
Article

Image-based wheat fungi diseases identification by deep learning

Mikhail A. Genaev^{1,2,3}, Ekaterina S. Skolotneva^{1,2}, Elena I. Gulyaeva⁴, Elena A. Orlova⁵, Nina P. Bechtold⁵ and Dmitry A. Afonnikov^{1,2,3,*}

¹ Institute of Cytology and Genetics of Siberian Branch of the Russian Academy of Sciences, Novosibirsk, Russia; mag@bionet.nsc.ru; skolotnevaES@bionet.nsc.ru

² Novosibirsk State University, Novosibirsk, Russia

³ Kurchatov Genomics Center of the Institute of Cytology and Genetics of Siberian Branch of the Russian Academy of Sciences, Novosibirsk, Russia

⁴ All Russian Institute of Plant Protection, Saint Petersburg-Pushkin, Russia; eigulyaeva@gmail.com

⁵ Siberian Research Institute of Plant Production and Breeding – Branch of the Institute of Cytology and Genetics, Siberian Branch of Russian Academy of Sciences, Novosibirsk, Russia; orlova.lena10@yandex.ru; Telichkinanina@mail.ru

* Correspondence: ada@bionet.nsc.ru; Tel.: +7 (383) 363-49-63

Abstract: Diseases of cereals caused by pathogenic fungi can significantly reduce crop yields. Many cultures are exposed to them. The disease is difficult to control on a large scale, thus one of the relevant approaches is the crop field monitoring, which helps to identify the disease at an early stage and take measures to prevent its spread. One of the effective control methods is disease identification based on the analysis of digital images with the possibility of obtaining them in field conditions using mobile devices. In this work, we propose a method for the recognition of five fungal diseases of wheat shoots (leaf rust, stem rust, yellow rust, powdery mildew, and septoria), both separately and in combination, with the possibility of identifying the stage of plant development. A set of 2414 images of wheat fungi diseases (WFD2020) was generated, for which expert labeling was performed by the type of disease. WFD2020 data are available freely at <http://wfd.sysbio.ru/>. In the process of creating this set, a method was applied to reduce the degeneracy of the training data based on the image hashing algorithm. The disease recognition algorithm is based on the convolutional neural network with the EfficientNet architecture. The best accuracy (0.942) was shown by a network with a training strategy based on augmentation and transfer of image styles. The recognition method was implemented by the authors as a bot on the Telegram platform, which allows assessing plants by lesions in the field conditions.

Keywords: wheat, leaf rust, powdery mildew, septoria, stem rust, yellow rust, image recognition, deep learning, convolutional neural network, phenotyping

1. Introduction

Wheat is one of the world's main crops and food sources for human consumption [1]. Wheat accounts for the largest planting area, and the food security of the population of most countries of the world depends on its yield. One of the main factors affecting wheat yield is fungi diseases: rust, septoria of leaves and ears, and powdery mildew [2] (Figure 1).

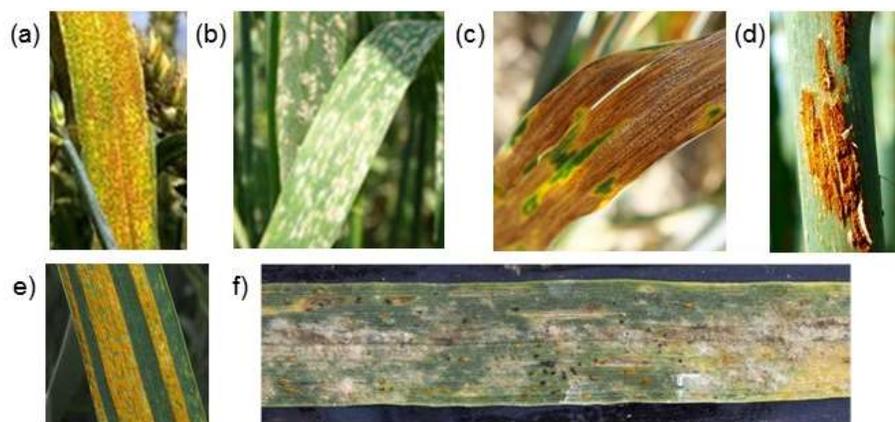


Figure 1. Examples of digital images showing manifestations of wheat plant disease: (a) leaf rust; (b) powdery mildew; (c) septoria; (d) stem rust; (e) yellow rust; (f) multiple diseases.

Leaf rust, powdery mildew, and septoria (pathogens *Puccinia triticina* Erikss., *Blumeria graminis* (DC.) Speer, *Zymoseptoria tritici* Rob. and *Parastagonospora nodorum* Berk.) are cosmopolitan pathogens. They are observed in the phytopathogenic complex on grain crops widely. However, the total damage caused by the causative agents of the above-listed diseases does not exceed 20% and is controlled by the timely use of fungicides [3]. The consequences of the epiphytotic of yellow and stem rust, leading to grain shortages of more than 30%, are of serious economic importance. In general, grain yield losses from these diseases, depending on the region and season conditions, can vary from 15 to 30% or more [4,5]. The most effective way to combat these diseases is their prevention and timely implementation of protective actions [4, 6, 7]. However, such a strategy is impossible without a timely and correct diagnosis of pathogens. In this case, it is important to identify diseases at the stage of seedlings, since at later stages of plant development, resistance to the pathogen is higher [8]. In turn, the effectiveness of such diagnostics largely depends on how accurate, labor-intensive, and resource-intensive it is.

Visual assessment of the diseases has been the main diagnostic method throughout the history of wheat cultivation. It allows identifying of plaque, pustules, spots, or necrosis [9, 10]. It requires the training of specialists in phytopathology and a lot of routine work in addition to the observations (keeping a record book, statistical processing of observations, etc.). In recent decades, molecular, spectral methods, and methods based on the analysis of digital images have appeared and have become widespread [11, 12]. These approaches differ both in the labor intensity, as well as in its cost and accuracy. The methods, using the analysis of digital RGB images, are based on determining changes in the color, texture, and shape of plant organs that arise as a result of changes in their pigment composition under the influence of the vital activity of pathogens [13, 14, 15, 16, 17]. The advantages of such methods are the low cost of monitoring equipment (it is sufficient to use a digital camera or a mobile phone) and the high monitoring speed. The disadvantages include low sensitivity (in comparison, for example, with spectral methods, see [12]).

Recently the technologies for plant diseases monitoring based on digital RGB images have received a powerful impulse due to the improvement of machine learning methods based on the use of neural network algorithms. A feature of deep learning neural networks in comparison with other methods is the multilayer architecture of neurons, in which the next layer uses the output of the previous layer as input data to derive ideas regarding the analyzed objects [18]. For example, for such an important task as image labeling, some of the most successful methods are convolutional neural networks (CNNs), for which several architecture options are used. Among the first types of CNN architecture were AlexNet [19] and VGG [20]. Further development of these approaches

made it possible to improve the convergence of the proposed algorithms (ResNet network) [21], reduce the number of parameters due to deep convolution (MobileNet network) [22], improve model training results due to adaptive recalibration of responses across channels (SENet network) [23]. These advances have expanded the applicability of deep learning neural networks. Moreover, they have also demonstrated exceptional success on complex problems of plant phenotyping [18; 24; 25; 26].

