Blue Apple - An algorithm to realize agricultural classification under difficult light and color situations

Jonas Credner[1], Peter Rehrmann[2], Waldemar Raaz[3] and Thomas Rath[4] iD

Abstract: Computer image processing becomes more and more important in the analysis of data in biological and agricultural research and practice. However, robust image processing is highly dependent on the histogram analysis algorithms used and the quality of the data being processed. The algorithm presented here aims to improve the accuracy of the classification of image data generated under complex boundary situations and inconsistent lighting conditions. Using the example of the determination of nitrogen content of tomato leaves and the qualitative determination of starch content of apples on the basis of color image processing, we showed that the developed algorithm is able to perform a robust classification and represents an improvement to simple histogram analysis.

Keywords: Classification, Histogram Analysis, Color Image Processing, Starch Detection Apples, N-Analysis Plants

1 Introduction & Objectives

Spatial data or images are the most important data sources in biology and agricultural science in practice and research. Today, image processing algorithms that analyze different color spaces with different color channels are mainly used to evaluate such data [HCR18, NAA21, PR18]. By using standard cameras the number of color channels is limited to three. Newer sensor data sets often also contain a higher number of different color channels (multispectral data) [Ra17]. The evaluation of such data is usually done in such a way that first the objects are separated that are of importance (foreground-background separation or binarization), then the objects are analyzed by color, shape and other morphological properties and compared with existing (calibrated) data sets. At the end the previously unknown dataset (image) is classified into one or more reference data. Histograms of the color channels are often used in this process [HLE18]. These histograms are evaluated additively, multiplicatively or by artificial intelligence methods. The special problem is that the histograms are usually not normally distributed, since they originate from a natural image or a natural environment. Thus, they are skewed, contain an unknown number of zero values, and, above all, are often provided with several peaks. Therefore, statistical methods that assume a normal distribution of the histograms cannot be used. Likewise, it significantly complicates the use of deep learning methods, since

---

[1] Biosystems Engineering Laboratory (BLab), Osnabrück University of Applied Sciences, Oldenburger Landstr. 24, 49090 Osnabrück, Germany, j.credner@hs-osnabrueck.de

[2] Biosystems Engineering Laboratory (BLab), Osnabrück University of Applied Sciences, Oldenburger Landstr. 24, 49090 Osnabrück, Germany, p.rehrmann@hs-osnabrueck.de

[3] Biosystems Engineering Laboratory (BLab), Osnabrück University of Applied Sciences, Oldenburger Landstr. 24, 49090 Osnabrück, Germany, waldemar.raaz@hs-osnabrueck.de

[4] Biosystems Engineering Laboratory (BLab), Osnabrück University of Applied Sciences, Oldenburger Landstr. 24, 49090 Osnabrück Germany, t.rath@hs-osnabrueck.de,
https://orcid.org/0000-0001-7277-7335

essentially the results depend on the infinitely possible combination of input data. In addition, different lighting situations in the reference data set and (or) within the test image complicate the evaluation, so that today the evaluation of biological color images under different natural lighting conditions is a major challenge [JTZ11]. Figure 1 illustrates the problem described above with a simple example. An algorithm should assign a test image in the RGB color space with the help of a simple histogram comparison to one of three images in a reference database; namely to the image which comes closest to one of the reference images. Due to the skewness of the histograms of the reference images and the many zeros in the histograms, the algorithm does not succeed in assigning the test image (and this, although from the optical impression a clear assignment would be possible).

In order to be able to solve the presented problem of the histogram evaluation in biological images with high variability, new procedures are necessary, which

- consider the non-statistical distributions of histograms
- take into account the histograms of all available color channels
- be independent of the number of color channels and color spaces
- be independent of the compression of the original images
- compensate lighting differences as far as possible

2 Methods & Algorithms developed

The Blue-Apple algorithm was developed with the image processing software Halcon (MVTec, Munich, Germany) with the intention to enable difficult color classifications of biological-agricultural objects and to improve or support previous mentioned procedures and methods. It can be divided into three sections.

