symmetry

MDPI

Article

Leaf Counting with Multi-Scale Convolutional Neural Network Features and Fisher Vector Coding

Boran Jiang [1], Ping Wang [1,*], Shuo Zhuang [1], Maosong Li [2], Zhenfa Li [3] and Zhihong Gong [3]

[1] School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China; boran.jiang1991@tju.edu.cn (B.J.); shuozhuang@tju.edu.cn (S.Z.)

[2] Institute of Agricultural Resources and Regional Planning, Beijing 100081, China; limaosong@caas.cn

[3] Tianjin Climate Center, Tianjin 300074, China; lzfaaa@126.com (Z.L.); gong041@126.com (Z.G.)

* Correspondence: wangps@tju.edu.cn

Received: 12 March 2019; Accepted: 8 April 2019; Published: 10 April 2019

Abstract: The number of leaves in maize plant is one of the key traits describing its growth conditions. It is directly related to plant development and leaf counts also give insight into changing plant development stages. Compared with the traditional solutions which need excessive human interventions, the methods of computer vision and machine learning are more efficient. However, leaf counting with computer vision remains a challenging problem. More and more researchers are trying to improve accuracy. To this end, an automated, deep learning based approach for counting leaves in maize plants is developed in this paper. A Convolution Neural Network(CNN) is used to extract leaf features. The CNN model in this paper is inspired by Google Inception Net V3, which using multi-scale convolution kernels in one convolution layer. To compress feature maps generated from some middle layers in CNN, the Fisher Vector (FV) is used to reduce redundant information. Finally, these encoded feature maps are used to regress the leaf numbers by using Random Forests. To boost the related research, a relatively single maize image dataset (Different growth stage with 2845 samples, which 80% for train and 20% for test) is constructed by our team. The proposed algorithm in single maize data set achieves Mean Square Error (MSE) of 0.32.

Keywords: maize; Convolution Neural Network; Multi-scales; Fisher Vector; Random Forests

1. Introduction

Precision agriculture, which focuses on optimizing production by accounting for variabilities and dealing with uncertainties in agricultural systems, has been under active research in recent years [1]. Feature monitoring and plant phenotyping are essential parts of precision agriculture. They can help in modeling the growth process of plants and guide farmers to obtain higher yields with appropriate fertilizer, irrigation, and disease control [2,3].

Traditional plant phenotyping, involves a large number of manual measurements, and this has been identified as the current bottleneck in modern plant breeding and research programs [4]. The number of leaves of a plant is considered one of the critical phenotypic metrics related to its development and growth stages [5,6], flowering time [7], and water condition. The traditional manual measurement is slow, tedious, and expensive. Therefore, several image-based and machine learning technologies have been introduced for leaf counting. However, counting leaves automatically is challenging [8], due to a plant's rapid growth and leaf occlusion and illumination problems. Moreover, most study on leaf counting are based on rosette plants, and the relevant algorithms are not suitable for maize plants. Considering this, we designed a model suitable for counting maize leaves.

In this study, we estimate the number of leaves on a maize plant at different growth stages. The problem is posed as a nonlinear regression problem, which does not require segmenting individual leaf instances. First, features are extracted from each sample image. Then these feature vectors are used to regress the number of leaves. For this model, the input is a maize image, and the output is the number of leaves.

Effective feature extraction is an important step for leaf counting regression and plant phenotyping research. Over the past years, substantial efforts have been dedicated to developing robust feature representation methods in different domains. The histogram of oriented gradients (HOG) has been used to detect the tasseling stage of maize [9]. Then, mid-level feature methods, such as wavelet transform and the Fisher vector, are used as feature descriptors since they attract much attention. In [10], the authors used wavelet transform to extract energy features and detect maize water stress. FV coding is combined with scale-invariant feature transforms (SIFT) for object detection [11].

Recently, deep learning, particularly the use of deep convolutional neural networks (CNN), has become the new state-of-the-art solution for object detection, recognition, and regression. Compared with traditional feature descriptors, the convolutional layer of CNN can extract low-level to high-level features. As the number of convolutional layers increases, more abstract features are extracted. Furthermore, more convolutional layers mean more parameters to be trained, but when the training samples are far fewer than the parameters, the risk of model over-fitting will increase. Therefore, some strategies of reducing parameters have been proposed such as Google Inception Net V3 [12] and residual networks [13]. For instance, in [14] Google Inception Net was used to identify leaf species. In addition, the traditional method is used to optimize network parameters. Such as adding constraint to optimize the parameters in the CNN output to improve the accuracy of low-accuracy classes [15].

Few recent works have demonstrated that the middle layer of CNN contains a large amount of useful information, which can improve the discrimination of feature representation. One example is that it can improve the discrimination of feature representation. In [16], the authors converted the input image into the multi-scale image and fixed-size image. Then the multi-scale image and the fixed-size image was fed into CNN with the same structure separately. Subsequently, the features of each convolutional layer were extracted from the multi-scale image and the features of the full connection layer are extracted from the fixed-size image. After Fisher vector coding and principal component analysis dimensionality reduction, these features were fed into the support vector machine.

Compared to the existing work, we used the inception structure from GoogLeNet. The multi-scale convolution kernel was used in one layer instead of inputting multi-scale images, and information loss may be caused by compressing the original image when generating multi-scale images. Before feeding the features into CNN, we divided the number of leaves into different ranges and reset the label of each image sample. In fact, during training, the CNN regresses the range of leaf numbers. Extracting feature maps from each layer is computationally intensive, therefore, we obtain feature maps from three layers. In the feature extraction layer, the number of convolution kernels is reduced, which plays a role in compressing the feature map. These feature maps of the three layers are encoded by FV as fixed length feature vectors, which deduces dimensionality. Moreover, the FV can count the frequency of visual words in the feature maps and count the difference between visual dictionaries with local features.

2. Related Work

It has been proved that second-order statistics significantly improves classification performance [17]. Some methods of CNN architectures that combine second-order statistics or coding method have been proposed. Symmetric positive definite matrix network (SPDNet) was proposed in [18]. Referenced by the structure of the CNN, it is designed with bilinear mapping layers and eigenvalue layers, instead of convolution layers and rectified linear units. In [19], the authors proposed a hybrid deep-learning architecture which allows to encode CNN features with log-Euclidean Fisher Vector (LE FV).