A number of deep learning methods, which have been developed to identify wheat diseases using digital RGB images, have proved to be effective. In the work by Barbedo [27], CNN of the GoogLeNet architecture was used to detect lesions in the leaf image, and on this basis, wheat diseases such as blast, leaf rust, tan spot, and powdery mildew were identified. In the work by Picon *et al.* [28], a method was developed for identifying four types of wheat diseases, taking into account their stage of development (septoria, tan spot, two types of rust) based on deep CNNs. Lu *et al.* [29] have developed an automatic system for diagnosing six types of wheat diseases, which recognizes the type of disease and localizes the lesions in the image, obtained in the field conditions.

It is important to note a number of problems that have to be faced when developing methods for recognizing diseases based on image analysis. First of all, to build a successful algorithm for disease recognition the set of a large number of labeled images is required [30]. The availability of such data in the public domain is the basis for improving modern methods of image recognition for plant phenotyping [31, 32] and identification of pathogens [33]. To increase the training sample, the information from several datasets is often combined, but in this case, it may turn out that the same images can fall into one training set, leading to its degeneracy.

Another difficulty is the simultaneous presence of symptoms caused by different diseases [34]. On the one hand, the problem of multiple classifications is difficult to solve, since it requires a larger volume of images for correct classification, in which various combinations of lesions will be presented in sufficient numbers. On the other hand, the visual manifestations of each of the diseases can be very similar (see Figure 1), which makes it difficult to make a correct classification.

Finally, one of the important conditions for creating a recognition method is the possibility of using it in field conditions [35]. This increases the efficiency of disease monitoring and, consequently, the chances of successful plant treatment by fungicides. One of the approaches is using mobile devices for this, both for semi-automatic determination of the degree of plant damage [36] and fully automatic analysis, including computer vision methods [37, 38] and deep learning networks [29, 28, 39,40].

Here we propose a method for the recognition of five fungi diseases of wheat shoots (leaf rust, stem rust, yellow rust, powdery mildew, and septoria), both separately and in combination, with the possibility of identifying the stage of plant development. During development, the Wheat Fungi Diseases (WFD2020) dataset of 2414 wheat images was formed, for which expert labeling was performed by the type of disease. In the process of creating this set, a data redundancy reduction procedure was applied based on the image hashing algorithm. The disease recognition algorithm is based on the use of a network with the EfficientNet architecture. The recognition method was implemented by the authors as a bot on the Telegram platform, which allows assessing plants by lesions in the field conditions.

2. Materials and Methods

2.1. Image Dataset

The images were obtained from various sources. Some images were taken from the dataset presented at the machine learning challenge "ICLR Workshop Challenge # 1: CGIAR Computer Vision for Crop" on the <https://zindi.africa> platform from January 29, 2020 to March 29, 2020. Some images were obtained from the plant disease detection platform (PDDP) dataset (<http://pdd.jinr.ru/>) [39]. Some images were taken from the

Internet using the Google Images service, and the other part of the dataset was obtained by the authors of the current work by plant imaging in laboratory and field conditions. The original dataset included 3076 images.

2.2. Filtration of images

A preliminary analysis of the original set of images showed that among them there were completely duplicated images of the same object with different levels of compression, resolution, or brightness. All these factors lead to redundancy of information in the training data, which could affect the training process of the classification method and its testing. It was decided to compose a non-redundant set of unique images based on the initial data. To do this, the hashing algorithm aHash was used, implemented in the python ImageHash library (<https://github.com/JohannesBuchner/imagehash>), the `average_hash()` method. This algorithm returns a hash value for an image file, a 64-bit integer that does not change when scaling, changing the aspect ratio, and slightly changing the contrast and brightness of the image. All initial images were grouped by hash function value. If the group includes more than one image, the image from the file with the largest size was selected as its representative. In particular, 612 images were excluded from the original Zindi.africa dataset, which included 1486 images.

As a result of such filtering, a non-redundant set of 2414 color images of wheat in jpeg format (Wheat Fungi Diseases dataset, WFD2020) was obtained. The total size of the image files was 4.84 GB. The average image resolution in this dataset was ~ 5MP.

A description of the number of image files from various sources is presented in Table 1. It includes data from the zindi.africa project, the platform for plant disease detection (PDDP), images obtained through the Google Images service, images obtained by the authors in field and laboratory conditions in 2016-2020 in St. Petersburg, and images obtained by the authors in field and laboratory conditions in 2017-2020 in the Novosibirsk Region.

Table 1. Sources of images in the dataset Wheat Fungi Diseases.

| Source | Number of images | Reference |
|------------------|------------------|---|
| Zindi.africa | 874 | https://zindi.africa/competitions/iclr-workshop-challenge-1-cgiar-computer-vision-for-crop-disease/data |
| Google Images | 259 | - |
| PDDP | 121 | [44] |
| Saint Petersburg | 367 | This work |
| Novosibirsk | 793 | This work |

2.3. Labeling of images

Each image in the WFD2020 dataset has been manually annotated. The images were classified into healthy plants (label 'healthy') and by six types of fungal diseases: leaf rust, powdery mildew, septoria, stem rust, and yellow rust. Additionally, the images were classified according to the stage of plant development: whether the plant is a seedling (label 'seedling') or not (absence of such a label). Images could be tagged with several labels at the same time. This occurs when the plant was affected by several diseases or was a seedling.

2.4. Validation scheme and multi-label data stratification

For machine learning, the images were split into 3 subsamples: training 1454 (60%) images, which are used to train the model; validation 480 (20%), images for choosing the best model during training; test/hold out 480 (20%) images to assess the accuracy of the

selected model. To ensure a balanced distribution of multilable classes in each subsample, the iterative stratification algorithm was used, implemented in the `iterative_train_test_split()` method of the `model_selection` python module of the `skmultilearn` library [41].

2.5. Neural network structure

We used the EfficientNet network architecture proposed earlier in the work [46], which has shown high efficiency in image labeling. EfficientNet is a family of CNN of similar architecture, which differ from each other in the depth of the layers, their width, and the size of the input image, while maintaining the ratios between these sets of parameters. On the ImageNet task, the EfficientNet-B7 model with the number of parameters 66 M achieved an accuracy of 84.3% [42]. Here we used the EfficientNet-B0 architecture, which is the simplest and fastest. Its structure is shown in Figure 2. It includes seven types of sequential blocks built on the basis of the Conv and MBConv layers [43]. In MBConv, blocks consist of a layer that first expands and then compresses the channels, so direct connections are used between bottlenecks that connect much fewer channels than expansion layers. This architecture has in-depth separable convolutions that reduce calculation compared to traditional layers [22].

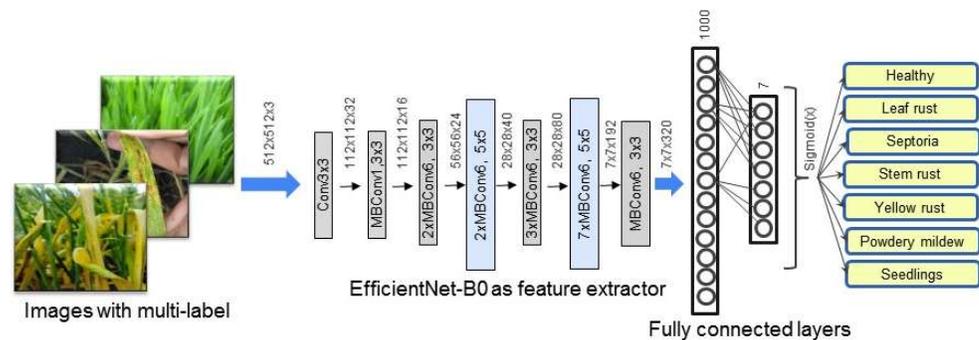


Figure 2. Diagram of the neural network used to predict fungal diseases of wheat.

