

# Contents

[Contents 1](#_Toc63720808)

[Introduction 1](#_Toc63720809)

[What is CamoEvo? 2](#_Toc63720810)

[How to Run a Game 2](#_Toc63720811)

[Installation 2](#_Toc63720812)

[Collecting Images 2](#_Toc63720813)

[Starting a Game. 3](#_Toc63720814)

[Playing the game 4](#_Toc63720815)

[Obtaining Data 4](#_Toc63720816)

[How Targets are Generated 5](#_Toc63720817)

[Animal Maculation (Reaction-Diffusion Method) 6](#_Toc63720818)

[Egg Maculation (Gaussian Method) 8](#_Toc63720819)

[How Slides are Generated 10](#_Toc63720820)

[How to run your own Image Analyses 11](#_Toc63720821)

[Additional CamoEvo features 11](#_Toc63720822)

[Phenotype Plotter 11](#_Toc63720823)

[Evolution gif 11](#_Toc63720824)

[How to Customise CamoEvo 11](#_Toc63720825)

[Changing the GA Settings 11](#_Toc63720826)

[Creating Custom Target Shapes 12](#_Toc63720827)

[File and Code Organisation 13](#_Toc63720828)

[1 CamoEvo: 13](#_Toc63720829)

[2 CamoReq: 13](#_Toc63720830)

[References 14](#_Toc63720831)

# 

# Introduction

Here we describe the features of the camouflage evolution game CamoEvo, how the game functions and how it can be used and modified for research. For examples of teaching practicals using CamoEvo see the Biological Sciences higher biology and Undergraduate practical sheets. For details regarding its genetic algorithm, refer to the ImageGA\_Supplement*.* These files can be found on [GitHub.](https://github.com/GeorgeHancock471)

# What is CamoEvo?

Just as evolution by natural selection provides a mechanism with which organisms adapt to their environment so too does evolution provide a tool for optimising solutions for problems with large parameter spaces. In computer science, genetic algorithms (GAs) are a heuristic global optimisation tool which mimic natural selection to solve complex problems (Goldberg and Holland, 1988). CamoEvo combines the game-like design of common psychophysics experiments, the search mechanisms of modern genetic algorithms and the image analysis features of the MICA toolbox and ImageJ to demonstrate the influence of natural selection on camouflage (Schneider et al., 2012, Bonney et al., 2014, Troscianko et al., 2017, Niu et al., 2018). The intention of this program is to be used as a research and teaching tool both in the lab and at home, and as a method for designing future camouflage optimisation experiments. CamoEvo outputs key camouflage statistics, graphs, and the stimuli from every round of the game should the user wish to run their own analyses (Troscianko and Stevens, 2015, Troscianko et al., 2017).

# How to Run a Game

## Installation

CamoEvo can be downloaded from either from  [GitHub](https://github.com/GeorgeHancock471?tab=repositories&q=&type=public&language=) or our [Website.](https://www.visual-ecology.com/) When downloading be sure to use the link specified for each operating system (Linux, Mac, and Windows). If you download CamoEvo in this way it should consist of ImageJ bundled with JAVA from NIH as well as CamoEvo, ImageJ and the MICA toolbox (Troscianko and Stevens, 2015).

## Collecting Images

To play CamoEvo you will need to obtain a collection of images which you will evolve your camouflage population on. These images can be taken from anywhere be it a garden, out in the field or online (Figure1). Ideally you should use a number of photos equal to the number of individuals within your population; by default, this is 24. These images should be uploaded to the device which you installed CamoEvo to and placed within a unique folder solely containing these photos.



