unchanged samples dominate the sample pool. Therefore, unbalanced training datasets are especially common for change detection tasks. A simple method to make the sample more balanced is to decrease the number of unchanged training samples. However, this raises an interesting question: will making the already unbalanced training worse by adding the more easy-to-get unchanged samples to the training sample pool impair the accuracy of other changed/unchanged classes? Finding an answer to this question would help better prepare training samples in practice. To that end, we increased the samples of NN from 21,000 to 42,000, retrained the classifier, and investigated its impact on the model performance.

4.1.4. Benchmark methods

To thoroughly evaluate the proposed methods, we also implemented three pixel-based methods that employ concatenation, differencing, and LSTM, respectively, as feature fusion methods in sites I and II. The pixel-based methods are different from their object-based counterparts in two aspects. First, pixel-based methods use a 5 × 5 neighborhood window (Mou et al., 2019) for a given pixel to create an image chip. This is in contrast with object-based methods, for which the image patch should always exactly enclose the object. Second, during the map generation procedure, given an image patch for object-based methods or image chip for pixel-based methods and the classification result, pixel-based methods assign the classification label to the central pixel of the image patch. Instead, object-based methods assign the classification result to all the pixels constituting the object.

The pixel-based method using differencing operations as a feature fusion method was proposed by Zhang and Lu (2019), and a pixel-based method using LSTM as its feature fusion method was developed by Mou et al. (2019). Both studies claimed that their methods were superior to existing methods, but a comparison of those two methods has not been found in the literature. Concatenation methods have not been found in the literature to be employed by either object-based or pixel-based approaches that rely on end-to-end CNN. To make a fair comparison between pixel-based and object-based change detection approaches using CNN, we ensure the difference between these methods is only on aspects of input generation (image chip from a 5 by 5 window vs. image patch covering extracted object) and the map producing procedure, as mentioned earlier. Therefore, the CNN module used in this study is the same for both object-based and pixel-based method and is different from what was used in Zhang and Lu (2019) or Mou et al. (2019) regarding the layer types and number of parameters.

4.2. Binary class change detection experiments for site III

The binary change detection experiments in site III implemented concatenation rather than differencing or LSTM operations as the change feature fusion method, because the concatenation method is easy to implement and our experiments in sites I and II indicate that differencing and LSTM did not show significantly better performance than the concatenation method, as will be described in the Experiment results section.

To satisfy the needs of rapid mapping of areas impacted by natural disasters, an automatic training sample generation method was adopted in this study. This method extracts training samples from automatically generated sampling areas, which include unchanged sampling area in vegetation regions, unchanged sampling area in infrastructure areas, and changed sampling areas. NDVI was used to generate vegetation masks for pre- and post-event images, and those vegetation masks were overlaid together to identify the overlapping vegetation area between the pre- and post-event images to derive the unchanged sampling area in vegetation regions. SIFT features were extracted and employed to generate the unchanged keypoints, which were used to derive the unchanged sampling area in the infrastructure region. The changed sampling area was generated based on the observation that the density of unchanged keypoints detected by the SIFT features in the changed area is much lower than the unchanged area. Readers are referred to Liu and Yang (2020) for details of the sampling area generation procedure.

After sampling areas were generated, the rest of the operations in site III were the same as what had been conducted on sites I and II for sample extraction, model training, and model inference.

4.3. Model implementation

The proposed change detection architecture was implemented using PyTorch (https://pytorch.org/). Stochastic Gradient Descent (SGD) was used to train weight parameters of the networks with batch size, momentum, and weight decay set as 1500, 0.8, and 0.0001, respectively. Learning rate was set as 0.1 for epoch 1 to 80, 0.01 for epoch 81 to 120, and 0.001 for epoch 121 to 150. The neural networks were trained with a NVIDIA K80 GPU that has a 12GB memory.

4.4. Evaluation metrics

For binary class change detection, the overall accuracy, true positive rate (TPR), true negative rate (TNR), positive predictive value (PPV), and negative predictive value (NPV) were reported. For four class land-use change detection, a confusion matrix was reported. In addition, the results of four class change detection were aggregated to calculate the

value of the binary class evaluation metrics.

5. Experiment results

5.1. Results of four class land-use change detection

5.1.1. Benchmark comparison

Fig. 7 shows the values of evaluation metrics for four class change detection results that were obtained with pixel- and object-based methods. The binary class overall accuracy, TPR, and TNR were calculated by aggregating the four class change detection results. According to Fig. 7, the object-based method obtained substantially higher overall accuracy and TPR compared with the pixel-based method. However, the object-based method showed slightly lower TNR, even though the TNR of the object-based method was as high as 90%.

Fig. 7 also demonstrates that three feature fusion methods showed no significantly difference in terms of overall accuracy, although the LSTM fusion method indicated better TPR to some degree than the other two.

5.1.2. Accuracy of misregistered testing data

Fig. 8 presents the overall accuracy obtained by object-based method using concatenation as a fusion method. The model was trained using a dataset that does not contain simulated misregistration errors, and the testing dataset contains various levels of simulated misregistration errors in directions 0°, 45°, and 90°, following the experiment setup illustrated in Fig. 6a. Fig. 8 demonstrates that shifting direction of misregistration on testing data does not impact the accuracy, regardless of shifting distance. Therefore, our discussion will only focus on the results in the horizontal direction hereafter.

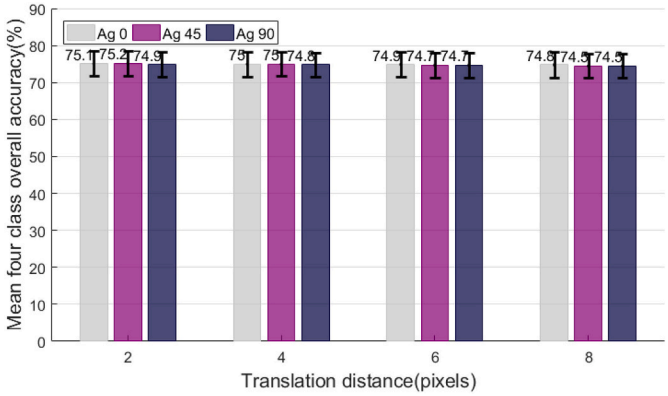

Fig. 8. Mean overall accuracy of fivefold cross-validation obtained by object-based method with concatenation as feature fusion approach. The model was trained with training data that does not contain intentionally added misregistration errors (data group #1) and tested on data groups #2 to #13.