The EfficientNet-B0 network structure is implemented using the `pytorch v1.7.1` [44] and `catalyst v20.10.1` (<https://github.com/catalyst-team/catalyst>) frameworks. The initial values of the network weights were taken from the `efficientnet_pytorch` repository (<https://github.com/lukemelas/EfficientNet-PyTorch>). In order to implement the classification of images converted using the EfficientNet-B0 network, two fully connected layers were added to the head of the network: 1000 nodes for the inner layer and 7 nodes for the output layer (according to the number of predicted label types). The Dropout (0.5) regularization was applied to the hidden fully connected layer, which randomly zeroes out half of the layer weights. As a loss function, we used a combination of the activation function `Sigmoid(x)` and the binary cross entropy `BCEWithLogitsLoss()`, implemented in the `nn` module of the `pytorch` library. The type of label in the final implementation of the network was determined by the threshold value of the corresponding neuron. If this value was greater than 0.5, then the image was considered to contain the corresponding label. Thus, the image could be marked with several labels at the same time.

2.6. Accuracy assessment

For each image, the authors' network predicted 7 binary numbers, each of which characterized the presence or absence of a particular label. To assess the accuracy of the method on a sample of test images, we compared the predicted set of such numbers and the true set for each image, so if the test sample contained M images, the authors made $7M$ of such binary comparisons and based on them calculated the true positive (TP) values, true negative values (TN), as well as the total number of positive (1, P) and

negative (0, N) values. Based on these values, at the end of each epoch, the accuracy (ACC) value (`catalyst.metrics.accuracy.multilabel_accuracy()` method) was calculated for images from the validation set according to the formula $ACC = (TP + TN) / (P + N)$.

2.7. Training strategies

To optimize the network parameters in the training process using the stochastic gradient descent method, the Adam algorithm was used (implemented in the `optim` module of the `pytorch` library [44]) with the regularization parameter `weight_decay=1e-6` and the initial learning rate `lr=1e-4`. The learning rate varied according to the `CosineAnnealingWarmRestarts` heuristic (implemented in the `optim.lr_scheduler` package of the `pytorch` library) with the parameters `T_0=10`, `T_mult=2`, `eta_min=1e-6`. The learning was performed by 150 epochs, the batch size was 16. The network was trained on RGB images transformed to 512x512 px while maintaining proportions.

Three training strategies were considered.

(1). Basic strategy `EfficientNet-B0_baseline`. Training the model using the above-described methods without any additional modifications to the data and training parameters.

(2). `EfficientNet-B0_augm` strategy with augmentation. The model was trained using the above-described methods with the introduction of random noise in the loss of 1%, according to the label smoothing method [45]. Image augmentation was used by the algorithms implemented in the `Albumentations` library [46]. Transformations to images from the training sample were applied with a probability of 0.3 and included: rescaling using the method `IAAPerspective(scale=(0.02, 0.05))`; random rotation ± 15 degrees `ShiftScaleRotate(rotate_limit=(-15.0,15.0))`; vertical and horizontal reflections using the `HorizontalFlip()` and `VerticalFlip()` methods; changing brightness and contrast using the `RandomBrightnessContrast(brightness_limit=0.2, contrast_limit=0.2)` method, as well as random masking of square areas using the `Cutout()` method [47].

(3) The strategy with augmentation and transfer of image styles `EfficientNet-B0_FDA`. This strategy used transformations as in the `EfficientNet-B0_augm` strategy with an additional transformation of the style transfer using the FDA algorithm [48], the `FDA()` method of the `domain_adaptation` package of the `Albumentations` library with the mask parameter $\beta=0.01$.

One of the options for using the style transfer method in the learning process is to choose a style from a randomly selected sample image. However, our preliminary analysis has shown that the approach is more effective when several typical styles are first identified for a sample, characteristic images are selected for these styles and their style is used in the learning process.

To highlight typical styles in the sample, 1000 random images were selected from it. For each image, the Fast Fourier Transform (FFT) method of the `fft2()` package of the `numpy` library was used. For the resulting transformation, the `ifftshift()` method was applied, which shifts the low-frequency components of the spectrum, carrying information about the image style, to the center of the spectrum. After that, the elements of the central part of the amplitude component of the spectrum were extracted with a size of 30x30. Thus, for each image, we obtained 900 characteristics, the amplitude components of the low-frequency part of the spectrum. The t-SNE method (t-distributed stochastic neighbor embedding) [49] was applied to this data to isolate two components. After that, the images were clustered in the space of two components by the k-means method ($k=10$). For each cluster, the image closest to the centroid was determined, which represented the corresponding type of image style. Thus, 10 source images were selected to transfer their style to other images in the training process.

A computer with a GPU Nvidia RTX 2080ti was used to train the models. Depending on the type of the applied transformations, the training time for one epoch ranged from 1 to 1.5 minutes.

Based on the results of the training process, the optimal values of the network parameters were determined by the epoch for which the maximum values of the ACC metric were achieved, calculated on the images of the validation sample.

2.8. Telegram bot

Based on the model, we developed a bot in the Telegram messenger wheat_healthy_bot (https://t.me/wheat_healthy_bot). To create the bot, the authors used the python aiotg library (<https://github.com/szastupov/aiotg>), the RabbitMQ message handling system (<https://www.rabbitmq.com>) and the FastAPI framework (<https://fastapi.tiangolo.com>) to build REST API.

3. Results

3.1. Wheat Fungi Diseases Dataset

The WFD2020 dataset used for training is available for public access at the website <http://wfd.sysbio.ru/>. All images can be downloaded from the site in the form of a zip archive; expert annotation for all images is available in csv format for annotations for training, validation and training subsamples that were used in the process of training the models. The images with a label of a certain class can be viewed on the website. The filename for the images matches the value of aHash. Since the hash values for similar images calculated are close in Hamming distance, similar images will be in files with similar names, and to search for similar images, it is sufficient to sort them by the file name. An example of three images of which two are similar, and the third one differs from them, along with the values of the hash functions, is shown in Figure 3.

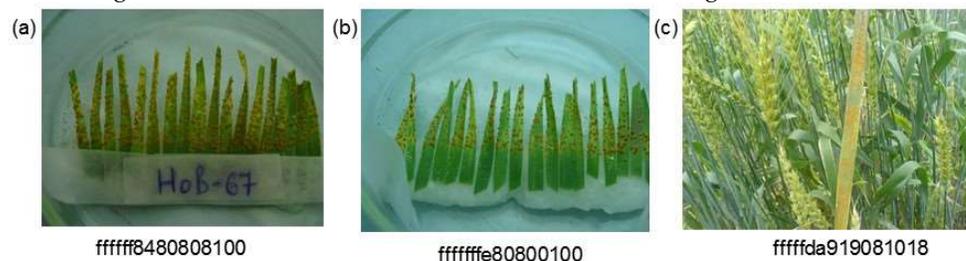


Figure 3. Examples of images of wheat leaves and shoots affected by leaf rust. Below the image, there are the file names (hash function values), which are similar for similar images (a) and (b) but differ significantly for different images (a), (c) and (b), (c).

