Long Short-Term Memory Neural Networks for Online Disturbance Detection in Satellite Image Time Series

Yun-Long Kong, Qingqing Huang *, Chengyi Wang, Jingbo Chen, Jiansheng Chen and Dongxu He

Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing 100101, China; kongyl@radi.ac.cn (Y.-L.K.); wangcy@radi.ac.cn (C.W.); chenjb@radi.ac.cn (J.C.); chenjs@radi.ac.cn (J.C.); hedx@radi.ac.cn (D.H.)
* Correspondence: huangqq@radi.ac.cn; Tel.: +86-10-6484-7442

Received: 27 December 2017; Accepted: 12 March 2018; Published: 13 March 2018

Abstract: A satellite image time series (SITS) contains a significant amount of temporal information. By analysing this type of data, the pattern of the changes in the object of concern can be explored. The natural change in the Earth’s surface is relatively slow and exhibits a pronounced pattern. Some natural events (for example, fires, floods, plant diseases, and insect pests) and human activities (for example, deforestation and urbanisation) will disturb this pattern and cause a relatively profound change on the Earth’s surface. These events are usually referred to as disturbances. However, disturbances in ecosystems are not easy to detect from SITS data, because SITS contain combined information on disturbances, phenological variations and noise in remote sensing data. In this paper, a novel framework is proposed for online disturbance detection from SITS. The framework is based on long short-term memory (LSTM) networks. First, LSTM networks are trained by historical SITS. The trained LSTM networks are then used to predict new time series data. Last, the predicted data are compared with real data, and the noticeable deviations reveal disturbances. Experimental results using 16-day compositions of the moderate resolution imaging spectroradiometer (MOD13Q1) illustrate the effectiveness and stability of the proposed approach for online disturbance detection.

Keywords: long short-term memory; LSTM; recurrent neural network; RNN; online disturbance detection; satellite image time series; SITS

1. Introduction

In recent years, the analysis of long time-series remote sensing data has attracted extensive attention from researchers who use multitemporal remote sensing images to construct time series and perform change detection using time series analysis methods [1]. Time series analysis is an important direction of data mining research and has been successfully employed in many fields, such as economics, medicine, meteorology, voice recognition and video analysis [2–6]. In the field of remote sensing, researchers worldwide have employed satellite image time series (SITS) in the dynamic monitoring of natural phenology, landscapes, disasters, agriculture, and other areas [7–10].

Based on their objectives, SITS change detection techniques can be grouped into two categories [11]: (1) methods oriented to detect land cover/use types change and (2) methods oriented to detect anomalous information due to sudden and unexpected events. The latter is used for detecting sudden changes, also called disturbances [12], caused by sudden and unexpected events (for example, deforestation, fires, plant diseases, and insect pests). Disturbance detection relies on detecting the anomalies in the time series curve. According to the timeliness, there are two types of methods used for detecting anomalous information in time series [13]: offline detection and online detection. Offline
Remote Sens. 2018, 10, 452

Detection assumes the time series already exists and identifies anomalous subseries by detecting changes in the characteristics of the time series. Online detection achieves the goal of detecting anomalies with minimum delay. Hence, online detection is real-time or near real-time detection. For disturbance detection, online detection is advantageous because, in many cases, detection of anomalies must be done in a timely manner so that counter-measures can be taken promptly (for example, responses to wildfire, flood, etc.).

A variety of time series analysis algorithms for online change detection have been successfully applied to SITS. Such algorithms include cumulative sum (CUSUM) [14], autoregressive integrated moving average (ARIMA) [15], weighted moving average (EWMA) [16], support vector machine (SVM) [17], random forest (RF) [17,18], hidden Markov model (HMM) [19], break detection for additive season and trend (BFAST) [12], and other algorithms.