Fig. 9 shows the overall accuracy achieved by the object-based method features are concatenated via fusion methods. Those are parts of the results derived from the experiments illustrated in Fig. 6a. Because misregistration direction does not have an obvious impact on accuracy, as shown in Fig. 8, only the results from data with misregistration on the horizontal level are presented in Fig. 9. Fig. 9 demonstrates that overall accuracy does not experience an obvious decrease with the change of

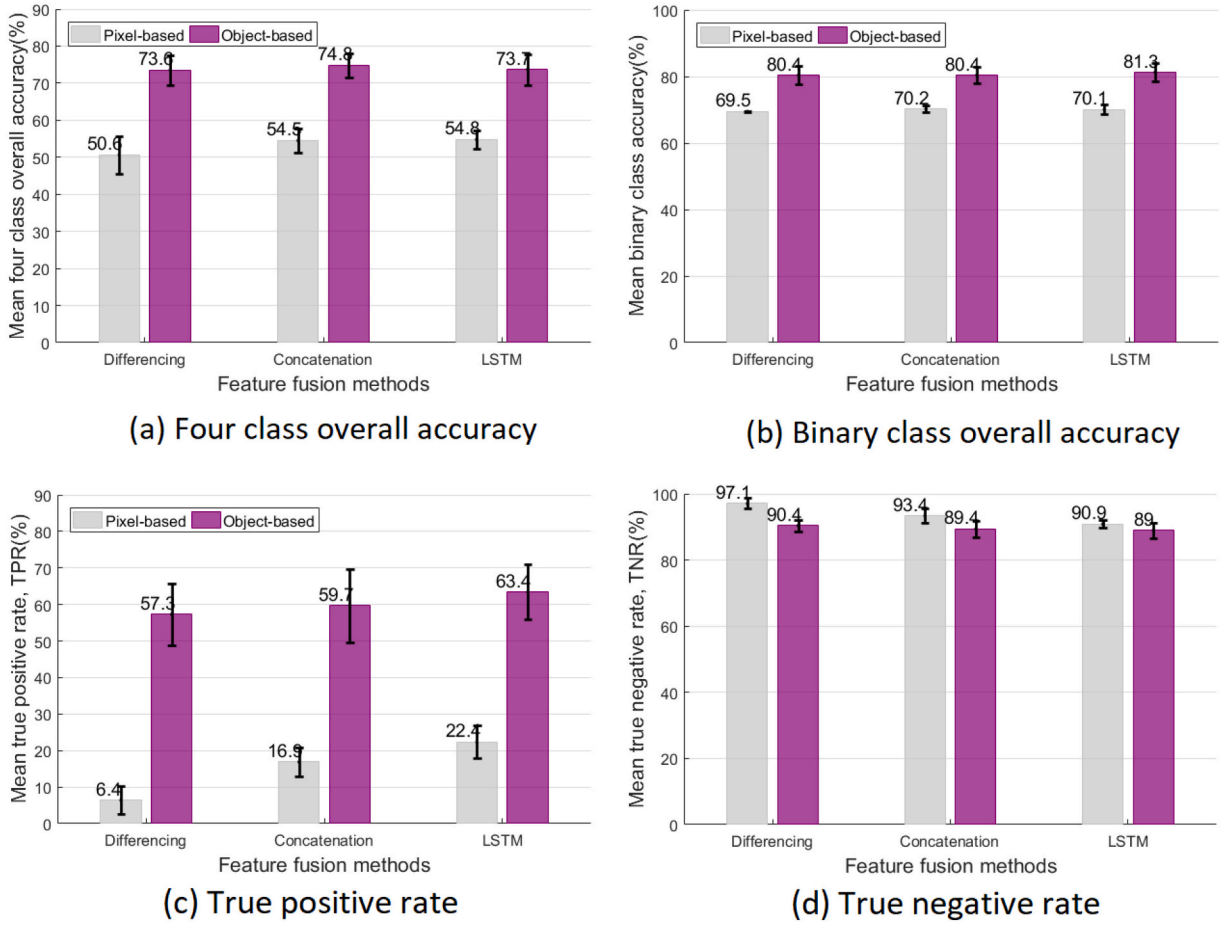

Fig. 7. Fivefold cross-validation results of pixel-based and object-based methods using training and testing data that do not contain intentionally added misregistration errors (i.e., data group #1 in Table 2). The comparison is made based four evaluation metrics including (a) four class overall accuracy, (b) binary class overall accuracy, (c) true positive rate, (d) true negative rate.

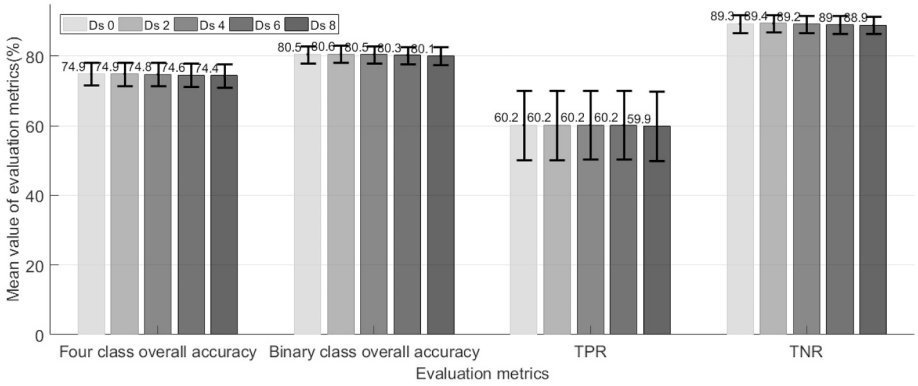

Fig. 9. Mean value of four evaluation metrics for fivefold cross-validation results derived from concatenation fusion method. The model was trained with training data that do not contain intentionally added misregistration errors (data group #1) and tested on data groups #1 to #5 that have various distances of translation in the horizontal direction.

translation distance of misregistration errors regardless of evaluation metrics, indicating that misregistration causes no effects on either changed or unchanged classes.

To better quantify our evaluation, we conducted a paired *t*-test to compare the fivefold results derived from misregistered data with that derived from original data (data group#1). The comparison results are shown in Table 3 in the form of *p*-values. A larger p-value means a smaller difference between the compared results. As shown in Table 3, the p-value tends to become smaller as the translation distance becomes larger, implying that the misregistration still impacts the change detection accuracy in a subtle way. However, none of these impacts are significant for concatenation methods (p-value $>$ 0.05), while the impact becomes significant (p-value $>$ 0.05) for differencing and LSTM methods when the translation distance of misregistration error is larger than 8 pixels and 4 pixels, respectively. The p-values for the concatenation feature fusion method are generally larger than the differencing and LSTM methods, indicating that the concatenation method is more robust than the differencing and LSTM methods.

Given the same testing data, both LSTM and concatenation generated significantly (p-value$<$0.005) higher accuracy than the differencing method, while the LSTM and concatenation methods were not significantly (p-value$<$0.05) different from each other even though concatenation produced a slightly higher mean overall accuracy than LSTM (e.g., 73.9% for LSTM versus 74.7% for concatenation with data group #1).

