SpotEgg: An image-processing tool for automatised analysis of colouration and spottiness

Jesús Gómez[1] and Gustavo Liñán-Cembrano[2]

[1]Departamento de Ecología de Humedales, Estación Biológica de Doñana (EBD-CSIC), Calle Américo Vespucio s/n, 41092, Sevilla, Spain
[2]Instituto de Microelectrónica de Sevilla (IMSE-CNM), Consejo Superior de Investigaciones Científicas (CSIC) and Universidad de Sevilla, Calle Américo Vespucio s/n, 41092, Sevilla, Spain

Corresponding author: J. Gómez, Departamento de Ecología de Humedales, Estación Biológica de Doñana (EBD-CSIC), Calle Américo Vespucio s/n, 41092, Sevilla, Spain. E-mail: j.gomez@ebd.csic.es

Abstract

Colouration and patterning are widespread amongst organisms. Regarding avian eggs, colouration (reflectances) has been previously measured using spectrometers whereas spottiness has been determined using human-based scoring methods or by applying global thresholding over the luminance channel on photographs. However, the availability of powerful computers and digital image-processing algorithms and software offers new possibilities to develop systematised, automatable, and accurate methods to characterise visual information in eggs. Here, we provide a computing infrastructure (library of functions and a Graphical User Interface) for eggshell colouration and spottiness analysis called SpotEgg, which runs over MATLAB. Compared to previous methods, our method offers four novelties for eggshell visual analysis. First, we have developed a standardised non-human biased method to determine spottiness. Spottiness determination is based on four parameters that allow direct comparisons between studies and may improve results when relating colouration and patterning to pigment extraction. Second, researcher time devoted to routine tasks is remarkably reduced thanks to the incorporation of image-processing techniques that automatically detect the colour reference chart and egg-like shapes in the scene. Third, SpotEgg reduces the errors in colour estimation through the eggshell that are created by the different angles of view subtended from different parts of the eggshell and the optical centre of the camera. Fourth, SpotEgg runs automatic Fractal Dimension analysis (a measure of how the details in a pattern change with the scale at which this pattern is measured) of the spots pattern in case researchers want to relate other measurements with this special spatial pattern. Finally, although initially conceived for eggshell analysis, SpotEgg can also be applied in images containing objects different from eggs as feathers, frogs, insects, etc., since it allows the user to manually draw any region to be analysed making this tool useful not only for oologist but also for other evolutionary biologists.

Key words

Colour measurement, egg volume, eggshell, fractal dimension, image processing, maculation, patterning

Introduction

Avian eggshell colouration and spottiness could play multiple roles (Kilner 2006, Cherry and Gosler 2010). Several hypotheses have been advanced to explain the occurrence of spots in avian eggshells. For example, in ground nesting birds, more highly spotted eggs have been shown to be better camouflaged, however, as a drawback, more spotted eggs might suffer from overheating when directly exposed to solar radiation (Gómez et al. 2016). Spottiness of eggs in host species has been considered an important trait to be mimicked by successful avian nest parasites (López-de-Hierro and Moreno-Rueda 2010, Avilés et al. 2006). Other functions of spottiness have also been proposed, such as physical eggshell protection (Bulla et al. 2012), antibacterial defence or UV protection (Maurer et al. 2011).

Despite its relevance, the characterisation of spottiness using digital image-processing tools has not been standardised along the different studies. Many times, spottiness determination has simply relied on subjective human perception (Gosler et al. 2000, Sanz and García-Navas 2009). In other cases, it has been determined by manually setting a threshold from the histogram of luminance of the input image (Duval et al. 2013, Duval et al. 2015), but this is also prone to observer's subjectivity. An unbiased detection of spots and background in eggshell digital images, as well as using reliable methods for colour analysis, is cornerstone to test the different functions proposed for avian eggshell pigmentation. It is also evident that the variety of patterns, colours, distribution and size of spots, etc., found in avian eggshells require many times using more than just one parameter (degree of maculation) for a proper characterisation and evaluation of several hypotheses. We could rely on the widely applied method proposed by Gosler et al. (2000) to obtain more than one parameter to characterise spottiness. However, this method has low inter-observer repeatability Brulez et al. (2014a) as it uses researcher' skills to visually quantify three scores (visual expert judgement): spot pigment intensity, distribution of spotting over the surface and average spot size. Furthermore, other studies suggest that these scores are not a good proxy indicator for pigment quantification (Brulez et al. 2014b, Wegmann et al. 2015) which may be result of its low repeatability.

Separating spots from background in eggshell images is far from being a trivial image-processing task. There is not a single threshold that can be applied to separate spots from background (Stoddard and Stevens 2010). Eggshell surfaces over which spots are to be detected are not uniformly illuminated in general (in particular when taking images in the field). Moreover, and conceptually more important, eggshells are 3D curved shapes and even assuming the eggshell as an uniformly illuminated Lambertian surface, the visual angles subtended from the optical axis of the system (perpendicular to the sensor plane passing through the optical centre) to different parts of the eggshell are different and, hence, we cannot expect to receive the same amount of reflected light per area unit from segments of the eggshell that are seen under different subtended angles, no matter how uniform illumination was. This effect is particularly prominent among eggshell areas having large differences in this angle (see, for instance, how setting a global threshold for spots segmentation fails near the pointed pole of the egg in Fig 1(a) in Duval et al. (2015).

Here, we introduce a tool called SpotEgg that works under Windows, Linux, and OS X operating systems running MATLAB v2012.b (or latter) (MathWorks, Natick, MA, USA). Together with the library of functions, we also provide (under request and also free in different repositories in the web, together with some video-tutorials to help users; see Data accessibility section) a Graphical User Interface (GUI) that permits our tool to be utilised by people that have never used MATLAB

before. Besides, as long as images are taken according with some guidelines (see next section), the tool can run almost automatically, minimizing the interactions from the user. SpotEgg has mainly been conceived to provide information in pictures about spottiness (number, size, distribution and shape of spots), colouration (reflectances), egg physical size and Fractal Dimension (FD, Mandelbrot 1967). Images may contain any arbitrary number of Regions of Interest (RoIs), which will normally correspond to eggshells. However, any other region whose spottiness or reflectance is to be measured can be also be manually defined. Finally, we have also incorporated a novel way to reduce the effect of the eggshell 3D shape in the reflectance measurements. In the manuscript we introduce the main features provided by SpotEgg and describe its main innovations, not going into implementation details for conventional image-processing parts.

