ACTA UNIVERSITATIS SAPIENTIAE
AGRICULTURE AND ENVIRONMENT, **10** (2018) 20–30

DOI: 10.2478/ausae-2018-0002

Automated evaluation of agricultural damage using UAV survey

Daniel STOJCSICS, Zsolt DOMOZI, András MOLNÁR

Institute of Applied Informatics, John von Neumann Faculty of Informatics,
Óbuda University, Hungary,
e-mail: stojcsics.daniel@nik.uni-obuda.hu; mr.zsolt.domozi@ieee.org;
molnar.andras@nik.uni-obuda.hu

Manuscript received January 20, 2018; revised February 21, 2018; accepted February 28, 2018

Abstract. In the last decade, the rate of the industrial usage of fixed-wing and blended wing aircraft has increased. A 1–2-km^2 area can be surveyed by such a drone within 30 to 60 minutes, without any special infrastructure, and this can be repeated at any time. This provides an opportunity to conduct automatized surveys and time series data testing, which can be used as a basis to decide specific processes. The state and the development of the plants can be monitored as well as the spread of pests and the efficiency of the procedures that protect against them. During the surveys, thousands of images are taken of the area, which can be converted to a georeferenced large-sized map within 20 to 40 hours, including post-production and a resolution varying from 0.01 to 0.1 cm/pixel. The paper provides a solution to the industrial post-production of these high-quantity data, in which a deep learning-based automated process using Matlab is presented, including a comparison of the results to the GIS data.

Keywords: deep learning, convolutional neural network, UAV, Matlab, survey, precision agriculture, game damage, time series

1. Introduction

In the construction industry and in agriculture, the demand for drone-based aerial mapping supplementing classical surveying technology and partly replacing it has increased in recent years. The drone family of larger companies, such as Parrot / Sensefly eBee or the Trimble UX5 fixed-wing drone aircraft, provide an easily available alternative to man-led, still expensively operated aerial mapping using large aircrafts.

Of course, the prepared systems also have an open-source alternative, such as the ArduPilot drone controller, which we, too, have been using for years. The technological basis of every solution is identical. On-board intelligence is provided by a microcontroller-led on-board unit, which is supplied with the sensors necessary for flying such as GPS and a barometric air velocity and height measurement device. Using a modem, the telemetry connects to the land server unit, to the GCS, which is generally the software running in a Windows or Linux environment. The most popular software package for ArduPilot is the Mission Planner software package. The robot controller is built into an aircraft selected specifically for the work.

Flying wing aircraft with 1–2-m wingspan proved to be particularly suitable for conducting such type of tasks as they are practically composed of only one swept-back wing and of a tiny body.

The operation, in other words, the take-off and landing is conducted from the side of the area to be surveyed, mostly from a road of the width of only one car, without any solid surface. As the basic component of these aircrafts is EPO foam (which, if necessary, is strengthened with a composite material), it can easily endure harder landings with damage. Physical damage can generally be repaired easily and fast between two flights. If a more serious damage occurs, it can be remedied easily by replacing the part with a spare one.

Generally, the sensor managing the remote control is placed inside the body of the aircraft in an area protected from weather. When taking RGB images, this is possibly a converted compact camera with a 16 to 28 megapixel resolution. When conducting multispectral measurements, this can also be a NIR or an imager operating within the main area, such as the FLIR DUO R or the TETRACAM ADC Micro, used in such cases by us.

Depending on the utilized imager and the field resolution to be achieved, flight altitude is chosen. In the case of an RGB orthophoto output agricultural survey, we used a Canon S100 camera whose resolution is 12 megapixel, its sensor size is 1/1.7” and 2.3 fps, and it is capable of capturing a full-resolution still image. We provided the camera with CHDK firmware, to which we created a unique script. As a result, when reaching the desired flight altitude, the machine selects a short shutter speed and takes intensely bright, sharp images until landing. Our chosen flight altitude for 3 cm/pixel field resolution was 150 m.

In this way, there will be a 60% to 70% longitudinal and latitudinal overlap, which is excellent for photogrammetry. During the flight, GCPs are also surveyed, which is done by hand-held GPS results within 5 to 10 m by applying DGPS results in < 1 m accuracy. It is important to mention that when surveyed locally, the accuracy of the images georeferenced using a regular GPS device turns out to be around 1 m too, only that they have a 5- to 10-m offset error compared to reality. This means that the area data measured at the output become practically identical to those having georeferenced by DGPS.

The output contains thousands of images, which can be converted into a georeferenced orthophoto within 20 to 40 hours using a PC with high-end GPGPU.

