VOLUME 16, N° 2 2022
Journal of Automation, Mobile Robotics and Intelligent Systems

Application of the OpenCV library in indoor hydroponic plantations for automatic height assessment of plants

Submitted: 26th April 2022; accepted 28th July 2022

Sławomir Krzysztof Pietrzykowski, Artur Wymysłowski

DOI: 10.14313/JAMRIS/2-2022/16

Abstract:
This paper presents a method for automatically measuring plants' heights in indoor hydroponic plantations using the OpenCV library and the Python programming language. Using the elaborated algorithm and Raspberry Pi-driven system with an external camera, the growth process of multiple pak choi cabbages (Brassica rapa L. subsp. Chinensis) was observed. The main aim and novelty of the presented research is the elaborated algorithm, which allows for observing the plants' height in hydroponic stations, where reflective foil is used. Based on the pictures of the hydroponic plantation, the bases of the plants, their reflections, and plants themselves were separated. Finally, the algorithm was used for estimating the plants' heights. The achieved results were then compared to the results obtained manually. With the help of a ML (Machine Learning) approach, the algorithm will be used in future research to optimize the plants' growth in indoor hydroponic plantations.

Keywords: *Hydroponics, Image Analysis, Automatics, Mechatronic Systems, OpenCV, Phenotyping*

1. Introduction

As noted by the second United Nations Sustainable Development Goal (UNSDG) – Zero Hunger – the need for improving the agricultural productivity of small-scale food producers, together with the diversification of the food-producing systems, is one of the key problems of today [1]. Increasing the volume of crops grown in urbanized areas – especially domestically – is one of the ways to increase the availability of high-quality food in regions where the amount of space that could be used for growing plants is limited [2]. Hydroponic plantations may partially solve these problems. In hydroponic growing, plants can be grown indoors, within stacked stations, and even without access to natural light. Hydroponics reduces land requirements for crops by 75% or more, and reduces water usage by 90%, compared to the traditional methods of growing plants [3]. At the same time, hydroponics can provide a yield over ten times greater than that of conventional equivalent methods [4]. On the other hand, the greatest challenge for hydroponics is its energy consumption, especially in enclosed systems with artificial light sources. Thus, there is a strong demand for the implementation of renewable energy sources in such applications [5]. Still, especially in regions suffering from hunger, i.e., East Africa, hydroponics is a promising technology for increasing food security. Implementation of this technology has the most potential in areas where water and fertilizer availability is limited, as well as in urbanized areas. Hydroponics also reduces negative impact on the environment compared to other methods, since it does not require pesticides and does not affect soil degradation [3, 5].

It should be underlined that this technology also shortens the supply chain and lowers the cost of healthy food, reducing transportation-related pollution [6].

With the fast development of digital technologies, the aforementioned hydroponic systems can be easily automated. Such technologies are getting cheaper and thus more affordable for domestic or SME (Small and Medium Enterprises) usage. The most promising digital technologies are image analysis and machine learning, which can be implemented alongside automation. It is even predicted that agriculture would benefit the most from digital technology implementation. Just a few applications of this technology include phenotyping, disease analysis, estimating crop yield, optimizing water usage, fertilizing, etc. [7–11].

From both food security and agricultural industry interest perspectives, one of the most important factors of crop production is cost-effectiveness, which limits both crop availability and profitability. Especially in systems based on artificial light usage, the longer a plant needs to grow, the more energy it consumes, increasing the overall cost. It also limits the production potential by occupying the system for an extended time.

In addition, the horticultural industry looks for cost reduction not only to grow food but also medicinal cannabis, the demands on light of which can be even seven times larger than in the case of crops.
There are multiple studies that elaborate on various photosynthesis mechanisms in order to increase the overall cost-effectiveness and the plants' growth rate [12–15].

The presented research's main goal is to propose an automatic plant assessment method for indoor grow systems that could be used, e.g., for growth rate optimization. In the elaborated case, a method based on the application of Python and OpenCV is proposed, and its implementation in a hydroponic system is shown. In chapter two, a short state-of-the-art of image processing and corresponding software tools are presented. Examples of the usage of such programs for hydroponics and crop cultivation are shown, pointing out the main difficulties in implementing these as universal tools. In the next chapter, the OpenCV library is described along with the performed experiment description, which refers to automatic height measurements of the plants grown in a hydroponic system. The goal was to extract the usable parameter for future plants' growth optimization. The fourth chapter contains information about the software and hardware used in the setup. Chapter five contains details concerning the elaborated method's algorithms. The sixth chapter compares results obtained with the elaborated method with those obtained with manual measurements. The last chapter contains a summary.

2. State-of-the-Art

Hydroponics is a method of growing plants in a liquid solution instead of soil. Most basic techniques use a simple mixture of tap water and organic fertilizers. However, these methods tend to result in lesser efficiency in growing crops. More advanced methods involve the usage of a mixture of demineralized water, synthetic fertilizers, carbon dioxide dosing, and pH balancing. In addition, some systems use reverse osmosis to reduce the amount of water waste.