Section 1: In a first binarization step, all image parts except for the "region of interest" are removed in order to use only relevant image information for the subsequent classification process. This includes at least the removal of the background as in the first example in chapter 3 (determination of the nitrogen content of tomato leaves). In the second example of starch analysis of apple halves, the removal of the apple core housing is also required. Thus, as defined in the test procedure, only the fruit flesh is analyzed in the discoloration reaction and false conclusions or pseudo-correlations due to the apple core housing are avoided. The corresponding algorithms are strongly based on the use of classical morphological operators and shape factors. They are briefly presented in sections 3.1 and 3.2, and the implemented Halcon code with the corresponding operators can be downloaded from [Re22] and [Cr22].

Section 2: The main procedure of the developed analysis algorithm is the subsequent conversion of the relative histograms of the individual color channels into Gaussian-like distributions. In the following, these converted histograms are referred to as pseudo-Gaussian histograms. This should demonstrate that they are not Gaussian distributions, since a statistically clearly defined Gaussian profile cannot be generated from data with arbitrary (non-normal) distribution. However, the following criteria are imposed on the pseudo-Gaussian histograms generated with our algorithm:

- There are no zeros in the entire definition range of the histogram, i.e. no element of the histogram vector may contain the value 0.
- The histogram values decrease towards the edges. Whether these decreases are still visible in the finally generated histogram or lie outside the definition range of the histogram is irrelevant.
- The peaks and data distributions present in the original histogram are taken into account as much as possible when creating the pseudo-Gaussian histograms.
- The pseudo-Gaussian histograms are normalized to a vector length of 1.
- The pseudo-Gaussian histograms consist of interpolation points between which the vector elements are linearly interpolated.
- The pseudo-Gaussian histograms are composed of two half-sided histograms (right side and left side), so that extremely skewed distributions can also be modelled through different histogram halves. The separation between the right and left halves does not have to be exactly in the middle of the histogram. Rather, it depends on the initial data set and its distribution.

If $H_{i,z}$ is a histogram with v values h_i of a color channel i of the image z, first a relative input histogram $H_{rel,i,z}$ with $h_{rel,i} = h_{abs,i} / \sum h_{abs,i}$ is generated from the absolute input histogram $H_{abs,i,z}$. Subsequently, the relative input histogram is extended at the beginning and at the end by at least x (default: x >= 256) values each. The new $h_{arb,i}$ values are set to 0. The histogram $H_{arb,i,z}$ formed in this way is called the working histogram and contains v = 3x numerical values, whereby the middle third contains the original relative histogram $H_{rel,i,z}$. Subsequently, interpolation points for the new pseudo-Gaussian histogram $H_{Gau,i,z}$ are derived from the working histogram separately for left and right sides. For this purpose, the expected value for the maximum of the working histogram is first determined by calculating the center of gravity line of the histogram area. Then linear interpolations are carried out between interpolation points in the working histogram.

The following steps were used:

Coming from the left, the point µL is determined in the histogram where the partial sum of h_1 to $h_{\mu L}$ exceeds 50% of the total area of the histogram above the x-axis. In the same way coming from the right, with the partial sum h_{max} to $h_{\mu R}$ the point µR is determined.

The indices of the important points of the working histogram, namely the peak h_p, $h_{\mu R}$, $h_{\mu L}$, are then recalculated by

$$p = \text{trunc}((\mu R + \mu L) / 2) \tag{1}$$
$$\mu L = p \tag{2}$$
$$\mu R = p + 1 \tag{3}$$

In addition to the important histogram points, the total histogram area HF is calculated from the sum of all vector elements of the working histogram.

The value of the histogram vector at the point p is calculated with

$$h_p = (h_{\mu R} + h_{\mu L}) \cdot 0.5 \tag{4}$$

The area below the histogram to the left of p is calculated with

$$F_{uL} = h_p \cdot 0.5 \tag{5}$$

It is thus simulated that the pseudo-Gaussian histogram initially consists of a straight line at the level of half of the theoretical peak at the value p. For further calculations, the increases of the area fractions are calculated according to one side of the Gaussian distribution at 1σ - coming from h_p - with $F_1 = 0.34135 \cdot HF$. At 2σ the further increase in area is $F_2 = 0.13590 \cdot HF$, at 3σ correspondingly $F_3 = 0.02140 \cdot HF$ and at 4σ correspondingly $F_4 = 0.00135 \cdot HF$.