5.1.3. Accuracy of classifier trained with misregistered training data

Fig. 10 shows the overall accuracy from the experiment design in Fig. 6b, in order to investigate how the misregistration error in training data would impact the change detection performance. Fig. 10 shows that given a classifier trained with a given group of misregistered data, the testing accuracy does not exhibit an obvious decrease with the simulated misregistration errors, although accuracy seems to be slightly higher when the translation distance in the testing sample and training sample is closer.

5.1.4. Change detection map by pixel-based and object-based methods

Because the maps produced by three feature fusion methods do not show any obvious difference, we only present the map product by feature concatenation. Fig. 11 shows the change maps for site I and site II including the prediction results of NN, NB, BN, and BB, four classes in total. Fig. 11 shows that in the lower left corner of site II many NN pixels were misclassified as BB, and a close look at that area shows that road and boats were the major sources for this type of error. Those are human-made features and share similar spectral characteristics with building footprints but do not belong to the building footprint land-use type. In addition, many NB and BN pixels in the prediction map actually correspond to NN pixels in the reference map.

To better examine the results, we zoomed in on two areas in site I and site II, the results of which are shown in Fig. 12. We also produced pixel-based maps for those two subareas and present them alongside the object-based maps for comparison. Fig. 12 shows that the map produced by the pixel-based method is much noisier than the change map produced by the object-based method. For example, road pixels in those two subareas should be classified as NN (Nonbuilding to Nonbuilding), but a large number of them were mistakenly classified as BB (Building to Building) by the pixel-based method. In contrast, the proposed object-based method successfully discovered most of those NN pixels for road area, as highlighted with a red box in Fig. 12. On the other hand, the yellow box in Fig. 12 highlights the case in which the pixel-based method failed to detect the change of newly built buildings.

5.1.5. Model performance using unbalanced samples

To quantitatively examine the accuracy for the four classes, we presented the confusion matrix of the change detection results from the proposed object-based method with feature concatenation in Table 4 for all the objects in sites I and II. Unsurprisingly, the changed classes that have a small share of training samples have lower accuracy, with the BN receiving the lowest producer accuracy (33.6%). In contrast, the class having the largest number of training samples among four classes enjoyed both the highest producer and user accuracy (90.8% producer

Table 3
Paired *t*-test results comparing misregistration data and original data regarding overall accuracy for each of the three feature fusion methods.

Feature fusion	Differencing					LSTM					Concatenation				
Fold\dis.	0	2	4	6	8	0	2	4	6	8	0	2	4	6	8
1	81.8	81.7	81.9	81.8	81.7	82.6	82.4	82.4	82.3	82.1	82.0	82.7	82.6	82.6	82.4
2	72.0	71.5	71.3	71.3	70.8	72.7	72.3	72.0	72.2	71.7	70.8	70.7	70.6	70.4	70.1
3	63.3	63.2	63.0	62.9	62.8	63.7	63.9	63.4	63.0	63.0	72.5	72.4	72.2	72.0	72.2
4	82.3	82.4	82.3	82.0	81.7	84.0	83.8	83.7	83.3	82.9	82.4	82.4	82.4	82.2	81.7
5	65.1	65.1	65.1	65.3	64.9	66.3	66.1	66.2	66.1	65.8	65.9	65.9	65.8	65.7	65.4
Mean	72.9	72.8	72.7	72.7	72.4	73.9	73.7	73.5	73.4	73.1	74.7	74.8	74.7	74.6	74.4
p-Value	–	0.15	0.14	0.10	0.03	–	0.09	0.02	0.005	0.002	–	0.27	0.50	0.26	0.07

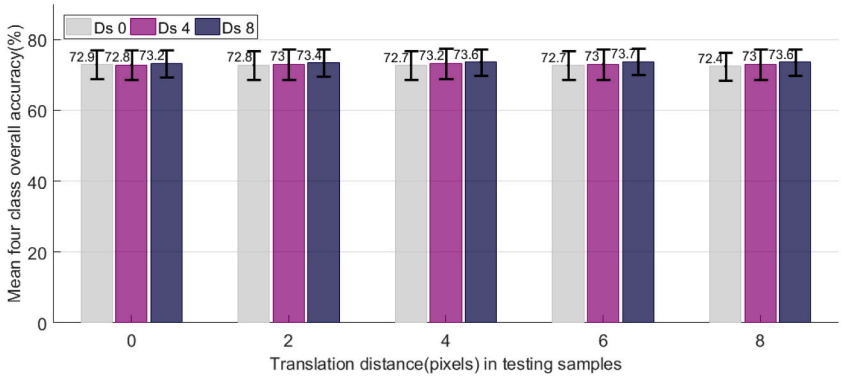

Fig. 10. Mean overall accuracy of fivefold cross-validation for concatenation feature fusion methods trained with the training data in groups #1, #3 and #5 and tested on testing data in the groups #1, #2, #3, #4, and #5.

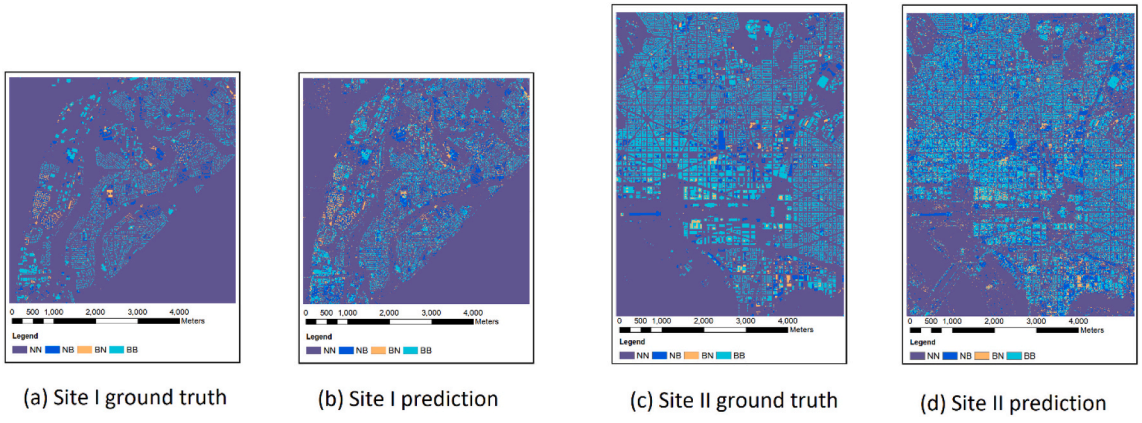

Fig. 11. Reference and prediction for site I and site II by the object-based change detection with feature concatenation.

