Euphytica (2024) 220:6
https://doi.org/10.1007/s10681-023-03258-2

RESEARCH

Determining flower colors from images using artificial intelligence

Ron Wehrens · Manya Afonso · Hubert Fonteijn · João Paulo · Gerrit Polder · Marcel Rijsbergen · Gerard van Hameren · Raoul Haegens · Mary van den Helder · Henk Zwinkels

Received: 12 April 2023 / Accepted: 7 November 2023 / Published online: 14 December 2023

Abstract Flower color is one of the most important traits in horticulture, and is one of the characteristics recorded to describe new varieties. In this paper, we examine four large real-world databases of roses and gerberas containing both images and color descriptions, and use state-of-the-art methods to automatically extract color descriptions from the images. Both Deep Learning and methods based on color histograms lead to success rates of approximately 85%. Deep learning has the advantage that no preprocessing is necessary—the more traditional methods lead to additional insight in the final color classification.

Keywords Plant variety testing · Ornamentals · Deep learning · Random forests · Color histograms

R. Wehrens (✉) · M. Afonso · H. Fonteijn · J. Paulo · G. Polder
Wageningen Plant Research, Wageningen University & Research, Wageningen, The Netherlands
e-mail: ron.wehrens@wur.nl

M. Afonso
e-mail: manya.afonso@wur.nl

J. Paulo
e-mail: joao.paulo@wur.nl

G. Polder
e-mail: gerrit.polder@wur.nl

M. Rijsbergen · G. van Hameren · R. Haegens
Naktuinbouw, Roelofarendsveen, The Netherlands
e-mail: m.rijsbergen@naktuinbouw.nl

G. van Hameren
e-mail: g.v.hameren@naktuinbouw.nl

R. Haegens
e-mail: r.haegens@naktuinbouw.nl

M. van den Helder · H. Zwinkels
Floricode, Roelofarendsveen, The Netherlands
e-mail: m.vandenhelder@floricode.nl

H. Zwinkels
e-mail: h.zwinkels@floricode.nl

Introduction

Flower color is one of the most important traits in floriculture, a sector in which The Netherlands play an important role with an annual export value of over 7 billion euros. Not only is it one of the characteristics based on which consumers often base their choice, it is also relevant in the evaluation of new varieties: these should be not too similar to existing varieties. One of the first steps in the admission procedure therefore is a consistent and thorough registration, leading to a description of a number of traits (among which color) that should uniquely define each variety. Trait descriptions such as these are also used in distinctness, uniformity and stability testing (DUS testing). In that context, for each flower species sets of traits have been defined by the International Union for the Protection of New Varieties of Plants (UPOV) in order to achieve worldwide consistency.

Trait characterisation is still manual and highly skilled labor. Even partial automation of the process, e.g., based on flower images taken under a strict protocol, can improve the registration process. The advantages include speeding up and simplifying procedures (e.g., by allowing breeders to perform part of the registration in house) and improving objectivity and consistency. In addition, automation would provide a solution for situations where it is hard to find or educate experts able to do the registration.

Automatic recognition for a variety of traits in roses and gerberas is described in Afonso et al. (2023). Here, the focus is on flower color: we investigate and compare ways to automatically obtain the main color from images taken under a standardized setup. Real-world data sets are considered, two for roses, and two for gerberas, and two different approaches are assessed, one based on Deep Learning (DL), the current state of the art in image processing, and one more traditional approach based on color histograms.

Background

Automatic phenotyping

High-throughput phenotyping currently relies heavily on imaging and image processing pipelines, often based on artificial intelligence (AI). Such an approach has, a.o., been used to recognise fruits (Afonso et al. 2020) and flowers (Afonso et al. 2019) in tomato greenhouses, but the way to routine application is not a smooth one (Fonteijn et al. 2021).

Also in the context of horticulture much work has been done. Zhenjiang et al. (2006) developed a rose variety recognition system using petal color as one of the features. Image analysis was applied to measure petal shape and picotee color patterns in Lisianthus (Yoshioka et al. 2006) and Primula (Yoshioka et al. 2004) flowers. Automatic flower classification on a large data set of similar classes was investigated by (Nilsback and Zisserman 2008) using color and shape features.

More recently, color recognition in flowers was reported for orchids (Apriyanti et al. 2021), where five different neural networks were compared in their ability to classify orchid images into a set of eleven predefined combinations of primary and secondary colors. The flower images were obtained from the web and contained a wide variety of backgrounds, and color labels (the ground truth) were obtained from separate orchid databases. Somewhat similar to our current setup, Wang et al. (2022) obtained pictures of 213 cultivars of large-flowered chrysanthemums, with at least 2–3 plants per cultivar. Top views were obtained, and images were manually annotated. Finally, deep learning was used to classify the images, where color was one of the categories (the others being flower type and petal type).

Color representation

Colors can be represented in a variety of ways. The simplest and most intuitive, but also the most coarse and least well defined, is the use of color names like red, yellow and pink. Their use is ubiquitous but limited – one cannot, e.g., calculate distances between colors. In addition, many colors exist for which different people would use different names. A straightforward extension is to define a (limited) number of color classes, e.g., by combining color names or defining combinations of primary and secondary colors (Apriyanti et al. 2021).

A much more precise definition of a color is to use coordinates in a color space. Several color spaces exist, of which red-green-blue (RGB) is the most familiar one – very often 256 (= 2^8) levels for each of the three base colors are used. Then, the RGB space would define 2^{24} colors, more than enough for most practical purposes. This RGB color space has one important drawback, however: distances do not always correspond well to human perception. That means that two colors that seem very similar to the human eye may have a relatively large distance in RGB space, and vice versa (Zeileis et al. 2009). Several other color spaces have been designed to remedy this, with varying degrees of success. Other well-known color spaces include Hue-Saturation-Value (HSV), a direct transformation of the RGB color space, and the CIELAB/CIELUV color spaces from the International Commission on Illumination (where CIELUV is used for light-emitting devices such as televisions or computer monitors, and CIELAB for passive colored surfaces). Here, we use (in addition to RGB space) the Hue-Color-Luminance (HCL) scale, a transformation into polar coordinates of the CIELUV color space, also known as polarLUV (Zeileis et al. 2009). This color space to a large degree succeeds in matching human perception: it is known to have some problems in the blue color region, but this is quite irrelevant for the current application given the absence of blue flowers in our data sets (and the rarity of blue flowers in general).

