

micaToolbox

User Guide

Descriptions of the tool box functions

PDF generated April 11, 2019

Table Of Contents

AcuityView 2.0	3
Adjacency Analysis	7
Boundary Strength Analysis	9
Chart-Based Cone-Catch Model	11
CIEXYZ and CIELAB	13
Colour Maps	14
Comparative Overview	21
Compile DCRAW for MacOS	23
Creating a calibrated mspec image	24
Creating Cone-Catch Models	29
Creating Presentation Images	30
Direction & Diffuseness of Lighting	31
Display Brightness & Dark Images	32
Essential Equipment	33
Essential Photo-Taking Checklist	34
Exposure	36
Focus and Aperture	38
Full Spectrum Camera	39
GabRat	41
Gaussian Acuity Control	47
Grey Standard	50
ISO	52
Light Source	53
Linearisation Modelling	55
List of Animal Spatial Acuities	59
Local Edge Intensity Analysis (LEIA)	65
Make Your Own: Camera Calibration	69
Make Your Own: Colour & Grey Standard	73
Make Your Own: Spectral Sensitivity & Illuminant Files	79
micaToolbox Installation Instructions	80
Naive Bayes Clustering	81
Negative Values in Cone Catch Images	83
Particle Analysis	84
Pattern Energy (Granularity) Analysis	86
Region Of Interest (ROI) selection tips & tricks	89
RNL Chromaticity XYZ Saturation Images	90
RNL Clustering	95
RNL Ranked Filter	100
Running the QCPA Framework	106
Scale Bar	118
Spectral Sensitivity-Based Cone Catch Model	119
Ultraviolet Photography	121
Underwater Photography	122
UV Lenses	131
UV-VIS filters	133
Visual Contrast Analysis	134

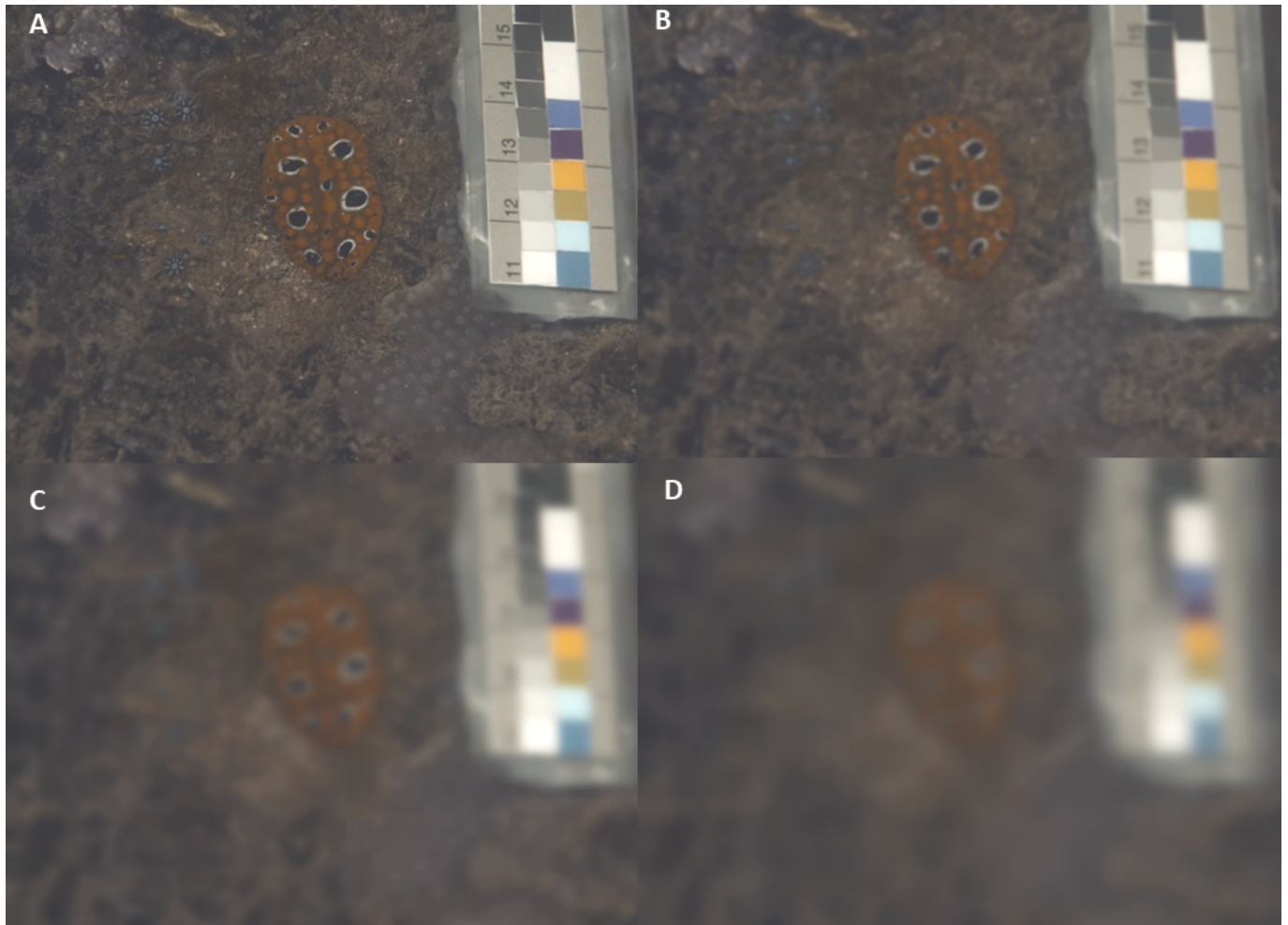
AcuityView 2.0

<http://www.empiricalimaging.com/knowledge-base/acuityview/>

Acuity correction is an important step in modelling receiver vision, and in the [QCPA framework](#), controlling for the receiver's spatial acuity, and the viewing distance. Initially developed in Matlab by Sönke Johnsen (Based on well-known Fourier math), later adapted for R and officially published as a stand-alone named 'AcuityView' by Eleanor Caves we have adapted AcuityView for the QCPA with some major modifications. The tool allows the user to remove the spatial frequencies (pattern information) of an image using a reverse Fourier transform, resulting in an image that only contains information above the resolution set by the visual acuity of a viewer and the viewing distance. Due to its mathematical properties AcuityView only works with rectangular images (the R version could only use square images). The user can either specify the dimension of the original image and the desired outcome in terms of MRA (Minimum resolvable angle) or, by using a size standard in the image in combination of viewer acuity and viewing distance.

This form of acuity control simply removes spatial information which would not be visible to the receiver, however it does so by creating soft, blurred images. Visual systems clearly do not see things as blurred in this way (consider people with bad eyesight who only realise how blurred their vision was after putting on prescription glasses). We therefore developed an element in the QCPA framework which can recover sharp edges from these blurred images through use of the [RNL Ranked Filter](#). QCPA furthermore provides the option of using a Gaussian filter for object oriented acuity modelling.

For convenience we have compiled a [list of animal spatial acuities](#).



Examples of a nudibranch, modelled as seen by a triggerfish (*R. aculeatus*) in 5m depth at various viewing distances modelled using AcuityView. A: No acuity modelling B: 10cm C: 30cm D: 50cm (See suppl. Material for details).

Input Requirements

AcuityView works on any 32-bit stack (e.g. normalised, linear reflectance image, or cone-catch image). You must also know the spatial acuity of the visual system you're modelling ([see the tables of acuity values here](#)), and the angular width of the image or the simulated viewing distance (in which case you also need a [scale bar](#) in the image).

Running AcuityView

Ensure the correct image is selected. You can either run this on its own (*plugins > micaToolbox > QCPA > Acuity View*), or as an option in the [QCPA framework](#) (*plugins > micaToolbox > QCPA > Run QCPA Framework*, and selecting "AcuityView" from the options).

AcuityView for ImageJ

Acuity Settings

Acuity units: Cycles per degree

Acuity value 20

Distance/Angle Settings

Method: Viewing distance

Distance or angle value 2000

If using 'Viewing distance' the image must contain a scale-bar, and the units entered above must match the scale bar units (e.g. mm). All angles are in degrees

Rescale to px per MRA 5

Automatically re-scale the image to a set number of pixels per MRA, 0=off

Background Info

AcuityView removes spatial information that isn't available to a given visual system, but a host of other processing steps may go into changing what an animal actually perceives.

If you use this tool, please also cite:
Caves, E. M. & Johnsen, S. AcuityView: An r package for portraying the effects of visual acuity on scenes observed by an animal. Methods in Ecology and Evolution 9, 793-797 (2018).

Help Cancel OK

Acuity Units

Select either "cycles per degree" or "minimum resolvable angle" depending on which is more convenient (acuity = 1/MRA)

Acuity Value

The number of cycles per degree, or MRA of the visual system (depending on which units you've selected above). See the [list of animal spatial acuities](#).

Method

Viewing Distance: Acuity is calculated from a combination of the known distance to the image plane, and the width of that plane. If this is selected the image must contain a scale bar (created by drawing a line selection along a scale bar visible in an mspec image and pressing "S").

Angular Width of Image: Select if you don't know the scale in the image, but do know the angular width of the image (this will depend on the lens/zoom factor, sensor size, and whether any image cropping has been used. In general this is less likely

Distance or Angle Value

to be useful.

Add the respective distance or image angular width. If "viewing distance" is chosen above the units used here must match the units used in the scale bar. We recommend using millimetres for all distance measurements as default to avoid confusion. If "Angular width" is chosen above then the units here are degrees.

Rescale to px per MRA

This option rescales the output image to uniform number of pixels per minimum resolvable angle. This is convenient for a number of reasons, as it reduces the image dimensions without loss of spatial information. It also speeds up image processing significantly, and ensures all images not only have the same level of spatial information, but have the same effective resolution (i.e. it will make various types of pattern or image analysis more consistent). We recommend 5 px/MRA, more than this is unnecessary, any less could cause loss of spatial information. Set to zero to turn off this feature.

Which acuity control method should I use?

If you will only ever be measuring entire (rectangular) images, and want the fastest possible processing speeds then use the AcuityView method. If you will be measuring (non-rectangular) image sections independently of their surrounding pixels, then use the [Gaussian Acuity Control](#). If you will be comparing whole (rectangular) image sections to non-rectangular sub-sections, then also use Gaussian Acuity Control for all image processing (to keep the processing steps consistent).

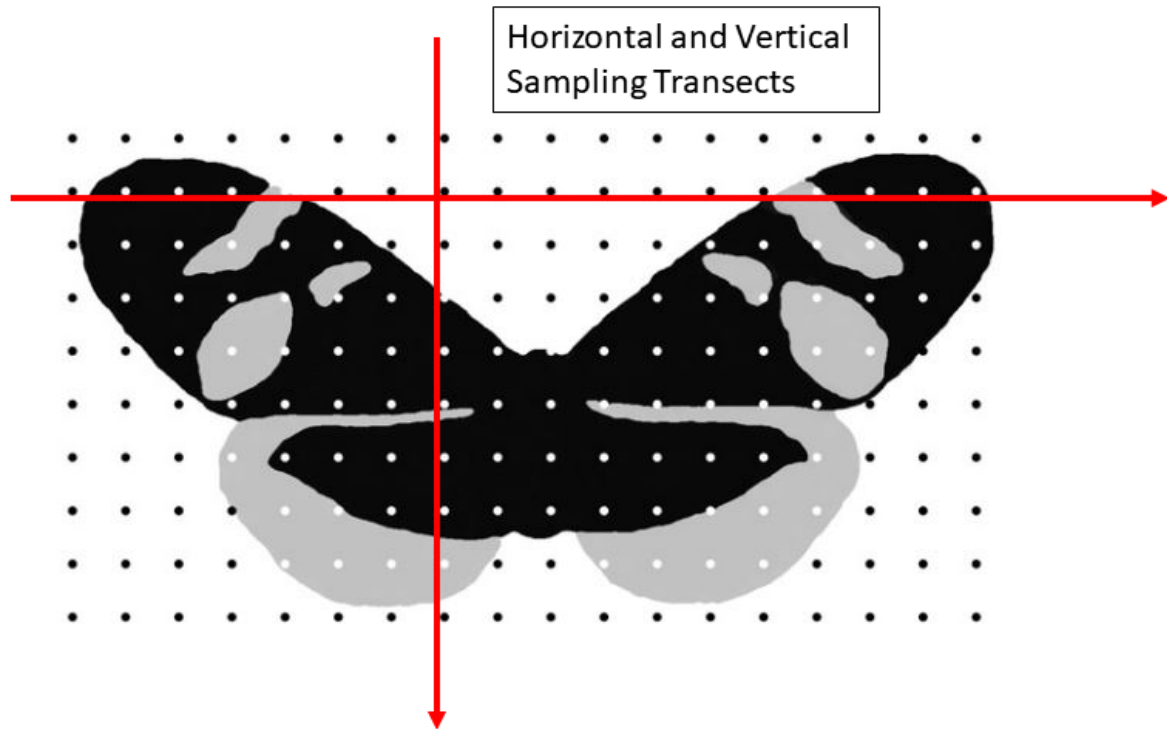
Citation

If you use AcuityView for ImageJ cite the original AcuityView:

Caves, E. M., & Johnsen, S. (2018). AcuityView: An r package for portraying the effects of visual acuity on scenes observed by an animal. *Methods in Ecology and Evolution*, 9(3), 793–797. doi:[10.1111/2041-210X.12911](https://doi.org/10.1111/2041-210X.12911)

Adjacency Analysis

<http://www.empiricalimaging.com/knowledge-base/adjacency-analysis/>



From Endler 2012: An example of a simplified cluttered butterfly consisting of the colour pattern elements 'black' and 'grey' and the background 'white'

The Adjacency Analysis (Endler 2012) is an analytical framework designed to capture colour pattern geometry. It is based on the concept of running horizontal and vertical sampling transects over a segmented image (Fig. 1) where the transitions from one pixel to the next are recorded. This results in a transition matrix (Fig. 2) which can then be used to calculate a vast array of pattern parameters.

Transition numbers			
	1 White	2 Grey	3 Black
1 White	127	24	32
2 Grey		20	44
3 Black			85

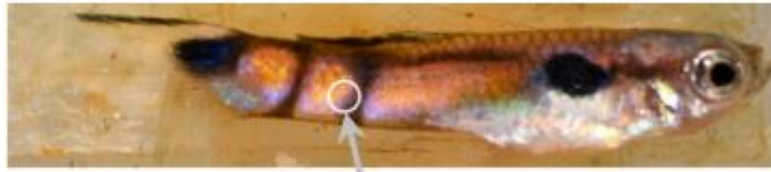
From Endler 2012: An example of a hypothetical transition matrix derived from the segmented image above. The diagonal shows 'non-transition' (Transition where the pattern element stayed the same) whereas the off-diagonal shows the transitions between pattern elements ('Who's next to whom and how much')

Detailed equations and a complete list of pattern parameters can be found in the suppl. material and the MICA manual. The parameter list is by far not exhaustive (Despite being significantly expanded beyond parameters discussed in Endler 2012) and we are happy to discuss new additions to the list.

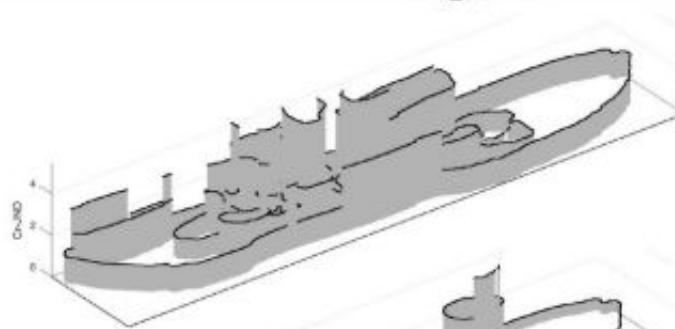
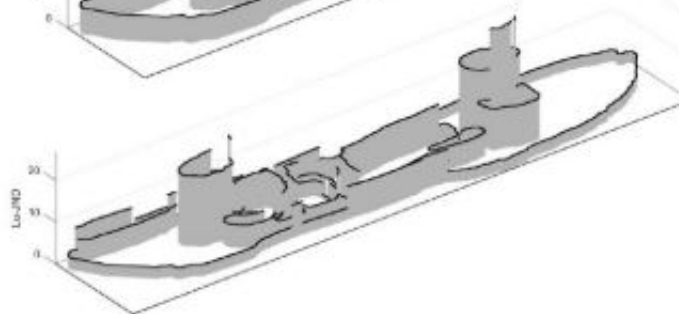
Boundary Strength Analysis

<http://www.empiricalimaging.com/knowledge-base/boundary-strength-analysis/>

The Boundary Strength Analysis (BSA) is essentially a combination and extension of both the Adjacency Analysis and the Visual Contrast Analysis, published by John Endler et al. in 2018 using Matlab. It uses the off-diagonal of the transition matrix to identify the properties of boundaries between colour pattern elements inside a colour pattern (See Adjacency Analysis). The length (relative abundance of off-diagonal transitions) of the boundaries in combination with their chromatic and achromatic contrast can then be used to describe if and how a colour pattern provides visual contrast through its internal borders. This can be statistically represented using a similar statistical approach as the Visual Contrast Analysis (e.g. Coefficients of Variation). We have adapted BSA as it was published but have systematically avoided the use of angular chromaticity metrics and have replaced them with euclidian distances in the log-transformed RNL colour space. We have also added parameters which are not presented in the original publication but may add additional insights. Please see the supplement of our publication for detailed parameter explanations and equations.

(a) Guppy photograph**(b) Part of zone map;
one color/luminance
code per pixel**

15	15	15	15	15	1	1
15	15	15	15	15	1	1
15	15	15	15	15	1	1
15	15	15	1	1	1	1
15	15	15	1	1	1	1
15	12	12	1	1	1	1
15	12	12	1	1	1	1
12	12	12	1	1	1	1
12	12	12	1	1	1	1

**(c) Map of patch
edges****(d) Chromatic ΔS
at each edge****(e) Luminance ΔS
at each edge**

From Endler et al. 2018.

QCPA does currently not allow the user to visualise the boundaries as depicted here (Side on 3D), but it shows the concept. However, when clustering and using the RNL clustering tool in QCPA and using LEIA analysis, the user can recreate a similar visualisation.

Chart-Based Cone-Catch Model

<http://www.empiricalimaging.com/knowledge-base/chart-based-cone-catch-model/>

Using a standard colour chart it is possible to create a cone-catch model. This method requires:

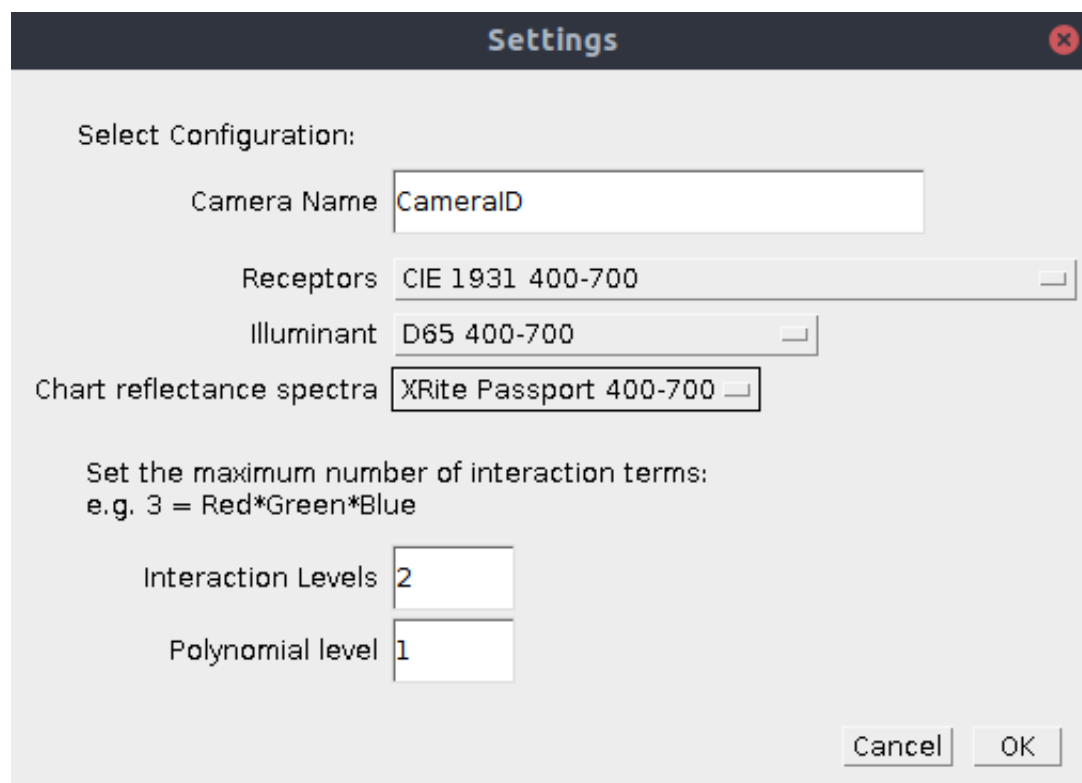
- A diffuse colour chart with precisely known spectral reflectance curves (the toolbox provides these curves for an X-Rite colorChecker passport). A .csv file containing the reflectance curves should be placed in *imagej/plugins/Cone Mapping/Charts*, reading the values in from top left to bottom right, one row at a time (see the existing example for the required format). Note that this method is incredibly sensitive to errors in reflectance intensity, so a standard spectrometer probe is not sufficient for measuring these curves (tiny differences in the distance from the probe to the surface will create unacceptable levels of measurement error). We measured the chart using natural sunlight and a spectroradiometer, although integration spheres may also provide suitable measurements.
- A known illumination spectrum. This method currently requires the chart to be photographed under the same illumination spectrum as that which is being modelled. Natural daylight in all but the deepest forests will conform to the CIE standard illumination curves well enough to use that in almost all situations (e.g. [D65](#)). Any variation from this spectral curve in normal daylight terrestrial conditions is likely to be controlled for sufficiently by the normalisation process.
- Linear, normalised images of the chart. If RAW photos are not available then a [linearisation model](#) for your camera must be made first.

Video Guide

https://youtu.be/blo_IHzQi4U

Process

1. Photograph the chart as described above.
2. Load the photograph as a calibrated multispectral image.
3. To measure the chart values go *plugins > micaToolbox > Camera Calibration > Measure Chart*.
4. Follow the instructions for measuring the chart values.
5. Save the chart measurements (e.g. as a .csv file).
6. Run *plugins > micaToolbox > Camera Calibration > Generate Cone Mapping Model from Chart*.
7. Choose the desired settings, ensuring the wavelength ranges (400-700nm or 300-700nm) are the same for each of the options.
8. Check the resulting model R^2 values are appropriate (ideally >0.98). If the values are much lower then something has gone wrong (linearisation not perfect, chart measurement not perfect, or shadows/lens artefacts degrading the quality of the chart photo).



Settings

Select Configuration:

Camera Name

Receptors

Illuminant

Chart reflectance spectra

Set the maximum number of interaction terms:
e.g. 3 = Red*Green*Blue

Interaction Levels

Polynomial level

Ultraviolet Camera Calibration

This chart-based method can be used for ultraviolet-visible camera calibration. However, you will not be able to use a standard colour-chart (because these are only developed for human-visible colour calibration, and use UV-protective additives, which block UV reflectance). In practice the easiest method is to buy a set of cheap artist's colouring pastels (e.g. Inscribe Soft Pastel Set of 64 pastels). Pastels are ideal because the pigments used have varied and interesting UV reflectance curves (there do not appear to be UV-blocking additives in the sets we have used), plus pastels are very diffuse, and are already in a chart layout. To use these pastels as a colour chart you should clean each pastel well (using a clean piece of paper to wear down one side of the pastel and ensure it is uniform and diffuse). Some pastels contain fluorescent dyes and these must not be used (they will interfere with the light source). Use a UV LED torch to identify any fluorescent pastels and remove them. Then measure the reflectance of each pastel. As above, this measurement needs to be incredibly accurate, and a standard spectrometer probe is not sufficient.

CIEXYZ and CIELAB

<http://www.empiricalimaging.com/knowledge-base/ciexyz-and-cielab/>

[CIEXYZ and CIELAB](#) are colour-spaces designed to roughly fit with human luminance and colour perception.

CIEXYZ images are essentially cone-catch images where the sensitivities are described by the CIE X, Y and Z receptors (which are hypothetical receptors designed to fit with human colour discrimination behaviour, corresponding to longwave, mediumwave and shortwave respectively).

CIELAB images can be created from CIEXYZ, where L describes the luminance channel, A describes a colour opponent channel between red and green, and B describes a second colour opponency channel between blue and yellow. This fits with various theories of how human colour vision works, though is by no means a perfect model.

The micaToolbox can be used to create and convert images to CIEXYZ and CIELAB colour spaces, so that these can in turn be used with all the other features of the toolbox.

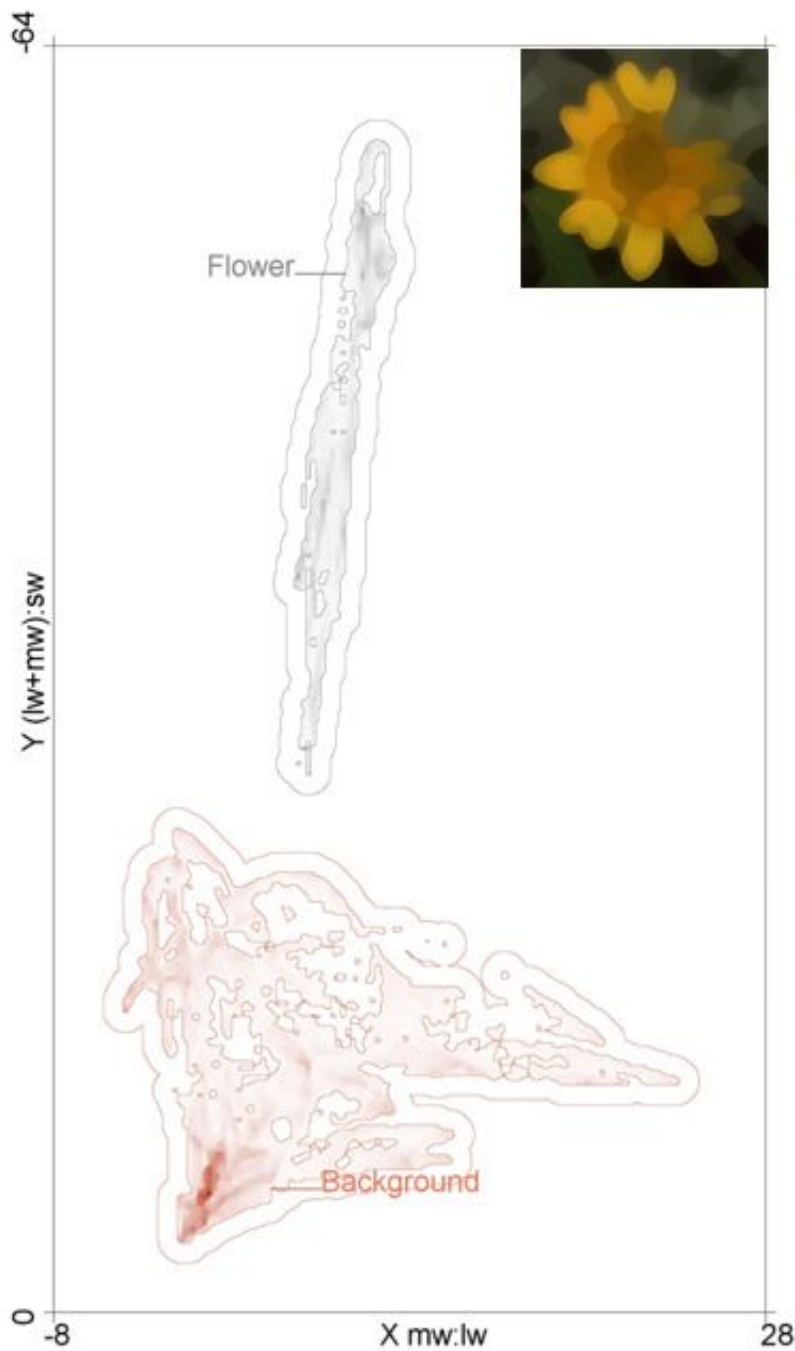
1. [Create a cone-catch model](#) for your camera using the "CIE XYZ 1931 400-700" receptors.
2. Load an [mspec image](#) and run: *plugins > micaToolbox > Convert to Cone Catch* and select the model you've just created to make a CIEXYZ image.
3. To convert to CIELAB run: *plugins > micaToolbox > Image Transform > XYZ to CIELAB 32-bit*

Colour Maps

<http://www.empiricalimaging.com/knowledge-base/colour-maps/>

Colour measurements have historically been plotted and compared as single points in a colour space (such as a Maxwell triangle, or a tetrahedral colour space). The area or volumes occupied by these average colour points has also been used to quantify colour differences between samples. However, natural objects will effectively never have a truly "flat" colour (this would require a perfectly flat surface, perfectly Lambertian reflectance, and perfectly diffuse light source).

We have developed colour maps which can plot the thousands or even millions of different colours present in any image region using the [RNL colour space](#) and [acuity correction](#). In addition, these colour maps can be used to quantify the overlap in colour between measurements. Unlike other methods, these colour maps are non-parametric, so no assumptions are made about the underlying error structure in the measurements (the full range of hues and noise is captured), while using a psychophysically calibrated colour space.



A Colour Map of the non-UV information in figure . The boundary around each ROI pixel cloud reflects 1 σ . The darker a part of the cloud, the more pixels in that ROI are located at that coordinate in the log-transformed RNL colour space. In this case, the flower and its background share no overlap (0%). For tetra-chromatic colour maps the Z-axis is represented as a stack of X&Y maps.

Video Guide

<https://youtu.be/RohhdD-dYbo>

Input Requirements:

We recommend using the [RNL ranked filtered](#) image for measuring colour maps because these preserve sharp edges and gradients, while having controlled for spatial acuity. However, colour maps can be created from any [cone capture image stack](#) which can be either untreated (original resolution), [acuity modelled](#) (blurry), or [RNL clustered](#) images.

To translate cone captures into RNL colour space it is necessary that [channel specific noise levels \(Weber fractions\)](#) are known for all chromatic channels. Colour maps automatically ignore the luminance channel.

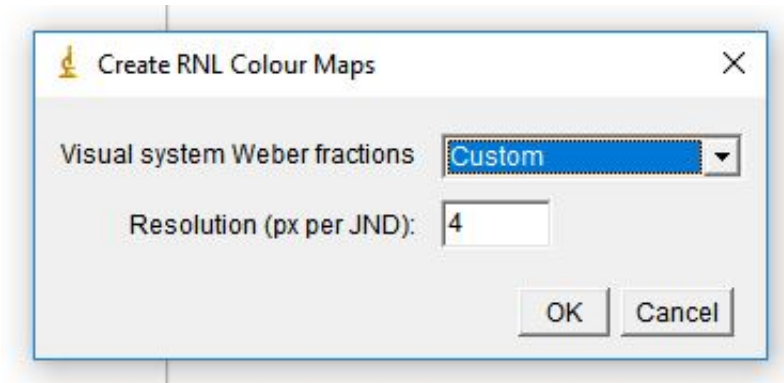
Consider whether your measurement should be made independently of its surroundings. If required, run the [QCPA framework](#) and use the [Gaussian Acuity Control](#) method and specify the ROI which should be measured independently of its background.