The leaf counting methods used in recent studies are mainly of two types: counting via object segmentation and direct counting via nonlinear regression model. Counting via object segmentation. This method involves segmenting the foreground and background points of the image and filtering the background before counting. Especially, the end-to-end instance segmentation method [20] combined with long short-term memory [21] segments one leaf at one time. In [22], the authors used a segmented image mask to generate a plant skeleton and then extracted some skeleton features such as skeleton length, convex hull circularity and the number of skeleton branch points. These features were used to regress the leaf count. A 3D color histogram has always been used to segment plants, and therefore, threshold was selected from the color histogram to segment the plant [23]. Then the segmented plant generates a distance map with highlighted peaks, which serve as leaf center points. The number of center points is the number of leaves. Influenced by illumination and the image shooting angle, some background points may remain in the segmented images. These noise points have a high probability to affect the leaf counting result.

Direct count. Here segmentation is not need, and the original image is directly used to count. In previous research, the architecture of the Resnet50 model has been modified. In [24], the modified network took as input an RGB(red, green, blue three channel) image of a rosette plant and outputted a leaf count prediction. Aich and Stavness [25] used VGGNet architecture to regress the number of leaves. The input data had four channels (segmentation+RGB; leaf counting competition offered segmented samples). There are many other similar approaches, which have different structures of the selected network. The advantage of these methods is that the image segmentation step is omitted and the original image information can be used directly to regress the leaf numbers. No additional noise will be introduced.

3. Materials

This experiment has been done in a laboratory environment. The samples selected in this study were Zhengdan No.958 grown in a pot under an in-house condition. Zhengdan which is the most popular cultivated in China and growing in Henan, Hebei and Shandong province. The average plant height is 240 cm, and the average panicle height is 100 cm. The pot's height we selected in this study is 0.5 m and the upper diameter is 0.4 m, the bottom diameter is 0.3 m. The soil type is medium loam.

In this research, different water content degree has been set. In a field environment, maize is likely to be drought because of the water shortage. The purpose of this research is to detect the number of leaves in different environments. Therefore, setting experimental samples of different water content degree can better simulate the natural environment and show the effect of different water content degree on the number of leaves. Table 1 shows the moisture control.

Table 1. Scheme of the soil moisture control.

Sample	Soil Moisture	Depth of Moisture Test (cm)
Sample1	65%–80%	20
Sample2	50%–60%	20
Sample3	40%–50%	20
Sample4	<40%	20

The growth stage of the maize selected in this paper was V8 (eight visible leaves) ~ VT (the last spike was visible). According to some research, the water supply of maize in two weeks before and after the pollination period will determine the final yield [26]. Therefore, it is more meaningful to study and detect the phenotype of maize in this period. Table 2 shows the growth stage and description. Figure 1 shows some example images of maize at different growing stage. The selected image collection equipment was Canon Eos 700D. This camera has 18 million effective pixels. The actual collection of the maize image samples had a resolution of 5184 × 3456. However, the original picture resolution was compressed to 441 × 441 in the calculation to improve the computational efficiency. The camera angle and focal length were adjusted with the growth of the maize. A picture was taken every 5 minutes at 5:30 am to 18:30 pm.

Table 2. Description of the maize plant vegetative growth period.

Growth Stage	Feature Description
VE	Emergence
V1	One leaf with collar visible
V2	Two leaves with collar visible
Vn	(n) leaves with collar visible
VT	Last branch of tassel is completely visible

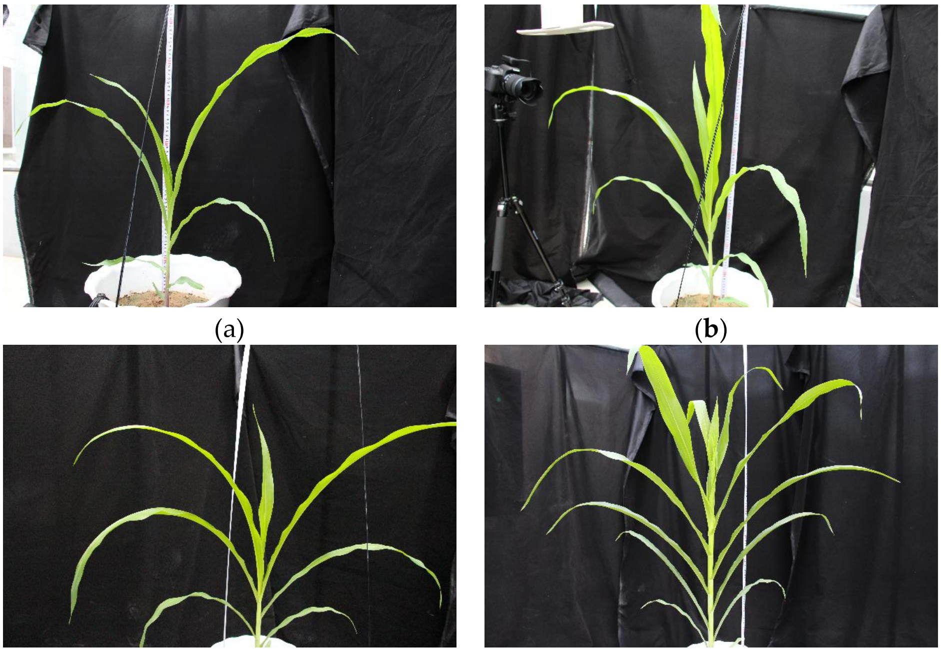

(a) (**b**)
(**c**) (**d**)

Figure 1. (**a**)–(**d**) shows maize image samples at different growth stage. From left to right, the growth days increased in turn.

4. Proposed Method

4.1. Setting Label According to the Number of Maize Leaves

We tried to use the classical network structure of CNN to directly regress the number of leaves, but the results were not suitable for maize plants. Sometimes the regression result approaches the average leaf number in the training samples. We assigned the labels according to the range of sample leaves; and assigned the same label to samples with similar leaf numbers. By observing the different samples of the maize plant, we find that the image samples with similar leaf numbers in the same species often have a lot of similarities in shape, size, and shooting angle. Therefore, we classified the samples with similar numbers of leaves into the same label and then utilized CNN to learn the standard features of samples of the same class; CNN can effectively learn features from the image.

Before we assigned labels to each sample, the distribution of leaf numbers of all samples was manually counted because sometimes one type of samples may occupy a large proportion of the data sets. Disregarding the leaf number distribution will lead to the unbalanced class distribution of the training samples, which will significantly influence the network training. The leaf number distribution in a sample set is shown in Figure 2.

The samples selected in this experiment correspond to four irrigation methods. Except for the first one, the other three were grown under different degrees of drought. Therefore, the distributions of the maximum and minimum values of each sample are all different. From the left figure image in Figure 2, we can find that the number of leaves mainly concentrated in 6,7 and 8. With the increase in the number of leaves, the corresponding number of samples gradually decrease. Because the number of leaves of a plant under suitable moisture increase steadily with time, whereas, a plant grows slowly in a drought state. The plant leaf number of the plant in drought is lower than that of the plant with suitable water content. Therefore, we needed to reset the label of the original sample and ensure that the reset label was uniformly distributed. The corresponding relationship between the leaf number and label can be found in Table 3.

