Agricultural plant hyperspectral imaging dataset

A.V. Gaidel [1,2,3], V.V. Podlipnov [1,2,3], N.A. Ivliev [1,2,3], R.A. Paringer [1,2,3],
P.A. Ishkin [4], S.V. Mashkov [4], R.V. Skidanov [1,2]
[1] IPSI RAS – Branch of the FRSC "Crystallography and Photonics" RAS;
443001, Samara, Russia, Molodogvardeiskaya St. 151;
[2] Samara National Research University,
443086, Samara, Russia, Moskovskoye shosse 34;
[3] Federal Research Center «Computer Science and Control» or Russian Academy of Sciences,
119333, Moscow, Russia, Vavilova 44;
[4] Samara State Agrarian University, 446442, Usty-Kinelyskiy, Russia, Uchebnaya 2

Abstract

Detailed automated analysis of crop images is critical to the development of smart agriculture and can significantly improve the quantity and quality of agricultural products. A hyperspectral camera potentially allows to extract more information about the observed object than a conventional one, so its use can help in solving problems that are difficult to solve with conventional methods. Often, predictive models that solve such problems require a large dataset for training. However, sufficiently large datasets of hyperspectral images of agricultural plants are not currently publicly available. Therefore, we present a new dataset of hyperspectral images of plants in this paper. This dataset can be accessed via URL https://pypi.org/project/HSI-Dataset-API/. It contains 385 hyperspectral images with a spatial resolution of 512 by 512 pixels and spectral resolution of 237 spectral bands. The images were captured in the summer of 2021 in Samara and Novocherkassk (Russia) using Offner based Imaging Hyperspectrometer of our own production. The article demonstrates the work of some basic approaches to the analysis of hyperspectral images using the dataset and states problems for further solving.

Keywords: hyperspectral imaging, image dataset, image processing, image segmentation, smart agriculture.

Citation: Gaidel AV, Podlipnov VV, Ivliev NA, Paringer RA, Ishkin PA, Mashkov SV, Skidanov RV. Agricultural plant hyperspectral imaging dataset. Computer Optics 2023; 47(3): 442-450. DOI: 10.18287/2412-6179-CO-1226.

Introduction

Smart agriculture is a new concept for the development of agriculture using modern IoT technologies, robotic technologies, computer vision, machine learning, etc. Today, smart agriculture is considered the most promising direction for the development of agriculture, which should significantly increase the production efficiency of food, industrial raw materials and other agricultural products [1]. Also, similar ideas for the development of agriculture might be referenced as "Agriculture 4.0" [2] or "Digital agriculture" [3].

Computer vision plays an important role in smart agriculture. The use of computer vision can automate the fight against weeds, watering plants, treating plants with fertilizers and herbicides, and more. To do this, object detection or image segmentation can be used in plant photographs captured with a conventional digital camera [4]. There are many publications devoted to the analysis of plant images using machine learning methods, including deep learning [5].

A hyperspectral camera can provide additional information on a digital image than a conventional one. Hyperspectral images store in each pixel information not only about three RGB bands, but about several hundred spectral bands reflecting the amount of energy in each of the spectral components of the visible electromagnetic field specter. In such images, computer vision algorithms can see even more information than the naked human eye can see. Due to this, approaches to the object detection and analysis in such images can work more efficiently than in conventional digital images [6]. This also applies to hyperspectral images of plants [7].

One can see that most of the predictive models used in the analysis of plant images are supervised and require a large learning sample for training. For natural comparison reasons, researchers use open plant imaging datasets such as PlantDoc dataset containing images of plants with various diseases [8] or BJFU100 dataset containing 100 species of ornamental plants in Beijing Forestry University campus captured by mobile device camera [9]. There is also DeepWeeds dataset with 17 thousand labelled images of eight weed species [10] and a lot of others.

However, all these datasets contain conventional RGB images. There are significantly fewer publicly available datasets containing hyperspectral images. This is because hyperspectral cameras are expensive and only available to qualified specialists. For example, there is a Specim hyperspectral camera, which can also be used for plant analysis [11], but its use in practice is quite difficult and requires specially trained personnel. It is also difficult to find a publicly open large dataset captured using this camera.

In [12], the open Hyperspectral Image Dataset is presented for detecting objects in a hyperspectral image with a size of 1024 by 768 pixels and 151 spectral bands. In total, it contains 60 images, the objects of which are a variety of outdoor objects, including some plants. The authors demonstrate the performance of some State-of-the-Art segmentation methods on images from this dataset, obtaining a maximum AUC-Borji Performance of 0.82. The objects in this article are not agricultural plants, so it cannot be used to solve smart agriculture problems.

Paper [13] presents an open dataset containing microscopy hyperspectral images of cholangiocarcinoma with a resolution of 1280×1024 pixels and 60 spectral bands. The authors show an example of region of interest segmentation using neural network and support vector machine approaches. As a result, they achieved an accuracy of 94 %. This dataset looks great, but again it has nothing to do with smart agriculture.

In [14], authors present their experiments aiming to differentiate between herbicide-resistant and herbicide-susceptible of weed kochia by the hyperspectral images. They collected a total of 152 hyperspectral images with a resolution of 640 by 2500 pixels and 240 spectral bands at the Montana State University Southern Agricultural Research Center. Using support vector machine with radial basis function kernel they achieved 80 % accuracy of classification. Unfortunately, it does not appear that the authors have published the dataset on which they conducted their experiments anywhere.

Thus, there is currently no open access to a sufficiently large dataset containing hyperspectral images of agricultural plants. Therefore, we created such a dataset and present it in this article. We describe the method of image registration, the characteristics of the dataset itself, and show an example of how basic approaches to the analysis of such images work. The presented dataset can be used in the future to train predictive models that solve the problems of smart agriculture, and to compare the performance of such models.