Deep learning has flourished in recent years, and such approaches have been increasingly applied to time series analysis [20]. Deep learning uses a cascade of multiple processing layers to learn features or representations of data with different levels of abstraction [21]. Deep learning is good at discovering the patterns of intricate structures in non-linear and high-dimensional data with no or little prior assumptions. Deep learning is robust to outliers, missing data, and noise [22]. Among deep learning models, recurrent neural networks (RNN) have a stronger adaptability to time series analysis because they introduce the concept of sequences into the structure design of network units [23]. However, the vanishing gradient and exploding gradient problems make it difficult to capture long temporal data using traditional RNN models [24]. To overcome this difficulty, long short-term memory (LSTM) networks with special hidden units were proposed for learning the time series over long time periods [25]. The ability to remember information in long time series makes LSTM successful in various domains, such as speech recognition, video analysis, and biology [26–28].

LSTM has also been applied in the domain of remote sensing. Wu and Prased [29] applied a convolutional RNN in hyperspectral data classification. Rubwurm and Korner [30] extracted temporal characteristics from SITS based on LSTM and achieved outstanding classification accuracy. Lyu et al. [31] proposed an offline land cover change detection method using SITS. Taking advantage of the time series predictive ability of LSTM, You et al. [32] presented a framework for crop yield prediction using SITS. Prediction methods are suitable for online change detection. Nevertheless, the powerful LSTM model is still underutilised in SITS for online disturbance detection.

In this study, we propose an LSTM-based framework for online disturbance detection in SITS. With SITS data, we trained LSTM networks using historical time series, and predicted new data using the trained LSTM networks. The sequential per-pixel anomaly alarms can be found by comparing the predicted data and the real data. Based on the results, the temporal and spatial extent of disturbances can be assessed in real-time. The proposed framework was applied to a challenging forest fire detection case study using SITS data. Experimental results demonstrate that this approach can achieve excellent performance.

This paper is organised as follows: Section 2 provides descriptions of the experimental data and details the proposed framework based on LSTM. Section 3 shows the experiment result. Section 4 addresses the discussion about the experiment result. Conclusions are drawn in Section 5.

2. Materials and Methods

In this section, we present the framework for online disturbance detection in SITS based on LSTM in detail. Firstly, a case study, which is used to assist in demonstrating the process and test the performance of our framework, is described in Section 2.1. The algorithmic procedure is then demonstrated in Section 2.2.

2.1. Case Study

A dataset of forest fire is selected for validating our method, because a forest fire is a typical kind of disturbance. The case study was conducted on the catastrophic fire that occurred on the Yinanhe
Forest Farm located near in 128°30′E, 48°40′N Heilongjiang Province, China in 2009 (Figure 1). The fire started on 27 April 2009.

Figure 1. Burned area on the Yinanhe Forest Farm in Heilongjiang Province, China.

Figure 2 shows the false colour images of the forest farm composited from Landsat Thematic Mapper (TM) band 4, band 3 and band 2 data at three time points. Figure 2a shows the image of the forest farm before the fire. Figure 2b provides an image of the forest farm approximately one month after the fire occurred with distinct fire marks. Figure 2c shows an image of the forest farm approximately five months after the fire occurred. The colour differences of these images are caused by the differences in imaging conditions (for example, atmospheric condition, solar elevation angle, viewing angle). The images shown in Figure 2 illustrate the process of the forest fire. Serious damage was caused by the fire, and vegetation recovered rapidly after the forest fire.

The experimental data of this forest fire to test our framework is generated from a moderate-resolution imaging spectroradiometer (MODIS) land product, MOD13Q1. This product is a 16-day composite data of max vegetation index, with spatial resolution of 250 m. In this study, 207 MOD13Q1 images are used for the period from 1 January 2001 to 31 December 2009 (23 images per year), the tile of h26v04. The global environment monitoring index (GEMI) [33] was calculated using these data. All data were converted to World Geodetic System 84 ellipsoid projections in the 52 N zone in the Universal Transverse Mercator coordinate system. By stacking these images, each pixel in the time axis contains a GEMI time series with 207 time steps. We implement the LSTM-based disturbance detection algorithm on each time series, and a spatial result (burned area map) is generated from pixel-by-pixel detection.
2.2. Methodology

In the following, we introduce the LSTM firstly (Section 2.2.1). Based on LSTM networks, our framework for online disturbance detection involves training (Section 2.2.2), prediction (Section 2.2.3), and detection (Section 2.2.4).