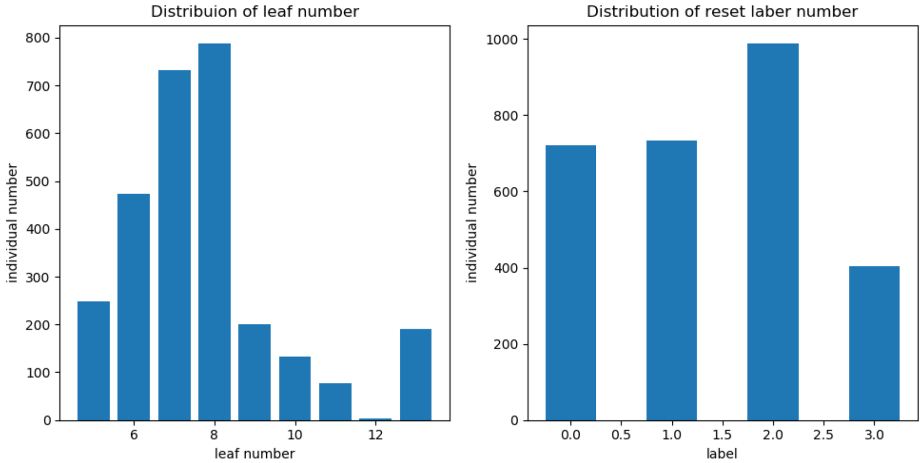


Figure 2. In the left picture, the horizontal axis shows the number of leaves, and the vertical axis represents the number of samples. The content of the right picture is similar to the left one, we reset the sample label according to the distribution of leaf number.

Table 3. Label range corresponding to the number of leaves.

Range of Leaf Number	Reset Label
[0,6]	0
(6,7]	1
(7,9]	2
(9,13]	3

4.2. Leaf Count Net

Our method refers to Google inception net V3 structure. The depth of the network can be maintained while the number of the parameters and the risk of over-fitting are effectively reduced. In our network, convolutional kernels of different sizes are used to extract multi-scale features. After the convolution operation, the feature maps are concatenated. However, there is a problem that must be considered, if all the feature maps are concatenated, the number of feature maps will be too large, which will increase the computational complexity. Therefore, we usually introduce 1*1 convolution operation to reduce the dimensionality of the feature maps. The number of output convolution kernels is less than the input feature maps, so as to reduce the dimensionality. Figure 3 shows the process of reducing the dimensionality of 256 feature maps to 128 feature maps by a 1×1 convolution kernel. Another operation to minimize parameters is to replace 3×3 with a two-layer convolution of 1×3 and 3×1, which can reduce the number of parameters by 33%.

When the training was finished, we took out feature maps in the middle of the network, and these feature maps had different scales. The green frame in Figure 4 represents the extracted feature maps, and their dimensions are ($53 \times 53 \times 128$), ($25 \times 25 \times 288$), ($3 \times 3 \times 64$). These feature maps will be used as multi-scale features to predict the final number of leaves. However, a secondary extraction of features is required before fitting, because all the feature maps have a large dimensionality, a property that will make the network very difficult to train. Therefore, we use FV to encode features and convert feature maps to vectors.

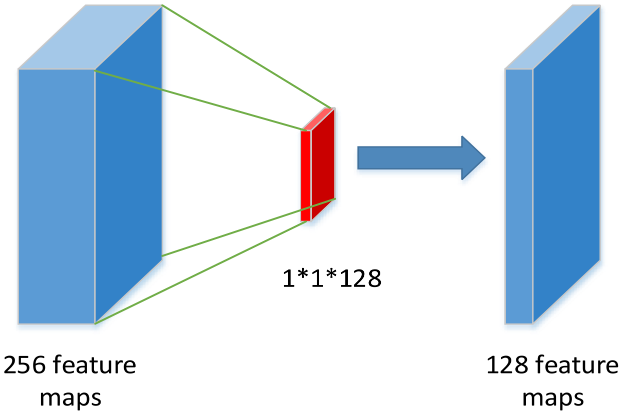


Figure 3. Feature map reducing dimension by using 1*1 convolution kernel.

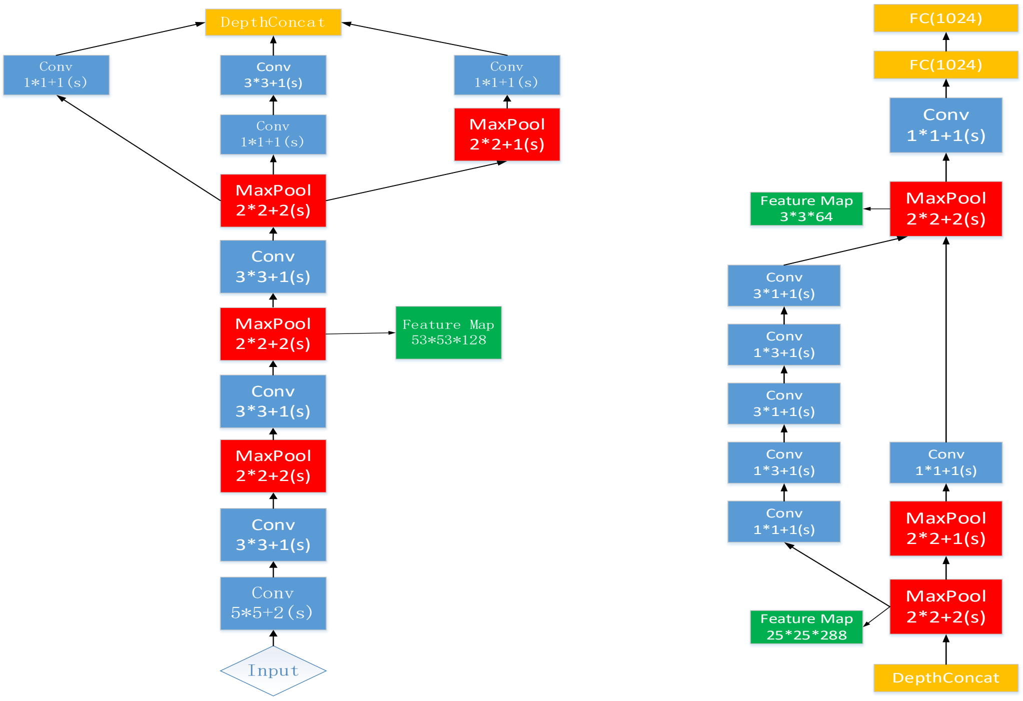


Figure 4. Leaf-count-net.