3.2. Results of labeling a set of images

After removing redundancy, the WFD2020 image set included 2414 wheat images. As a result of annotation, the images from the dataset have 3147 labels, 1.30 labels per image in average. The number of occurrences of labels of each class in the dataset is presented in Table 2.

Table 2. The number of labels of each class in the WFD2020 dataset.

| Class | Number of labels (%) |
|---------------|----------------------|
| leaf rust | 655 (20.8%) |
| stem rust | 591 (18.8%) |
| yellow rust | 571 (18.1%) |
| powder mildew | 277 (8.8%) |
| septoria | 185 (5.9%) |
| seedlings | 569 (18.1%) |
| healthy | 299 (9.5%) |
| Total | 3147 (100%) |

Table 3 shows the number of occurrences of various combinations of labels in the WFD2020 dataset.

Table 3. The number of images with different combinations of labels in the WFD2020 dataset.

| Class | Number of labels (%) |
|-------------------------------------|----------------------|
| stem rust | 426 |
| yellow rust | 362 |
| leaf rust | 334 |
| healthy | 253 |
| leaf rust, seedlings | 198 |
| seedlings, yellow rust | 197 |
| powdery mildew | 187 |
| septoria | 133 |
| seedlings, stem rust | 127 |
| leaf rust, powdery mildew | 65 |
| healthy, seedlings | 46 |
| leaf rust, stem rust | 20 |
| leaf rust, septoria | 17 |
| powdery mildew, septoria | 12 |
| leaf rust, powdery mildew, septoria | 11 |
| septoria, stem rust | 11 |
| leaf rust, yellow rust | 7 |
| stem rust, yellow rust | 2 |
| powdery mildew, stem rust | 2 |
| leaf rust, stem rust, yellow rust | 2 |
| septoria, yellow rust | 1 |
| leaf rust, seedlings, stem rust | 1 |
| Total | 2414 |

3.3. Clustering images by style

One of the strategies of the analysis, EfficientNet-B0_FDA, implied a procedure for transferring image styles in the training set during the augmentation process. In order to select the characteristic image styles in the sample based on the low-frequency part of their Fourier spectra, the authors classified the styles as described in the "Training Strategies" section for 1000 randomly selected training sample images. The results are presented in Figure 4. This diagram shows the distribution of images in the space of two components obtained using the t-SNE dimensionality reduction method for low-frequency components of Fourier spectra of images. The distribution of images on the diagram is non-uniform and a number of clearly visible clusters can be distinguished in it. Clustering images in the space of these two components by the k -means method allowed obtaining 10 characteristic image styles, the centroids of which are indicated in Figure 4 by the corresponding numbers. In particular, cluster 5 corresponds to images of wheat in plots in the field conditions. Cluster 2 includes the images obtained under laboratory conditions for leaves affected by diseases, in Petri dishes. In the center of the diagram (clusters numbered 4,1 and 7), there are images containing hands. Thus, the sample of images was heterogeneous in their styles since the images acquired using different protocols were used to obtain it. The 10 characteristic types of styles identified by cluster analysis were further used in the training process according to the EfficientNet-B0_FDA strategy.

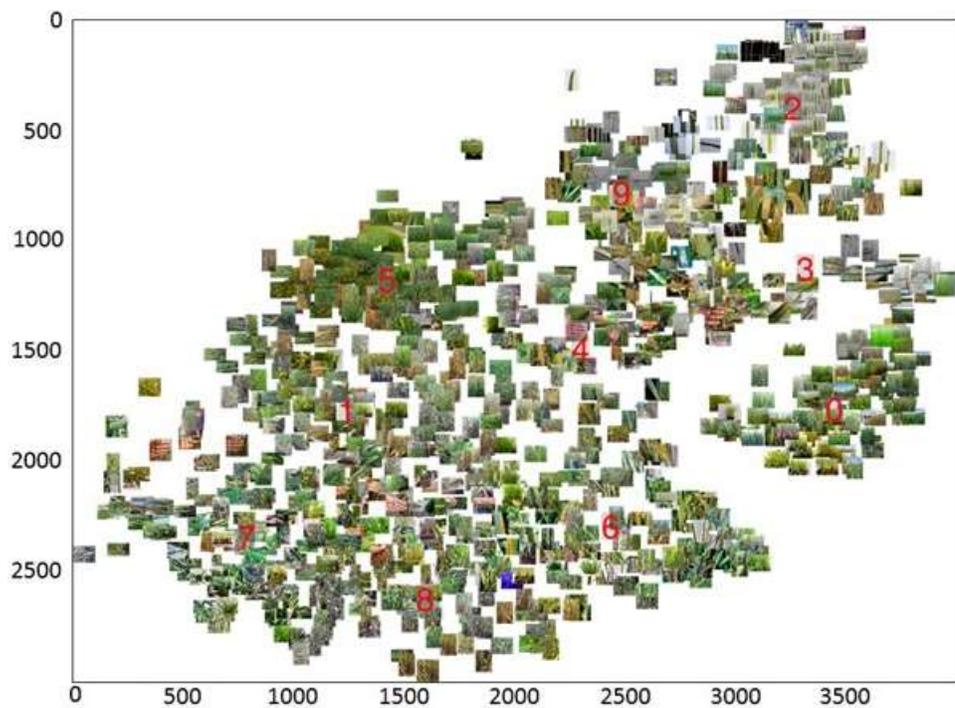


Figure 4. Clustering 1000 random images from the training sample based on the low-frequency part of the spectrum obtained by the fast Fourier transform.

3.4. Assessment of image classification accuracy

An example of the learning curves obtained for the EfficientNet-B0_FDA strategy are shown in Figure 5. It can be seen from the figure that for the chosen number of epochs, 150, the Loss and ACC values for the training and validation data become stationary. The ACC values after becoming stationary for the validation sample are within 0.92-0.94, reaching a maximum value of 0.943 at epoch 110. The network parameters obtained for this epoch were used by the authors for further testing on the hold-out sample, the accuracy was 0.942 (see Table 4). The difference in accuracy between validation and deferred samples was 0.1%, which indicates that there is no overfitting effect.

For the EfficientNet-B0_baseline model, the stationary values of Loss and ACC were reached already at epoch 10, and the optimal parameters were obtained for epoch 18 (Table 3). On the test sample, the accuracy of this network reached 0.933. For the EfficientNet-B0_augm strategy, it became stationary at epoch 60, and the optimal parameters were obtained for epoch 140. The accuracy of this model on the test sample was 0.939 (Table 4).