2.2.1. Long Short-Term Memory (LSTM)

Compared to the conventional deep neural network, an improvement of RNN is that it treats the hidden layers as successive recurrent layers (Figure 3, left). This structure can be unfolded to produce the outputs of a neuron component sequence at discrete time steps corresponding to the input time series (Figure 3, right). With this architecture, RNN has a strong capability of capturing the historical information embedded in the past elements of the input for an amount of time [24].

\[
h_t = \sigma(Wx_t + Uh_{t-1} + b^{(h)})
\]

\[
o_t = \sigma(Vh_t + b^{(o)})
\]

Figure 3 shows the diagrammatic sketch of an RNN. The process of carrying memory forward can be described mathematically.
\[ a_t = \sigma \left( Wh_t + b^o \right) \]

where \( x_t \) is an input vector, \( h_{t-1} \) is the previous hidden state in RNN, \( h_t \) is the hidden state corresponding to \( x_t \), \( a_t \) is the output vector, \( W \), \( U \), and \( V \) are the weight matrices, \( b \) denotes bias vector, and \( \sigma \) is an activation function.

The architecture of RNN makes it good at making predictions for the next time step in a time series based on the input data at the current time step. However, it is challenging to train the RNN using backpropagation for long time series. The growth or shrinkage of the backpropagated gradients at each time step can accumulate. This accumulation causes gradients to explode or vanish over many time steps [24]. Long short-term memory is introduced to overcome this challenge to ensure the RNN has long-term memory ability [25]. Long short-term memory allows the network to capture information from inputs for a long time using a special hidden unit, called the LSTM cell, instead of a simple RNN unit (the green node \( h \) shown in Figure 3) in the hidden layer.

In this study, we use the LSTM cell as described in [34]. The structure of an LSTM cell is shown in Figure 4. It contains forget gate \( (f_t) \), input gate \( (i_t) \), input modulation gate \( (m_t) \), output gate \( (o_t) \), memory cell \( (c_t) \), and hidden state \( (h_t) \). The gates are computed by:

\[ f_t = \sigma(W^f x_t + U^f h_{t-1} + b^f) \] (forget gate)  
\[ i_t = \sigma(W^i x_t + U^i h_{t-1} + b^i) \] (input gate)  
\[ m_t = \tanh(W^m x_t + U^m h_{t-1} + b^m) \] (input modulation gate)  
\[ o_t = \sigma(W^o x_t + U^o h_{t-1} + b^o) \] (output gate)  

where \( x_t \) is an input vector, \( h_{t-1} \) is the previous hidden state in LSTM network, \( W \) and \( U \) are the weight matrices, \( \sigma \) is the logistic sigmoid function, \( \tanh \) is the hyperbolic tangent function and \( b \) is the bias vector. From these gates, the memory cell \( (c_t) \) and hidden state \( (h_t) \) are calculated as:

\[ c_t = i_t \cdot m_t + f_t \cdot c_{t-1} \] (7)  
\[ h_t = a_t \cdot \tanh(c_t) \] (8)

where “\( \cdot \)” means pointwise multiplication of two vectors. The memory cell \( (c_t) \) contains two parts: (1) the information of the previous memory cell \( (c_{t-1}) \) modulated by the forget gate \( (f_t) \), (2) the information of the current input \( (x_t) \) and previous hidden state \( (h_{t-1}) \) modulated by the input modulation gate \( (m_t) \). The forget gate \( (f_t) \) allows the LSTM to selectively forget its previous memory \( (c_{t-1}) \). The input gate \( (i_t) \) controls the LSTM to consider the current input. The input modulation gate \( (m_t) \) modulates the information of the input gate \( (i_t) \). The output gate \( (o_t) \) controls the memory cell \( (c_t) \) to transfer to the hidden state \( (h_t) \).
The LSTM network, a recurrent neural network consisting of LSTM cells, is capable of learning the fluctuation pattern in long-term time series and then making predictions.