4.3. Coding Multi-Scale Feature Maps by Using Fisher Vector(FV)

We extract multi-scale feature maps from middle layers instead of from a single layer. Because the feature maps are compressed when they pass through the pooling layer, some features may be lost in this process, and the missing features may include useful information for the final regression results. As it is well known, the value of one point in a feature map represents a receptive field in the original image. For this large-scale feature representation, there is a high probability of missing some detailed information. For a local region in one image, it is not specific enough to represent the whole region using the value of one point. Therefore, extracting feature maps from different layers of the network can reduce the feature loss caused by the pooling process, and can better describe the original image from different scales. These feature maps can be regarded as some local feature descriptors, similar to SIFT [27]. Figure 5 describes the FV encoding process. After coding all three scale feature maps, high-level features are obtained, and all of the vectors are fused to form the feature vectors that we finally use to predict.

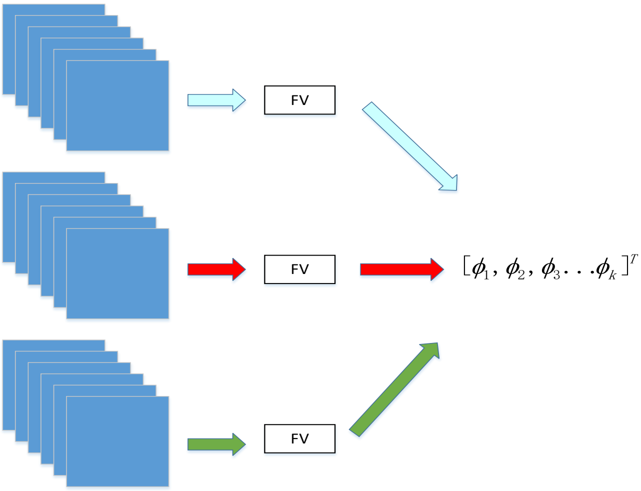


Figure 5. FV encodes feature maps for three scales and concatenates into one feature vector.

We were inspired by [28], in which the authors used the SIFT operator to extract the descriptor of the face image, and then the FV was used to code the descriptor. Finally, each face image was represented by a feature vector. In this paper, H, W, and D are used to describe height, width, and dimension, respectively. Therefore, for each feature map, the number of feature points is H × W, the feature dimension of each feature point is D. Then we can use $X = \{x_{H \times W}, t = 1, 2 \ldots H \times W\}$ to describe the image.

The FV encoding is based on the Fisher kernel, which groups a dense set of local features into a high-dimensional descriptor (features are better distinguished in a high-dimensional space) representing the image-level features. The descriptor uses the gradient, based on a probability function, to calculate the log-likelihood of the local features. In general, this is performed by fitting a parametric generative model, e.g. the Gaussian mixture model(GMM), and then the derivatives of the log-likelihood of the model are encoded with respect to the model parameters [28,29]. FV not only considers the gradient with respect to the weights but also considers the derivatives with respect to the mean and standard deviation [30,31]. We can assume that each feature point is independent and identically distributed. Then we use the GMM to represent the distribution of features. λ is a parameter in the GMM, and $\lambda = \omega_i, \mu_i, \Sigma_i, i = 1, 2 \ldots k$, where ω_i represents the probability that the feature points belong to the i-th Gaussian distribution, μ_i is the mean of the feature points at the i-th Gaussian distribution, and Σ_i represents the covariance between feature points; σ_i is standard deviation and $\sigma_i^2 = diag(\Sigma_i)$. Equations (1)–(5) describe the specific solution process:

$$N_k = \sum_{i=1}^{N} \gamma(i,k) \tag{1}$$

$$\gamma(i,k) = \frac{\pi_k N(x_i \big| \mu_k, \Sigma_k)}{\sum\limits_{j=1}^{\mathrm{K}} \pi_j N(x_i \big| \mu_j, \Sigma_j)} \tag{2}$$

$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N} \gamma(i,k) x_i \tag{3}$$

$$\Sigma_k = \frac{1}{N_k} \sum_{i=1}^{N} \gamma(i,k)(x_i - \mu_k)(x_i - \mu_k)^T \tag{4}$$

$$\omega_k = \frac{1}{N}\sum_{i=1}^{N}\gamma(i,k) \tag{5}$$

In Equations (1) and (2), $\gamma(i,k)$ represents the probability that the sample point x_t belongs to the kth Gaussian model. Subsequently, the partial derivatives of the GMM parameters are calculated. Then we obtain the gradient vectors $U_x = \left[\frac{\partial f(X|\lambda)}{\partial\omega_i}, \frac{\partial f(X|\lambda)}{\partial\mu_i}, \frac{\partial f(X|\lambda)}{\partial\sigma_i^d}\right]$, where the d in U_x represents the dimension. In U_x the dimensions of the three eigenvectors are k, k × D, and k × D respectively (k represents the number of Gaussian distribution), while ω_i has a constraint $\sum_i \omega_i$=1, there will be a decrease by one free variable. Finally, the dimension of U_x is (2D+1) × K−1. Then the gradient is normalized using the FV information matrix to get the Fisher feature vector. Equations (6)–(8) give the final representation of the Fisher eigenvectors. For the specific derivation process can refer to the literature [30].

$$\xi_{\omega_k}^X = \frac{1}{\sqrt{\omega_k}}\sum_{t=1}^{T}(\gamma_t(k) - \omega_k) \tag{6}$$

$$\xi_{\mu_k}^X = \frac{1}{\sqrt{\omega_k}}\sum_{t=1}^{T}\gamma_t(k)(\frac{x_t - \mu_k}{\sigma_k}) \tag{7}$$

$$\xi_{\sigma_k}^X = \frac{1}{\sqrt{\omega_k}}\sum_{t=1}^{T}\gamma_t(k)\frac{1}{\sqrt{2}}[\frac{(x_t - \mu_k)^2}{\sigma_k^2} - 1] \tag{8}$$

For each set of feature maps, a (2D+1) × K−1 dimensional feature vector descriptor is obtained. We encode the feature maps for the three middle layers in our network. Finally, the random forest algorithm is used to fit the features to predict the number of maize leaves. The detailed algorithm flow chart is illustrated in Figure 6.