Creating Colour Maps:

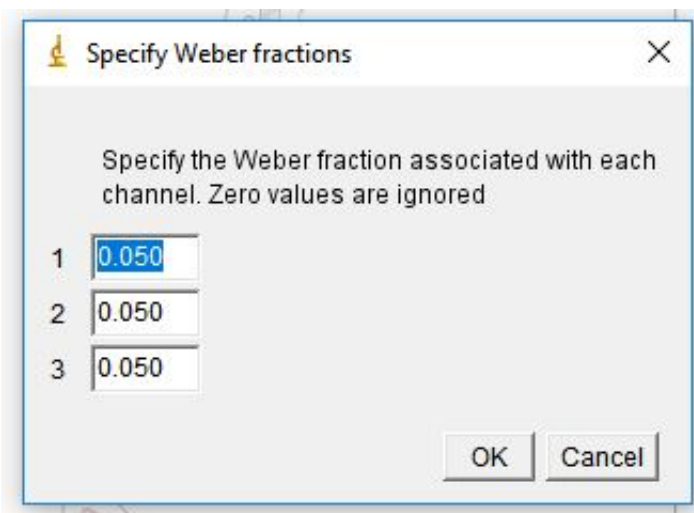
When going to *plugins > micaToolbox > RNL Colour Maps* you will find three options:

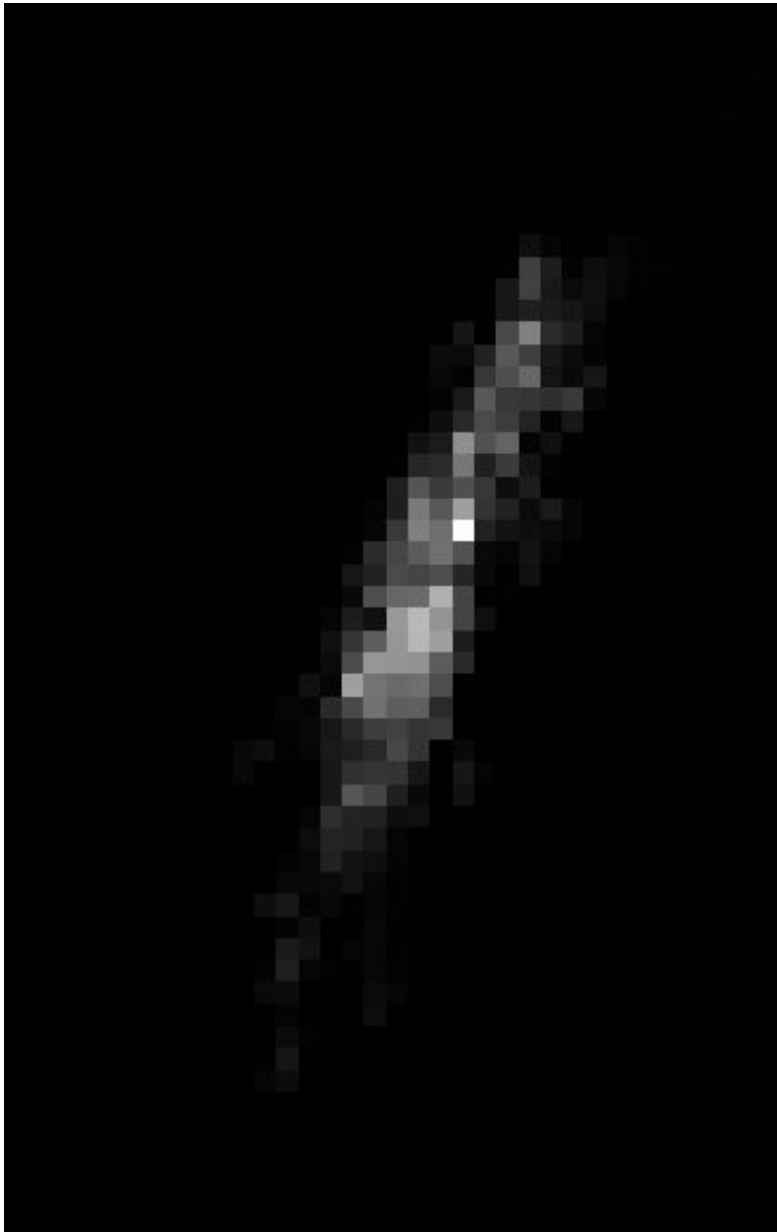
- Create RNL colour maps from ROI
- Combine RNL colour maps
- Plot and compare RNL colour maps

'Create RNL colour maps from ROI' requires either an entire picture or an [ROI that has been saved to the ROI manager](#). It uses the available cone catch image stack to create a single colour map which is a grey scale intensity image that does not contain any axes but already shows the distribution of pixels in the log transformed [RNL colour space](#) with the intensity representing the amount of pixels in an ROI sharing the same coordinates. The user is first prompted to enter the respective Weber fractions that correspond to the chromatic channels in the selected ROI. The can be entered manually or taken from a pre-defined file. The channel names specified in the pre-defined files must match the slice labels in the cone-catch image (it will show an error if they do not match). The user can define the resolution of the colour map in pixels per JND. We recommend the default value of 4 unless you have specific requirements.

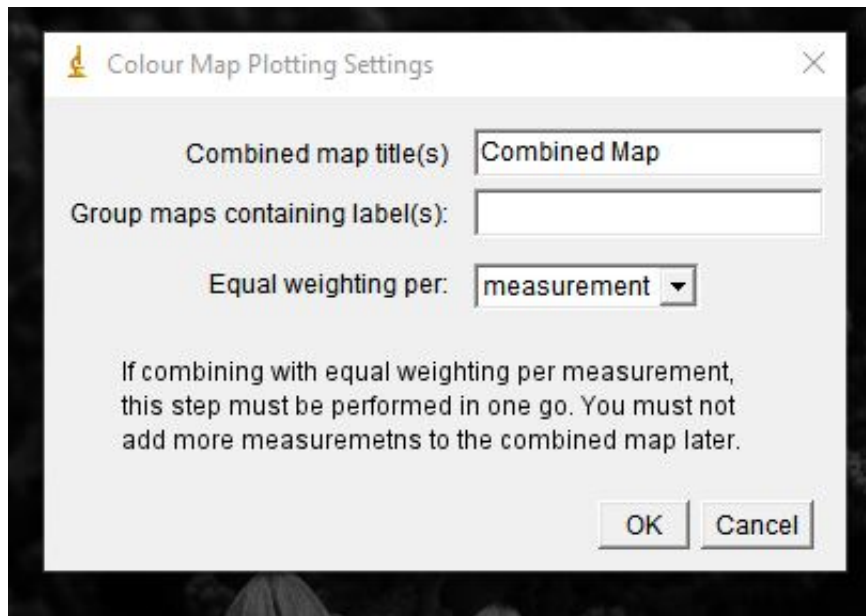


Next, the user is asked to select a folder to which the colour map is stored. Once a folder is selected or created, the user gets asked to enter the Weber fractions (if 'Custom' was chosen). Use this step to check that the luminance channel is ignored or set it to 0 if it is displayed. QCPA is automatically ignores the last layer of an image stack, assuming it to be the luminance channel. Any other channel can also be ignored by setting it to 0 (e.g. if you want to ignore the UV channel). The colour map of each ROI is then saved to the selected folder as well as displayed as grey-scale intensity image.

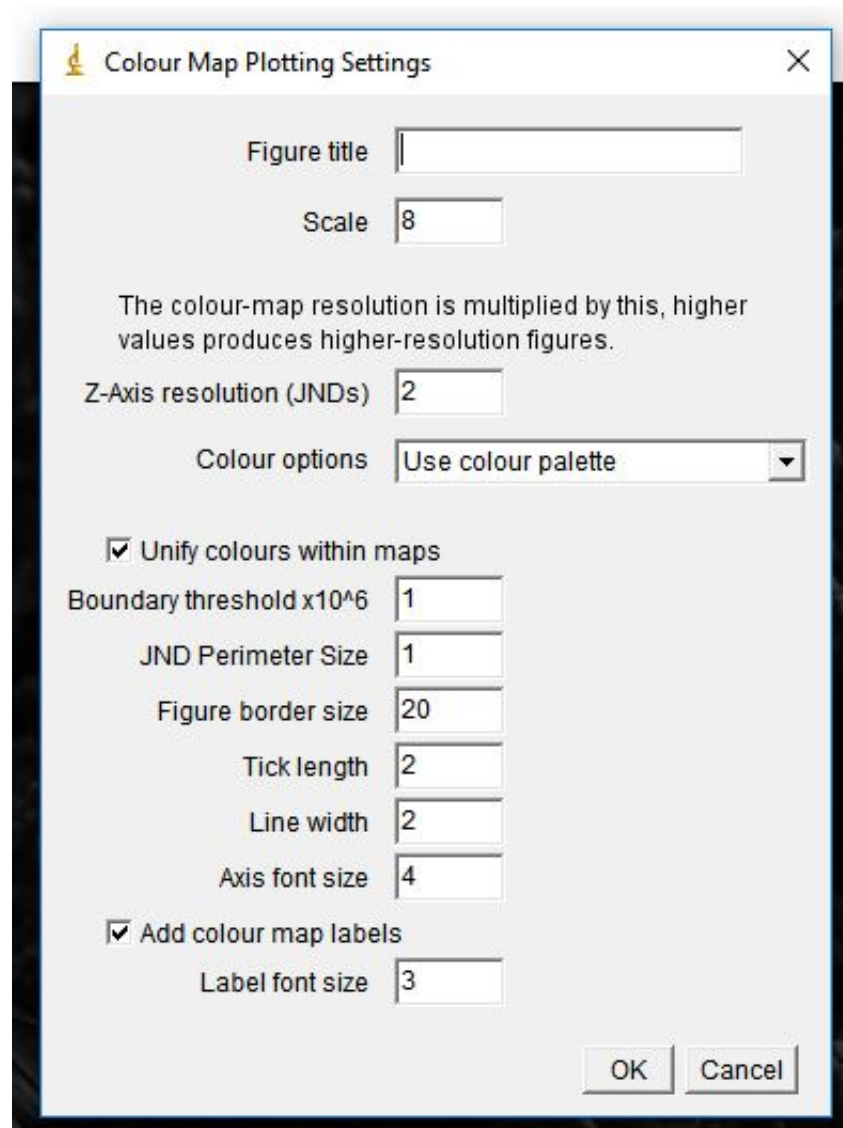




'Combine RNL colour maps' allows the user to combine colour maps from different ROIs. i.e. this may be used when having multiple colour maps of the same body part from different animals.



'Plot and compare RNL colour maps' allows the user to plot single or multiple colour maps and measure overlap in colour space.



The user can define the plotting parameters, particularly the axes properties, a JND perimeter and the colour of the colour maps. Once the parameters are set, the user then needs to choose the folder that contains the colour maps which then creates a figure such as the one at the very top of this article.

Comparative Overview

<http://www.empiricalimaging.com/knowledge-base/comparative-overview/>

	QCPA	MICA	Patternize	PAT-GEOM	PAVO 2.0	Colourvision	NaturePatternMatch
Modelling of non-human Photoreceptor stimulation	✓	✓			✓	✓	
Modelling of visual acuity	+						
Visual Contrast Analysis	+						
Local Edge Intensity Analysis (LEIA)	★						
Adjacency Analysis	+				✓		
Boundary Strength Analysis	✓				✓		
Bandpass based pattern analysis		+					
Alternative pattern metrics	★	★✓+	✓	+			
Landmark based pattern analysis			+				+
Image segmentation using perceptual thresholds	★						
Alternative image segmentation tools	+	+	✓		✓		
Plotting colours in colour space	+				+	+	
Plotting of spatiochromatic information in colour space	★						
Graphical user interface							
Software / Language	ImageJ	ImageJ	R	ImageJ	R	R	C++ (Matlab)

✓ Adapted as per original scientific publications
 + Substantially modified
 ★ Novel concept or methodology

A comparison of the QCPA framework to other existing pattern analyses and frameworks. For patternize see (Bellegheem et al., 2017). For PAT-GEOM see Chan et al. (2018). For PAVO see Maia et al. (2013). For NaturePatternMatch see Stoddard et al. (2014). For Colourvision see Gawryszewski (2018). We would also like to point out approaches by Pike (2018) which operates on a similar principle to NaturePatternMatch and Weller & Westneat (2018) which shares similarities with PAVO.

References:

Bellegheem, S.M. Van,
 Papa, R., Ortiz-zuazaga, H., Hendrickx, F., Jiggins, C.D., Mcmillan, W.O., *et al.* 2017. Patternize : An R package for quantifying color pattern variation.

Chan, I.Z.W., Stevens,
 M. & Todd, P.A. 2018. PAT? GEOM: A Software Package for the Analysis of Animal Patterns. *Methods Ecol. Evol.* 2041–210X.13131.

Maia, R., Gruson, H.,
 Endler, J.A. & White, T.E. 2018. pavo 2 . 0 : new tools for the spectral

and spatial analysis of colour in R. *bioRxiv*, doi: 10.1101/427658.

Pike, T.W. 2018.

Quantifying camouflage and conspicuousness using visual salience. *Methods Ecol. Evol.* 0–2.

Stoddard, M.C., Kilner,

R.M. & Town, C. 2014. Pattern recognition algorithm reveals how birds evolve individual egg pattern signatures. *Nat. Commun.* **5**: 1–10.

Nature Publishing Group.

Weller, H. &

Westneat, M. 2018. Quantitative color profiling of digital images with earth mover's distance using the R package colordistance. *PeerJ Prepr.* 1–31.

Compile DCRAW for MacOS

<http://www.empiricalimaging.com/knowledge-base/compile-dcraw-for-macos/>

DCRAW is used by the micaToolbox for loading the RAW images from a huge range of cameras. Huge thanks to [Dave Coffin](#) for developing this amazing piece of software. Dave Coffin has ceased developing DCRAW, however the project is being continued in other forms (e.g. [libRaw](#)).

DCRAW for Windows and Linux seems to work reliably across different operating systems, however, MacOS sometimes requires re-compilation of DCRAW. If the Toolbox does not work with the supplied files and throws out an error message when opening a RAW image you will need to compile your own DCRAW binary file (this is the bit of software that converts the RAW image into a usable image for ImageJ).

1. Create a new folder on your desktop (e.g. called myDCRAW).
2. Get the original source code for DCRAW from this website: <http://www.cybercom.net/~dcoffin/dcraw/dcraw.c>, in your browser go File>Save Page As.. and save the code as "**dcraw.c**" in your "**myDCRAW**" folder.
3. Open **System Preferences**, Choose the **Keyboard** option and then the "**Shortcuts**" tab. Under "Services", check the **New Terminal at Folder** option. Close System preferences.
4. **Right-click** your **myDCRAW folder** and choose the "**New Terminal at Folder**" option. A terminal will load up, you can check it's in the right location by typing "ls" and hitting enter, it should list dcraw.c in response.
5. Paste the following command into the terminal and hit enter: **llvm-gcc -o dcraw dcraw.c -lm -DNO_JPEG -DNO_LCMS -DNO_JASPER**
6. As you hit Enter, the computer will download Xcode.app, compile and install the source code. Ignore the warning messages on the terminal. When it is done, check the myDCRAW folder and there should be a file (grey with green "exe" label) named 'dcraw'.
7. Copy this **dcraw** file (not dcraw.c) across to your **imagej/plugins/dcraw** folder, and rename it "**dcrawMac**", overwriting the existing file.
8. Check whether it works in ImageJ by going *plugins > micaToolbox > Tools > DCRAW Import* and try to load a RAW image.
9. If it doesn't work it's possibly because the executable dcraw doesn't have permission to run. To give it permission enter this into the terminal (you may need to specify a different path if you've placed imagej elsewhere): **chmod 777 /Applications/ImageJ/plugins/dcraw/dcraw**

Creating a calibrated mspec image

<http://www.empiricalimaging.com/knowledge-base/creating-a-calibrated-image/>

<https://youtu.be/MmnpEIUTI-8>

When making objective measurements from any digital image it is critical to calibrate the image to control for various sources of error. This is one of the core functions of the micaToolbox.

Typical digital images are non-linear, meaning pixel values do not scale linearly with light intensity. RAW digital images preserve the linear data when opened in the correct way (micaToolbox uses DCRAW to preserve linearity). In cases where RAW images are not available (e.g. JPGs or frames from video footage) a [linearisation model](#) is required.

Another critical aspect of image calibration is normalisation, controlling for the inevitable difference in lighting intensity and colour, and image exposure between any two images. In this step, one or more [grey standards](#) of known reflectance are measured in the image, and image pixel values are scaled to reflectance relative to the standards.

The micaToolbox loads linear images to 32-bits per channel to ensure there is no loss of information. Importantly, the toolbox never changes any aspect of the original image, instead it re-loads the mspec images each time without modifying the source. This ensures there is absolutely no destructive modification of any image files, while also reducing file sizes to a minimum. For image archiving purposes we recommend saving the images produced directly from the camera without any form of modification (whether RAW, JPG, or other formats). Converting to any other file format (even DNG, which Adobe recommends) using third-party software is not recommended, and will result in loss of information.

Input Requirements

Follow the [guidelines](#) when taking photos, and ensure images contain [grey standard\(s\)](#) of known reflectance. Correct image [exposure](#) is critical (also, see "Exposure Checking" below). In some cases it is necessary to use the "sequential" method, in which case a photograph of the grey standard(s) can be taken using identical exposure settings (shutter speed, aperture ISO and lighting conditions all identical).

To create a calibrated mspec image run:

plugins > micaToolbox > Generate Multispectral Image

Multispectral Image Generation

Camera type: Visible

Image type: RAW Photo

Grey Standards

Grey standards in: Same photo

☐ Estimate black point (useful with one standard)

Standard reflectance(s): 5.95

☐ Customise standard levels

☐ Standards move between photos

Alignment & Scaling

Alignment: None

Output

Image output: Linear Colour Image

Image Name: test

☒ Rename Image files

Help Cancel OK

Camera Type

This option specifies the type of camera system you're using. most people will only be using "standard" cameras in the human-visible range (first option), but the toolbox can combine any number of channels and photos in a single multispectral image.

Visible: This is the standard option for a photograph in the human-visible spectrum (or three colour channels in a single image).

Visible & UV: Select this option if you're using UV photography so that the toolbox can combine a visible-spectrum image and a UV-spectrum image. This will load and calibrate the three human visible channels (vR, vG, vB), in addition to the two channels which tend to be UV sensitive (uB and uR). Note the convention is to list channels in decreasing peak spectral sensitivity from longwave to shortwave.

Custom: You can create your own camera configuration file if you have a series of photographs across two or more digital images(create a configuration file in ImageJ/plugins/Multispectral Imaging/cameras)

Image Type

This is the type of image format you're using, which changes the level of processing required.

RAW Photo: This is the best option if your camera is able to take RAW photographs, because RAW images are already linear and preserve the highest bit-depth possible from your camera.

Linear Image: This option will very rarely be used unless you've already created linear images somehow (or e.g. are taking linear images from some other system like MATLAB).

Custom Non-Linear: Select this option if you're using non-linear images (e.g. JPG photos), and have already created your own [linearisation model](#). You will be asked to select the model in a subsequent dialog box.

sRGB: If you are not able to create a linearisation model then you could use this function for e.g. images downloaded from the internet, making the assumption the image used the sRGB format correctly. However, in our experience very few modern cameras conform to the sRGB linearisation model (possibly because they have a larger dynamic range than the design specification).

Grey Standards in

This setting describes where the grey standards are located.

Same Photo: This is the standard case where the grey standard(s) are visible in the image being measured. Wherever possible ensure this is the case.

Separate Photo: This is the sequential method (see above) where grey standard(s) can be measured from a second photo, so long as it has identical exposure (shutter speed, aperture and ISO) and identical lighting.

Estimate Black Point

Sensor noise and light bleeding/lens flare can make it difficult for a camera (and subsequent image processing) to work out which pixel value actually corresponds to zero (black). Ticking this box simply allows the calibration to assume that the minimum pixel value in each channel of the image is equal to zero reflectance. Under normal/ideal photography conditions this shouldn't be required. If you notice that dark areas of the image have oddly high reflectance values consider using this method. The caveat is that the image should contain some extremely dark objects/areas, which are equally dark in all channels.

Standard Reflectance(s)

List the reflectance values of the standard(s) in the image. If you're using multiple standards simply separate values with a comma.

Customise Standard Levels

An ideal grey standard is perfectly grey (i.e. has equal reflectance across the entire spectral range the camera is sensitive to). However, sometimes a standard might be dirty, or you need to use a non-grey standard. This option allows you to specify different reflectance levels for the different image channels. You can work out what these levels are by photographing your imperfect/non-grey standard against a true grey standard, and measuring the reflectance of your non-grey standard relative to the true grey.

Standards Move Between Photos

When more than one image is being combined in the mspec image (e.g. visible plus UV), the alignment is performed and then the grey standards are measured automatically from the aligned image. However, if you know that the grey standard(s)

Alignment

moved between these images tick this box to enable manual grey standard selection in each image.

When combining more than one image, how should the images be aligned (this setting is not required for visible-only images)?

None: No alignment (images will be directly overlaid).

Auto-align: Alignment will be attempted automatically, with XY position and scale. Additional options will be displayed if this is selected.

Manual align: You will be given the chance to manually align the images (useful if the automatic version doesn't work, or you want to align only a sub-section of the image).

Image Output

Select how the calibrated mspec image should be displayed:

Linear Normalised Reflectance Stack: A stack of greyscale images, scaled to normalised reflectance. Can be used for measuring directly, and for converting to cone-catch.

Linear Stack: Non-normalised greyscale stack, used it to investigate illuminance (you will very rarely want this format - no normalisation is applied!)

Linear Colour Image: A human-visible colour image with linear, normalised values (often looks dark, but can be measured directly). This is a good option for standard colour images which you want to measure directly. Note, if the image is too dark you can also change the [display brightness](#), which doesn't affect pixel values, or select the non-linear options below:

Non-linear Colour Image: A human-visible colour image with non-linear normalised values (less dark but the pixel values can't be measured). This image display setting is useful when drawing ROIs over dark sections of the image. Note, if the image is too dark you can also change the [display brightness](#), which doesn't affect pixel values.

Non-linear Colour VIS-UV Image: False colour visible/UV combination image with non-linear normalised values (not for measuring). As above, but combining UV and visible channels into a single trichromatic image.

Image Name

Specify a name for the image. Use standard characters (letters, numbers, hyphens and underscores only). We recommend using the label to systematically identify each image and relevant treatments associated with the image (e.g. use underscores to list any number of relevant labels/treatments for each image).

Rename Image Files

If ticked the source photographs will be renamed, with the above image label appended to the filename (the original filename of the image is preserved in the suffix). This is useful when working with a directory containing a large number of images, allowing easy identification of the processed images, and ensuring they sort alongside their mspec configuration files.

Exposure Checking

Using an image with the correct exposure is very important, and can be a major source of error or even produce misleading measurements. During the calibration process you will be shown the image in order to draw a selection area over the grey standard. This image will have an overlay, where blocks of dark solid colour are placed over the image to highlight any regions where the pixels are over-exposed. It is critically important that over-exposed grey standards are not used as this will dramatically affect all calibrated pixel values, so don't measure any grey standard which has blocks of overlaid colour. Once calibrated, you should also avoid measuring any parts of the image which are over-exposed. Note that if an image is only over-exposed in areas which you do not plan to measure, it is still fine to use this image.

Output

The calibration process will display the mspec image using the selected output settings. A small ".mspec" file will be created alongside the original image file, which tells the toolbox how to load this image again.

Creating Cone-Catch Models

<http://www.empiricalimaging.com/knowledge-base/creating-cone-catch-models/>

If your hypotheses depend on the appearance of the object to a specific visual system/model species, or absolute measures of colour are required then cone-catch images are recommended.

This conversion produces images based on the spectral sensitivities of a given visual system. The images are device independent (different cameras, or spectrometers should all produce the same results).

The equipment required will depend on the sensitivities of the visual system in question (see [equipment](#) section).

Additionally, conversion to cone-catch images is essential when using non-linear images which use complex colour spaces (other than sRGB). For example, the DCI-P3 colour space, which is becoming more common (e.g. high-end smart phones now use this as standard).

There are two methods available in the micaToolbox for generating cone-catch models:

- [Chart-based models](#)
 - [Spectral-sensitivity-based models](#)
-

Creating Presentation Images

<http://www.empiricalimaging.com/knowledge-base/creating-presentation-images/>

By default the micaToolbox uses linear 32-bit images because these are required for visual modelling. However, it is often useful to turn these images into images compatible with human colour vision and typical display equipment, requiring non-linear images. If you simply find the image too dark to see certain details, see [display brightness](#).

To create a presentation image, select a 32-bit (e.g. cone-catch) image stack, and run:

plugins > micaToolbox > Image Visualisation > Make Presentation Image

or, for batch processing an entire directory full of mspec images:

plugins > micaToolbox > Image Visualisation > Batch Make Presentation Image

Direction & Diffuseness of Lighting

<http://www.empiricalimaging.com/knowledge-base/direction-diffuseness-of-lighting/>



3D renderings of a crab, showing "perfectly" diffuse light on the left (which in practice can never be achieved in the real world), and with "shiny" spectral surface reflectance on the right. The shiny surface reflections make it look like the crab has white markings, so great care must be taken to understand whether it is important to attempt to remove these reflections, or allow them because they are what a receiver might see.

The direction and diffuseness of the light source can interact with the three-dimensional shape of the target, and with the "shininess" or "glossiness" of its surfaces. A point-source of light (such as the sun, or a small light bulb) is highly directional, so the reflectance measured from a point relative to the grey standard will be highly dependent on its angle relative to the light source as well as the surface reflectance. Therefore, under highly directional lighting the reflectance measured from a surface by your camera will only be accurate if that surface has perfect Lambertian reflectance (is very diffuse), and the angle of the surface is exactly equal to that of the grey standard. Clearly these conditions are almost never true, so care must be taken with directional point light sources. Direct sunlight is arguably a different case because measurements made by a camera in the field will be qualitatively similar to the reflectance measured by an eye, and the lighting conditions are ecologically relevant. However, as long as uniform lighting angles relative to the target are being used, point sources are suitable for comparing reflectance values within a given study.

The effect of the surface angle (i.e. the shape of the target) and its diffuseness can be minimised by using a diffuse light source. Shiny objects and complex 3D objects will therefore benefit most from diffuse lighting conditions. Artificial light sources can be made more diffuse with standard photography umbrellas or diffusers in the human-visible range. However, particular care must be taken when diffusing UV light sources as the diffuser must be UV reflective. White plastic umbrellas and diffusers should therefore not be used for UV sources. Metal-coated umbrellas or sheets of natural white PTFE (Polytetrafluoroethylene, easily bought online from plastic stockists) are suitable for diffusing UV light sources.

In the field, lighting angle and diffuseness can be difficult to control for. If possible, attempt to photograph only under sunny or overcast conditions, not both. If this is not practical, or you are interested in the variation in natural lighting conditions then take note of the lighting for each photograph and take multiple photographs of each sample under all ecologically relevant lighting conditions. Then lighting conditions can be entered into the statistical model during analysis.

Display Brightness & Dark Images

<http://www.empiricalimaging.com/knowledge-base/display-brightness/>

The micaToolbox uses 32-bits/channel for all image processing. These images will generally be linear (unless you've selected [non-linear](#) when loading the image), and as a result will often be very dark in appearance (dark areas of the image are almost impossible to see).

ImageJ displays 32-bit images by specifying the minimum and maximum displayed value, which for calibrated images defaults between 0 and 100% reflectance, and for cone-catch images defaults between 0 and 1.

You can manually alter the display brightness using the "Set Min and Max" tool provided with the toolbox:

plugins > Measure > Set Min and Max

If the image is too dark, reduce the upper value (e.g. try max=50 for calibrated images, or max=0.5 for cone-catch images).

Note that changing the min and max display values does not affect image pixel values at all - measurements are unaffected - and this is a useful tool for seeing darker information in an image.

See also [making presentation images](#).

Essential Equipment

<http://www.empiricalimaging.com/knowledge-base/equipment-essentials/>

For making objective measurements of reflectance (the amount of light reflected by an object relative to a reflectance standard), colour, and pattern all you need is a digital camera that can produce RAW images (or one you have made a linearisation model for), a good quality lens (e.g. with no vignetting – darkening at the image edges), and a grey or white standard (or, in certain circumstances two or more standards light and dark standards). Most pattern analysis requires a scale bar. Lighting also needs to be considered carefully.

The images produced by this set-up will be objective in terms of measuring reflectance levels, being repeatable with the same equipment, and robust against changes in lighting conditions, but are strictly device-specific. Different camera models have different spectral sensitivities, so measurements made with a different set-up will produce slightly different results. The JPG photographs produced by some equipment may also not be objective until transformed to cone-catch images. This is because of the complex colour spaces the images are converted to.

Equipment check-list for objective measurements

Camera

Cameras which can take RAW photos are best, because this format saves linear, uncompressed pixel values. This includes digital SLRs and mirrorless cameras. However the toolbox has tools which support almost any camera which you have made a linearisation model for (even standard smart phones). The camera must cover the biologically relevant spectral range for your hypothesis (e.g. consider whether you need a [UV camera](#)). Ideally use a “prime” (i.e. not a zoom) lens. A high quality lens is desirable as this will minimise vignetting (images getting darker towards the corners), radial distortion (e.g. if you photograph a chequer board the lines in the photo should all be perfectly straight), and chromatic distortion.

Lens

Diffuse grey standard(s)

These can be purchased from most photography suppliers and will specify their own reflectance value. The X-Rite colorChecker passport is convenient as it has a range of grey levels (which can also be used for linearisation modelling and cone-catch modelling). See [here](#) for more information on standards.

Scale Bar

A ruler, or some object of known size is required for various imaging techniques (e.g. pattern analysis, acuity control), see the [scale bar](#) section.

Lighting

It is essential to consider lighting. See the [lighting](#) section for more information.

Colour Chart

Colour charts can be used for [creating a cone-catch model](#), and can be used to create a [linearisation model](#) for cameras which do not have RAW output. You must know the spectral reflectance of the colours in the colour chart in order to create a cone-catch model. The X-Rite colorChecker Passport data are included with the toolbox.

Essential Photo-Taking Checklist

<http://www.empiricalimaging.com/knowledge-base/essential-photo-taking-checklist/>

Taking photographs for objective analysis requires careful consideration. Here are the important aspects:

<i>Spectral Range</i>	If you intend to model the vision of a particular animal, ensure the spectral range of your equipment is appropriate for its visual system (e.g. is a UV system required?) Photograph a colour chart with known reflectance values under the same lighting as your typical subject if you wish to convert to animal vision cone-catch images (this only needs to be done once for a given illuminant/camera/visual system combination).
<i>Format</i>	Photograph in RAW format if possible. The camera white-balance is not important (this is ignored during RAW import). If using non-raw images (e.g. JPGs or video frames), ensure the highest possible quality image is used and be aware that you will need to create a linearisation model for your camera. If your non-raw images are not in sRGB space (e.g. the newer DCI-P3 colour space) you will probably also need to convert to cone-catch before the image colours are objective.
<i>Aperture</i>	Use a fixed aperture between all photos in the study if possible to have a uniform depth of field – use changes in shutter speed to control the exposure.
<i>Zoom</i>	When using a zoom lens only use the maximum or minimum focal length, not an intermediate.
<i>Lighting</i>	Ensure the lighting is suitable and the angle of the grey standard(s) relative to the light source follows a clear, justifiable rule, and that the colour and brightness of the light falling on the standard is as close as possible to that of the sample.
<i>Scale bar</i>	For pattern analysis or scale measurements place a scale bar level with the sample and photograph from a consistent angle (e.g. overhead).
<i>Label</i>	Place a label in each photograph if practical and make a note of the photo number linked to each sample.
<i>Background</i>	If you are placing samples against an artificial background use a spectrally flat colour (e.g. black, grey or white). When photographing samples <i>in-situ</i> in the field their natural background is normally most suitable unless you are comparing the sample directly to its surroundings, in which case it may be desirable to photograph the sample separately against a dark background so that it is measured independently of its surrounds (obviously only if it can be moved without risk of harm to yourself or the sample).
<i>Tripod</i>	Use a stable tripod & remote shutter release if possible (tripods are essential for UV photography).
<i>Exposure</i>	Consider using exposure bracketing to get the best exposure and regularly check photo histograms on the camera to make sure they are well exposed. The bars of the histogram should be as evenly spread from left to right as possible without quite touching the right-hand side.
<i>"Looks good"</i>	Always check the photos “look good”, that everything

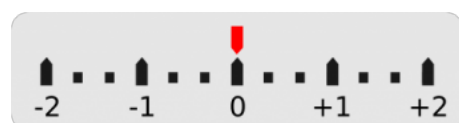
important is in focus, and fills the frame as much as possible

Exposure

<http://www.empiricalimaging.com/knowledge-base/exposure/>

Correct exposure is important for objective imaging, but can be difficult to achieve. Over-exposure will result in the brightest pixels in the image being “clipped” or “saturated”. As pixels become brighter they reach this level and cannot go any higher, which is a big problem for objective photography as it will result in false data being produced. Any pixels that have reached saturation point should not be measured. Under-exposure results in higher levels of noise, reducing the effective dynamic range of the image (so dark things just look black, rather than a range of low values). When generating mspec images the tools highlight any pixels which are over-exposed.

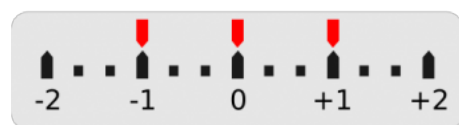
Getting well-exposed photographs is therefore essential. As a simple rule, you should use images with exposures which are as high as possible without quite reaching the top of the range.