2.2.2. Training LSTM Using SITS

In this study, the SITS described in Section 2.1 were used to train the LSTM networks. Each one-dimensional GEMI time series corresponding to a pixel of the stacked images in the time axis was used to train one LSTM network independently at a time. All of the SITS with 207 time steps (from 1 January 2001 to 31 December 2009) were divided into two parts for online disturbance detection: (1) long historical time series with the first 184 time steps (from 1 January 2001 to 31 December 2008) as inputs for training LSTM, and (2) test time series with the last 23 time steps (from 1 January 2009 to 31 December 2009) for verifying the LSTM prediction and for detecting anomalies.

A sliding time window-based method advised in [35] is used to obtain a number of samples for training LSTM from a single historical SITS. For a time series \((x_1, x_2, \ldots, x_n)\) with \(n\) time steps, assuming the size of the time window is \(l\) \((l < n)\), the first sample is input subsequence \((x_1, x_2, \ldots, x_l)\) for output \(x_{l+1}\). The time window slides one time step ahead at a time to generate one training sample. The subsequence \((x_{t-l}, \ldots, x_{t-2}, x_{t-1})\) in the time window corresponds to the output \(x_t\) one time step ahead. Thus, \(n - l + 1\) samples can be obtained for training the LSTM network. For the LSTM hidden layers in this study, a dropout layer with a dropout rate of 0.5 was added to prevent over-fitting [36]. In addition, the mean square error (MSE) loss function and the root mean square propagation (RMSprop) optimiser [37] were used for LSTM training.

2.2.3. Prediction Based on Trained LSTM

Depending on the training method described in the previous section, LSTM can only predict data one step at a time. With an \(l\)-step sliding time window, the output \(y_{t+1}\) would be predicted using the input sequence \((x_{t-l+1}, \ldots, x_{t-1}, x_t)\).

A strategy to implement prediction of \(n\) steps is performed iteratively using the predictive outputs of the previous steps to compose the input sequence to the next step. Therefore, the output \(y_{t+n}\) would be predicted using the input sequence \((x_{t+n-l}, \ldots, x_t, y_{t+1}, \ldots, y_{t+n-1})\).

The method of one-step prediction and the method of multistep prediction are both implemented and compared in this study.

2.2.4. Online Disturbance Detection

If a disturbance does occur, its impact on the ground will be manifested by the corresponding changes in SITS. Therefore, compared to the predicted data that contains a regular fluctuation pattern, an anomaly would have a significant deviation from the real data. In this study, we set a threshold on the deviation between the predicted data and the real data to detect possible anomalies.

The detected anomalies may be false alarms caused by noise. To reduce false alarms, an alternative strategy is to use newly acquired data to verify the detected anomalies. The impact of disturbances on Earth usually lasts for a long time (months or years). Thus, detecting continuous deviations greater than the threshold between the predicted data and the newly acquired data can confirm that the disturbances did indeed occur. The cost of decreasing false alarms is to reduce timeliness.

3. Results

In this section, we implement the framework presented in Section 2.2 on the data of the case study to test the effectiveness of our method. Firstly, we use a sample GEMI time series to demonstrate the process of our method (Section 3.1). Then we generate the spatial result (Section 3.2) by applying the method to the SITS pixel by pixel in the stacked images described in Section 2.1.
3.1. Forest Fire Detection in SITS

An example GEMI time series affected by the forest fire was used to illustrate the process of detection. For demonstrating the process more clearly, we employed a composite time series instead of a single-pixel SITS. A $10 \times 10$ pixel window was set in the burned area firstly. From the window, 100 time series affected by the forest fire, corresponding to 100 pixels, were obtained. By calculating the average values of the 100 time series at each time step, a mean-value time series was generated. The composite time series, shown in Figure 5, is a typical GEMI time series affected by the forest fire with less noise. The time series covers the period from 2001 to 2009. The historical time series from 2001 to 2008 (blue series in Figure 5) was used for training the LSTM network. The real GEMI time series in 2009 (red series in Figure 5) was used to detect the forest fire.

