

Article

Deep Learning Based Oil Palm Tree Detection and Counting for High-Resolution Remote Sensing Images

Weijia Li ^{1,2}, Haohuan Fu ^{1,2,3,*}, Le Yu ^{1,2} and Arthur Cracknell ⁴

¹ Ministry of Education Key Laboratory for Earth System Modeling, Department of Earth System Science, Tsinghua University, Beijing 100084, China; liwj14@mails.tsinghua.edu.cn (W.L.); leyu@tsinghua.edu.cn (L.Y.)

² Joint Center for Global Change Studies (JCGCS), Beijing 100084, China

³ National Supercomputing Center in Wuxi, Wuxi 214072, China

⁴ Division of Electronic Engineering and Physics, University of Dundee, Dundee DDI 4HN, UK; apcracknell774787@yahoo.co.uk

* Correspondence: haohuan@tsinghua.edu.cn; Tel.: +86-158-1084-7817

Academic Editors: Janet Nichol and Prasad S. Thenkabail

Received: 5 November 2016; Accepted: 28 December 2016; Published: 30 December 2016

Abstract: Oil palm trees are important economic crops in Malaysia and other tropical areas. The number of oil palm trees in a plantation area is important information for predicting the yield of palm oil, monitoring the growing situation of palm trees and maximizing their productivity, etc. In this paper, we propose a deep learning based framework for oil palm tree detection and counting using high-resolution remote sensing images for Malaysia. Unlike previous palm tree detection studies, the trees in our study area are more crowded and their crowns often overlap. We use a number of manually interpreted samples to train and optimize the convolutional neural network (CNN), and predict labels for all the samples in an image dataset collected through the sliding window technique. Then, we merge the predicted palm coordinates corresponding to the same palm tree into one palm coordinate and obtain the final palm tree detection results. Based on our proposed method, more than 96% of the oil palm trees in our study area can be detected correctly when compared with the manually interpreted ground truth, and this is higher than the accuracies of the other three tree detection methods used in this study.

Keywords: oil palm trees; deep learning; convolutional neural network (CNN); object detection

1. Introduction

Oil palm trees are important economic crops. In addition to their main use to produce palm oil, oil palms are also used to generate a variety of products such as plywood, paper, furniture, etc. [1]. Information about the locations and the number of oil palm trees in a plantation area is important in many aspects. First, it is essential for predicting the yield of palm oil, which is the most widely used vegetable oil in the world. Second, it provides vital information to understand the growing situation of palm trees after plantation, such as the age or the survival rate of the palm trees. Moreover, it informs the development of irrigation processes and maximizes productivity [2].

Remote sensing has played an important role in various studies on oil palm productivity, the age of oil palm trees and oil palm mapping, etc. [3–8]. In recent years, high-resolution remote sensing images have become increasingly popular and important for many applications including automatic palm tree detection. Previous palm tree or tree crown detection research has usually been based on traditional methods in the computer vision domain. For instance, a tree detection–delineation algorithm was designed for high-resolution digital imagery tree crown detection, which is based

on the local maximum filter and the analysis of local transects extending outward from a potential tree apex [9]. Shafri et al. [10] presented an approach for oil palm tree extraction and counting from high spatial resolution airborne imagery data, which is composed of many parts including spectral analysis, texture analysis, edge enhancement, segmentation process, morphological analysis and blob analysis. Ke et al. [11] reviewed various methods for automatic individual tree-crown detection and delineation from passive remote sensing, including local maximum filtering, image binarization, scale analysis, and template matching, etc. Srestasathiern et al. [12] used semi-variogram computation and non-maximal suppression for palm tree detection from high-resolution multi-spectral satellite images.

Moreover, some researchers have also applied machine learning-based methods to palm tree detection studies. Malek et al. [2] used a scale-invariant feature transform (SIFT) and a supervised extreme learning machine classifier to detect palm trees from unmanned aerial vehicle (UAV) images. Manandhar et al. [13] used circular autocorrelation of the polar shape matrix representation of an image as the shape feature and a linear support vector machine to standardize and reduce dimensions of the feature. This study also used a local maximum detection algorithm on the spatial distribution of standardized features to detect palm trees. Previous palm tree or tree crown detection studies have focused on detecting trees that are not very crowded and have achieved good detection results for their study areas. However, the performance of some of these methods would deteriorate when detecting palm trees in some of the regions of our study area. For instance, the local maximum filter based method [9] cannot detect palm trees correctly in regions where the trees are very young and small, as the local maximum of each filter does not locate around the apex of young palm trees. The template matching method [10] is not suitable for regions where palm trees are very crowded and where their crowns overlap.

The convolutional neural network (CNN), a widely used deep learning model, has achieved great performance in many studies in the computer vision field, such as image classification [14,15], face recognition [16,17], and pedestrian detection [18,19], etc. In recent years, deep learning based methods have also been applied to hyperspectral image classification [20,21], large-scale land cover classification [22], scene classification [23–25], and object detection [26,27], etc. in the remote sensing domain and achieved better performance than traditional methods. For instance, Chen et al. [20] introduced the concept of deep learning and applied the stacked autoencoder method to hyperspectral remote sensing image classification for the first time. Li et al. [22] built a classification framework for large-scale remote sensing image processing and African land cover mapping based on the stacked autoencoder. Zou et al. [24] proposed a deep belief network based feature selection method for remote sensing scene classification. Chen et al. [26] proposed a hybrid deep convolutional neural network for vehicle detection in high-resolution satellite images. Vakalopoulou et al. [27] proposed an automated building detection framework from very high-resolution remote sensing data based on deep convolutional neural networks.