An intermediate level of granularity in the context of ornamentals is achieved by the colors defined by the color charts of the Royal Horticultural Society (RHS). These contain almost 1000 colors, each exactly defined by coordinates in RGB space. Each of the RHS codes is assigned a main color name (e.g., “orange”), and a subname (“light orange”). For the data used in this paper, an RHS color chart was compared to the flower (by physically overlaying the flower with the color chart) to get a visual match that is as close as possible.

The influence of color representation in assessing flower colors has been investigated in several publications. Lootens et al. (2007) assessed the tepal color of *Begonia x tuberhybrida Voss.* using traditional image analysis. They found that parameters based on the green color channel of the RGB color space resulted in the largest discriminating power, while parameters based on the HSV color space performed less well. Singh et al. (2011) performed image analysis on Gerbera images to quantify color variation based on mean values of the RGB components and hue and saturation (HS) color channels using high-quality images taken with prominent blooms from consistent angles and reliable lighting conditions.

Van der Heijden et al. (1999) proposed a database system for finding and retrieving rose varieties with similar appearance, using binarized color histograms. They showed that RGB-histograms and normalized RG-histograms outperformed several other color spaces. Comparisons between several color spaces in flower recognition has been conducted for twelve species from two families by Rosyani et al. (2018). In that paper, the HSV color space performed the worst amongst the investigated color spaces (using a Support Vector Machine classification approach). Perez-Udell et al. (2023) built an automated pipeline for supervised classification of petal color from public data repositories, using *k*-means clustering in the HSV color space.

All aforementioned papers make use of cameras which, similarly to the human visual system, use a red, green and blue channel. For precise matching of subtle color differences, having more and smaller wavelength bands in the visible spectrum often improves performance (Stokman et al. 2000). Van der Heijden et al. (2000) showed that normalized spectral images have a better discriminating power for classifying red roses than color-constant spectral images and RGB-images, even when the latter had been recorded under highly optimized standard conditions.

Materials and methods

Data

This paper concentrates on flower color in four real-world data sets that have been recorded over the course of several years. The data consist of rose and gerbera images from two private companies, Floricode (www.floricode.com) and Naktuinbouw (www.naktuinbouw.nl). Floricode is doing variety registration, e.g., for use in auctions, whereas Naktuinbouw uses the registration for purposes such as plant breeders’ rights protection – the two companies therefore register different (numbers of) traits and also use different trait levels. In all cases, however, the main flower color is present, either as a color name (all cases) or, additionally, as an RHS code (all cases except for Naktuinbouw Rose). This paper addresses possibilities to automatically obtain the name of the main flower color from the images. An overview of the four data sets is shown in Table 1. Although the number of flowers in each data set seems quite large, it should be noted that for many machine-learning approaches, and in particular for deep learning such as applied here, much larger data sets are often needed, depending on the specific questions asked and the nature of the images.

The Floricode images had a much higher resolution than the Naktuinbouw images, and therefore were reduced in size in order to speed up calculations.

Table 1 Characteristics of the four data sets used in this paper

Provenance	Flower	# Images	# Colors	Avg. image size (MB)
Floricode	Gerbera	1449	6	0.2
Floricode	Rose	1641	7	0.2
Naktuinbouw	Gerbera	570	6	0.7
Naktuinbouw	Rose	768	6	0.3

Since small details are irrelevant in this particular application, this does not lead to any information loss, as was checked in the early stages of this research. Colors were coded as RGB using 256 intensity levels for each channel. Rare colors with fewer than 50 cases in the data sets, such as green and brown, were removed since such classes are virtually impossible to learn in an example-based approach like deep learning, and leaving those colors in would lead to a biased comparison. The colors in the Naktuinbouw Rose set originally contained eleven distinct colors, but compound colors (like orange blend or yellow orange) were reduced to their main color names (here: orange), leading to six colors overall.

Image preprocessing

In all cases, the images have been recorded using pre-specified and consistent protocols (different for Floricode and Naktuinbouw, however). A couple of example images from each of the four data sets are shown in Fig 1. Immediately we see some differences between the data sets: the Naktuinbouw images (second and fourth rows) show flowers only, whereas the Floricode images show other features of the plant as well—stems in the case of gerberas (top row), and stems and leaves for roses (third row). Note that the image background may contribute to errors in the color estimation procedure: a uniform white background, e.g., may lead to overestimation of the number of white flower pixels, thereby increasing the risk of wrongly classifying a flower as white. To avoid this, in a preprocessing step the image is segmented into foreground (flower) and background, and all following steps will consider the foreground pixels only.

In both Floricode data sets, the background is white and relatively uniform. Thus, unless the roses or gerberas are white, the background can be segmented by applying a threshold specific to all three color channels, above which a pixel is set as background. Given the large difference between the background and most flowers, choosing the threshold is not particularly difficult or sensitive—many different values will give similar results. For white flowers, a separate approach is followed in which the foreground is segmented by a combination of color thresholding and edge detection, in combination with connected components (Bovik 2010).