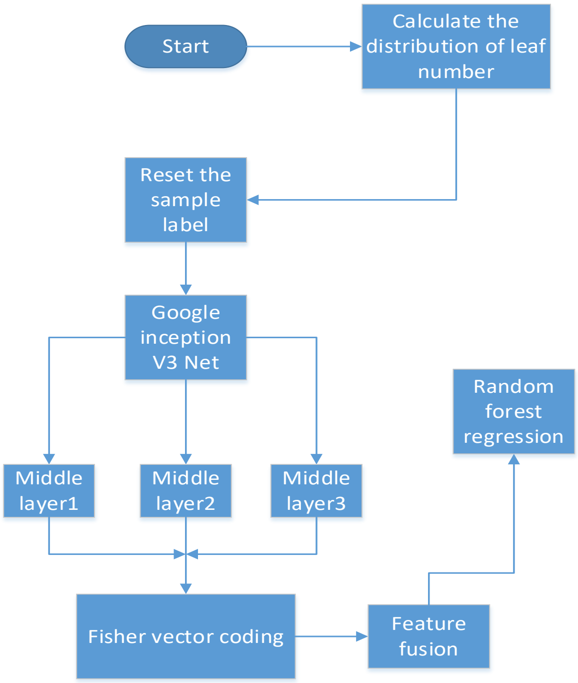


Figure 6. Presented all the steps in algorithm.

5. Results and Discussion

In this section, we evaluate the proposed leaf counting approach on an image data set of maize plant. First, we discuss the experimental settings and parameters. Next, we present the details of the training and testing samples. Then we show a comparison between the experimental results and existing methods to prove the effectiveness of our algorithm. In this paper, we assume that samples with similar numbers of leaves have similar leaf features. There was a high similarity of features learned by CNN for samples with the same label. When only the feature map of the last layer in the network was used, the feature discrimination of samples in the same label was small, and this is not conducive to the subsequent leaf number fitting (when setting labels for samples, samples with the same label may not have the same leaf number). Therefore, we extracted the feature maps from the middle layer in the same way, and the difference of these feature maps was greater than that of the last layer. Meanwhile, these features are a good complement to detailed features, which also explains why we used multi-scale features.

5.1. Implentation Details

(1). The framework of Python+Tensorflow have been used to build the network. Some training parameters are shown in Table 4.

Table 4. Part of training parameters in convolutional neural network.

Batchsize	Epochs	Learning Rate	L2 Weight Decay
32	200	10^{-3}	10^{-4}

(2). Our CNN-net trained and tested under Windows 10 64-operation system on Intel Core i7-8700 at 3.2GHZ with 32-GB RAM. The GPU is GTX 1080Ti.

(3). Finally, we use random forests to classify the number of maize leaves, the number of trees in the forest was set 70 and the max features was set 44(sqrt(features)).

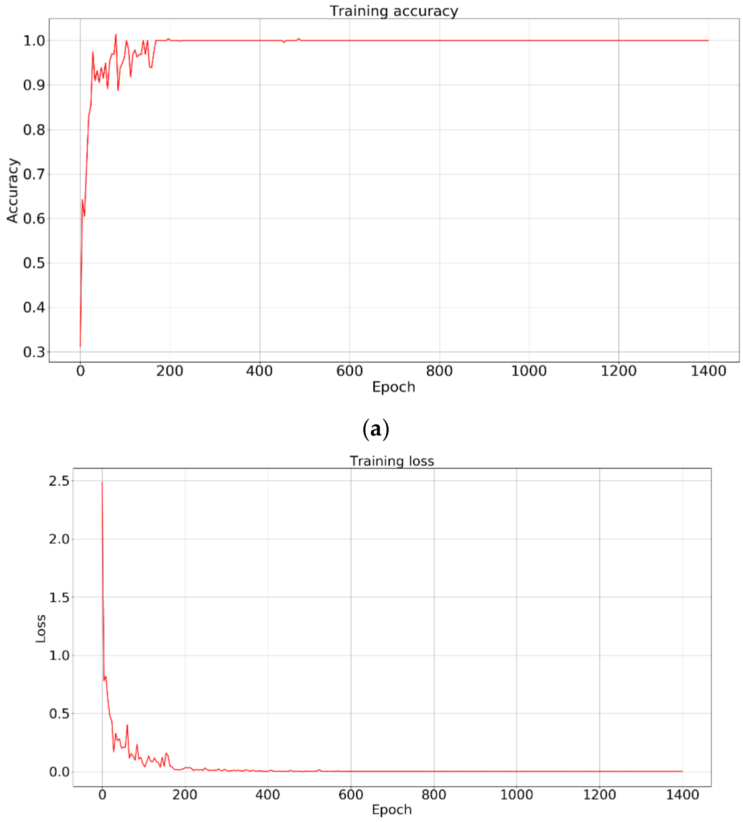
5.2. Image Data

In 2.4, we present the method to get image data samples. The samples of the four water levels were 701,644,851 and 649 in number (because of the problem of shooting angle and illumination, we removed some poor-quality samples). The total number of samples was 2845, of which 80% were training samples and 20% are testing samples. The numbers of final training samples and testing samples were 2276 and 569.

5.3. Experimental Results and Comparison with Other Methods

The training accuracy and training loss are shown in Figure 7. With an increase in training epochs, the model gradually converged. As we can see from Figure 7, with an increase in training epochs, the training loss converged quickly and training accuracy was close to 100% in 200 epochs. This indicates that the features learned by CNN are suitable for classification. It is reasonable to assign the same label for maize samples with similar leaf numbers. By training the classification of maize samples, the model can roughly determine the range of leaf numbers of a single maize sample. The label assigned to the sample represents the range of the leaf numbers. Therefore, the CNN model extracts feature by learning to predict the range of plant leaf numbers. As we know, a high accuracy rate is very helpful for further encoding feature maps. The correct rate directly reflects the feature extraction ability of our network. Test samples were not reserved because our aim was not to classify them. Finally, to avoid over-fitting, the weight model obtained by the 200th iteration was saved to extract the feature from the middle layer.

In FV coding, there exists an important parameter, parameter k (the number of Gaussian distribution), which should be assigned during the whole process. To select a reasonable parameter k, four water level samples from different growth stage were selected. The number of this part is 650. The result can be seen in Figure 8. We can see k = 77 has the best performance.



(**a**)

(**b**)

Figure 7. In (**a**,**b**), the vertical axis represents the correct rate and the loss value, respectively. And the horizontal axis represents the number of iterations.