In this paper, we introduce the deep learning based method to oil palm tree detection for the first time. We propose a CNN based framework for the detection and counting of oil palm trees using high-resolution remote sensing images from Malaysia. The detection and counting of oil palm trees in our study area is more difficult than for the previous palm detection research mentioned above, as the trees are very crowded and their crowns often overlap. In our proposed method, we collect a number of manually interpreted training and test samples for training the convolutional neural network and calculating the classification accuracy. Secondly, we optimize the convolutional neural network through tuning its main parameters to obtain the best CNN model. Then, we use the best CNN model obtained previously to predict the labels for all the samples in an image dataset that are collected through the sliding window technique. Finally, we merge the predicted palm tree coordinates corresponding to the same palm tree (spatial distance less than a certain threshold) into one coordinate, and obtain the final palm tree detection results. Compared with the manually interpreted ground truth, more than 96% of the oil palm trees in our study area can be detected correctly, which is higher than the accuracies of the other three tree detection methods used in this study. The detection accuracy

of our proposed method is affected, to some extent, by the limited number of our manually interpreted samples. In our future work, more manually interpreted samples will be collected to further improve the overall performance of our proposed method.

The rest of this paper is organized as follows. Section 2 presents the study area and the datasets of this research; Section 3 describes the flowchart and the details of our proposed method; Section 4 provides the detection results of our proposed method and the performance comparison with other methods; and Section 5 presents some important conclusions of this research.

2. Study Area and Datasets

In this research, a QuickBird image acquired on 21 November 2006 is used. The QuickBird satellite has one panchromatic (Pan) band with 0.6-m spatial resolution and four multi-spectral (MS) bands with 2.4-m spatial resolution. The Gram–Schmidt (GS) spectral sharpening fusion method [28], which is implemented in the ENVI software (version 5.3, Exelis Visual Information Solutions, Boulder, CO, USA), was employed to integrate Pan and MS bands to obtain a higher sharpness and spectral quality (0.6-m spatial resolution, four bands) dataset for further image processing and analysis.

The study area of this research is located in the south of Malaysia, as shown in Figure 1. The manually interpreted samples used in this study were collected from two typical regions of our study area (denoted by the blue rectangles in Figure 1). To evaluate the performance of our proposed method, we selected another three representative regions in our study area (denoted by the red squares in Figure 1) and compared the detected images of these regions with the ground truth collected by manual interpretation.

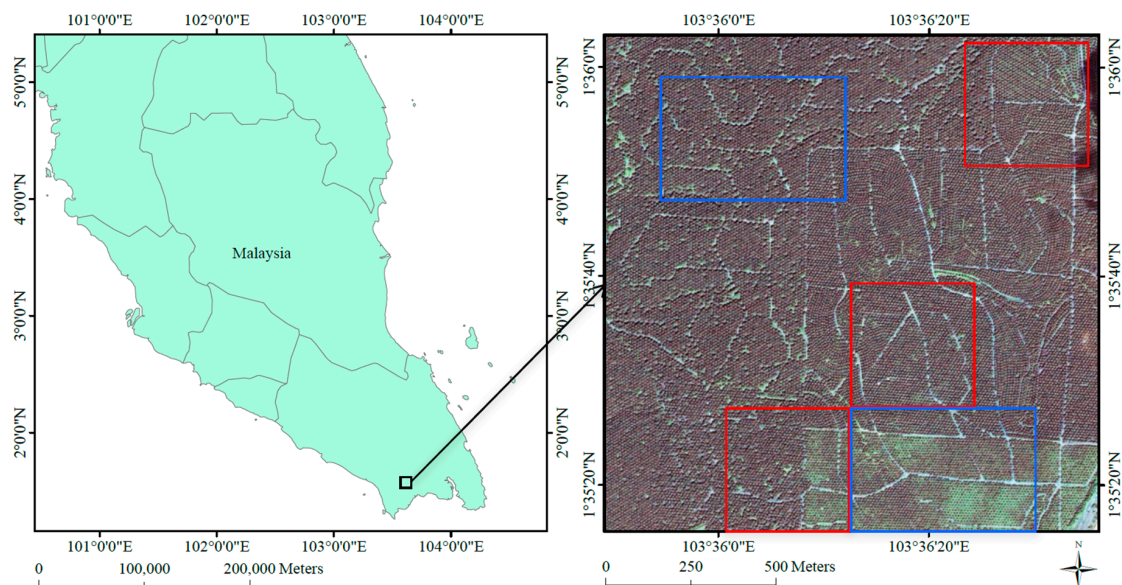


Figure 1. The study area of this research in the south of Peninsular Malaysia. The **blue** rectangles show the two regions from which the manually interpreted samples are collected. The **red** squares show the three selected regions for evaluating the performance of our proposed method.

3. Methods

3.1. Overview

The flowchart of our proposed method is shown in Figure 2. First, the convolutional neural network [14] was implemented based on the Tensorflow framework [29]. We used a number of training samples collected previously by manual interpretation to train the CNN, and calculated the classification accuracy based on a number of test samples collected independently of training samples.

The main parameters of the CNN (e.g., the number of kernels in the first convolutional layer, the number of kernels in the second convolutional layer and the number of hidden units in the fully connected layer) were adjusted continuously until we found the best combination of parameters of which the overall accuracy was the highest on our test samples. By tuning the parameters, we achieved the best CNN model and saved it for further use. Secondly, the image dataset for palm tree detection was collected through the sliding window technique (the window size is 17×17 and the sliding step is three pixels). Then, we used the best CNN model obtained previously to predict the label for each sample in the image dataset. Thirdly, for all samples that were predicted as “palm tree” class, we merged the coordinates corresponding to the same palm tree sample (spatial distance less than a certain threshold) into one coordinate, and obtained the final palm tree detection results.

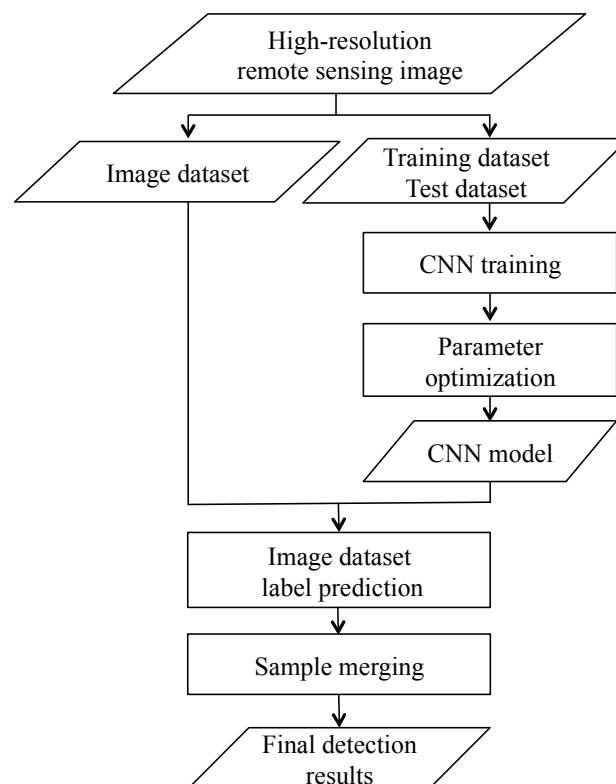


Figure 2. The flowchart of our proposed method.