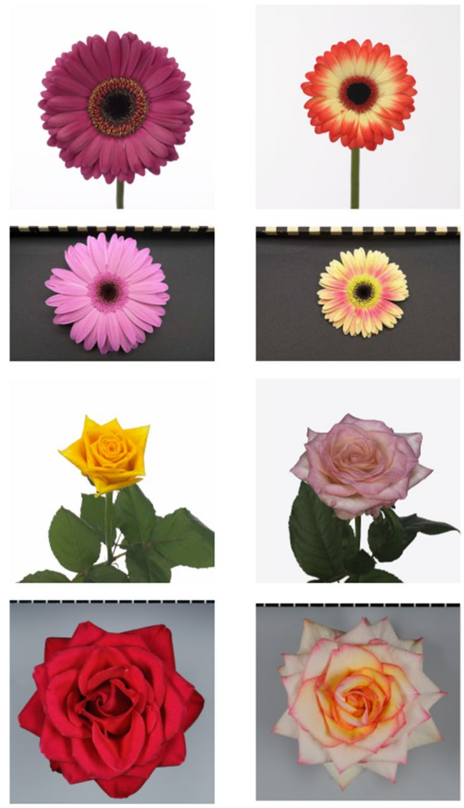


Fig. 1 Examples of raw images from the four data sets. Top to bottom: Floricode gerbera, Naktuinbouw gerbera, Floricode rose, and Naktuinbouw rose

In the Naktuinbouw data sets the background is not uniform in color, and a color- or edge-based approach is not sufficient. In these cases, the foreground has been segmented using the background remover app (https://github.com/nadermx/backgroundremover) which applies a deep learning model, based on the U^2-Net architecture and trained on general-purpose data sets with semantic foreground/background ground truth data (Qin et al. 2020). Some images in the Naktuinbouw data sets contain top as well as bottom views of two different flower samples of the same variety. In these cases only the top view is selected (manually, through visual inspection). In a few cases where the two flower samples in the image have no gap between them, the segmentation is corrected manually.

The Floricode rose data set is different from the other three in that the whole top of the plant is present in the image, including many of the leaves. For this particular data set, we have included a pre-processing step in which large areas of green were removed, by applying a threshold on the term G - 0.5 (R + B), which emphasizes the color green. This was done using the $I_1I_2I_3$ colorspace transform in Halcon. The results of applying these pre-processing steps on the images in Fig. 1 can be seen in Fig. 2, third row. The outline of the plant is still visible, due to the simple nature of our plant removal algorithm, but consists of too few pixels to influence the results. Similarly, the small parts of the stems that are visible in the Floricode gerbera images do not affect the results and have not been removed.

Color recognition

Recognising colors in images is a notoriously hard problem, even though at the same time it seems almost trivial. There are several reasons for this: first of all, the lighting in the image is of considerable influence. A white surface can be made to resemble any color, simply by changing the color of the light. Obviously, all other colors are affected by this, too. Secondly, our perception of colors is complicated and influenced by many characteristics other than just the wavelength intensities of the light reaching our eyes: surrounding colors have an effect, as does the nature of the material, to name but two (Bosten 2022; Emery et al. 2017a; Emery et al. 2017b). Thirdly, reporting colors in an understandable yet comprehensive way is not trivial: apart from the inherent subjectivity involved in the process, different color naming systems are in use (such as the RHS color charts and the UPOV color descriptions) that are not completely compatible.

Whereas these aspects are general and apply to any object, there are also considerations that are particular to the flowers that we consider here: the color of a flower is not uniform, but different gradients are present in, e.g., the petals. This also raises the question of what color feature is being determined: the overall (apparent) color of the flower, or the color of different flower parts (such as petal tips). The more local the color feature is, the easier it is to objectively determine color by using for instance a colorimeter (van Eck and de Vries 1995). However, a more global color feature such as the main flower color will remain a mixture of several local features and will be difficult to obtain objectively.

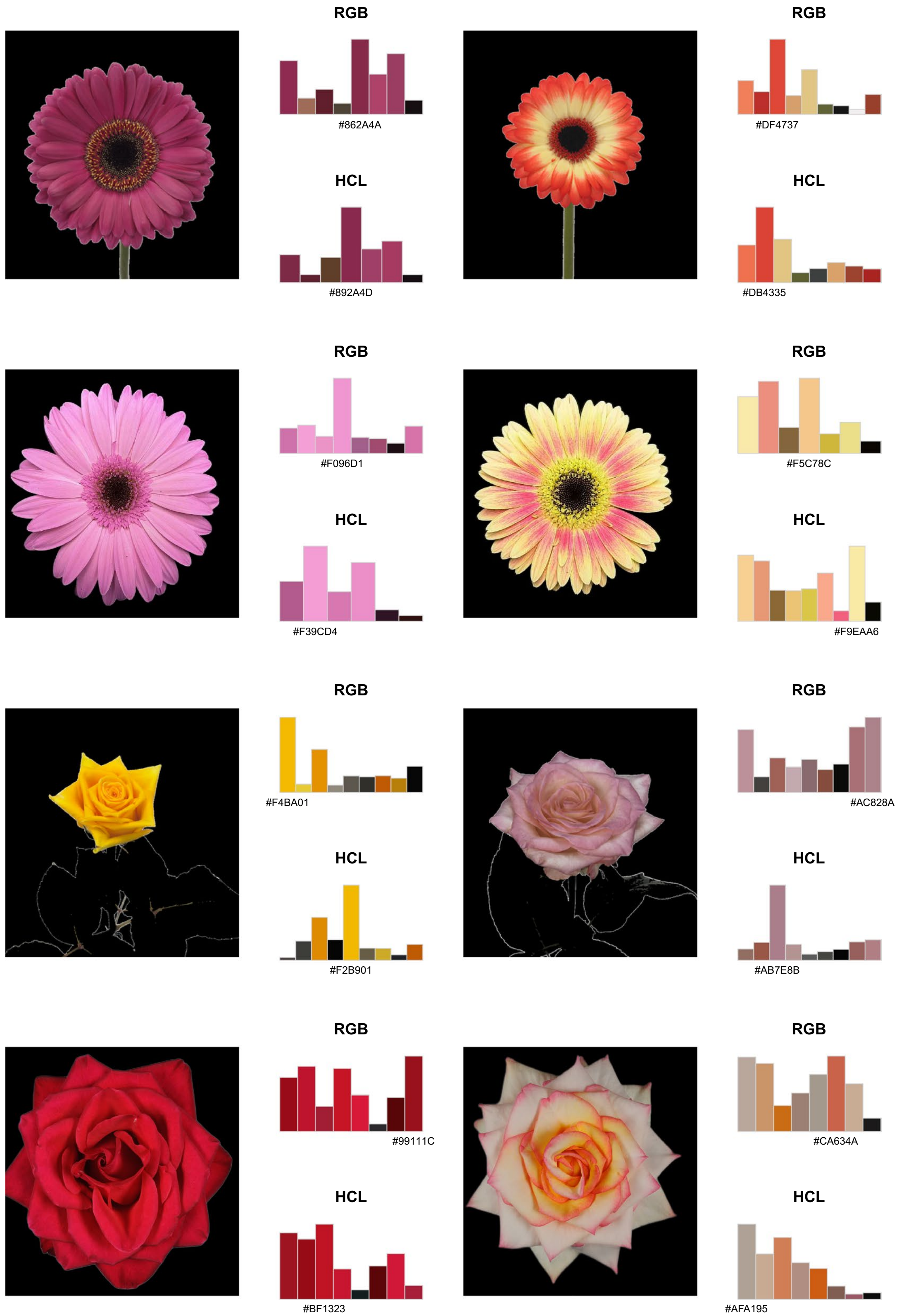
In this work we use and compare two different approaches to recognising colors. The first is the machine-learning approach implemented through deep learning, casting the task into a classification framework. This basically constitutes a black box in which the individual images are linked to the associated color labels, based on a training process using as many examples as possible. The second, more conventional approach is based on grouping pixels together on the basis of their color, leading to so-called color histograms. Then, one of these groups is chosen as the main color, usually based on the number of pixels in each group. The final step is then to convert this color, identified by coordinates in a color space, to a color name.