In a next step, 8 interpolation points (4 (L1 to L4) to the left of p and 4 (R1 to R4) to the right of p) are determined for the pseudo-Gaussian histogram. Starting from the point μL to the left, the working histogram values are added until the sum value reaches or exceeds F_1. This point is noted as the first interpolation point left = L1. In case the F_1 limit is not reached, L1 is assigned to the first number of the input histogram but positioned in the working histogram. In the second step, the procedure for determining L2 is carried out similarly, with the difference that now the area $F_1 + F_2$ is tested for. Since the original histogram is not Gaussian distributed, nothing can be predicted about the location of L1 to L2, so the following query is of great importance afterwards:

If L2 >= L1 then L2 is set to L1   (6)

This means that L2 can in any case not lie to the right of the grid point L1. However, the markers L1 and L2 can fall on the same work histogram element. The interpolation points L3 and L4 are determined in the same way. The test areas are increased by F_3 and then by F_4 and the test from equation 6 is of course always carried out with the grid point determined in the previous step. The determination of the interpolation points to the right of μR is done analogously, except that if the area is not reached, the position of the last element of the input histogram is noted in the working histogram as the interpolation point. Additionally the test in equation 6 is carried out with <=.

At the end of the interpolation point determination, an average distance on the left and right side of the histogram is calculated from the distances of the determined points (Eq. 7 & 8). Afterwards the interpolation points LN and RN are calculated (see Eq. 9 & 10).

$$DL = ((L1 - L2) + (L2 - L3) + (L3 - L4)) / 3 \qquad (7)$$
$$DR = ((R2 - R1) + (R3 - R2) + (R4 - R3)) / 3 \qquad (8)$$
$$LN = L4 - DL \qquad (9)$$
$$RN = R4 + DR \qquad (10)$$

Because of the extending of the input histogram to the working histogram, LN or RN may well lie within the range of the original data of the input histogram. They are then either set to the first digit before the range of the input histogram in the working histogram (left-hand side) or after the last digit of the input histogram in the working histogram (right-hand side). This later guarantees that no zero values appear within the pseudo-Gaussian distribution.

The values at the grid points should be based on the original histogram, but at the same time there is also the requirement that the new histogram should ultimately have a Gaussian distribution. This requirement was implemented by assuming for each interpolation point the histogram value that would result from the distance to the next grid point and the area sizes F_1 to F_4 according to Gauss. For example, for L1 the vector value is calculated from the distance from L1 to μL in relation to the area of Gaussian distribution F_1. The following calculations result in the same way for the other interpolation points:

$$h_{L1} = F_1 / (\mu L - L1) \tag{11}$$
$$h_{L2} = F_2 / (L1 - L2) \tag{12}$$
$$h_{L3} = F_3 / (L2 - L3) \tag{13}$$
$$h_{L4} = F_4 / (L3 - L4) \tag{14}$$
$$h_{LN} = 0 \tag{15}$$

In the case that the distance between the grid points is 0, i.e. that both grid points lie on the same histogram element, the respective area value without divisor is taken as the histogram value. This results in the histogram values h_{L1} to h_{L4}:

$$\text{if } (\mu L - L1) = 0 \quad \text{then} \quad h_{L1} = F_1 \tag{16}$$
$$\text{if } (L1 - L2) = 0 \quad \text{then} \quad h_{L2} = F_2 \tag{17}$$
$$\text{if } (L2 - L3) = 0 \quad \text{then} \quad h_{L3} = F_3 \tag{18}$$
$$\text{if } (L3 - L4) = 0 \quad \text{then} \quad h_{L4} = F_4 \tag{19}$$

Analogous equations result for the right-hand working histogram.

In this step, the values of the working histogram are calculated by interpolation between the neighboring interpolation points. In the case that two interpolation points occupy the same histogram point, no calculation is carried out and the previously calculated value is left. Otherwise the pseudo-Gaussian histogram results from the linear interpolations between the grid points (LN,0) and (L4,h_{L4}), (L4, h_{L4}) and (L3,h_{L3}), (L3,h_{L3}) and (L2,h_{L2}), (L2,h_{L2}) and (L1,h_{L1}), (L1,h_{L1}) and (R1,h_{R1}), (R1,h_{R1}) and (R2,h_{R2}), (R2,h_{R2}) and (R3, h_{R3}), (R3,h_{R3}) and (R4,h_{R4}), (R4,h_{R4}) and (RN,0). Note that both LN and RN are definitely outside the original input histogram at any points in the working histogram.