3.2. CNN Training and Parameter Optimization

The LeNet convolutional neural network used in this study is constructed of two convolutional layers, two pooling layers and a fully connected layer, as shown in Figure 3. The input to the fully connected layer is the set of all features maps at the layer below. The fully connected layers correspond to a traditional multilayer perceptron constructed by a hidden layer and a logistic regression layer. We use the Rectified Linear Unit (ReLU) as the activation function of the CNN. In this research, we manually interpreted 5000 palm tree samples and 4000 background samples from two regions of our study area (denoted by the blue rectangles in Figure 1). Then, we randomly select 7200 of these samples as the training dataset of the convolutional neural network, and the other 1800 samples as its test dataset. Only a sample with a palm located at its center will be labeled as “palm tree”. Each sample corresponds to 17×17 pixels with three bands (Red, Green and Blue) selected from the original four bands. The main parameters of CNN are adjusted continuously until we find the best combination of parameters for which the overall accuracy is the highest from 1800 test samples. After parameter tuning, we achieve the best CNN model that will be used in the subsequent process of image dataset label prediction.

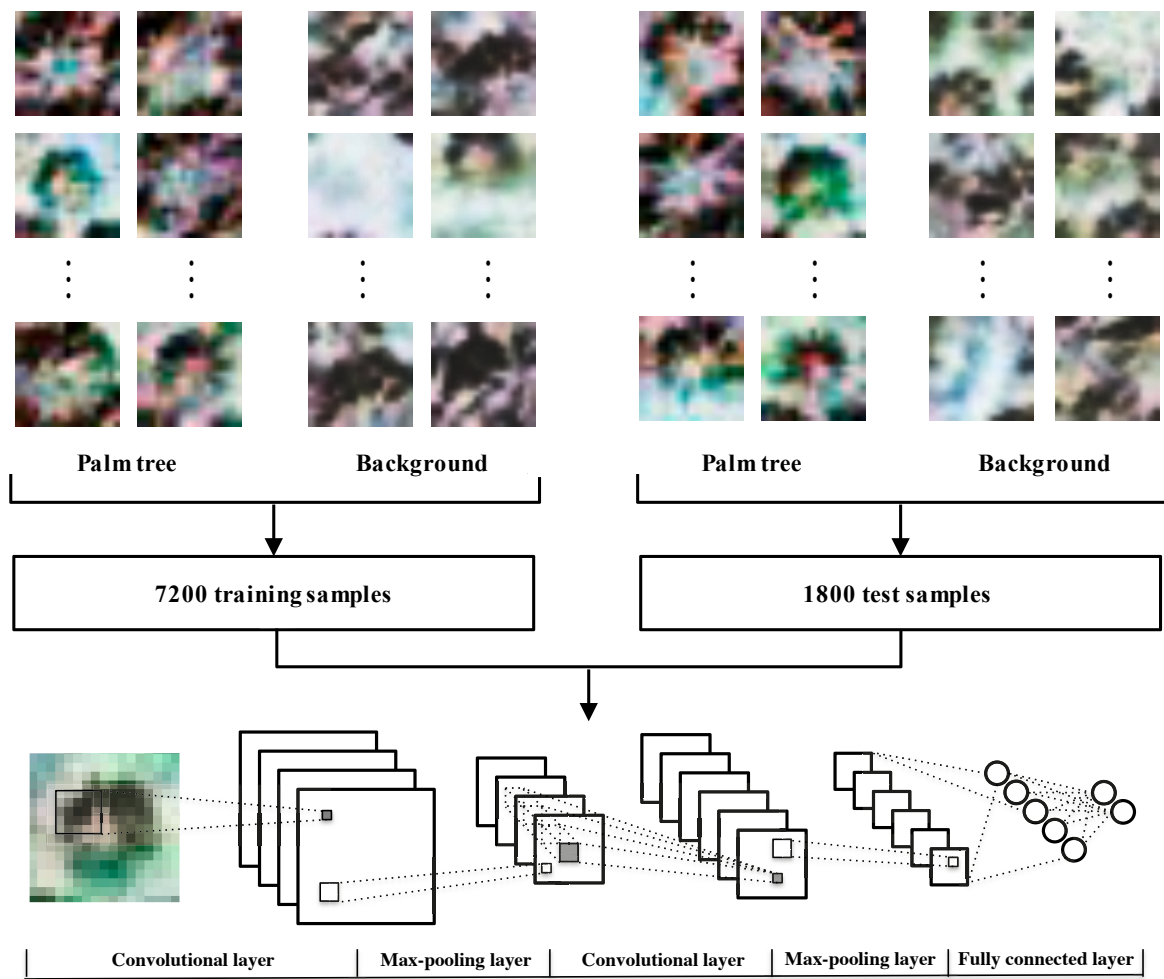


Figure 3. The structure of the convolutional neural network (CNN).

3.3. Label Prediction

The image dataset for label prediction is collected through the sliding window technique, as shown in Figure 4. The size of the sliding window is 17×17 pixels, which is consistent with the feature size of our training and test samples. In addition, the sliding step (the moving distance of the sliding window in each step) will have a great influence on the final palm tree detection results. If the sliding step is too large, many palm samples will be missed and will not be detected. On the other hand, if the sliding step is too small, one palm sample might be detected repeatedly. Moreover, the process of label prediction will become slower due to the increasing number of samples in the image dataset, which is actually unnecessary and a waste of time. In this study, the sliding step is set as three pixels through experimental tests. After collecting all samples of the image dataset through the sliding window technique, we use the best CNN model obtained in Section 3.2 to predict the label for each sample in the image dataset.