Generally, 30 to 60 minutes are needed to survey a 1- to 2-km^2 area, including the survey of GCPs, and in this way it can be repeated weekly. Based on the time series analyses, the exact time and reason for the development of certain diseases during the growth of the plant can be determined, just like the size of the damage and its process.

During our examination, we surveyed a given agricultural area several times. We detected the exact level and location of the damage, which were used as bases for protection measures. Our analysis was partly manual and partly automatized. In the following sections, we will introduce a deep-learning-based new examination technology, which can be used to accelerate processing and to make it easier at the same time.

2. Materials and methods

A. Deep learning

An artificial neural network is a type of mathematical model which is based on biological observations. Its main application area is machine learning, whose aim is to set up a system for these networks and to become capable of learning. Considering its structure, a neural network is a graph-based model and its basic elements are neurons which communicate with one another.

The model of the neuron used today is based on Hebbian learning, which describes that learning is not a passive process, but it is the composition of the physiological processes occurring in the biological network.

An artificial neuron is the elemental computing unit of the neural network; it is a much more simplified model of the biological neuron. Generally, we build layers of artificial neurons, and in this way neural network calculations can be described by matrix operations. The type of the layer is determined by what operation it conducts, while, on the other hand, network architecture is determined by the type, the number, and the order of the applied subsequent layers [1].

The layers managing typical tasks can also be the following:

1. Fully connected: creates the combination of the input and the stored weight matrix (1):
$$H = XW + b, \tag{1}$$
where x stands for input matrix, w for weight matrix, and b is an optional weight vector for translation.
2. Recurrent: it recovers its own output per input (2):
$$H_t = XW_x + b_x + H_{t-1}W_h + b_h, \tag{2}$$

where X is input matrix, W_x and b_x are the weights belonging to it, H_{t-1} is the previous output, and W_h and b_h are the weights belonging to it.

3. Convolutional: there is no full connection – the neurons only receive a part of the input. It conducts cross-correlation using the weight matrix of the neurons on the input matrix (3):

$$H = X * W + b, \tag{3}$$

where * marks the cross-correlation.

Neural networks are composed of three logical units:

1. Input layer: it forwards input data towards other parts of the network. The number of neurons is determined by the size and shape of the input data.
2. Hidden layers: also called inner layers whose task is to transform input information.
3. Output layer: the output function and the number of output neurons are determined by the nature of the given problem.

By increasing computational capacity, more complex architectures appear, which contain more and more layers and junctions. By increasing the depth of neural networks, the ability for abstraction increases, and the layers on different depth levels become capable of managing more and more complex tasks. Deep learning is essentially a term for a highly complex neural network which is capable of providing the expected output to complex inputs [2], [3].

CNNs can be utilized well for analysing top-down photos. Top-down photos can be satellite photos or aerial photos too. The former ones are only available for research purposes in a resolution which does not make it possible to draw conclusions of any kind regarding the given area [4]. One characteristic of aerial photos is that they are usually taken for a well-defined purpose at identical heights, using identical camera settings. The purpose can be to analyse a specific agricultural land, to track down areas [5] contaminated by weeds, or to search for any other kind of changes.

The search for lesions is usually conducted involving persons having professional qualifications and experience, and it demands time-consuming manual work besides significant costs. On the contrary, CNN networks provide a new opportunity for analysing images, as it can be managed using CNNs too.

For CNN-based analyses, unmanned aerial vehicles (UAV) are ideal to produce the necessary images because, besides low costs, they are capable of taking photographs of a relatively large area, and the quality of the produced images is ideal for further utilization, as shown in [6], [7], [8], and [9].

In order to analyse the image material collected in this manner, search samples must be defined. These samples can originate from the actual or from a previous survey – it is more important that a large number of training samples be available.

For CNNs, training samples can be quickly managed, and from that point on CNNs can be used to analyse survey images. Naturally, in this case too, the assistance of the expert of the area is many times indispensable; however, this is necessary for a much shorter amount of time, only until defining training samples.

Considering its field of use, it is an exceptionally diverse area where CNN-based image analysis can be used with a result and where a conclusion can be drawn regarding the actual area based on any aspect. In this way, it is even suitable for searching for fungal infections in agricultural lands or even before starting cultivation to classify seeds [10], [11]. In cultures representing an exceptionally high value, it is capable of revealing the deficiencies of the vegetation and even of quantifying it. In this way, for example, a significant profit increase can be achieved in tobacco production areas [12].

By using CNNs, a search can be conducted in areas under cultivation, and even animals present in their environment can be searched for. Thus, the approximation of the population growth of pests and their classification become possible [13].