Preliminary Considerations for Optimum Use of SpotEgg

One (see for more functions in Table 1) of purposes of SpotEgg is to generate reflectance measurements from some selected objects in a digital picture. Unfortunately, digital cameras outputs cannot be directly used for that purpose for two main reasons. First, pixel values in a digital image are just dimensionless digital words of a number of bits (8-10-12 bits most commonly). Hence, in order to scale these digital numbers to produce reflectance values, a process known as *normalization,* we need to have portions of the image where their pixels correspond to objects whose reflectance is known (i.e. we need reflectance targets or colour charts in the scene). Assuming that both the object and the reflectance target receive the same incident light power and both act as Lambertian surfaces (reflect equally in all directions) one can use pixel values from the known areas to scale the whole image and produce reflectance values. The second reason is that cameras are intended to produce images that are visually appealing to their main users. To that purpose, the outputs from the camera sensors, which are linear with respect to radiance, are non-linearly transformed by a so-called radiometric transfer curve. Basically speaking this transformation tries to maximize details in the highly illuminated areas of the scene and discard information in poorly illuminated areas. Sadly, for our purposes, such non-linear transformation precludes these output images to be directly used for reflectance measurements (even after carrying out scaling). The options at this point are (a) to access the real information captured by the camera sensors –what is called the raw pixel data, or (b) to have a number of targets or different known reflectances in the scene, to use their reflectance values to estimate which radiometric transfer function has been applied and revert its effects into the camera output image. SpotEgg offers both options. If it finds that the image provided contains raw pixel data, it employs the free tool DCRAW (Coffin 2015) to extract a so-called linear (with respect to radiance) image (and also does an extra-linearization using patches if they are in the scene, see *Linearization and Scaling Factor)*. If the image provided does not contain raw pixel data, SpotEgg will try to linearize it from the information of the reflectance targets in the scene. Once the images are linearized and scaled, SpotEgg runs its spot analyses and reports information about total spottiness, physical characteristics of the spots (size, shape, position and colour information), also reporting colour characteristics for the background.

SpotEgg capabilities (Table 1) are better exploited when the image-taking process and the data are executed/organized according to some basic requirements (detailed in Supplementary Material). Here we highlight the most important of these considerations.

(1) Images must be taken, when possible, with same or at least very similar camera settings. We normally employ ISO 100, F8, and no zoom, but other configurations are fine as long as the image is well exposed and correctly focussed.

(2) Egg patterning details need to be captured at sufficient resolution to be analysed properly. Clearly, it the image lacks of details, SpotEgg will not be able to provide precise spot detection. You should try to put the reflectance target close to the eggs and set image framing so that you are not wasting image areas in parts which are not to be processed later. As a rule of thumb, eggs should occupy at least 1% of the scene (this is the minimum size the automatic egg finder tool requires for reliable operation), but, obviously, the larger the better.

(3) Raw format is preferred since it allows SpotEgg to read the real information captured by the sensors in your camera. However SpotEgg also accepts jpg, png, bmp, and tif formats. In these cases, you need to have at least three different neutral colour reflectance targets in the scene for linearization. In any case, make sure you save your images using a lossless compression scheme when not using raw pixel data.

(4) In order to extract reflectances, images must contain at least one neutral colour patch of known reflectance for normalization. Though this is not absolutely mandatory (Troscianko and Stevens 2015), acquiring reference chart information from a different image forces us to guarantee that illumination conditions and camera setup are kept constant between the two acquisitions, any difference there will degrade the accuracy of the results.

(5) SpotEgg uses the free tool DCRAW (Coffin 2015) to extract pixel values from raw images. This software must be installed into your system.

(6) When images do not contain raw pixel data, SpotEgg can also linearize them for reflectance calculation as long as they contain reflectance targets or colour charts having at least three neutral colour patches of known reflectance (the larger the number of patches the better the linearization results). To save researcher time, SpotEgg includes an option for automatic detection of a specific standard colour chart (ColorChecker Passport, USA X-Rite Inc.).

(7) Proper illumination is essential for getting good results. Images should not contain partial shadows, artefacts, and overexposed/underexposed areas. Ornés et al. (2014) introduced the use of a "black-box" to help in controlling lighting conditions in the field, but also other resources as neutral colour reflectors or diffusive neutral colour umbrellas could be used, especially in places of controllable illumination conditions (lab, museum, etc.). If only spottiness is required, camera bracketing may be useful to produce High Dynamic Range (HDR) versions of the scene (note that HDR alter colours).

(8) Finally, since SpotEgg runs the programmed analyses for all the pictures in the selected folder, we recommend to create new folders for each species because the parameters for spottiness analyses are defined only once for a whole dataset (input folder).

SpotEgg Operational Flowchart

SpotEgg is basically a GUI that uses a library of image-processing functions created for MATLAB. Running SpotEgg requires both MATLAB (v2012b or latter) and DCRAW (Coffin 2015) to be installed and added to the system path (explained in the video-tutorials we have created to help users). Once MATLAB is running, one simply navigates to SpotEgg installation folder (setting the work directory), and types its name (case-sensitive) in the MATLAB command window, to start the application.

Main Window. – Fig.1 shows the main window of our tool. Two boxes in the upper part of the GUI allow the user to define the input folder (all images here will be processed) and the output folder where results will be stored. Buttons in the control box (grey-shaded area) allow the user to command a Linearisation (obtaining equivalent reflectance images), RoI definition (auto or manual), or colouration and spottiness analysis (which includes spot detection, measurement, and FD calculation). Alternatively, users can also command a *run all* action, which executes these three steps in a sequenced process. Fig. 1 of Supplementary Material shows the operational flowchart of SpotEgg. The three large coloured boxes correspond (from top to bottom) to the steps of creating the reflectance image, defining the RoIs, and running colouration and spottiness analysis.

Linearization and Scaling Factor. – When reading the input image, the tool decides if it comes as raw data or not in order to make a call to DCRAW to get linear pixel data as written by the camera. Once the image is available as a MATLAB' variable, SpotEgg checks if automatic chart detection has been selected. In this case, it employs an internal function to identify the Passport X-Rite chart (ColorChecker Passport, USA X-Rite Inc.) and to try to find its 24 patches, and label them. When detection is successful (checked internally by the tool), see Fig. 2(a), a data structure containing the location of the centroid of each patch, dimensions, and measured RGB values, is written to a file (together with file name, and other parameters). Our tool also obtains a pixel-pitch to millimetre scaling coefficient from the size of the found patches. This coefficient permits to report metric measurements of egg and spots (provided that the chart is located next to the subject of interest so that both of them are at roughly the same distance from the camera). When the image contains a different calibration chart, or if X-Rite auto detection has failed, the user is requested to click on a point on each patch and to provide reflectances either by typing values in a box or by selecting a text file (using comma separated values). Additionally, users are also requested to define the pixel-to-mm scaling constant by clicking on two points of known separation. We wish to clarify two important factors. First, when the chart included in all the pictures to be processed is the same, the user provides reflectance information only once. Second, SpotEgg does not simply employ the user-handpicked pixel on each patch for registering RGB values, but computes mean values over the whole area of the patch. Patches are reconstructed from the selected point using an iterative thresholding and morphological image-processing algorithm. The RGB values of the patches are employed to linearize the input image and to produce an equivalent RGB average reflectance image (in unsigned 16-bit format). Our tool runs its linearization algorithm even when DCRAW was used to extract the image in linear format from raw data. Such extra linearization (using a 3rd degree polynomial fitting) improves the accuracy in determining the reflectance of the known patches as shown in Fig. 3.

There, we see how SpotEgg produces errors whose averages tend to zero across the six grey patches in the X-Rite chart. Conversely, using the values as provided by DCRAW linear output produces errors that, obviously, exhibit a minimum at the patch employed for scaling but are not zero-mean across the six grey patches on the chart (see blue squares in Fig. 3).