Finally, the middle third of the working histogram is extracted and the vector values are normalized to a total length of 1. This histogram then represents the relative pseudo-Gaussian histogram with which further calculations can be carried out. These steps of the procedure were implemented under Halcon in H-Develop and can be downloaded as source code under [Ra22].

Section 3: The classification of the data sets is done by simple difference formation of the vectors involved (or histograms of the color channels). For each channel, the Hamming distance between the test histogram and every of the reference histograms is first determined individually. Since these are relative histograms, the maximum distance (S_{max}) between two histograms can be 2. If several histograms are considered, the individual distances of each histogram are simply added. If three color channels are taken into account, S_{max} is 6 and if there is perfect agreement, S_{real} will be 0. If S_{real} is the distance between the histograms resulting from the calculations, the numerical agreement value U* results in

$$U^* = (1-(S_{real}/S_{max})) \cdot 100 \ (\%) \tag{20}$$

The image that achieves the highest U* value from the reference data set can be considered the best match. Optionally, pseudo-Gaussian histograms and rel. original histograms can be used together in the algorithm for the classification process. Both histogram types are treated equally in the classification process. In this case, a maximum distance value of 2 times 2 = 4 results per color channel. For example, with three color channels, a S_{max} of 6 times 2 = 12 is possible. The principle of the evaluation process always remains the same, regardless of how many or which color channels from which color space are entered

into the classification algorithm. In the same way, the image data can be compressed to a low color or pixel resolution before analysis.

Figure 1 demonstrates the effect of the Blue Apple algorithm by using a simple example, created manually with a computer. On the top of the figure the images A to C are given as the reference data set. The Image D is the test image which should be classified into one of A, B or C. The original histograms of the red-channels (of RGB) of A, B and C are shown in the bottom left part of the figure 1. One sees immediately that with our U*-calculation (Eq. 20) no one of the A, B, C images could be chosen as a result, because the U*-values are always 0. Converting the histograms into a pseudo-Gaussian distribution leads directly to the result that Image D is closest to C. Of course, this theoretical example is very extreme, but it is only meant to explain how the Blue Apple algorithm works. In the next chapter, concrete applications will demonstrate the applicability in complex situations.
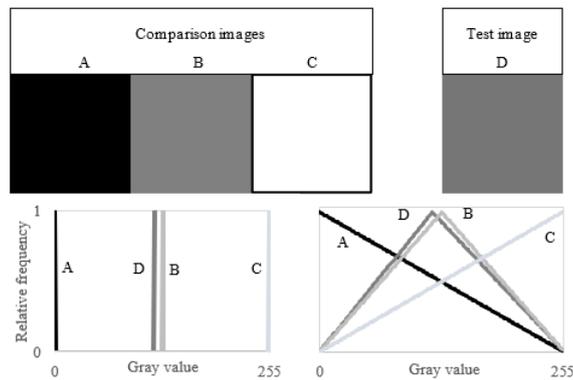
Fig. 1: Example for using the Blue Apple algorithm by means of 4 manually created easy images (top) and their responding histograms (bottom left: original, bottom right: pseudo-Gaussian distribution after conversion). For more explanations see text.

3 Evaluation & Examples

3.1 N-Analysis Tomato Leaves

With this experiment we tested the practical use of the Blue Apple algorithm in a specific plant based task like plant nutrition.

Solanum lycopersicum 'Hoffmanns Rentita' were sown in week 14/20 in sand and they were transplanted in week 18/20 in nutrient solution Ferty Basis 1 plus calcium nitrate, EC 1.5. The experimental variants were 0.5, 1.0, 2.0, 4.0, 8.0 and 16 mmol NO_3 or NH_4 in the nutrient solution. Cultivation was done in 2 L vessels with 3 plants per pot. For the studies, 3 leaves were taken from each N stage from different plants in the middle area. Leaves were photographed under standardized conditions in artificial light using a Go-Pro 8 camera (Go-Pro Inc., San Mateo, USA) with the following settings: photo / fov linear / superphoto off / protune on / shutter 1/125 ......(Auto) / iso min 100 /iso max

3200 / ev comp 0 / white balance 2800 K / sharpness high / color flat / no magnification. The total nitrogen content was determined in the dry mass by dry combustion (DIN EN 16168). The image files were prepared by selecting the green content with the image processing software Halcon. The code can be downloaded under [Re22].