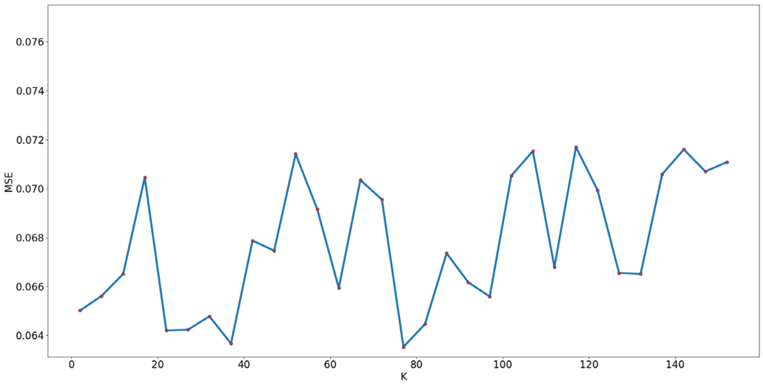


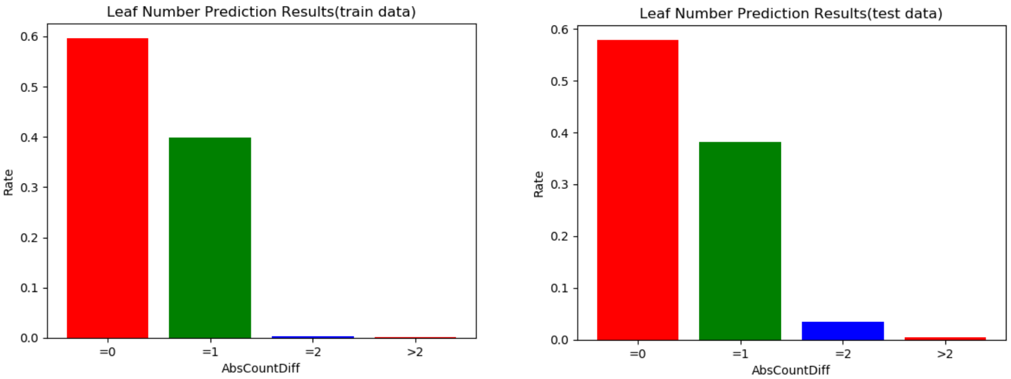
Figure 8. The result of MSE corresponding to different k values. The horizontal axis represents the different k values and the vertical axis represents the mean square error of true value and predict value.

The results of the comparison are shown in Table 5, "CountDiff" refers to the mean and standard deviation of the difference in count averaged over images. "AbsCountDiff" is the absolute of "CountDiff." "MSE" is the abbreviation for mean-square error [25]. Table 5 compares the results of our algorithm and the method of directly fitting with the deep neural network. Method (1) [32] and method (2) [33] shows that the method of directly fitting the leaf number with depth network has a high mean-square error. In the experiment, we found that the result of these method is close to the mean of the training samples, especially for samples with a large number of leaves. It can be seen from (3) and (4) that, the deep neural network is more powerful for sample feature extraction than that traditional local feature extraction algorithms, such as SIFT. In method (4), there is a large gap between the training result and test result, the over-fitting is serious. These imply that extracting multi-scale features from CNN combined with the traditional machine learning is more advantageous than the single CNN method for estimating the number of maize leaves.

Table 5. Compare with other methods.

Methods	AbsCountDiff	CountDiff	MSE
(1) Alex-net	Train:1.39	0.065	3.58
	Test:1.43	0.038	3.84
(2) VGG	Train:1.36	0.063	3.37
	Test:1.43	−0.019	3.78
(3) Proposed (Leaf-count-net+FV)	Train:0.17	−0.003	0.069
	Test:0.35	0.0018	0.31
(4) Sift+FV	Train:0.40	0.013	0.31
	Test:0.91	0.017	1.70

From Figure 9, we can see that most of the prediction errors are within one leaf. Comparing the (a) and (b), the range of error distribution of the training set and test set was consistent, and there was no large fluctuation in the distribution, which proves that our model is stable and has practical application value.



(**a**) (**b**)

Figure 9. Comparison between predicted and real values. (**a**) Train data. (**b**) Test data. The horizontal axis represents the absolute value of the difference between the predicted value and the true value, and the vertical axis represents the proportion of samples that satisfying the condition.

To verify the robustness of the proposed approach, a cross-validation experiment was designed. One type sample was reserved as the validation set and the other three samples were the training set. Therefore, each of the four water level samples was treated as a validation set. Then each group of experiments was repeated five times. The error bar of MSE was shown in Figure 10.

As can be seen from Figure 10, our model performed well for different water level samples, and performance was worst in the first validation compared to other times. In the first validation, samples 1, 2 and 3 were the training set and sample 4 was the validation set. Sample 4 was the most severely affected by drought stress and its leaves were fewer than those of other samples in the same period. This indicates that the feature vectors of sample 4 were quite different from those of the other three type samples.

5.4. Misclassified Image Analysis

In this study, some incorrect regress samples have been shown in Figure 11a–f are the samples of incorrect leaf count at different maize growth stages. The counting error of (a)–(d) are within 2 leaves. As can be seen from the samples, influenced by illumination there are some leaves in (a)–(d) are hard to distinguish with white pots. These leaves are located in the lower part of the maize. To (c) and (d), some leaves are withered because of the water shortage. These factors increase the difficulty of leaf counting and lead to counting errors. The counting error of (e) and (f) are more than 2 leaves. These samples have more leaves than (a)–(d) and the leaves are shaded from each other, therefore the counting error increases dramatically.

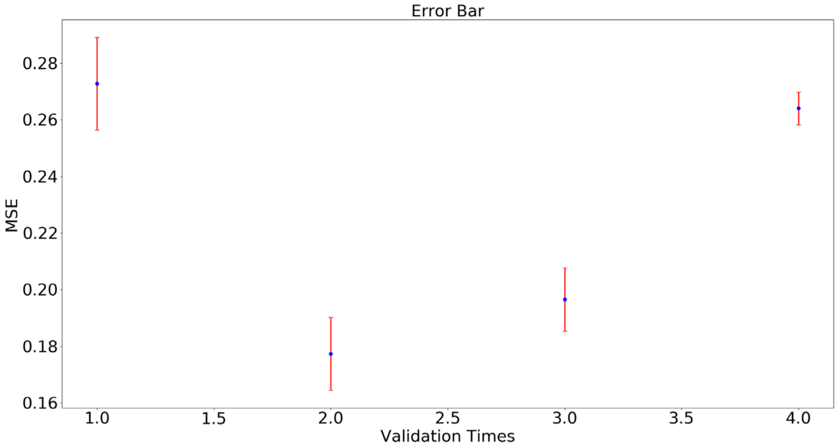


Figure 10. The performance of MSE. The vertical axis represents the MSE value. And the horizontal axis represents the validation times.