Figure 4. The sliding window technique.

3.4. Sample Merging

After the labels of all samples in the image dataset are predicted, we collect the spatial coordinates of all the samples that are predicted as “palm tree” class. At this point, the number of predicted palm tree coordinates could be larger than the actual number of palm trees because one palm tree might correspond to several predicted palm tree coordinates. To avoid this problem, the coordinates corresponding to the same palm tree sample will be merged into one coordinate iteratively, as shown in Figure 5. Assuming that, in our study area, the spatial distance between two palm trees cannot be less than 8 pixels, the merging process will take six iterations. In each iteration, all groups of coordinates with the Euclidean distance less than a certain threshold (3, 4, 5, 6, 7, 8 pixels) will be merged into one coordinate. That is, the original group of coordinates will be replaced by their average coordinate. The remaining palm tree coordinates after the merging process represent the actual coordinates of detected palm trees.

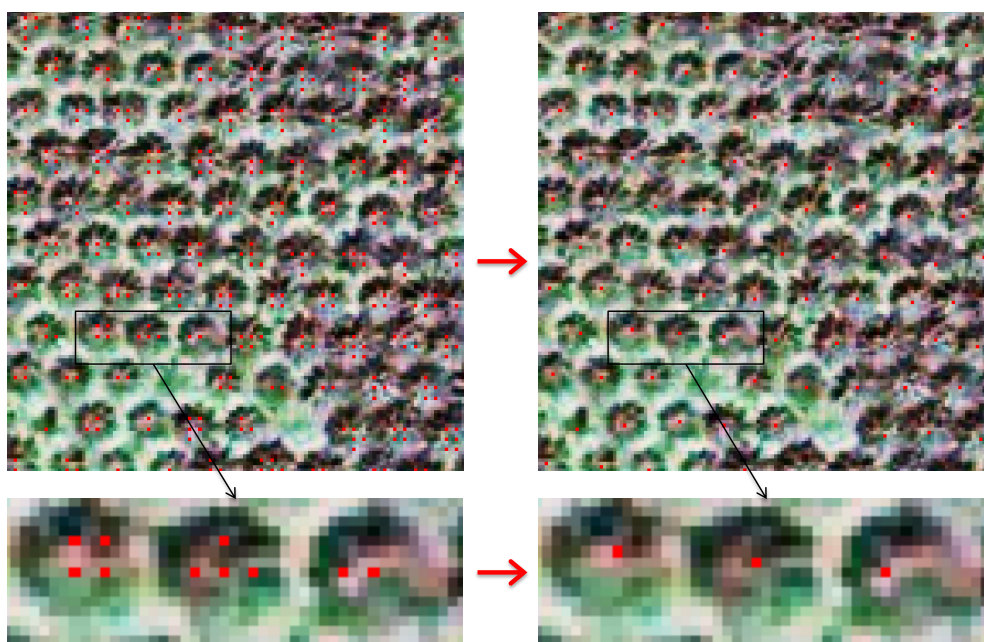


Figure 5. Sample merging.

4. Results

4.1. Classification Accuracy and Parameter Optimization

In this study, the classification accuracy of our CNN model was assessed by 1800 test samples collected independently from 7200 training samples. The classification accuracy can be affected by many parameters, such as the size of the convolutional kernel and the max-pooling kernel, the number of kernels in each convolutional layer and hidden units in fully connected layers, etc. For our CNN model, the size of the convolutional kernel is five, the size of the max-pooling kernel is two, the size of mini-batch is 10 and the maximum number of iterations is 8000. We adjusted three important parameters to optimize the model: the number of kernels in the first convolutional layer, the number of kernels in the second convolutional layer and the number of hidden units in the fully connected layer. Experimental results showed that we could obtain the highest overall accuracy of 95% after 7500 iterations when the number of kernels in two convolutional layers are set as 30 and 55 and the number of hidden units in fully connected layers is set as 600.

4.2. Detection Results Evaluation

To evaluate the performance of our proposed oil palm tree detection method quantitatively, we calculate the precision, recall and overall accuracy of the palm tree detection results through comparison with the ground truth. The precision is the probability that a detected oil palm tree is valid, as described in Formula (1); the recall is the probability that an oil palm tree in ground truth is detected, as described in Formula (2); the overall accuracy is the average of precision and recall, as described in Formula (3). A palm is regarded as detected correctly only if the distance between the center of a detected palm and the center of a palm in ground truth is less than or equal to five pixels:

$$\text{Precision} = \frac{\text{The number of correctly detected palm trees}}{\text{The number of all detected objects}}, \quad (1)$$

$$\text{Recall} = \frac{\text{The number of correctly detected palm trees}}{\text{The number of palm trees in ground truth}}, \quad (2)$$

$$\text{Overall Accuracy} = \frac{\text{Precision} + \text{Recall}}{2}. \quad (3)$$

Table 1 shows that the overall accuracies of regions 1, 2 and 3 are 96.05%, 96.34% and 98.77%, respectively. In addition, for each of the three regions, the difference between the predicted number of palm trees (the number of all detected objects) and the true number of palm trees (the number of palm trees in ground truth) is less than 4%. These evaluation results show that our proposed method is effective for both palm tree detection and counting.

Table 1. Detection results of convolutional neural network (CNN).

Evaluation Index	Region 1	Region 2	Region 3
The number of correctly detected palm trees	1651	1607	1683
The number of all detected objects	1729	1695	1706
The number of palm trees in ground truth	1709	1642	1702
Precision	95.49%	94.81%	98.65%
Recall	96.61%	97.87%	98.88%
Overall accuracy	96.05%	96.34%	98.77%

5. Discussion

To further evaluate our proposed palm tree detection method, we implemented three other representative existing palm trees or tree crown detection methods (i.e., Artificial Neural Network (ANN), template matching, and local maximum filter) and compared their detection results with

our proposed method. The procedure of the ANN based method is the same as our proposed method, including the ANN training, parameter optimization, image dataset label prediction, and sample merging.