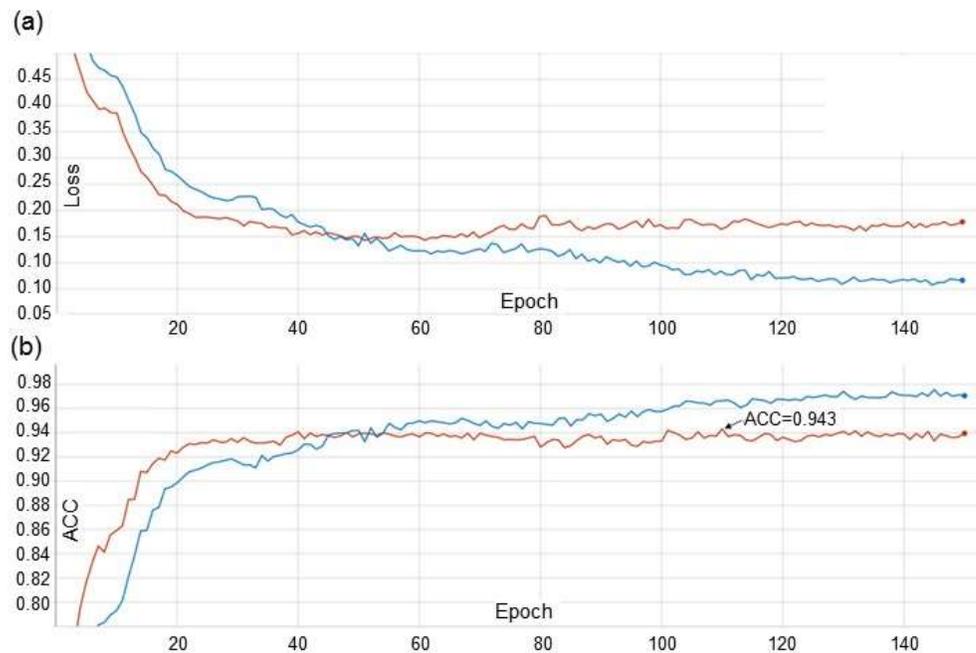


Figure 5. Changes in the loss function (a) and ACC accuracy (b) from the training epoch of the EfficientNet-B0_FDA model. Blue line – train dataset; red line – validation dataset.

From the data presented in Table 4, it can be concluded that the use of augmentations in the training process increases the accuracy on deferred sampling by 0.6% (EfficientNet-B0_augm vs EfficientNet-B0_baseline). The addition of the FDA style transfer to the augmentations increases the image recognition accuracy by another 0.3% (EfficientNet-B0_FDA vs EfficientNet-B0_augm). For the EfficientNet-B0_FDA model, we additionally evaluated the disease prediction accuracy for seedlings and adult plants separately. In the first case, the ACC value was 0.963, in the second one, 0.926.

Table 4. The value of the target metric of the model accuracy on the validation and test sample for different model training strategies. The strategies are described in the section "2.7. Training strategies".

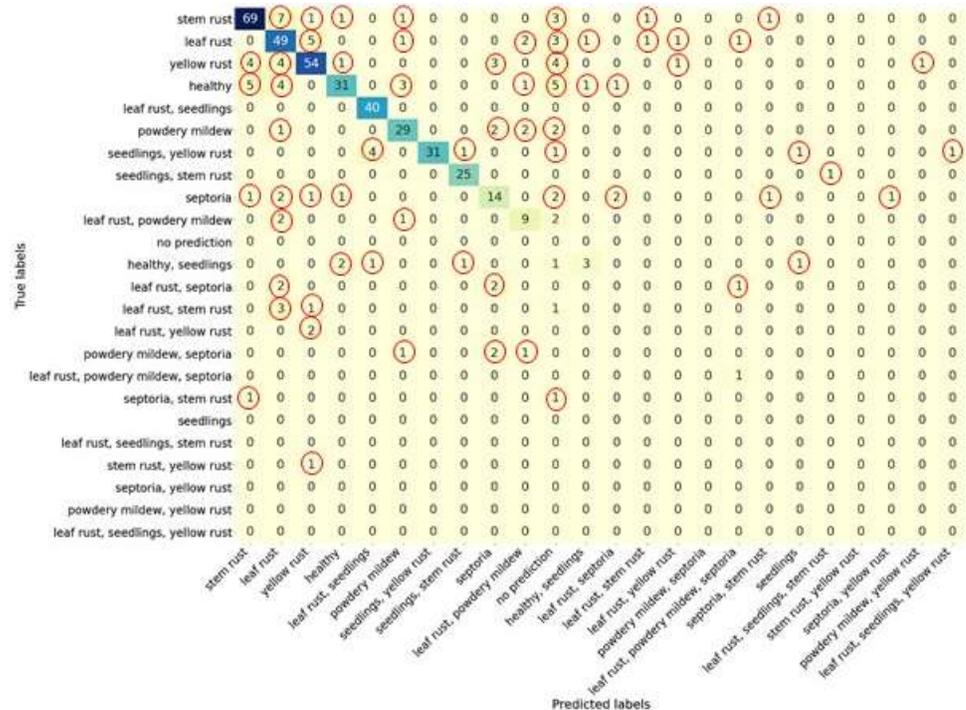
| Model name | Number of epochs | ACC valid | ACC test |
|--------------------------|------------------|-----------|----------|
| EfficientNet-B0_baseline | 18 | 0.938 | 0.933 |
| EfficientNet-B0_augm | 140 | 0.941 | 0.939 |
| EfficientNet-B0_FDA | 110 | 0.943 | 0.942 |

The recognition accuracy of individual labels for images of the test sample is presented for all three strategies for training the network in Table 5. It shows that the improvement in accuracy when changing the strategy for different labels does not always change systematically depending on the strategy. The increase in accuracy with the increasing complexity of the strategy (EfficientNet-B0_baseline -> EfficientNet-B0_augm -> EfficientNet-B0_FDA) is typical for disease-related labels, but it decreases slightly when identifying healthy plants and seedlings. The highest recognition accuracy among disease labels is achieved for 'septoria' (0.956), and the lowest one – for 'leaf rust' (0.890).

Table 5. Accuracy of determining various types of labels on the test sample for the three used neural strategies of neural network training.

| Label | EfficientNet-B0_baseline | EfficientNet-B0_augm | EfficientNet-B0_FDA |
|----------------|--------------------------|----------------------|---------------------|
| leaf rust | 0.881 | 0.881 | 0.890 |
| stem rust | 0.910 | 0.925 | 0.929 |
| yellow rust | 0.927 | 0.915 | 0.931 |
| powdery mildew | 0.940 | 0.952 | 0.954 |
| septoria | 0.927 | 0.952 | 0.956 |
| seedlings | 0.990 | 0.990 | 0.988 |
| healthy | 0.933 | 0.958 | 0.938 |

A confusion matrix for identifying labels in images for the EfficientNet-B0_FDA model, built on the basis of deferred sampling, is shown in Figure 6. The diagonal elements of the matrix are significantly larger than the off-diagonal ones, which characterizes the high accuracy of image classification by our method. An analysis of the matrix shows that the largest fraction of misclassifications among rust diseases falls on other rust diseases. Thus, for the label 'stem rust', 69 images (82%) are classified correctly, 4 images are classified as 'yellow rust', and 5 images as 'healthy'. For the label 'leaf rust' 49 images (77%) are classified correctly, 7 images are classified as 'stem rust', 4 images as 'yellow rust', and 4 images as 'healthy'. For the 'yellow rust' class, 54 images (75%) are classified correctly, 5 images are incorrectly classified as 'leaf rust', and 1 image as 'stem rust'. A slightly smaller, but noticeable part of the images from these classes appeared in the 'no prediction' class. Another type of error for these labels was placing them into the categories with multiple labels of rust diseases.