RoI Definition. – Once reflectance images are obtained, we need to define the RoIs where spottiness and colouration have to be evaluated. Being initially conceived for eggshell analysis, and primarily aimed to save researcher time, SpotEgg includes a module which tries to automatically find egg-like shaped areas in the scene, however, if the analysis is to be executed over a different kind of object (e.g., butterfly wings), RoI borders can be drawn as closed polygonal lines with any number of vertexes. When running automatic eggshell detection, SpotEgg uses an automatic foreground/background multi-threshold (histogram-based) segmentation algorithm and checks whether the border of each found foreground object can be made to fit (using MATLAB nonlinear optimization toolbox) to the equation of an ellipsoid defined as in Eq.1 in Troscianko (2014). Objects having large fitting errors are discarded whereas objects having low fitting errors are stored to be presented to the user for visual examination and validation at a later stage. Our egg-like shape detector uses a rotation-invariant algorithm, hence it does not require the egg to be placed in any privileged orientation in the scene. When all images in the input folder have been processed, results are displayed (using a red overlay), as illustrated in Fig. 2(b). At this point the user can accept the result, modify it, or discard it. Modification options include eroding/dilating/moving the whole shape, and adding/deleting/moving any number of vertexes to border of the RoI. Besides, the tool also allows adding any number of new RoIs to an image by manually drawing their shapes. Once RoIs are created, we employ the pixel-to-mm scaling coefficient for this image to calculate a set of geometrical properties for each of the RoIs in the scene. Among these measurements our tool reports length and width (in mm), volume (mm^3) and area (mm^2) (assuming the RoI is the generator of a 3D revolving surface), and also volume (mm^3) using Hoyt's formula (Hoyt 1979) (for comparative purposes). In addition to that, it also finds the orientation of the RoI and the position of its centroid and pointed pole. RoI features are written to a file in a structured format (each RoI is an element of the structure) that can easily be loaded into MATLAB's workspace either for spottiness analysis using our tool or for any other purpose.

Analysing Spottiness and Colouration. – Our spot detector uses a complex image-processing algorithm, whose operation is determined with four parameters (see *Spot Detection*), to characterise spottiness in each of the RoIs in an image. The tool produces three .CSV output files per input image, namely the Features Report (FR), the Spots Report (SR), and the Colour Profiles Report (PR).

      The Features Report contains per-RoI information about; *Geometrical properties* of the RoI: volume, area, length, width, centroid, pointed pole; *Spottiness*: global degree of spottiness (% of RoI area), fractal dimension FD (Mandelbrot 1967), number of spots, average size of spots (% of RoI area); and *Colouration*: average reflectance in the R, G, and B, channels for the whole RoI, spots, and background.

      The Spots Report supplies, in one row per spot, information about position of centroid, eccentricity, size (% of RoI area), axial (D) and angular (Ɵ) distances (to the pointed end of the RoI, (Supplementary Material Fig. 2), and per-channel average reflectances.

Finally, the Colour Profile report, contains information about average reflectances (per colour channel) for 100 equally spaced slices across the longitudinal axis of the RoI, allowing to study how global colouration varies from pole-to-pole of the RoI.

When commanded so, SpotEgg can also save two images displaying maculation for each input file. First, it produces an uncompressed black and white image (white areas corresponding to spots) that can be used for masking purposes either in MATLAB, or any other layer-based image-processing tool (e.g. Adobe Photoshop, ImageJ, etc.). Additionally, it also saves an uncompressed version of the original input image showing detected spots borders in red over the original image.

Spot Detection. – Probably, the main improvement and innovation provided by SpotEgg is its algorithm for spot detection. Spot detection in eggshell images (or any other RoI in general), can be understood as a segmentation problem over a slightly texturised non-uniformly illuminated coloured background. Spots are, in general, non-uniformly illuminated coloured objects of arbitrary size and shape whose only *a priori* known characteristic is that of being darker than the background in their vicinity (the spots being brighter might also be incorporated in the tool). As mentioned in the introduction, eggshells are curved 3D shapes, and equally-sized areas located on different positions over the eggshell will be, in general, projected onto different areas in the sensor focal plane (similar to what happens when projecting earth surface into maps). This results in, first, a distortion in the apparent size of spots in areas seen from very different subtended angles Hartley and Zisserman (2003), and, second these different angles makes two lambertian equally-illuminated identical areas to reflect different amounts of light power into the camera's field of view, with the one subtending the smaller angle appearing darker in the captured image. Such darkening produces two important errors when computing spottiness and eggshell colouration. On the one hand, the apparent *shading* produces errors in the computation of reflectances, especially in areas that are viewed under a very low subtended angle, because their apparent darker colouration is not due to a lower reflectance but to the fact that we are not receiving equal light power per area unit from this part of the image (Supplementary Material Fig. 3). On the other hand, it also affects the spots segmentation process as it significantly darkens the background in these areas, making impossible to define a single threshold for separating spots from background for the whole RoI. Indeed, we usually find egg pictures in our test data set where the background in areas next to the borders of the egg (in particular close to the pointed end) appears even darker than the spots in other areas of the same egg. Clearly, using a single threshold would produce effects like those in Fig 1(a) Duval et al. (2015).

Perhaps the simplest solution is to manually draw an exclusion mask to avoid including these areas in our analyses. However, our aims were to create a standardised method not relying on subjective appreciations of what areas to exclude, and to minimise user interactions and save researcher time. Hence, instead of using global threshold techniques, SpotEgg creates per-pixel optimised thresholds (spatially-variable thresholding techniques) which are obtained from low-pass spatial filtered versions of each colour channel of the input image by using 2-D averaging disk-shaped kernels (Shapiro and Stockman 2001) of programmable size. This spatial low-pass version of the input image serves as a threshold indicator for segmentation as described in (Supplementary Material: Defining Local Thresholds for spots segmentation) making the decision whether a pixel corresponds to a spot or not only dependant on the difference between this pixel value and an average in a

programmable number of pixels around it. Only four parameters define the operation of the algorithm these are:

(1) The radius in mm of the averaging filter for calculating the initial local threshold denoted as R.

(2) The minimum difference (threshold) between a pixel and its neighbourhood to consider it a spot, denoted as T,

(3) A similarity filling factor which avoids the central area of very large spots to be incorrectly assigned as background in case these central areas are brighter than those at the periphery of the spot, denoted as SF.

(4) The minimum spot size to be detected (MS, mm^2).

Fig. 4(c) shows spot detection by SpotEgg on a Kentish plover, *Charadrius alexandrinus*, eggshell. Our method is able to detect spots in areas where a human-based single threshold method has failed (Fig 4).

Spot Detector Training. – It is impossible for us to provide numerical values for the four parameters {R, T, SF, MS} that control our spot detection algorithm for all species and image capturing conditions. Instead, the tool includes a training interface (see Training Run box in Fig. 1 Supplementary Material). Thus, at the beginning of a spot detection run, the tool asks the user whether a *config* file for these images exists or not. When not available, the training routine is executed. Here, the user first selects a relevant image (which does not need to be from the same dataset) for training and the tool runs spot detection for five configurations {R, T, SF, MS}$_{1\ldots5}$. Results are presented for visual examination and the user is requested to select the configuration providing better spot detection. At this point, a small control with four sliders is displayed and the user can try any number of new configurations. SpotEgg shows the resulting spot detection image for any new combination of parameters provided by the user until the user decides that the training is satisfactory. The resulting parameters are saved to a configuration file and employed during the analysis for all the images in the data set. It is worth mentioning that this systematisation of the analysis allows researchers to replicate studies (by sharing the configuration files) from other authors and systematically compare them without introducing any human-based biasing.