![Figure 5. The example GEMI time series.](image)

As a vegetation index, GEMI has a significant seasonal variation pattern in one year. Sufficiently learning this seasonal pattern is necessary for making accurate predictions based on the LSTM. For the sliding time window-based training, the size of the time window should cover one cycle of seasonal variation. In this case, 16-day composed data has 23 time steps in one year. Therefore, a sliding time window with a size of 23 time steps was used to train the LSTM. As described in Section 2.2.3, the LSTM predicts data one time step ahead in general, and a strategy with self-iteration can make multistep prediction. In this study, we tested one-step prediction and multistep prediction at the same time. The performance of these two methods was compared. The predictions made by LSTM using the 2001–2008 time series (Figure 5) are shown in Figure 6. The red series is the one-step prediction. The yellow series is the 23-step prediction using the method described in Section 2.2.3. The blue series is the real data. The results of the two prediction methods were similar. Along with more time steps, the result of prediction for one step ahead was adjusted by the real data and became closer to the real data. However, in this case the real data contains anomalies caused by the forest fire. The anomalies affect the result of the prediction one step ahead to make the wave peak appear earlier. Therefore, the result of prediction for 23 steps ahead reflects more of the seasonal fluctuation pattern without the impact of anomaly data.

Disturbances in forests (for example, forest fires, insects, and deforestations) always resulted in a decline in the vegetation index. Thus, the abrupt decreasing of GEMI in the time series is useful for disturbance detection. If the observation GEMI is lower than the predicted GEMI by a threshold, a disturbance alarm would be detected. With the strong influence of the forest fire, the threshold was set as 0.1, which is about 1.5 times the mean absolute error of the trained LSTM network in this study. Figure 7 shows the detection result in the example GEMI time series. As the deviations greater than
the threshold start at the ninth step both in Figure 7a,b, the two prediction methods detected a disturbance at the ninth time step. This disturbance could be further verified by more observation data.

![Real data and predicted data in 2009.](image)

**Figure 6.** Real data and predicted data in 2009.

![Comparison between the real data and the predicted data in 2009: (a) the prediction for one step ahead; and (b) the prediction for 23 steps ahead.](image)

**Figure 7.** Comparison between the real data and the predicted data in 2009: (a) the prediction for one step ahead; and (b) the prediction for 23 steps ahead.

This detected disturbance was further verified with ancillary data. The forest fire in the study area started on 27 April 2009. This date corresponds to the period of the eighth step (from 23 April to 8 May). However, the occurrence of the forest fire was detected at the ninth step of 2009 (from 9 May to 24 May). The time of the occurrence of the fire determined based on the detection results was one time step (16 days) behind the actual start time of the forest fire. The MOD13Q1 data were synthesised from the maximum value of the vegetation index and the fire occurred at the late stage of the time period covered by the eighth step data. Thus, the data at the eighth step were probably synthesised from the data before the fire occurred. As a result, the detected disturbance time lagged behind the actual disturbance time. Therefore, the temporal accuracy of this disturbance detection method is limited by the synthesis methods used to generate the time series and the temporal resolution of the original data.
3.2. Spatial Application

The process described in Section 3.1 was applied to the MODIS data described in Section 2.1 pixel by pixel. A map of the detected burned area was obtained.

The ground truth data indicating the exact spatial extent of this fire were not available. A single Landsat TM image was used to identify the spatial extent of the forest fire for validating the detection result. The Landsat TM image was shot on 23 May 2009. Using visual interpretation, a burned area map with low omission error was generated by choosing the pixels with a burn area index (BAI) [38] less than 1.15. The commission errors were then reduced by manual modification with visual interpretation. A total of 769.95 km² of burned area was identified in the validation image.

The burned area detected by our method and the validation image were vectorised, respectively. The vectorised maps were then overlaid and compared. The comparison maps are shown in Figure 8a,b. The red areas signify the correct detection results (true burned areas). The yellow areas signify the areas that were detected by the algorithm proposed in this study but were not detected by the semi-automatic interpretation method (falsely-detected burned areas). The orange areas signify the areas that were detected by the validation image, but which were not detected by the algorithm based on LSTM (omitted burned areas).