**Figure 6.** Confusion matrix for the test sample for the EfficientNet-B0_FDA model.

In the case of images of plants affected by several pathogens, in the overwhelming majority of cases, one of the pathogens is more often predicted correctly, but the symptoms of the second pathogen is recognized with less accuracy due to the low value of the corresponding output neuron. For example, for the class 'leaf rust, powdery mildew' 9 images are classified correctly, 2 images are classified as 'leaf rust', 2 images as 'powdery mildew', and 1 image as 'powdery mildew, septoria'. An example of such an image classified by the expert as 'leaf rust, powdery mildew' is shown in Figure 7A. In the center of the leaf, an area affected by leaf rust is clearly visible, and two spots affected by powdery mildew are visible on the left above and on the right below. However, the area on the right below is blurred due to the lack of focus of the camera. This is likely the reason why the label "powdery mildew" was not identified.

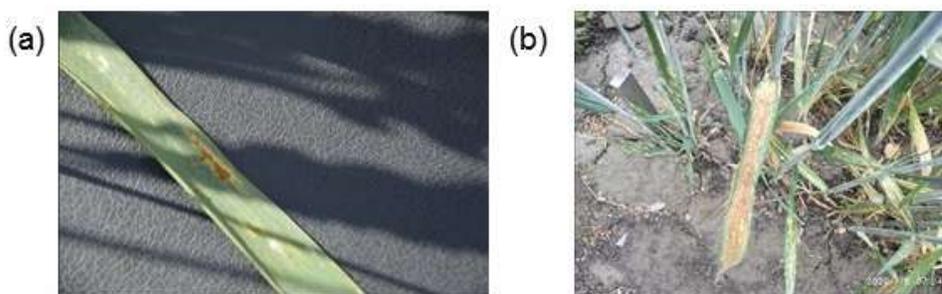


Figure 7. Examples of images marked with several disease labels at the same time, for which one of the diseases is not recognized. (a) Image 00c0f87c20f0fcff.jpg: labels 'leaf rust' (identified) and 'powdery mildew' (not identified). (b) Image 0dedf8d8db703948.jpg: labels 'leaf rust' (not identified) and 'septoria' (identified).

For the multiple label 'leaf rust, septoria' no images were predicted correctly, however, out of 5 such images, four were assigned to individual labels: 2 to 'leaf rust' and 2 to 'septoria'. An example of one such image is shown in Figure 7B. It shows the shoots strongly affected by septoria, especially the leaves in the foreground of the image. Against such a background, symptoms of leaf rust damage are barely noticeable (the plant is in the upper left corner of the frame near the peg).

Another example is the definition of joint labels 'leaf rust, powdery mildew, septoria' 1 image is classified correctly, 1 image as 'leaf rust', and 1 image as 'leaf rust, septoria' (Figure 6).

It is interesting to consider the case of classifying such a label as 'seedling'. It is noted that such a label was not separately presented in the labeling of any image. This label has always been found either together with the label of some disease or with the label of a healthy plant (see Table 3). Analysis of the confusion matrix shows that in most of all test images in which this label was present, the authors' algorithm predicted it. This gives an accuracy of 0.959 for the 'seedling' plant type. In several cases, the model assigned the label 'seedlings' to images that were not tagged with such a label. In two cases, the 'seedlings' label was predicted to be the only one (for the 'healthy seedlings' and 'seedlings, yellow rust' labels), but it should be noted that in these two cases the plants in the images were seedlings.

It should be noted that if no weight is determined for any of the labels greater than the specified threshold of 0.5, the model returns the label 'no prediction' (absence of classification). This label was assigned to 25 images (5%). The largest share of such classifications was obtained for the label 'healthy' (16%).

3.5. Bot in the Telegram messenger

The proposed recognition model was implemented for use as the bot @wheat_healthy_bot in the Telegram messenger https://t.me/wheat_healthy_bot. The Telegram application is available for all popular platforms, which allows using this service both in the field conditions via mobile devices and in the laboratory using a stationary computer. The user can send the bot an image of a plant and receive a prediction of its state (damage by a disease and/or developmental stage). This service allows users to identify plant diseases using mobile devices in field conditions. The user interface for interacting with the bot @wheat_healthy_bot is shown in Figure 8.



Figure 8. User interface for interacting with the bot @wheat_healthy_bot in Telegram messenger.

4. Discussion

In this paper, a method was proposed for identifying wheat diseases based on digital images obtained in the field conditions and the algorithm based on the EfficientNet-B0 architecture network. It should be noted that a sample of well-annotated images for training and testing is one of the important conditions for creating a successful method of disease recognition [30]. Unfortunately, the existing open databases of images of plant diseases contain few images of wheat. For example, there are 9 of them in the PDDDB database [34], but they are absent in the well-known PlantVillage dataset [33]. For the analysis of wheat diseases, in the work by Picon et al. [28], the sample included 8178 images (according to the imaging protocol, these data contained images of individual leaves, but not the whole plant). In the work by Lu et al. [29], a sample of Wheat Disease Database 2017 (WDD2017) was collected from 9230 images of not only leaves but also plants in field conditions. Arsenovich et al. [50] used a dataset that included information on 5596 images of wheat (the authors do not provide details). These three above-mentioned sets, however, are closed. Therefore, for the authors' work, based on several open sources and the authors' own data, a set of wheat images was formed both in the field and in the laboratory conditions. It is important to note that in the authors' sample, images from external sources were re-labeled by phytopathologists for six types of diseases, both separately and jointly. The WFD2020 data is freely available for downloading.

One of the features of the sample is a special class of images for seedlings. Special attention was paid to this since the timely diagnosis of the harmful complex in the early periods of the vegetational season is of particular importance. Under favorable weather conditions, diseases can develop rapidly and lead to significant losses of yield. Therefore,

to obtain great yields, it is important to monitor the lesion of flag and pre-flag leaves and diagnose the type of pathogen of the disease, which makes it possible to select effective means of protection [51]. As a result of the research, it appeared that the accuracy of disease prediction is higher for seedlings. However, this may also be related to the dataset peculiarities: the variety of shooting conditions for seedlings is less than when shooting adult plants.

In this work, the approach was applied to determine plant diseases based on the classification of images, and not on the highlighting of areas of plant organs affected by diseases, as was done in the works by Picon et al. [28] and Lu et al. [29]. At the same time, Picon et al. [28] used a network architecture based on ResNet50 to identify lesions of septoria, tan spot or rust (either leaf or yellow). The accuracy of disease identification was 0.97%. Lu et al. [29] tried several options of neural networks for deep learning and multiple instance learning to identify wheat lesions by the six most common diseases. The recognition accuracy ranged from 95 to 97%.