The local maximum filter based method [9] and the template matching based method [11] are two traditional tree crown detection methods. For the template matching based method, we used 5000 manually labeled palm tree samples as the template dataset, and a 17×17 window to slide through the whole image. We chose the CV_TM_SQDIFF_NORMED provided by OpenCV [30] as our matching method. A sliding window will be detected as a palm tree if it matches any sample in the template dataset (the difference between the sliding window and the template calculated by the CV_TM_SQDIFF_NORMED method is less than a threshold. In this study, the threshold is set as five through experimental tests).

For the local maximum filter based method, we first applied a non-overlapping 10×10 local maximum filter to the absolute difference image of the NIR and red spectral bands. Then, we conducted transect sampling and a scaling scheme to obtain potential tree apexes, and adjusted the locations of tree apexes to the new local maximum positions.

Finally, the outputs of the template matching based method and the local maximum filter based method are post-processed (described in Section 3.4) to obtain the final palm tree detection results. Figures 6–8 show the detection images of each method for extracted areas of regions 1, 2 and 3, respectively. Each red circle denotes a detected palm tree. Each green square denotes a palm tree in ground truth that cannot be detected correctly. Each blue square denotes a background sample that is detected as a palm tree by mistake.

Tables 2–4 show the detection results of ANN, template matching (TMPL), and local maximum filter (LMF), respectively. Table 5 summarizes the performance of all four methods in terms of the number of correctly detected palm trees. Table 6 summarizes the performance of all four methods in terms of precision, recall and overall accuracy (OA). The proposed method (CNN) performs better than any of the other three methods in the number of correctly detected palm trees and in OA. Generally, machine learning based approaches (i.e., CNN and ANN) perform better than traditional tree crown detection methods (i.e., TMPL and LMF) in our study area, especially in region 1 and region 2. For example, the local maximum filter based method cannot detect palm trees correctly for regions where palm trees are very young and small (see Figure 7d), as the local maximum of each filter does not locate around the apex of young palm trees. The template matching method is not suitable for regions where the palm trees are very crowded and the canopies often overlap (see Figure 6c).

Table 2. Detection results of artificial neural network (ANN).

Evaluation Index	Region 1	Region 2	Region 3
The number of correctly detected palm trees	1648	1585	1679
The number of all detected objects	1800	1725	1718
The number of palm trees in ground truth	1709	1642	1702
Precision	91.56%	91.88%	97.73%
Recall	96.43%	96.53%	98.64%
Overall accuracy	94.00%	94.21%	98.19%

Table 3. Detection results of template matching (TMPL).

Evaluation Index	Region 1	Region 2	Region 3
The number of correctly detected palm trees	1429	1392	1608
The number of all detected objects	1532	1493	1684
The number of palm trees in ground truth	1709	1642	1702
Precision	93.28%	93.24%	95.49%
Recall	83.62%	84.77%	94.48%
Overall accuracy	88.45%	89.01%	94.99%

Table 4. Detection results of local maximum filter (LMF).

Evaluation Index	Region 1	Region 2	Region 3
The number of correctly detected palm trees	1493	1397	1643
The number of all detected objects	1719	1675	1761
The number of palm trees in ground truth	1709	1642	1689
Precision	86.85%	83.40%	93.30%
Recall	87.36%	85.08%	97.28%
Overall accuracy	87.11%	84.24%	95.29%