As mentioned, hydroponics can be run with or without access to natural sunlight. In this case, advanced grow light systems are being used. By limiting the light spectrum to the absorption spectra of the plants' dyes, it is possible to achieve cost-efficient light sources, which can even increase the plants' growth rate in comparison to plants grown under natural sunlight. Additionally, due to the low power consumption of LED light sources, it is possible to use renewable energy sources with them, such as solar panels.

Machine learning techniques and digital image analysis are beneficial in automatic hydroponics applications. For example, there are a number of commercial and free software libraries supporting both digital image analysis and machine learning. Combining both can help to classify or extract key parameters of plants from digital images. The current paper focuses on digital image analysis, while in future research, it is planned to combine both.

Currently, the most popular free software for digital image analysis and recognition of plants are OpenCV [16], SimpleCV [17], BoofCV [18], PlantCV [19], PyCV [20], etc. Although multiple image-processing tools can be used for plant and crop analysis, a vast majority of them have some limitations. The first one, which highly reduces the possible implementations, is the requirement of providing a plain background for photographing plants [21–23]. This impediment almost excludes the possibility of application of such tools to indoor hydroponic stations, where, very often, a reflective foil is used. Such a foil is commonly used to get as much light as possible and reduce the same light dispersion. The drawbacks of such a solution, in the case of image analysis, are the plants' reflections. In order to use such tools, a user would need to take the plant out of the plantation to take pictures. From the automation point of view, such a procedure is unwanted and may set the whole point under question.

Other solutions that allow multi-plant analysis, such as PlantCV, rely on setting up special markers to separate the plants' areas and/or are based on a pattern [16]. Such an approach is hard to implement as a universal solution, requires additional space, and may cause problems when trying to implement such a solution for automatic analysis of plants, e.g., their height assessment. Since this also requires a plain background, the possibility for application for stations with reflective foil is thus limited.

Multiple studies elaborate on the problem of phenotyping using machine vision. However, this mostly refers to outdoor plantations, where the UAV (Unmanned Aerial Vehicle) usage is being considered and implemented [24–25]. Other methods that try to focus on the height assessments in indoor plantations report the usage of the RGBD (Red Green Blue Depth) sensors [26]. Despite very interesting results, there is a considerable lack of verification methods for these obtained values, which were estimated instead of measured. Due to this fact, there is no actual information about the accuracy of such methods, and the report about the results is questionable.

3. Materials and Methods

The following paper refers to the application of one of the most advanced free software libraries for image analysis, which is OpenCV [16]. It was assumed that the above library could be used for image-processing and thus optimization of crops' growth rate in domestic, indoor hydroponic plantations. OpenCV contains API (Application Programming Interfaces) for different programming languages like C++, Python, Java, and Matlab [27].

The presented research uses the OpenCV library to analyze the commercially available hydroponic station by using an external camera. The above experimental setup was used for investigating pak choi cabbage *(Brassica rapa L. subsp. Chinensis*) growth. In the research, we focused on the problem of measuring the plants' height in an environment that could cause issues such as reflections with the image processing. The main problem was distinguishing between the real plant and its counterpart or even counterparts, as there could be more than one reflection in the background.

The performed experiments' primary motivation was to implement image recognition techniques to provide a simple parameter (plants' height) that could be used for the optimization of the crops' growth rate in hydroponic plantations. The elaborated algorithm introduces novelty by focusing on applications in which the environment's background is not plain and where reflective foil is being used. The test experiments were based on a commercially available hydroponic station and were allowed to gather referential data that could be used in the future experiments, which plan to focus on using machine learning algorithms in order to maximize the plants' growth rate in hydroponic plantations.

3.1. Experimental Setup

The experimental setup consisted of the following hardware and software:

- commercial hydroponic system "Green Farm,"
- OpenCV library,
- single board computer Raspberry Pi version 3B running under Raspbian OS,
- OV5647 5Mpx camera.

A diagram of the applied system is presented in Figure 1.

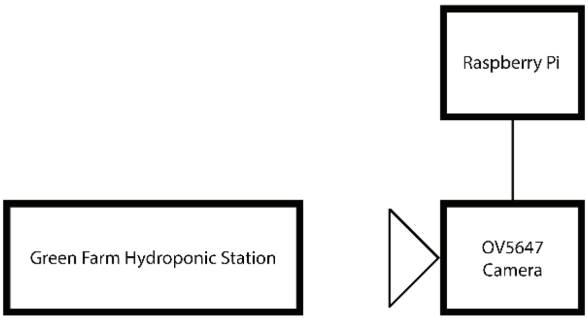

Fig. 1. Diagram of the designed and applied system

The decision to use the commercially available hydroponic system was chosen in order to implement automatic height measurements in a way that could be replicated in domestic applications. The Green Farm hydroponic station provided basic solutions for controlling the plants' environment and allowed them to obtain sustainable growth. It was modified with a reflective foil to reproduce these conditions in enclosed stations.