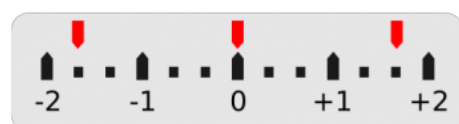
The camera thinks this is the optimal exposure.



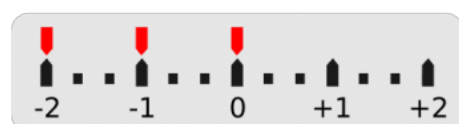
The camera thinks this is one stop (EV) over-exposed. This might produce a good exposure for bright targets photographed against a darker background.



Exposure bracketing. With this setting three photos will be taken with +/- 1 EV. i.e. one photo under-exposed, one over-exposed and one normal.



Exposure bracketing as above, but with a larger EV range. This is recommended in highly unpredictable situations or for very high contrast photos.



Exposure bracketing with +/- 1 EV, but this time they are all shifted down one stop. Use this if the camera tends to over-expose.

Using exposure compensation and exposure bracketing is recommended to ensure that photographs of the ideal exposure are created. Almost all DSLRs provide this function, and many phone camera apps do too ([Open Camera](#) is recommended for Android). The bottom setting in the above image is recommended for most cameras and scenes, though will vary hugely on camera and scene properties.

Most digital cameras provide on-screen histograms and these are very useful for judging whether a photo is over-exposed as you take the photos (though they cannot be relied on entirely because they use non-linear levels and are often conservative). Ideally, separate red, green and blue histograms should be used, but some cameras only provide a grey-scale histogram. Single grey-level histograms can be less reliable for UV photography. If the grey values are calculated as the average of RGB this will under-estimate the pixel levels because the green channel is not sensitive to UV, while the red is most sensitive, meaning the red could be saturated. Nevertheless, once you have become accustomed to your camera's histogram performance it is a very useful tool for judging exposure.

The recommended camera mode depends on conditions. In fixed lab conditions where lighting intensity will not fluctuate substantially the best solution is to use “M” (full manual aperture and shutter speed control), and spend some time working out the best settings, then use those settings for the whole experiment. Note that any changes in the position or intensity of the light

source will likely affect these settings, so check histograms regularly. Some cameras do a very good job of calculating the optimal exposure, in which case “A” mode (aperture priority) can be used. In this mode you set the aperture to use (which should remain fixed across the whole study), and the camera automatically decides on the best shutter speed. You can then alter the exposure value (“EV” or “+/-” symbol on most cameras), telling the camera to under- or over-expose the photo by a set number of exposure values compared to what the camera thinks is best. For example, if you are photographing something against a dark background the camera will generally try to over-expose, so selecting EV-1 will tell the camera to under-expose slightly. DSLR cameras tend to be good at working out the optimal exposure in visible wavelengths, but are poor at working out UV exposures, so manual control (M) is recommended for UV photography. Mirrorless cameras (such as the Samsung NX1000) tend to do a better job in UV as their exposures are based on the light hitting the sensor rather than dedicated light meters, so can often be used in aperture priority mode. In either case though it is often possible to use the exposure values as references that will vary between camera models and will require some experimentation to get used to.

Focus and Aperture

<http://www.empiricalimaging.com/knowledge-base/focus-and-aperture/>

A wider aperture lets more light onto the sensor, so is useful in lower light conditions, however wide apertures also reduce the depth-of-field (so out-of-focus parts of a scene are more blurred) and will tend to exacerbate lens design imperfections (and all lenses have some degree of distortion and imperfection). So as a rule try to keep the aperture constant across all your photos and rely on shutter-speed to change the exposure. Smaller apertures are recommended to get more of the scene in focus. Most lenses have an optimal aperture of around f/8 for the sharpest images (assuming the whole image is in focus), smaller apertures can create blurring through diffraction, but will make out of focus objects sharper.

Larger image sensors (e.g. 35mm "full frame" sensors) will always have a shallower depth-of-field than small sensors. e.g. the small sensors in smart phones have a very wide depth-of-field, meaning everything is more in focus. When modelling animal vision, or in general for objective photography it is better to have a wide depth-of-field.

Full Spectrum Camera

<http://www.empiricalimaging.com/knowledge-base/full-spectrum-camera-conversion/>

Camera sensors are developed for recreating images in the human-visible range, however the sensors themselves (CMOS or CCD) are actually typically sensitive down into the relevant UVA range for modelling animal vision (typically about 320nm), and the same sensors are extremely sensitive to the near-infrared range (up to about 900nm). Almost all cameras have a filter directly in-front of the sensor which blocks out the "unwanted" UV and IR wavelengths. Removal of the filter results in a "full spectrum" camera, which is predominantly sensitive to IR and human-visible wavelengths, but also to the critical UV range. Now filters can be put in front of the lens to photograph in the spectral regions of interest (typically visible and UV).

In order to image in the UV range a filter is required which blocks out all IR and visible-spectrum light. ***It is essential that there is absolutely no IR leakage in the UV filter!*** This can be tested by photographing an IR light source and seeing whether you can see anything in the image (e.g. a television remote control), also healthy plant foliage should appear dark in UV (but bright in IR) when photographed under natural light conditions.

Does the filter need replacing with a new full-spectrum filter?

Many camera-conversion specialists will offer to place a filter back over the sensor which either has full-spectrum transmission (e.g. quartz, which is fine), or will block out some portion of the spectrum, such as UV-only (it is essential to avoid anything like this - these methods require a camera which is sensitive to both UV and the visible-range).

However, these additional filters are not at all necessary, and may even increase internal reflections in the camera (reducing dynamic range). They primarily offer mechanical protection for the sensor (so just be careful when cleaning the sensor and you'll be fine without one of these filters). The other function these filters can provide is restoring the refractive properties of the camera system given the designed back-focal-length of the lens. So it is possible that, for example, focussing to infinity might not be possible if the sensor location is not readjusted. DSLRs and mirrorless cameras have sensors mounted on adjustable plates, which can typically be used to compensate fully for this effect.

Where to acquire a full-spectrum camera?

Ready-converted

Ready-converted full spectrum cameras can be bought online. e.g. check sites such as eBay using the search term "full spectrum UV camera". They are used by astrophotographers, forensics experts, and (ridiculously) for ghost hunting. Note that if you see a "ghost" in your UV images, but not your visible-range image, this is due to internal reflections in the lens, which are often more extreme in the UV and IR ranges because the lens coatings are not optimised for these wavelengths.

Have your own camera converted

A number of businesses sell camera conversion as a service. e.g. [ACS](#) (empirical imaging has no affiliation with this company).

DIY conversion

Some cameras can be converted manually with nothing but a small screwdriver (though note that there is a very real possibility of damaging the camera - some are very tricky to take apart - empirical imaging is not responsible for any damage you may cause to your camera). Search the internet for a "teardown" of your camera model. [Here](#) is an example of the Samsung NX1000 (which even given its age, is an excellent UV camera).

Recommended Cameras

Sony Alpha Range

These are excellent mirrorless cameras. Mirrorless systems are actually preferable over standard DSLRs for full-spectrum photography because they perform the metering based on sensor measurements (rather than other optical systems which are not able to work in the UV range). Mirrorless cameras therefore tend to have better automatic exposure abilities in the UV range. The Sony A7 range is an excellent choice, though the cropped-sensor versions will also be suitable.

Samsung NX range

As above, but these can often be bought extremely cheap second hand. So that you might feel more confident attempting to do a DIY full-spectrum conversion.

GabRat

<http://www.empiricalimaging.com/knowledge-base/gabrat/>



GabRat is a method developed for measuring the level of edge disruption around a given target. The method uses Gabor filtering, and measures the ratio (hence "Gabor -ratio" > GabRat) of the targets' "true" outline edge intensity, compared to the intensity of any orthogonal "false edges" (which run at 90 degrees to the target's outline). A higher ratio of false edges which run counter to the target's outline create higher levels of edge disruption, and was found to be by far the best predictor of detection times in an experiment where humans were searching for camouflaged items ([Troscianko et al. 2017](#)).

Input Requirements

GabRat can be used on almost any image: [calibrated mspec images](#), [cone-catch images](#), [CIELAB](#) or [RNL chromaticity \(XYZ\)](#) images are recommended. The latter two are interesting because they allow you to measure chromatic edge disruption. There are two version of the tool. The first will go through each ROI in the ROI manager and provide GabRat measurements for each ROI in each image channel separately. The second version will only measure the currently selected ROI in the current image channel. If the processing throws up an error, it is likely because the edge of the target is too close to the edge of the image given the scale of the Gabor filter being used. Reduce the sigma value, or ensure any targets are not too close to the image boundaries.

The input image should ideally be scaled to match the viewer's peak acuity contrast sensitivity (note this is different from an animal's spatial acuity limit). To do this, find the acuity at which you expect the peak contrast sensitivity (e.g. see Souza et al. 2011 for a list of species peak contrast sensitivity values). Then scale the image using the [QCPA framework](#), selecting only the first acuity correction step (either AcuityView or Gaussian acuity control). Ensure the output image is scaled to 5 pixels per minimum resolvable angle. This will automatically scale the image and all of its ROIs, creating a blurred, scaled image (do not

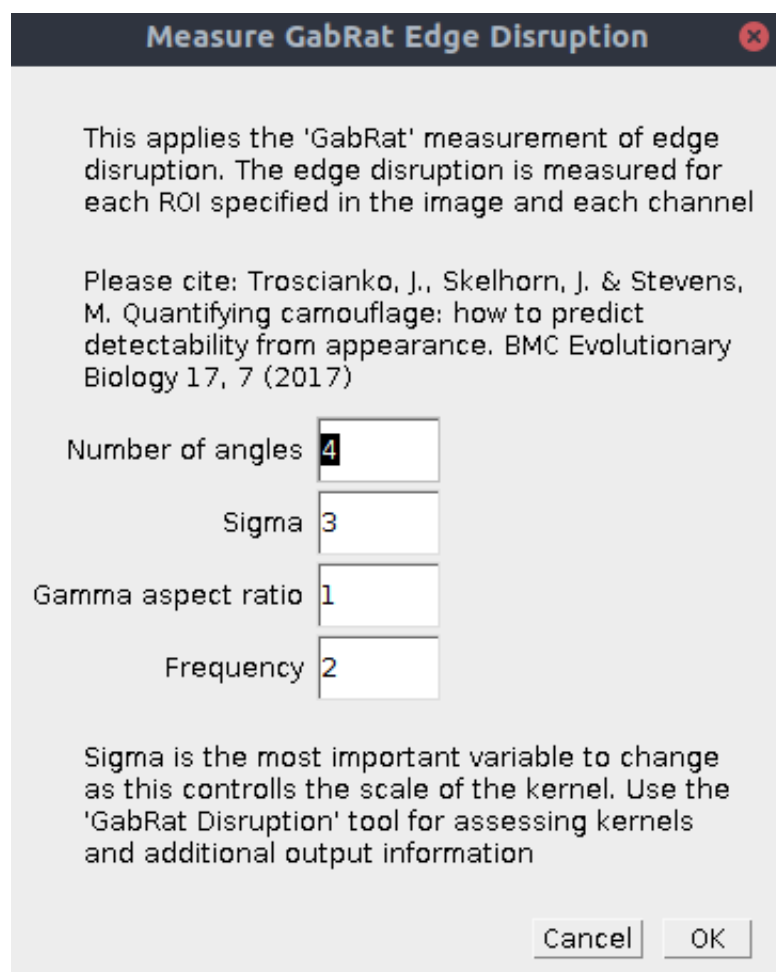
apply RNL ranked filter or clustering). Now run the GabRat filter on the acuity controlled image with a sigma matching the px/MRA value (i.e. 5).

Run GabRat Measurement Tools

The recommended method for running GabRat measurements is with:

plugins > micaToolbox > Image Analysis > GabRat Disruption ROIs

You should have already specified any targets you wish to measure by creating ROIs for each of them. The tool will measure the GabRat of each target across all image slices.



However, in order to access some additional options (such as outputting the kernels, masks, and outline disruption measurements) you can also run:

plugins > micaToolbox > Tools > Gabrat Disruption

This will only measure the current slice in an image stack, but will give you extra output for debugging, or visualising the GabRat filter (see table below).

GabRat Disruption Measurement

Gabor Filter Settings

Number of angles

Sigma

Gamma aspect ratio

Frequency

Output Images

☐ Gabor Kernel

☐ Edge and Mask

☐ GabRat

Label

Number of Angles

Sigma

Specify the number of angles for the Gabor filter. The default of 4 is generally sufficient and this will measure angles in the vertical (\updownarrow), diagonal (\nearrow), horizontal ($-$) and other diagonal (\nwarrow). Sigma is generally the only number you will want to change. This specifies the overall scale of the Gabor filters (larger sigma = larger filter/kernel). Ideally you would match the sigma size to the receiver's peak contrast sensitivity function if this is known (note, this is different to the species visual acuity limit, see Souza *et al.* 2011). We recommend using an image which has been scaled to the receiver's [spatial acuity](#) or one

*Gamma Aspect Ratio**Frequency**Output Gabor kernel**Edge and Mask*

GabRat

which has been scaled to a specific size (number of pixels per millimetre). To measure the disruption at the visual acuity limit selected, use a sigma value equal to the number of pixels per minimum resolvable angle chosen when scaling the image. A value of 5 pixels per MRA is recommended, in which case a sigma of 5 is also recommended.

This changes the shape of the Gabor filter, use the "GabRat Disruption" tool (without the ROI) to output the kernel and play with the shape. The default value of 1 is recommended.

This changes the frequency of the Gabor filter. A higher number will have more waves in the kernel. 2 is recommended, and fits with the receptive fields of edge detector neurophysiology.

This and the following options are only available in the standalone version of the tool (second tool shown above). If ticked it will output a stack of the Gabor kernel at different angles.

If ticked an additional stack showing the mask (an image showing where the target is and is not), and outline of the target ROI will be shown.

If ticked this will output a slice showing the GabRat value at each point around the edge of the target ROI. This can be used identify specific locations with low or high disruption, and to create a false-colour image of the GabRat intensity around the edge of the image (like the image at the top of this page). To create this presentation image, use a Gaussian blur on this output image equal to the sigma value used above. Then set the min and max appropriately and choose a suitable LUT (lookup table). Then overlay this image on the original (e.g. copy the GabRat image, then set the paste mode to "Add", and paste over the original greyscale image).

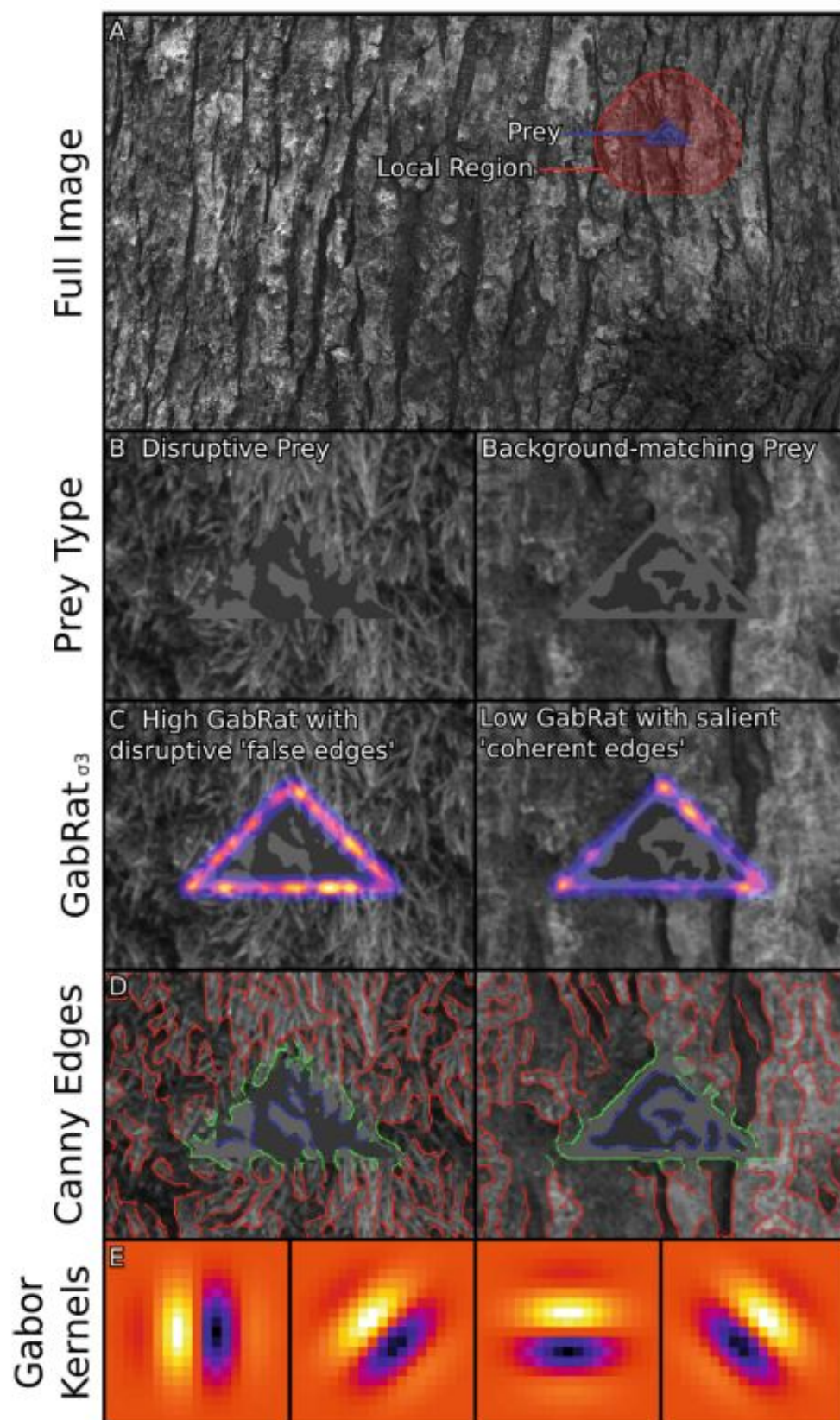
Output

If using the standard tool, the GabRat value will be measured for each slice in the image. If using the standalone tool (second tool above) you have the option for extra output images, useful for debugging, interpreting, and visualising the output.

Citation

If you use any GabRat measurements please cite:

Troscianko J., Skelhorn, J., & Stevens M. (2017). Quantifying camouflage: how to predict detectability from appearance. *BMC Evolutionary Biology*. 17:7. – [link \(open access\)](#) – [Blog Post](#) –



References

Souza, G. da S., Gomes, B. D., & Silveira, L. C. L. (2011). Comparative neurophysiology of spatial luminance contrast sensitivity. *Psychology & Neuroscience*, 4(1), 29–48. doi:[10.3922/j.psns.2011.1.005](https://doi.org/10.3922/j.psns.2011.1.005)

Gaussian Acuity Control

<http://www.empiricalimaging.com/knowledge-base/gaussian-acuity-control/>

Acuity correction is an important step in modelling receiver vision, and in the [QCPA framework](#), controlling for the receiver's spatial acuity, and the viewing distance. This method uses a Gaussian convolution to eliminate spatial information from an image to simulate the spatial acuity of a given receiver. The principal is identical to [AcuityView](#), however the processing methods are quite different. AcuityView uses fast Fourier transforms (FFT) to apply the modulation transfer function (i.e. blur the image), and these methods can only be applied to rectangular images. It is often necessary to measure specific (non-rectangular) parts of an image independently of their background (e.g. a butterfly in a museum case, where you don't want to measure the background at all). Under these circumstances FFT-based methods, and even methods using standard "Gaussian filters" (such as those used by MATLAB, Photoshop and base ImageJ) are inappropriate because they use a more computationally efficient separable convolution function, which require rectangular image selections.

We therefore wrote our own 2D Gaussian convolution function which can be applied to ROIs of any size and shape, to blur images appropriately independently of their backgrounds. The method is therefore less computationally efficient than alternatives, however the tools pre-scale the images to a suitable number of pixels per minimum-resolvable-angle (MRA) so that in practice this step is unlikely to take more than a few seconds for a typical image.

For convenience we have compiled a [list of animal spatial acuities](#).

Which acuity control method should I use?

If you will only ever be measuring entire (rectangular) images, and want the fastest possible processing speeds then use the [AcuityView](#) method. If you will be measuring (non-rectangular) image sections independently of their surrounding pixels, then use the Gaussian Acuity Control. If you will be comparing whole (rectangular) image sections to non-rectangular sub-sections, then also use Gaussian Acuity Control for all image processing (to keep the processing steps consistent).

Input Requirements

The Gaussian acuity control works on any 32-bit stack (e.g. normalised, linear reflectance image, or cone-catch image). You must also know the spatial acuity of the visual system you're modelling ([see the tables of acuity values here](#)), and the angular width of the image or the simulated viewing distance (in which case you also need a [scale bar](#) in the image).

Running Gaussian Acuity Control

Ensure the correct image is selected. You can either run this on its own (*plugins > micaToolbox > QCPA > Gaussian Acuity Control*), or as an option in the [QCPA framework](#) (*plugins > micaToolbox > QCPA > Run QCPA Framework*, and selecting "Gaussian Acuity Control" from the options).

Gaussian Acuity Control

Acuity Settings

Acuity units:

Acuity value

Distance/Angle Settings

Method:

Distance or angle value

If using "Viewing distance" the image must contain a scale-bar, and the units entered above must match the scale bar units (e.g. mm). All angles are in degrees

Rescale to px per MRA

Automatically re-scale the image to a set number of pixels per MRA

Specify ROI

ROI:

Acuity Units

Select either "cycles per degree" or "minimum resolvable angle" depending on which is more convenient (acuity = 1/MRA)

Acuity Value

The number of cycles per degree, or MRA of the visual system (depending on which units you've selected above). See the [list of animal spatial acuities](#).

Method

Viewing Distance: Acuity is calculated from a combination of the known distance to the image plane, and the width of that plane. If this is selected the image must contain a scale bar (created by drawing a line selection along a scale bar visible in an mspec image and pressing "S").

Angular Width of Image: Select if you don't know the scale in the image, but do know the angular width of the image (this will depend on the lens/zoom factor, sensor size, and whether any image cropping has been used. In general this is less likely to be useful.

Distance or Angle Value

Add the respective distance or image angular width. If

Rescale to px per MRA

"viewing distance" is chosen above the units used here must match the units used in the scale bar. We recommend using millimetres for all distance measurements as default to avoid confusion. If "Angular width" is chosen above then the units here are degrees.

This option rescales the output image to uniform number of pixels per minimum resolvable angle. This is convenient for a number of reasons, as it reduces the image dimensions without loss of spatial information. It also speeds up image processing significantly, and ensures all images not only have the same level of spatial information, but have the same effective resolution (i.e. it will make various types of pattern or image analysis more consistent). We recommend 5 px/MRA, more than this is unnecessary, any less could cause loss of spatial information. Set to zero to turn off this feature.

ROI

This method can be applied to the entire image, or just a single ROI. Unlike almost all other methods for blurring images, this tool can blur ROIs completely independently of their surrounding pixels. Select the desired ROI to process here.

Output

A new image stack will be produced where each slice/channel in the input image has undergone the same level blurring and rescaling appropriate to the acuity of the selected visual system. Contrast at the spatial scale of the MRA will now be reduced to 2% of the original amplitude.

Grey Standard

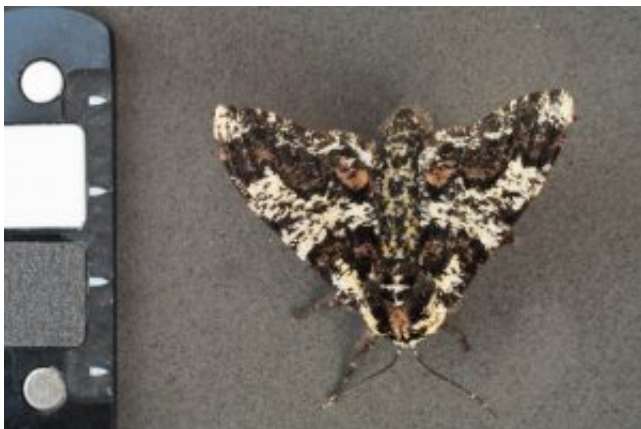
<http://www.empiricalimaging.com/knowledge-base/grey-standard/>

At least one grey standard of known reflectance is required in each photograph (or in a separate photograph taken under exactly the same conditions and with identical camera settings).

Greyness and Diffuseness

The standards used should ideally have equal levels of spectral reflectance across the entire range the camera is sensitive to. e.g. a 40% standard should have 40% reflectance at all wavelengths from around 420-680nm for a standard human-visible camera. In practice this is never achieved, and we normally make an assumption that something is grey, even when it isn't perfectly grey (e.g. even Spectralon grey standards vary by a few percentage points across their working range).

Standard diffuseness is also extremely important. While no standard is likely to have "perfect" [Labertian reflectance](#), it's important that standards are extremely diffuse, so that the angle of lighting and photography don't affect the measured reflectance.



When photographing roughly 2D objects the grey standard(s) should always be in the same plane as that object and ideally the same distance from the light source.

Postion of grey standards

The position and angle of the grey standards is critically important for objective measurements. The light falling on the standard should match as closely as possible the light falling on the object being measured with respect to angle relative to the light source, intensity, and colour (see lighting [direction & diffuseness](#)). If only one standard is being used then a value around 20 to 50% reflectance is appropriate for most natural objects, as a rule the ideal standard should have a reflectance near the upper range of your sample. For example, when measuring very dark objects a lower reflectance standard can be used (e.g. 2 to 10%). If you use a very light standard (e.g. 99%) to measure dark objects, then you will need to use a camera exposure value which is not optimal for the scene in order to ensure the white standard is not over-exposed. This increases the noise in estimates of low reflectance objects.



When photographing more complex 3D objects, or when the angles you photograph from cannot be predicted, then the grey standard(s) should be angled relative to the light source. In the field the grey standard should normally be level with the ground so that it collects light from the whole sky, not angled directly towards the sun.

Multiple standards (e.g. white and dark grey) allow the software to overcome some optical glare (light bleeding onto the sensor), or photographing through reflective or hazy materials such as underwater, and through the surface of water (or through the side of a fish-tank). When photographing from the air into water the grey standards must be positioned next to the target (e.g. both standards underwater as close to the sample as possible), and the background sky or ceiling being reflected by the surface of the water must be uniform (e.g. this won't give correct values if there are any visible ripples or if you can see blue sky and white clouds reflected in the water surface).

When measuring 2D objects the surface of the standard should be in the same plane as the target. For complex objects the standard should normally be angled relative to the light source (e.g. flat on the ground). Whatever rule is used must be kept constant throughout the experiment and between treatments.

Remember that light “spills” off nearby objects. Avoid photographing the grey standards near brightly coloured objects that you have introduced to the scene. This includes objects colourful in UV, many surfaces that look white to us absorb UV.

ISO

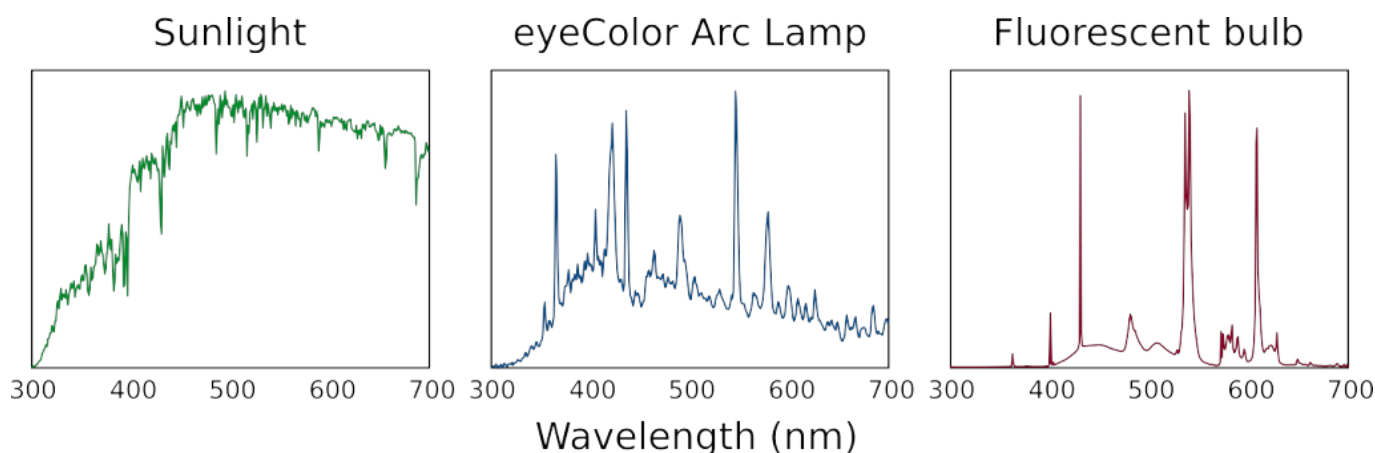
<http://www.empiricalimaging.com/knowledge-base/iso/>

ISO refers to the level of sensor gain, taking the name from "film speeds" back when we photographed on analogue film. Higher ISOs make the camera sensor more "sensitive", allowing for photography in lower light levels. However, increasing the ISO beyond the camera's "native" ISO (this varies between model, but is often around 200-400) will increase the level of noise, which in turn reduces the dynamic range of the images. ISO should therefore ideally be kept constant throughout your entire study. Different models of camera vary substantially in the quality of noise produced at different ISO levels, so comparisons between cameras should be treated with caution (e.g. the bandpass pattern analysis will be very sensitive to noise levels in images at the very high spatial frequencies).

Light Source

<http://www.empiricalimaging.com/knowledge-base/lighting/>

Photography is entirely dependant on light and the way it bounces off or through objects, so is an important consideration before starting data collection. In an ideal world, all photographs that you want to compare should be taken under uniform lighting conditions, however in practice this can be impossible to achieve. The emission spectrum of a light source affects colour reproduction, which can generally be controlled for by using a grey standard (see below). However, some emission spectra are so “spikey” that the spikes can interact with reflectance spectra in ways that cannot be controlled for with a grey standard (e.g. causing metamerism). Fluorescent light sources (energy saving bulbs, tube lights etc.) are the worst and should be avoided if at all possible. Flickering can also be problematic. If your photographs have odd looking horizontal banding this is caused by a flickering light source and a much longer exposure (shutter speed) should be used to eliminate this effect if no other source is available. Also be aware that many artificial light sources change their spectra as they warm up (over a few minutes), could change with voltage fluctuations in the mains or battery supply, and can change as the bulb ages. So always try to get a grey standard into each photograph if possible rather than relying on the sequential method (taking a photograph of the standard before or after the target has been photographed).



Emission spectra of different light sources (normalised so that the maximum spectral radiance = 1). The sun is the most ecologically relevant light source and provides a good broad emission spectrum, but this also varies substantially with time of day, latitude, and atmospheric conditions. Arc lamps have relatively "spikey" emission spectra, but this example (Iwasaki eyeColor with its UV filter removed) does a relatively good job of recreating sunlight. Fluorescent tubes are very poor light sources for accurate colour rendering because of their very sharp spikes with some wavelengths almost entirely absent and others over-represented. These spikes can interact with reflectance spectra to make colours look quite different to our eyes or to a camera under sunlight versus fluorescent lighting.

The light source you use must cover the entire range of wavelengths you are photographing, *it is not acceptable to use two or more light sources with different emission spectra* to cover the whole range you are interested in unless these light sources can be blended together effectively (which is not trivial). For example, using one light source that emits human-visible light should not be combined with a second light to add the UV component. This is because the lights will interact with the 3D surface angles of the target, making some regions appear to have colours that they do not.

The Iwasaki eyeColour MT70D E27 6500K arc lamp available from CP-lighting (www.cp-lighting.co.uk) can be converted into a good UV-visible band light source by removing its UV/IR protective filter. The filter is just visible as an oily rainbow effect when the bulb is held up to the light. By using a hand-held drill with a steel wire circular brush this filter can be removed without damaging the glass to increase the UVA emissions of this bulb. Sensible precautions and protective equipment (goggles and gloves) should be worn when doing this. The bulb's emissions in the UVA range are increased by this process, so eye and skin protection must be worn when working near this light source for long periods, and it will fade nearby colours faster. Being an arc lamp this bulb needs to be operated by a suitable ballast (also available from CP-lighting). Photography stands that have an E27 socket can also readily be used with this bulb. Consult an electrician to wire this system together if you are not confident in doing it yourself.

Alternative visible-UV light sources are lamps designed for vivariums, e.g. arc lamps or mercury vapour lamps.