Results and discussion example 1

An increasing nitrogen content was observed in the leaf samples corresponding to the increase in concentration in the nutrient solution, which was also visually recognizable by a more intense green coloration. (see Figure 2)
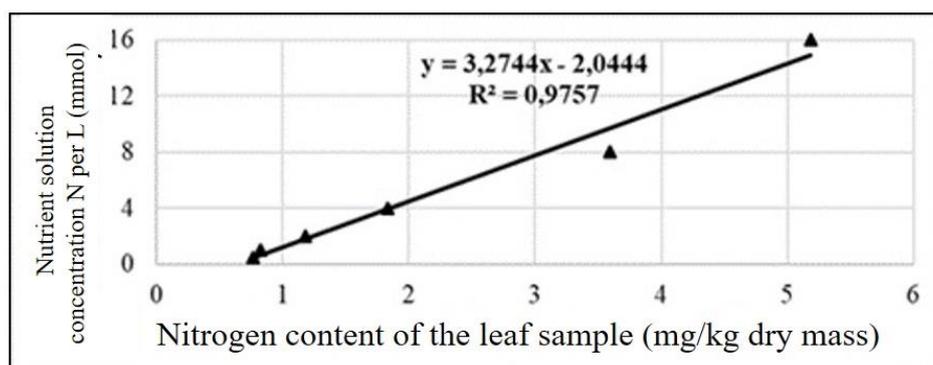
Fig. 2: Increasing nitrogen content in leaf samples of *Solanum lycopersicum* 'Hoffmanns Rentita' as a function of nitrogen concentration in the nutrient solution. The triangle dots are the values of 6 reference images.

The images (Fig. 2 and see for examples Figure 3) of the gradient experiment were analyzed with the Blue Apple methods in different color spaces which results in the reference data sets. The same procedure was done with randomly taken test samples. These were also photographed under the standardized conditions and the images were subsequently analyzed with the Blue Apple method. After digital capture, the leaves were dried and then the nitrogen content was determined.

standard 1   standard 6   sample 1
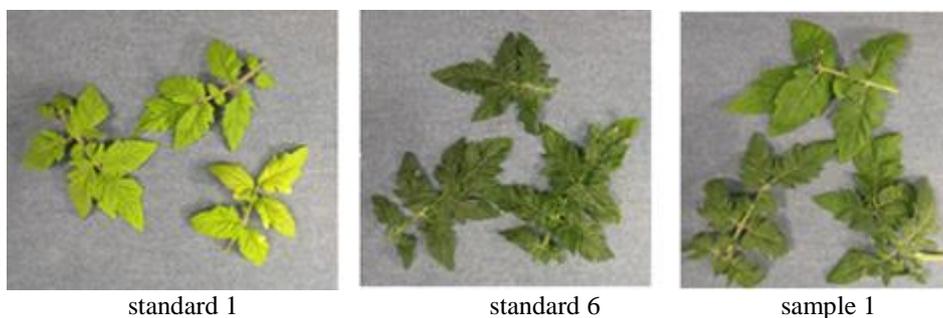
Fig. 3: Reference leaves (standard 1 and standard 6) and test sample 1

RGB color space		Nutrient Solution	Test Sample 1	Test Sample 2	N-content measured reference data	N-content measured test samples
		mmol N/L	Match with reference		mg N/kg dry mass	mg N/kg dry mass
			%	%		
Image 1	Standard 1	0,5	48,2	73,4	0,76	
Image 2	Standard 2	1,0	45,2	69,7	0,82	
Image 3	Standard 3	2,0	63,7	**80,1**	**1,18**	**1,21**
Image 4	Standard 4	4,0	83,2	71,7	1,83	
Image 5	Standard 5	8,0	**87,9**	57,9	**3,59**	**2,71**
Image 6	Standard 6	16,0	77,7	53,1	5,18	

Tab. 1: Matching results of two test images comparing with the reference images in the RGB color space and relating N-content and measured N-content of best matches

Image analysis using the Blue Apple program demonstrated a sufficient correlation between leaf coloration and nitrogen content. The best agreement was provided by the analysis in the RGB color space. The data for other color spaces are not shown.