Background Colour Constancy Option. – A perfect correction of the apparent darkening of areas that are viewed under different subtended angles requires a well knowledge of many physical parameters of the setup (camera-to-subject distance, focal length, F# number, lens distortion, etc.) as well as a precise definition of the 3D structure of the scene. The later involves to know how to represent each point in the eggshell into the camera's coordinate system (x, y, and z axes). Most of these parameters are hard to be obtained, in particular when working in the field. Therefore, we must accept that in most cases some darkening effect will be always present. However, our tool includes an option to attenuate this error (Fig. 5). This technique, which is enabled by checking a box in the main GUI, has been denoted as the *Background colour constancy option.* Basically, it relies on assuming that the average colouration of background of the eggshell is constant across a coarse tessellation of its shape. Whether or not this assumption is valid is something the user has to consider based on its own experience, knowledge, species, and image taking conditions for each study. Here we just provide a computing tool that, if this assumption is accepted, reduces the effect of the apparent darkening in eggshell areas that are viewed under low subtended angles (see Supplementary Material: Eggshell Apparent Darkening Due to the Visual Subtended Angle). The algorithm operates in two passes. First it

runs a normal spot detection. Afterwards, it detects what are defined as high-likelihood background locations (positions across the eggshell where certainty about reliable detection of the background is higher, obtained by combining spot detection and per-colour-channel histogram of the reflectance image). Now the bounding box of each eggshell image is divided into 25x25 equal-size sectors, and per-sector average reflectances of high-likelihood background locations are calculated. This information is employed to create a transformation matrix (same size as the RoI image) that scales the obtained average reflectance (per-channel) in each sector to that of a central area around the centroid of the RoI (assuming that pictures are taken so that this central area exhibits the lowest projective distortion). The central area is iteratively enlarged until it occupies at least 10% of the area of the eggshell, and contains at least 500 high-likelihood background pixels. The transformation matrix is low-pass filtered (spatial) in order to smooth it and avoid the creation of artefacts. Finally, RoI colour channels are multiplied by their corresponding transformation matrix to obtain the new image. Spot detection is then executed over this new image. Obviously, when the image contains more than one eggshell, the background constancy algorithm is applied independently for each of them.

Fig. 6 shows a comparison of the reflectances obtained for background and spots when using/not using the background colour constancy algorithm for a dataset of 10 Kentish plover eggshells. Notice that, as expected, measured reflectances increase when the background constancy algorithm is used, and the same happens for the contrast (difference background vs. spots). Indeed, this contrast gain produces a better spot detection during the second run, especially noticeable in highly darkened areas.

Fractal Dimension (FD). – SpotEgg calculates Fractal dimension, Mandelbrot (1967), using a box counting method (Huang and Turcotte 1990). In our case, the user does not need to prepare the image to be processed to meet the 2^n requirement for region width (as in the conventional implementation of the box counting methods, Supplementary Material: An Approximation to Box Counting Methods over non-rectangular RoIs), thus simplifying its usage and making region cropping/padding transparent to the user.

Examination of Results. – SpotEgg informs about the end of a run by displaying the word *IDLE* in its *status* window. Processing time obviously depends on the performance of the computer, the size of the input images, and the number of images in the dataset. In our case, we recorded our video tutorials using an old 2008 4Gb RAM iMAC running OSX 10.11.5, over 3888x2592 pixel images. For this setup, automatic chart detection (which includes raw file accessing by DCRAW) takes some 25s, egg detection takes 15s, and spot detection takes 10s when the background constancy is disabled and 65s when it is enabled. Summarizing, processing time amounted to some 105s per image in the worst case.
Since SpotEgg executions can take quite a long time, specially for large datasets, individual reports are provided as soon as an image is processed. Thus, users can visualize spot detection results, or check individual reports while a process is still running (e.g. Fig. 7). Only the unified summary reports (whose generation is enabled by a check-box in the main GUI) are compiled at the end of a run. This possibility saves time as it allows users to stop the analyses at an early processing stage if there is a need to re-tune the spot detection parameters.

Discussion

SpotEgg is a library of functions for MATLAB, including a GUI, which is intended to systematically analyse spot patterns and colouration in digital images. Although developed for eggshell analysis, the tool is versatile enough to process RoIs associated to any other kind of object (insects, frogs, etc.), although in these cases, measurements about volume and total egg area are not valid as these RoI do not correspond to 3D surfaces obtained from the 3D rotation of the RoI over its major axis. This software facilitates digital image processing to oologists and drastically reduces the time devoted for routine tasks by the researcher, especially when having large datasets. Furthermore, it produces fully replicable and unbiased results that are more accurate than those from previous methods based on human scoring or global thresholding (Fig. 5), providing a standardized method to characterize spottiness and colouration.

Spectrometers were, till few years ago, the most suitable tool to obtain eggshell colouration, being frequently used in ecology and evolutionary biology (Avilés 2008, Cassey et al. 2010). However, spectrometers are, generally speaking, single point probes which do not allow to measure 2D patterns. Moreover, measuring the reflectance of isolated spots becomes quite difficult in many species due to the small size of their characteristic markings. Gosler et al. (2000) introduced a method to quantify and describe eggshell spottiness based in human visual scoring (a kind of expert judgement). However, this method exhibits low repeatability (Brulez et al. 2014a). But, similar as it is happening with other techniques and devices that are revolutionising some fields in biology (Fairhust et al. 2013, Gómez et al. 2014, Mulero-Pázmány et al. 2015), sensory ecology is evolving rapidly with the use of digital cameras (Stevens 2013). Specifically, digital-image processing is becoming ubiquitous among evolutionary biologists and oologists studying colouration and patterns (e.g. Stoddard et al. 2014, Stevens 2011, Troscianko and Stevens 2015). However, researches are not yet fully exploiting its capabilities, perhaps due to the fact that image-processing experts are not normally involved in these studies. Likely, future patterning and colour research (not only for eggshell studies) will make extensive use of multi-spectral cameras (Russell and Dierssen 2015). However they are still too expensive, have low resolution, and are slow and bulky to take them out of the lab and considering using them in the field. Alternatively, even currently available low-cost digital cameras can be used to properly quantify eggshell colouration (in the colour bands provided by the camera) and patterning, assuming that some basic requirements are met (Stevens et al. 2007).