The Raspberry Pi allows remote measurements without the need for a stationary PC, which is also a good choice in the case of domestic applications. It is one of the most popular SBC (Single-Board Computers), as it is both cheap and effective. The built-in port CSI (Camera Serial Interface) allows for easy implementation of external cameras. The Raspbian OS with implemented support for Python language allows for the use of OpenCV without any flaws.

The OV5657 camera is one of the most popular, cheap camera modules for Raspberry Pi and Arduino (ArduCam). The provided resolution of 5 Mpx and 72.4° camera angle provided good quality pictures for future processing.

The actual experimental setup is presented in Figure 2.

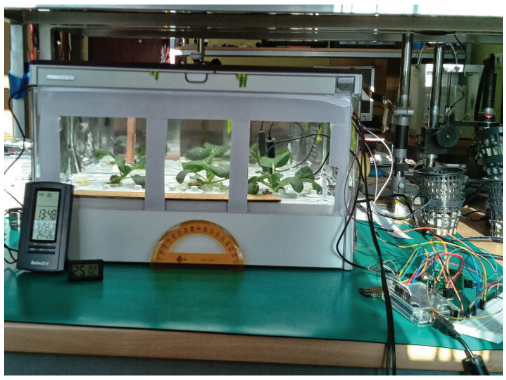

Fig. 2. Experimental setup

3.2. Hydroponic System

In the research, the commercial Green Farm hydroponic station was used, which is presented in the figure 3. The device provided a simple set of sensors, such as a real-time clock, temperature sensor, and analog water level meter. These helped to provide essential and sustainable growth of the plants in a controlled hydroponic environment.

Fig. 3. Green Farm hydroponic station

To be able to reproduce the setup in most indoor hydroponic systems, it was decided to set up an OV5647 camera, together with Raspberry Pi, outside of the system. This decision made the setup easily reproducible in most commercial hydroponic stations, simply by placing the camera outside of the setup and – if needed – opening the grow box without taking the plants outside of the station.

However, some more advanced hydroponic plantations use special grow boxes and reflective foil, such as mylar, to increase the amount of light delivered. It must be emphasized that putting the camera inside of the station's tank would be more accurate. However, this would limit the number of cases in which the method could be applied and would require additional modifications to the already-made stations. The reason for choosing this solution was to find out whether OpenCV is a capable tool for collecting data for optimization of the plants' growth rate, even in the case of commercially available hydroponic plantations that do not include machine learning technologies. To increase the challenges of plant detection and make the case more difficult and universal for indoor plantations, we added plain reflective foil on the investigated hydroponic station walls. Such a solution can simulate both mylar and cheap foil, which are often used in cheap and homemade stations. In order to take pictures of the plants, small windows were cut from the foil.

3.3. Software and Hardware Tools

For image processing, the OpenCV library and Python programming language were used. The advantages of the OpenCV library are advanced image processing methods and an available API for Python language, which combines the best qualities of OpenCV C++ API with Python.
The design and application of the algorithm consisted of the following stages:

- finding the bases of the plants, separating the areas containing plants from the rest of the image using the OpenCV library,
- finding the plants within the specified areas,
- obtaining their tops' Y position and calculating the height.

The features of the OpenCV library that were used in the experiment, together with their functionality, are given in Table 1. Since OpenCV hardly relies on the NumPy library for providing compatibility and calculations, the functions used from the NumPy library for Python are given in the table 2.

The core of the system was a Raspberry Pi, with an OV5647 camera connected to it. The decision to use a Raspberry Pi was made due to the low cost and simplicity of programming, using the external camera, and later processing of the images. Due to the implemented solutions for remote access, by using SSH (Secure Shell), VPN (Virtual Private Network), and/or VNC (Virtual Network Computing), remote control and access were possible.

The camera module that was used – OV5657 – provided a resolution of 5 Mpx and a 72.4° camera angle, which resulted in reasonable resolution and quality of the pictures for future processing. The pictures of the hydroponic station containing three plants were taken once a day. These pictures were then used for processing in order to extract the values of the plants' height.

Tab. 1. OpenCV library's functions and their descriptions

Function	Description
Imread; Imwrite	reading and saving the image files
bitwise_not; bitwise_and	bitwise operations on pictures
absdiff	finding the differences between images
gaussianBlur	blurring the masks
threshold	creating a mask of specific values
Dilate	morphological expansion of the masked areas
cvtColor	converting color spaces (RGB, HSV, etc.)
convexHull	Finding the convex hull of a 2D point
findContours	finding the contours in the image
inRange	checking if array elements are placed between the elements of two other arrays

Table 2. NumPy library's functions and their descriptions

Function	Description
array	creating arrays
zeros	creating arrays filled with zeros
where	returning the chosen element from the arrays depending on the condition
vstack	stacking the arrays in sequence vertically

3.4. Image Recognition Method

The elaborated image recognition algorithm consisted of the following steps:

- removing white and gray areas,
- removing the background with absdiff function,
- HSV (Hue Saturation Value) and RGB (Red Green Blue) filtering,
- separating images with the plants,
- finding contours,
- checking the size criteria,
- calculating the distance between the reference starting points and the plants' tops.