3.2 Starch-Analysis Apples

In the second example we wanted to validate the Blue Apple algorithm in complex outdoor situations with different lightning conditions. To do this we choose an example in the field of fruit analyses.

A total of 42 apples were collected on October 5, 2020 at the Osnabrück University of Applied Sciences. Twelve apples of the 'Elstar' variety, nine apples of the 'Topaz' variety, seven apples of the 'Jona Gold' variety and 14 apples of the 'Braeburn' variety were selected. The apples of the 'Elstar', 'Topaz' and 'Jona Gold' varieties were taken from a ULO (Ultra-Low-Oxygen) store (harvest season 2019) and the 'Braeburn' variety was harvested unripe from an apple tree.

Subsequently, the starch-iodine index of 28 apples (nine apples each of the varieties 'Elstar' and 'Braeburn' and five apples each of the varieties 'Topaz' and 'Jona Gold') was determined in the laboratory using the ART-LB light box 3.0 (UP GmbH, Ibbenbüren, Germany). For this purpose, the apples were cut in half and the cut surface of one half of the apple was immersed in a Petri dish containing iodine solution. The apple halves were set aside with cut side facing up and after 15 minutes excess iodine solution was dabbed off. The starch-iodine index of the colored apple halves was determined using the ART-LB light box 3.0. Subsequently, the apple halves were photographed individually on a black background under a cloudy sky and under standardized lighting conditions in the laboratory using a mobile phone (HTC, One M9, Taoyuan, Taiwan). Photographs were taken with automatic white balance and exposure time. The remaining 14 apples were stored at room temperature for four weeks and on November 2, 2020, the starch-iodine index was determined as described above and the colored apple halves were photographed. The total sample size was 42 apples and 84 image files.

By pre-processing the image files with the Halcon image processing software, the apple halves were isolated from the black background and the core was removed with specific algorithms from the image (see following chapter).

The pre-treated image files were evenly divided into a test data set and a reference data set. 21 image files were assigned to both the test and the reference data set. Each image file in the test data set was compared to all image files in the reference data set and the image file with the highest match was selected from the reference data set. Only image files photographed under the same lighting conditions were compared.

Results and discussion example 2

Figure 4 shows the individual steps of image processing pre-treatment of the image files. In the untreated image, the half of the apple is selected and isolated from the black background. The image is reduced to the region of the apple half. Then the core housing is automatically recognized and removed from the image file. For more information about the Halcon code developed for this see [Cr22]. The image files pre-treated in this way can be processed with the Blue Apple algorithm.

Fig. 4: Pre-treatment of the image files using the example of an apple of the 'Braeburn' variety. Left: Untreated image file of the apple on a black background. Middle: Isolation of the apple from the background. Right: Core removal from the image file

Figure 5 shows the result of the manual assignment of the starch-iodine index to the actual starch-iodine index. The coefficient of determination achieved in this way is the best possible theoretical result that the Blue Apple algorithm can achieve in this experiment.
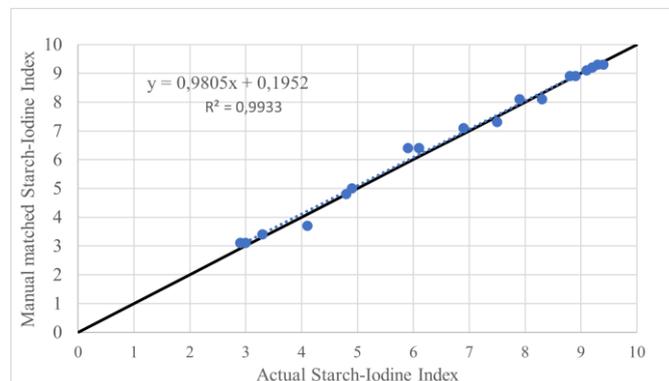
Fig. 5: Result of the manual assignment of the starch-iodine index of the test images.