With SpotEgg, we provide evolutionary biologists and oologists with a powerful image-processing tool of very simple use that makes transparent to the user its internal operation. We have chosen MATLAB as our operating framework, despite being a licensed software, due to its capability for implementing complex image-processing algorithms (including nested loops, conditional executions and multiple-passes algorithms) as compared to other general-purpose tools like Adobe Photoshop (Adobe, San Jose, CA, USA) or ImageJ. Furthermore, MATLAB is widely used worldwide and image-processing experts share many of their codes through the MATLAB Central. It could be argued that the OpenCV framework (Open Computer Vision) is a license-free alternative to MATLAB, which, roughly, offers the same computing capabilities. However OpenCV is primarily employed in applications whose purpose is to achieve real-time operation. Moreover, during algorithm design and prototyping phases, MATLAB allows, in our opinion, a faster developing time and an easier collaboration with other groups. Additionally, it also allows us to

integrate functions from other MATLAB Toolboxes like non-linear optimization, statistical analysis, etc., into SpotEgg. In any case, now that SpotEgg has reached a stable version, we will start the migration of SpotEgg functions into an OpenCV-based tool working under Linux. Unfortunately we cannot provide a prediction of the release time for this new version as it will depend on the available funds.

Our tool has been conceived to operate as automatically and autonomously as possible, requiring very little interactions from the user, always through a GUI, with the main purpose of saving researcher time devoted to routine, non−creative, tasks. In addition to these automation advantages, SpotEgg provides a standardised, automatable, reproducible, and parameterised spot detection. Besides, it also incorporates a method that partially compensates the errors in reflectance estimation induced by the darkening produced by the natural 3D curvature of the eggshell.

Since eggshell properties (colour, spottiness, size, etc.) vary among species, we recommend the use of the same spot detection parameters for a whole data set (digital images for eggshells of the same species) in order to allow for accurate comparisons. Moreover, the parameterisation of our spot detector allows authors to easily replicate other studies to compare results, also making easier a meta-analysis of the topic. This is something that, in our opinion, was an important lack in previous spottiness characterization techniques as neither human-based scoring, nor manual global thresholding allow for accurately reproducing any other study results.

Another important feature provided by SpotEgg, is the possibility to obtain Fractal Dimension without requiring user interactions to crop the region of interest into a rectangular sector as it is normally required. Fractal Dimension figure provides a measurement of how details in the spottiness pattern change with the scale at which the subjects are observed. In our opinion, FD results, although not used yet, could be relevant for eggshell studies as FD might be related with pigments deposition and/or patterns. For example, Pérez-Rodríguez et al. (2013) found that higher bib FD indicates bird´s quality in red patridge (*Alectoris rufa*), so there is some possibility (it is not the purpose of our work here to check this hypothesis) that FD of eggshell spottiness is relevant for eggshell communication/signal theory, as suggested for background eggshell colour (Moreno and Osorno 2003).

To conclude, by creating SpotEgg we have tried to simplify eggshell analyses, improving results, and their repeatability, and saving researcher time devoted to automatable tasks. We believe this tool may be useful for a significant number of researchers that work in general with eggs and more specifically they that are investigating function and evolution of eggshell colouration and spottiness (Avilés et al. 2006, Martínez-de la Puente 2007, Soler et al. 2008, Maurer et al. 2011, Amat et al. 2012, Cassey et al. 2012, Talabante et al. 2013, Portugal et al. 2014,Wilson-Aggarwal et al. 2016). Finally, thanks to its flexibility, SpotEgg can be extended beyond the analysis of eggshell patterns and colouration to other type of subjects.

Data accessibility

For video-tutorials and the latest version of SpotEgg see http://www2.imse-cnm.csic.es/~linan/GLC_HomePage/SpotEgg.html

References

Amat, J.A., Monsa, R. & Masero J.A. 2012. Dual function of egg-covering in the Kentish plover *Charadrius alexandrinus*. — Behaviour. 149: 881–895.

Avilés, J.M., Stokke, B.G., Moksnes, A., Røskaft, E., Åsmul, M. & Møller, A.P. 2006. Rapid increase in cuckoo egg matching in a recently parasitized reed warbler population. — J. Evol. Biol. 19: 1901–1910.

Avilés, J. M. 2008. Egg colour mimicry in the common cuckoo *Cuculus canorus* as revealed by modelling host retinal function. — Proc. R. Soc. Lond. B, Biol. Sci. 275: 2345–2352.

Brulez, K., Choudhary, P.K., Maurer, G., Portugal, S.J., Boulton, R.L., Webber, S.L. & Cassey, P. 2014a. Visual scoring of eggshell patterns has poor repeatability. — J. Ornithol. 155: 701–706.

Brulez, K., Cassey, P., Meeson, A., Mikšík, I., Webber, S.L., Gosler, A.G. & Reynolds, S.J. 2014b. Eggshell spot scoring methods cannot be used as a reliable proxy to determine pigment quantity. — J. Avian Biol. 45: 94–102.

Bulla, M., Šálek, M. & Gosler, A. G. 2012. Eggshell spotting does not predict male incubation but marks thinner areas of a shorebird's shells. — Auk. 129: 26–35.

Cassey, P., Portugal, S.J., Maurer, G., Ewen, J.G., Boulton, R.L., Hauber, M.E. & Blackburn, T.M. 2010. Variability in avian eggshell colour: a comparative study of museum eggshells. — PLoS one. 5: e12054–e12054.

Cassey, P., Mikšík, I., Portugal, S.J., Maurer, G., Ewen, J.G., Zarate, E., Sewell, M.A., Karadas, F., Grim, T. & Hauber, M. E. 2012 Avian eggshell pigments are in two Turdus thrushes. — J. Avian Biol. 43: 503–512

Cherry M.I. and Gosler A.G. 2010. Avian eggshell coloration: new perspectives on adaptive explanations. — Biol. J. Linn. Soc. 100: 753–762.

Coffin, D., DCRAW Application. <https://www.cybercom.net/~dcoffin/dcraw/> accesed 10 November 2015.

Duval, C., Cassey, P., Lovell, P.G., Mikšík, I., Reynolds, S.J. & Spencer, K.A. 2013 Eggshell appearance does not signal maternal corticosterone exposure in Japanese quail: an experimental study with brown-spotted eggs. — PLoS One. 8: e80485.

Duval, C., Cassey, P., Lovell, P.G., Mikšík, I., Reynolds, S.J. & Spencer, K.A. 2015. Maternal influence on eggshell maculation: implications for cryptic camouflaged eggs. — J. Ornithol. 157: 303–310.

Fairhurst, G.D., Marchant, T.A., Soos, C., Machin, K.L. & Clark, R.G. 2013. Experimental relationships between levels of corticosterone in plasma and feathers in a free-living bird. — J. Exp. Biol. 216: 4071–4081.

Gómez, J., Michelson, C.I., Bradley, D.W., Norris, D.R., Berzins, L.L., Dawson, R.D. & Clark, R. G. 2014. Effects of geolocators on reproductive performance and annual return rates of a migratory songbird. — J. Ornithol. 155: 37–44.

Gómez, J., Pereira, A.I., Pérez-Hurtado, A., Castro, M., Ramo, C. & Amat, J. A. 2016. A trade-off between overheating and camouflage on shorebird eggshell colouration. — J. Avian Biol. (doi: 10.1111/jav.00736)

Gosler A.G., Barnett P.R. & Reynolds SJ. 2000. Inheritance and variation in eggshell patterning in the great tit *Parus major*. — Proc. R. Soc. Lond. B, Biol. Sci. 267: 2469–2473.