B. Game damage

The exact establishing of the damage in large-sized agricultural areas, even of several km^2, is an extremely difficult task from the side of the land. The state of the land is mostly approximated based on local sampling by experts, but this is not at all an accurate process. In case of damage, the larger the interest of the owner, the bigger the interest of the insurance company is to establish a smaller value. Accurate data can be provided by aerial photography and not only momentary but temporally changeable rates can also be monitored.

C. Gopher nest detection on wheat field

In the summer of 2015, gophers caused damage to a field of wheat which has been surveyed several times. They were able to completely destroy a 1 to 5 m^2 area of the young plant around the nest. Using the classical GIS method, we calculated and localized the number of nests and their position. The reference manual measurement was made in ArcGIS, as seen in ***Fig. 1***.

Using the exact position of gopher nests, rodenticide is possible to be applied since only local – nest-wise – application is allowed (Par. 2(1) of the Ministry of Agriculture and Rural Development 43/2010. (23 April) on plant protection orders protective measures taken against gophers by farmers and other land users).

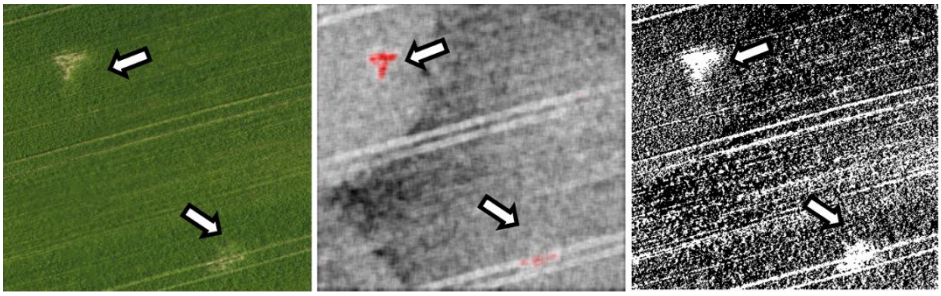

Figure 1. Reference measurement on wheat field – gopher nests marked with arrows

The survey was done on April 24th, 2015, and the full nest count was 56. By repeating the survey one month later, we experienced that the protection measures had been successful (*Fig. 2*). The number of nests has not increased; moreover, the damage has not escalated either. During our previous analysis, we conducted nest detection using a method based on image processing [14], but after some initial successes we found out that the procedure can be used successfully only in the given test case.

Subsequently, we turned towards a procedure which is neural-network-based. Based on a given number of well-chosen training samples, it is capable of detecting feature areas (nests) even outside the test area. Deep learning in a classical sense is based on a large number of training samples, which, after the network has learned them – in other words, stored the relevant information using its neurons from the hidden layers –, it is capable of classifying images which are newly entered and still unknown as inputs.

In our case, the goal is to use CNN in such a manner that we choose a given type of irregularity in the high-resolution images, which in the given case have been taken at different times, and then we detect it. However, we do not provide a large number of samples to the network as usual ([15], [16], [17]), but we choose a small number of positive and negative training samples randomly. In practice, this means 10 to 20 samples per class. One of the classes is the searched sample, the other 2 to 3 classes are agriculturally different features but have similar attributes in the RGB domain, analysed from an informatics point of view.

If the irregularity that is searched for is a gopher nest, then a training sample against it is, for example, a furrow track, an unsown furrow, and a gully. Examined in the RGB domain, these all constitute similar classes, but when examining their structure and contour they constitute different and clearly identifiable classes.

Figure 2. Gopher nest after prevention

D. R-CNN Gopher nest detector

Region-Based Convolutional Networks (R-CNN) use a small number of random samples. According to our theory, the operator only has to enter a few, clearly detectable training samples on the searched feature in the large-sized orthophoto, or rather the same number of similar features which are not from the same class as the searched pattern (*Fig. 3*).

The CNN is composed of 11 layers. The size of the input layer is 32 x 32 pixels by applying the RGB colour channel. Several convolutional layers will be in the intermediate layers, whose width and height determines the filter size the training procedure uses while scanning the images. In our case, we applied 32 filters per layer, which means that the same number of neurons connect to the only identical part of the input. This parameter determines the number of feature maps.

This layer is always followed by a Batch Normalization layer, which normalizes the activations and gradients propagating through a network, making network training an easier optimization problem. Applying the layer, the neural network can be made faster.