1. Image acquisition

Images were acquired using a self-produced Offner based Imaging Hyperspectrometer. The optical design of a compact hyperspectrometer based on the Offner scheme was described in [15 – 16]. A feature of this scheme is the need to manufacture a lattice on the convex surface of the mirror. At the same time, the quality and profile of such a lattice significantly affects the efficiency and performance of the final device [17 – 18]. Modeling and experimental studies have managed to achieve high indicators [19 – 20]. The calibration procedure for this device is described in [21]. The capturing was carried out in the summer of 2021 on agricultural land in Russia in the Samara region and in the Irkutsk region. Days with sunny weather and low clouds were predominantly chosen for shooting. The wind was moderate (2 – 4 meters per second). The objects of the capturing were such agricultural

crops as corn, oats, border areas of field plots, as well as borders of fields with areas of growing weeds. The most widespread among weeds is the common amaranth.

Fig. 1 shows the Offner based Imaging Hyperspectrometer appearance capturing agricultural field of the farm of E.P. Tsirulev located in the Samara region on a spot with coordinates 52.81 degrees latitude and 48.61 degrees longitude. As one can see, the shooting was carried out by scanning, by installing a hyperspectral camera on a special shooting tripod. On the left in the Fig. 1 there are plantings of corn, there are oats on the right, and there is a strip of amaranth between them.

Fig. 1. The appearance of a scanning hyperspectrometer on a swivel platform capturing plants
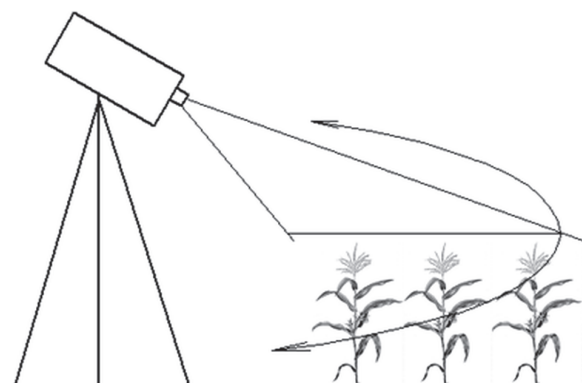

Fig. 2. Capturing scheme with a hyperspectral camera placed on a rotating tripod

For the survey, cultivated and irrigated areas were selected, predominantly with a uniform distribution of one crop over the survey area, as well as areas where several crops border. For shooting, the camera was mounted on a special rotating tripod equipped with an angular rotation drive with the ability to set the rotation speed in the range of 0.2 – 3 rpm. A hyperspectrometer with an Offner optical scheme was installed during shooting so that the slit diaphragm was perpendicular to the spatial scanning vector. The tripod is also equipped with a mechanical device that allows one to set different tilt angles of the camera relative to the subject.

Changing the installation height and the tilt angle makes it possible to capture hyperspectral images of different scales, and a certain depth of the scene is formed

on one image, where the same vegetation objects are simultaneously located near the camera (near the center of the scene) and at some distance from the camera (the edge of the image). It can also be noted that hyperspectral panoramic images have spatial distortions. The imaging quality can be evaluating using reference images in the manner described in [22].

For shooting objects, a lens with a fixed focal length MIR-1V 2.8 / 37 (Russia) was chosen, with the aperture set at a value of approximately 3.2. The choice of the specified lens is due to the sufficient field of view from such a short distance to the subjects. The equivalent focal length for a sensor with a crop factor of 2.7 is approximately 85mm, which corresponds to an angle of approximately 25 degrees. The frame rate in all scenes is fixed and corresponds to 15 fps, which ensures the consistency of spatial resolution in all obtained images. Due to the use of a reflective diffraction grating with glare in the Offner optical scheme, a sufficiently high illumination on the matrix sensor is provided. Fig. 3 shows the internal structure of the Offner based Imaging Hyperspectrometer.
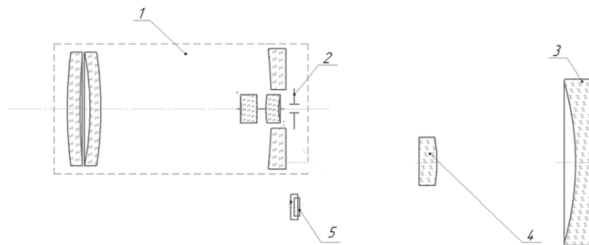

Fig. 3. Schematic representation of the hyperspectrometer optical layout: 1 – lens, 2 – slit diaphragm, 3 – spherical mirror, 4 – diffraction grating; 5 – visible range photodetector

Fig. 4 shows the original grayscale image projected onto the photosensitive matrix CMV4000. One can clearly see the bright scanning optical slit at the top of the image. There is also visible spectral decomposition of the image passed through the slit at the bottom of the image. Thus, the horizontal direction in this image is spatial, and the vertical direction is spectral. We reconstruct the final hypercube from the series of such images using our own approach presented in [23].

2. Dataset description

Fig. 5 shows an example of image reconstruction result from a hypercube, the capture of which is shown in Fig. 1. An extended horizontal artifact can be seen caused by the quality of the optical slit. Also, plants look blurry in some regions, since the recording is taken for a long time, and the plants move in the wind. Despite this, one can notice that the illumination is sufficient to obtain a clear, bright image. There is an X-Rite ColorChecker in the center of the image. It presents in many other images too, so one can compare color rendering.