Figure 6 shows the results of assigning the starch-iodine index of the test images by the Blue Apple algorithm when comparing the unmodified histograms.
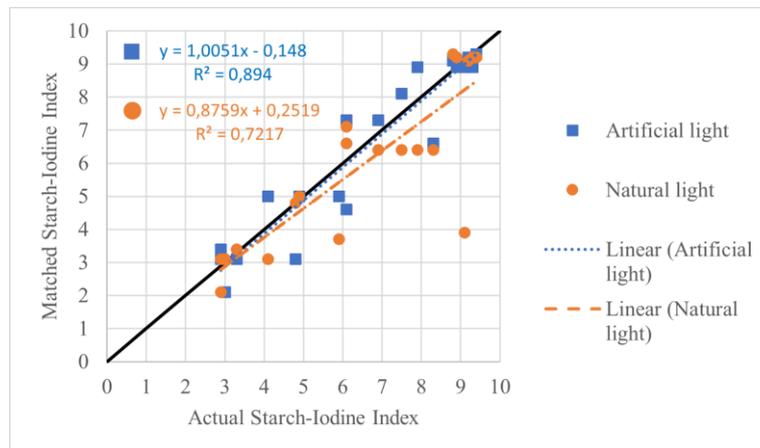
Fig. 6: Results of the assignment of the starch-iodine index by the Blue Apple algorithm when comparing the unchanged histograms of test and reference images under different lighting conditions

Figure 7 shows the results of assigning the starch-iodine index of the test images by the Blue Apple algorithm using the pseudo-Gaussian histograms.
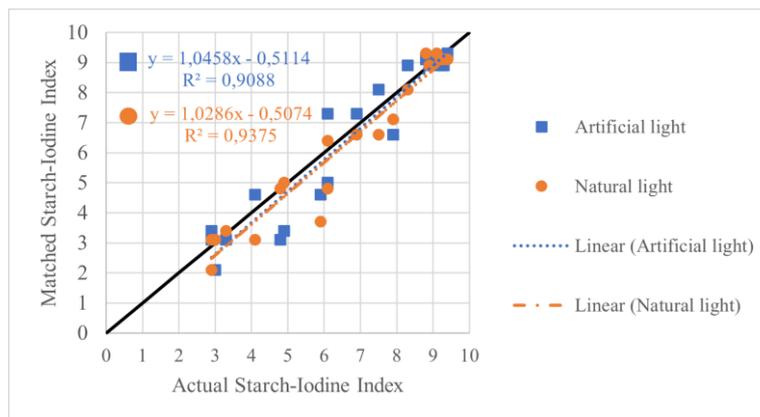
Fig. 7: Results of the assignment of the starch-iodine index by the Blue Apple algorithm when comparing the pseudo-Gaussian histograms of the test and reference images under different lighting conditions

Figure 8 shows the results of assigning the starch-iodine index of the test images by the Blue Apple algorithm using unmodified and pseudo-Gaussian histograms together.
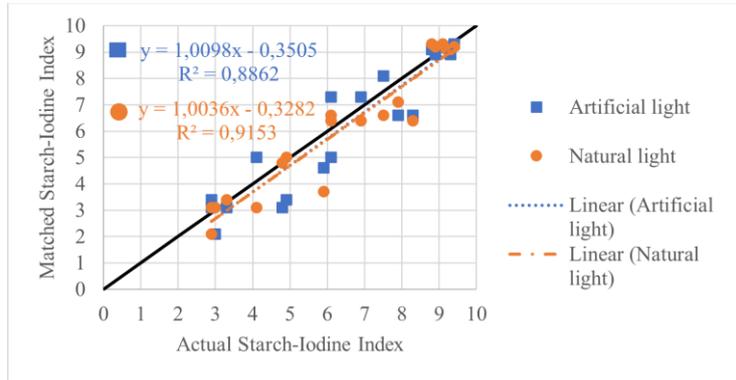
Fig. 8: Results of the assignment of the starch-iodine index by the Blue Apple algorithm in the combined comparison of the pseudo-Gaussian and unmodified histograms of the test and the reference image files under different lightning conditions.

Tab. 2 shows the summarized results of all comparisons performed by the Blue Apple algorithm and the manual index assignment.