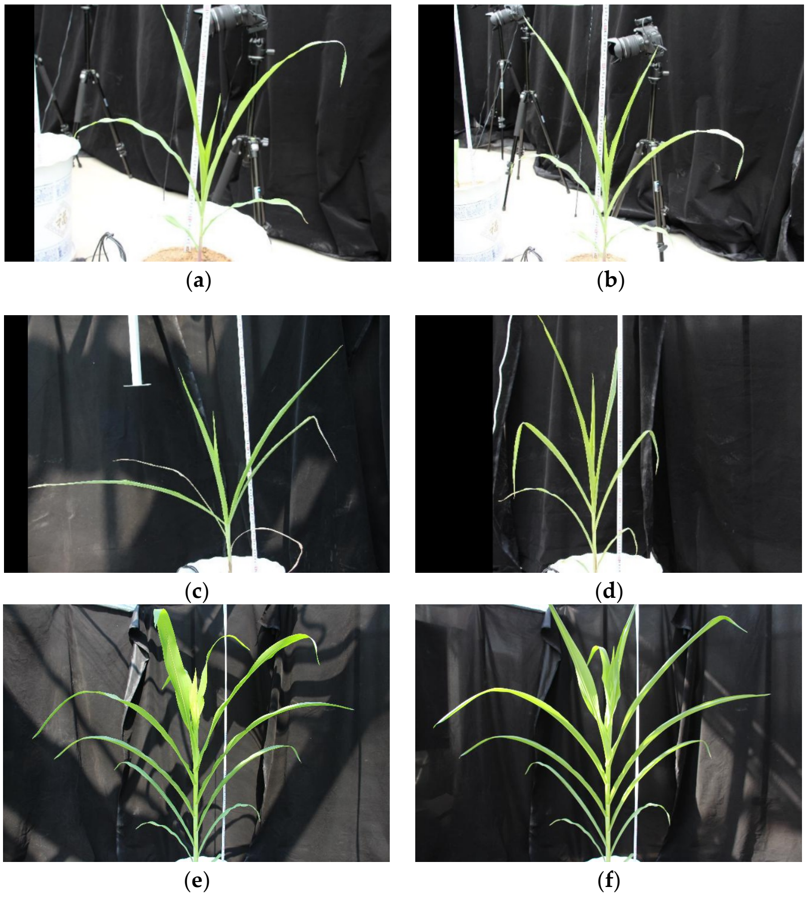


Figure 11. (**a**)–(**d**) are the samples of detection errors within 2 leaves. We preprocessed the original image. Because when obtaining the image, some leaves of other maize may as the noise sample appear in the image. These noise leaves are artificially filtered out. (**e**,**f**) are the samples of detection errors more than 2 leaves.

5.5. The Relationship Between Maize Leaf Number and Water Content

The number of leaves can reflect the water content of a maize plant. Figure 12 shows a line chart of the changes in the number of leaves over time for the four samples. The observation period was 32 days. The blue line represents the sample with suitable moisture. We can see that the number of leaves increase stepwise with time, and there is a clear demarcation line between the other three water-deficient samples. The green line represents moderate drought. Although it overlaps with the other two water-deficient samples, the total leaf number also rises stepwise; the rising rate is much lower than that of the suitable moisture sample and higher than that of the other two water-deficient samples. The discrimination between samples 3 and 4 was small, the number of leaves in sample 4 first increased and then decreased with time. By observing the actual image of the sample, it was found that the leaves in Sample 4 dropped seriously due to dry up. Sample 3 also exhibited the same condition. Our algorithm does not detect the drooping leaves, because their color is close to that of the soil. Therefore, the distributions of samples 3 and 4 are similar in Figure 12.

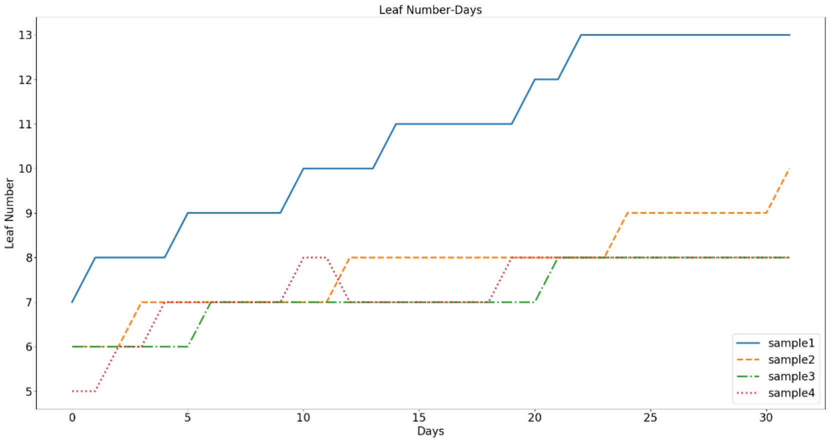


Figure 12. The number of leaves changing with time. Four different color lines represent four different samples.

6. Conclusions

In this paper, a deep learning approach combined with the traditional machine learning is been proposed. The CNN is responsible for extracting multi-scale features from different layers. Multi-scale features extraction can compensate for the loss of features caused by pooling layers. A FV maps the multi-scale features to a higher dimensional space. This can enhance the expression ability of the CNN and make the model perform well. Our method does not require segmentation, and new noise regions, which are associated with the error segmentation algorithm, are not introduced. The experimental results demonstrate that this method effectively counts maize leaves. However, for the samples with abnormal illumination and leaf occlusion, there are still large errors in counting. This indicates that some work still needs to be done.

In future extensions of this paper, we plan to enrich our data set by collecting more images of new maize species. Moreover, the current related studies are mainly conducted in a laboratory environment; future works should focus on field environment. In the preprocessing, the sample labels for different leaf numbers need to be manually marked to ensure the partitioned samples have similar morphological features and a uniform distribution. However, manual operation is inconvenient, and the automatic partitioning method needs be developed. Furthermore, to avoid redundancy of information, only three-layer feature maps were extracted according to the CNN structure. We cannot guarantee that these three-level feature maps are the best combination. In subsequent works, we will continue to study how to select the optimal combination.

Author Contributions: B.J. and P.W. proposed the method for this work, conducted the experiments and drafted the manuscript. S.Z. providing technical supports. M.L., Z.L. and Z.G. contributed in constructing the dataset, designing the experiments.

Funding: This work was supported by the National Key Technology Research and Development Program of the Ministry of Science and Technology of China.

Acknowledgments: The authors thank the Institute of Agricultural Resources and Regional Planning for providing maize samples.