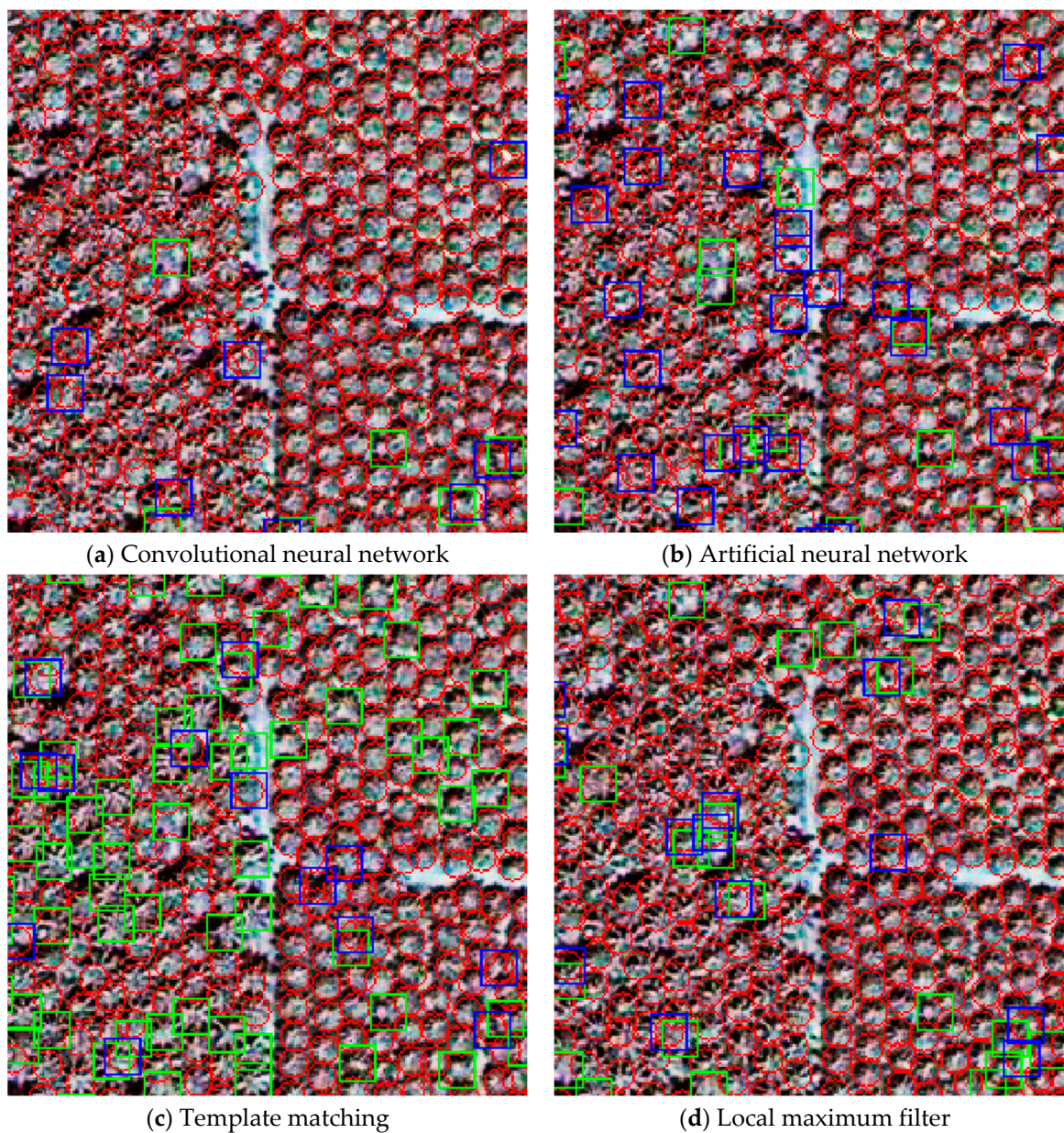


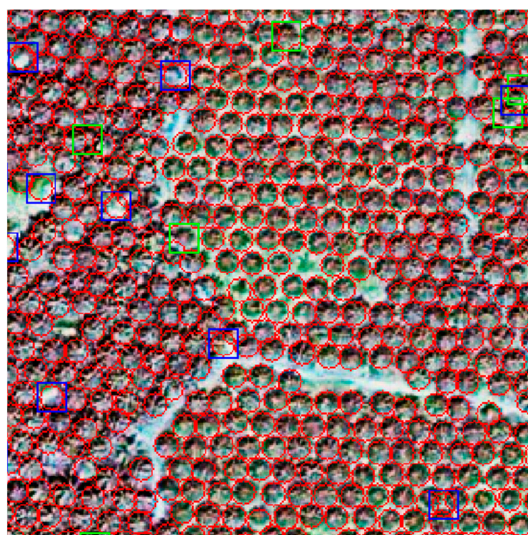
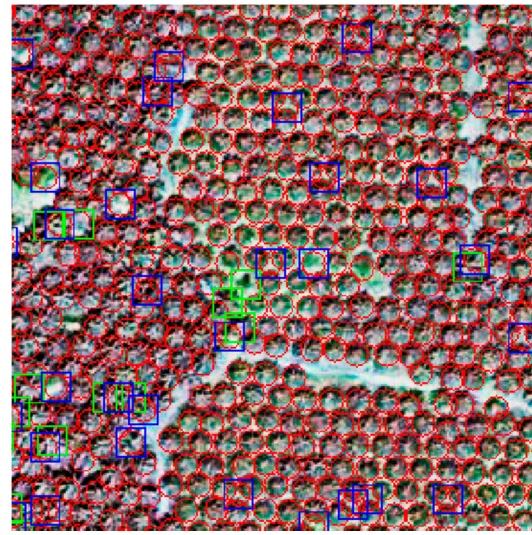
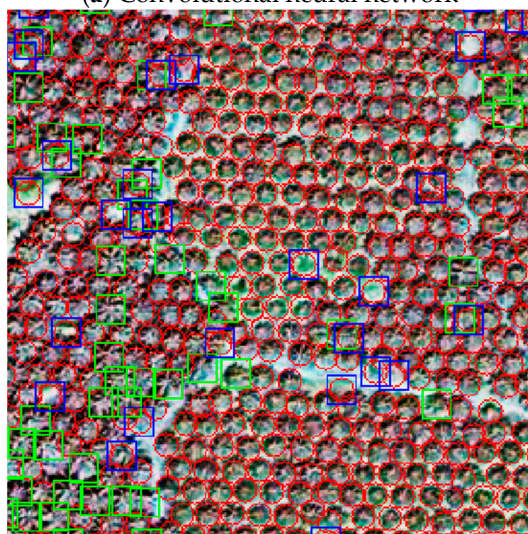
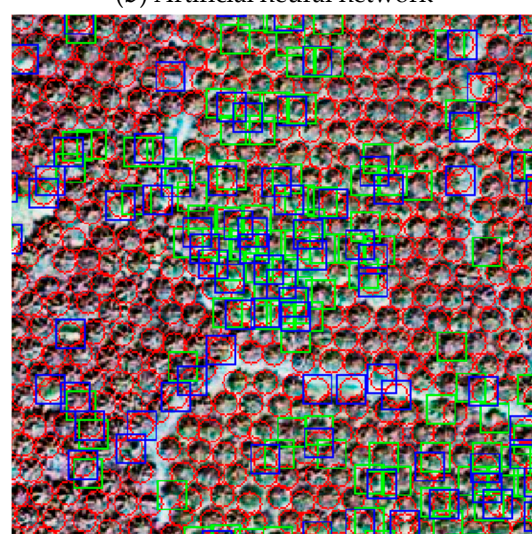
Figure 6. Detection image of each method for region 1 (extracted area). Each **red** circle denotes a detected palm tree. Each **green** square denotes a palm tree in ground truth that cannot be detected correctly. Each **blue** square denotes a background sample that is detected as a palm tree by mistake.

Table 5. Summary of the number of correctly detected palm trees for all four methods.

Methods	Region 1	Region 2	Region 3
CNN	1651	1607	1683
ANN	1648	1585	1679
TMPL	1429	1392	1608
LMF	1493	1397	1643

Table 6. Summary of the precision, recall and overall accuracy (OA) of all four methods.

Methods	Region 1			Region 2			Region 3		
	Precision	Recall	OA	Precision	Recall	OA	Precision	Recall	OA
CNN	95.49%	96.61%	96.05%	94.81%	97.87%	96.34%	98.65%	98.88%	98.77%
ANN	91.56%	96.43%	94.00%	91.88%	96.53%	94.21%	97.73%	98.64%	98.19%
TMPL	93.28%	83.62%	88.45%	93.24%	84.77%	89.01%	95.49%	94.48%	94.99%
LMF	86.85%	87.36%	87.11%	83.40%	85.08%	84.24%	93.30%	97.28%	95.29%

**(a)** Convolutional neural network**(b)** Artificial neural network**(c)** Template matching**(d)** Local maximum filter**Figure 7.** Detection image of each method for region 2 (extracted area).