Hartley, R. and Zisserman, A. 2003. Multiple View Geometry in computer vision. — Cambridge University Press.

Hoyt D.F. 1979. Practical methods of estimating volume and fresh weight of bird eggs. — Auk. 96: 73–77.

Huang, J.I.E. & Turcotte, D.L. 1990. Fractal image analysis: application to the topography of Oregon and synthetic images. — J Opt Soc Am A Opt Image Sci Vis. 7: 1124–1130.

Kilner R.M. 2006. The evolution of egg colour and patterning in birds. — Biol. Rev. 81: 383–406.

López-de-Hierro, M.D.G. & Moreno-Rueda, G. 2010. Egg-spot pattern rather than egg colour affects conspecific egg rejection in the house sparrow (*Passer domesticus*). — Behav. Ecol. Sociobiol. 64: 317–324.

Mandelbrot, B. 1967. How long is the coast of Britain? Statistical self-similarity and fractional dimension. — Science. 156: 636–638.

Martínez-de la Puente, J., Merino, S., Moreno, J., Tomas, G., Morales, J., Lobato, E., Garcia-Fraile, S. & Martinez, J. 2007. Are eggshell spottiness and colour indicators of health and condition in blue tits *Cyanistes caeruleus*? — J Avian Biol. 38: 377–384.

Maurer, G., Portugal, S.J. & Cassey, P. 2011. Review: an embryo's eye view of avian eggshell pigmentation. — J Avian Biol. 42: 494–504.

Moreno, J. & Osorno, J. L. 2003. Avian egg colour and sexual selection: does eggshell pigmentation reflect female condition and genetic quality? — Ecol Lett. 6: 803–806.

Mulero-Pázmány, M., Barasona, J.Á., Acevedo, P., Vicente, J. & Negro, J. J. 2015. Unmanned Aircraft Systems complement biologging in spatial ecology studies. — Ecol. Evol. 5: 4808–4818.

Ornés, A.S., Herbst, A., Spillner, A., Mewes, W. & Rauch, M. 2014. A standardized method for quantifying eggshell spot patterns. — J. Field Ornithol. 85: 397–407.

Pérez-Rodríguez, L., Jovani, R. & Mougeot, F. 2013. Fractal geometry of a complex plumage trait reveals bird's quality. — Proc. R. Soc. Lond., B, Biol. Sci. 280: 2012–2783.

Portugal, S. J., Maurer, G., Thomas, G.H., Hauber, M.E., Grim, T. & Cassey, P. 2014 Nesting behaviour influences species-specific gas exchange across avian eggshells. — J. Exp. Biol. 217: 3326–3332.

Russell, B. J. & Dierssen, H.M. 2015. Use of Hyperspectral Imagery to Assess Cryptic Color Matching in Sargassum Associated Crabs. — PloS one. 10: e0136260.

Sanz, J.J. & García- Navas, V. 2009. Eggshell pigmentation pattern in relation to breeding performance of blue tits *Cyanistes caeruleus*. — J. Anim. Ecol. 78: 31–41.

Shapiro, L.G. & Stockman, G. C. 2001. Computer Vision. Prentice Hall. Soler, J.J., Navarro, C., Contreras, T.P., Avilés, J.M. & Cuervo, J.J. 2008. Sexually selected egg coloration in spotless starlings. — Am. Nat. 171: 183–194.

Stoddard, M.C., & Stevens, M. 2010. Pattern mimicry of host eggs by the common cuckoo, as seen through a bird's eye. — Proc. R. Soc. Lond. B, Biol. Sci. 277: 1387–1393.

Stoddard, M.C., Kilner, R.M. & Town, C. 2014. Pattern recognition algorithm reveals how birds evolve individual egg pattern signatures. — Nat. Commun. 5: 4117.

Stevens, M., Párraga, C.A., Cuthill, I.C., Partridge, J.C. & Troscianko, T.S. 2007. Using digital photography to study animal coloration. — Biol. J. Linn. Soc. 90: 211–237.

Stevens, M. 2011. Avian vision and egg colouration: concepts and measurements. — Avian Biol. Res. 4: 168–184.

Stevens, M. 2013. Sensory ecology, behaviour, and evolution. — OUP Oxford.

Talabante, C., Gómez, J., Aparicio, A., & Arroyo, B. 2013. Mixed clutches in Montagu's harrier *Circus pygargus* nests: a maladaptive brood parasitism by galliform birds. — Bird Study. 60: 414–416.

Troscianko, J. 2014. A simple tool for calculating egg shape, volume and surface area from digital images. — Ibis. 156: 874–878.

Troscianko, J. & Stevens, M. 2015. Image Calibration and Analysis Toolbox–a free software suite for objectively measuring reflectance, colour and pattern. — Methods Ecol. Evol. 6: 1320–1331.

Wegmann, M., Vallat-Michel, A. & Richner, H. 2015. An evaluation of different methods for assessing eggshell pigmentation and pigment concentration using great tit eggs. — J.Avian Biol. 46: 597–607.

Wilson-Aggarwal J.K., Troscianko J.T., Stevens M., Spottiswoode C.N. 2016. Escape distance in ground-nesting birds differs with individual level of camouflage. —Am Nat. doi: 10.1086/687254

TABLE LEGEND

Table 1. Summarized functionality of SpotEgg organized step by step.

Step	Function	Concept (short description)	SpotEgg Result
1	Linearization	Most usually, cameras transform the output from the sensors in their pixels by using a non-linear function which maximizes details in the highly illuminated areas at the expense of a loss of details in the darker areas. This transformation precludes the output images to be directly used for reflectance measurements. Two options arise depending on whether we can access raw pixel information (which is linear with respect to radiance) or not. If raw data are available, SpotEgg calls DCRAW to create an image whose colour channels are linearly dependent on the incident light power. If not, SpotEgg can still linearize the input image provided that it was stored in a lossless information format, and that it contains at least three neutral colour patches of known reflectance (although the larger the number of neutral colour patches the better)	Linearized image: the three camera channels now become linearly dependent on incident light power.
	Normalization	Pixel values are dimensionless digital numbers. In order to obtain reflectance values images must be scaled using the pixel values from an image sector of known reflectance (e.g. neutral colour reflectance target, or the neutral patches from a colour chart).	An equivalent reflectance image is obtained. Each colour channel corresponds to average reflectance in this channel's spectral band.