Each of the four data sets was randomly divided in training and test sets in the ratio of 2/3—as usual, the training data are used to obtain a model, and the test data allow us to obtain a measure of accuracy of the model predictions. In all approaches, the same divisions in training and test data were used to allow unbiased comparisons. It should be noted that the color histogram approach using the color translation table does not use a training set per se; it simply applies the predefined strategy to all images in the training and test sets. For purposes of comparison, only the results of the test sets are reported here – their characteristics are summarized in Table 2.

Note that in principle it would be possible to aggregate the four data sets into, e.g., two sets, one for rose and one for gerbera, or even one large set. This would increase the number of examples, and would make application of especially deep-learning approaches easier. Here we have opted for a separate analysis for each data set: roses and gerberas are different, and flower-specific features may be picked up by our methods leading to improved performance. In addition, the data sets have been set up using different

Table 2 Test set characteristics for the four data sets. In the column "Class size range" the numbers of images in the smallest and largest color classes, respectively, are shown

Provenance	Flower	# Images	# Colors	Class size range
Floricode	Gerbera	450	5	45–149
Floricode	Rose	529	6	22–220
Naktuinbouw	Gerbera	144	5	11–70
Naktuinbouw	Rose	249	5	36–62


RGB
#862A4A
HCL
#892A4D
RGB
#DF4737
HCL
#DB4335
RGB
#F096D1
HCL
#F39CD4
RGB
#F5C78C
HCL
#F9EAA6
RGB
#F4BA01
HCL
#F2B901
RGB
#AC828A
HCL
#AB7E8B
RGB
#99111C
HCL
#BF1323
RGB
#CA634A
HCL
#AFA195

◂**Fig. 2** Examples of color histograms obtained in RGB and HCL space. Top to bottom: Floricode gerbera, Naktuinbouw gerbera, Floricode rose, and Naktuinbouw rose. For each data set, two examples are shown. The color selected as the main color is indicated below the corresponding color bar in hexadecimal RGB notation. Background pixels, determined during preprocessing the images, are shown in black

protocols with slightly different aims, and use different color names—although a consensus color definition may be possible it is not easy to create it automatically. This would add an additional uncertainty to the classification task. As a consequence, we have chosen to analyse each set separately.

Deep Learning

The problem of predicting the color using deep learning is a multiple class classification problem, with the labels being the colors. The neural network architecture therefore used is a standard 18-layer ResNet (He et al. 2016) with the final softmax layer having as many outputs as colors to be predicted. ResNet18, while significantly deeper than architectures such as VGG (Simonyan and Zisserman 2015), can work with fewer labelled images than ResNets with more layers, which in this case with relatively low numbers of examples is a definite advantage.

A fixed number of training epochs was used to avoid the need for a validation set: such a set is often used to determine when to stop training. Given the relative scarcity of the data it was felt that results would improve by making the training set as large as possible, while keeping the size of the test set large enough to have meaningful estimates of error rates even in cases where categories were sparsely populated.

Color histograms

In the color-histogram approach we first cluster all pixels in RGB or HCL space, aiming to obtain a low number of groups, where each group corresponds to one color or color shade. As a clustering method, we applied model-based clustering (Fraley and Raftery 2002), which has the advantage of automatically suggesting an optimal number of clusters. Each cluster is characterised by multivariate normal distribution, defined by a center and a covariance matrix describing the shape of the cluster. To speed up the fitting procedure, we applied the clustering on a random sample of 1000 pixels, a simpler version of the strategies suggested in Wehrens et al. (2004), which is appropriate here because we are only interested in the major cluster(s), and not in a correct grouping of smaller clusters. We have found that in some cases it is beneficial to further merge clusters. Since clustering involves an element of randomness repeated clusterings may lead to somewhat different results. It may happen, e.g., that a particular color is described with two or more groups instead of one, and in such case choosing the main color on the basis of the number of pixels in each individual group would lead to a wrong selection. Grouping very similar colors using a single-linkage criterion should avoid this. The implementation is based on the Bhattacharyya distance between normal distributions, using 1.5 as the cutoff for merging clusters. Once this grouping has been achieved, the center of the largest cluster from the largest group is picked as the main color.

This procedure leads to an estimate of the main flower color, represented as an RGB or HCL value, depending on the color space in which the calculations have been performed. To back-transform this into a color name, one possibility is to use a translation table linking RHS codes to RGB triplets. If our clusters are obtained in HCL space, the back-transformation uses HCL values, calculated from the RGB values in the lookup table, in order to determine the closest RHS code. Finally, the color associated with the closest RHS code is the color name that we are after.

As an alternative to the approach based on a lookup table, a machine-learning approach to obtain color names, based on Random Forests (Breiman 2001), was implemented as well. In all instances random forest default settings were used. This approach is using the same training/test set divisions as the deep learning approach, so that results are directly comparable between these two techniques—note that, again, reported results pertain to the test data only.