Linearisation Modelling

<http://www.empiricalimaging.com/knowledge-base/linearisation-modelling/>

We recommend using RAW photographs wherever possible because they preserve a linear relationship between the radiance of light hitting the sensor and the pixel values. However, in many circumstances it is desirable to use non-RAW images for objective image analysis. Examples include the use of cheap smart phones, or compressed video formats. These formats are non-linear, compressing brightness information in a manner which compliments human vision and display screens without much loss of visual information.

Most other software makes the assumption that non-linear images conform to the sRGB standard, however the dynamic range of cameras has increased substantially since that format was introduced, and in practice we have not found any cameras which conform to the standard to an acceptable degree, so wherever possible we recommend modelling the linearisation curves for each camera model manually.

Modelling a camera's linearisation requires a photograph of a range of [grey standards](#), ideally 6 or more from about 1-2% to 99-100%. This step only needs to be done *once* per camera.

The linear output images are typically normalised during normal image processing to create images where the pixel values scale with reflectance relative to the standard. This holds true for sRGB images, however it may not be the case for other colour spaces such as [DCL-P3](#), as used in Apple iPad/iPhone cameras. Cameras using colour spaces other than sRGB must therefore be [converted to cone-catch values](#).

Video Guide

<https://youtu.be/GCRWy61eln8>

Taking a suitable picture of the standards

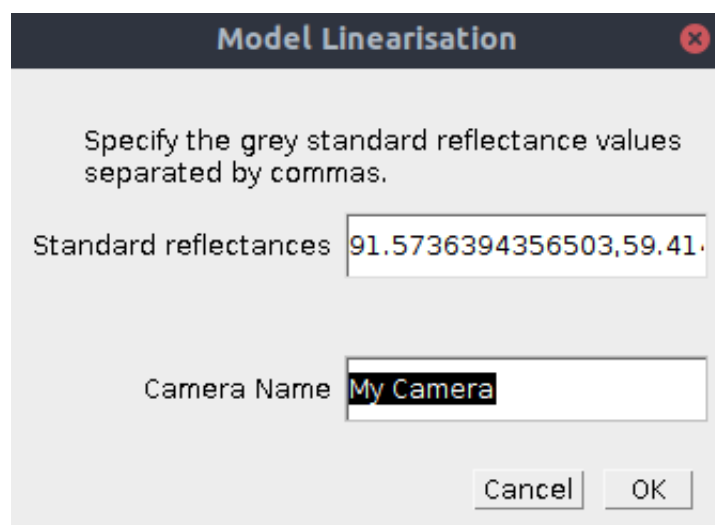
It is crucial that the lighting falling on each standard is as perfectly uniform as possible. Make sure the standard is either in full sun in an open space, or in full shade in an open space, with no objects around it. Using artificial light is possible, but not recommended unless you know how to ensure perfectly uniform lighting on each standard.

This method is sensitive to tiny changes in lighting conditions which will be imperceptibly small to your eyes.

Do not photograph too close to the standards (the camera itself will block some light). Ensure the standards are central in the shot.

Take photos with a range of exposures and select the one which has the brightest white standard where none of the pixels are saturated (i.e. all the pixel values are as near to 255 in each channel as possible without actually ever reaching 255).

Modelling



Model Linearisation

Specify the grey standard reflectance values separated by commas.

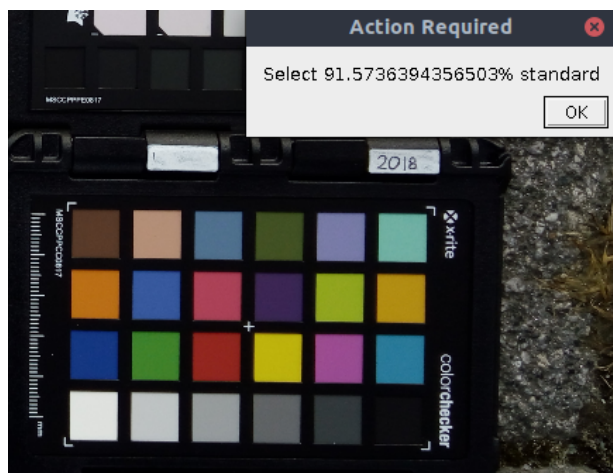
Standard reflectances

Camera Name

Go: *plugins > micaToolbox > Camera Calibration > Model Linearisation Function*

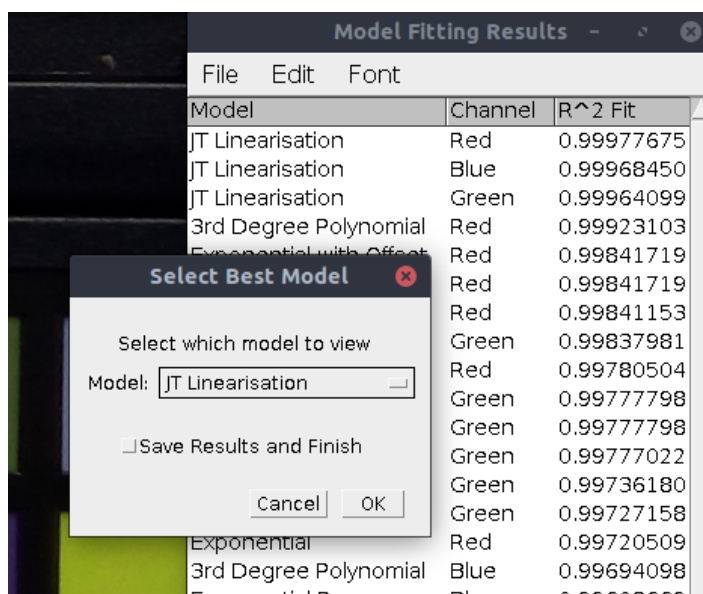
Standard Reflectances: Here you should specify the reflectance values of the grey standards in percentage (0-100 scale). Values should be separated by a comma (it doesn't matter which order you use). The default values loaded by the script are measurements of an X-Rite ColorChecker passport's 6 grey levels (note that there may well be some variation between charts and/or error in our measurements).

Camera Name: Choose a name to use for this camera (e.g. make/model).



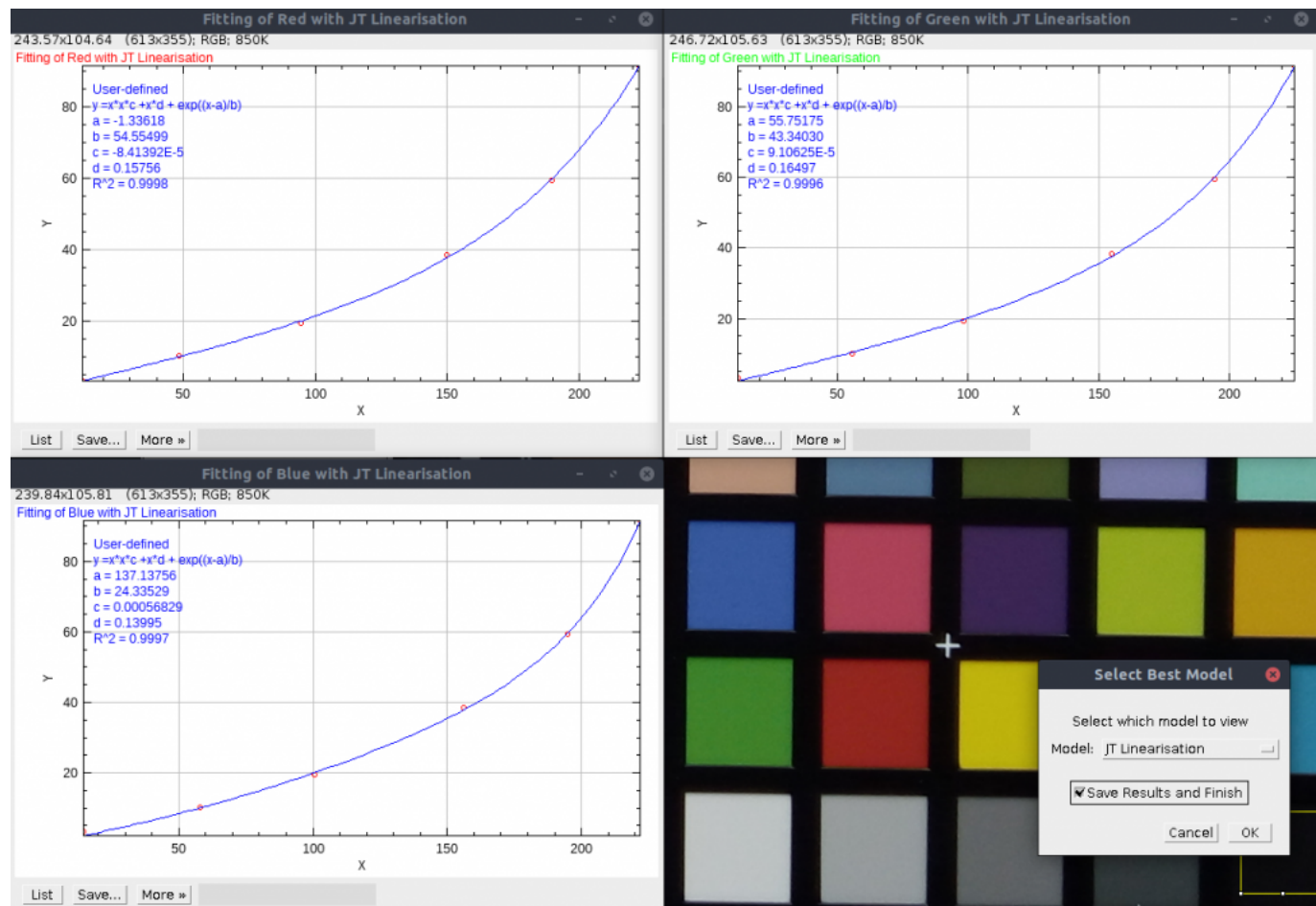
The script will then ask you to select an image containing the standards, and will ask you to draw a selection area over each standard in turn. Simply follow the on-screen prompts.

Selecting the best model



The script will model linearisation curves for your camera using a range of different equations. The equations and their fits are listed in the table (best at the top). Using the "Select Best Model" dialog box to create graphs of the different models.

When you have decided which equations work best tick "Save Results and Finish":



There should be a near perfect fit between the measured pixel values and the standard reflectance values (R^2 values > 0.999 ideally). Pay particular attention to the top and bottom values.

After saving the results you can generate a multispectral image as usual, just select "**Custom Non-Linear**" under **Image Type**, and select the camera linearisation model you have just created when prompted.

List of Animal Spatial Acuities

<http://www.empiricalimaging.com/knowledge-base/list-of-animal-spatial-acuities/>

This page lists the spatial acuity values measured for a large range of species (currently over 60 species). If you are aware of any other species to be added please get in touch via the forum. The "cycles-per-degree" values listed can be used in the [AcuityView](#) tool, or the [Gaussian Acuity Control](#) tool. If the acuity of your receiver species isn't listed, these lists should help you determine a likely "upper estimate" and "lower estimate" based on taxonomic relatedness, eye size and behavioural ecology. Repeat your analysis with both upper and lower estimates.

Collected Species

This table is based on table S1 published with AcuityView: Caves, E. M., & Johnsen, S. (2018). AcuityView: An r package for portraying the effects of visual acuity on scenes observed by an animal. *Methods in Ecology and Evolution*, 9(3), 793–797. doi:[10.1111/2041-210X.12911](https://doi.org/10.1111/2041-210X.12911)

If you use any of the values from this table please cite the original source in addition to Caves & Johnsen et al. (2018).

Common Name	Species Name	Acuity (cyc/deg)	Citation
Eagle	<i>Aquila audax</i>	140	(Reymond, 1985)
Human	<i>Homo sapiens</i>	73	(Land 1981)
		72	(Land and Nilsson 2012)
Octopus	<i>Octopus vulgaris</i>	46	(Young 1962; Land 1981)
Jumping Spider	<i>Portia fimbriata</i>	12	(Williams & McIntyre 1980)
Cat	<i>Felis catus</i>	10	(Wässle 1971)
Goldfish	<i>Carassius auratus</i>	7	(Penzlin & Stubbe 1977)
Triggerfish	<i>Rhinecanthus aculeatus</i>	2.75	(Champ et al. 2014)
Dragonfly	<i>Aeschna grandis</i>	2	(Mazokhin-Porshnyakov 1969; Land 1997)
Hooded Rat	<i>Rattus norvegicus</i>	1	(Wiesenfeld & Branchek 1975)
Worker Bee	<i>Apis mellifera</i>	0.5	(Wehner 1981; Land 1997)
Crab	<i>Leptograpsus variegatus</i>	0.3	(Stowe 1980)
Scallop	<i>Pecten maximus</i>	0.3	(Land 1981)
Wolf Spider	<i>Lycosa spp.</i>	0.2	(Widmann 1908; Homann 1931)
Fly	<i>Drosophila melanogaster</i>	0.09	(Wehner 1981; Land 1997)
Horseshoe	<i>Limulus</i>	0.08	(Barlow, Chamberlain &

Crab	<i>polyphemus</i>		Levinson 1980)
Nautilus	<i>Nautilus spp.</i>	0.06	(Land 1981)
Flatworm	<i>Euplanaria</i>	0.01	(Land 1981)
	<i>gonocephala</i>		

Birds

This table is from [Graham Martin's "Sensory Ecology of Birds" \(2017\) Oxford University Press](#), reproduced with the kind permission of the author. This textbook is an excellent introduction to avian sensory ecology and is highly recommended reading! If you use any value from this table please cite the original source, and Graham Martin's textbook.

Order	Family	Species	Acuity (cyc/deg)	Acuity (minutes of arc)	Method	Sources
Struthioniformes	Struthionidae (Ostriches)	Common Ostrich <i>Struthio camelus</i>	19.8	1.5	ganglion	(Boire et al. 2001)
Anseriformes	Anatidae (Ducks, Geese and Swans)	Mallard <i>Anas platyrhynchos</i>	11.9	2.52	ganglion	(Lisney et al. 2013a)
		Canada Goose <i>Branta canadensis</i>	9.6	3.1	ganglion	(Fernandez-Juricic et al. 2011)
Galliformes	Phasianidae (Pheasants and allies)	Grey Partridge <i>Perdix perdix</i>	10.2	2.9	ganglion	(Lisney et al. 2012)
		Japanese Quail <i>Coturnix coturnix</i>	9.7	3.1	ganglion	“
		Red Jungle Fowl <i>Gallus gallus</i>	8.3	3.6	ganglion	(Ehrlich 1981)
		Common Pheasant <i>Phasianus colchicus</i>	12.9	2.3	ganglion	(Lisney et al. 2012)
		Indian Peafowl <i>Pavo cristatus</i>	20.6	1.5	ganglion	(Hart 2002)
Procellariiformes	Hydrobatidae (Northern Storm Petrels)	Leach's Storm Petrel <i>Oceanodroma leucorhoa</i>	7.1	4.2	ganglion	(Mitkus 2015)
	Procellariidae (Petrels, Shearwaters)	Northern Fulmar <i>Fulmarus glacialis</i>	44.7	0.7	ganglion	“
Suliformes	Phalacrocoracidae (Cormorants,	Great Cormorant <i>Phalacrocorax</i>	3.3 (underwater)	9.1	behaviour	(White et al. 2007)

Accipitriformes	Shags) Cathartidae (New World Vultures)	<i>carbo</i>					
		Turkey	15.4	1.9	ganglion	(Lisney et al. 2013b)	
		Vulture <i>Cathartes</i>					
		<i>atratus</i>					
			Black	15.8	1.9	ganglion	“
			Vulture <i>Coragyps</i>				
			<i>aura</i>				
	Accipitridae (Kites, Hawks and Eagles)	Indian	135	0.2	behaviour	(Fischer 1969)	
		Vulture <i>Gyps</i>					
		<i>indicus</i>					
		Griffon	104	0.3	behaviour	“	
		Vulture <i>Gyps</i>					
		<i>fulvus</i>					
		Egyptian	135	0.2	behaviour	“	
vulture <i>Neophron</i>							
		<i>percnopterus</i>					
		Wedge-tailed	142	0.2	behaviour	(Reymond 1985)	
		Eagle <i>Aquila</i>					
		<i>audax</i>					
		Black	37.3	0.8	behaviour	(Potier et al. 2016)	
		Kite <i>Milvus</i>					
		<i>migrans</i>					
		Harris’s	29.3	1.0	behaviour	“	
		Hawk <i>Parabuteo</i>					
		<i>unicinctus</i>					
Columbiformes	Columbidae (Pigeons, Doves)	American	7.6	3.9	ganglion	(Dolan and Fernandez-Juricic 2010)	
		Mourning Dove					
		<i>Zenaida</i>					
		<i>macroura</i>					
		Rock	18	1.7	behaviour	(Hodos et al. 1976)	
		Dove <i>Columba</i>					
		<i>livia</i>					
Strigiformes	Tytonidae (Barn Owls)	Western	4.0	7.5	behaviour	(Harmening et al. 2009)	
		Barn Owl <i>Tyto</i>					
	Strigidae (Owls)	<i>alba</i>					
		Great	7.5	4.0	behaviour	(Fite 1973)	
		Horned Owl					
		<i>Bubo virginianus</i>					
		Tawny	11.1	2.7	behaviour	(Martin and Gordon 1974)	
		Owl <i>Strix</i>					
		<i>aluco</i>					
Coraciiformes	Alcedinidae (Kingfishers)	Laughing	41	0.7	ganglion	(Moroney and Pettigrew 1987)	
		Kookaburra					
		<i>Dacelo</i>					
		<i>novaeguineae</i>					
		Sacred	26	1.2	ganglion	“	
		Kingfisher					
		<i>Todiramphus</i>					
		<i>sanctus</i>					
Falconiformes	Falconidae (Caracaras, Falcons)	American	40	0.75	behaviour	(Hirsch 1982)	
		Kestrel <i>Falco</i>					
		<i>sparverius</i>					
		Brown	73	0.4	behaviour	(Reymond 1987)	
		Falcon <i>Falco</i>					

Psittaciformes	Psittaculidae (Old World Parrots)	<i>berigora</i> Bourke's Parrot	9.4	3.2	behaviour	(Lind et al. 2012)
		<i>Neopsephotus bourkii</i> Budgerigar	11.7	2.6	behaviour	“
		<i>Melopsittacus undulatus</i> Brown Honeyeater	20.2	1.5	photoreceptor spacing	(Coimbra et al. 2015)
Passeriformes	Meliphagidae (Honeyeaters)	<i>Lichmera indistincta</i> Red Wattlebird	40.8	0.7	photoreceptor spacing	“
		<i>Anthochaera carunculata</i> Yellow-rumped Thornbill	25.6	1.2	photoreceptor spacing	“
		<i>Acanthiza chrysorrhoa</i> Blue Jay	19	1.6	ganglion	(Fite and Rosenfeld-Wessels 1975)
	Corvidae (Crows, Jays)	<i>Cyanocitta cristata</i> Eurasian Magpie	33.3	0.9	behaviour	(Dabrowska 1975)
		<i>Pica pica</i> Rook	29.5	1.0	behaviour	“
		<i>Corvus frugilegus</i> Tufted Titmouse	6.6	4.5	ganglion	(Moore et al. 2013)
	Paridae (Tits, Chickadees)	<i>Baeolophus bicolor</i> Carolina Chickadee	5	0.6	ganglion	“
		<i>Poecile carolinensis</i> Silvereye	18.5	1.6	photoreceptor spacing	(Coimbra et al. 2015)
		<i>Zosterops lateralis</i> Common Blackbird	22.5	1.3	behaviour	(Donner 1951)
	Zosteropidae (White-eyes)	<i>Turdus merula</i> European Robin	6	5.0	behaviour	“
		<i>Erithacus rubecula</i> House Sparrow	4.8	6.3	ganglion	(Dolan and Fernandez-Juricic 2010)
		<i>Passer domesticus</i> Common Chaffinch	22.5	1.3	behaviour	(Donner 1951)
	Fringillidae (Finches)	<i>Fringilla coelebs</i> House Finch	4.7	6.4	ganglion	(Dolan and Fernandez-Juricic 2010)
		<i>Haemorhus mexicanus</i>				

	Emberizidae (Buntings, New World Sparrows)	Yellowhammer <i>Emberiza citrinella</i>	9.7	3.1	behaviour	(Donner 1951)
		Common Reed Bunting <i>Emberiza schoeniclus</i>	7.8	3.8	behaviour	“
Human			72	0.4	behaviour	(Land and Nilsson 2012)

References

Barlow, R.B., Chamberlain, S.C. & Levinson, J.Z. (1980). *Limulus* brain modulates the structure and function of the lateral eyes. *Science*, **210**, 1037–1039.

Homann, H. (1931). Beiträge zur Physiologie der Spinnenaugen. I. Untersuchungsmethoden. II. Das Sehvermögen der Salticiden. *Z. Vergl. Physiol.*, **14**, 40–67.

Land, M.F. (1981). Optics and vision in invertebrates. *Handbook of Sensory Physiology* (ed H. Autrum), pp. 471–592. Springer, Berlin.

Land, M.F. (1997). Visual acuity in insects. *Annual review of entomology*, **42**, 147–177.

Mazokhin-Porshnyakov, G.A. (1969). *Insect Vision*. Plenum Press, New York.

Penzlin, V.H. & Stubbe, M. (1977). Studies on the visual acuity in the Goldfish. *Zool. Jb. Physiol. Bd.*, **81**, 310–326.

Reymond, L. (1985). Spatial visual acuity of the eagle *Aquila audax*: A behavioural, optical and anatomical investigation. *Vision Research*, **25**, 1477–1491.

Stowe, S. (1980). Spectral sensitivity and retinal pigment movement in the crab *Leptograpsus variegatus* (Fabricius). *Journal of Experimental Biology*, **87**, 73–99.

Wässle, H. (1971). Optical quality of the cat eye. *Vision Research*, **11**, 995–1006.

Wehner, R. (1981). Spatial vision in arthropods. *Handbook of Sensory Physiology, Vol. VII/6C* (ed H. Autrum), pp. 287–616. Springer, Berlin.

Widmann, E. (1908). Über den feineren Bau der Augen einiger Spinnen. *Z. Wiss. Zool.*, **90**, 258–312.

Wiesenfeld, Z. & Branchek, T. (1975). Refractive state and visual acuity in the hooded rat. *Vision Research*, **16**, 823–827.

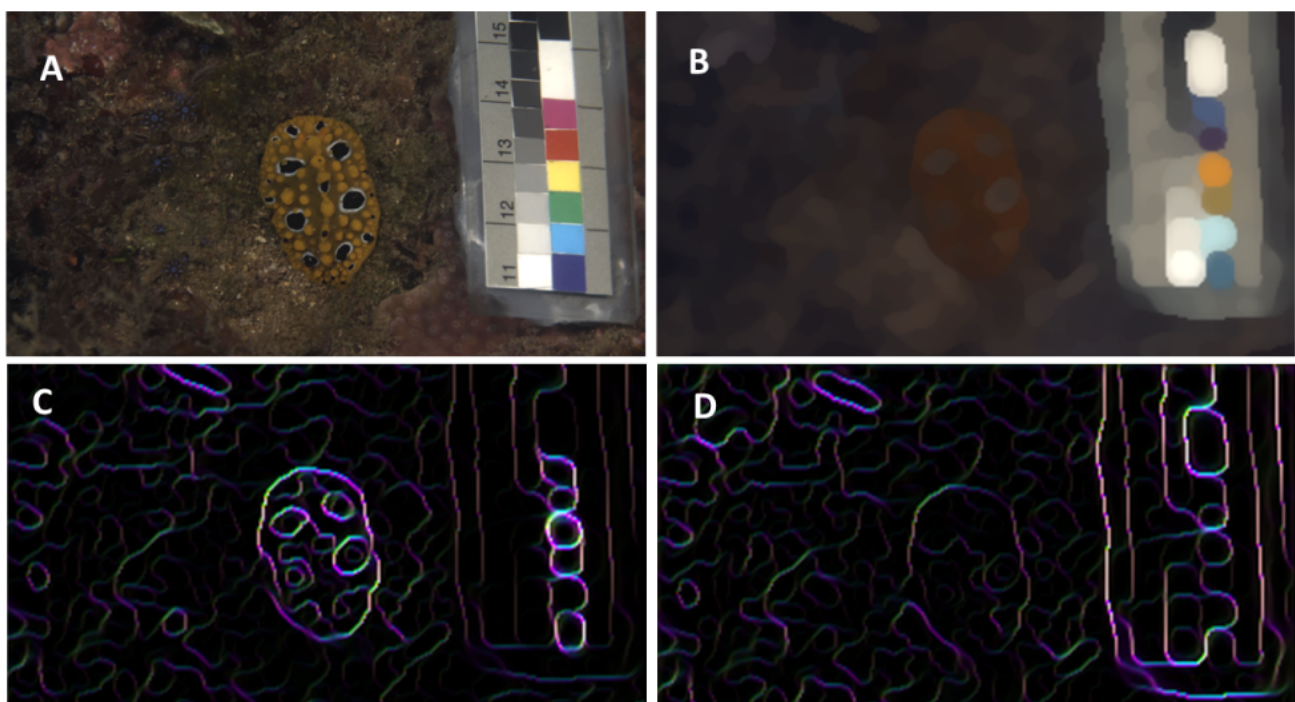
Williams, D.S. & McIntyre, P. (1980). The principal eyes of a jumping spider have a telephoto component. *Nature*, **288**, 578–580.

Local Edge Intensity Analysis (LEIA)

<http://www.empiricalimaging.com/knowledge-base/local-edge-intensity-analysis-leia/>

The Local Edge Intensity Analysis (LEIA) measures edge intensities across a one-pixel-neighbourhood in four directions: horizontal, vertical and both diagonals. The contrast is measured as the Euclidean distance in the log-transformed Receptor Noise Limited (RNL) colour space and provided as both chromatic and achromatic contrast between neighbouring pixels. This provides a local measure of edge contrast, which in combination with [visual acuity control](#) and the [RNL ranked filter](#) provides an approximation of centre surround field scaled edge detection. Conceptually, this is very similar to the [Boundary Strength Analysis](#) (BSA), however the major difference is that this runs on unclustered/unsegmented images (e.g. the unique case in the BSA where each pixel is its own cluster). Additionally, the output has been modified to account for the fact that in any natural image the distribution of Delta-S values will follow something like a log distribution, meaning the data should be transformed and/or measured in ways which account for this non-normal distribution.

The output images show false colour, where the hue specifies the angle of the boundary, and the intensity shows the Delta-S of the boundary.



An example of LEIA. A nudibranch mollusc photographed (A) and then modelled as seen by a triggerfish from 30cm distance at 5m depth (B). Local chromatic (C) and achromatic (D) edge intensity shown with different colours indicating different angle contrasts.

Input Requirements

We recommend this method should be used following acuity control ([FFT-based AcuityView](#) or [Gaussian acuity control](#)) and [RNL ranked filtering](#). This pre-processing is essential for ensuring the detected edges will be visible to the receiver at a given distance, while also recovering appropriate colour boundaries following acuity control.

The method requires a 32-bits/channel cone-catch image (di-, tri- or tetrachromatic), and the user is asked to provide the receiver-specific Weber fractions for each receptor channel (or select the species from a drop-down list if using the QCPA framework).

Zero pixel values in any of the cone channels will be ignored for the analysis. This is a useful method for measuring only sub-sections of in image. If there is any ROI currently selected only the region within the ROI will be measured.

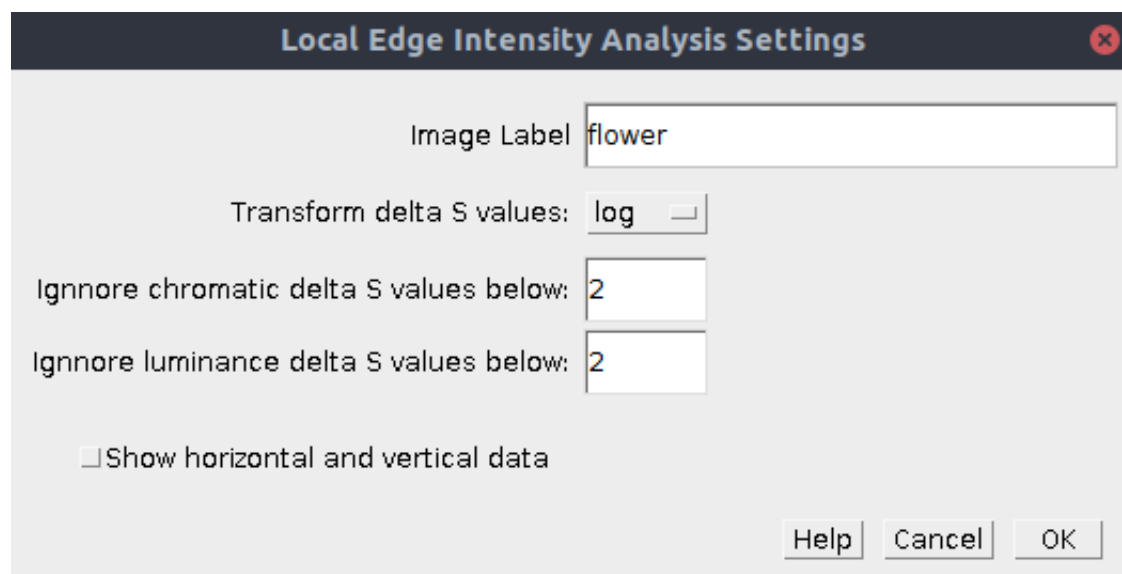
Running the Local Edge Intensity Analysis

We recommend running this analysis as part of the [QCPA framework](#) with [acuity correction](#) and [RNL Ranked Filtering](#) applied:

Plugins > micaToolbox > QCPA > Run QCPA Framework

However, it can also be run in isolation on any cone-catch image (you will need to specify the Weber fractions yourself):

Plugins > micaToolbox > QCPA > Tools > Local Edge Intensity Analysis



The image shows a dialog box titled "Local Edge Intensity Analysis Settings" with a close button (X) in the top right corner. The dialog contains the following fields and controls:

- Image Label:** A text input field containing the word "flower".
- Transform delta S values:** A dropdown menu currently set to "log".
- Ignore chromatic delta S values below:** A numeric input field containing the value "2".
- Ignore luminance delta S values below:** A numeric input field containing the value "2".
- Show horizontal and vertical data:** An unchecked checkbox.
- Buttons:** "Help", "Cancel", and "OK" buttons are located at the bottom right.

Image Label

Transform delta S values

Select a label. This will be used in the output table.

none: Do not transform the Delta-S values, just use the raw output of the RNL model (note that the RNL model still uses the log-cone-catch values)

sqrt: Square root transform. Sometimes the log option (below) is considered too extreme, in which case the square-root is a less aggressive choice. You can create histograms of the output images to judge the distribution yourself.

log: Natural log transform the delta-S values. This tends to create a normal distribution for images of natural scenes, and is therefore probably the most sensible option. These normal distributions allow the subsequent metrics (such as coefficient of variation, skewness and kurtosis to perform better).

Ignore chromatic delta S values below

Sub-threshold Delta-S values are arguably imperceptible, and can therefore be eliminated. All values below this threshold are set to zero. However, this may also interfere with the distribution metrics, so is up to the user to decide whether to use this option (if in doubt, set to zero to turn this feature off).

Ignore luminance delta S values below

Sub-threshold Delta-S values are arguably imperceptible, and can therefore be eliminated. All values below this threshold are set to zero. However, this may also interfere with the distribution metrics, so is up to the user to decide whether to use this option (if in doubt, set to zero to turn this feature off). If ticked, output data will also be calculated for the horizontal and vertical directions independently. This could be useful when comparing patterns which are always angled in a certain direction (the images must also be standardised to show the patterns in a consistent orientation).

Show horizontal and vertical dat

Output

Two images are produced, one showing the chromatic edge Delta-S values (with the suffix "_Col_LEIA") and another showing luminance (with the suffix "_Lum_LEIA"). Each image is a 32-bit stack of four slices, where each slice shows the values measured in different angles (horizontal, diagonal, vertical and other diagonal, labelled with symbols showing the angle: -, /, |, \). You can measure these values independently if desired. The stacks have been given false colour though lookup tables where the intensity shows the Delta-S values, and the colour varies between slice (so therefore shows angle).

A "Local Edge Intensity Analysis" results table is produced which shows measurements of Delta-S values in the image.