2	RoI definition	Definition of the image areas to be analysed. SpotEgg offers two options. (1) Automatic: SpotEgg can find objects which are egg-like shaped without any interaction from the user, provided that they occupy at least 1% of the image area and that the background around them is not too complex. (2) Manual: If automatic detection fails, or if the objects to be analysed are not egg-like shaped, the user can manually draw any number of objects to be analysed in the image.	Creation of files where SpotEgg stores the definition of the different shapes to be analysed within each image. Additionally, a csv report containing information about location, physical properties (area, length, width, etc.) and average reflectances within each region is created.
3	Spot Detection	SpotEgg runs spot detection over each of the regions of interest (RoIs) defined for the current image. It offers the user the possibility of tuning the parameters which rule spot detection through an interactive process. SpotEgg provides two options for spot detection: (1) Simple spot detection (~10-15s for a 10Mpix image): Single pass execution using the parameters defined by the user (2) Advanced spot detection (~95s for a 10Mpix image): Two passes spot detection using the Background Colour Constancy algorithm. Specially recommended when the subject under study is not planar. The first pass provides high likelihood background locations. The second pass divides the region of interest in 25x25 tiles and scales each tile image so that the average reflectance of the high likelihood background locations in each tile matches that of the central area of the subject. When (1) or (2) are completed, SpotEgg employs the detected spots to produce different CSV reports and images. In addition to the colouration (average reflectances in each of the camera's colour channels) SpotEgg obtains Fractal Dimension of the spot pattern without requiring the user to crop the image to be analysed to a rectangular region. Fractal Dimension can be understood as a measure of how the details in a pattern change with the scale at which this pattern is observed.	(1) Black and white image marking the location of the spots. (2) Original Colour image with spots borders marked in red (3) Features report: a CSV file containing information about RoI area, volume (assuming 3D shape obtained by rotation along its major axis), length, width, …, Fractal Dimension of spottiness pattern, average reflectance in each of the camera's colour channel (R, G, and B) for spot, backgrounds, and the whole RoI, and amount of maculation (% over whole RoI area) (4) Spot Report: a CSV file with information about location of each spot, physical size, eccentricity of its shape, and average reflectance in each of the camera's colour channel (R, G, and B) (5) Profile Report: A CSV file with information about average reflectance in each of the camera's colour channel (R, G, and B) for 100 equal width slices across the major axis of the RoI.

FIGURES LEGENDS

Fig. 1. SpotEgg Graphical User Interface (GUI)