Software

Background removal for the images was implemented using a combination of in-house developed Python code and the MVTec Halcon software (Halcon User Guide 2022). The deep learning methods were implemented in Python/Jupyter, using the

PyTorch framework (www.pytorch.org) (Paszke et al. 2019). All calculations regarding the color histogram approaches were executed in R version 4.2.1 (R Core Team 2022), making heavy use of the packages **randomForest** (Liaw and Wiener 2002), **colorspace** (Zeileis et al. 2020) and **mclust** (Scrucca et al. 2016).

Results

The deep learning approach is, as already stated, a black box, and not much can be shown except for the final results, later in this paragraph. The color histograms, however, can lead to interesting observations, especially in cases where predictions are not what one would expect. A few examples, corresponding to the raw images in Fig. 1, are shown in Fig. 2. The color histograms next to each flower image indicate which colors have been detected. The "main" color is identified with a hexadecimal RGB code under the selected color bar. In general, RGB and HCL spaces lead to very similar results. Sometimes they differ in the number of classes, but usually very similar colors are picked in the two approaches. As discussed earlier, differences between results of the two color spaces can occur because distances between colors in RGB space are different from distances in HCL space: this, in turn, influences the clustering of the pixels, and the subsequent grouping of the main cluster colors. A case in point is the second Naktuinbouw rose example, bottom right in Fig. 2. The RGB analysis picks a dark orange color as the main color (corresponding to the petal tips), whereas the HCL approach selects a shade of white, more in line with the central petal areas. In this example, the HCL result is the correct one; there are also cases where RGB leads to better results. It is very hard to define an optimal strategy because of the enormous variety in color distribution across flowers and, indeed, across petals or locations within petals of a flower.

Summary results for the recognition of colors in the four data sets for all analysis strategies are shown in Fig. 3. The interpretation is quite clear. The machine-learning approaches (deep learning, indicated with DL, and the color histogram approaches using random forests to obtain color names) show the lowest errors, for each data set achieving more or less similar results. For Naktuinbouw Rose DL is slightly better; for the other three data sets the random forest is better. The overall best error rate is 13.8%, again indicating that the task of recognising colors is by no means trivial.

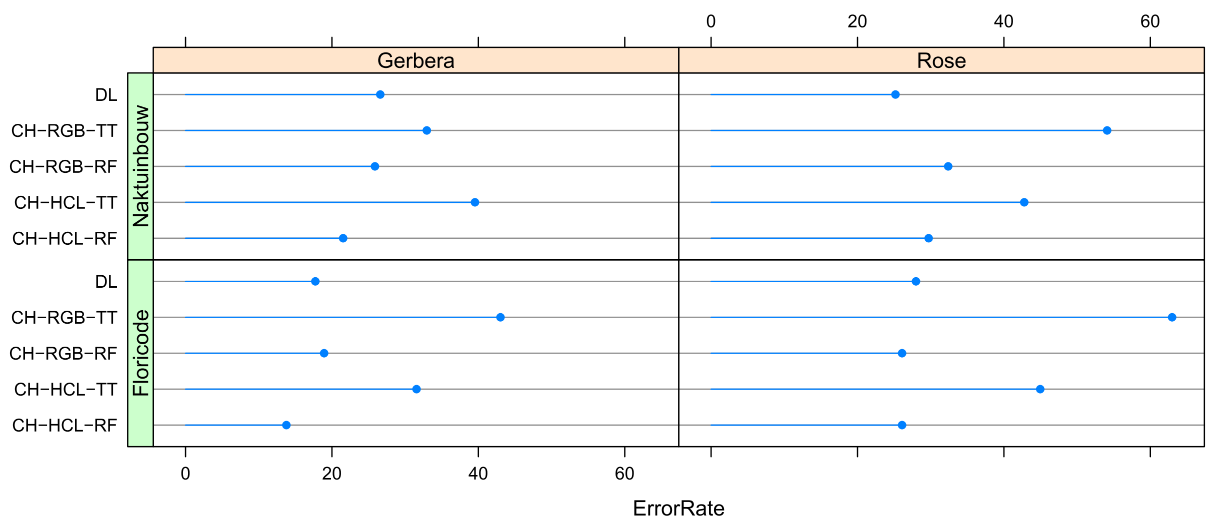


Fig. 3 Error rates (percentages) for color name recognition in the four data sets, using five different analysis methods. DL indicates deep learning, methods starting with CH indicate color-histogram-based methods. TT indicates that the translation table has been used to convert color space coordinates to color names, and RF indicates that this has been achieved with Random Forests. Finally, RGB and HCL refer to the color spaces employed in identifying the main color and obtaining the corresponding color name through the color-histogram-based approach

Of particular interest is the Floricode rose set – the results here, as stated before, are generated after having removed leaves and stems from the images. Deep learning, however, has been shown in many applications to be able to ignore irrelevant information in images. Indeed, Deep Learning applied to the original Floricode rose images, with background removed, as before, but including stems and leaves, leads to an error rate of 27.5%, virtually the same as the error rate shown in Fig. 3 (28.0%). This is a significant advantage, since it decreases the preprocessing effort drastically. Indeed, even removing the background is not necessary: Deep Learning will lead to virtually identical results when applied to the original, unprocessed images. Also for other traits than color we have seen that removing the background is not necessary to achieve good results using deep learning (Afonso et al. 2023).

The methods using the color translation table for the conversion of color coordinates to color names (instead of the random forest) show consistently worse error rates, typically by 10–20%. This is somewhat surprising, since it indicates that there are cases where the color names and the coordinates in color space are not quite matching. However, it should be noted that the six or seven main color names each cover a large part of the color space, and some RHS codes will of course be close to the boundary. In fact, training a random forest model on a random subset of two thirds of the 969 RHS codes and predicting the main color names of the remainder we obtain a success rate of only 85%, similar to the best results here. One possible reason is that the boundaries between the main colors are not smooth functions of the coordinates in color space; another, perhaps more credible, is that the data contain biases (perhaps at the level of the images) that both random forests and deep learning are able to learn in the training process.