The dataset itself can be accessed via URL https://pypi.org/project/HSI-Dataset-API/ and consists of 385 hyperspectral images with a spatial resolution of 512 by 512 pixels and spectral resolution of 237 spectral bands with wavelengths from 420 nm to 979 nm. These images were manually cropped from 59 different raw hyperspectral images of a larger size. All hyperspectral images are stored as 3D NumPy arrays in NumPy binary NPY format [24]. The first dimension is spectral and the other two dimensions are spatial.
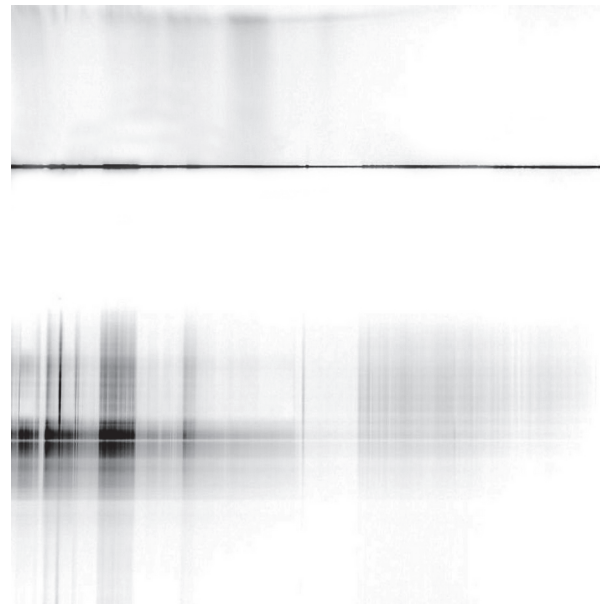

Fig. 4. The original image formed on the photosensitive matrix (Invert)

Fig. 5. Image reconstructed from a hypercube

The pixels in the images are labeled for 16 different classes: apple tree, beet, cabbage, carrot, corn, cucumber, eggplant, grass, milkweed, oats, pepper, potato, shchiritsa (amaranth), strawberry, soy, and tomato. The annotation was processed in the semi-automatic way using the most informative indexes [25]. The binary masks obtained from the informative indexes were manually adjusted to more closely match the boundaries of the objects. After that, the masks were divided according to the type of

plant. The fragments of the original full-size hyperspectral images that were the most meaningful in terms of the number of pixels corresponding to plants were selected to create a set of hypercubes. Binary masks corresponding to different plants for one cube were combined into a single mask, where each plant has its own value, which is unique within the entire set. The final label masks are stored in PNG format.

Fig. 6a shows the distribution of hyperspectral images in the dataset by types of plant presented. Fig. 6b represents the detailed distribution of the individual pixels in all images in total by classes. The number of pixels in the figure should be multiplied by 10^7, as marked above the axis. As one can see, the most common plant presented in the images is soy. The least frequent plants are apple tree, cabbage, eggplant, and shchiritsa (amaranth).
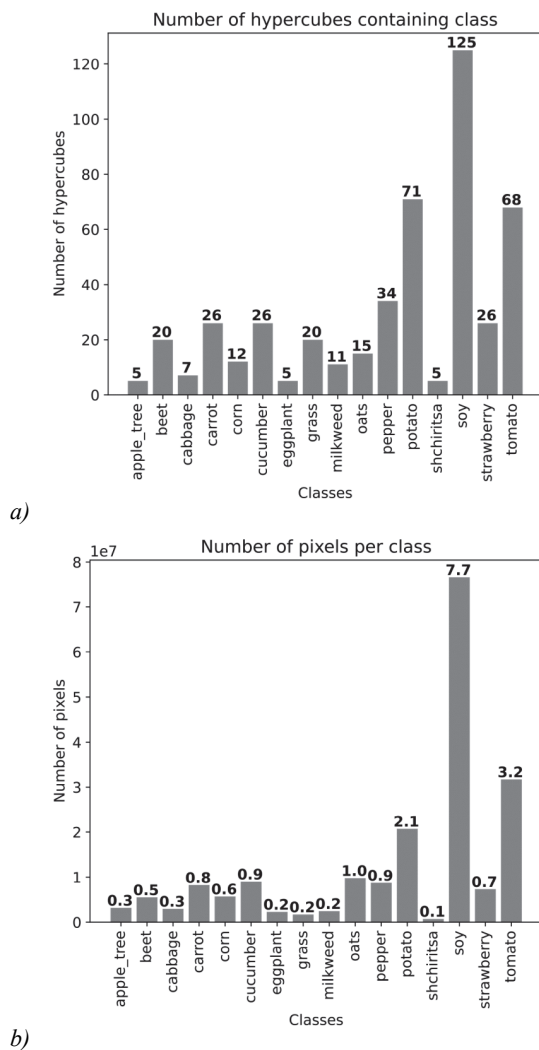

a)

b)

Fig. 6. Distribution of images (a) and pixels (b) by classes

Metadata is described in text YAML files. There is file meta.yml containing general information about classes and wavelength to spectral band mapping. Also, for each image there is a YAML file with the same name describing presented classes, image size, and some other less important information.

For convenient work with the dataset, a public API was developed using the Python language. This is a regular Python package that can be installed using standard Python tools, for example the pip package management system. The API source code is publicly available in the open GitHub repository. In addition, the repository includes a Jupyter notebook that shows an example of working with a dataset. The example shows how to prepare data and how to train the model using the Scikit-learn software package [26], which is widely used in solving data analysis problems.