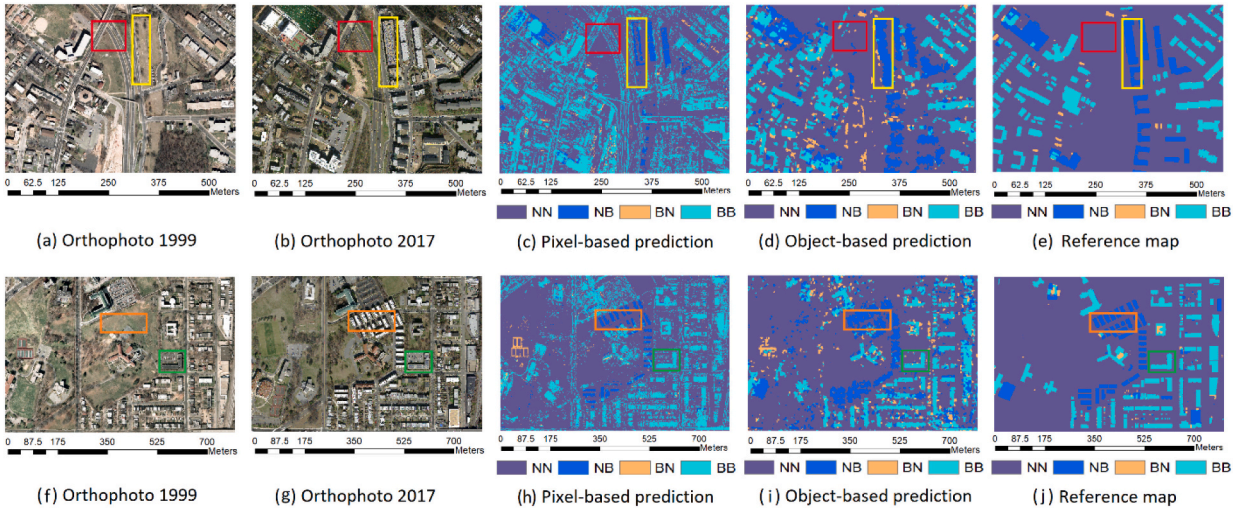

Fig. 12. Maps produced by pixel-based and object-based change detection methods for a section of site I (first row) and site II (second row).

accuracy and 96.1% user accuracy for NN).

Table 5 shows the confusion matrix of the new results derived by the retrained classifier for the same area. After new NN training samples were injected into the sample pool, all the classes showed improvement in both user and producer accuracy. This indicates that making the training samples more unbalanced by adding more dominant class training samples does not necessarily decrease the model accuracy. In other words, decreasing the number of dominant class samples to create a more balanced training dataset does not serve the purpose of improving classier performance and thus is not advisable, according to our experiment results. However, this observation should be taken with caution when applied to other types of feature representation, since changing the number of training sample would change feature representation of CNN, while some other types of features are not impacted by the change of training samples (e.g., features from CVA).

Table 4
Confusion matrix of results by classifier trained with samples that has 21,000 NN samples.

	NN	NB	BN	BB	Producer accuracy (%)
NN	1,892,437	78,546	26,155	87,551	90.8
NB	15,265	35,198	570	8333	59.3
BN	11,500	1067	8873	4965	33.6
BB	50,951	25,006	9107	243,176	74.1
User accuracy (%)	96.1	25.2	19.8	70.7	87.2

Table 5
Confusion matrix of results by classifier trained with samples that has 42,000 NN samples.

	NN	NB	BN	BB	Producer accuracy (%)
NN	1,939,219	56,857	20,566	68,047	93.0
NB	11,051	40,620	505	7190	68.4
BN	8332	811	12,800	4462	48.5
BB	44,132	23,950	8103	252,055	76.8
User accuracy (%)	96.8	33.2	30.5	76.0	89.8

5.2. Binary change detection to map impacted area caused by tornado

Fig. 13 presents the tenfold cross-validation of the binary change detection model that was implemented for site III. Compared with fivefold cross-evaluation of four class change detection results for sites I and II shown in Fig. 7, the tenfold cross-evaluation of the binary change detection model for site III as presented in Fig. 13 shows considerably higher value for all the evaluation metrics, with 98.6%, 91.2%, 99.3%, 92.4%, and 99.2% obtained for overall accuracy, TPR, TNR, PPV and NPV, respectively.

Fig. 14a shows the map of impacted area that was generated by the proposed change detection approach. Visually, the map aligns well with the tornado track. To further quantify the performance, the reference data described in Section 2.2 was overlaid on the impacted-area map (Fig. 14b). We can see that the boundary of FEMA data generally aligns well with our generated map. The model correctly classified 3722 out of 4226 destroyed or major-damaged structures identified by FEMA that fall in the impacted area, leading to 88.1% as correctness.

Fig. 15 shows the pre-tornado and post-tornado images along with the predicted impacted area and FEMA reference data for two zoomed-in areas in Fig. 14. Those zoomed-in areas show that the map covers most FEMA points without introducing many false positives even though obvious radiometric differences between pre- and post-tornado images exist for the unaffected area, which further confirms the effectiveness of our proposed method.

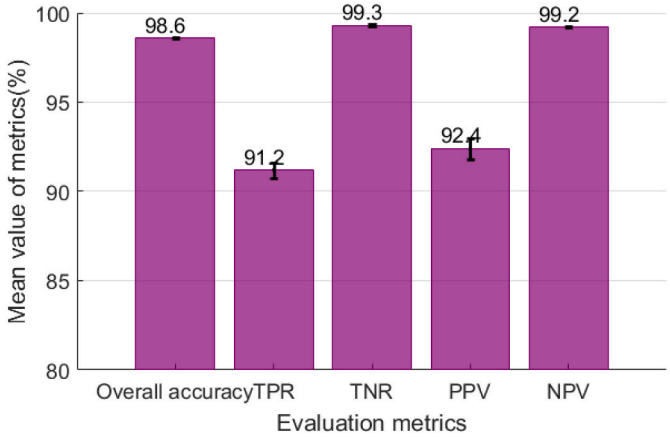

Fig. 13. Tenfold cross-validation of impacted area mapping model.

6. Discussion

6.1. Pixel-based and object-based change detection approaches

Recently, pixel-based methods using differencing and LSTM as feature fusion approaches in Zhang and Lu (2019) and Mou et al. (2019), respectively, have shown better performance than existing methods. In this paper, for the first time, an object-based change detection model with a CNN end-to-end learning architecture has been proposed and compared against a pixel-based method. Our experiments demonstrated that our proposed object-based approach performed consistently and substantially better than the pixel-based method, regardless of the types of the feature fusion strategies. Such favorable accuracy is not only significant in the quantitative assessment reported in Fig. 7 but is also visually noticeable from results shown in Fig. 12. This may be because the image segments are generally larger than the 5×5 pixel size, the size of image patch used in the pixel-based approach. Therefore, the image segments used in the object-based method contain more information than what was used for the pixel-based method, allowing the CNN to extract more meaningful features. Even though increasing the size of image patches may improve the accuracy, the advantage of object-based methods in this regard is that they do not require the user to choose the size of the image patches, because the image patch size for object-based methods is guided by the size of objects.