The first issues we encountered were with black, white, and gray-colored areas. These are some of the common colors of tanks, pipes, housing, baskets for plants, etc. The easiest way to remove these areas from the pictures in OpenCV is with simple image-processing, such as removing any pixels between a specific threshold. In this case, the threshold was implemented in the RGB color space with the values of R, G, and B between 162 and 255 for each channel.

The second way of dealing with the background was to use the differences between the captured images. In some cases, such a solution can easily detect the plants by using the picture of a plantation without plants as the reference image. However, in the investigated hydroponic station, the water level is not constant, and the plants' elevation varies from one picture to another, which is crucial for height assessment. Thus, such a method is not always viable. However, it is a good practice to remove the background, even when the light conditions slightly vary. Therefore, it was decided that we would implement the differential method by comparing the picture taken on the first day of the experiment with those taken on successive days. The following steps are shown in Figure 4

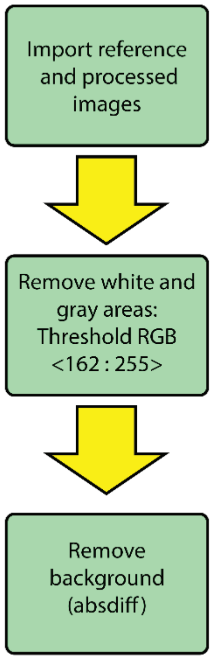

Fig. 4. Diagram of the steps that were taken to remove the background from the picture

The next step was to remove from the picture other objects/artifacts aside from plants. An excellent solution to this problem is to use color filtering. In the presented case, since there is a lot of noise coming from the background, station interior, and reflections, it was decided to use double-step filtering – both in the RGB and the HSV color spaces. It was noticed that RGB is great for filtering the wider ranges of color, but using the HSV color space helps to filter out narrow ranges of color. The lower RGB ranges for both the plants and bases detection were 190, 160, and 125, respectively, while for the top range, the value of 255 was used for each channel. In the elaborated case, the filtering had to be very strict due to extreme reflection conditions. This filtering resulted in the deconstruction of the plants' shapes. However, to avoid detecting the reflections, this was a necessary step. To deal with the deconstruction of the plant images, the analysis of the image was split into multiple phases. In the first stage, the images of the plants were cropped basing on the contours of the bases of the plants, which in the early stages of the growing process are easier to distinguish. The values of HSV filters for this step were adjusted to detect the surrounding rockwool.

Multiple other approaches could also be implemented. When using small inserts made of foam, the HSV filter rules could be adjusted, or this step could even be replaced with the detection of small plants in the early stages of growth. However, it was found that this way creates the most repetitiveness in the measurements.

For the case of HSV filtering, the *cv2.dilate()* function was used. After this step, all the aforementioned filters were combined using the *cv2.bitwise_and()* function on the source images, using the filtered images as the masks. The lower values of HSV filtering for the bases were set to 26, 91, and 118, respectively, while the top of the range was set to H = 33, S = 113, and V = 223. For the plants, the lower HSV threshold was set to 37, 65, 95, respectively, and 58, 250, 255 for the top values. This step is illustrated in Figure 5a.

For detecting the contours, both for the bases and plants, all the external contours on the picture were detected using the *findContours()* function. An array filled with zeros, called *cont_status*, was created afterward. Then, iterating over all of the contours, the distances between them were checked. If the criteria were met, the *cont_dist* variable for the contours was set to *True*, and the contours were connected together, creating multiple bigger, continuous shapes. The same was done with the sizes of evaluated contours and the *cont_area* variable. In this case, the distances and the sizes of the contours were investigated. After this process, all the connected contours were put into a separate *cont_connected* list.

As the last step for both of the stages, the *cont_connected* contours were checked again against another size criterion. This was done by iterating over all of the contours in the *cont_connected* list and checking if their size was bigger than the *min_shown_area* variable. All the contours that met the criteria were put into the *cont_final* list. For the iterations made for plant analysis, another *cont_connected_copy* list was made. The reason for that was an error that occurred when trying to process the same list with *cv2.ContourArea()* function multiple times with the OpenCV library version 4.4.0.44 that was used.

For most extreme cases, in which none of the final contours would meet the size criteria, a loop that decreased the value of the size threshold was implemented. This safety feature allowed us to get at least partial contours when the reflections did not allow for obtaining the full contours of the plants.