Fig. 7 shows examples of images from the dataset. Fig. 7a represents the color-synthesized image obtained from the original 237-band image using average by three bands with wavelengths 476 nm, 550 nm, and 667 nm respectively. Again, one can see some vertical jitter in the image caused by the vibration of plants and shooting equipment in the wind. Fig. 7b represents the semi-automatic segmentation of the image by plants type. One can see two beds of different plants on the left and right. This image is auto contrasted: the actual grayscale values in the image is 0, 1, and 2. Different gray levels correspond to different plants. Zero value corresponds to the background.
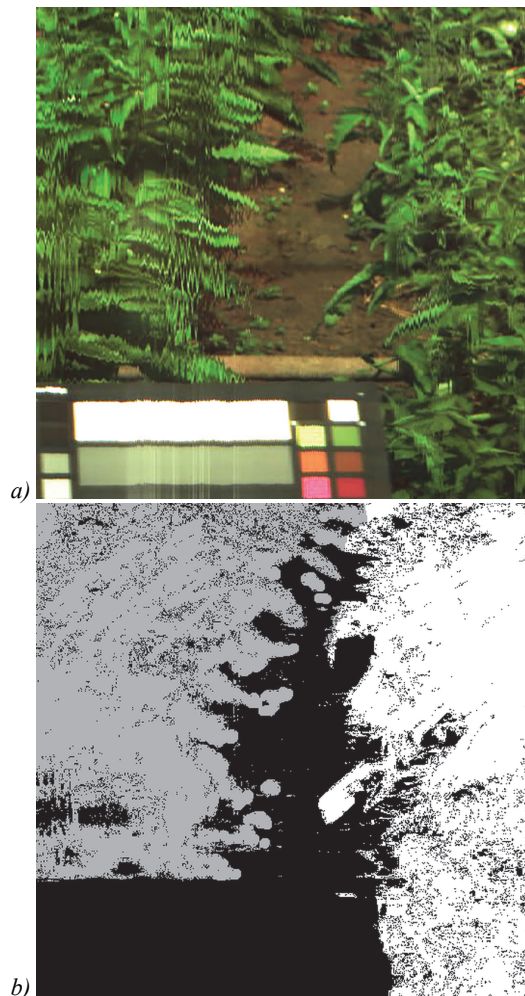

a)

b)

Fig. 7. Examples of images from the dataset: a color-synthesized hyperspectral image (a) and a manually segmented mask for it (b)

3. Processing hyperspectral images from the dataset

As an example of an applied problem that can be solved using the presented dataset, we have chosen the problem of hyperspectral image segmentation to distinguish some plant species from each other. The problem is to select a region of the image that corresponds to a certain type of plant. For simplicity, we consider this problem as a pixel-by-pixel classification of spectral vectors into a given number of classes. Thus, we do not care about the spatial relationships between pixels but take into account only the spectral characteristics of each particular pixel.

In order to eliminate the class imbalance that is observed in the Fig. 6 in advance, we took only the four rarest classes: apple tree, cabbage, eggplant, and shchiritsa (amaranth), as well as all the classes of plants that are found in the images, in which plants of these four classes are found. For the same reasons, we did not consider the background as a separate class, so the total number of different classes was 9. So, we took all the pixels in the selected images, corresponding to the above nine classes, and put them in a general sample $U \subseteq \mathbf{R}^L$, where $L = 9$ is a number of classes, and $\mathbf{R}$ is a set of real numbers. For each pixel x from the sample U, we know its real manually annotated class $\Phi(x): \mathbf{R}^L \to [1; L] \cap \mathbf{Z}$, where $\mathbf{Z}$ is a set of integers.

We can solve segmentation problem by constructing the operator $\tilde{\Phi}(x): \mathbf{R}^L \to [1; L] \cap \mathbf{Z}$, which relies only on knowledge of the learning sample $\tilde{U} \subseteq U$. This is a classic pattern recognition problem that can be solved using any known classifier. Also, we can employ various classification metrics to evaluate the classification quality using the test sample $\hat{U} \subseteq U \backslash \tilde{U}$.

Fig. 8 shows the class distribution in the sample we use for the experimental research the same way as Fig. 6 shows it for the whole dataset. Fig. 8a presents the number of hyperspectral images we included in the sample for each of 9 classes. Similarly, Fig. 8b shows the distribution of pixels in the selected sample by classes. So, Fig. 8 gives an idea of the materials for the problem being solved. As we can see, classes here look more balanced than in Fig. 6.

We employed Logistic Regression, Quadratic Discriminant Analysis, Random Forest and K Nearest Neighbors (KNN) as classifiers just for example as some popular universal classifiers. We developed a program in Python using Scikit-learn implementations [20] of these classifiers. We did not use any further data preprocessing, except for the one that was previously described in the article.

Logistic regression (LR) is a linear classification model that considers the l-th outcome possibility in a form of logistic function

$$f_l(x) = \frac{1}{1 + exp(-\omega_l^T x - c_l)},$$

where $\omega_l \in \mathbf{R}^L$ and $c_l \in \mathbf{R}$ are selectable parameters. The training algorithm varies these parameters trying to minimize the cost function [27]
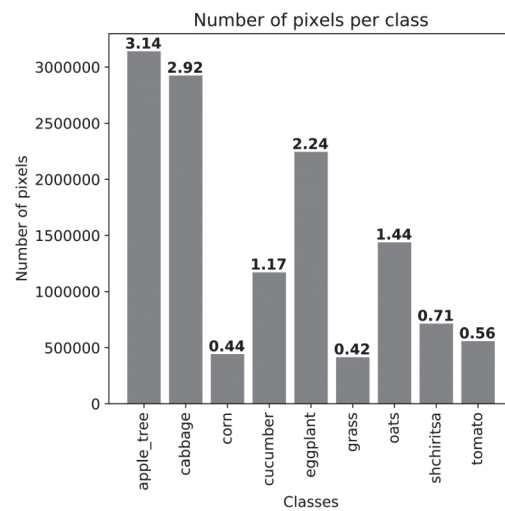
$$J_l^{(LR)} = \frac{1}{2}\omega_l^T \omega_l + \sum_{x \in U} ln\left(exp\left(-y_l(x)\left(\omega_l^T x + c\right)\right) + 1\right),$$

where $y_l(x)$ equals 1 if $\Phi(x) = l$ and (-1) otherwise. We used Broyden–Fletcher–Goldfarb–Shanno algorithm to solve this nonlinear optimization problem [28]. The final multinomial decision rule was based on the softmax function:

$$p_l^{(LR)}(x) = \frac{exp\left(\omega_l^T x + c_l\right)}{\sum_{l=1}^{L} exp\left(\omega_l^T x + c_l\right)}.$$
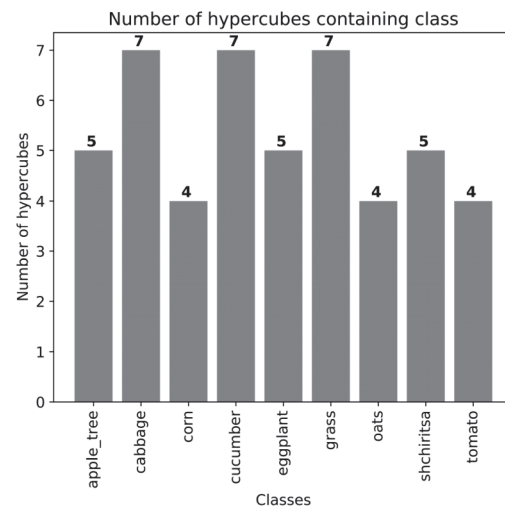
a)

b)