![Comparison between the results based on LSTM and the Landsat TM validation image: (a) Detection based on LSTM with the prediction for one step ahead; (b) detection based on LSTM with the prediction for 23 steps ahead; and (c) detection based on BFAST.](image)

Figure 8. Comparison between the results based on LSTM and the Landsat TM validation image: (a) Detection based on LSTM with the prediction for one step ahead; (b) detection based on LSTM with the prediction for 23 steps ahead; and (c) detection based on BFAST.

In addition, another detection method using the prediction based on BFAST [12] was also implemented for comparison with our methods. BFAST is an excellent algorithm specially designed for the change detection, including disturbance detection, of SITS. The detection result based on BFAST was also validated by the Landsat TM validation image (Figure 8c).

The validation results for the methods mentioned above are shown in Table 1.

<table>
<thead>
<tr>
<th></th>
<th>One-Step LSTM Prediction</th>
<th>23-Steps LSTM Prediction</th>
<th>BFAST Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Burned area (km²)</td>
<td>838.75</td>
<td>860.88</td>
<td>827.38</td>
</tr>
<tr>
<td>Correct detection area (km²)</td>
<td>529.61</td>
<td>550.21</td>
<td>542.60</td>
</tr>
<tr>
<td>Correct detection rate</td>
<td>68.8%</td>
<td>71.5%</td>
<td>70.5%</td>
</tr>
<tr>
<td>False detection rate</td>
<td>36.9%</td>
<td>36.1%</td>
<td>34.4%</td>
</tr>
<tr>
<td>Total detection accuracy</td>
<td>85.3%</td>
<td>85.9%</td>
<td>86.3%</td>
</tr>
</tbody>
</table>

Table 1. Validation results between the detection results and the validation image.

The comparison between the result based on LSTM prediction for one step ahead and the validation image is shown in Figure 8a. A burned area of 838.75 km² was detected using LSTM prediction for one
step ahead. The correct detection area was $529.61 \text{ km}^2$. This result has a correct detection rate of 68.8%, a false detection rate of 36.9%, and a total detection accuracy of 85.3%.

Another detection map was generated by LSTM prediction for 23 steps ahead. This method detected a burned area of $860.88 \text{ km}^2$. Compared with the visual interpretation image, the correct detection area was $550.21 \text{ km}^2$. The result showed an overall accuracy of 85.9%, a burned area detection accuracy of 71.5% with commission error of 36.1%. The comparison is shown in Figure 8b.

The detection method based on BFAST detected a burned area of $827.38 \text{ km}^2$. Compared with the validation image, the result had a correct detection rate of 70.5% with 542.60 km$^2$ of the correct detection area, a false detection rate of 34.4%, and a total detection accuracy of 86.3%.

4. Discussion

In addition to the errors originating from the methods, there are also errors that resulted from the visual misinterpretation of the TM image, the much coarser resolution of the MODIS data and the registration errors. For online detection, a relatively high false alarm rate is acceptable. These results detected on LSTM show that a serious disturbance occurred and can generate a strong alert. Assessments with higher precision can be further carried out by post-processing with more observation data.

Comparing the two detection methods based on the LSTM, the evaluation indicates that multistep prediction is better than one time step prediction using LSTM for disturbance detection. This is because the one-step prediction is constantly affected by the observation data. Conversely, the multistep prediction reflects more of the fluctuation pattern learned from the historical data, which is conducive to discovering anomalies.

The detection based on the multistep prediction using LSTM has a similar accuracy with the powerful detection method based on BFAST. The correct detection rate of the method based on LSTM is a litter higher, while the method based on BFAST has a higher total detection accuracy with a lower false detection rate. The disturbance detection strategy of BFAST is monitoring structural change [12]. By using more observation data, structural change detection is beneficial to decreasing false alarms. However, in this study, the rapid recovering of the vegetation reduces structural changes caused by the forest fire in the GEMI time series. The correct detection of BFAST is also affected by structural change detection.