Our approach made it possible to significantly simplify the procedure for annotating images: no time-consuming work is required to select individual fragments of an image. Nevertheless, within the framework of this approach, it is possible to annotate both plant symptoms by single and multiple. At the same time, the accuracy of determining diseases using the developed method for the most optimal learning strategy (EfficientNet-B0_FDA) was 0.946, which is comparable to the results of other authors. This accuracy is comparable to the accuracy of methods for assessing plant diseases not only for wheat but also for other crops. For example, in the work [50], using a PlantVillage dataset and the authors' data, a two-stage network architecture was developed which showed an average accuracy of disease recognition of 0.936. In the work [40], the problem of simultaneous identification of 14 crop species and 26 diseases was also solved on the basis of the PlantVillage dataset. The network of the ResNet50 architecture was used, which achieved an accuracy of 99.24%.

Analysis of the matrix of errors showed that their main sources were incorrect classification of diseases, for example, rust ones, among themselves. Interestingly, the cross misclassification between rust diseases and the other two (septoria and powdery mildew) was found to be higher for septoria. This is explainable since the visual symptoms of these diseases can be very similar. In addition, for some images, the prediction of the pathogen type was not reliable enough to perform classification ('no prediction' result). However, in general, the proposed approach has demonstrated high accuracy in classifying both separate and joint types of plant diseases.

The possibility of using the technology of identification of crops by pathogens in the field is one of the necessary properties of the system for operational monitoring of diseases. In this direction, methods are being actively developed that integrate the results of prediction by the method of deep learning and the implementation of access to them via smartphones and tablets [29, 28, 39, 40]. This, however, requires additional effort to build and maintain mobile applications. Here, we took advantage of the possibility of a simple interface through the use of a Telegram messenger bot. Recently, this type of application has become popular and is used for crowdsourcing problems in social research [52], for predicting real estate prices [53], for determining the type of trees [54], etc. Its advantages are that there is no need to develop and maintain a smartphone graphical interface, at the same time it allows sending an image as a request to a server for further processing and displaying the results. At the same time, access to the service is possible wherever there is access to the Internet both from a mobile device and from a desktop PC.

5. Conclusions

The paper proposes a method for recognizing plant diseases on images obtained in field conditions based on the method of deep machine learning. Fungal diseases of wheat as leaf rust, stem rust, yellow rust, powdery mildew, septoria, and their combinations are

recognized. Additionally, the network determines whether the plant is a seedling. The algorithm is based on the EfficientNet-B0 neural network architecture. The training and testing of the network were conducted on the basis of a specially formed and labeled sample of 2414 images, which is freely available. Of the three examined learning strategies, the best accuracy was provided by the method using augmentation and transfer of image styles (the accuracy was 0.942). The interface to the method is implemented on the basis of the telegram bot, which provides access to the program via the Internet both from a mobile device and using a desktop PC.

Author Contributions: Conceptualization, M.A.G. and D.A.A.; methodology, M.A.G.; software, M.A.G.; validation, M.A.G., E.S.S. and D.A.A.; formal analysis, M.A.G.; investigation, M.A.G.; resources, D.A.A.; data curation, E.S.S., E.I.G., E.A.O., N.P.B.; writing—original draft preparation, M.A.G., E.S.S., E.I.G., E.A.O.; writing—review and editing, D.A.A.; visualization, M.A.G.; supervision, M.A.G.; project administration, M.A.G.; funding acquisition, M.A.G. All authors have read and agreed to the published version of the manuscript.

Funding: The dataset preparation, annotation and Telegram bot development was supported by RFBR grant 17-29-08028. Development of algorithms was funded by the Kurchatov Genome Center of the Institute of Cytology and Genetics of Siberian Branch of the Russian Academy of Sciences, agreement with the Ministry of Education and Science of the Russian Federation No. 075-15-2019-1662. The data analysis performed by the computational resources of the “Bioinformatics” Joint Computational Center supported by the budget project №0259-2021-0009.

Data Availability Statement: The WFD2020 dataset is available at <http://wfd.sysbio.ru/>. Telegram bot for disease recognition is accessible at https://t.me/wheat_healthy_bot.

Acknowledgments: Authors are grateful to Alexey Mukhin for help in telegram bot servise

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

References

1. Curtis, B.-C.; Rajaram, S.; Gomez Macpherson, H. Bread Wheat: Improvement and production, Food and Agriculture Organization of the United Nations (FAO), 2002.
2. Figueroa, M.; Hammond-Kosack, K. E.; Solomon, P. S. A review of wheat diseases—a field perspective. *Mol. Plant Pathol.* **2018**, *19*, 1523-1536.
3. Shamanin, V.; Pototskaya, I.; Shepelev, S.; Pozherukova, V.; Salina, E.; Skolotneva, E.; Hodson, D.; Hovmøller, M.; Patpour, M.; Morgounov, A. Stem rust in Western Siberia – race composition and effective resistance genes. *Vavilov J. Genet. Breeding* **2020**, *24*, 131-138.
4. Sanin, S.S. *Epiphytotics of cereal crops diseases: theory and practice. Izbrannyye trudy*. Voshod-A, Moscow, Russia, 2012; pp. 161-166. (in Russian)
5. Bhathal, J. S.; Loughman, R.; Speijers, J. Yield reduction in wheat in relation to leaf disease from yellow (tan) spot and septoria nodorum blotch. *Eur. J. Plant Pathol.* **2003**, *109*, 435-443.
6. Ficke, A.; Cowger, C.; Bergstrom, G.; Brodal, G. Understanding yield loss and pathogen biology to improve disease management: Septoria nodorum blotch—a case study in wheat. *Plant Dis.* **2018**, *102*, 696-707.
7. Simón, M. R.; Ayala, F. M.; Golik, S. I.; Terrile, I. I.; Cordo, C. A.; Perelló, A. E.; Moreno V.; Chidichimo, H. O. Integrated foliar disease management to prevent yield loss in Argentinian wheat production. *Agronomy J.* **2011**, *103*, 1441-1451.
8. Broers, L. H. M. Influence of development stage and host genotype on three components of partial resistance to leaf rust in spring wheat. *Euphytica* **1989**, *44*, 187-195.
9. Parker S.R.; Shaw M.W.; Poyle D.J. The reliability of visual estimates of disease severity on cereal leaves. *Plant Pathol.* **1995**, *44*, 856-864.
10. Bock, C. H.; Poole, G. H.; Parker, P. E.; Gottwald, T. R. Plant disease severity estimated visually, by digital photography and image analysis, and by hyperspectral imaging. *Crit. Rev. Plant Sci.* **2010**, *29*, 59-107.
11. Martinelli, F.; Scalenghe, R.; Davino, S.; Panno, S.; Scuderi, G.; Ruisi, P.; Villa, P.; Stroppiana, D.; Boschetti, M.; Goulart, L. R.; Davis, C. E.; Dandekar, A. M. Advanced methods of plant disease detection. A review. *Agron. Sustain. Dev.* **2015**, *35*, 1-25.
12. Farber, C.; Mahnke, M.; Sanchez, L.; Kurouski, D. Advanced spectroscopic techniques for plant disease diagnostics. A review. *Trends Anal. Chem.* **2019**, *118*, 43-49.
13. Mahlein, A. K. Plant disease detection by imaging sensors—parallels and specific demands for precision agriculture and plant phenotyping. *Plant Dis.* **2016**, *100*, 241-251.
14. Lindow, S. E.; Webb, R. R. Quantification of foliar plant disease symptoms by microcomputer-digitized video image analysis. *Phytopathol.* **1983**, *73*, 520-524.