Col mean	The mean chromatic RNL Delta-S values over the entire image, where at each pixel location only the maximum Delta-S across all four angles is measured.
Col sd	Standard deviation of chromatic Delta-S values (as above)
Col CoV	Chromatic Delta-S Coefficient of variation.
Col skew	Chromatic Delta-S skewness. Skewness is a measure of the asymmetry of the distribution of Delta-S values about the mean. Generally a higher number would imply the image has a larger proportion of high AND low Delta-S values (possibly a prediction of a salient signal).
Col kurtosis	Chromatic Delta-S kurtosis. Kurtosis is a measure of the degree of extreme outliers creating tails at either end of the distribution. A higher kurtosis implies very low variation in Delta-S values about the mean.
Lum mean	The mean luminance (achromatic) RNL Delta-S values over the entire image, where at each pixel location only the maximum Delta-S across all four angles is measured.
Lum sd	Standard deviation of luminance Delta-S values (as above)
Lum CoV	Luminance Delta-S Coefficient of variation.
Lum skew	Luminance Delta-S skewness. Skewness is a measure of the asymmetry of the distribution of Delta-S values about the mean. Generally a higher number would imply the image has a larger proportion of high AND low Delta-S values (possibly a prediction of a salient signal).
Lum kurtosis	Luminance Delta-S kurtosis. Kurtosis is a measure of the degree of extreme outliers creating tails at either end of the distribution. A higher kurtosis implies very low variation in Delta-S values about the mean.
Transform	Specifies which transform was applied to the Delta-S values before the above metrics are calculated (none, square-root, or natural log).
Col thresh	Specifies the selected chromatic Delta-S threshold, below which all values are set to zero for calculating the above metrics.
Lum thresh	Specifies the selected luminance Delta-S threshold, below which all values are set to zero for calculating the above metrics.

Make Your Own: Camera Calibration

<http://www.empiricalimaging.com/knowledge-base/make-your-own-camera-calibration/>

There are multiple ways one can calibrate a camera to be used for MICA. Here, we present the "standard" and most thorough methodology:

Calibrating a Camera using a Monochromator and a Spectroradiometer

In essence, the process entails:

1. Projecting a light beam of narrow bandwidth onto a diffuse white standard (e.g. spectralon 99%) using a monochromator.
2. Taking a picture of that light spot with the camera that is to be calibrated (1picture/5nm).
3. Take a measure of the light intensity ($\mu\text{W}/\text{cm}^2/\text{s}$) using a spectroradiometer (note this must be fully calibrated to account for the spectroradiometer's sensitivity and absorbance by optical media, e.g. using a calibration light source or using a lab which can perform this).
4. Measure the linear R, G and B pixel value and correlate it to the spec measurement.
5. Repeat for every 5nm to get entire 400-700nm spectral sensitivity of the camera (or 300-700nm for VIS-UV full spectrum cameras).



Camera settings:
f2.8
ISO 200
Focus manual
WB Auto

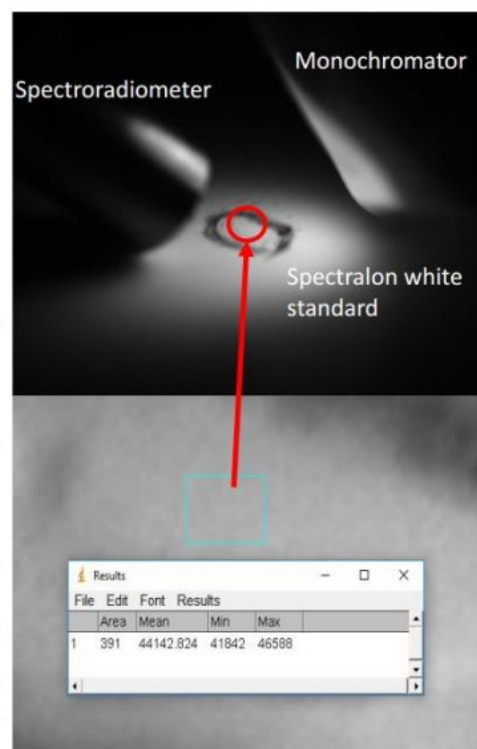
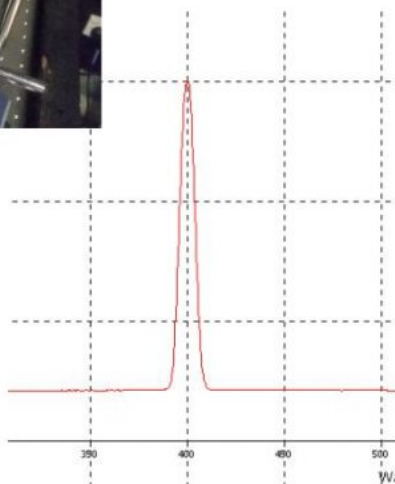


Figure 1: Setup on optical bench (needed for precision). Top left: USB2000 400nm fibre optic cable on the left, monochromator fibre on the right. Spectralon in the middle, camera fixed at lower center. Top right: Close up of spectralon white. Central area of illumination indicated with pencil circle as it can be hard to see sometimes. Bottom right: Average pixel response for one channel. Center: Spec reading of 400nm monochromator light. Bottom left: settings used for the Olympus E-PL5.

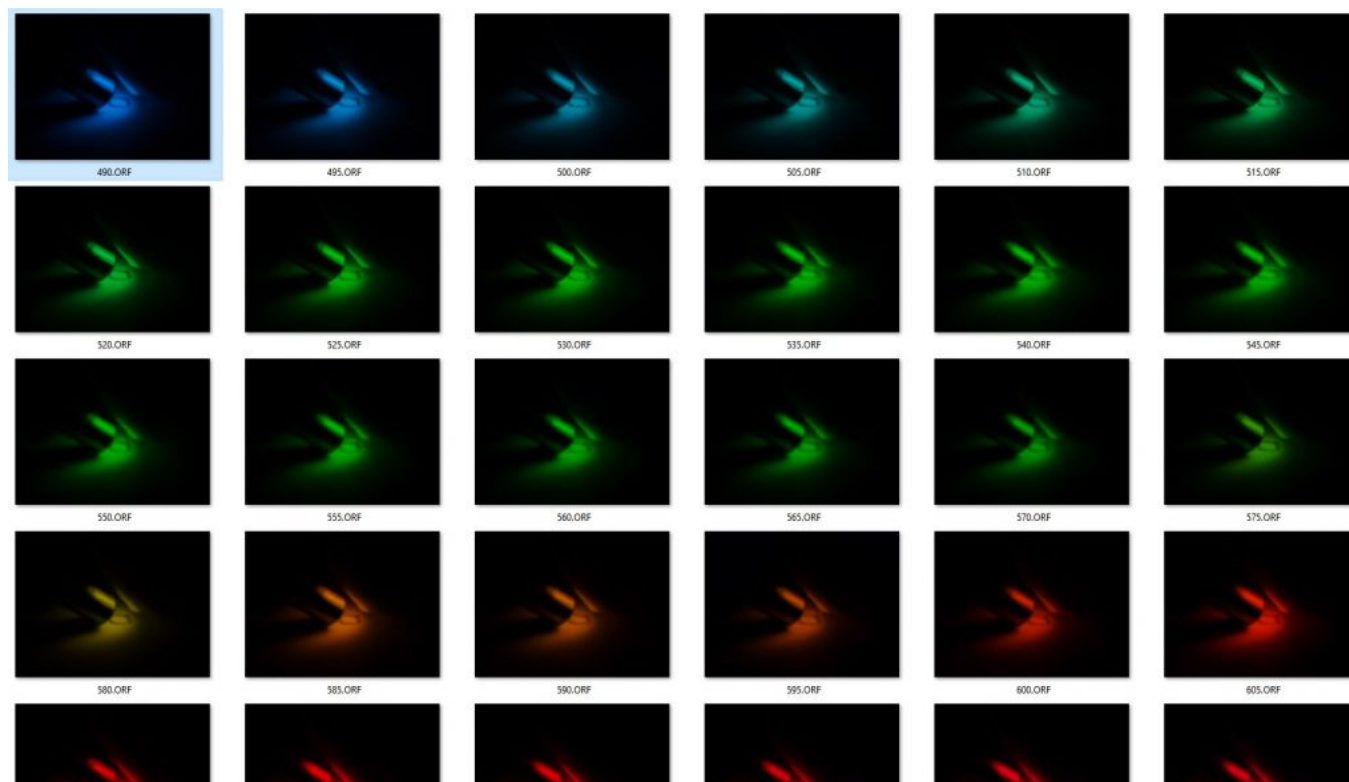


Figure 2: Examples of RAW images taken for camera calibration

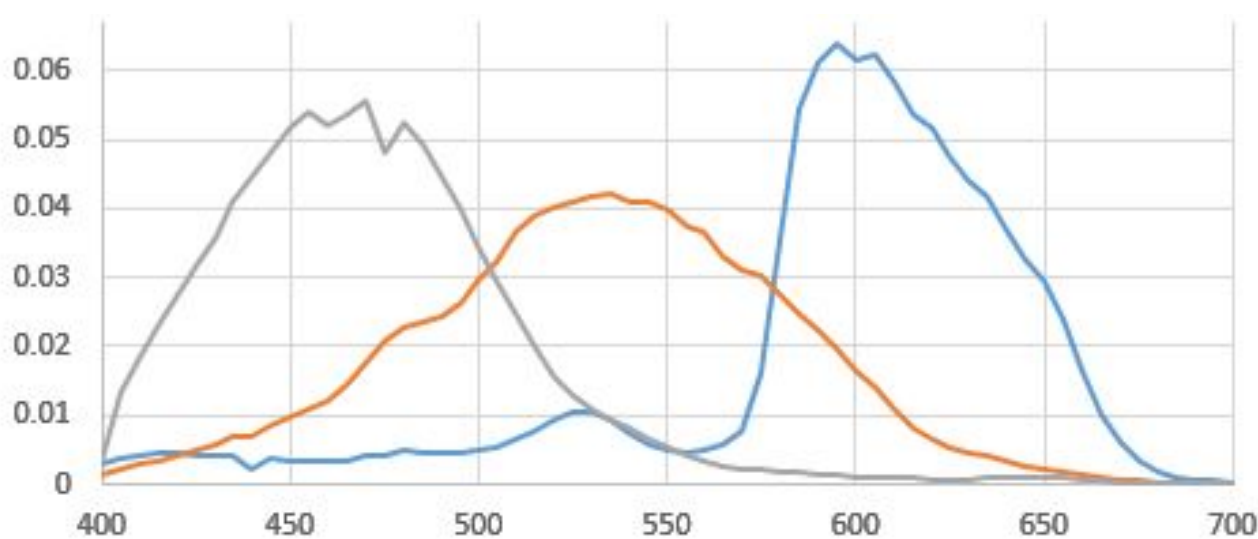


Figure 3: Example of normalised (sum under curve = 1) spectral sensitivity of the Olympus E-PL5

Detailed Instructions:**I. Setup of Hardware (See Fig 1):**

You will have to work on an optic bench to be able to hold everything in place and get exact readings. Get an absolutely clean and unsratched spectralon 99% reflectance standard (Sand it down with a wet 200 or 400 (better) grid sandpaper and let it dry). Grab a piece of black electric tape and punch a hole with a hole punch. Place the punctured tape on the spectralon standard so that the hole is in the center.

Get 3 sockets to fix the recording sensors around the standard (Fig1). You will need 2 sockets for holding a plastic rod in place to which either the output fibre of the monochromator or the recording fibre (400nm, non-bifurcated cable) is taped to so it will absolutely not move. The third socket will be used to fix the camera in place. Slightest shifts will induce large variations in your recordings and make your measurements useless. Position the monochromator, camera and the optic fibre as depicted in Fig 1. Essentially with the monochromator and spec cables almost touching the white standard.

Disclaimer: If you don't have a calibration file for the specific USB2000 unit and the specific cable you are working with you will need to create one prior to the camera calibration. It's a pain to figure it out but is quickly done once you know how to.

II. Setup of Camera

The camera calibration will have to be done with a lens that can't zoom. If your camera is able to zoom (either by built-in or added lenses) you will need to ensure that all your images will be taken with identical zoom settings. The reason for this being that physical zoom distorts the optic properties of the lens and subsequently messing up your calibration. Note that this means that any images will have to be taken with the exact same zoom settings and lenses once you have calibrated the camera.

Your white balance can be set to automatic as the linearisation and normalisation of the images in the MICA toolbox will correct for that. Your aperture (f-value) will need to be fixed. I recommend using exposure bracketing to avoid taking overexposed images. In any case, choose the settings on your camera that allow you to see the exposure histogram. This will greatly help to avoid overexposure. You will need to record the shutter speed of each image. Your ISO will need to be fixed e.g. at

200, which is reasonable (also when you are taking images in the field). I recommend taking images using exposure bracketing to ensure optimal exposure. This is crucial as the brightness of the monochromator varies substantially across the light spectrum. You will likely have to play around with this for a while.

III. Analysing the measurements

The idea is to have a series of images for which you know exactly how long (exposure time) the sensors have been exposed to a certain number of photons (sum of absolute irradiance measured by the spectrophotometer) and what pixel value (RGB value) that stimulation corresponds to.

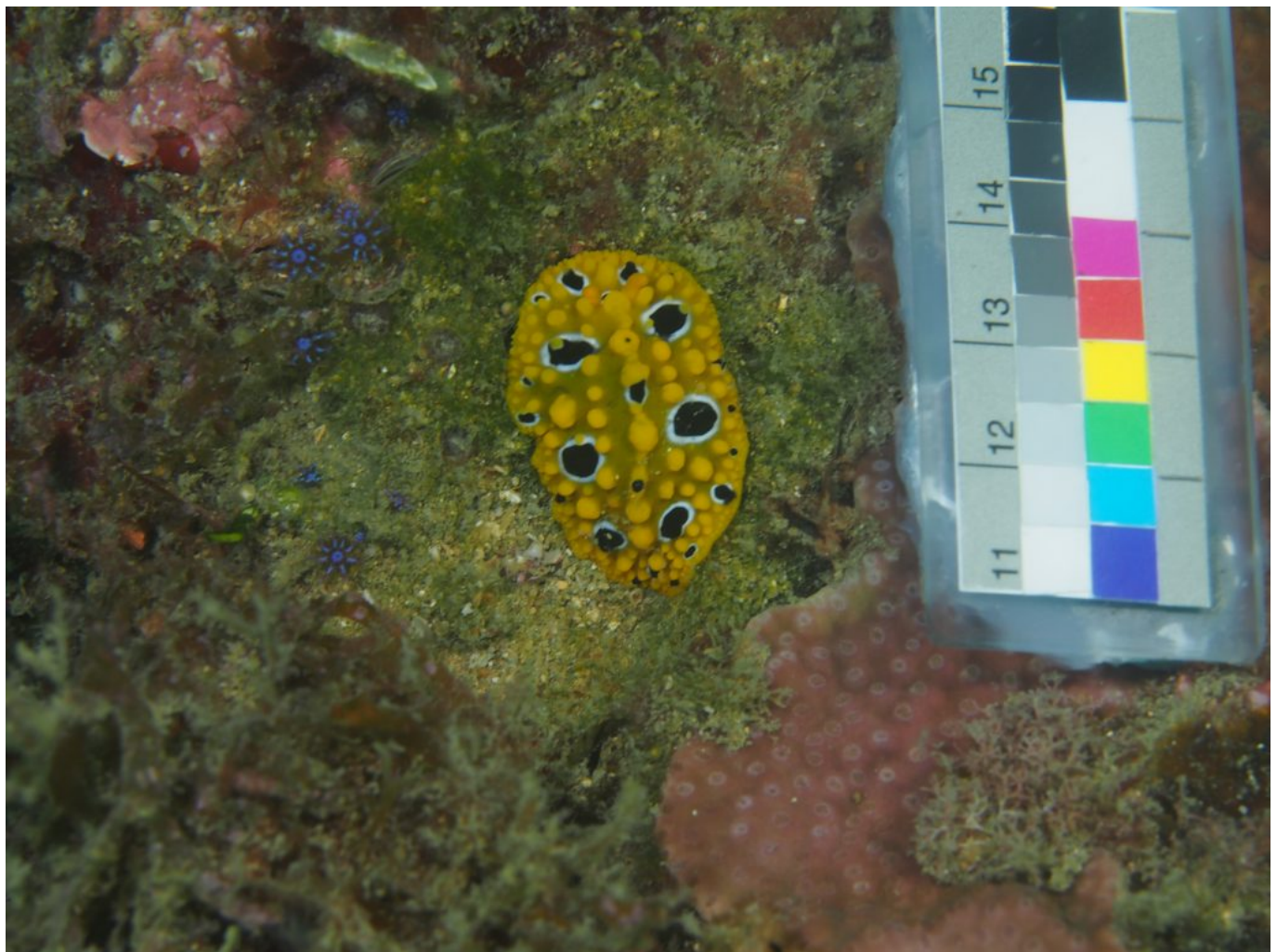
So, on one hand you will need a well exposed (bright enough but not overexposed) series of images of the light spot of the monochromator on the spectralon white every 5nm from 400nm-700nm (Or other depending on your needs, e.g if your camera can detect UV). On the other hand you need the corresponding absolute irradiance measurements.

Using the MICA toolbox you can open your .RAW images using the DCRaw plugin (*Plugins > micaToolbox > Tools > DCRAW Import*). This will open your image as a linearised image. Using the circular polygon selection tool you can then select the illuminated spot. Don't worry, all we need is a couple of dozen pixels at the minimum, so you don't need to precisely select the ENTIRE spot. Just the most homogeneously illuminated bit in the center. Select the ROI by pressing 't' (Or whatever letter you fancy, other than 's' which is for size standards specifically). Then press "m" to measure the current selection. Record the average of the reading for each channel (R, G and B). Divide this value by your shutter speed (e.g. 0.05) and this will give you your normalised pixel response. Note that you want the exact integration time of the exposure. e.g. 1/1000th of a second is normally actually 1/1024 and the times often follow this base-2 system. Divide this value by the sum of the absolute irradiance from 400nm-700nm. This is how sensitive your R, G and B sensors in your camera are to light at a given wavelength. This value does not have any units (X amount of Light per time divided by X amount of light per time). Interpolate using R or Matlab to transform this 5nm bracketing to 1nm measures. Normalise these values to Max=1 or Sum=1 according to the requirements of the calculations you use these for. Assuming you will be using it for MICA you will find the file specifications in the MICA manual.

Make Your Own: Colour & Grey Standard

<http://www.empiricalimaging.com/knowledge-base/make-your-own-colour-grey-standard/>

Colour and grey standards are diverse and should be fitted to their purpose. As they fundamentally define the quality of the image data they deserve careful attention prior to image acquisition. There is a range of commercially available colour standards (e.g. the [Macbeth Colour Checker](#)) and many of these perform perfectly well under specific conditions (often designed for indoors or nice weather). However, they often might not possess the properties needed by a visual ecologist seeking to take calibrated imagery in the field. This includes: The right size, weather & water proof, kinds of colours & greys, size scale, etc...



An example of a custom made resin cast colour, grey and size standard of 5x2cm built by Cedric van den Berg. Detailed information on how to make one can be found below.



An example of a grey standard (Using Spectralon) built by Jolyon Troschianko of 4x2cm dimension. More information below.

A Resin Cast Solution:

The resin cast colour standard depicted above is essentially a miniaturised 'Kodak Colour Separation Guide' as shown below. It has very pretty reflection spectra (e.g. perfectly flat greys, nice step functions for the red etc...). However, it does not reflect UV which would be a reason not to use this colour standard if you intend to do UV photography. Also, it is slightly glossy which is a property best avoided, although acceptable if known and used accordingly. And, it's cheap. A LOT cheaper than spectralon standards.



A selection of 12 grey and 10 colours was made of which each a 5x5mm square was cut and glued onto waterproof paper. The grey of the standard provides a nice base for the size standard. Having multiple greys is a nice way of controlling for optimal exposure in an image while having multiple grey standards aids to linearise the images in MICA if lighting conditions are not perfect (e.g. hazy). Having colour tiles is not necessary but it allows to quickly judge how the visual modelling transforms the images.

The miniaturised colour standard was cast into optically clear resin hardened under vacuum to avoid air bubbles. The resin was then sanded back until only a hint of resin remained in front of the colour tiles. The resin was finished with a matte polish that reduces specular reflection. Lead weights were attached to the back of the standard using silicon and shape fitted fishing weights. This helps to keep the standard in place and make it nicely negatively buoyant. A dive clip and string were attached to a hole drilled into the standard to allow for easy attachment to the camera rig when not used.

Positive:

- Great selection of colours and greys for little money
- Can easily be cleaned and sanded down
- 100% Waterproof
- Great to stay in place in rough conditions and on uneven surfaces

Negative:

- Resin suffers from degradation by UV and saltwater.
- Fairly sensitive to specular reflection. Not lambertian.

- No UV reflection

The Spectralon Solution:

Spectralon is essentially a foamed version of PTFE (polytetrafluorethylene or teflon). While this makes the material extremely susceptible to scratching and dents, it also makes it near-perfectly lambertian (e.g. it has the same reflectance no matter the viewing angle). And spectralon is expensive. However, it can be purchased in various grey levels and shapes and cut into shape as per the second standard depicted above. Spectralon does not have lambertian properties underwater (caused by the thin layer of air between the material and the water). However, this effect is expected to be reduced at greater depths where water pressure forces that air layer deeper into the material.



Positive:

- Best reflective properties. Lambertian in air, near lambertian underwater.
- Can easily be cleaned
- UV reflective down to 250nm

Negative:

- Spectralon is very expensive

Alternative approaches:

Colour and grey standards can be made of a variety of other materials. Essentially, you could take a piece of wood, spray paint it with matte greys and take a picture with a calibrated camera to determine the % response of your R, G and B against a known grey standard using MICA. It would work. However, few materials other than the ones shown above have the durability and the visual properties of the solutions described above. A list of alternative materials that can be (And have been) used as colour and grey standards:

- Barium Sulfate
- Aluminium
- Aluminium + Magnesium Fluoride
- Chalk
- Paint
- Thread seal tape

Desirable properties are:

- Broad reflection spectrum
 - Little or no specular reflection
 - For greys: An even (flat) reflectance spectrum
-

Make Your Own: Spectral Sensitivity & Illuminant Files

<http://www.empiricalimaging.com/knowledge-base/make-your-own-spectral-sensitivity-illuminant-files/>

The MICA toolbox allows the user to import any desired spectral sensitivities or illuminant. It is important that the files have 1nm bins and are formatted correctly as .csv files. Please use existing files in the toolbox to see how the spectra and sensitivities need to be normalised as well as in what order the channels need to be arranged. If you would like to have a published spectral sensitivity or an illuminant included in the MICA toolbox by default, please contact us in the forum or via email.

Note that spectral illuminant data must be in photon counts (flux). The toolbox typically uses normalised illuminants where the peak=1 (allowing for the calculation of relative cone-catch quanta).

How to include old (non-digitized) data

Many publications on spectral sensitivities and light spectra predate the digital era and data may only be available in print. To extract the spectra from these publications we recommend using freely available tools such as the [WebPlotDigitizer](#) and others. However, this does not obviate the need for correctly citing the original authors of that data.

micaToolbox Installation Instructions

<http://www.empiricalimaging.com/knowledge-base/micatoolbox-installation-instructions/>

1 Download & Install ImageJ

The micaToolbox is an add-on for the standard NIH ImageJ (not FIJI). ImageJ can be downloaded for almost any operating system [here](#). In general, use a version bundled with Java. Note the specific instructions for MacOS (path randomisation) which are flagged on the download page.

To install ImageJ simply unzip the contents of the zip file to a location on your computer. In windows, don't unzip to *Program Files*. Use some other location on your hard drive (e.g. somewhere in *Documents*).

Double-click the "ImageJ" or "ImageJ.exe" file to run ImageJ. You can make a shortcut to this file if you want easy access.

2 Download & Install the micaToolbox

Download the latest micaToolbox version from the [Downloads](#) section of this website.

Unzip the contents of the zip file to your *ImageJ/Plugins* directory. If ImageJ is currently running, either restart it, or go *help>refresh menus* to load the toolbox.

Naive Bayes Clustering

<http://www.empiricalimaging.com/knowledge-base/naive-bayes-clustering/>

The [naive Bayes classifier](#) is a simple but effective classification algorithm which can be used for image segmentation/clustering. We have created a custom naive Bayes classifier and integrated it into the [QCPA framework](#). This method is suitable only when you have prior knowledge of the maximum number of image segments expected, and their average colours, unlike the [RNL Clustering](#) technique. The means and standard deviations of each cluster across n-dimensions of channels in the image must be provided.

Video Guide

https://youtu.be/_f_iKhEGEps

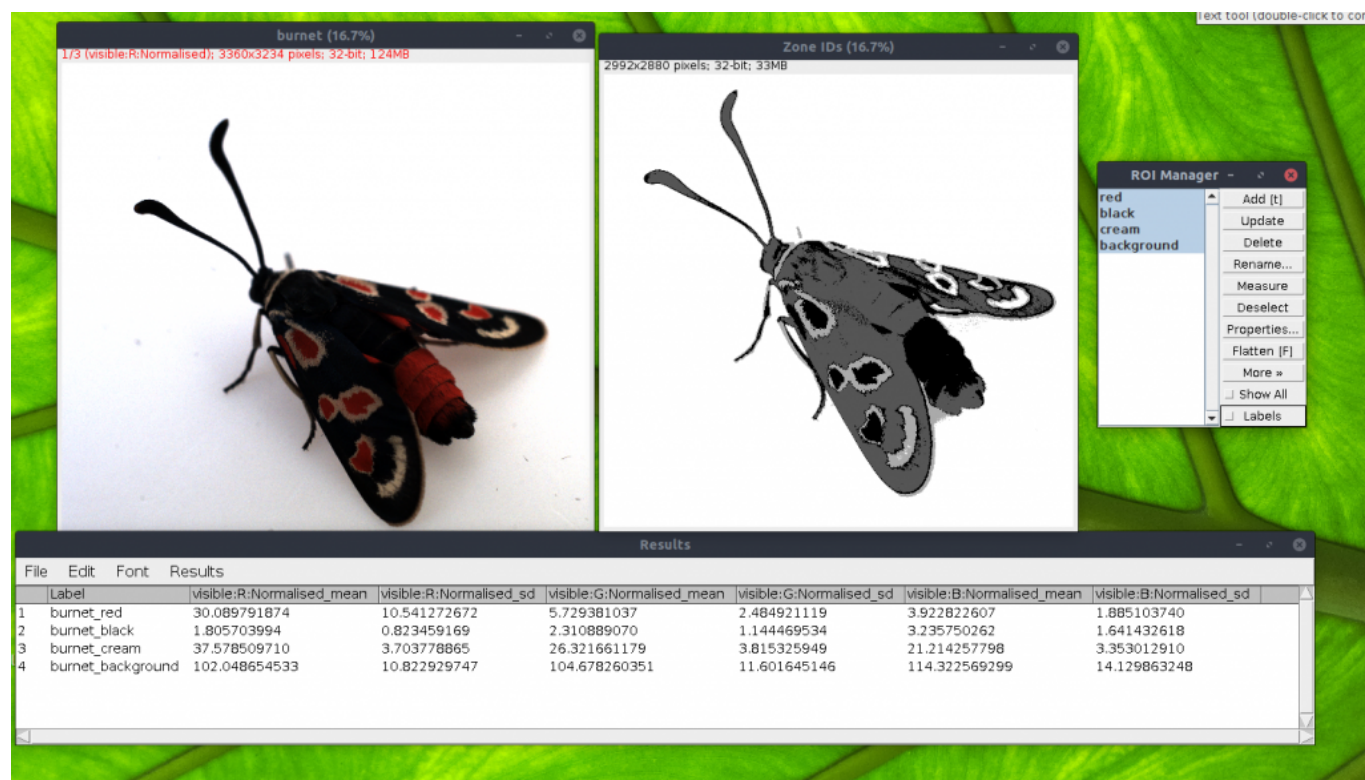
Input Image

This process requires a 32-bit image stack, such as a calibrated (linear & normalised) multispectral image stack, or a cone-catch image. It also requires a results table open which has the mean and standard deviations of each cluster for each image channel. The columns must be labelled with the channel name followed by "_mean" and "_SD" for means and standard deviations respectively. These results can easily be created by selecting one ROI as an example of each cluster in a multispectral image, and pressing "R" on the keyboard to provide means and standard deviations for each ROI. e.g. in the example below, a small section of "red" moth wing was selected as the first ROI, a small section of "black" for the black wing parts etc...

Running Naive Bayes Clustering

With the input image highlighted run:

plugins > micaToolbox > QCPA > Naive Bayes Quantisation



Output

The script outputs an image where each pixel value specifies a cluster ID. In the above example, the first measurement is "red" on the burnet moth wing, so parts of the image classified as "red" will have a pixel value of 1. "black" will be 2, etc... up to the full number of clusters (it can deal with up to 255 clusters, but if you have that many you should probably be using a different method!)

The output clusters can be converted into ROIs by using the "[Cluster Particle Analysis](#)" tool:

plugins > micaToolbox > QCPA > Cluster Particle Analysis

Negative Values in Cone Catch Images

<http://www.empiricalimaging.com/knowledge-base/negative-values-in-cone-catch-images/>

When creating a cone-catch image it is fairly common to have some negative values, which are highlighted in red overlay (plus the log lists the frequency of negative values in each image). Negative values can be caused by two main factors:

- Image calibration imperfections. These could be due to incorrect measurement of the reflectance of the grey standards used to calibrate the image. This is particularly important when two or more standards are used. Negative values can also occur where the intensity of light falling on the standards does not match the intensity on the samples being measured (e.g. if the sample is slightly in shade compared to the standards, it could be seen as being negative). This highlights the importance of correct use of [grey standards](#) and [lighting](#). These negative values tend to occur in very dark sections of the image, and setting the values to very low values (near zero default) is appropriate.
- Alternatively, when negative values occur in very colourful samples this can imply the colour is out of gamut of either the camera or visual system. In this instance the numbers being produced (both in the channel with negative values and the other channels) should be treated with caution. You should note in your results that these colours are likely to be at or beyond the limit of the equipment's or cone-mapping model's colour reproduction abilities.

Any image processing step which requires calculation of RNL-space values (e.g. various aspects of the QCPA framework, JND calculation etc...) cannot accept negative values. In this case, and if deemed appropriate you can have the 'convert to cone catch' script change all negative values to a near-zero number which is accepted by the RNL calculations.

Particle Analysis

<http://www.empiricalimaging.com/knowledge-base/particle-analysis/>

Particle analysis provides information on the spatial properties of clusters/patches in an image, such as the shape, size angle and distribution of particles within a given cluster/patch. This is typically run within the QCPA framework, but can be used in isolation on any clustered image.

Input Requirements:

This method is integrated into the [QCPA framework](#), and can simply be selected as an additional analysis to run.