Fig. 8. Distribution of images (a) and pixels (b) by classes in the sample used for the experimental research

The classifier based on *Quadratic Discriminant Analysis* (QDA) constructs the quadratic decision surface with a help of the Bayesian rule [29]

$$p_l^{(QDA)}(x) = \frac{P\{\Phi(x) = l\} p(x \mid l)}{\sum_{k=1}^{L} P\{\Phi(x) = k\} p(x \mid k)},$$

where prior probabilities are inferred from the training data as

$$P\{\Phi(x)=l\} = \frac{\left|\tilde{U}_l\right|}{\left|\tilde{U}\right|}.$$

The $\left|\tilde{U}\right|$ means the number of elements in the finite set $\tilde{U}$ and $\tilde{U}_l = \{x \in \tilde{U} \mid \Phi(x)=l\}$ is a set of vectors of the l-th class in the training sample. The $p(x \mid l)$ is considered to be Gaussian:

$$p(x \mid l) = (2\pi)^{-L/2} \left|R_l\right|^{-1/2} exp\left(-\frac{1}{2}(x-\mu_l)^T R_l^{-1}(x-\mu_l)\right),$$

where μ_l is a mean value of the class l:

$$\mu_l = \frac{1}{\left|\tilde{U}_l\right|} \sum_{x \in \tilde{U}_l} x,$$

and R_l is an estimation of correlation matrix for the l-th class:

$$R_l(i,j) = \frac{1}{\left|\tilde{U}_l\right|} \sum_{x \in \tilde{U}_l} (x-\mu_l)(x-\mu_l)^T.$$

So, the predicted class should maximize the log posterior probability

$$J_l^{(QDA)} =$$
$$= -\frac{1}{2}ln\left|R_l\right| - \frac{1}{2}(x-\mu_l)^T R_l^{-1}(x-\mu_l) + lnP\{\Phi(x)=l\}.$$

Random Forest (RF) is an ensemble classifier consisting of randomized decision trees. We build each of 100 decision trees in the ensemble from a bootstrap sample with random replacements considering only randomly chosen $\sqrt{L}$ of L features [30]. We evaluate the quality of each split using Gini impurity measure:

$$J^{(RF)} = 1 - \sum_{l=1}^{L} \left(P\{\Phi(x)=l\}\right)^2.$$

So, the best split in the decision tree should minimize the weighted mean of Gini impurity among the nodes of the tree. The final decision rule is based on simple majority voting across all decision trees.

K Nearest Neighbors (KNN) classifier just assigns the input feature vector x to the class to which most of its K nearest neighbors from the training sample $\tilde{U}$ belong [31]. We considered the number of neighbors $K=5$ and used classic Euclidean distance to find the nearest of them:

$$\rho(x,y) = \sqrt{(x-y)^T(x-y)}.$$

To evaluate the quality of prediction model we used different scoring parameters: accuracy, F-macro, F-weighted, precision macro, precision weighted, recall macro, and recall weighted.

Classification *accuracy* is simply the proportion of correctly classified items from the test sample $\hat{U}$:

$$J_{accuracy} = \frac{\left|\left\{x \in \hat{U} \mid \tilde{\Phi}(x) = \Phi(x)\right\}\right|}{\left|\hat{U}\right|}.$$

Let us consider the precision and recall measures for each class l:

$$P_l = \frac{\left|\left\{x \in \hat{U} \mid \tilde{\Phi}(x) = \Phi(x) = l\right\}\right|}{\left|\left\{x \in \hat{U} \mid \tilde{\Phi}(x) = l\right\}\right|},$$

$$R_l = \frac{\left|\left\{x \in \hat{U} \mid \tilde{\Phi}(x) = \Phi(x) = l\right\}\right|}{\left|\left\{x \in \hat{U} \mid \Phi(x) = l\right\}\right|}.$$

As we can see, precision is the fraction of correctly classified objects among the objects classified in the class l, and recall is the fraction of correctly classified objects among the objects that really belong to the class l. In that case we can define precision macro, precision weighted, recall macro, recall weighted, F-macro and F-weighted as follows:

$$J_{\text{precision−macro}} = \frac{1}{L}\sum_{l=1}^{L} P_l,$$

$$J_{\text{precision−weighted}} = \frac{1}{\left|\hat{U}\right|}\sum_{l=1}^{L} \left|\left\{x \in \hat{U} \mid \Phi(x) = l\right\}\right| P_l,$$

$$J_{\text{recall−macro}} = \frac{1}{L}\sum_{l=1}^{L} R_l,$$

$$J_{\text{recall−weighted}} = \frac{1}{\left|\hat{U}\right|}\sum_{l=1}^{L} \left|\left\{x \in \hat{U} \mid \Phi(x) = l\right\}\right| R_l,$$

$$J_{\text{F−macro}} = \frac{1}{L}\sum_{l=1}^{L} \frac{P_l R_l}{P_l + R_l},$$

$$J_{\text{F−weighted}} = \frac{1}{\left|\hat{U}\right|}\sum_{l=1}^{L} \left|\left\{x \in \hat{U} \mid \Phi(x) = l\right\}\right| \frac{P_l R_l}{P_l + R_l}.$$

The most relative metrics are accuracy, F-macro and F-weighted. Table 1 shows the results of the classification quality evaluation. As one can see, simple KNN classifier outperforms other classifiers by all quality metrics. It correctly classifies 96 % of pixels in the images. Other classifiers are also doing well, especially the Random Forest. For the convenience of visual perception, the main results from the Tab. 1 are also shown in the Fig. 9 as a bar chart.