Finally, it is interesting to assess the effect of the color space (relevant only in the methods relying on color histograms). When random forests are used to convert color coordinates to color names, the effect of using a different color space are consistent but limited, with HCL having an advantage of 0–5% over RGB. For the translation table approach, however, the differences are much larger. An overview of the confusion matrices obtained by the color translation table approach is given in Fig. 4, gerbera data sets only. The diagonals in each plot indicate the numbers of correct classifications. Looking at, e.g., the top left panel, Floricode Gerbera RGB, we see that all red flowers are indeed categorized as red, and that some yellow flowers have been given the label orange. The Floricode results show that white and yellow lead to substantially improved results when using the HCL color space, orange and pink show a slight improvement with HCL, purple shows equal performance for both color spaces and red shows the highest performance in the RGB color space. The HCL color space shows a tendency to classify red flowers as pink. The same effects are observed in the Naktuinbouw data, with the exception of orange, which now shows better performance in RGB space. Overall, when using the color translation table, for three out of four data sets the HCL color space leads to superior performance, improving upon RGB by 12–18%. The Naktuinbouw Gerbera data set is different, showing an advantage of almost 7% for the RGB space. This difference in performance patterns for the Naktuinbouw Gerbera data set can be explained by the relatively high proportion of red flowers in the data set: RGB simply does better than HCL for red flowers. These results also indicate that recognising the main color and finding the right name for it are two separate sides of the coin, the latter being non-trivial for a straightforward approach based on look-up tables.

A final remark on computation times: these are of course highly dependent on software and hardware, and no attempts have been made to optimize our pipelines in that respect. Nevertheless, some rough figures may be useful information. Whereas the training of a deep learning neural network can take quite a lot of time (in our case using a NVIDIA Titan XP GPU, a couple of hours of training for each data set), subsequent predictions for new (or old) images are basically instantaneous. In contrast, processing an individual image using the color histogram approach (on a regular desktop) takes a minute in our pipeline. Almost all time is spent in identifying the main color (in RGB or HCL space)—translation to a color name, either through the look-up table or through random-forest classification, is very quick.

Conclusions and outlook

Recognizing colors from images may seem like a trivial task, especially when performed at a relatively

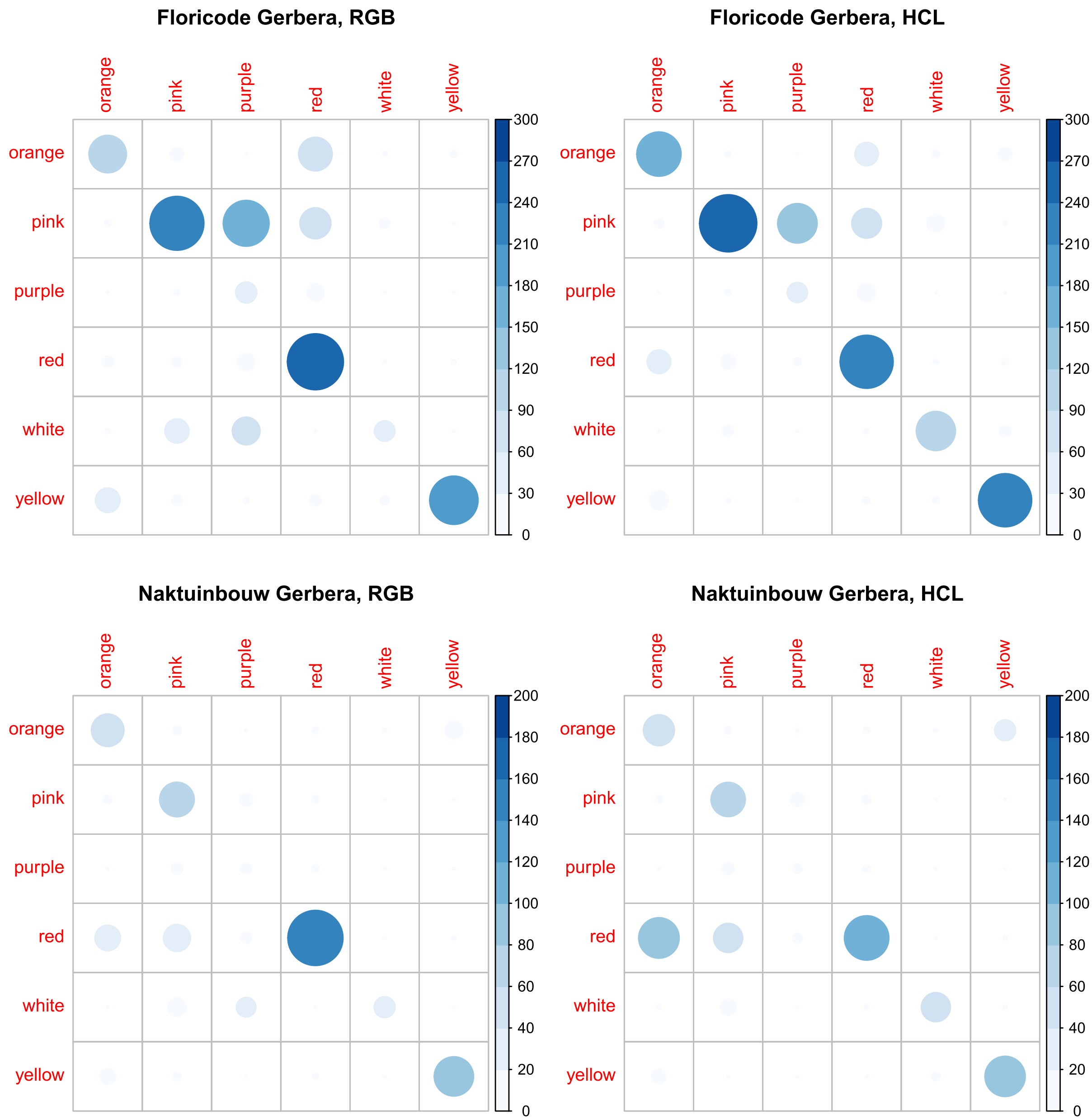


Fig. 4 Breakdown of classification errors for color-histogram methods using the translation table, gerbera data sets. The size and the color of the circle indicates the number of examples in each category. Rows indicate true colors, columns indicate estimates. Top row: Floricode gerberas; bottom row: Naktuinbouw gerberas. Left column: RGB-based analysis; right column: HCL-based analysis

coarse level like everyday color names. Still, it presents a significant challenge for automatic analysis. Apart from experimental factors such as lighting, there is quite some variability because flowers can have a complex structure, making shading and important factor. Often, multiple colors are present in one flower, which presents an additional complication in defining the main color.