After obtaining the list of final contours, by subtracting the top Y-position value obtained, the height of the plants was calculated. This whole process is illustrated in Figure 5b.

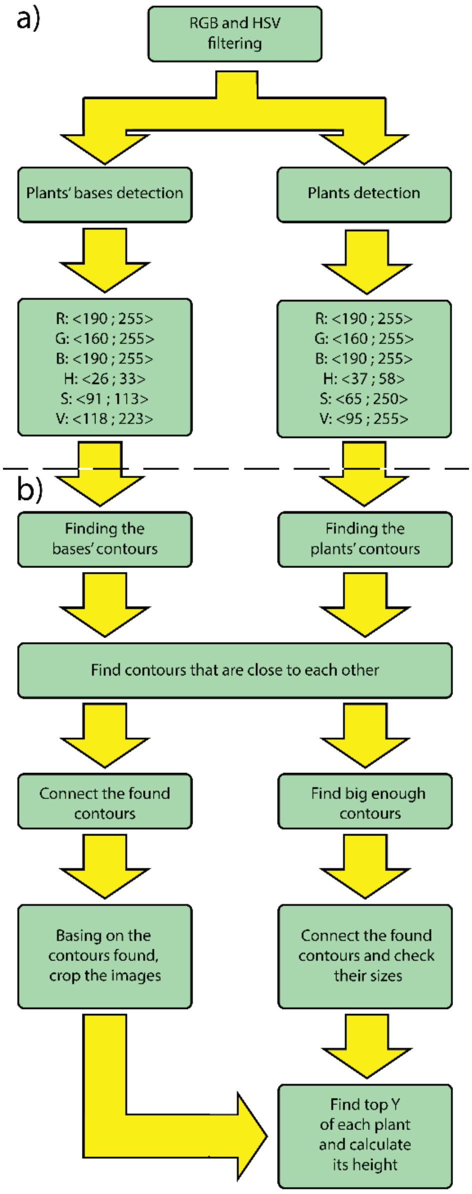

Fig. 5. Diagram of a) the HSV and RGB filtering, together with the values used, and b) the process of connecting the contours and calculating the plants' heights

Results and Discussion

As a result of the image processing, a series of continuous measurements of the three plants' heights was estimated. The process took 15 days, resulting in 3 reference pictures of the plants and 42 separate images that were processed to estimate the plants' heights each day.

The results obtained by the algorithm were calculated in millimeters using the plantations' height as a reference. This is represented by the following formula:

$$height[mm] = \frac{h_{ref}[mm]}{h_{ref}[px]} * h_{plant}[px] \quad (1)$$

where: h_{ref} is a reference height object (the hydroponic station's height) and h_{plant} is the plant's height.

From all the measurements, the mean of plants' height for each day was calculated using formula 2. The results of numerical assessments ($h_{assessed}$) are shown in Figure 6. The values were fitted with an exponential function, the parameters of which are shown in Table 3. The confidence band of 99.7% was then calculated:

$$h_{mean}[mm] = \frac{\sum_{i=1}^{n} h_i}{n} \quad (2)$$

where h_i is the i-plant's height and n is the total number of plants grown.

To check if the results obtained by the algorithm were valid, the mean calculated height values were compared to the results of the manual height assessment, and Er (Relative Mean Error) values were calculated for each measurement point. Manual measurements were conducted by measuring the pixel distance between the plants' rockwool base and the top of the plants, and calculating the distance using the reference distance of a ruler and the hydroponic station's height. Then, the results were confirmed by measuring the plants with a ruler. Er values were then approximated with a linear curve, and a 99.7% confidence band was calculated, which corresponds to the 3sigma standard. Both means, together with 99.7% confidence bands, were calculated according to the equation 3 and are shown in Figure 7.

$$C_b = \hat{y} \pm t_{\alpha/2} \quad (3)$$

where C_b is the confidence band, ŷ is the actual measured value at x_p, α is the confidence level, and $t_{\alpha/2}$ is the standard error of prediction for the value.

Additionally, the relative error Er is shown in Figure 8. Note that the plants' mean height is very precisely calculated in the vast majority of the cases, showing that the elaborated algorithm allowed for achieving reliable measurements of the plants' height.

Despite the reflections and uneven background, the measurements were performed with very satisfying results. Parts of the source pictures with the overlaid result images from the first, fifth, and fourteenth day of the experiment are shown respectively in Figures 9–11. In the first days, after growing their stems, the plants were observed to expand their leaves, which was the dominant process affecting the plants' height up to the fifth day. From this point onward, the plants continued to grow their leaves. On the eighth day, plants spread out, and increased the number of, their leaves, which caused the observed height to drop on the ninth day. From this point, an uninterrupted height increase was observed.