Alternatively, you can run the particle analysis on a suitable image by running:

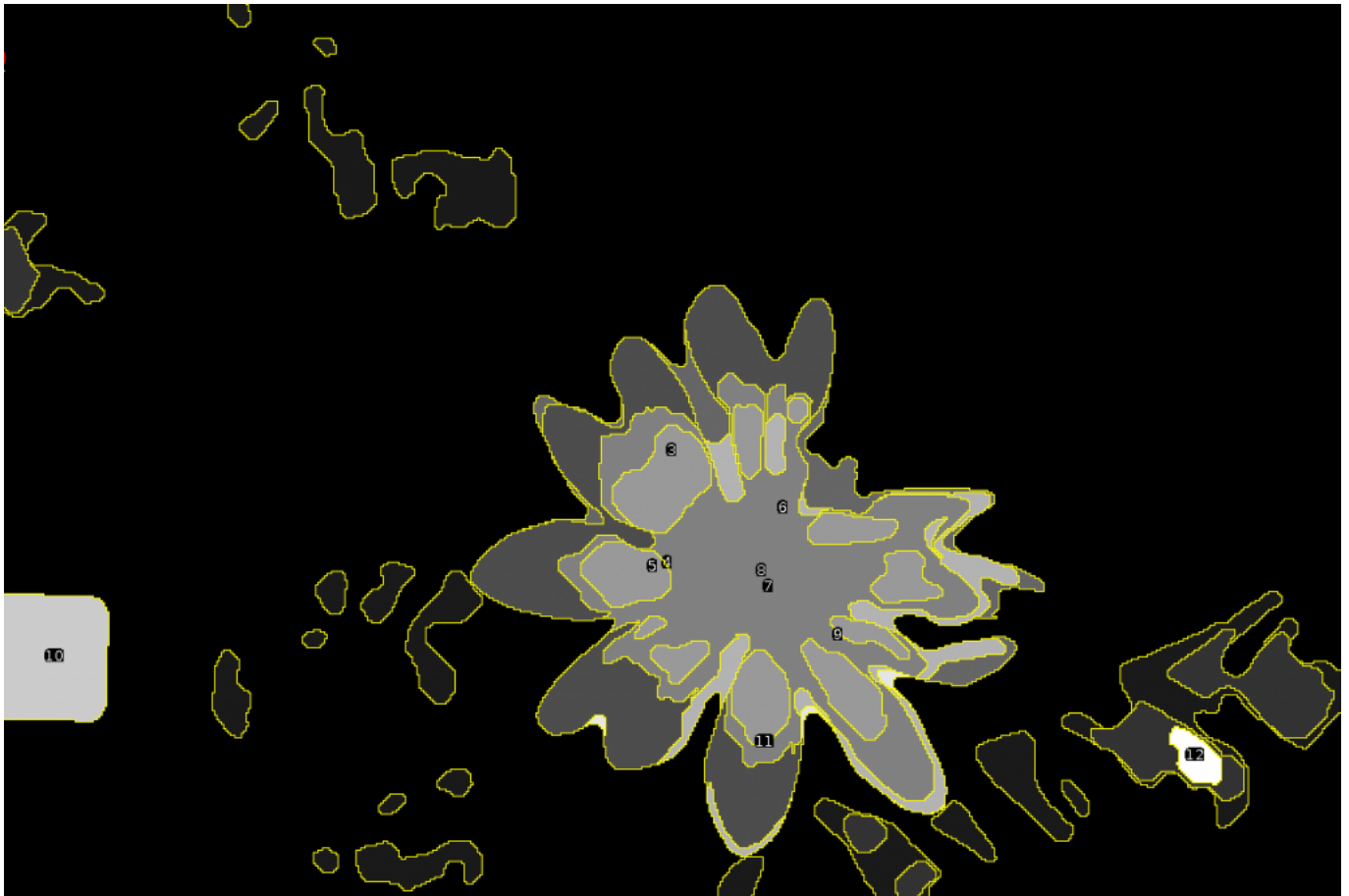
plugins > micaToolbox > QCPA > Cluster Particle Analysis

The input image must be a cluster ID image (such as that produced by the [Naive Bayes Clustering](#) or [RNL Clustering](#) filters). In these images each pixel denotes a cluster ID.



Output:

The particle analysis creates ROIs specific to each cluster ID. These ROIs can be useful for selecting regions of other instances of the same image (e.g. go to the RNL Rank Filtered image and use the cluster ROIs to measure specific image sections). The particles analysis also provides two tables; one summary table, which shows metrics for each cluster/patch ID across all particles making up that cluster/patch. The second table shows the data for each individual particle (each cluster/particle is likely to be made up of a number of different patches).



Pattern Energy (Granularity) Analysis

<http://www.empiricalimaging.com/knowledge-base/pattern-energy-granularity-analysis/>

We provide tools for performing a pattern analysis based on Fast Fourier bandpass filtering, often called a granularity analysis. This form of analysis is increasingly widely used to measure animal markings (Godfrey et al., 1987; Stoddard and Stevens, 2010), and is loosely based on our understanding of low-level neuro-physiological image processing in numerous vertebrates and invertebrates. Each image is filtered at multiple spatial frequency scales, and the “energy” at each scale is measured as the standard deviation of the filtered pixel values. The pattern energy of non-rectangular regions of interest are measured by extracting the selection so that all surrounding image information is removed (i.e. the selection area is measured against a black background). This image is duplicated and the selection area is filled with the mean measured pixel value (to remove all pattern information inside the selection area). Identical bandpass filtering is performed on both images and the difference between the images is calculated before measuring its energy for a shape-independent measure of pattern.

To perform pattern analysis using the batch measurement tool one must first select which channel to use for pattern analysis. Pattern processing in humans is thought to rely on the luminance channel (which is the combination of LW and MW sensitivities) and double cones in birds (Osorio and Vorobyev, 2005). If images are not being converted to cone catch quanta then the green channel is recommended (Spottiswoode and Stevens, 2010), or a combination of red and green (which is available as an automated option). Next, the desired measurement scales must be selected in pixels, along with the desired incrementation scale (e.g. linear increases of two pixels would provide measurements at 2, 4, 6, 8 pixels, and so on). A multiplier can be used instead, for example multiplying by two to yield measurements at 2, 4, 8 16 pixels and beyond. A size no larger than the scaled image dimensions should be used to cover the entire available range. The number of scale increments should be judged based on processing speed, although increasing the number of scales measured will yield progressively less additional information. We find that increasing the scale from 2 by a multiple of 2 up to the largest size available in the scaled images produces good results in many situations.

Summary statistics (Chiao et al., 2009; Stoddard and Stevens, 2010) of the pattern analysis are saved for each region of interest (or pooled regions). These include the maximum frequency (the spatial frequency with the highest energy; i.e. corresponding to the dominant marking size), the maximum energy (the energy at the maximum frequency), summed energy (the energy summed across all scales, or amplitude; a measure of pattern contrast), proportion energy (the maximum energy divided by the summed energy; a measure of pattern diversity, or how much one pattern size dominates), mean energy and energy standard deviation. In addition the energy spectra (the raw measurement values at each scale) can be output and used for pattern difference calculations (see below). Pattern maps can also be created – these are false colour images (Image 1) of the filtering at each scale – produced for subjectively visualising the process.

$$Pattern_{diff} = \sum_{s=max}^{s=min} \left| A_s - B_s \right|$$



Image 1: Example of pattern energy map outputs

Animal markings and natural scenes often have pattern energy spectra with more than one peak frequency, meaning the pattern descriptive statistics (above) can arbitrarily jump between peaks with similar energy levels in different samples (Image 2). An alternative approach we recommend when directly comparing two samples, rather than deriving intrinsic measurements to each sample, is to calculate the absolute difference between two spectra (A and B) across the spatial scales measured s :

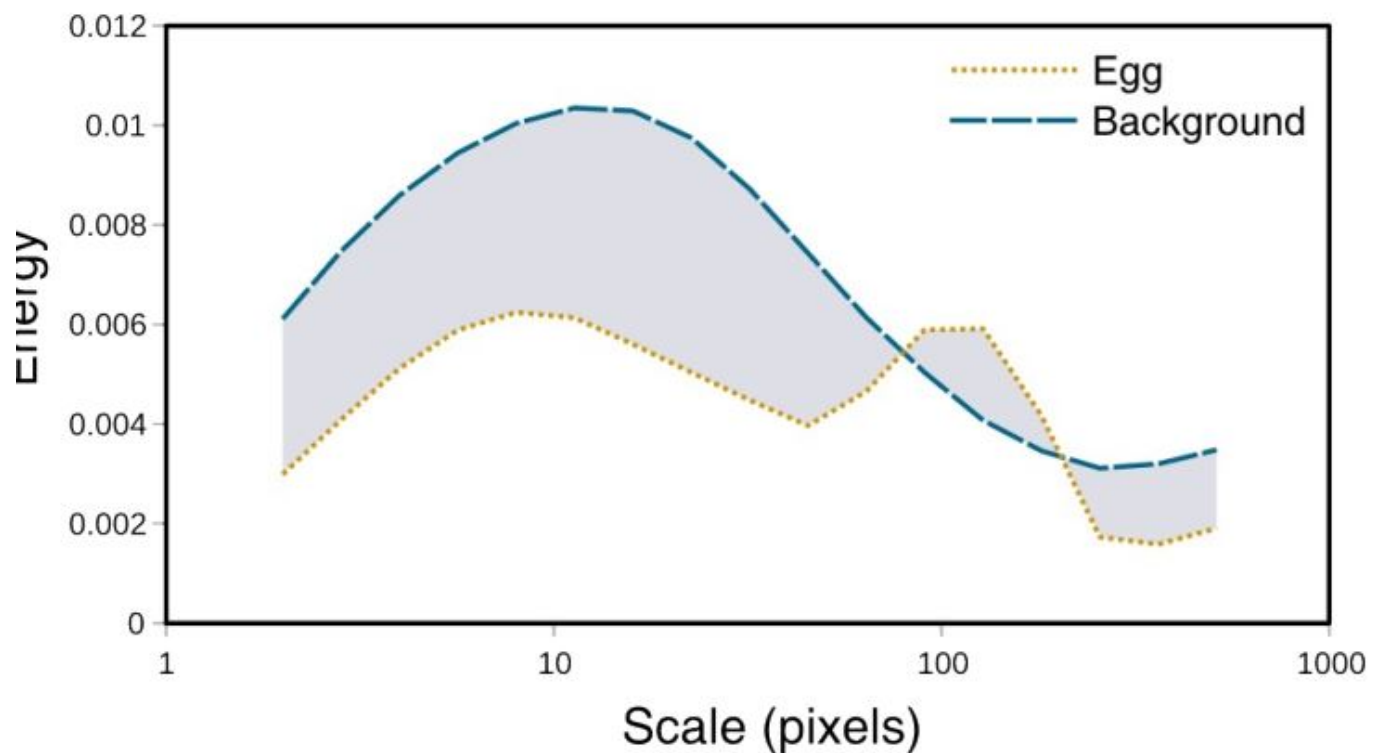


Image 2: Pattern spectra for an egg against its background. Note there are two peaks in spatial energy for the egg, implying it has both small spots and large spots. The peak frequency could therefore either be around 10 pixels or 100 pixels.

This is equivalent to the area between the two curves in Image 2. Any two patterns with very similar amounts of energy across the spatial scales measured will produce low pattern difference values irrespective of the shape of their pattern spectra. These differences will rise as the spectra differ at any frequency. As with colour, these differences can be calculated automatically between regions of interest, within or between images from the pattern energy spectra (above) using the “Pattern and Luminance Distribution Difference Calculator”. Luminance distribution differences are calculated similarly, summing the differences in the number of pixels in each luminance bin across the entire histogram. This method for comparing luminance is recommended when the regions or objects being measured do not have a normal distribution of luminance levels. For example, many animal patterns have discrete high and low luminance values (such as zebra stripes or egg maculation), in which case a mean value would not be appropriate. Comparing the luminance histograms overcomes these distribution problems.

References:

Chiao, C.-C., Chubb, C., Buresch, K., Siemann, L., and Hanlon, R.T. (2009). The scaling effects of substrate texture on camouflage patterning in cuttlefish. *Vision Res.* 49, 1647 - 1656.

Godfrey, D. Lythgoe, J.N., and Rumball, D.A. (1987). Zebra stripes and tiger stripes: the spatial frequency distribution of the pattern compared to that of the background is significant in display and crypsis. *Biol. J. Linn. Soc.* 32, 427-433

Stoddard, M.C., and Stevens, M. (2010). Pattern mimicry of host eggs by the common cuckoo, as seen through a bird's eye. *Proc. R. Soc. B Biol. Sci.* 277, 1387-1393

Region Of Interest (ROI) selection tips & tricks

<http://www.empiricalimaging.com/knowledge-base/region-of-interest-roi-selection-tips-tricks/>

One of the great benefits of the micaToolbox and its components is that it runs on ImageJ which comes with an extensive suite of tools for image manipulation. Particularly useful is the ability to do very easy and efficient selecting of Regions of Interest (ROIs) which can be used to then run pattern analyses (e.g. QCPA components) on specific image regions. E.g. if you wanted to compare an animal to its background. The [ImageJ manual](#) provides detailed information on how to achieve the best selection possible. Valuable functions include the ability to select regions and then to either combine these or subtract one from the other.

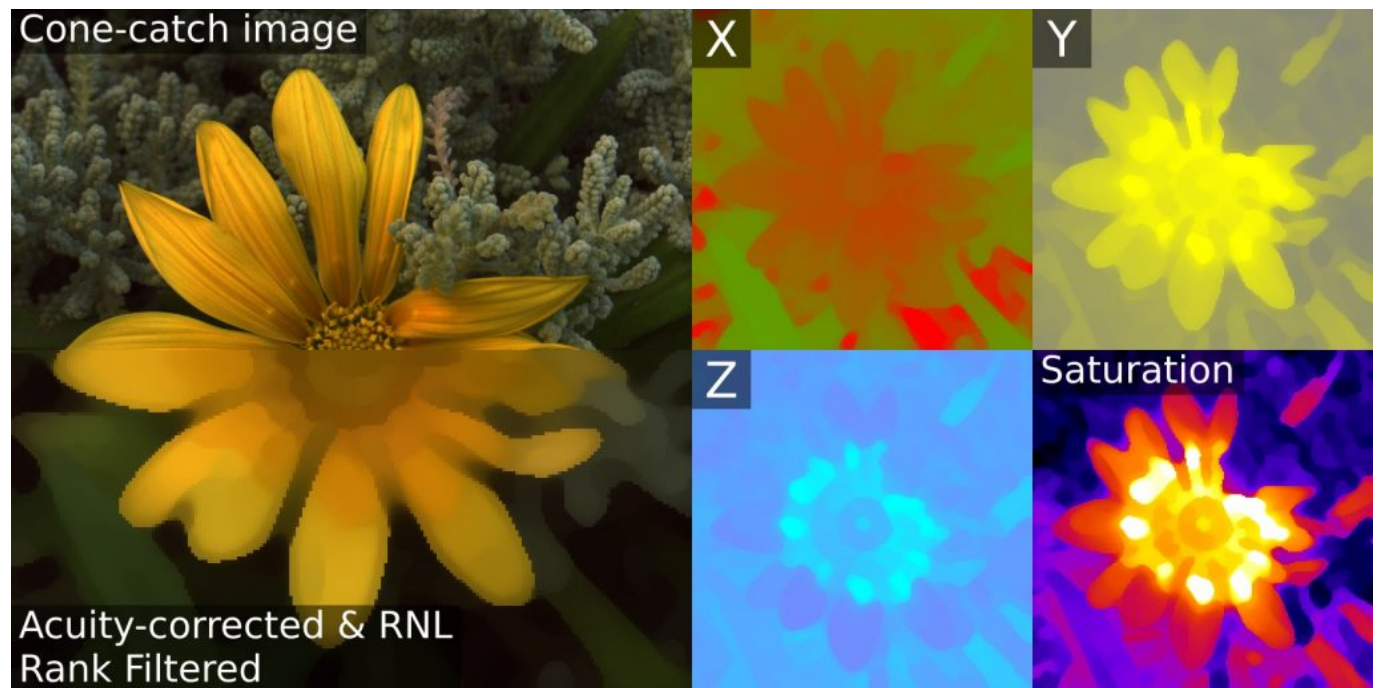
The toolbox has built upon the basic ImageJ functionality by providing the ability to create automatically labelled and numbered ROIs with a single button click, and then to perform basic image measurements of these ROIs immediately. This tool is started automatically whenever an mspec image is created or loaded, however it can also be started manually by going:

plugins > micaToolbox > Tools > Save ROIs

Additionally, images can be segmented using the [Naive Bayes Clustering](#) or [RNL Clustering](#) tools, and these clusters can be converted to ROIs by running the [Particle Analysis](#) tool. The easiest way to perform these functions together is by using the [Run QCPA Framework](#) tool.

RNL Chromaticity XYZ Saturation Images

<http://www.empiricalimaging.com/knowledge-base/rnl-xyz-images/>



The receptor noise limited (RNL) chromaticity colour space is convenient because the Euclidean distance between any two points in this space is equal to the Delta-S of the RNL model (in units of "just noticeable differences", JNDs). This means that if any two colours have a Euclidean distance of less than ~1-3 JNDs it is unlikely the two colours will be distinguishable. Where we know only a limited amount about the colour vision system of a given receiver this space is particularly useful and has been found to describe behavioural data well in a number of species (e.g. bees, humans, birds, lizard, and fish).

Input Requirements

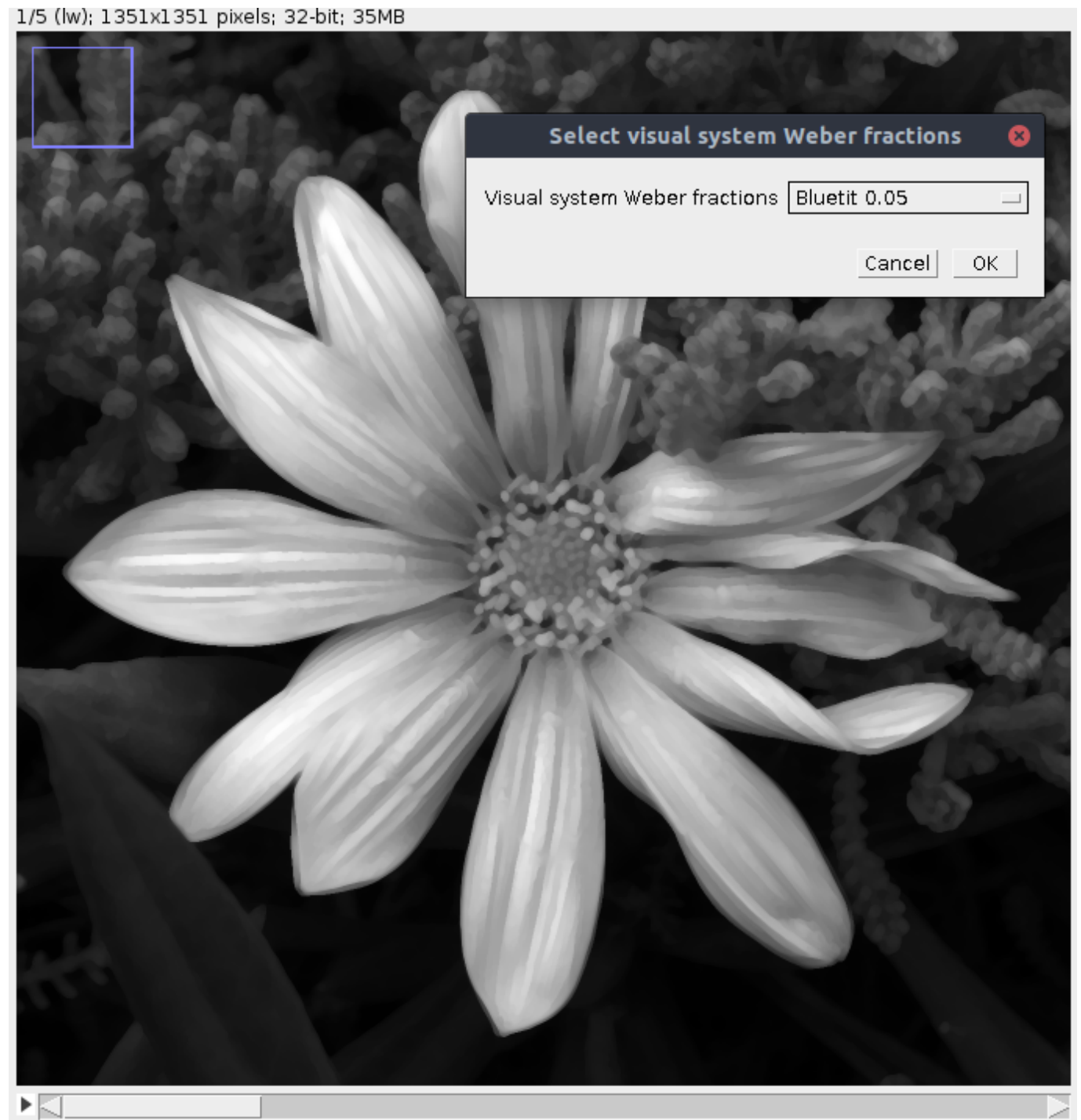
This tool requires a 32-bit [cone-catch image](#) where the slice labels are named correctly (matching the names in the Weber fractions file for the visual system, or you can specify your own), and the channels should ideally go from longest to shortest wavelength sensitivity (this is the default for the toolbox).

Running the Tool

We recommend using the [QCPA framework](#) to perform this function, however you can run it on its own. Select a [cone catch image](#) and run:

plugins > micaToolbox > QCPA > Cone Catch to RNL Chromaticity

You will be asked to specify the Weber fractions for each channel. These are typically based on relative cone abundance if behavioural data are not available. You can either select the correct species (in which case the receptor names in the image and specification file must match, you will be notified if they do not), or specify you own. If the cone-catch image stack has a final luminance channel (as is required for the [QCPA framework](#)), this will be ignored appropriately.



Output

The output will vary depending on the number of receptor channels. Dichromatic cone-catch images will create a single "X" channel (showing opponency between LW and SW). Trichromatic cone-catch images will create a stack with an X and a Y channel ("X" is opponency between LW and MW (red-green), "Y" is opponency between the combination of LW and MW to SW (yellow-blue)). Tetrachromatic images are the same as trichromatic, but with a third "Z" channel, which is opponency between the combination of LW, MW and SW to UV.

The final slice shows saturation. This is the distance from each pixel's XYZ coordinates to the achromatic point.

The output is simply a 32-bit greyscale stack suitable for measuring pixel values. Note you can also use these image for colour pattern analysis, such as measuring [GabRat edge disruption](#). You may want to adjust the [preview brightness/contrast](#) of the stack.

1/4 (X); 1351x1351 pixels; 32-bit; 28MB



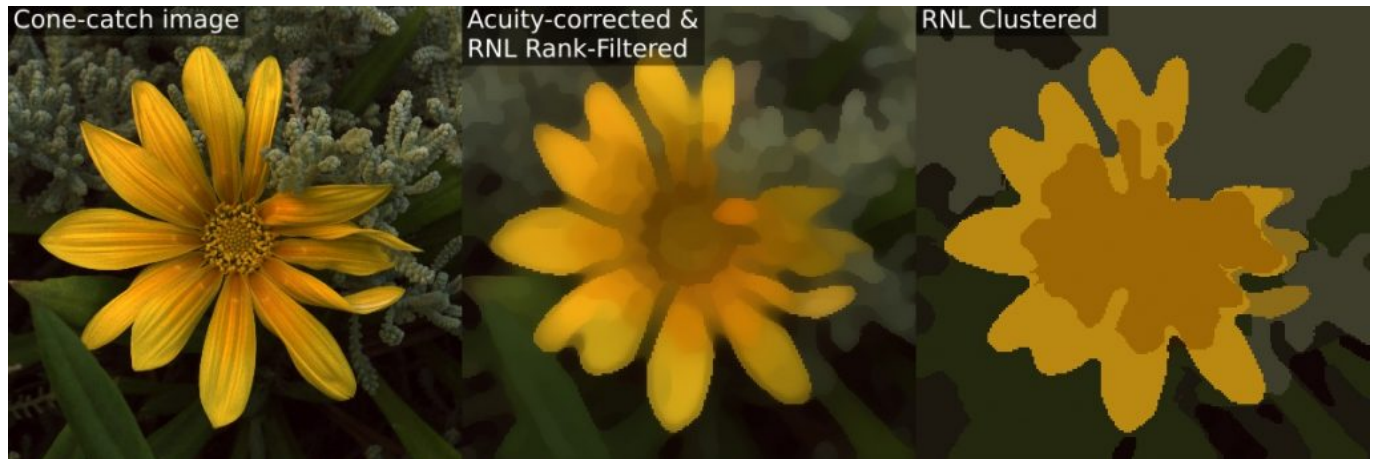
To create a false-colour image like the ones at the top of this page follow these steps:

- Select the channel you wish to use from the output stack (e.g. X).
- Run *Image > Duplicate* and select only the current channel (do not tick the "Duplicate Stack" tickbox).
- Copy the current image (CTRL+C).
- Run *Image > Stack > Add Slice*.
- Paste: (CTRL+V).

- Run *Process > Math > Multiply*, and select "-1". And select "No" when asked whether to process the entire stack.
- Run *Image > Color > Make Composite*, and select "Composite".
- The default settings will now show a red-green opponent image. For a trichromat the RNL X channel is represented by red-green, and the RNL Y channel is blue-yellow. You can select whichever colours you want by selecting the first or second slice in the image stack and selecting a "lookup table". e.g. *Image > Lookup Tables > Yellow* to select yellow. For a dichromat the X channel should be blue-yellow. For a trichromat X = red-green, Y = blue-yellow. Tetrachromats are the same as dishchromats, but the Z channel could e.g. be cyan-magenta.
- Importantly, the min and max viewing range must be set, and must be equal in positive and negative values (e.g. -10 and +10, or -30 and +30). If you do not do this the achromatic point will not be in the middle. Go *Plugins > Measure > Set Min And Max*, and choose values which give a suitable colour range. It's important that whatever range you use is standardised and equal across images if you're using these images to show differences between samples.
- The "Saturation" channel can simply be shown on its own in greyscale, or you can choose some other lookup table to show it. e.g. the example at the top of this page is using the "Fire" lookup table.
- With a lookup table applied and min & max values set you can simply save the current image as a JPG.

RNL Clustering

<http://www.empiricalimaging.com/knowledge-base/rnl-clustering/>



The RNL clustering technique was developed specifically for the [QCPA workflow](#), parts of which require a colour image be segmented into different pattern elements or 'clusters'. Unlike previous approaches this technique uses log receptor noise limited (RNL) modelling of Delta-S distances and perceptual thresholds (which can be behaviourally validated), for chromatic (colour) and achromatic (luminance) clustering.

Advantages:

- The algorithm uses animal vision modelling when deciding whether any two colour patches will be combined.
- The algorithm only needs to be told what perceptual thresholds are required for the given visual system, you do not need to know how many different pattern elements or clusters there are in the image (unlike [naive Bayes](#) or k-mean clustering, for example).
- The algorithm supports di- tri- and tetrachromatic visual systems.
- The influence of colour and luminance can be combined or separated in clustering.
- There is no random element to the clustering - the same input will always create the same output with constant settings - and the method is robust against image rotation.
- Increasing the size of the receptive field, while simultaneously decreasing the number of discrete clusters is computationally efficient, and matches plausible colour processing steps in the brain.
- Clustering can be performed on a sub-section of any image (with ROIs of any shape).

Input Image

We recommend this method should be used following acuity control ([FFT-based AcuityView](#) or [Gaussian acuity control](#)) and [RNL ranked filtering](#). RNL clustering requires a cone-catch image stack. To create a cone-catch image you must generate or

load a [calibrated mspec image](#) and [convert to cone catch using a model](#). The first slices in the stack must be the chromatic channels, with a final luminance slice in the stack. Here are examples of input image stacks for tetra-, tri- and di-chromatic modelling:



If the cone-catch model doesn't create a final luminance channel automatically (e.g. the "double" cones in birds), you will need to add your own. This will vary between visual system. There is a tool provided for [adding a luminance channel](#).

plugins > micaToolbox > Image Transform > Create Luminance Channel

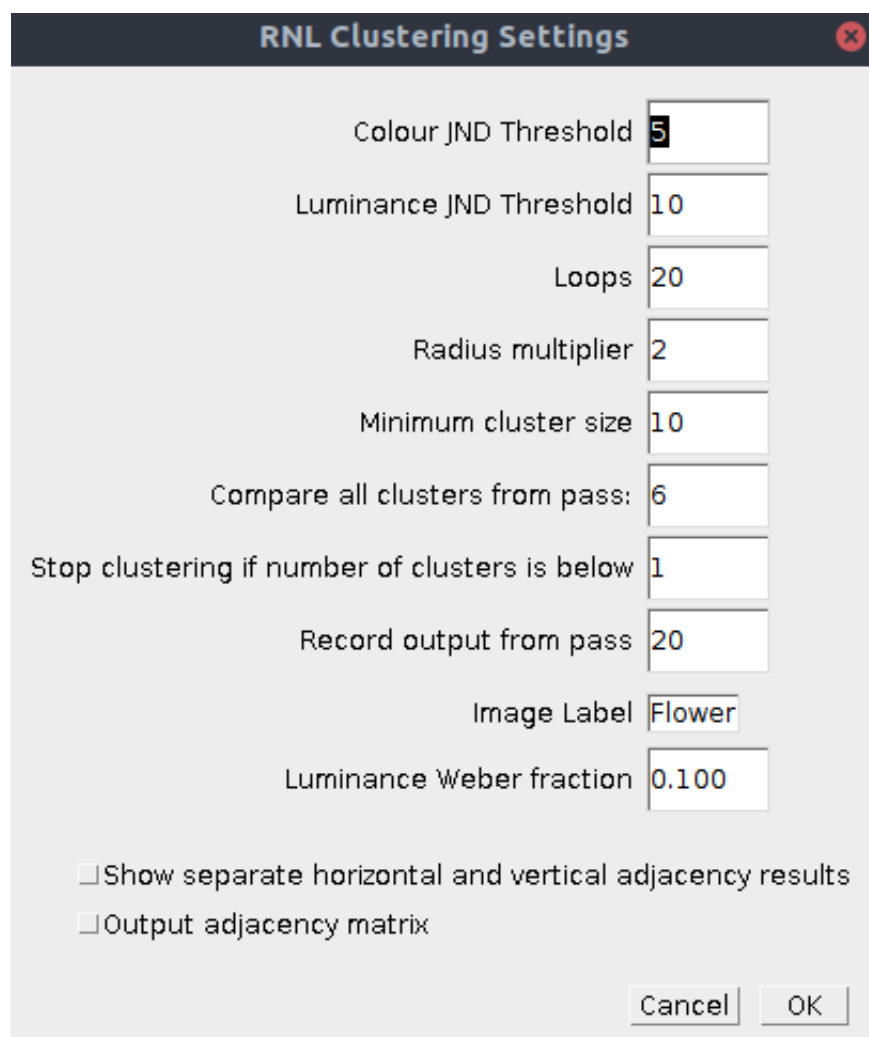
This tool adds a final luminance channel at the end of the cone-catch image stack, based on an equal average of all the selected channels.

Any pixel with zero values or negative values in any channel is ignored in clustering (the RNL model can't deal with them, but this is also a method for screening out sections of the image which you do not want to include in clustering).

Running RNL Clustering

With the appropriate image open in ImageJ, we recommend running the [QCPA framework](#) and selecting "RNL Clustering" in the relevant drop-down menu, together with the relevant visual system. We recommend this method because it also allows integration of spatial acuity controls, noise reduction and sharp-edge recovery through the [RNL Ranked Filter](#), and saves the settings previously used:

plugins > Multispectral Imaging > QCPA > Run QCPA Framework



RNL Clustering Settings

Colour JND Threshold	<input type="text" value="5"/>
Luminance JND Threshold	<input type="text" value="10"/>
Loops	<input type="text" value="20"/>
Radius multiplier	<input type="text" value="2"/>
Minimum cluster size	<input type="text" value="10"/>
Compare all clusters from pass:	<input type="text" value="6"/>
Stop clustering if number of clusters is below	<input type="text" value="1"/>
Record output from pass	<input type="text" value="20"/>
Image Label	<input type="text" value="Flower"/>
Luminance Weber fraction	<input type="text" value="0.100"/>

☐ Show separate horizontal and vertical adjacency results
☐ Output adjacency matrix

Alternatively the script can be run directly:

*plugins > measure > RNL Clustering**Colour JND Threshold*

The log chromatic RNL Delta-S threshold below which clusters can be combined. These numbers can be behaviourally validated, but sensible numbers are in the range of around 1-5. Set to e.g. 9999 to disable the chromatic threshold.

Luminance JND Threshold

The log luminance RNL Delta-S threshold below which clusters can be combined. These numbers can be behaviourally validated, but sensible numbers are in the range of around 1-10. Set to e.g. 9999 to disable the luminance threshold.

Loops

Maximum number of passes of the hierarchical agglomerative clustering algorithm. Images are typically clustered after around 10-12 passes, however the later passes become very fast. 20 is a suitable number for almost all images.

Radius Multiplier

The initial radius over which clusters are compared to their neighbours is 1 pixel. With each pass the radius can be set to get larger (simulating an increasing receptive field size). The radius is multiplied by this number with each pass, so a value of 2 means doubling the size of the receptive field with each pass.

Minimum Cluster Size

If a cluster is below the minimum cluster size it is forced to combine with the nearest (in chromatic and luminance Delta-S terms) cluster within the relevant radius. The minimum cluster size increases linearly with each pass so that by the final pass it is equal to the specified number. This is useful for eliminating single-pixel clusters.

Compare all clusters from pass

After a specified number of passes it becomes more computationally efficient to compare every cluster to every other cluster in the image (rather than searching spatially for each cluster's nearest neighbour with ever increasing receptive field sizes). Typically this becomes optimal after approximately 5-7 passes depending on the image size.

Stop clustering if number of clusters is below

If the number of clusters reaches the specified value before the clustering has finished, it stops clustering at the current pass.

Record output from pass

This option allows the user to see the output from passes before the final pass (useful if you're interested in seeing how the clustering progresses over subsequent passes).

Image Label

Specify an image label. This will be the label used in the results table, so is useful to define for subsequent statistics comparing images.

Luminance Weber fraction

Specify the luminance Weber fraction. Unless you have behavioural data to justify a particular value, numbers in the range of 0.1 to 0.2 are typical.