In this paper we have attempted to reproduce expert opinion in four large data sets using advanced data analytics. We found that the best methods achieve success rates of 75–85%, something that in

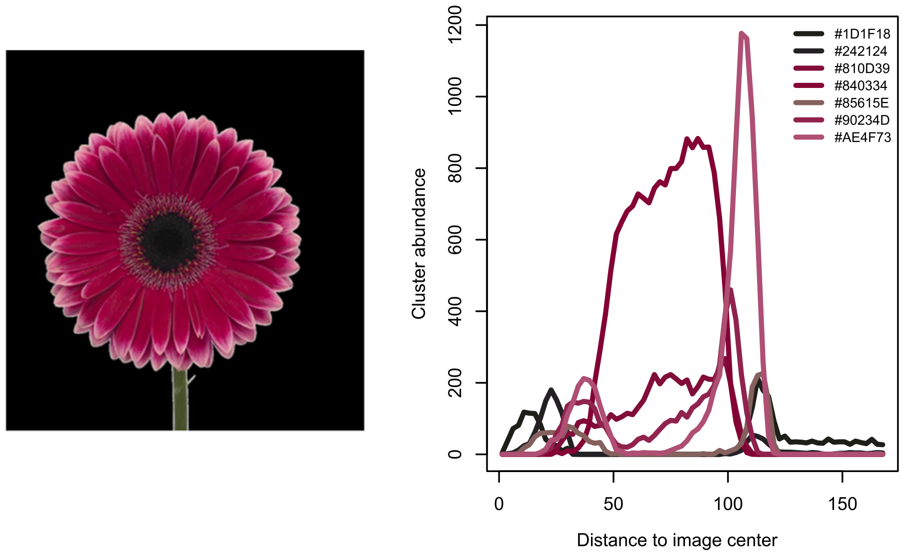


Fig. 5 Distribution of the flower colors in a Floricode Gerbera as a function of radius

our experience is not too different from the reproducibility of color assessments within the same organisation—with color assessments at different locations, under different conditions or with slightly different aims the reproducibility will be even lower. There are several other reasons why this performance level, while not 100%, is deemed quite good. In the first place, no color reference is present in the images, so that the effects of different lighting effects, camera settings and reflectance cannot be compensated for. Note that the images have been taken under relatively constant conditions, but still this may induce a bias, as suggested above. Especially when assessments from different locations are to be combined, some form of color calibration is expected to lead to marked improvements (see, e.g., Sunoj et al. (2018)). Secondly, flower color is a somewhat ill-defined concept. In the color-histogram strategy we have adopted a simple counting approach, that assumes that the color that dominates appearance is the main flower color, but this is not always correct. In contrast, the deep-learning approach (and to some extent, the combination of the color histograms with random forests) sidesteps this issue by learning the ground truth classification using all features available in the images, including any (consistent) biases. Thirdly, different individual flowers from the same variety will differ in their color features, and in many cases the flower that was photographed was not identical to the flower that was used by the expert in when entering the color features into the database.

This work does allow us to draw some conclusions on the suitability of the different methods for obtaining flower color in practical applications. Deep learning is—by far—the method that takes least preparation: basically one has to select a suitable network architecture, decide on a training scheme, and divide the data into training, validation (optional, as in this paper) and test sets. If enough data are available, this should lead to results that are state of the art. Moreover, prediction for new, hitherto unseen images will be very fast, making it possible to include this kind of prediction modules in real-time applications. The two downsides are that many examples are needed (where obtaining the ground truth for all examples usually is the bottleneck), and that it is very hard to say anything about the reasons for a particular classification – this is where the black-box nature of the method rears its head.

The color-histogram approach is fundamentally different, showing the user all colors constituting the image (typically, segmented images using only the clustered colors will give a very reasonable approximation of the original image), and also showing the choices that are made during the execution of the pipeline. This particular approach makes it possible to apply species-specific rules of thumb. An example is shown in Fig. 5. Gerberas show a distinct symmetry, where colors are usually dependent on the distance to the heart of the flower. The colors from the color histograms can be shown as a function of radius, allowing for specific choices such as the color at the tips of the petals, or at the heart of the flower. In fact, such plots can also be used to determine other characteristics, such as flower size and the size of the heart—note that the Naktuinbouw images in Fig. 1 include a ruler, allowing direct conversion of numbers of pixels to a measure like millimeters.

Two further conclusions can be drawn from the results in this paper: first of all, the use of lookup tables to convert color coordinates to color names is inferior to simply training a machine learning algorithm. Secondly, the use of the HCL color space in most cases is to be preferred over the RGB color space, the exception being the (rather important) category of red flowers.

It is to be expected that automatic extraction of information from flower images will become more and more important in horticulture. This paper has shown that although color recognition is far from trivial, there is ample potential for application. The low-hanging fruit will be to standardize the imaging protocol as much as possible, and at the very least include standard color charts in the images so that post-hoc color correction becomes possible. Further improvements very much will depend on the exact goal of the experiment, and may include imaging individual petals and other parts of the plants. At the very least, such protocols will make a big step towards interoperability of the data, something that with ever increasing economic interests will become more and more important.