It is worth noting that due to the heliotropism and the growth mechanisms of the plants, on different day cycles, the plants' height varies. In the first stages of growth, the plant is looking for a light source, expanding its stem. Then, the process of expanding the leaves begins, which often cannot be held up by the thin stem. That is the reason why the B and C plants' height in the Figure 10 is lower than on the first day (Figure 9). In addition, the plants' height and the leaves' shape may vary during the light cycle.

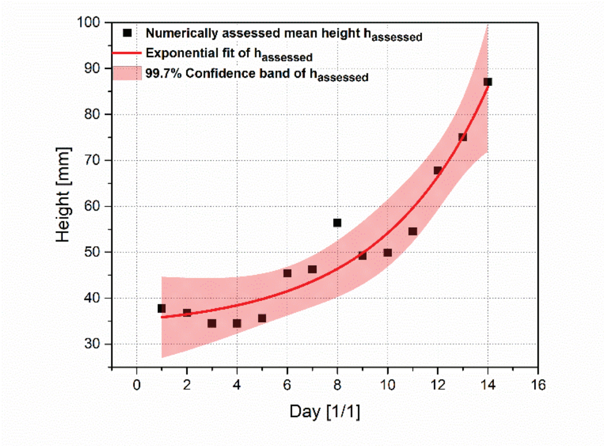

Fig. 6. Numerically assessed plants' mean height and its exponential fit, together with a 99.7% confidence band

The obtained results prove the accuracy of the method, which was confirmed by the manual assessments. The maximum relative error was captured at 10%, which translated to 7 mm. The mean absolute error value was 2 mm.

With the exception of the measurement on the eighth day of the experiment, all the measurements fit into the 99.7% confidence bands of exponential approximation of the height vs. time function. This suggests that another mechanism was dominant in this growth phase, which was explained in previous paragraphs. However, the obtained results proved the usefulness of the method for automating plants' height assessments. Additionally, it should be underlined that the plant's height was assessed according to the evaluated rockwool bases' positions and the measured top positions of the plants' contours.

Tab. 3. Parameters of exponential approximation of the plants' mean height

Parameter	Value
Model	Exponential
Equation	
y_0	
A	
R_0	
Reduced Chi-Sqr	19.42
R-Square(COD)	0.94
Adj. R-Square	0.93

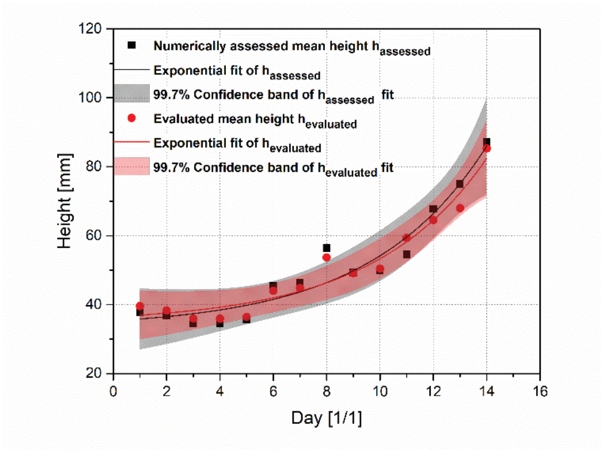

Fig. 7. Capturing and measuring plants' mean heights together with exponential fits and a 99.7% confidence band

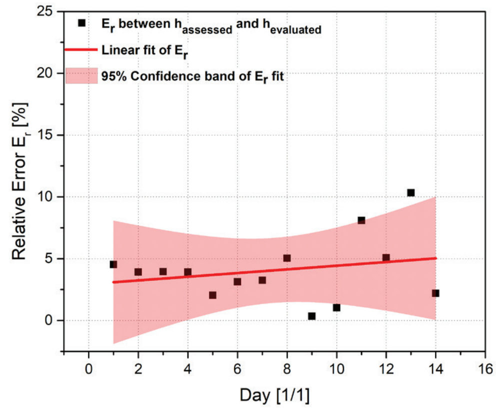

Fig. 8. Relative Error Er between numerically assessed and manually measured height values together with linear approximation and a 99.7% confidence band

The elaborated algorithm will be used in the future research as a base for an autonomic hydroponic plantation, supported by machine learning techniques, that will fully control and optimize plants' growth rate with light spectra adjustments, dosage of fertilizers, water, pH control, etc.