The next layer is a non-linear activation function, more precisely the Rectified Linear Unit (ReLU). A ReLU layer performs a threshold operation to each element of the input, where any value less than zero is set to zero (4).

$$f(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases}. \tag{4}$$

The network also contains a Max Pooling Layer, whose task is a down-sampling operation that reduces the spatial size of the feature map and removes redundant spatial information. The result of the reduction is that the number of filters can be increased; in this way, we can create a deeper network without the increase of the necessary computing power.

The Fully Connected Layer, as its name suggests too, is a layer whose neurons are in connection with all the neurons of the previous layer. It can combine the simpler features already learned by the previous layer, and it is capable of producing an even more complex conclusion for the sake of classifying the images.

The last two layers are the Softmax and the Classification Layer. Their task is to normalize the network output and, based on the probabilities of the network, to provide the class which belongs most to the image at the output.

The whole R-CNN with 11 x 1 layer array with layers:

1. Image Input, 32 x 32 x 3 images with “zerocenter” normalization;
2. Convolution, 32 filter, 3 x 3 convolutions with stride [1 1] and padding [1 1 1 1];
3. Batch normalization;
4. ReLU;
5. Convolution, 32 filter, 3 x 3 convolutions with stride [1 1] and padding [1 1 1 1];
6. Batch normalization;
7. ReLU;

8. Max pooling, 3 x 3 max pooling with stride [2 2] and padding [0 0 0 0];
9. Fully connected, 128 fully connected layer;
10. ReLU
11. Fully connected, 4 fully connected layer;
12. Softmax;
13. Classification output, crossentropyex.

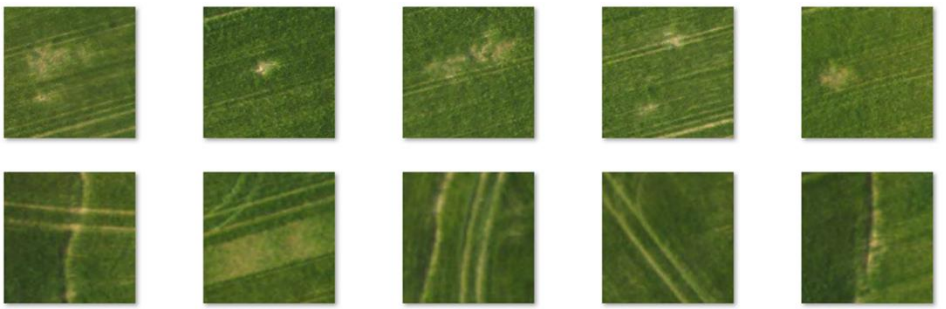

Figure 3. Training samples – upper line: gopher nest, bottom line: not gopher nest

3. Conclusions

Tests have been conducted examining the training time and the classification of the R-CNN.

Test PC:

- Operating system: Windows 7 SP1 64bit,
- Processor: Intel i7-3820 @3.60 GHz,
- Number of physical cores: 4,
- Number of logical processors: 8,
- Memory: 32GB DDR3,
- GPU: NVidia GeForce GTX TITAN Black 3GB.

Generally speaking, because of the low number of training samples, the training, using CUDA GPGPU acceleration [18], is expressly rapid: it is finished within 5 minutes (depending on the size of the regions selected for training purposes). In comparison, detection is much slower.

In case of full resolution orthophotos, which consist of 27,000 × 17,000 pixels [19], the processing time generally fluctuates around 30 minutes per image. The R-CNN was successfully trained in the mentioned way, in a way that during its run it is capable of detecting gopher nests with above 80% accuracy. Due to counter-

training samples, we managed to reduce the number of false detections to zero (*Fig. 4*).

Generally speaking, although the created method was originally designed for a large number of training samples, in this case, it was still applicable. During our work, we analysed the construction of R-CNNs, and we created the layer order that seemed to be the most ideal for our task, including its parameters.

Figure 4. Detected gopher nests, no false detections

During the test, we processed large-sized georeferenced orthophotos and proved the validity of the procedure. The next task is to increase the reliability of the created method, or rather to increase match accuracy (above 95%).

Acknowledgements

This study was supported by the ÚNKP-2017-4-III New National Excellence Program of the Ministry of Human Capacities.

References

[1] Zhang, C., Pan, X., Li, H., Gardiner, A., Sargent, I., Hare, J., Atkinson, P. M. (2018), A hybrid MLP-CNN classifier for very fine resolution remotely sensed image classification. *ISPRS Journal of Photogrammetry and Remote Sensing* 140, 133–144.

[2] Pouliot, D., Latifovic, R., Pasher, J., Duffe, J. (2018), Landsat super-resolution enhancement using convolution neural networks and Sentinel-2 for training. *Remote Sensing* 10(3), art. no 394.