Show separate horizontal and vertical adjacency results

If ticked, the code also outputs the different horizontal and vertical pixel transition data separately (the default combines horizontal and vertical transitions)

Output adjacency matrix

Option for also outputting the counts of pixels of each cluster which are adjacent to each other. Useful if you wish to perform your own matrix calculations.

Output

The RNL clustering outputs two images; first, an image with the suffix "_Cluster_IDs", where each cluster is shown numerically from 1 to n clusters. The numbers conform to the cluster ID numbers listed in the results table. Note that is image, plus the "Cluster Results" table are required for performing the adjacency analysis on an ROI. The second image (with the suffix "_Clustered") is a stack showing the mean cluster cone-catch values for each receptor class, and can - for example - be used for creating a false colour output image.

Two results tables are produced; "Summary Results" lists all the BSA/adjacency/average image statistics, with one line per image. "Cluster Results" lists one row for each cluster in the image, detailing average cluster statistics.

Details of the Algorithm

The algorithm uses agglomerative hierarchical clustering; at the start of clustering each pixel is its own unique cluster. Each cluster is then compared to its neighbouring clusters within a given radius (receptive field). Each cluster is then combined with its neighbour which shares the lowest Delta-S value (in chromatic/colour distance, achromatic/luminance distance, or both), only if the distances are below the respective colour and luminance thresholds. Clusters can be combined with multiple neighbours if, for example, cluster A is nearest to cluster B, but cluster B is nearest to cluster C. Meaning it is possible that a string of related colours can be combined in one pass. The algorithm then repeats the clustering over a number of passes, increasing the receptive field of the cluster comparisons with each pass. By around pass 5-7 a typical 1 megapixel image would be reduced to a few thousand clusters, at which point n-by-n cluster comparisons become more computationally efficient (conceptually similar to increasing the receptive field to the entire image).

Modifying the Algorithm

Further details of the algorithm are outlined in the original QCPA publication and suppl. material. The method for combining colour and luminance thresholds can be manually changed in the JAVA source file.

Alternative Clustering Methods

The [naive Bayes clustering tool](#) can be used for clustering in cases where you know exactly how many cluster you want/expect (and can justify this because the sample/organism clearly has a limited number of discrete colours which can easily be distinguished).

RNL Ranked Filter

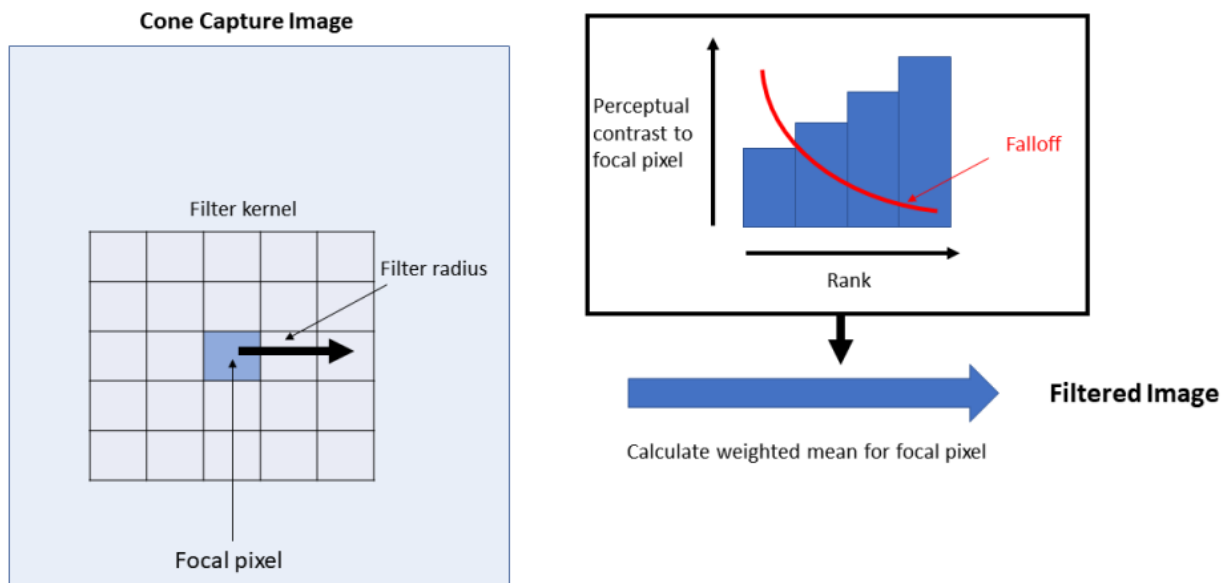
<http://www.empiricalimaging.com/knowledge-base/rnl-ranked-filter/>



The Receptor Noise Limited (RNL) Ranked Filter performs noise reduction while preserving chromatic or luminance edges, and is an important step in the [QCPA framework](#). Importantly, it is also able to reconstruct sharp edges in images which have undergone smoothing (e.g. [acuity-correction](#) smoothing). The latter function is particularly useful for [clustering](#) or [Local Edge Intensity Analysis](#), where intermediate colours created by the smoothing process would interfere with the processing.

This filter compares each pixel in the image with its neighbouring pixels within a given radius. The pixels are compared in terms of chromatic and luminance RNL delta-S distances for di- tri- or tetra-chromatic images. The pixels within the kernel are ranked from nearest to furthest (in combined chromatic and luminance delta-S distance) from the focal pixel. The focal pixel is then exponentially smoothed with its neighbours, so that it is averaged most with the pixels of a similar colour and least with dissimilar pixels. The shape of the exponential falloff curve can be specified by the user, in addition to the kernel radius and photoreceptor abundance and noise levels. The filter can be repeated a number of times to enhance the effect and reconstruct sharp edges. The result of the filter is the smoothing of perceptually similar areas while maintaining or recreating edge information.

To combine this filter with acuity control, ensure the number of pixels per MRA match the radius of the RNL Ranked Filter. In practice a value of 5 is ideal; below this there is the potential for loss of spatial information, and above this the RNL Ranked filter is prohibitively slow.



A schematic representation of the RNL Ranked Filter

Input Requirements

RNL clustering requires a cone-catch image stack. To create a cone-catch image you must generate or load a [calibrated mspec image](#) and [convert to cone catch using a model](#). The first slices in the stack must be the chromatic channels, with a final luminance slice in the stack. Here are examples of input image stacks for tetra-, tri- and di-chromatic modelling:



If the cone-catch model doesn't create a final luminance channel automatically (e.g. the "double" cones in birds), you will need to add your own. This will vary between visual system. There is a tool provided for [adding a luminance channel](#).

plugins > micaToolbox > Image Transform > Create Luminance Channel

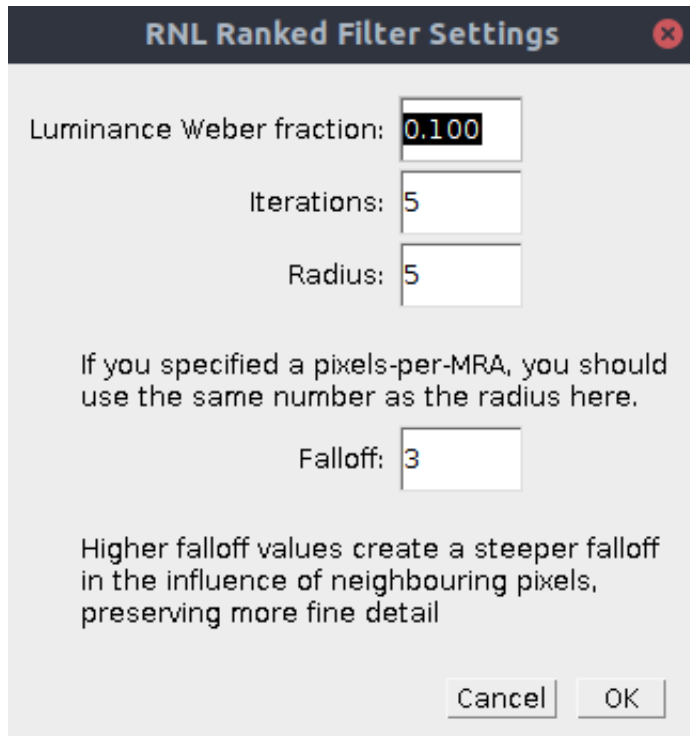
This tool adds a final luminance channel at the end of the cone-catch image stack, based on an equal average of all the selected channels.

Any pixel with zero values or negative values in any channel is ignored in processing (the RNL model can't deal with them, but this is also a method for screening out sections of the image which you do not want to include in filtering).

Running RNL Ranked Filter

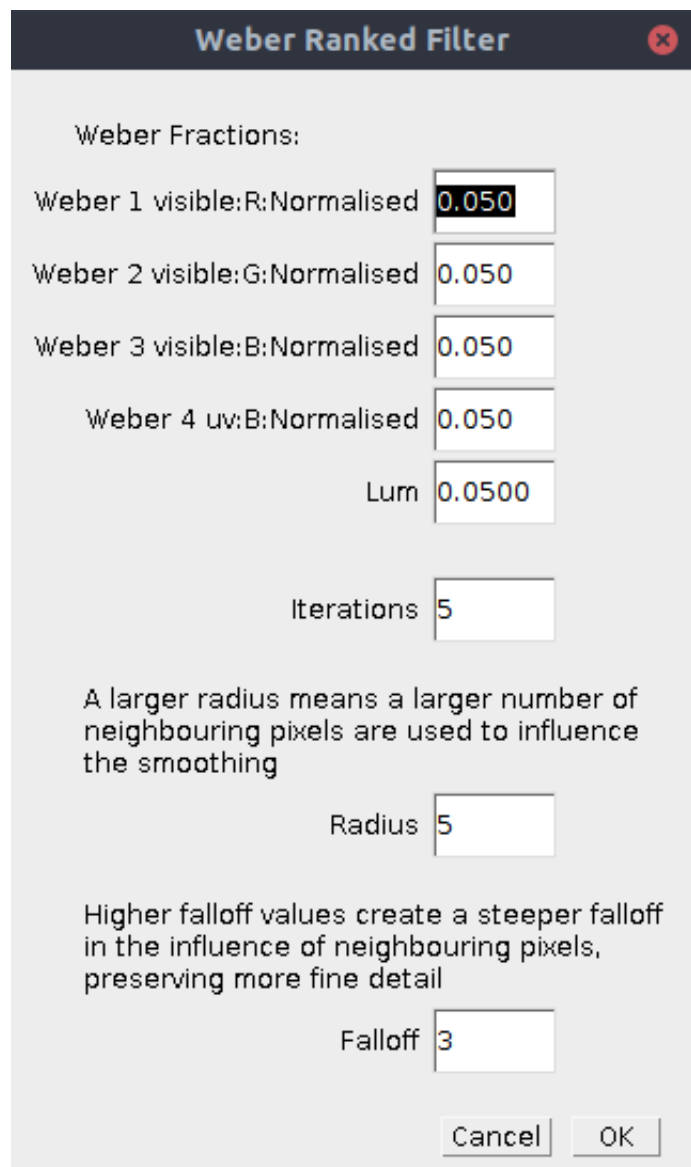
The filter is easiest to use within the [QCPA framework](#). Simply run the framework and tick the RNL Ranked Filter box.

plugins > micaToolbox > QCPA > Run QCPA Framework



Alternatively the filter can be run on its own:

plugins > Measure > RNL Ranked Filter



Weber Ranked Filter [X]

Weber Fractions:

Weber 1 visible:R:Normalised

Weber 2 visible:G:Normalised

Weber 3 visible:B:Normalised

Weber 4 uv:B:Normalised

Lum

Iterations

A larger radius means a larger number of neighbouring pixels are used to influence the smoothing

Radius

Higher falloff values create a steeper falloff in the influence of neighbouring pixels, preserving more fine detail

Falloff

Weber fractions

Specify the desired Weber fractions for your visual system for each receptor class (there are based on cone abundance and receptor noise). If you use the [QCPA Framework](#) these can be selected automatically from a list of visual systems in the first dialog box.

Lum (Luminance Weber Fraction)

Specify the Weber fraction to be used for the luminance distance calculation (it uses the log "Siddiqi et al. 2004" method).

Iterations

Specify how many repeats of the filter to apply. We recommend 5 as there are diminishing returns past this number of repeats.

Radius

The kernel radius. Use a value equal to or just below the "pixels per MRA" value of the image if using this after acuity control. This will ensure the filter can reach across the area of blurred pixels at a boundary to reconstruct the edge.

Falloff

Specify the shape of the falloff curve used for exponential

smoothing of the focal pixel with its ranked neighbours. 3 is a sensible number. Higher values create a steeper falloff curve, meaning the focal pixel is averaged with a smaller number of more similarly coloured pixels. Lower numbers flatten the curve, so that it eventually becomes a mean filter within the given radius.

Output

The script simply outputs a filtered image.

Running the QCPA Framework

<http://www.empiricalimaging.com/knowledge-base/running-the-qcpa-framework/>



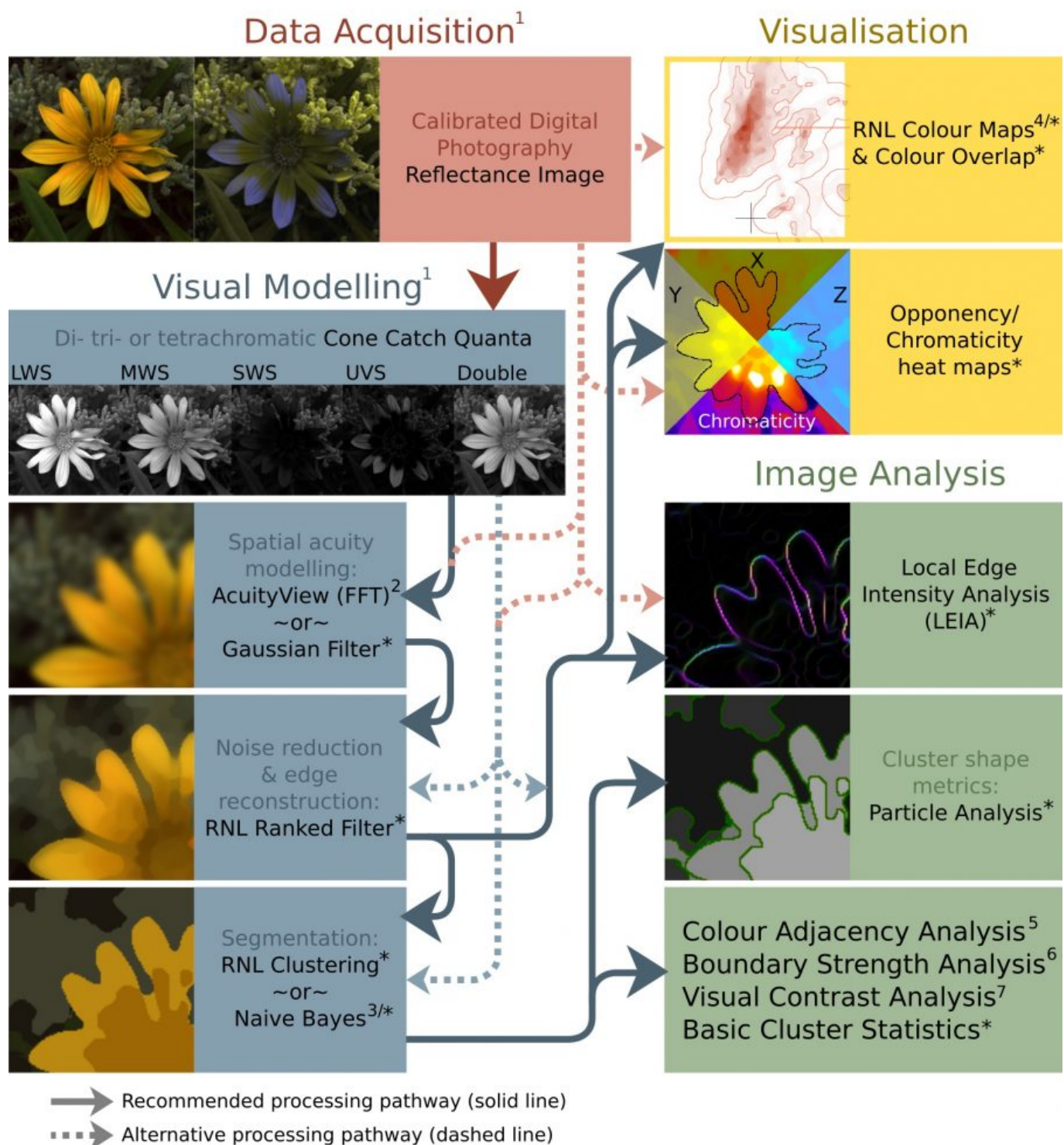
The Quantitative Colour and Pattern Analysis (QCPA) framework is an image processing workflow which takes an "animal vision" cone-catch image, applies spatial acuity and viewing distance correction, and then performs a series of sophisticated colour and pattern analysis procedures. Historically colour and pattern have been analysed separately, however this framework can perform analyses based on di- tri- and tetra-chromatic visual systems.

The framework uses the Vorobyev and Osorio (1998) Receptor Noise Limited (RNL) model throughout, which has been behaviourally validated in a range of species.

Many of the analysis procedures used by the QCPA framework require the image to be reduced to a small number of discrete colour patches, so the process can utilise novel image segmentation algorithms based on colour discrimination thresholds.

While there is a clear recommended processing pathway (see figure below) the framework is highly flexible, and can include or exclude various steps, and in many cases there are different options for achieving each step.

Quantitative Colour Pattern Analysis Framework



Video Guides

<https://youtu.be/ENZ43CVvBvo>

<https://youtu.be/vda3G1DVyis>

Input Requirements

The QCPA framework requires a 32-bits/channel [cone-catch image](#). For acuity control you will need to know the receiver's spatial acuity ([see list](#)), and also need a [scale bar](#) in the image (unless you know the image's angular width). For RNL modelling ([RNL Ranked filter](#), [RNL clustering](#), and many of the analyses) you will need to know the Weber fractions for each receptor class, or failing that the approximate cone ratios (a range of species are included in the toolbox). For [naive Bayes clustering](#) you will either need to have a results table containing the mean and standard deviation of pixel values in each channel, for each patch, or two or more ROIs specified which will be measured and used for the naive Bayes clustering. Additionally, you can have a region of interest (ROI) specified, where - for example - you can analyse a section of the image completely independently of its background.

Note that the framework has various options to switch on or off different processing steps. So, for example, you could save the RNL Rank filtered image stack, and re-run the framework on this image stack (deselecting the processing steps up to and including RNL ranked filtering).

To create a cone-catch image you must generate or load a [calibrated mspec image](#) (linear normalised reflectance stack) and [convert to cone catch using a model](#). The first slices in the stack must be the chromatic channels, with a final luminance slice in the stack. Here are examples of input image stacks for tetra-, tri- and di-chromatic modelling:



If the cone-catch model doesn't create a final luminance channel automatically (e.g. the "double" cones in birds), you will need to add your own. This will vary between visual system. There is a tool provided for [adding a luminance channel](#).

plugins > micaToolbox > Image Transform > Create Luminance Channel

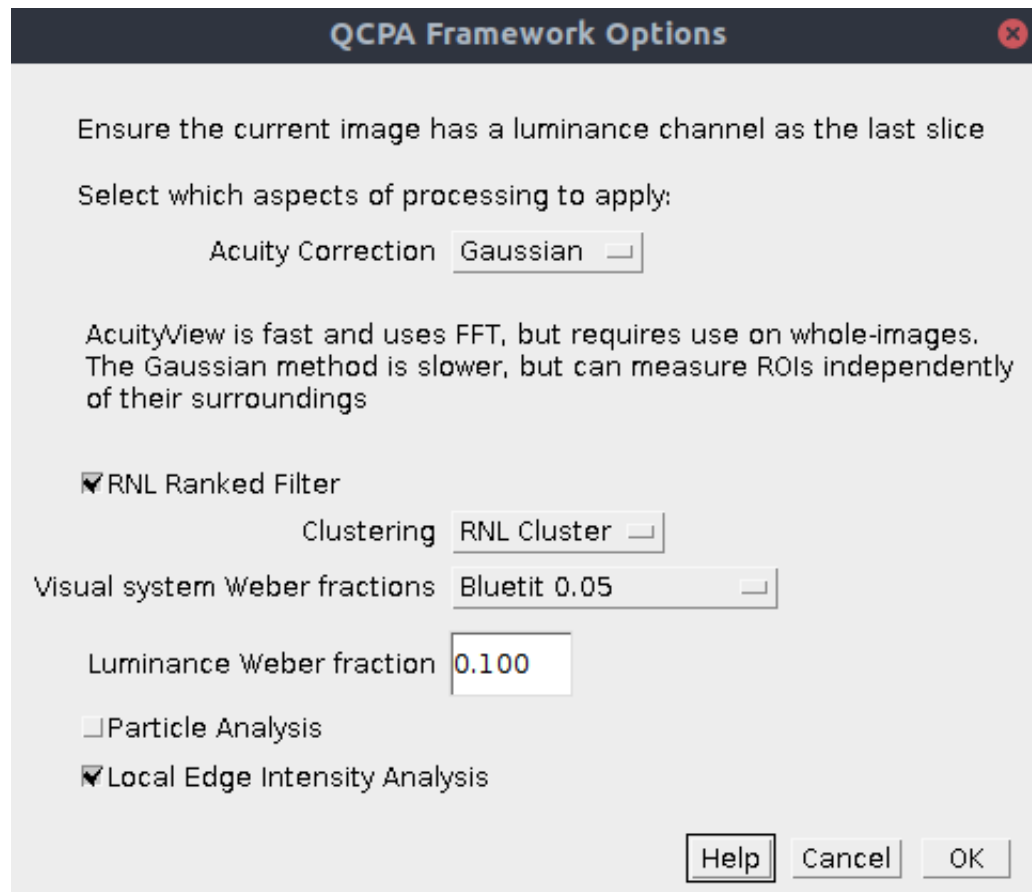
This tool adds a final luminance channel at the end of the cone-catch image stack, based on an equal average of all the selected channels.

Any pixel with zero values or negative values in any channel is ignored in RNL Ranked filtering and clustering (the RNL model can't deal with them, but this is also a method for screening out sections of the image which you do not want to include in clustering).

Running the QCPA Framework

Ensure the correct cone-catch image is selected and run:

plugins > micaToolbox > QCPA > Run QCPA Framework



The first dialog box lets you select which components of the framework you wish to run. Subsequent dialog boxes will ask for component-specific settings (see the page for each for relevant information).

Acuity Correction

AcuityView: Use the [AcuityView](#) method for controlling for spatial acuity. This uses FFT and a convolution in the frequency domain. As such, it can only be used on whole (rectangular) images. Any ROI will be smoothed with its surroundings.

Gaussian: Use the custom-written [Gaussian Acuity Control](#) method. This uses a non-separable Gaussian convolution which is able to apply the appropriate level of smoothing on irregularly shaped ROIs completely independently of their surrounding pixels.

Choose whether to apply the [RNL Ranked filter](#).

RNL Ranked Filter

Clustering

RNL Cluster: Use [RNL clustering](#) on the image. This breaks down the image into a discrete number of segments/clusters/patches based on visual models of colour and luminance discrimination. This is the default option when you don't know exactly how many patches there should be in your image.

Naive Bayes Cluster: This uses [naive Bayes clustering](#) to segment the image. Use this option if you know exactly how many patches there should be in your image. You should either have a results table listing the mean and standard deviation of each patch and each channel, or have a number of ROIs in the ROI manager which can be used to create these values automatically.

Visual system Weber fractions

You can specify the Weber fractions of each channel for the given visual system. In the absence of behavioural data on each of these values, the receptor class with the most abundant cones is generally given the base Weber fraction (common values used are 0.05 or 0.1), and the less abundant receptor classes are given higher values based on cone abundance ratios. See Vorobyev & Osorio (1998) for details. A number of visual systems have been included by default, or you can select "Custom" to specify your own. If you really have no idea which numbers to use, then set every channel to 0.1 (not recommended) and make sure you declare the numbers used in any publication.

Luminance Weber Fraction

Specify the luminance Weber fraction. This will often be around 0.1 or 0.2, but varies with species and requires behavioural validation. If in doubt, select 0.1.

Particle Analysis

This will run the [particle analysis](#) following clustering, providing data on the shape and abundance of patches of each cluster. Additionally it outputs each cluster as a new ROI so that each can be measured independently if desired (e.g. select the RNL Ranked filtered image, and use the ROIs to measure different image regions based on automated clustering).

Local Edge Intensity Analysis

This will run the [local edge intensity analysis](#), providing data similar to the boundary strength analysis, but on the un-clustered image.

Output

The output will vary depending on the processing steps chosen, however if the entire framework is used the output will be:

- **Spatial acuity-controlled cone-catch stack** (this looks like a blurred image, which will have been scaled to the desired number of pixels-per-minimum-resolvable-angle). Note the ROIs are scaled to match this image, and all subsequent image. The original ROIs (saved with the mspec image) are unaffected.
- **RNL Rank filtered cone-catch stack**, where edges have been reconstructed following spatial acuity modelling. Measuring this image is recommended for creating [colour maps](#). You can also make a colour [presentation image](#) of this image to show the results of spatial acuity modelling and edge reconstruction.
- **Cluster cone-catch average stack** (with the suffix "_Clustered"). This stack shows the average cluster value for the

selected range of recorded passes. Use this image for creating a colour "[presentation image](#)" of the clustered image.

- **Cluster ID image** (with the suffix "_Cluster_IDs"). This image shows which cluster each pixel belongs to. The numbers correspond to the clusterID numbers in the "Cluster Results" window.
- **Cluster Results table**. This outputs the summary statistics for each cluster (e.g. average cone-catch value, RNL XYZ coordinates etc...) Optionally also shows the adjacency matrix (the number pixels where each cluster is found adjacent to every other cluster, used for various analyses).
- **ROI Cluster Results table**. As above, however this breaks down the cluster results by ROI, showing only the clusters which are present in each ROI.
- **Summary Results table**. This output contains the QCPA results from a large number of analyses which require a clustered image (e.g. boundary colour strength analysis, adjacency analysis and visual contrast analysis). Note that this analysis is repeated for each ROI in the image.
- **Cluster Particle Analysis Summary Results table**. Details information on the average shape/size descriptors of particles in each cluster.
- **Individual Particle Results table**. Shows shape/size descriptors for each individual particle (each cluster will often be composed of a number of particles)
- **Local Edge Intensity Analysis images**. Two images are produced for the chromatic LEIA (suffix "_Col_LEIA") and luminance LEIA (suffix "_Lum_LEIA"). The false-colours show the angle of each edge, and the brightness shows the intensity of the edge. each image is a stack where each slice shows the results at different measured angles (vertical, horizontal and two diagonals).
- **Local Edge Intensity Analysis results table**. Shows the LEIA metrics.
- **Log**. The log shows exactly which steps were performed, and the user-controlled settings for each. Saving the log may be useful for remembering exactly which settings were used for a given analysis, and gives all the information required to reproduce your image analysis process which you should specify in any publication.

Summary Table Output Parameters

There are a large number of different numbers produced by the QCPA analysis, each describing different image properties. The table below gives a very brief description of each of these variables. For specific details and equations, please see the Appendix released with the QCPA publication. The default output shows the horizontal and vertical matrix analyses combined, however most of the variables can also be split into separate values by ticking the relevant box during processing (see above). If you are interested in comparing horizontal and vertical values separately (e.g. because pattern orientation is interesting in your study system), then make sure all of your images are rotated to a uniform orientation before analysis.

Abbreviation

CAA:Sc

CAA:Jc

CAA:St

CAA:Jt

Variable Name

Simpson colour diversity - Sc (eq. 1). This describes how evenly different colours in an image are represented, in terms of area coverage. The higher the number of different colours (number of discrete colour clusters), the higher the potential maximum value of Sc (up to the number of different colours).

Relative Simpson colour diversity - Jc (eq. 2). As above, but controlling for the number of different colours, so describes how evenly the colours are represented (in terms of area) independent of the number of colours.

Simpson transition diversity - St (eq. 3). This describes the regularity of transitions between different colours in an image. St is equal to the number of different colours when all transitions are equally frequent.