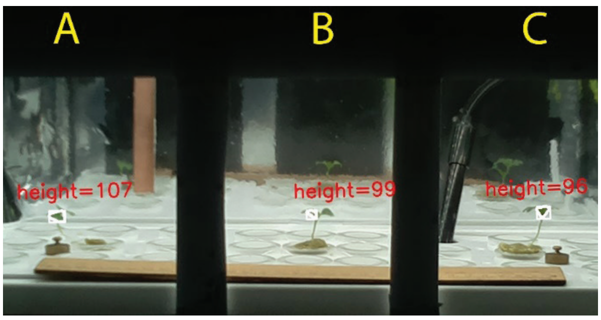

Fig. 9. A fragment of the source picture from the first day with the overlaid result images, together with notations for each plant – starting from the left – A, B, and C

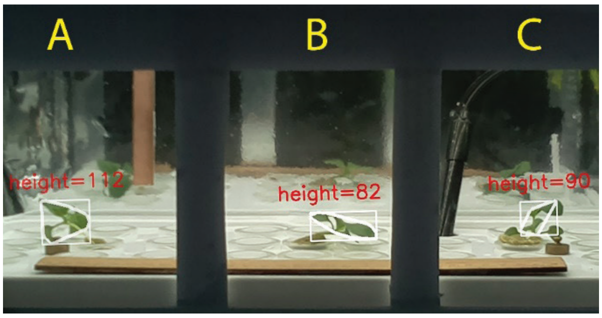

Fig. 10. A fragment of the source picture from the fifth day with the overlaid result images, together with notations for each plant – starting from the left – A, B, and C

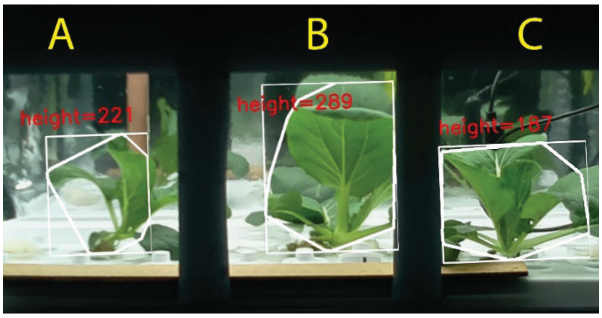

Fig. 11. A fragment of the source picture from the fourteenth day with the overlaid result images, together with notations for each plant – starting from the left –A, B, and C

4. Summary

The goal of the research was to design a universal tool that would allow for automatic height assessments of the plants in hydroponic plantations.

According to the brief state-of-the-art, it was shown that most of the tools available for computer-vision-based automatic height assessments are problematic for indoor hydroponic plantations, where multiple plants are being grown, especially when a reflective foil is used.

Based on the elaborated hardware and software setup, an algorithm for OpenCV Python's library was proposed and explained. This algorithm was then used to process images of the Green Farm hydroponic indoor plantation, which was modified by using a reflective foil, where three pak choi cabbages were being grown. Based on 15 source images that were captured over 14 days, 42 result images, together with the numerically assessed plants' heights, were prepared by the algorithm.

The results allowed for calculations of the mean plants' height values, which were compared to the values obtained with manual measurements. For both numerically assessed and manually measured values, an approximation was made, together with 3sigma confidence bands. These values were then compared, and the relative error between them was calculated. The results have shown that the elaborated algorithm has very good precision and that it can be used as a viable tool for automatic height assessments in indoor hydroponic plantations with multiple plants, even when reflective foil is used.

The obtained results will be used as a base for future experiments that will take place in order to optimize the plants' growth rate in a fully-automatic, machine-learning-driven hydroponic system supported by a number of detectors, sensors, and actuators.

AUTHORS

Sławomir Krzysztof Pietrzykowski* – Faculty of Microsystem Electronics and Photonics, Wroclaw University of Science and Technology, Janiszewskiego Street 11/17, 50-372 Wroclaw, Poland,
e-mail: slawomir.pietrzykowski@pwr.edu.pl.

Artur Wymysłowski – Faculty of Microsystem Electronics and Photonics, Wroclaw University of Science and Technology, Janiszewskiego Street 11/17, 50-372 Wroclaw, Poland,
e-mail: artur.wymyslowski@pwr.edu.pl.

*Corresponding author

REFERENCES

[1] United Nations, "2030 Agenda for Sustainable Development", available from: https://www.un.org/ga/search/view_doc.asp?symbol=A/RES/70/1&Lang=E, accessed: 2022-06-21

[2] Food and Agriculture Organization of the United Nations, "Farming in urban areas can boost food security", 2005. Available from: https://reliefweb.int/report/world/farming-urban-areas-can-boost-food-security

[3] Barbosa, Guilherme Lages et al. "Comparison of Land, Water, and Energy Requirements of Lettuce Grown Using Hydroponic vs. Conventional Agricultural Methods", International journal of environmental research and public health vol. 12, no. 6, 2015, 6879–6891. .DOI:10.3390/ijerph120606879

[4] Stefan P. et al. "A comparative cost-effectiveness analysis in different tested aquaponic systems", Agriculture and Agricultural Science Procedia, vol. 10, 2016, 555–565. DOI: 10.1016/j.aaspro.2016.09.034.

[5] Margaret S. et al. "Building sustainable societies trough vertical soilless farming: A cost-effectiveness analysis on a small-scale non-greenhouse hydroponic system", Sustainable Cities and Society, vol. 83, 2022, 101882. DOI: 10.1016/j.scs.2022.103923.