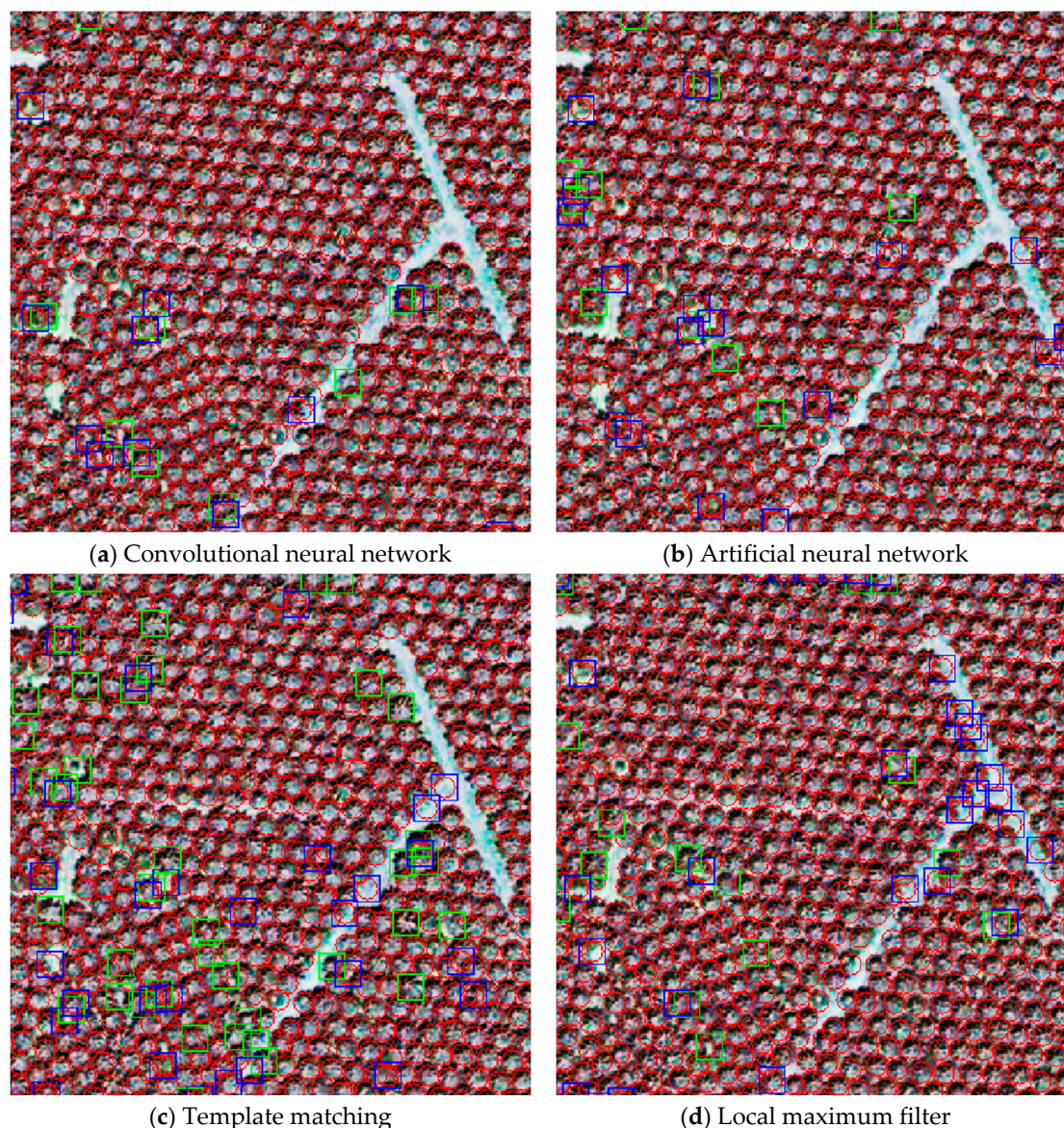


Figure 8. Detection image of each method for region 3 (extracted area).

6. Conclusions

In this paper, we designed and implemented a deep learning based framework for oil palm tree detection and counting from high-resolution remote sensing images. Three representative regions in our study area are selected for assessment of our proposed method. Experimental results show the effectiveness of our proposed method for palm tree detection and counting. First, the palm tree detection results are very similar to the manually labeled ground truth in general. Secondly, the overall accuracies of region 1, region 2 and region 3 are 96%, 96% and 99%, respectively, which are higher than the accuracies of the three other methods used in this research. Moreover, the difference between the predicted number of palm trees and the true number of palm trees is less than 4% for each region of the study area. In our future work, the palm tree detection results should be further improved through enlarging the number of manually interpreted samples and optimizing our proposed CNN based detection framework. We also want to take the computation time of different detection methods into consideration, and explore the deep learning based detection framework for larger scale palm tree detection studies.

Acknowledgments: This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 61303003 and 41374113), the National High-Tech R&D (863) Program of China (Grant No. 2013AA01A208), the Tsinghua University Initiative Scientific Research Program (Grant No. 20131089356), and the National Key Research and Development Plan of China (Grant No. 2016YFA0602200).