In addition to the higher accuracy, the proposed object-based method also showed considerably higher computational efficiency than pixel-based methods during the map generation procedure. With the same computational resources (Nvidia K80 GPU with 12GB), it took 110 s to generate the map for Fig. 12(d) with the object-based method, whereas 18,000 s was required to produce Fig. 12(c) with the pixel-based method. A significant 160× increase in speed over the pixel-based method is offered by exploiting the proposed object-based change analysis. The speedup comes from the fact all the pixels within a given object received the same classification label that resulted from a single inference by classifier, whereas each of those pixels needs individual inference with the pixel-based method. The computational advantage is especially important for large-scale change detection using high- or very high-resolution remote sensing imagery, which is becoming increasingly available today and important for such applications as detailed as building footprints.

Even though the object-based method comes with a segmentation procedure as overhead computing cost compared with pixel-based methods, our experiments indicate that segmentation is fast. We timed the segmentation procedure of site III and showed that segmenting all 52 tiles in site III took less than 1 min using the SLIC segmentation method and a machine with 24 computation cores. The dimension of each tile is 2000×2000 pixels, and all 52 tiles cover 80 km^2. The Quickshift algorithm is relatively slower, with one tile taking 114 s and 52 tiles taking 583 s in parallel.

Because the segmentation was conducted using the 1999 orthophoto only, it did not catch the straight lines of new building boundaries in orthophoto 2017. This would cause the boundaries for NB objects to be indistinguishable in the change map derived by the object-based method, which is highlighted in the orange box in Fig. 12. However, if the building footprints exist in the 1999 orthophoto, the boundaries of those buildings can be successfully captured by segmentation. Therefore, BB objects are free from this negative impact, as highlighted by the green box in Fig. 12. Note that such negative impacts by segmentation using single orthophotos are expected to be alleviated by performing segmentation using orthophotos from both years, and it requires verification in future study.

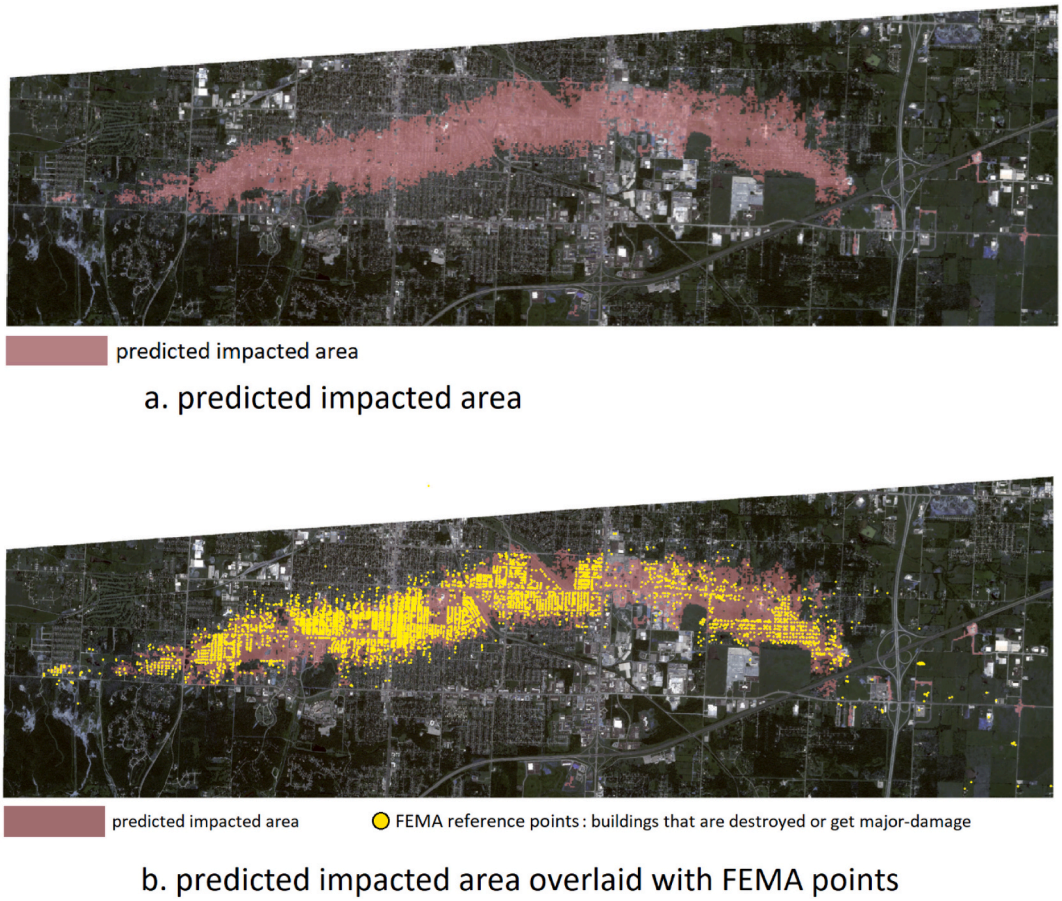

Fig. 14. The map of a) impacted area and b) FEMA reference dataset overlaid on the impacted area.