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

References

1. Gebbers, R.; Adamchuk, V.I. Precision agriculture and food security. *Science* **2010**, *327*, 828–831. [CrossRef] [PubMed]
2. Li, Q.; Dong, B.; Qiao, Y.; Liu, M.; Zhang, J. Root growth, available soil water, and water-use efficiency of winter wheat under different irrigation regimes applied at different growth stages in north China. *Agric. Water Manag.* **2010**, *97*, 1682. [CrossRef]
3. Knezevic, S.Z.; Evans, S.P.; Blankenship, E.E.; Lindquist, A.J.L. Critical period for weed control: The concept and data analysis. *Weed Sci.* **2002**, *50*, 773–786. [CrossRef]
4. Furbank, R.T.; Tester, M. Phenomics–technologies to relieve the phenotyping bottleneck. *Trends Plant Sci.* **2011**, *16*, 635–644. [CrossRef] [PubMed]
5. Telfer, A.; Bollman, K.M.; Poethig, R.S. Phase change and the regulation of trichome distribution in *Arabidopsis thaliana*. *Development* **1997**, *124*, 645–654. [PubMed]
6. Walter, A.; Schurr, U. The modular character of growth in *Nicotiana tabacum* plants under steady-state nutrition. *J. Exp. Bot.* **1999**, *50*, 1169–1177. [CrossRef]
7. Koornneef, M.; Hanhart, C.; van Loenen-Martinet, P.; Blankestijn de Vries, H. The effect of daylength on the transition to flowering in phytochrome-deficient, late-flowering and double mutants of *Arabidopsis thaliana*. *Physiol. Plant.* **1995**, *95*, 260–266. [CrossRef]
8. Giuffrida, M.V.; Minervini, M.; Tsaftaris, S.A. Learning to Count Leaves in Rosette Plants. Available online: http://tsaftaris.com/preprints/Giuffrida_CVPPP2015.pdf (accessed on 11 March 2019).
9. Ye, M.; Cao, Z.; Yu, Z. An image-based approach for automatic detecting tasseling stage of maize using spatio-temporal saliency. In *MIPPR 2013: Remote Sensing Image Processing, Geographic Information Systems, and Other Applications*; International Society for Optics and Photonics: Bellingham, WA, USA, 2013.
10. Zhuang, S.; Wang, P.; Jiang, B.; Li, M.; Gong, Z. Early detection of water stress in maize based on digital images. *Comput. Electron. Agric.* **2017**, *140*, 461–468. [CrossRef]
11. Cui, S.L.; Tian, F. Face recognition method based on sift feature and fisher. *Comput. Eng.* **2009**, *35*, 195–197.
12. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2818–2826.
13. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
14. Jeon, W.S.; Rhee, S.Y. Plant leaf recognition using a convolution neural network. *Int. J. Fuzzy Logic Intell. Syst.* **2017**, *17*, 26–34. [CrossRef]
15. Choi, H. CNN output optimization for more balanced classification. *Int. J. Fuzzy Logic Intell. Syst.* **2017**, *17*, 98–106. [CrossRef]
16. Li, E.; Xia, J.; Du, P.; Lin, C.; Samat, A. Integrating multilayer features of convolutional neural networks for remote sensing scene classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 1–13. [CrossRef]
17. Yu, K.; Salzmann, M. Second-order convolutional neural networks. *Clinic. Immunol. Immunopathol.* **2017**, *66*, 230–238.
18. Huang, Z.; Van Gool, L. A Riemannian network for SPD matrix learning. *arXiv* **2016**, arXiv:1608.04233.

19. Akodad, S.; Bombrun, L.; Yaacoub, C.; Berthoumieu, Y.; Germain, C. Image classification based on log-Euclidean Fisher Vectors for covariance matrix descriptors. In Proceedings of the 2018 Eighth International Conference on Image Processing Theory, Tools and Applications (IPTA), Xi'an, China, 7–10 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–6.
20. Ren, M.; Zemel, R.S. End-to-end instance segmentation with recurrent attention. *arXiv* **2017**, arXiv:1605.09410.
21. Donahue, J.; Anne Hendricks, L.; Guadarrama, S.; Rohrbach, M.; Venugopalan, S.; Saenko, K.; Darrell, T. Long-term recurrent convolutional networks for visual recognition and description. *arXiv* **2015**, arXiv:1411.4389v4.
22. Pape, J.M.; Klukas, C. Utilizing machine learning approaches to improve the prediction of leaf counts and individual leaf segmentation of rosette plant images. In Proceedings of the Computer Vision Problems in Plant Phenotyping Workshop, Swansea, UK, 7–10 September 2015.
23. Pape, J.M.; Klukas, C. *3-D Histogram-Based Segmentation and Leaf Detection for Rosette Plants. Asia-pacific Conference on Conceptual Modelling*; Australian Computer Society, Inc.: Sydney, Australia, 2015.
24. Dobrescu, A.; Giuffrida, M.V.; Tsaftaris, S.A. Leveraging multiple datasets for deep leaf counting. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshop (ICCVW), Venice, Italy, 22–29 October 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 2072–2079.
25. Aich, S.; Stavness, I. Leaf counting with deep convolutional and deconvolutional networks. *arXiv* **2017**, arXiv:1708.07570.
26. Aslam, M.; Maqbool, M.A.; Ceng, R. *Drought Stress in Maize (Zea mays L.)*; Springer Briefs in Agriculture: Berlin, Germany, 2015; Chapter Effect of Drought on Maize; pp. 5–17.
27. Lowe, D.G. Object recognition from local scale-invariant features. In Proceedings of the 1999 Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; IEEE: Piscataway, NJ, USA, 1999; Volume 2, pp. 1150–1157.
28. Simonyan, K.; Parkhi, O.M.; Vedaldi, A.; Zisserman, A. *Fisher Vector Faces in the Wild*; University of Oxford: Oxford, UK, 2013; Volume 2, p. 4.
29. Jaakkola, T.; Haussler, D. Exploiting Generative Models in Discriminative Classifiers. Available online: https://papers.nips.cc/paper/1520-exploiting-generative-models-in-discriminative-classifiers.pdf (accessed on 11 March 2019).
30. Perronnin, F.; Dance, C. Fisher Kernels on Visual Vocabularies for Image Categorization. In Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 17–22 June 2007; IEEE: Piscataway, NJ, USA, 2008.
31. Koh, J.E.; Ng, E.Y.; Bhandary, S.V.; Hagiwara, Y.; Laude, A.; Acharya, U.R. Automated retinal health diagnosis using pyramid histogram of visual words and Fisher vector techniques. *Comput. Biol. Med.* **2018**, *92*, 204–209. [CrossRef] [PubMed]
32. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet Classification with Deep Convolutional Neural Networks. Available online: https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf (accessed on 11 March 2019).
33. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.