Test variant	Light condition	Difference between actual and matched starch-iodine index	
		Mean deviation	Standard deviation
Pseudo-Gauss. histogram	Artificial light	0.63	0.51
	Natural light	0.46	0.53
Original histogram	Artificial light	0.61	0.55
	Natural light	0.83	1.15
Pseudo-Gauss. and original histogram	Artificial light	0.69	0.57
	Natural light	0.52	0.57
Manual	-	0.15	0.14

Tab. 2: Summarized results of the comparison of test and comparison images with different test variants and lighting conditions. Values rounded to two decimals.

It could be shown that the Blue Apple algorithm is basically able to match an apple to a reference apple with similar starch-iodine index. The mean deviation from the measured starch-iodine index was 0.83 with a standard error of 1.15 in the worst case and 0.46 with a standard error of 0.53 in the best case. The natural light experiments had the clearest effects, while the artificial light experiments gave different results. Furthermore, looking at the level of index fluctuations, the result of the assignment could be improved when using the pseudo-Gaussian histogram compared to using the unmodified histogram under both standardized and natural lighting conditions. The coefficient of determination increased from 0.72 to 0.94 under natural light conditions and from 0.89 to 0.91 under standardized light conditions.

4   Discussion and Outlook

The newly developed Blue-Apple algorithm was designed for a better image processing in RGB or similar color spaces in the field of agriculture and horticulture. In the case of

using manually generated extreme color channel histograms, we were able to show that the algorithm achieves a clear improvement in classification tasks. Its use in biological and agronomic practice was successfully confirmed by the example of nitrogen determination of plant images. The comparison with conventional histogram evaluation based on the starch determination of apples showed that classification improve with Blue-Apple even under diverse boundary conditions (degree of ripeness, apple variety). The algorithm performed particularly well under difficult natural lighting conditions. Blue-Apple is knowledge based and statistical data driven. Therefore, conversion of grey-scale histograms into so-called pseudo-Gaussian histograms and thus the quality of Blue-Apple's success always depends on the distribution of the histograms to be evaluated. Nevertheless, or rather precisely for this reason, the use of the method appears to be promising, especially for color classifications in the field of plant cultivation. In addition to histogram evaluations in image processing, the use of Blue-Apple is also conceivable for processing other types of sensor data (e.g. fluorescence data or laser distance data). Likewise, one could use it as a pre-processing step of deep learning methods. Blue-Apple thus represents a viable and promising extension of evaluation tools in biological and agricultural research and practice.

5   Literature and Supplementals

[Cr22]   Credner, J., Fadami, M., Rehrmann, P., Rath, T.: Blue Apple – Isolation and removal of apple core housings in color images. BLab Code No. 20222, http://www.blab-osna-brueck.de/blue-apple2.txt, 2022.

[HCR18]   Hameed, K., Chai, D., Rassau, A.: A comprehensive review of fruit and vegetable classification techniques. Image and Vision Computing, 80, 24-44, 2018.

[HLE18]   Hassanein, M., Lari, Z., El-Sheimy, N.: A new vegetation segmentation approach for cropped fields based on threshold detection from hue histograms. Sensors, 18(4), 1253, 2018.

[JTZ11]   Jeon, H. Y., Tian, L. F., Zhu, H.: Robust crop and weed segmentation under uncontrolled outdoor illumination. Sensors, 11(6), 6270-6283, 2011.

[NAA21]   Ngugi, L. C., Abelwahab, M., Abo-Zahhad, M.: Recent advances in image processing techniques for automated leaf pest and disease recognition–A review. Information processing in agriculture, 8(1), 27-51, 2021.

[PR 18]   Patrício, D. I., Rieder, R.: Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. Computers and electronics in agriculture, 153, 69-81, 2018.

[Ra17]   Ravikanth, L. et. al.: Extraction of spectral information from hyperspectral data and application of hyperspectral imaging for food and agricultural products. Food and bioprocess technology, 10(1), 1-33, 2017.

[Ra22]   Rath, T., Credner, J., Fadami, M., Rehrmann, P.: Blue Apple – Transformation of histograms to Gaussian distributions. BLab Code No. 20223, http://www.blab-osna-brueck.de/blue-apple3.txt, 2022.

[Re22]   Rehrmann, P., Credner, J., Fadami, M., Rath, T.: Blue Apple – Object green free. BLab Code No. 20221, http://www.blab-osnabrueck.de/blue-apple1.txt, 2022.