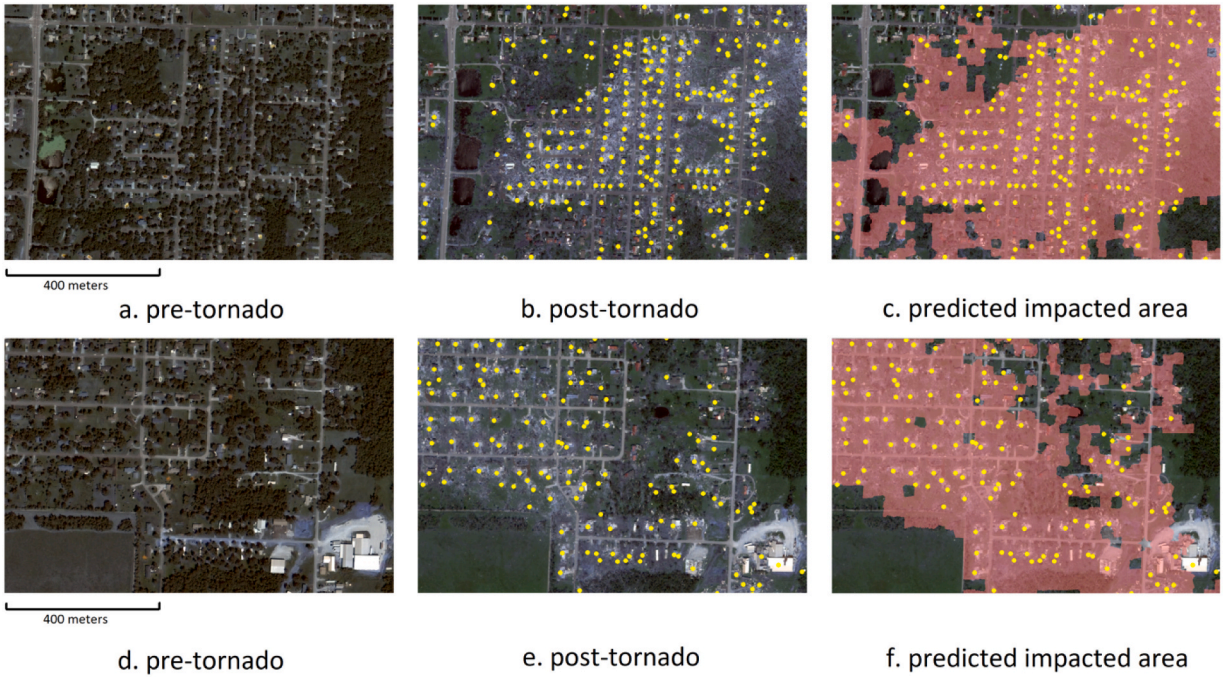

Fig. 15. Pre-tornado (first column) and post-tornado (second column) images for three zoomed-in areas of Fig. 14. In the third column, red color represents the predicted impacted area, and the yellow dots show the locations of destroyed structures and major damage structures identified by FEMA. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

6.2. Robustness of the proposed method toward misregistration

The impact of misregistration observed in bitemporal images on change detection accuracy has been studied before and described in several publications (Chen et al., 2014; Dai and Khorram, 1998; Townshend et al., 1992). However, this study, for the first time, systematically evaluated the impact of misregistration on change detection accuracy under a supervised change detection setting. The robustness to misregistration errors was assessed on both testing and training datasets through the special experiment design shown in Fig. 6. Our experiments indicate that our proposed method is robust to misregistration for both testing and training datasets, as evidenced in Fig. 9 and Fig. 10. The robustness may be attributed to the max pooling units which "take input from patches that are shifted by more than one row or column, thereby reducing the dimension of the representation and creating an invariance to small shifts and distortions" (LeCun et al., 2015). In addition, the convolutional operation can be considered as a procedure to summarize local statistics of an image patch, and the weight sharing characteristic of the convolutional filters makes these statistics less sensitive to location (LeCun et al., 1989; Raj et al., 2016), which may also contribute to the robustness of our proposed method. Additionally, the proposed object-based method used homogeneous image segments as analysis units, which allows the same feature pattern to be detected from those image segments, even if they are slighted shifted.

It should be noted that even though the misregistration does not seem to impact the overall accuracy to a significant level when the translation of misregistration is as large as 8 pixels with feature concatenation, the proposed object-based method is not completely free from the influence of image misregistration. A slight downward trend in overall accuracy is visible for all three feature fusion methods as the translation distance of misregistration error becomes larger in testing data, as shown in Fig. 9. Regarding the classifier trained with misregistered training data, highest overall accuracy tends to happen to the testing data that has the same level of misregistration with training data (Fig. 10).

LSTM was proposed to fuse change detection features in previous studies (Mou et al., 2019), but our experiments indicate LSTM does not perform better than the simple concatenation method (Table 3). LSTM was originally developed to overcome the short-term memory problem of recurrent neural networks (RNN). Because two time steps represent the shortest term of time series, an LSTM that is designed to summarize information in long-term sequences may lead to information loss compared with direct concatenation. Therefore, we think direct concatenation might be a better choice to fuse bitemporal features for change detection, as evidenced by the insignificant but slightly higher accuracy obtained by concatenation than LSTM in Table 3.

6.3. Binary change detection

In addition to evaluating the proposed change detection method in a multi-class change detection application in sites I and II using aerial images, this study also assessed the proposed method in a binary class change detection application using satellite images. Leveraging the speed benefits from this object-based framework, the whole procedure took 275 min to finish the processing of pre- and post-tornado VHR satellite images covering a 80 km^2 area with 416 million pixels. Compared with the pixel-based method, image segmentation is the overhead computation for object-based method. Starting from pre- and post-event image scenes and ending with the output of change map, it took 275 min in total for a DGX-1 machine with 80 cores to finish all the computation. Such change mapping capability would benefit government agencies and other entities by helping them quickly identify infrastructure that has been destroyed or received major damage due to a natural disaster and assist them in arranging humanitarian assistance accordingly. To evaluate the performance of the proposed model on this end, the reference prepared by FEMA was overlaid on the map of the impacted area and showed 88.1% correctness. It should be noted that because minor damage that happened to the vertical structures of the buildings (e.g., wall and window) was not visible from the nadir view of VHR satellite images even for the human eye, the proposed method is not expected to detect minor damage. The pixel-based method was not implemented for site III, because its relatively poor performance in terms of accuracy and computational efficiency in comparison with the proposed method has been demonstrated by experiments in sites I and II.

SIFT is designed to be robust to the radiometric difference between pre- and post-event images, and NDVI is based on the relative difference between near infrared and red bands; therefore, radiometric correction, which is generally required by many change detection methods as a preprocessing procedure (Chen et al., 2014), is not necessary for our method. The robustness of SIFT and NDVI allows the creation of many high-quality unchanged training samples automatically, regardless of radiometric difference between pre- and post-tornado images. On the other hand, the changed sampling area generation is based on the density of extracted unchanged points; therefore, it is also free from the impact of radiometric misregistration.

Like the standard object classification applications using CNN, the proposed change detection method requires that the input image patch be resized to have the same dimension before being input into the CNN modules due to the use of the fixed-size fully connected layer in the model. The dimension of the input is considered as a hyper parameter of the model and was determined based on our empirical experience. According to previous studies (Liu and Abd-Elrahman, 2018), for the land cover task the input dimension closer to the average area of image objects tends to produce higher accuracy. In our study, for sites I and II we randomly selected a subset of the objects and calculated the average size of objects as 164 square meters, which corresponds to the area of a 64 × 64 square widow with a pixel size of 0.2 m. Our experiments indicate that the model performance does not seem to be very sensitive to this hyperparameter. For example, in site III the average size of objects is 571 square meters, and it is translated to a square window with 37 × 37 pixels with a pixel size of 0.62 m. We still use the 64 × 64 window for site III experiments, and we achieved decent results, as shown in Figs. 13 and 15.

7. Future study

Thanks to the progress of deep learning techniques, developing an operational algorithm to map the building footprints at large scale has become a reality, as demonstrated by researchers from ORNL (Yang et al., 2018) and Microsoft (Microsoft, 2020). In the future, the proposed object-based change detection can be directly conducted on the building footprints, with each building footprint readily available in existing databases and being treated as objects. This not only avoids the image segmentation step completely but also potentially allows more accurate change detection on building level.