The online disturbance detection based on BFAST performs effectively in this case. Using trigonometric functions to fit the seasonal component, BFAST is powerful for detecting disturbances under the influence of seasonal changes, but is limited to areas with vegetation cover [12]. For the SITS without obvious seasonal patterns, such as the observations of water, the effect of BFAST is not good enough. Our method based on LSTM has potential for more extensive applications. Since LSTM is also capable of learning and predicting time series without seasonal patterns [22]. Figure 9 shows the real data and predictions of a time series with water in 2009. The blue series is the real data. The pattern of the real data is difficult to fit with trigonometric functions. Therefore, the prediction based on BFAST is unsatisfactory (red series). The yellow series is the prediction based on LSTM, and has a similar pattern with the real data.

Considering some problems for disturbance detection in SITS and exploiting other powerful abilities of LSTM, further work can be discussed in detail.

1) An assumption for our proposed method is that the historical data is free of disturbances. This assumption limits the applications of the method. The disturbances occurred during the period of training time series affect the training of the LSTM networks and reduced the accuracy of the prediction further. BFAST has strategies for selecting the training time series without disturbances and detecting multiple disturbances in a time series. Combining the strategies contributes to improving the detection capability and accuracy of the algorithm based on LSTM.

2) We used the MOD13Q1 data in this study because the data could provide time series of equal time intervals with less invalid data. As the vegetation recovered rapidly after the forest fire in this
case, using a max vegetation index in 16 days increases the difficulty of detection. Using different data with different spatial resolutions (such as 30 m of Landsat images) and different temporal resolutions (such as one day of MODIS data), LSTM can be applied more broadly. However, a problem is data contamination. Cloud and noise are common in remote sensing images. Missing data in SITS is generated by excluding cloud and noise. It is difficult to obtain SITS without missing data from high spatial or temporal resolution images. Many algorithms fail to analyse time series with missing data. LSTM can learn from training time series with missing data. Thus, our method has the potential to be extended to detect disturbances using unequal time interval SITS. Such an extension is useful for disturbance detection from data with higher spatial resolutions and higher temporal resolutions.

(3) Our method in this study used only temporal information in SITS. The accuracy of disturbance detection in a single time series is easily affected by random noise. Remote sensing images contain abundant spatial information. Combining spatial information is beneficial to improve the stability of detection. A strategy to combine spatial information is using adjacent pixels to construct the multidimensional time series. LSTM has the ability of learning and predicting high-dimensional data. Further work can focus on using multidimensional SITS for disturbance detection.

5. Conclusions

In this paper, we introduced an approach that applies the LSTM neural networks to online disturbance detection in SITS. The method employed the LSTM technique to predict new data from the historical time series. We compared the real observation data with the predicted results for disturbance detection. Using forest fire detection as an example, the validation results indicated the effectiveness of the proposed approach. The prediction based on LSTM for multiple steps had better performance compared to the one-step prediction, and is thus recommended.

Due to the adaptability and robustness of LSTM, this method can be extended to support the analysis of different types of time series. According to the specific objectives, this method can also be customised for remote sensing data of different spatial or temporal resolutions.

Acknowledgments: This work is jointly supported by the National Key Research and Development Program of China (grant no. 2016YFB0502503), the “135” Strategy Planning of the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences (grant no. Y7SG0800CX), the National Science and Technology Major Project of China (grant no. 30-Y20A04-9001-17/18 and grant no. 21-Y20A06-9001-17/18).

Author Contributions: Yun-Long Kong had the original idea for the study and wrote the original manuscript. Qingqing Huang supervised the research and reviewed and edited the manuscript. Chengyi Wang and Jingbo Chen performed the experiments, and analysed the data. Jiansheng Chen and Dongxu He were responsible for the data collection and examination. All the authors read and approved the final manuscript.
Conflicts of Interest: The authors declare no conflicts of interest.

References


© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).