Tab. 1. Classification report

	LR	QDA	RF	KNN
Accuracy	0.83	0.79	0.95	0.96
F-macro	0.68	0.71	0.91	0.93
F-weighted	0.82	0.80	0.94	0.96
Precision macro	0.71	0.73	0.93	0.94
Precision weighted	0.82	0.84	0.95	0.96
Recall macro	0.67	0.75	0.89	0.92
Recall weighted	0.83	0.79	0.95	0.96

Fig. 10 shows an example of image segmentation result obtained using Random Forest pixel-wise classification. There is an original color-synthesized image in

Fig. 10*a*. Fig. 10*b* shows the semi-manual annotation of this image and Fig. 10c shows the result of automatic image segmentation by pixel-wise classification using Random Forest classifier. As one can see, the differences between Fig. 10*b* and Fig. 10*c* are not noticeable to the naked eye. That means in this case image segmentation works almost perfectly.
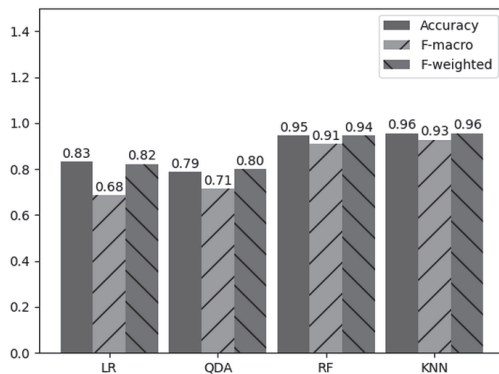

Fig. 9. Classification performance

Another approach to segmentation of images from the presented dataset using convolutional neural networks

can be found in [32]. Authors of that paper achieve the classification accuracy of 94 %.

Conclusion

We managed to create a new dataset of hyperspectral images of plants, suitable for researching methods of image processing of this kind. It can be useful for the further development of smart agriculture technologies. Experts in this field can use our dataset to develop and test computer vision systems that automatically analyze plant health and agricultural decision support systems.

Unfortunately, the shooting conditions hardly allow using this dataset as an ultimate tuning table, for which the characteristics of the image spectrum obtained once could be used in the future for other hyperspectral cameras. At least the possibilities of this kind have not been proven and require additional research. We hope to continue working on the creation of datasets of this kind and, finally, to obtain a reference calibration dataset, the use of which within a certain calibration procedure can allow the creation of unified hyperspectral image processing methods for any hyperspectrometers.
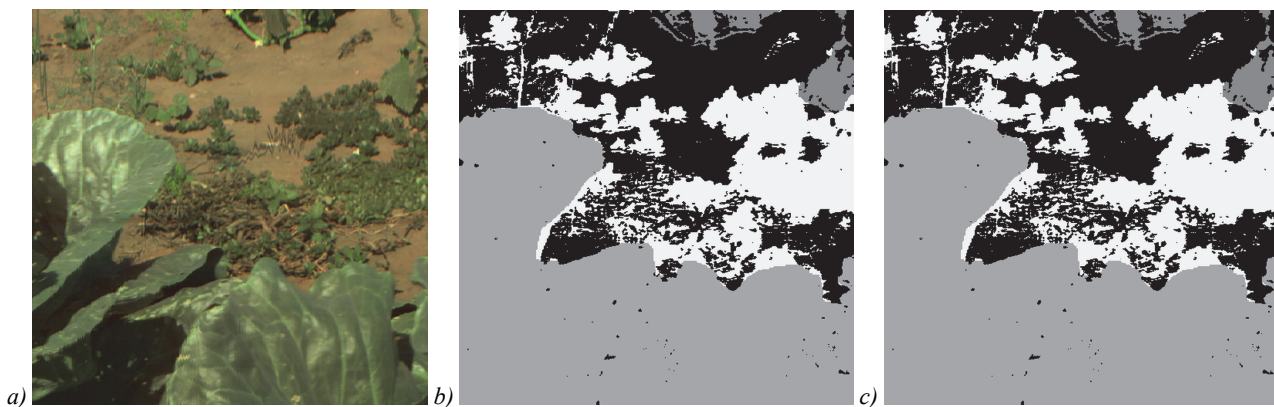

Fig. 10. Example of image segmentation using Random Forest: original color-synthesized image (a), semi-manual segmentation (b), automatic segmentation (c)

We presented an example of simple image segmentation approach based on pixel-wise classification on the reduced version of the dataset. After going through four popular universal classifiers, we achieved a classification accuracy of 96 % using the KNN classifier with Euclidian distance. This indicates to the fine quality of the prepared dataset and the fundamental possibility of pattern recognition with its help. Of course, it would be interesting to conduct a larger-scale study on the possibility of segmentation of images from this dataset on the full set and considering the spatial relationships between pixels.

We have so far produced several hyperspectrometers capable of capturing images like those presented in this dataset [33]. We are interested in the opportunities of using these devices for solving applied problems, including for smart agriculture and not only this. We have a service that allows one to collect hyperspectral data from unmanned aerial vehicles and even from satellites. We would be glad if potential customers who are experiencing the need to solve such problems would contact us.

References

[1] Yang X, Shu L, Chen J, Ferrag MA, Wu J, Nurellari E, Huang K. A survey on smart agriculture: Development modes, technologies, and security and privacy challenges. IEEE/CAA J Autom Sin 2021; 8(2): 273-302. DOI: 10.1109/JAS.2020.1003536.