While we did not make model generalization as the focus of this study due to the length limit of the paper, we believe improving the generalization is important for the practical use of the change detection model. In our extended experiments, which are not shown in our study, it's demonstrated that our model has good generalization performance from site I to site II, but did not generalize well from site I to site III since site I differs from site III in terms of sensor type and definitions of change. In the future, it is worthwhile to investigate how the analysis unit, feature representation and feature fusion methods impact the generalization power of the change detection model.

8. Conclusion

A novel change detection approach was proposed in this study by the integration of CNN and OBIA frameworks. Based on extensive experimentation, our proposed object-based change detection methods were shown to substantially outperform the pixel-based benchmark methods

in terms of both accuracy and computational efficiency. It was also demonstrated that three change feature fusion methods do not show a significant difference of accuracy, whereas the concatenation method gave a slightly higher overall accuracy than LSTM and differencing feature fusion methods for the proposed change detection approach. In addition, our study indicates the proposed method is robust to misregistration errors in both testing and training data. Finally, we used a real-world application to confirm that the proposed change detection method is suitable for rapidly mapping the impacted area caused by natural disaster.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships, that could have appeared to influence the work reported in this paper.

Acknowledgments

This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan). Finally, we thank the editors and five anonymous reviewers for their valuable comments, which helped improve the quality of this manuscript substantially.

References

Abuelgasim, A., Ross, W., Gopal, S., Woodcock, C., 1999. Change detection using adaptive fuzzy neural networks: environmental damage assessment after the Gulf War. Remote Sens. Environ. 70, 208–223.
Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., 2012. SLIC superpixels compared to state-of-the-art superpixel methods. IEEE Trans. Pattern Anal. Mach. Intell. 34, 2274–2282.
Benedek, C., Szirányi, T., 2009. Change detection in optical aerial images by a multilayer conditional mixed Markov model. IEEE Trans. Geosci. Remote Sens. 47, 3416–3430.
Blaschke, T., Hay, G.J., Kelly, M., Lang, S., Hofmann, P., Addink, E., Feitosa, R.Q., van der Meer, F., van der Werff, H., van Coillie, F., 2014. Geographic object-based image analysis–towards a new paradigm. ISPRS J. Photogramm. Remote Sens. 87, 180–191.
Boldt, M., Thiele, A., Schulz, K., 2012. Object-based urban change detection analyzing high resolution optical satellite images. In: Earth Resources and Environmental Remote Sensing/GIS Applications III. International Society for Optics and Photonics, p. 85380E.
Celik, T., 2009. Unsupervised change detection in satellite images using principal component analysis and $ k $-means clustering. IEEE Geosci. Remote Sens. Lett. 6, 772–776.
Chen, G., Hay, G.J., Carvalho, L.M., Wulder, M.A., 2012. Object-based change detection. Int. J. Remote Sens. 33, 4434–4457.
Chen, G., Zhao, K., Powers, R., 2014. Assessment of the image misregistration effects on object-based change detection. ISPRS J. Photogramm. Remote Sens. 87, 19–27.
Cleve, C., Kelly, M., Kearns, F.R., Moritz, M., 2008. Classification of the wildland–urban interface: a comparison of pixel-and object-based classifications using high-resolution aerial photography. Comput. Environ. Urban. Syst. 32, 317–326.
Comber, A., Fisher, P., Wadsworth, R., 2004. Assessment of a semantic statistical approach to detecting land cover change using inconsistent data sets. Photogramm. Eng. Remote Sens. 70, 931–938.
Dai, X., Khorram, S., 1998. The effects of image misregistration on the accuracy of remotely sensed change detection. IEEE Trans. Geosci. Remote Sens. 36, 1566–1577.
Desclée, B., Bogaert, P., Defourny, P., 2006. Forest change detection by statistical object-based method. Remote Sens. Environ. 102, 1–11.
Duro, D., Franklin, S., Dubé, M., 2013. Hybrid object-based change detection and hierarchical image segmentation for thematic map updating. Photogramm. Eng. Remote Sens. 79, 259–268.
Ehlers, M., Sofina, N., Filippovska, Y., Kada, M., 2014. Automated techniques for change detection using combined edge segment texture analysis, GIS, and 3D information. In: Global Urban Monitoring and Assessment Through Earth Observation. CRC Press, pp. 346–373.
El-Kawy, O.A., Rød, J., Ismail, H., Suliman, A., 2011. Land use and land cover change detection in the western Nile delta of Egypt using remote sensing data. Appl. Geogr. 31, 483–494.
FEMA, 2016. Damage Assessment Operations Manual. https://www.fema.gov/media-library/assets/documents/109040.
FEMA, 2019. Historical Damage Assessment Database. https://communities.geoplatform.gov/disasters/historical-damage-assessment-database/.
Fu, B., Wang, Y., Campbell, A., Li, Y., Zhang, B., Yin, S., Xing, Z., Jin, X., 2017. Comparison of object-based and pixel-based Random Forest algorithm for wetland vegetation mapping using high spatial resolution GF-1 and SAR data. Ecol. Indic. 73, 105–117.
Gao, P., Trettin, C.C., Ghoshal, S., 2012. Object-oriented segmentation and classification of wetlands within the Khalong-la-Lithuny a catchment, Lesotho, Africa. In: Geoinformatics (GEOINFORMATICS), 2012 20th International Conference on. IEEE, pp. 1–6.
Gong, J., Yue, Y., Zhu, J., Wen, Y., Li, Y., Zhou, J., Wang, D., Yu, C., 2012. Impacts of the Wenchuan Earthquake on the Chaping River upstream channel change. Int. J. Remote Sens. 33, 3907–3929.
Gong, M., Zhan, T., Zhang, P., Miao, Q., 2017. Superpixel-based difference representation learning for change detection in multispectral remote sensing images. IEEE Trans. Geosci. Remote Sens. 55, 2658–2673.
Greff, K., Srivastava, R.K., Koutník, J., Steunebrink, B.R., Schmidhuber, J., 2016. LSTM: a search space odyssey. IEEE Trans. Neural Networks Learn. Syst. 28, 2222–2232.
He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.
Healey, S.P., Cohen, W.B., Yang, Z., Brewer, C.K., Brooks, E.B., Gorelick, N., Hernandez, A.J., Huang, C., Hughes, M.J., Kennedy, R.E., 2018. Mapping forest change using stacked generalization: an ensemble approach. Remote Sens. Environ. 204, 717–728.
Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9, 1735–1780.
Ioffe, S., Szegedy, C., 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. arXiv preprint. arXiv:1502.03167.
Karpatne, A., Jiang, Z., Vatsavai, R.R., Shekhar, S., Kumar, V., 2016. Monitoring land-cover changes: a machine-learning perspective T2 - IEEE geoscience and remote sensing magazine. IEEE Geosci. Remote Sens. Magazine 4, 8.
Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105.
Lambin, E.F., Strahlers, A.H., 1994. Change-vector analysis in multitemporal space: a tool to detect and categorize land-cover change processes using high temporal-resolution satellite data. Remote Sens. Environ. 48, 231–244.
LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L. D., 1989. Backpropagation applied to handwritten zip code recognition. Neural Comput. 1, 541–551.
LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436–444.
Liu, T., Abd-Elrahman, A., 2018. Deep convolutional neural network training enrichment using multi-view object-based analysis of unmanned aerial systems imagery for wetlands classification. ISPRS J. Photogramm. Remote Sens. 139, 154–170.
Liu, X., Lathrop Jr., R., 2002. Urban change detection based on an artificial neural network. Int. J. Remote Sens. 23, 2513–2518.
Liu, T., Yang, L., 2020. A fully automatic method for rapidly mapping impacted area by natural disaster. In: 2020 IEEE International Geoscience and Remote Sensing Symposium(accepted). IEEE.
Liu, J., Gong, M., Qin, K., Zhang, P., 2016. A deep convolutional coupling network for change detection based on heterogeneous optical and radar images. IEEE Trans. Neural Networks Learn. Syst. 29, 545–559.
Liu, T., Abd-Elrahman, A., Jon, M., Wilhelm, V.L., 2018. Comparing fully convolutional networks, random Forest, support vector machine, and patch-based deep convolutional neural networks for object-based wetland mapping using images from small unmanned aircraft system. GISci. Remote Sens. 55 (2), 243–264.
Lunga, D., Yang, H.L., Reith, A., Weaver, J., Yuan, J., Bhaduri, B., 2018. Domain-adapted convolutional networks for satellite image classification: a large-scale interactive learning approach. IEEE J. Select. Topics Appl. Earth Observ. Remote Sens. 11, 962–977.
Lyu, H., Lu, H., Mou, L., 2016. Learning a transferable change rule from a recurrent neural network for land cover change detection. Remote Sens. 8, 506.
Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G., Johnson, B.A., 2019. Deep learning in remote sensing applications: a meta-analysis and review. ISPRS J. Photogramm. Remote Sens. 152, 166–177.
Microsoft, 2020. Computer Generated Building Footprints in US. https://github.com/microsoft/USBuildingFootprints.
Mou, L., Bruzzone, L., Zhu, X.X., 2019. Learning spectral-spatial-temporal features via a recurrent convolutional neural network for change detection in multispectral imagery. IEEE Trans. Geosci. Remote Sens. 57, 924–935.
Nielsen, A.A., Conradsen, K., Simpson, J.J., 1998. Multivariate alteration detection (MAD) and MAF postprocessing in multispectral, bitemporal image data: new approaches to change detection studies. Remote Sens. Environ. 64, 1–19.
NOAA, 2020. U.S. Climate Extremes Index (CEI): Graph. https://www.ncdc.noaa.gov/extremes/cei/graph/us/cei/01-12.
Pande-Chhetri, R., Abd-Elrahman, A., Liu, T., Morton, J., Wilhelm, V.L., 2017. Object-based classification of wetland vegetation using very high-resolution unmanned air system imagery. Eur. J. Remote Sens. 50, 564–576.