Author Contributions: Weijia Li, Haohuan Fu and Le Yu conceived of the study; Weijia Li wrote code, performed the analysis and wrote the article; Haohuan Fu performed the analysis and wrote the article; Le Yu and Arthur Cracknell performed the analysis and commented on the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Suhaily, S.; Jawaaid, M.; Khalil, H.A.; Mohamed, A.R.; Ibrahim, F. A review of oil palm biocomposites for furniture design and applications: Potential and challenges. *BioResources* **2012**, *7*, 4400–4423.
2. Malek, S.; Bazi, Y.; Alajlan, N.; AlHichri, H.; Melgani, F. Efficient framework for palm tree detection in UAV images. *IEEE J. Sel. Top. Appl. Earth Obs.* **2014**, *7*, 4692–4703. [[CrossRef](#)]
3. Cracknell, A.P.; Kanniah, K.D.; Tan, K.P.; Wang, L. Evaluation of MODIS gross primary productivity and land cover products for the humid tropics using oil palm trees in Peninsular Malaysia and Google Earth imagery. *Int. J. Remote Sens.* **2013**, *34*, 7400–7423. [[CrossRef](#)]
4. Tan, K.P.; Kanniah, K.D.; Cracknell, A.P. A review of remote sensing based productivity models and their suitability for studying oil palm productivity in tropical regions. *Prog. Phys. Geogr.* **2012**, *36*, 655–679. [[CrossRef](#)]
5. Tan, K.P.; Kanniah, K.D.; Cracknell, A.P. Use of UK-DMC2 and ALOS PALSAR for studying the age of oil palm trees in southern peninsular Malaysia. *Int. J. Remote Sens.* **2013**, *34*, 7424–7446. [[CrossRef](#)]
6. Kanniah, K.D.; Tan, K.P.; Cracknell, A.P. UK-DMC2 satellite data for deriving biophysical parameters of oil palm trees in Malaysia. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, Munich, Germany, 22–27 July 2012; pp. 6569–6572.
7. Cheng, Y.; Yu, L.; Cracknell, A.P.; Gong, P. Oil palm mapping using Landsat and PALSAR: A case study in Malaysia. *Int. J. Remote Sens.* **2016**, *37*, 5431–5442. [[CrossRef](#)]
8. Cracknell, A.P.; Kanniah, K.D.; Tan, K.P.; Wang, L. Towards the development of a regional version of MOD17 for the determination of gross and net primary productivity of oil palm trees. *Int. J. Remote Sens.* **2015**, *36*, 262–289. [[CrossRef](#)]
9. Pouliot, D.A.; King, D.J.; Bell, F.W.; Pitt, D.G. Automated tree crown detection and delineation in high-resolution digital camera imagery of coniferous forest regeneration. *Remote Sens. Environ.* **2002**, *82*, 322–334. [[CrossRef](#)]
10. Shafri, H.Z.; Hamdan, N.; Saripan, M.I. Semi-automatic detection and counting of oil palm trees from high spatial resolution airborne imagery. *Int. J. Remote Sens.* **2011**, *32*, 2095–2115. [[CrossRef](#)]
11. Ke, Y.; Quackenbush, L.J. A review of methods for automatic individual tree-crown detection and delineation from passive remote sensing. *Int. J. Remote Sens.* **2011**, *32*, 4725–4747. [[CrossRef](#)]
12. Srestasathiern, P.; Rakwatin, P. Oil palm tree detection with high resolution multi-spectral satellite imagery. *Remote Sens.* **2014**, *6*, 9749–9774. [[CrossRef](#)]
13. Manandhar, A.; Hoegner, L.; Stilla, U. Palm Tree Detection Using Circular Autocorrelation of Polar Shape Matrix. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *III-3*, 465–472. [[CrossRef](#)]
14. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–8 December 2012; pp. 1097–1105.
15. Ciregan, D.; Meier, U.; Schmidhuber, J. Multi-column deep neural networks for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Rhode Island, RI, USA, 16–21 June 2012; pp. 3642–3649.
16. Sun, Y.; Wang, X.; Tang, X. Deep learning face representation from predicting 10,000 classes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 24–27 June 2014; pp. 1891–1898.
17. Li, H.; Lin, Z.; Shen, X.; Brandt, J.; Hua, G. A convolutional neural network cascade for face detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5325–5334.

18. Ouyang, W.; Wang, X. Joint deep learning for pedestrian detection. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 2056–2063.
19. Zeng, X.; Ouyang, W.; Wang, X. Multi-stage contextual deep learning for pedestrian detection. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 121–128.
20. Chen, Y.; Lin, Z.; Zhao, X.; Wang, G.; Gu, Y. Deep learning-based classification of hyperspectral data. *IEEE J. Sel. Top. Appl. Earth Obs.* **2014**, *7*, 2094–2107. [[CrossRef](#)]
21. Yue, J.; Zhao, W.; Mao, S.; Liu, H. Spectral–spatial classification of hyperspectral images using deep convolutional neural networks. *Remote Sens. Lett.* **2015**, *6*, 468–477. [[CrossRef](#)]
22. Li, W.; Fu, H.; Yu, L.; Gong, P.; Feng, D.; Li, C.; Clinton, N. Stacked Autoencoder-based deep learning for remote-sensing image classification: a case study of African land-cover mapping. *Int. J. Remote Sens.* **2016**, *37*, 5632–5646. [[CrossRef](#)]
23. Hu, F.; Xia, G.S.; Hu, J.; Zhang, L. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sens.* **2015**, *7*, 14680–14707. [[CrossRef](#)]
24. Zou, Q.; Ni, L.; Zhang, T.; Wang, Q. Deep learning based feature selection for remote sensing scene classification. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2321–2325. [[CrossRef](#)]
25. Penatti, O.A.; Nogueira, K.; dos Santos, J.A. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Boston, MA, USA, 7–12 June 2015; pp. 44–51.
26. Chen, X.; Xiang, S.; Liu, C.L.; Pan, C.H. Vehicle detection in satellite images by hybrid deep convolutional neural networks. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 1797–1801. [[CrossRef](#)]
27. Vakalopoulou, M.; Karantza, K.; Komodakis, N.; Paragios, N. Building detection in very high resolution multispectral data with deep learning features. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, Italy, 26–31 July 2015; pp. 1873–1876.
28. Laben, C.A.; Brower, B.V. Process for Enhancing the Spatial Resolution of Multispectral Imagery Using Pan-Sharpener. U.S. Patent 6,011,875, 4 January 2000.
29. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015. Available online: <https://arxiv.org/abs/1603.04467> (accessed on 31 August 2016).
30. Bradski, G.; Kaehler, A. *Learning OpenCV: Computer Vision with the OpenCV Library*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2008.



© 2016 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/>).