[3] Shen, Y., Zhou, H., Li, J., Jian, F., Jayas, D. S. (2018), Detection of stored-grain insects using deep learning. *Computers and Electronics in Agriculture* 145, 319–325.
[4] Ha, J. G., Moon, H., Kwak, J. T., Hassan, S. I., Dang, M., Lee, O. N., Park, H. Y. (2017), Deep convolutional neural network for classifying Fusarium wilt of radish from unmanned aerial vehicles. *Journal of Applied Remote Sensing* 11(4), art. no 042621.
[5] Uzal, L. C., Grinblat, G. L., Namías, R., Larese, M. G., Bianchi, J. S., Morandi, E. N., Granitto, P. M. (2018), Seed-per-pod estimation for plant breeding using deep learning. *Computers and Electronics in Agriculture* 150, 196–204.
[6] Zhao, T., Wang, Z., Yang, Q., Chen, Y. Q. (2017), Melon yield prediction using small unmanned aerial vehicles. *Proceedings of SPIE – The International Society for Optical Engineering* 10218, art. no 1021808.
[7] Li, L., Fan, Y., Huang, X., Tian, L. (2016), Real-time UAV weed scout for selective weed control by adaptive robust control and machine learning algorithm. *2016 American Society of Agricultural and Biological Engineers Annual International Meeting, ASABE 2016*.
[8] Bejiga, M. B., Zeggada, A., Nouffidj, A., Melgani, F. (2017), A convolutional neural network approach for assisting avalanche search and rescue operations with UAV imagery. *Remote Sensing* 9(2), art. no 100.
[9] Ammour, N., Alhichri, H., Bazi, Y., Benjdira, B., Alajlan, N., Zuair, M. (2017), Deep learning approach for car detection in UAV imagery. *Remote Sensing* 9(4), art. no 312.
[10] Andrea, C.-C., Mauricio Daniel, B., Jose Misael, J. B. (2018), Precise weed and maize classification through convolutional neuronal networks. *2017 IEEE 2nd Ecuador Technical Chapters Meeting, ETCM 2017, January 2017*, 1–6.
[11] Lee, S. H., Chan, C. S., Mayo, S. J., Remagnino, P. (2017), How deep learning extracts and learns leaf features for plant classification. *Pattern Recognition* 71, 1–13.
[12] Fan, Z., Lu, J., Gong, M., Xie, H., Goodman, E. D. (2018), Automatic tobacco plant detection in UAV images via deep neural networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 11(3), 876–887.
[13] Kellenberger, B., Volpi, M., Tuia, D. (2017), Fast animal detection in UAV images using convolutional neural networks. *International Geoscience and Remote Sensing Symposium (IGARSS), July 2017*, 866–869, art. no 8127090.
[14] Stojcsics D., Molnar A. (2018), Automated evaluation of agricultural damage using UAV survey. *MACRo 2017: 6th International Conference on Recent Achievements in Mechatronics, Automation, Computer Science and Robotics, Târgu-Mureş, Romania, 27–28.10.2017*, 40–46.
[15] Vetrivel, A., Gerke, M., Kerle, N., Nex, F., Vosselman, G. (2018), Disaster damage detection through synergistic use of deep learning and 3D point cloud features derived from very high resolution oblique aerial images, and multiple-kernel-learning. *ISPRS Journal of Photogrammetry and Remote Sensing* 140, 45–59.
[16] Qayyum, A., Malik, A. S., Saad, N. M., Iqbal, M., Faris Abdullah, M., Rasheed, W., Rashid Abdullah, T. A., Bin Jafaar, M. Y. (2017), Scene classification for aerial images based on CNN using sparse coding technique. *International Journal of Remote Sensing* 38(8–10), 2662–2685.
[17] Sheppard, C., Rahnemoonfar, M. (2017), Real-time scene understanding for UAV imagery based on deep convolutional neural networks. *International Geoscience and Remote Sensing Symposium (IGARSS), July 2017*, 2243–2246, art. no 8127435.
[18] Kertész, G., Szénási, S., Vámossy, Z. (2015), Parallelization methods of the template matching method on graphics accelerators. *16th IEEE International Symposium on Computational Intelligence and Informatics – CINTI*, 161–164.
[19] Lovas I., Molnár A. (2018), Orthophoto creation based on low resolution thermal aerial images. *IEEE 12th International Symposium on Applied Computational Intelligence and Informatics (SACI 2018)*. Timişoara, Romania, 219–224.