Funding This work was supported by the Dutch Topsector Horticulture and Starting Materials, Grant Number LWV20.172.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

References

Afonso M, Mencarelli A, Polder G, Wehrens R, Lensink D, Faber N (2019) Detection of tomato flowers from greenhouse images using colorspace transformations. In: Oliveira PM, Novais P, Reis LP (eds) EPIA 2019. Lecture Notes in Artificial Intelligence, vol 11804. Springer, pp 146–155

Afonso M, Fonteijn H, Fiorentin FS, Lensink D, Mooij M, Faber N, Polder G, Wehrens R (2020) Tomato fruit detection and counting in greenhouses using deep learning. Front Plant Sci 11:1759

Afonso M et al (2023) Automatic flower registration from images through deep learning. Submitted for publication

Apriyanti DH, Spreeuwers LJ, Lucas PJF, Veldhuis RNJ (2021) Automated color detection in orchids using color labels and deep learning. PLoS ONE 16:1–27

Bosten JM (2022) Do you see what I see? Diversity in human color perception. Ann Rev Vis Sci 8(1):101–133

Bovik AC (2010) Handbook of image and video processing. Academic Press, Cambridge

Breiman L (2001) Random forests. Mach Learn 45:5–32

Eck JW, Vries DP (1995) Impartial assessment of rose (Rosa, 'Sweet Promise') flower colour with the aid of a chromameter. Plant Var Seeds 7:29–35

Emery KJ, Volbrecht VJ, Peterzell DH, Webster MA (2017) Variations in normal color vision. VI. Factors underlying individual differences in hue scaling and their implications for models of color appearance. Vis Res 141:51–65

Emery KJ, Volbrecht VJ, Peterzell DH, Webster MA (2017) Variations in normal color vision. VII. Relationships between color naming and hue scaling. Vis Res 141:66–75

Fonteijn H, Afonso M, Mooij M, Faber N, Vroegop A, Polder G, Wehrens R (2021) Automatic phenotyping of tomatoes in production greenhouses using robotics and computer vision: from theory to practice. Agronomy 11(8):1599

Fraley C, Raftery AE (2002) Model-based clustering, discriminant analysis, and density estimation. J Am Stat Assoc 97(458):611–631

Halcon User Guide MVTec Gmbh (2022) Germany. www.mvtec.com/products/halcon

He K, Zhang X, Ren S, Sun (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778

Liaw A, Wiener M (2002) Classification and regression by randomForest. R News 2(3):18–22

Lootens P, Van Waes J, Carlier L (2007) Evaluation of the tepal colour of *Begonia x tuberhybrida* Voss. for DUS testing using image analysis. Euphytica 155:135–142

Nilsback M-E, Zisserman A (2008) Automated flower classification over a large number of classes. In: 2008 Sixth Indian conference on computer vision, graphics & image processing, pp 722–729

Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Kopf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S (2019) Pytorch: an imperative style, high-performance deep learning library. In: Lyu H, Sha N, Qin S, Yan M, Xie Y, Wang R (eds) Advances in neural information processing systems, vol 32. Curran Associates Inc., Red Hook, pp 8024–8035

Perez-Udell RA, Udell AT, Chang S-M (2023) An automated pipeline for supervised classification of petal color from citizen science photographs. Appl Plant Sci 11:e11505

Qin X, Zhang Z, Huang C, Dehghan M, Zaiane OR, Jagersand M (2020) U2-Net: going deeper with nested U-structure for salient object detection. Pattern Recogn 106:107404

R Core Team: R (2022) A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna

Rosyani P, Taufik M, Waskita AA, Apriyanti DH (2018) Comparison of color model for flower recognition. In: 2018 3rd International conference on information technology, information system and electrical engineering (ICITISEE), pp 10–14

Scrucca L, Fop M, Murphy TB, Raftery AE (2016) mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. R J 8(1):289–317

Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: 3rd International conference on learning representations (ICLR 2015), pp 1–14

Singh S, Dhyani D, Yadav AK, Rajkumar S (2011) Flower colour variations in gerbera (*Gerbera jamesonii*) population using image analysis. Indian J Agric Sci 81(12):1130–1136

Stokman HMG, Gevers T, Koenderink JJ (2000) Color measurement by imaging spectrometry. Comput Vis Image Underst 79(2):236–249

Sunoj S, Igathinathane C, Saliendra N, Hendrickson J, Archer D (2018) Color calibration of digital images for agriculture and other applications. ISPRS J Photogramm Remote Sens 146:221–234

Van der Heijden GWAM, Polder G, Gevers T (2000) Comparison of multispectral images across the internet. In: Schettini R, Beretta GB (eds) Internet imaging, SPIE conference 26–28 January 2000, vol 3964, pp 196–206 (2000)

Van der Heijden GWAM, Polder G, Van Eck JW (1999) FLORES: a JAVA-based image database for ornamentals. In: Huijsmans DP, Smeulders AWM (eds) Visual information and information systems. Springer, Berlin, pp 641–648. Lecture Notes in Computer Science 1614

Wang J, Tian Y, Zhang R, Liu Z, Tian Y, Dai S (2022) Multi-information model for large-flowered chrysanthemum cultivar recognition and classification. Front Plant Sci 13:806711

Wehrens R, Buydens LMC, Fraley C, Raftery AE (2004) Model-based clustering for image segmentations and large datasets via sampling. J Classif 21:231–253

Yoshioka Y, Iwata H, Ohsawa R, Ninomiya S (2004) Quantitative evaluation of flower colour pattern in regions of interest by image analysis and principal component analysis of *Primula sieboldii* E. Morren. Euphytica 139:179–186

Yoshioka Y, Ohsawa R, Iwata H, Ninomiya S, Fukuta N (2006) Quantitative evaluation of petal shape and picotee colour pattern in Lisianthus by image analysis. J Am Soc Horticult Sci 131:261–266

Zeileis A, Murrel P, Hornik K (2009) Escaping RGBland: selecting colors for statistical graphics. Comput Stat Data Anal 9:3259–3270

Zeileis A, Fisher JC, Hornik K, Ihaka R, McWhite CD, Murrell P, Stauffer R, Wilke CO (2020) colorspace: a toolbox for manipulating and assessing colors and palettes. J Stat Softw 96(1):1–49

Zhenjiang M, Gandelin M-H, Baozong Y (2006) An OOPR-based rose variety recognition system. Eng Appl Agric Intel 19:79–101