Fig. 2. Automatic Chart and Egg Detection from SpotEgg. (a) Reflectance Image + Detected Chart (red dots marking the centre of the patches) from SpotEgg' automatic detection. Note that reflectance images are less attractive for human perception but colourimetry correct. (b) Egg-like shaped RoIs Detector Result over the image produced by the camera (without any linearisation/correction)
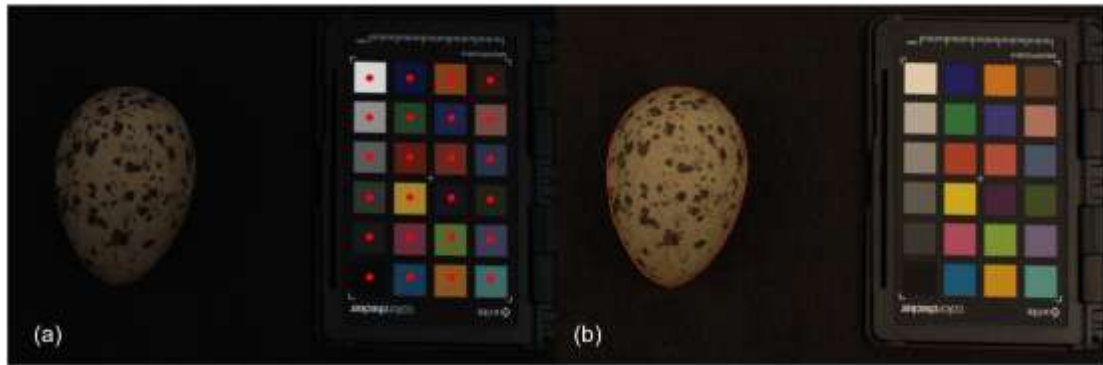
Fig. 3. Improving DCRAW Linearisation in SpotEgg using the grey patches in X-Rite Passport chart. Blue squares (with dashed line) correspond to mean (10 images) reflectance reconstruction errors over the 6 grey patches in the chart when the linear output from DCRAW is simply scaled to the reflectance of one of these grey patches (patch#21, average reflectance 34.16%). Red circles (solid line) display the same error when using the additional linearisation technique implemented in SpotEgg. Standard deviation bars are also plotted.
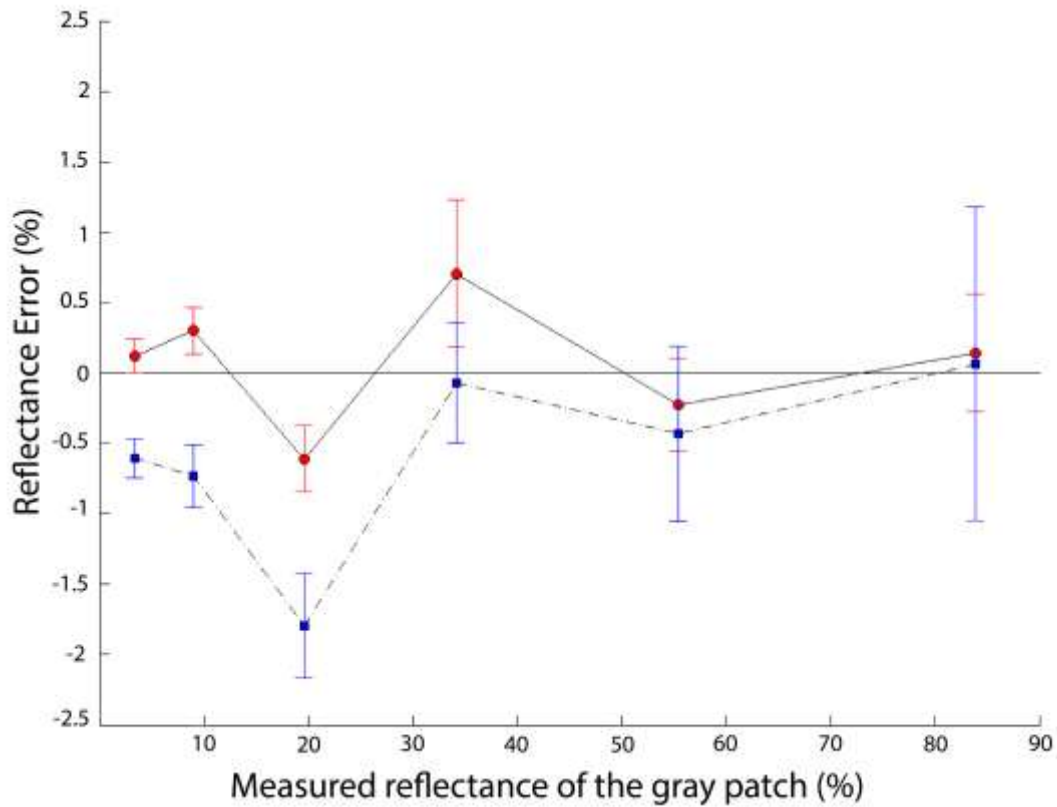
Fig. 4. Global thresholding vs. SpotEgg Algorithm for Spots detection. (a) A Kentish plover egg is photographed on top of an 18% grey standard background under natural lighting conditions. (b) The original picture is manually thresholded using its histogram to determine the spot/background segmentation. (c) The original image is processed by SpotEgg using its locally variable thresholding algorithm.
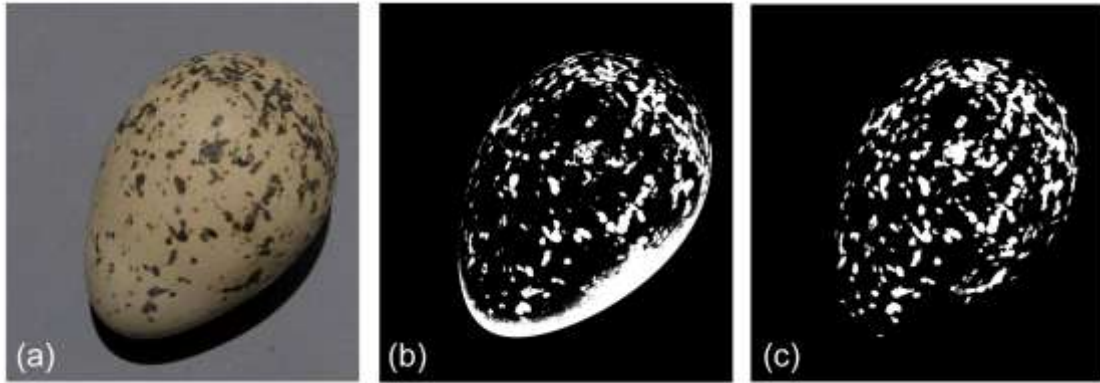
Fig. 5. Reflectance profiles across the major axis. Top image: Profile report for the original reflectance image. Bottom image: Profile report after applying SpotEgg *background constancy* option to corrects the *shading* effect produced by the egg natural curvature.
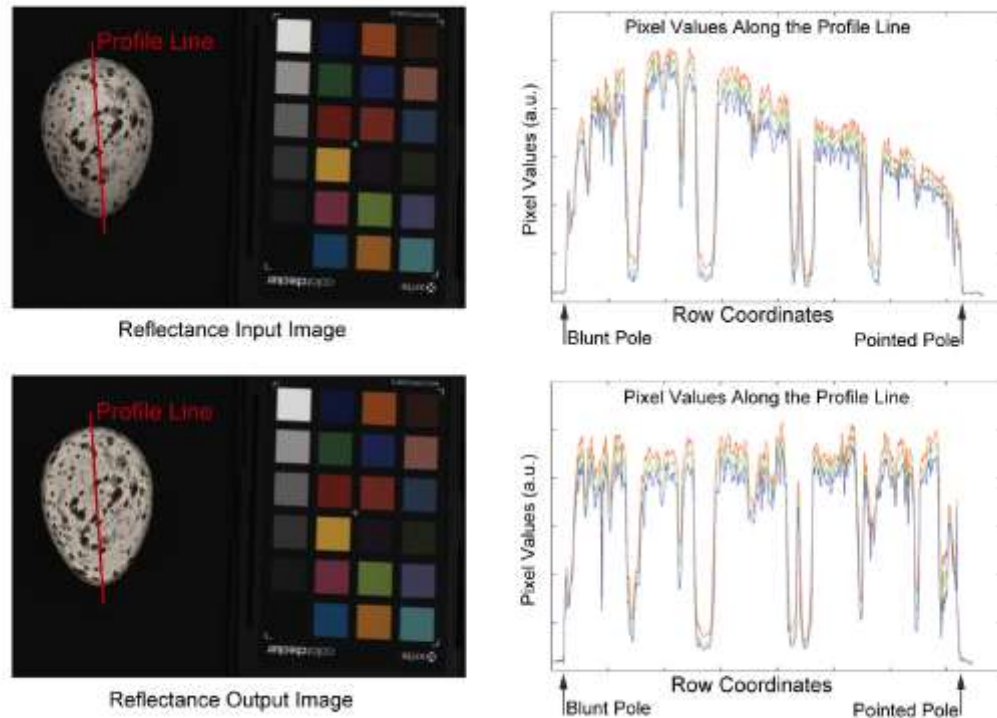
Fig. 6. Background and spot colouration results for a set of 10 Kentish plover eggshells (sample mean ± SE). The figure illustrates the effect of using/not using the background constancy constrain in the execution of the spot detection algorithm. Black lines correspond to runs where background constancy is enabled whereas grey lines correspond to the opposite case.
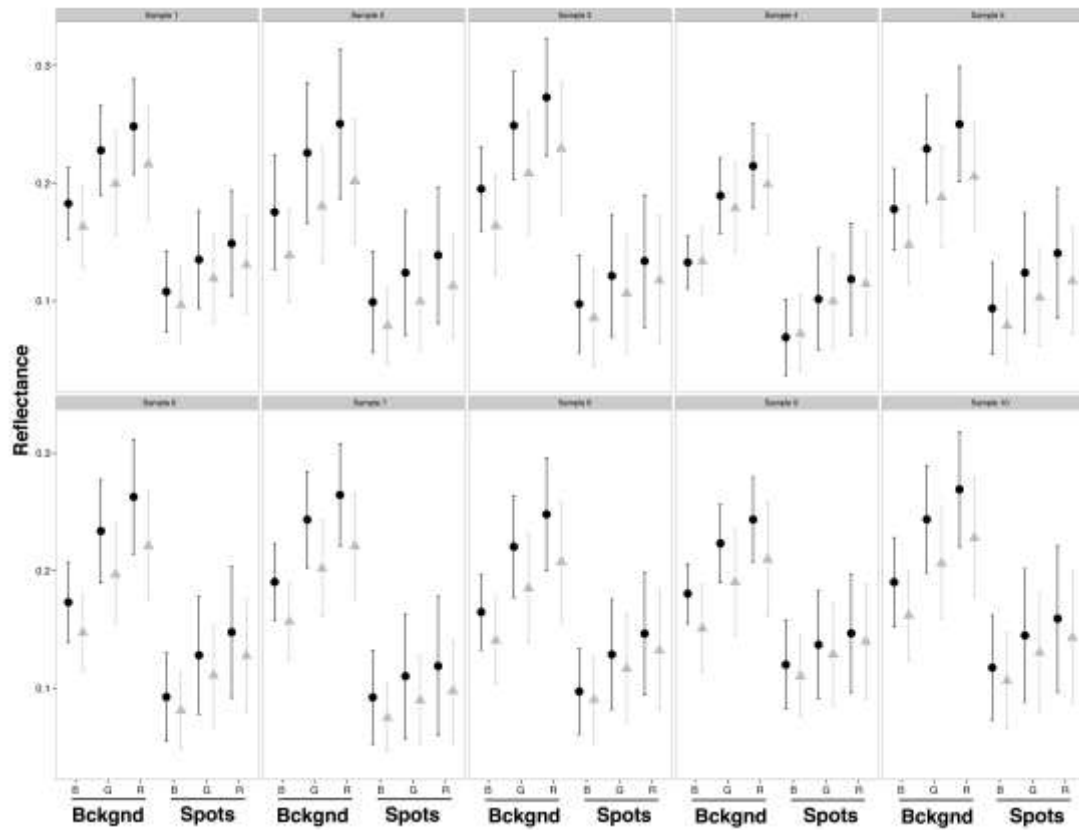
Fig.7. Some Outputs from SpotEgg for a Kentish plover eggshell: (a) Spots' size vs. relative distance to the pointed pole. (b) Histogram of spots sizes (% of eggshell area).