[2] Rose DC, Chilvers J. Agriculture 4.0: Broadening responsible innovation in an era of smart farming. Front Sustain Food Syst 2018; 2: 87. DOI: 10.3389/fsufs.2018.00087.

[3] Bertoglio R, Corbo C, Renga FM, Matteucci M. The digital agricultural revolution: A bibliometric analysis literature review. IEEE Access 2021; 9: 134762-134782. DOI: 10.1109/ACCESS.2021.3115258.

[4] Patrício DI, Rieder R. Computer vision and artificial intelligence in precision agriculture for grain crops: A

systematic review. Comput Electron Agric 2018; 153: 69-81. DOI: 10.1016/j.compag.2018.08.001.

[5] Hasan RI, Yusuf SM, Alzubaidi L. Review of the state of the art of deep learning for plant diseases: A broad analysis and discussion. Plants 2020; 9(10): 1302. DOI: 10.3390/plants9101302.

[6] Manolakis D, Shaw G. Detection algorithms for hyperspectral imaging applications. IEEE Signal Process Mag 2002; 19(1): 29-43. DOI: 10.1109/79.974724.

[7] Mishra P, Asaari MSM, Herrero-Langreo A, Lohumi S, Diezma B, Scheunders P. Close range hyperspectral imaging of plants: A review. Biosyst Eng 2017; 164: 49-67. DOI: 10.1016/j.biosystemseng.2017.09.009.

[8] Singh D, Jain N, Jain P, Kayal P, Kumawat S, Batra N. PlantDoc: A dataset for visual plant disease detection. CoDS COMAD 2020: Proc 7th ACM IKDD CoDS and 25th COMAD 2020: 249-253. DOI: 10.1145/3371158.3371196.

[9] Sun Y, Liu Y, Wang G, Zhang H. Deep Learning for plant identification in natural environment. Comput Intell Neurosci 2017; 2017: 7361042. DOI: 10.1155/2017/7361042.

[10] Olsen A, Konovalov DA, Philippa B, Ridd P, Wood JC, Johns J, Banks W, Girgenti B, Kenny O, Whinney J, Calvert B, Azghadi MR, White RD. DeepWeeds: A multiclass weed species image dataset for deep learning. Sci Rep 2019; 9: 2058. DOI: 10.1038/s41598-018-38343-3.

[11] Behmann J, Acebron K, Emin D, Bennertz S, Matsubara S, Thomas S, Bohnenkamp D, Kuska MT, Jussila J, Salo H, Mahlein A-K, Rascher U. Specim IQ: Evaluation of a new, miniaturized handheld hyperspectral camera and its application for plant phenotyping and disease detection. Sensors 2018; 18: 441. DOI: 10.3390/s18020441.

[12] Imamoglu N, Oish Y, Zhang X, Ding G, Fang Y, Kouyama T, Nakamura R. Hyperspectral image dataset for benchmarking on salient object detection. 2018 Tenth Int Conf on Quality of Multimedia Experience (QoMEX) 2018: 1-3. DOI: 10.1109/QoMEX.2018.8463428.

[13] Zhang Q, Li Q, Yu G, Sun L, Zhou M, Chu J. A multidimensional choledoch database and benchmarks for cholangiocarcinoma diagnosis. IEEE Access 2019; 7: 149414-149421. DOI: 10.1109/ACCESS.2019.2947470.

[14] Nugent PW, Shaw JA, Jha P, Scherrer B, Donelick A, Kumar V. Discrimination of herbicide-resistant kochia with hyperspectral imaging. J Appl Remote Sens 2018; 12(1): 016037. DOI: 10.1117/1.JRS.12.016037.

[15] Kazanskiy NL, Kharitonov SI, Karsakov SI, Khonina SN. Modeling action of a hyperspectrometer based on the Offner scheme within geometric optics. Computer Optics 2014; 38(2): 271-280. DOI: 10.18287/0134-2452-2014-38-2-271-280.

[16] Kazanskiy NL, Kharitonov SI, Doskolovich LL, Pavelyev AV. Modeling the performance of a spaceborne hy-perspectrometer based on the Offner scheme. Computer Optics 2015; 39(1): 70-76. DOI: 10.18287/0134-2452-2015-39-1-70-76.

[17] Karpeev SV, Khonina SN, Kharitonov SI. Study of the diffraction grating on a convex surface as a dispersive element. Computer Optics 2015; 39(2): 211-217. DOI: 10.18287/0134-2452-2015-39-2-211-217.

[18] Kazanskiy NL. Modeling diffractive optics elements and devices. Proc SPIE 2018; 10774: 107740O. DOI: 10.1117/12.2319264.

[19] Kazanskiy NL, Morozov AA, Nikonorov AV, Petrov MV, Podlipnov VV, Skidanov RV, Fursov VA. Experimental study of optical characteristics of a satellite-based Offner hyperspectrometer. Proc SPIE 2018; 10774: 1077411. DOI: 10.1117/12.2318853.

[20] Rastorguev AA, Kharitonov SI, Kazanskiy NL. Numerical simulation of the performance of a spaceborne Offner imaging hyperspectrometer in the wave optics approximation. Computer Optics 2022; 46(1): 56-64. DOI: 10.18287/2412-6179-CO-1034.

[21] Podlipnov VV, Skidanov RV. Calibration of an imaging hyperspectrometer. Computer Optics 2017; 41(6): 869-874. DOI: 10.18287/2412-6179-2017-41-6-869-874.

[22] NikonorovA, Petrov M, Yakimov P, Blank V, Karpeev S, Skidanov R, Kazanskiy N. Evaluating imaging quality of the offner hyperspectrometer. 9th IAPR Workshop on Pattern Recogniton in Remote Sensing (PRRS) 2016: 1-6. DOI: 10.1109/PRRS.2016.7867020.