15. Camargo, A.; Smith, J. S. An image-processing based algorithm to automatically identify plant disease visual symptoms. *Biosyst. Eng.* **2009**, *102*, 9-21.
16. Dammer, K. H.; Möller, B.; Rodemann, B.; Heppner, D. Detection of head blight (*Fusarium* spp.) in winter wheat by color and multispectral image analyses. *Crop Protection* **2011**, *30*, 420-428.
17. Ma, J.; Du, K.; Zhang, L.; Zheng, F.; Chu, J.; Sun, Z. A segmentation method for greenhouse vegetable foliar disease spots images using color information and region growing. *Comput. Electron. Agricult.* **2017**, *142*, 110-117.
18. Singh, A. K.; Ganapathysubramanian, B.; Sarkar, S.; Singh, A. Deep learning for plant stress phenotyping: trends and future perspectives. *Trends Plant Sci.* **2018**, *23*, 883-898.
19. Krizhevsky, A.; Sutskever, I.; Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Adv. Neur. Inf. Proc. Syst.* **2012**, *25*, 1097-1105.
20. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
21. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. *Proc. IEEE Conf. Comp. Vis. Patt. Rec.* **2016**, pp. 770-778.
22. Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
23. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. *Proc. IEEE Conf. Comp. Vis. Patt. Rec.* **2018**, 7132-7141.
24. Toda, Y.; Okura, F. How convolutional neural networks diagnose plant disease. *Plant Phenomics* **2019**, 2019, 9237136.
25. Saleem, M.H.; Potgieter, J.; Arif, K.M. Plant Disease Detection and Classification by Deep Learning. *Plants* **2019**, *8*, 468.
26. Hasan, R. I.; Yusuf, S. M.; Alzubaidi, L. Review of the state of the art of deep learning for plant diseases: A broad analysis and discussion. *Plants*, **2020**, *9*, 1302.
27. Barbedo, J.G.A. Plant disease identification from individual lesions and spots using deep learning. *Biosyst. Eng.* **2019**, *180*, 96-107.
28. Picon, A.; Alvarez-Gila, A.; Seitz, M.; Ortiz-Barredo, A.; Echazarra, J.; Johannes, A. Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. *Comp. Electron. Agricult.* **2019**, *161*, 280-290.
29. Lu, J.; Hu, J.; Zhao, G.; Mei, F.; Zhang, C. An in-field automatic wheat disease diagnosis system. *Comp. Electron. Agricult.* **2017**, *142*, 369-379.
30. Barbedo, J. G. A. Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification. *Comp. Electron. Agricult.* **2018**, *153*, 46-53.
31. Minervini, M.; Fischbach, A.; Scharr, H.; Tsafaris, S. A. Finely-grained annotated datasets for image-based plant phenotyping. *Patt. Rec. Lett.* **2016**, *81*, 80-89.
32. Lu, Y.; Young, S. A survey of public datasets for computer vision tasks in precision agriculture. *Comp. Electron. Agricult.* **2020**, *178*, 105760.
33. Hughes, D.; Salathé, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv* **2015**, arXiv:1511.08060.
34. Barbedo, J. G. A. A review on the main challenges in automatic plant disease identification based on visible range images. *Biosyst. Eng.* **2016**, *144*, 52-60.
35. Mohanty, S.P.; Hughes, D.P.; Salathé, M. Using deep learning for image-based plant disease detection. *Front. Plant Sci.* **2016**, *7*, 1419.
36. Pethybridge, S. J.; Nelson, S. C. Leaf doctor: a new portable application for quantifying plant disease severity. *Plant Dis.* **2015**, *99*, 1310-1316.
37. Johannes, A.; Picon, A.; Alvarez-Gila, A.; Echazarra, J.; Rodriguez-Vaamonde, S.; Navajas, A. D.; Ortiz-Barredo, A. Automatic plant disease diagnosis using mobile capture devices, applied on a wheat use case. *Comp. Electron. Agricult.* **2017**, *138*, 200-209.
38. Petrellis, N. A smart phone image processing application for plant disease diagnosis. In *2017 6th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, IEEE, 2017; pp. 1-4.
39. Uzhinskiy A., Ososkov G., Goncharov P., Nechaevskiy A. Multifunctional platform and mobile application for plant disease detection. In *Proc. 27th Symp. Nucl. Electr. Comp. CEUR Workshop Proc. 2507*; Korenkov V., Strizh T., Nechaevskiy A., Zaikina T., Eds.; Joint Institute for Nuclear Research, Dubna, Moscow Region, Russia, 2019; pp. 110-114.
40. Pallagani, V.; Khandelwal, V.; Chandra, B.; Udutalapally, V.; Das, D.; Mohanty, S. P. dcrop: A deep-learning based framework for accurate prediction of diseases of crops in smart agriculture. In *2019 IEEE International Symposium on Smart Electronic Systems (iSES)(Formerly iNiS)*, IEEE, 2019; pp. 29-33.
41. Szymański, P.; Kajdanowicz, T. A scikit-based Python environment for performing multi-label classification. *arXiv* **2017**, arXiv:1702.01460.
42. Tan, M.; Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks. *Int. Conf. Mach. Learning PMLR*, **2019**, pp. 6105-6114.
43. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. *Proc. IEEE Conf. Comp. Vis. Patt. Rec.* **2018**, pp. 4510-4520.
44. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, A.; Lerer, A. Automatic differentiation in pytorch. *31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, 2017
45. Müller, R.; Kornblith, S.; Hinton, G. When does label smoothing help? *arXiv* **2019**, arXiv:1906.02629.

-
46. Buslaev, A.; Iglorikov, V. I.; Khvedchenya, E.; Parinov, A.; Druzhinin, M.; Kalinin, A. A. Alumentations: fast and flexible image augmentations. *Information* **2020**, *11*, 125.
 47. DeVries, T.; Taylor, G.W. Improved regularization of convolutional neural networks with cutout. *arXiv* **2017**, arXiv:1708.04552.
 48. Yang, Y.; Soatto, S. Fda: Fourier domain adaptation for semantic segmentation. *Proc. IEEE/CVF Conf. Comp. Vis. Patt. Rec.* **2020**, pp. 4085-4095.
 49. Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579-2605.
 50. Arsenovic, M.; Karanovic, M.; Sladojevic, S.; Anderla, A.; Stefanovic, D. Solving current limitations of deep learning-based approaches for plant disease detection. *Symmetry* **2019**, *11*, 939.
 51. Chumakov, A. E. *Harmfulness of crop diseases*; Agropromizdat: Moscow, Russia, 1990; 127 p. (In Russian)
 52. Sangati, F.; Abramova, E.; Monti, J. DialettiBot: a Telegram Bot for Crowdsourcing Recordings of Italian Dialects. In *Proc. 5th Italian Conf. Comp. Ling. CLiC-it.* **2018**, 342-347.
 53. Salem, H.; Mazzara, M. ML-based Telegram bot for real estate price prediction. *J. Phys.: Conf. Ser.* **2020**, *1694*, 012010.
 54. Korotaeva, D.; Khlopotov, M.; Makarenko, A.; Chikshova, E.; Startseva, N.; Chemysheva, A. Botanicum: a telegram bot for tree classification. In *2018 22nd Conference of Open Innovations Association (FRUCT)*; IEEE 2018; pp. 88-93.