[6] A. Malik et al. "A review on the science of growing crops without soil (soilless culture) – a nov-

el alternative for growing crops", International Journal of Agriculture and Crop Sciences, vol. 7, 2014, 833–842.

[7] X. E. Pantazi et al. "Detection of biotic and abiotic stresses in crops by using hierarchical self-organizing classifiers", Precision agriculture, vol. 18 no. 3, 2017, 383–393. DOI:10.1007/s11119-017-9507-8.

[8] S. Amatya et al. "Detection of cherry tree branches with full foliage in planar architecture for automated sweet-cherry harvesting", Biosystems Engineering, vol. 146, 2016, 3–15. DOI: 10.1016/j.biosystemseng.2015.10.003.

[9] M. Ebrahimi et al. "Vision-based pest detection based on SVM classification method", Computers and Electronics in Agriculture vol. 137, 2017, 52–58 DOI: 10.1016/j.compag.2017.03.016.

[10] X.-E. Pantazi et al. "Active learning system for weed species recognition based on hyperspectral sensing", Biosystems Engineering, vol. 146, 2016, 193–202. DOI: 10.1016/j.biosystemseng.2016.01.014.

[11] G. L. Grinblat et al. "Deep learning for plant identification using vein morphological patterns", Computers and Electronics in Agriculture, vol. 127, 2016, 418–424. DOI: 10.1016/j.compag.2016.07.003.

[12] F. M. Westmoreland et al. "Cannabis lighting: Decreasing blue photon fraction increases yield but efficacy is more important for cost effective production of cannabinoids", PLOS ONE, vol. 16, no. 3, 2021. DOI: 10.1371/journal.pone.0248988

[13] V. Rodriguez-Morrison, "Cannabis Yield, Potency, and Leaf Photosynthesis Respond Differently to Increasing Light Levels in an Indoor Environment", Frontiers in Plant Science, vol. 12, 2021. DOI: 10.3389/fpls.2021.646020

[14] M. W. Jenkins, "Cannabis sativa L. Response to Narrow Bandwidth UV and the Combination of Blue and Red Light during the Final Stages of Flowering on Leaf Level Gas-Exchange Parameters, Secondary Metabolite Production, and Yield", Agricultural Sciences vol. 12, no. 12, 2021, 1414–1432. DOI: 10.4236/as.2021.1212090

[15] M. Moher et al. "High Light Intensities Can Be Used to Grow Healthy and Robust Cannabis Plants During the Vegetative Stage of Indoor Production", Preprints (2021). DOI: 0.20944/preprints202104.0417.v1

[16] I. Culjak et al. "A brief introduction to OpenCV", 2012 Proceedings of the 35th International Convention MIPRO, 2012, 1725–1730.

[17] SimpleCV, http://www.simplecv.org, accessed: 2021-02-07.

[18] BoofCV, http://boofcv.org/, accessed: 2021-02-07.

[19] M. Gehan (Dong) et al. "PlantCV v2.0: Image analysis software for high-throughput plant phenotyping". *PeerJ*, vol. 5, 2017. DOI: 10.7287/peerj.preprints.3225.

[20] M.-T. Pham, T.-J. Cham, "Online learning asymmetric boosted classifiers for object detection", 2007 IEEE Conference on Computer Vision and Pattern, Minneapolis, USA, 2007, 1–8. DOI: 10.1109/CVPR.2007.383083.

[21] A. K. Hase et al. "Detection, categorization and suggestion to cure infected plants of tomato and grapes by using OpenCV framework for Android environment", 2017 2nd International Conference for Convergence in Technology (I2CT), 2017, 956–959. DOI: 10.1109/I2CT.2017.8226270.

[22] I. Suryawibawa et al. "Herbs recognition based on android using OpenCV, International Journal of Image, Graphics and Signal Processing", vol. 7, 2015, 1–7. DOI: 10.5815/ijigsp.2015.02.01.

[23] Y. Chen et al. "Research on pest image processing method based on android thermal infrared lens", IFAC-PapersOnLine, vol. 51, no. 17, 2018, 173–178.DOI: 10.1016/j.ifacol.2018.08.083.

[24] P. Hu et al. "Estimation of plant height using a high throughput phenotyping platform based on unmanned aerial vehicle and self-calibration: Example for sorghum breeding", European Journal of Agronomy, vol. 95, 2018, 24–32. DOI: 10.1016/j.eja.2018.02.004

[25] MA Hassan et al. "Accuracy assessment of plant height using an unmanned aerial vehicle for quantitative genomic analysis in bread wheat", Plant Methods vol. 15, 2019, 15–37. DOI: 10.1186/s13007-019-0419-7

[26] B. Franchetti et al. "Vision Based Modeling of Plants Phenotyping in Vertical Farming under Artificial Lighting", Sensors, vol. 19, 2019. DOI: 10.3390/s19204378

[27] OpenCV – about, https://opencv.org/about/, accessed: 2021-02-07.