Relative Simpson transition diversity - Jt (eq. 4). As above,

	but controlling for the number of different colours, so describes how evenly the colours transition independent of the number of colours.
CAA:Hc	Shannon colour diversity - H_c (eq. 5). Also referred to as “entropy”, this is similar to Sc above, but uses a different diversity index.
CAA:Qc	Relative Shannon colour diversity - Q_c (eq. 6). As above, but controlling for the number of different colours, so describes how evenly the colours are represented (in terms of area) independent of the number of colours, with a maximum value of 1.
CAA:Ht	Shannon transition diversity - H_t (eq. 7). This describes the regularity of transitions between different colours in an image (using the Shannon rather than Simpson index). H_t is equal to the number of different colour clusters when all transitions are equally frequent.
CAA:Qt	Relative Shannon transition diversity - Q_t (eq. 8). As above, but controlling for the number of different colours, so describes how evenly the colours transition independent of the number of colours, with a maximum value of 1.
CAA:Scpl	Simpson colour pattern complexity $Scpl$ (eq. 9). This combines J_t and J_c (colour diversity and transition diversity respectively), such that colour patterns with uniform spacing and coverage (such as a chessboard) would have a maximum value of 1.
CAA:Qcpl	Shannon colour pattern complexity $Qcpl$ (eq. 10). As above, using Shannon instead of Simpson indices.
CAA:C	Pattern Complexity - C (eq. 11). This describes the geometric complexity of colour patterns based on the number of same-colour and different-colour transitions. Patterns with more complex structures will have higher values.
CAA:PT	Average patch size – PT . The average patch size of each colour based on horizontal and vertical “run lengths” of each colour (note that the particle analysis tool may be more informative).
CAA:PT HrZ	Average horizontal patch size - $PTHrZ$. The average “run length” (in pixels) of horizontal colour patches.
CAA:PT Vrt	Average vertical patch size - $PTVrt$. The average “run length” (in pixels) of vertical colour patches.
CAA:Asp	Aspect ratio – A . 0 implies horizontally elongated patterns, 0.5 indicates equal circular or random patterns, 1 indicates vertically elongated patterns.
VCA:ML	Weighted mean of pattern luminance contrast - ML (eq. 13) The mean luminance of the image (weighted by the area of each colour).
VCA:sL	Weighted standard deviation of pattern luminance contrast - sL (eq. 14). Standard deviation of the luminance of the image (weighted by the area of each colour). Higher values imply more variance in luminance values.
VCA:CVL	Weighted CoV of pattern luminance contrast - CVL (eq. 15) Coefficient of variation of luminance in the image (based on the above two values). Higher values imply more variance, while controlling for the mean value.
VCA:MDmax	Weighted mean of pattern D_{max} contrast - MD_{max} (eq. 16) The mean D_{max} contrast of the image (weighted by the area of

	each colour). Dmax is the most stimulated pairwise opponent channel (a colour channel which describes chromaticity, i.e. both hue and saturation) of each colour relative to grey (the achromatic point). Consider using the RNL saturation-based values instead for well characterised visual systems because it is a more perceptually uniform colour space.
VCA:sDmax	Weighted standard deviation of pattern Dmax contrast - sDmax (eq. 17). Standard deviation of the Dmax contrast of the image (weighted by the area of each colour). Higher values imply more variance in Dmax chromaticity values, and a more “colourful” image.
VCA:CVDmax	Weighted CoV of pattern Dmax contrast - CVDmax (eq. 18). Coefficient of variation of Dmax contrast in the image (based on the above two values). Higher values imply more variance, while controlling for the mean value.
VCA:MSsat	Weighted mean of pattern RNL saturation contrast - MSsat (eq. 19). The mean RNL saturation of the image (weighted by the area of each colour). RNL saturation is the distance (in receptor noise limited space, Vorobyev & Osorio 1998) of each colour from grey (the achromatic point). E.g. a fairly uniformly bright “red” image will have a high overall saturation, and a grey image will have a low value. Images with extremely varied colours (e.g. red, green and blue equally weighted) will also have a low mean, but high variance (see below).
VCA:sSSat	Weighted standard deviation of pattern RNL saturation contrast - sSSat (eq. 20). Standard deviation of the RNL saturation of the image (weighted by the area of each colour). Higher values imply more variance in RNL saturation values, and a more “colourful” image relative to grey. See also the VCA:MS value, which measures the internal differences between colours (i.e. colourful relative to each-other, instead of relative to grey).
VCA:CVSSat	Weighted CoV of pattern RNL saturation - CVSSat (eq. 21). Coefficient of variation of RNL saturation in the image (based on the above two values). Higher values imply more variance, while controlling for the mean value.
VCA:MSL	Weighted mean of RNL luminance pattern contrast - MSL (eq. 22). This is the average RNL luminance difference between all of the colours in the image, weighted by the area covered by each colour. Contrasts use the Siddiqi <i>et al.</i> 2004 method. Higher values imply larger luminance differences between colours in the image.
VCA:sSL	Weighted standard deviation of RNL luminance pattern contrast - sSL (eq. 23). The standard deviation of RNL luminance differences between all of the colours in the image, weighted by the area covered by each colour. Higher values imply larger variance in internal luminance differences, e.g. lots of similar and dissimilar luminance differences.
VCA:CVSL	Weighted CoV of RNL luminance pattern contrast - CVSL (eq. 24). Coefficient of variation of internal RNL luminance differences in the image (based on the above two values). Higher values imply more variance, while controlling for the mean value.
VCA>MS	Weighted mean of pattern RNL chromaticity contrast - M?S (eq. 25). This is the average colour difference between all

	<p>of the colours in the image, weighted by the area covered by each colour. Contrasts use the Vorobyev & Osorio 1998 Receptor Noise Limited method. Higher values imply larger internal colour differences between colours in the image. Therefore an image with lots of different colours will have a high value, irrespective of how far those colours are from the grey point. E.g. if the entire image is fairly uniformly bright “red” this value will be low, because all of the internal colours are similar. However, the saturation of the image (e.g. MSsat above) will be high.</p>
VCA:sS	<p>Weighted standard deviation of pattern RNL chromaticity contrast - s^2S (eq. 26). The standard deviation of RNL colour differences between all of the colours in the image, weighted by the area covered by each colour. Higher values imply larger variance in internal colour differences, e.g. lots of similar and dissimilar colours.</p>
VCA:CVS	<p>Weighted CoV of pattern RNL chromaticity contrast - CV^2S (eq. 27). Coefficient of variation of internal RNL colour differences in the image (based on the above two values). Higher values imply more variance, while controlling for the mean value.</p>
BSA:BML	<p>Weighted mean of luminance boundary strength - BML (eq. 28). The average luminance difference (based on Michelson contrasts) of boundaries between colours across the image, weighted by the relative frequency of transitions between each potential colour combination. Higher values imply higher luminance differences across colour boundaries.</p>
BSA:BsL	<p>Weighted standard deviation of luminance boundary strength - BsL (eq. 29). The standard deviation of luminance differences (based on Michelson contrasts) of boundaries between colours across the image, weighted by the relative frequency of transitions between each potential colour combination. Higher values imply higher variance in luminance differences across colour boundaries.</p>
BSA:BCVL	<p>Weighted CoV of luminance boundary strength - BCVL (eq. 30). Coefficient of variation of luminance differences of boundaries in the image (based on the above two values). Higher values imply more variance, while controlling for the mean value.</p>
BSA:BMDmax	<p>Weighted mean of Dmax boundary strength - BMDmax (eq. 31). The average Dmax chromaticity difference (based on Michelson contrast) of boundaries between colours across the image, weighted by the relative frequency of transitions between each potential colour combination. Higher values imply higher Dmax chromaticity differences across colour boundaries. Dmax is the most stimulated pairwise opponent channel (a colour channel which describes chromaticity, i.e. both hue and saturation) of each colour relative to grey (the achromatic point). Consider using the RNL saturation instead (below), which is closer to a perceptually uniform colour space.</p>
BSA:BsDmax	<p>Weighted standard deviation of Dmax boundary strength - BsDmax (eq. 32). The standard deviation of Dmax chromaticity differences of boundaries between colours across the image, weighted by the relative frequency of transitions</p>

BSA:BCVDmax	<p>between each potential colour combination. Higher values imply higher variance in Dmax differences across colour boundaries.</p> <p>Weighted CoV of Dmax boundary strength - BCVDmax (eq. 33). Coefficient of variation of Dmax chromaticity differences of boundaries in the image (based on the above two values). Higher values imply more variance, while controlling for the mean value.</p>
BSA:BMSsat	<p>Weighted mean of RNL saturation boundary strength - BMSSat (eq. 34). The average RNL saturation difference (i.e. the distance of each colour from the grey point, compared using Michelson contrast) of boundaries between colours across the image, weighted by the relative frequency of transitions between each potential colour combination. Higher values imply higher RNL saturation differences across colour boundaries.</p>
BSA:BsSsat	<p>Weighted standard deviation of RNL saturation boundary strength - BsSSat (eq. 35). The standard deviation of RNL saturation differences of boundaries between colours across the image, weighted by the relative frequency of transitions between each potential colour combination. Higher values imply higher variance in RNL saturation across colour boundaries.</p>
BSA:BCVSSat	<p>Weighted CoV of RNL saturation boundary strength - BCVSSat (eq. 36). Coefficient of variation of RNL saturation differences of boundaries in the image (based on the above two values). Higher values imply more variance, while controlling for the mean value.</p>
BSA:BMSL	<p>Weighted mean of RNL luminance boundary strength - BMSL (eq. 37). The average RNL luminance difference (using the Siddiqi <i>et al.</i> 2004 method) of boundaries between colours across the image, weighted by the relative frequency of transitions between each potential colour combination. Higher values imply higher luminance differences across colour boundaries. Consider using the “LEIA” method instead, which is similar but does not require a clustered image.</p>
BSA:BsSL	<p>Weighted standard deviation of RNL luminance boundary strength - BsSL (eq. 38). The standard deviation of RNL luminance differences of boundaries between colours across the image, weighted by the relative frequency of transitions between each potential colour combination. Higher values imply higher variance in luminance difference across colour boundaries.</p>
BSA:BCVSL	<p>Weighted CoV of RNL luminance boundary strength - BCVSL (eq. 39). Coefficient of variation of RNL luminance differences of boundaries in the image (based on the above two values). Higher values imply more variance, while controlling for the mean value.</p>
BSA:BMS	<p>Weighted mean of RNL chromaticity boundary strength - BM?S (eq. 40). The average RNL colour difference (based on Vorobyev & Osorio 1998) of boundaries between colours across the image, weighted by the relative frequency of transitions between each potential colour combination. Higher values imply higher colour differences across colour boundaries. Consider using the “LEIA” method instead, which</p>

BSA:BsS

is similar but does not require a clustered image.

Weighted standard deviation of RNL chromaticity boundary strength - Bs?S (eq. 41). The standard deviation of RNL colour differences of boundaries between colours across the image, weighted by the relative frequency of transitions between each potential colour combination. Higher values imply higher variance in colour difference across colour boundaries.

BSA:BCVS

Weighted CoV of RNL chromaticity boundary strength - BCV?S (eq. 42). Coefficient of variation of RNL colour differences of boundaries in the image (based on the above two values). Higher values imply more variance, while controlling for the mean value.

Citation & Reporting

If you use the QCPA framework you must cite our (forthcoming) paper. Note that you should also report the image analysis settings used in any publication. These are provided in the log window.

References:

Vorobyev, M., & Osorio, D. (1998). Receptor noise as a determinant of colour thresholds. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 265(1394), 351–358.

Siddiqi, A., Cronin, T. W., Loew, E. R., Vorobyev, M., & Summers, K. (2004). Interspecific and intraspecific views of color signals in the strawberry poison frog *Dendrobates pumilio*. *Journal of Experimental Biology*, 207(14), 2471–2485.

Scale Bar

<http://www.empiricalimaging.com/knowledge-base/scale-bar/>

[Pattern analysis](#) and [acuity control](#) require all images be scaled to a uniform number of pixels per unit length – a number that will vary with every study. Photographing different samples from slightly different distances can be controlled for with a scale bar, nevertheless we recommend taking photographs at uniform distances from the subject whenever possible. Particular care should be taken to photograph different treatments at the same distances for unbiased pattern analysis. Position the scale bar level with the sample and photograph from overhead if possible. If photographing from an angle rather than directly overhead a horizontal circular disk could be used as a scale bar (the disk's maximum width will always equal its diameter whatever angle it's viewed from) or place a straight scale bar/ruler side-on to the camera. Even if your initial hypothesis does not concern pattern, size or distance it is always good practice to include a scale bar.

Measuring the Scale Bar

Once you have [generated/loaded an mspec image](#), simply select the line-selection tool in ImageJ, draw a line along the scale bar in the image and press "S" on your keyboard. A dialog box will ask you to specify how long the selected length is. The units are arbitrary, however the same unit must be used throughout the entire image processing stream. e.g. if you use millimetres then when specifying a viewing distance for [acuity control](#) that distance must also be in millimetres. We recommend using millimetres for all scales. Once you have created the scale bar an ROI is created which stores the relevant information, then press "0" (zero) on your keyboard to save the scale bar ROI to the mspec image.

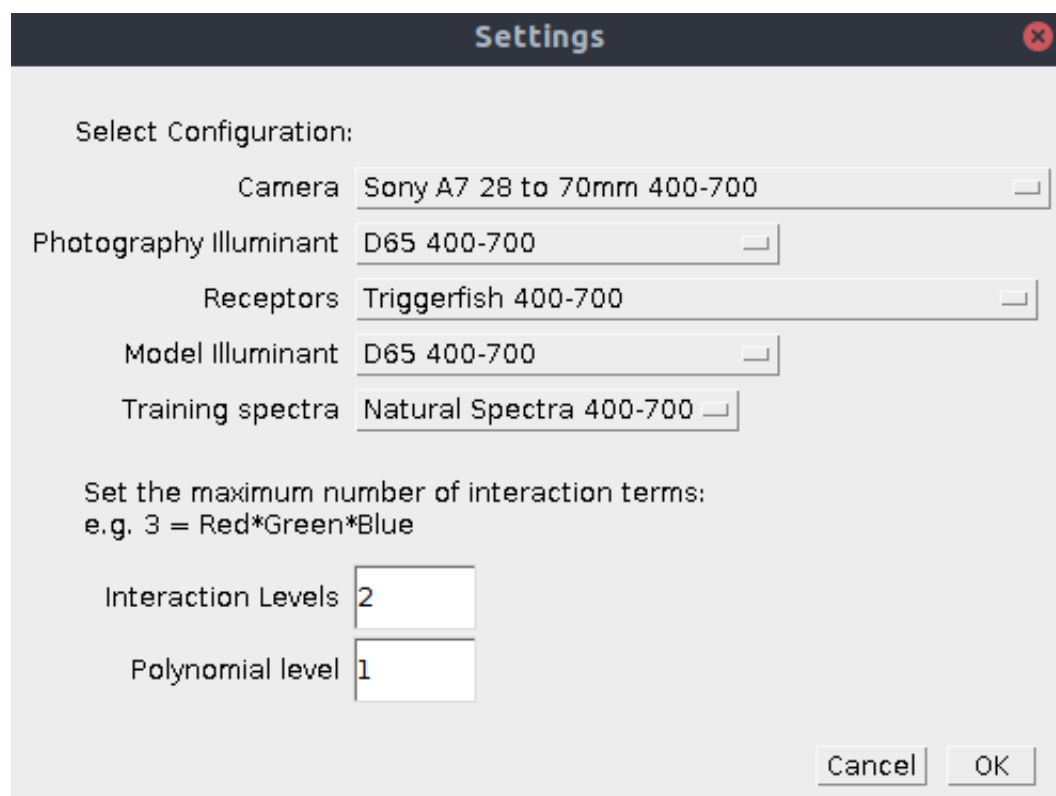
Spectral Sensitivity-Based Cone Catch Model

<http://www.empiricalimaging.com/knowledge-base/spectral-sensitivity-based-cone-catch-models/>

This is the standard old method (see [Troschianko & Stevens 2015](#)). It requires knowing the spectral sensitivities of the camera, which in practice are extremely difficult to measure. You will need a source of monochromated light (e.g. monochromator, or bandpass filters at regular 5 or 10nm increments across the camera's entire sensitivity range), plus a spectroradiometer. Sensitivities can then be calculated by combining measurements from a series of images at each spectral band. Please research this method yourself fully and ensure you have access to the relevant equipment before asking us any questions about how to do it.

We have measured the spectral sensitivities of a range of cameras, and if you use the exact same equipment you will be able to simply use these ready-made sensitivities.

1. Run *plugins > micaToolbox > Camera Calibration > Generate Cone Mapping Model from Spectral Sensitivities*.



Settings

Select Configuration:

Camera: Sony A7 28 to 70mm 400-700

Photography Illuminant: D65 400-700

Receptors: Triggerfish 400-700

Model Illuminant: D65 400-700

Training spectra: Natural Spectra 400-700

Set the maximum number of interaction terms:
e.g. 3 = Red*Green*Blue

Interaction Levels: 2

Polynomial level: 1

Cancel OK

2. Select the desired options, ensuring the wavelength range for each matches (e.g. 400-700nm or 300-700nm). The 'photography illuminant' is the illuminant you photographed the scene under (e.g. D65 if outside, or a known artificial light source in a dark room/museum setting), while the 'model illuminant' is the illuminant which images will be

converted to for simulating cone-catch quanta.

3. A cone-catch model will be generated and integrated into the toolbox

Ultraviolet Photography

<http://www.empiricalimaging.com/knowledge-base/ultraviolet-photography/>

Vision into the ultraviolet spectrum is thought to be the "norm" for tetrapods (e.g. amphibians, reptiles, birds), fish and insects. Even small mammals have shortwave sensitivities which stretch far deeper into the UV than human vision. Photography in the UV range is therefore essential for addressing many questions in visual ecology related to these species.

Ultraviolet photography presents a number of major difficulties:

1. Camera sensors are typically only sensitive to light in the human-visible range. So cameras need modifying to become sensitive to UV. See [full-spectrum cameras](#).
 2. Lenses designed for cameras typically do not transmit into the ultraviolet, and if they do they tend to have severe chromatic aberrations and focus shift. The options are to use expensive UV-VIS dedicated lenses, use existing lenses and accept poor-quality UV images, or acquire second-hand lenses which happen to have excellent UV transmission. See [UV lenses](#).
 3. Calibration is more tricky because the [grey standards](#) and colour charts used must extend into the UV range.
-

Underwater Photography

<http://www.empiricalimaging.com/knowledge-base/underwater-photography/>

The Light Enviroment

Water strongly attenuates long wavelenghts of light & UV (See video below, Fig 1). Depending on the visibility and the type of dissolved matter in the water, the ambient light quickly turns into a dim blue (Clear water), green (Algae) or brown (Tanins or mud).

[YouTube Video](#)

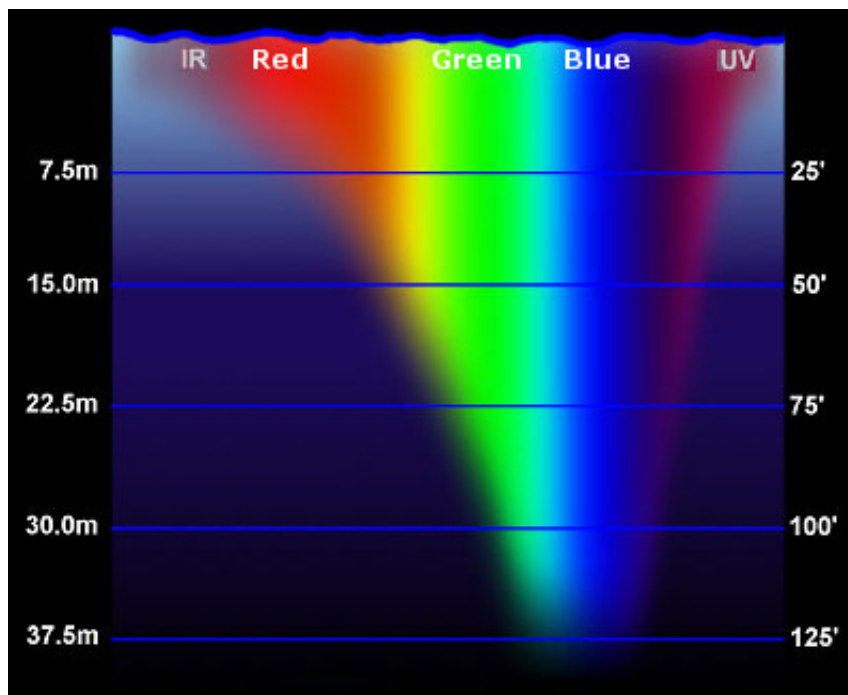


Figure 1: Light attenuation by depth in clear water. Source: Tom Morris, Fullerton College.

While it is crucial to understand that vision and colouration has adapted to a specific light environment, the available amount of light is often insufficient for high quality callibrated photography. As such, underwater photographers are often forced to spend large amounts of time and money on various lighting solutions. Choosing the most suitable lightling solution to your needs is crucial, and should focus on:

- Known illuminant spectra in the image
- Higher focal depth
- Faster shutter speeds

Calibrated digital photography (particularly in the context of image segmentation using luminance thresholds as introduced in QCPA) requires an even level of illumination across a scene or object. While in shallow water the problem of light attenuation is less dominant (Or inexistent), caustics (video below) and wave action make high quality photography and even illumination difficult to achieve. Therefore, it can be favourable to seek deeper (>5m) locations and try to solve the lighting issues there.

[Vimeo Video](#)

Another issue frequently encountered with underwater photography is backscatter. Backscatter refers to light that gets reflected back into the camera due to particulate matter in the water (Fig 2). This can pose severe limitations and often dictates the type of lens and lighting that can be used under given conditions. The key is to minimise the amount of water between the lens and the object in order to reduce the amount of particulate matter in front of the lens. Wide-angle lenses provide a suitable solution for larger subjects whereas macro-photography usually suffers less from this issue (Unless large amounts of background need to be captured, Fig. 3). Another way to minimize backscatter is to use indirect ways of illuminating a scene. There is a rich amount of literature and web-based sources on how to best achieve that.



Figure 2: A good example of backscatter. The distance of to the subject is approx. 2-3m. This picture was taken on Lizard Island in the far northern Great Barrier Reef which is a supposedly oligotrophic environment with comparably clear

water. Backscatter is an even greater issue in most nutrient rich waters.

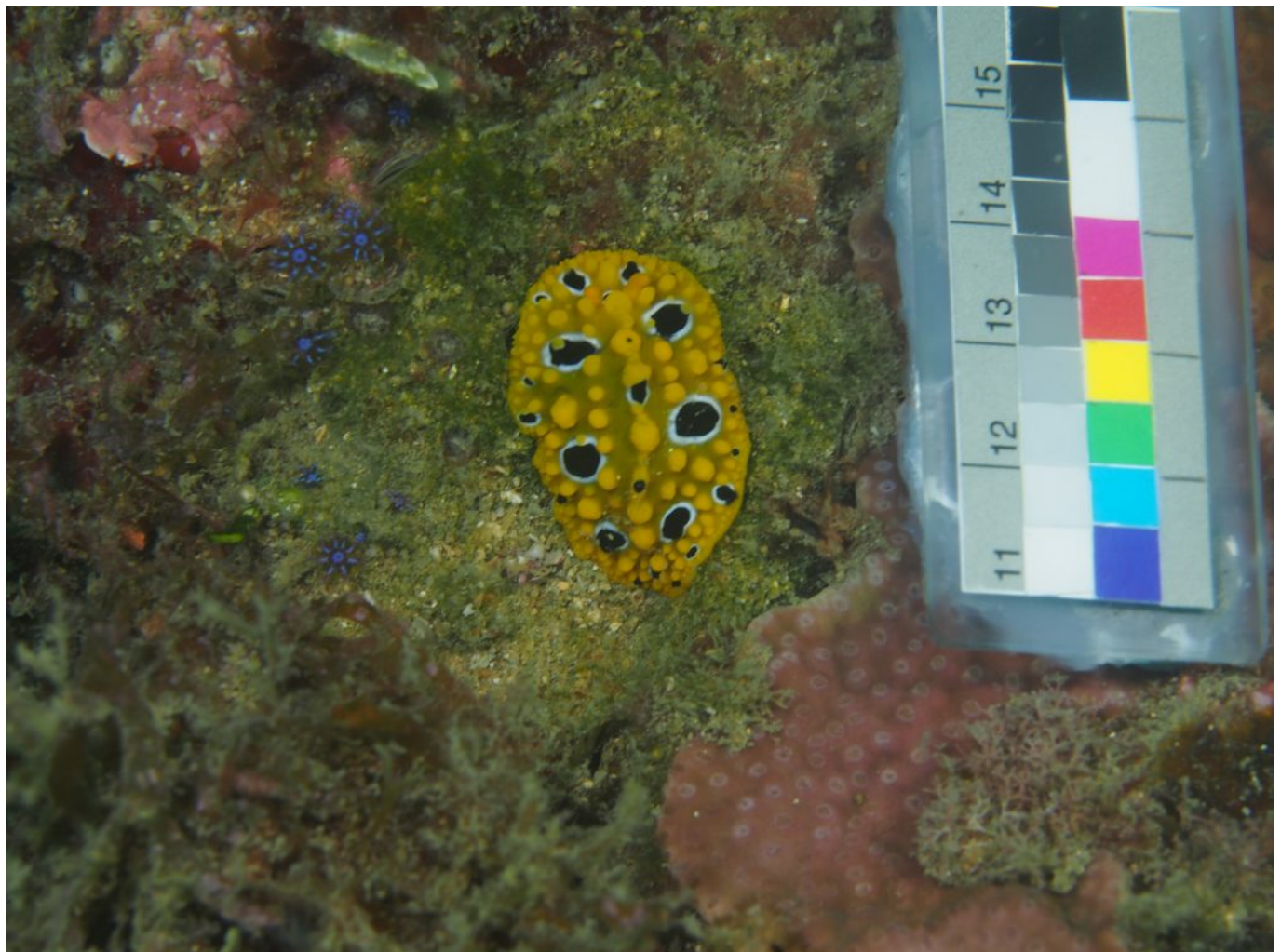


Figure 3: A picture of a nudibranch mollusc taken at 12m depth using a self-made ring light and custom made colour and size standard. The picture was taken from approx. 60cm distance using a 60mm macro lense and an Olympus PEN lite E-PL 5 camera.

Lighting solutions

There are four frequently used approaches to lighting in underwater photography:

- Natural light
- Flash lights (strobes)
- Hand-held or camera-mounted video lights
- Ring lights

Natural light:

Taking images using the naturally available light seems straight forward (But not necessary as MICA can simulate any given light environment). It is obviously the cheapest solution too! And yes, if none of the limitations discussed above stand in your way then, yes, try to take your images using natural light (Fig. 4). MICA and QCPA will work just fine. Also, if you want to correct your images to make them 'as if seen under white light' for artistic reasons, most camera sensors are good enough these days to capture the slightest remnants of long wavelenghts which can be amplified in post-production. However, the result will always be worse than that from a well illuminated image unless you are working really close to the surface. This can be an issue regarding visual modelling as you won't be able to simulate light environments etc... (e.g. Your image at a different depth) if you don't have the information captured by your camera in the first place.

positive:

- Cheap
- Less backscatter
- Less gear to carry around

negative:

- limited to shallow depths (max. 10-15m)
- unknown illuminant spectra (highly variable)
- Chance of losing information



Figure 4: A picture taken of a nudibranch mollusc at about 12m depth in coastal waters off the east coast of Australia without artificial lighting. Note the clearly visible green ambient light and the limited focal depth due to the low light levels as well as extensive shadows in deeper sections.

Flash Lights (Strobes):

Strobes are probably the most commonly used source of light in underwater photography. They provide brief and powerful burst of light and last for an entire day of shooting. However, it is exactly those properties which makes them tricky to use for visual modelling. Because they provide a strong focal source of light they are likely to cast shadows in the image (Fig 5). Shadows which wouldn't be there naturally and cannot be corrected for in post-production. While these shadows are desired for artistic photography, they are highly unfavourable for visual modelling. Also, because they are not constantly turned on it is hard to gauge the resulting lighting in an image prior to taking the shot. Therefore, taking evenly illuminated images using strobes is the way for well-trained underwater photographers. Using multiple strobes and diffusion techniques it is indeed possible to repeatedly achieve evenly illuminated pictures without artificial shadows. Despite these drawbacks, the amount of light produced by a set of powerful strobes is often the only way to achieve enough illumination to capture larger objects and scenes.

positive:

- Very powerful
- Long battery life
- Relatively cheap (\$200-\$600 AUD per strobe)
- Good for light sensitive animals (Ambush photography)

negative:

- Difficult to anticipate the resulting image
- Requires high level of skill



Figure 5: How no to: An artistic picture of a Nudibranch mollusc illuminated by a strobe.

Hand-held & Camera-mounted video lights:

Initially designed for videography and functional purposes such as orientation in dim-light, overhead or nighttime conditions, video lights have become a valuable solution for underwater photography. As they provide a constant source of light it is possible to anticipate the evenness of a shot prior to taking an image which makes the process a lot less time consuming, predictable and repeatable (Video below). However, to achieve remotely comparable light intensities to strobes it is necessary to invest substantially into powerful light sources. Recent developments in LED and battery technology have seen the power and battery life of video lights sky rocket while prices have come down significantly. To achieve similar levels of light to a medium range strobe we advise to seek at least 20,000-30,000 lumen of output (And more!). Given that recently marketed video lights have outputs of up to 20,000 lumen, this has become feasible. For any object larger than 30cm that needs to be captured with an adequate amount of background we recommend going well beyond 40,000 lumen. This will set you back approx. \$0.1 AUD / Lumen with current prices and technology (2019), making a 40,000 lumen setup cost approx. \$4,000 AUD. Using handheld video lights is not recommended due to the interference with camera operation and the usually low output of handheld video lights.

Positive:

- Constant light source
- Can do video & photo
- Often have UV and red light
- Easy for less trained photographers

Negative:

- Heavy
- Expensive

[Vimeo Video](#)

Ring lights:

Even illumination can be achieved through various diffusion techniques and the use of multiple light sources at the same time (e.g. as shown above). An well-known (Although not common) alternative approach is the use of ring lights. These can come in three different types: Powered by external video lights (Fig 6), powered by external strobes and powered by built-in LED arrays (Fig. 7). Ring lights are a comparably recent thing for commercially available lighting solutions, particularly the ones with built-in LED arrays. They are usually designed for capturing really small objects from very close up (max. 10 cm). We have successfully managed to increase that distance to up to 1m using custom made video light powered systems. However, these are still highly inefficient and have huge potential for future engineering efforts. Commercially available solutions are either limited to being powered by strobes or built-in LED arrays. However, the latter are currently still too weak for the capture of significant proportions of visual background or larger subjects. However, this can be mitigated by choosing different types of lenses. A wide angle lens will help to reduce the distance between camera and subject but might introduce image distortion which will have to be corrected.

Positive:

- Extremely even illumination

Negative:

- Can only be used on very small objects
- Can be difficult to fit to a specific camera setup



Figure 6: The 'Medusa II', a 37'000 Lumen video light powered ring light and an adjustable optic fibre bundle array.



Figure 7: A 3,000 lumen ring light with built in LED array

UV Lenses

<http://www.empiricalimaging.com/knowledge-base/uv-lenses/>

Photography lenses are typically designed for operating in the human-visible ~420-680nm range. Within this range the lenses have good transmission, and control for chromatic aberrations (e.g. the red, green and blue channels are all in focus).

UV photography ideally requires a lens which has transmission down to about 300 or 320nm (deeper than this the sun's emissions are so low it is unlikely to be ecologically relevant to almost any animal). It should also hold focus between the UV and visible range, and control for other types of aberration. These place extreme difficulties on lens manufacture. Few types of optical glass transmit in the UV range well (e.g. fused silica and calcium fluoride), but these glass types tend to have very similar refractive indices and Abbe numbers, meaning it's difficult to control for various aberrations. Add to this the requirement to hold focus over an even larger spectral range than a standard lens and things get very tricky.

Recommended Lenses (empirical imaging has no affiliation with these suppliers):

[Jenoptik/CoastalOpt 105mm and 60mm UV-VIS-IR](#)

These are expensive, but fairly good UV lenses. Under some photography conditions they suffer from internal reflections and "hot-spots", which must be eliminated before a photo can be used.

[Nikon UV-105](#)

Company7 is selling these lenses. We have not been able to test them ourselves, but they have an excellent reputation in the UV community.

[Nikkor EL 80mm](#)

This is a second-hand lens, produced in the 1980s for dark room photo enlargement. Note that only the older, metal-body version works well (this has chrome detailing and no plastic on show). This lens typically goes for about £60-90 on ebay. It happens to have excellent transmission down to about 320nm, and has almost no focus shift. Being an enlarger lens it has no built-in focussing function. To use this lens on a standard camera you will need: an M42 adapter, an M42-M42 focussing helicoid with a range from 25-55mm (or a bellows), and an M42-M39 adapter ring. Nikons have a deeper back-focal-length than other systems, so a shorter helicoid is required for a Nikon camera to focus on infinity. In total these should cost about £50, and are readily available on

Amazon/eBay/AliExpress.

Novoflex Noflexar 35mm

This lens was produced in West Germany in the 60s-70s, and comes up on eBay occasionally. It has good UV transmission and holds focus well enough that no focus adjustment is necessary between visible and UV photos. It is the widest-angle lens suitable for UV photography that we are aware of. This also makes it the best option for macro photography (the lens can be reverse-mounted for >1:1 enlargement of small samples). The lens is available with different mounts, the recommended mount is the M42, which only needs a readily available adapter for use on almost all modern cameras (though

Standard Lenses

Nikon DSLRs will not focus to infinity)

Modern "standard" photography lenses can be used for UV photography, however they tend to suffer focus shift, and have poor transmission (typically not much transmission below 360nm, which is not ideal for many animals which have peak UV sensitivity in this region). As such they are not recommended.

UV-VIS filters

<http://www.empiricalimaging.com/knowledge-base/uv-vis-filters/>

UV photography requires a UV-pass filter (for UV-only photography) and a visible-spectrum-pass filter (also called a UV/IR cut or blocking filter). Photographs are taken with each of these filters using a [full-spectrum camera](#).

Available filters (empirical imaging has no affiliation with these companies):

[Baader Venus-U filter](#)

This is the standard filter which has been used for many years. It has excellent IR blocking, and the transmission is in a suitable range. It can also be easily paired with the Baader UV/IR blocking filter for photography in the visible region. One downside is that the thread sizes are standard for astronomical instruments, but not cameras.

[Andrea "U"](#)

The specifications of this filter are excellent, and it uses a more standard thread size than the Baader filters.

[Kolari Vision UV Photography Filter](#)

The specification of this filter sounds ok (though with fairly low maximum transmission of 50% at 365nm), and decent sounding IR suppression. But they haven't published a full transmission profile. We haven't tested it though. The colour-correcting UV/IR cut filter from the same supplier sounds ideal for getting a more sensible colour in the visible-range than the above Baader UV/IR cut filter.

Visual Contrast Analysis

<http://www.empiricalimaging.com/knowledge-base/visual-contrast-analysis/>

The 'Visual Contrast Analysis' is an umbrella term for a variety of pattern parameters which were first conceptualised by John Endler in the early 90s (Endler 1990, 1991, 1992; Endler & Mielke 2005). These parameters seek to combine both chromatic and achromatic properties of colour pattern elements with their spatial properties, specifically their relative abundance in a given colour pattern to describe the visual contrast (Saliency, vividness, detectability, memorability) a colour pattern might provide to an observer. A range of these parameters have been tested on biological systems such as guppies (Endler & Houde 1995) with many more awaiting testing.

We have taken every possible parameter which has ever been formulated as well as expanding on the existing selection ([See here](#)). However, we have systematically excluded angular chromaticity metrics and replaced them with metrics derived from the log-transformed Receptor Noise Limited (RNL) colour space. For detailed information and equations please see the suppl. Material.

References:

Endler, J.A. & Houde, A.E. 1995. Geographic Variation in Female Preferences for Male Traits in *Poecilia reticulata*. *Evolution (N. Y.)* **49**: 456–468.

Endler, J.A. 1990. On the measurement and classification of colour in studies of animal colour patterns. *Biol. J. Linn. Soc.* **41**: 315–332.

Endler, J.A. 1992. Signals, signal conditions, and the direction of evolution. *Am. Nat.* **139**.

Endler, J.A. 1991. Variation in the appearance of guppy color patterns to guppies and their predators under different visual conditions. *Vision Res.* **31**: 587–608.

Endler, J.A. & Houde, A.E. 1995. Geographic Variation in Female Preferences for Male Traits in *Poecilia reticulata*. *Evolution (N. Y.)* **49**: 456–468.

Endler, J.A. &

Mielke, P.P.W. 2005. Comparing entire colour patterns as birds see them. *Biol. J. Linn. Soc.* **86**: 405–431.

PDF generated April 11, 2019 at 8:49 AM