Raj, A., Gupta, S., Verma, N.K., 2016. Face detection and recognition based on skin segmentation and CNN. In: 2016 11th International Conference on Industrial and Information Systems (ICIIS). IEEE, pp. 54–59.
Ridd, M.K., Liu, J., 1998. A comparison of four algorithms for change detection in an urban environment. Remote Sens. Environ. 63, 95–100.
Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15, 1929–1958.
Townshend, J.R., Justice, C.O., Gurney, C., McManus, J., 1992. The impact of misregistration on change detection. IEEE Trans. Geosci. Remote Sens. 30, 1054–1060.
Van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., Yu, T., 2014. scikit-image: image processing in Python. PeerJ *2*, e453.
Vedaldi, A., Soatto, S., 2008. Quick shift and kernel methods for mode seeking. In: European Conference on Computer Vision. Springer, pp. 705–718.
Vu, T.T., Matsuoka, M., Yamazaki, F., 2004. LIDAR-based change detection of buildings in dense urban areas. In: IGARSS 2004. 2004 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 3413–3416.
Walter, V., 2004. Object-based classification of remote sensing data for change detection. ISPRS J. Photogramm. Remote Sens. 58, 225–238.
Wang, F., Xu, Y.J., 2010. Comparison of remote sensing change detection techniques for assessing hurricane damage to forests. Environ. Monit. Assess. 162, 311–326.
Wang, S., Ma, Q., Ding, H., Liang, H., 2018. Detection of urban expansion and land surface temperature change using multi-temporal landsat images. Resour. Conserv. Recycl. 128, 526–534.
Wold, S., Esbensen, K., Geladi, P., 1987. Principal component analysis. Chemom. Intell. Lab. Syst. 2, 37–52.
Yang, L., Xian, G., Klaver, J.M., Deal, B., 2003. Urban land-cover change detection through sub-pixel imperviousness mapping using remotely sensed data. Photogramm. Eng. Remote Sens. 69, 1003–1010.
Yang, H.L., Yuan, J., Lunga, D., Laverdiere, M., Rose, A., Bhaduri, B., 2018. Building extraction at scale using convolutional neural network: mapping of the United States. IEEE J. Select. Topics Appl. Earth Observ. Remote Sens. 11, 2600–2614.
Yuan, J., Yang, H.-H.L., Omitaomu, O.A., Bhaduri, B.L., 2016. Large-scale solar panel mapping from aerial images using deep convolutional networks. In: 2016 IEEE International Conference on Big Data (Big Data). IEEE, pp. 2703–2708.
Yuhas, R.H., Goetz, A.F., Boardman, J.W., 1992. Discrimination among Semi-Arid Landscape Endmembers Using the Spectral Angle Mapper (SAM) Algorithm.
Zhan, Y., Fu, K., Yan, M., Sun, X., Wang, H., Qiu, X., 2017. Change detection based on deep siamese convolutional network for optical aerial images. IEEE Geosci. Remote Sens. Lett. 14, 1845–1849.
Zhang, W., Lu, X., 2019. The spectral-spatial joint learning for change detection in multispectral imagery. Remote Sens. 11, 240.
Zhu, Z., Zhang, J., Yang, Z., Aljaddani, A.H., Cohen, W.B., Qiu, S., Zhou, C., 2019. Continuous monitoring of land disturbance based on Landsat time series. Remote Sens. Environ. 238 p.111116.