[23] Karpeev SV, Khonina SN, Murdagulov AR, Petrov MV. Alignment and study of prototypes of the Offner hyperspectrometer. Vestnik of Samara University. Aerospace and Mechanical Engineering 2016; 15(1): 197-206. DOI: 10.18287/2412-7329-2016-15-1-197-206.

[24] Walt S, Colbert SC, Varoquaux G. The NumPy array: A structure for efficient numerical computation. Comput Sci Eng 2011; 13(2): 22-30. DOI: 10.1109/MCSE.2011.37.

[25] Paringer RA, Mukhin AV, Kupriyanov AV. Formation of an informative index for recognizing specified objects in hyperspectral data. Computer Optics 2021; 45(6): 873-878. DOI: 10.18287/2412-6179-CO-930.

[26] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay É. Scikit-learn: Machine learning in Python. J Mach Learn Res 2011; 12(85): 2825-2830.

[27] Tolles J, Meurer WJ. Logistic regression: Relating patient characteristics to outcomes. JAMA 2016; 316(5): 533-534. DOI: 10.1001/jama.2016.7653.

[28] Fletcher R. Practical methods of optimization. New York: John Wiley & Sons; 1987. ISBN: 978-0-471-91547-8.

[29] Baudat G, Anouar F. Generalized discriminant analysis using a kernel approach. Neural Comput 2000; 12(10): 2385-2404. DOI: 10.1162/089976600300014980.

[30] Breiman L. Random forests. Mach Learn 2001; 45: 5-32. DOI: 10.1023/A:1010933404324.

[31] Fix E, Hodges JL. Discriminatory analysis, nonparametric discrimination: Consistency properties. Technical Report 4, USAF School of Aviation Medicine, Randolph Field; 1951.

[32] Firsov NA, Podlipnov VV, Ivliev NA, Nikolaev PP, Mashkov SV, Ishkin PA, Skidanov RV, Nikonorov AV. Neural network-aided classification of hyperspectral vegetation images with a training sample generated using an adaptive vegetation index. Computer Optics 2021; 45(6): 887-896. DOI: 10.18287/2412-6179-CO-1038.

[33] Kazanskiy N, Ivliev N, Podlipnov V, Skidanov R. An airborne Offner imaging hyperspectrometer with radially-fastened primary elements. Sensors 2020; 20(12): 3411. DOI: 10.3390/s20123411.

Authors' information

Andrey Viktorovich Gaidel (b. 1989). Graduated from Samara State Aerospace University in 2012, majoring in Applied Mathematics and Informatics. He received his Candidate of Science degree in Physics and Math in 2015 from the SSAU. Currently he is a teaching assistant of the Technical Cybernetics sub-department and Engineer at SSAU's laboratory SRL-35 of Samara State Aerospace University, also working as an intern researcher of the Image Processing Systems Institute of the Russian Academy of Sciences – Branch of the FSRC "Crystallography and Photonics" RAS, Samara, Russia. His research interests currently focus on computer image processing, pattern recognition, data mining and theory of computation. E-mail: *andrey.gaidel@gmail.com* .

Vladimir Vladimirovich Podlipnov, (b. 1987), an engineer at Samara National Research University's Lab-35, an engineer of the laboratory of Micro- and Nanotechnology of the Image Processing Systems Institute of the RAS – Branch of the FSRC "Crystallography and Photonics" of the Russian Academy of Sciences. His research interests: mathematical modeling, electron-beam lithography, optimization of etching procedures in microelectronics, diffractive optics and techniques for surface processing and inspection. E-mail: *podlipnovvv@ya.ru* .

Nikolay Aleksandrovich Ivliev, (b. 1987), graduated from Samara State Aerospace University in 2010 (presently, Samara National Research University, short – Samara University), majoring in Design and Technology of Radioelectronic Equipment. Candidate of Engineering Sciences (2015). Currently he works as the researcher at the Image Processing Systems Institute of RAS – Branch of the FSRC "Crystallography and Photonics" RAS, assistant at Technical Cybernetics sub-department of Samara University. Research interests: surface physics, micro- and nanotechnology. E-mail: *ivlievn@gmail.com* .

Rustam Alexandrovich Paringer, (b. 1990), received Master's degree in Applied Mathematics and Informatics from Samara State Aerospace University (2013). He received his PhD in 2017. Associate professor of Technical Cybernetics department of Samara National Research University and researcher of IPSI RAS – Branch of the FSRC "Crystallography and Photonics". Research interests: data mining, machine learning and artificial neural networks. E-mail: *rusparinger@ssau.ru* .

Sergey Vladimirovich Mashkov, (b. 1983) he received his PhD in Economics in 2009. Associate Professor, Rector of the Samara State Agrarian University, Head of the Department of Electrification and Automation of the Agro-Industrial Complex. Research interests: digital and electrical technologies in agriculture, mechanization and automation of agriculture, economic methods for assessing agricultural machinery in crop production technology. E-mail: *mash_ser@mail.ru* .

Pavel Aleksandrovich Ishkin, (b. 1982) he received his PhD in 2008. Head of the research laboratory "Agrocybernetics", associate professor of the department "Electrification and automation of agro-industrial complex" Samara State Agrarian University. Research interests: energy efficient tillage technologies, digital technologies in agriculture, precision farming. E-mail: *ishkin_pa@mail.ru* .

Roman Vasilyevich Skidanov, (b. 1973). Graduated with honors (1990) from Samara State University (SSU)), majoring in Physics. He received his Doctor in Physics & Maths (2007) degrees from Samara State University. He is the head of Micro- and Nanotechnologies laboratory of the Image Processing Systems Institute of RAS – Branch of the FSRC "Crystallography and Photonics" of the Russian Academy of Sciences, holding a part-time position of professor at SSU's Technical Cybernetics sub-department. He is co-author of 160 scientific papers, 7 monographs. His current research interests include diffractive optics, mathematical modeling, image processing, and nanophotonics. E-mail: *romans@smr.ru* .