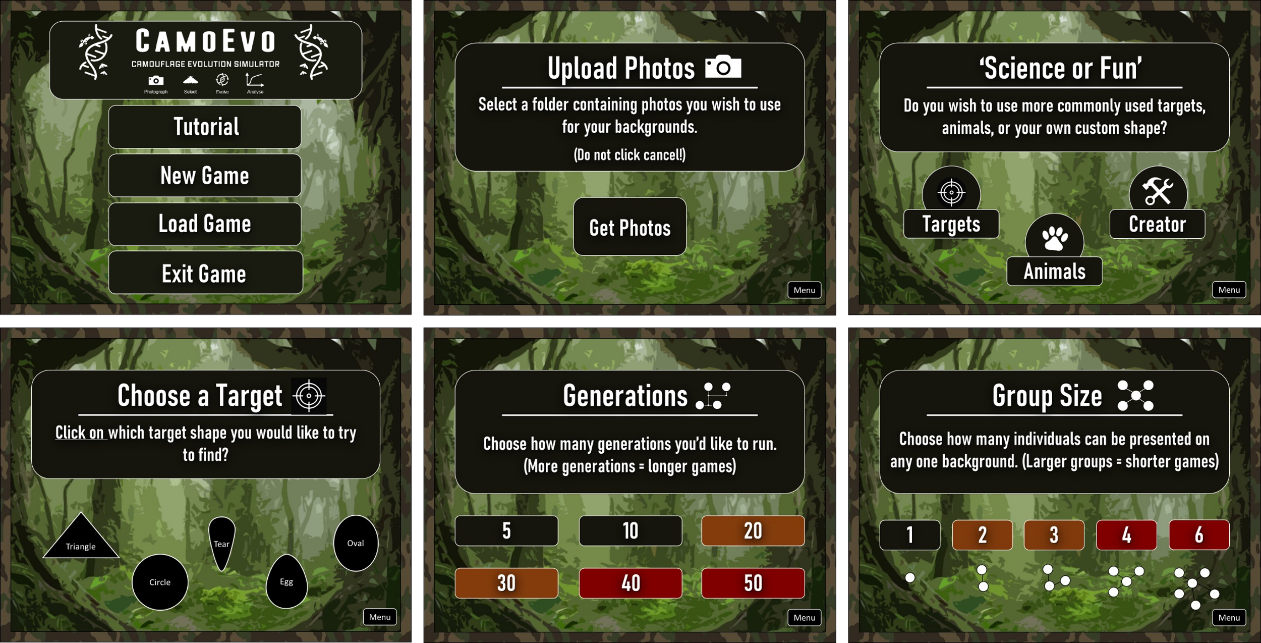
***Figure 1****. Example habitat photos taken from various locations at equal distance: A) estuary gravel, B) estuary pebbles, C) leaf litter and D) wet grass. Photos A) and B) were taken under direct lighting (cloudless day) and have strong shadows.*

## Starting a Game.

Load up CamoEvo from ImageJ. The CamoEvo plugin should start automatically when launching ImageJ with the version downloaded with CamoEvo pre-packaged. However, should it not, it can be loaded from *plugins/1 CamoEvo/CamoEvo Game*. You can start a new game by clicking new game or you can load an existing game by clicking load game (Figure 2).

When starting a new game, you will need to upload photos by selecting the get photos button and choosing the folder containing the photo gallery you wish to use for your backgrounds. Once you have selected a gallery, you can choose the type of targets you wish to use. You can use targets designed for camouflage experiments (targets), animal shapes (animals) or create your own custom shape using the creator. Different shapes use different pattern generation systems. The targets (triangle, circle, tear, oval) use reaction-diffusion. The animals use the same, but with bilateral symmetry and the egg targets use a unique pattern generation system based on egg maculation theory.

You will then be able to choose, if you wish, from several advanced settings. These include the number of generations (how long the game will be played for), the population size (the number of individuals per generation), group size (the number of individuals shown at a time) and time out time (how much time is given to find the targets). Once you choose the settings you desire, you must give your population a unique name before continuing to play.



***Figure 2****. Example UI from CamoEvo for setting up the game. (Top Left) CamoEvo main menu, (Top Centre) photo selection, (Top Right) target category. (Bottom Left) target choice, (Bottom Centre) number of generations, (Bottom Right) number of individuals in the population.*

## Playing the game

Once a population has been setup CamoEvo will ‘evolve’ the first generation. This will typically take 30 to 60 seconds; you will then be given the option to continue to play or exit back to the main menu. The game tasks you with searching for each individual in the population as fast as possible. Between each generation you will be given the option to exit and come back later.

Once you click to play you will be asked to move your cursor to the centre of the screen and hold it there. The circle at the centre of the screen will gradually turn white and once completely white you will be presented with a slide featuring one of the backgrounds you uploaded, and a number of targets equal to the group size selected. You will then have to click on all the targets as fast as you can, only moving your cursor from the centre when you spot a target. If you fail to spot the target in time the game will move onto the next screen regardless. Once you have done this for all individuals, the next generation will evolve.

When the final generation is complete you can either run an analysis on the completed generations or click to continue playing. If you click to continue playing the game will run for an additional 5 generations before giving you the option to analyse them again. All camouflage measures are done post game to reduce the load time between each generation.

## Obtaining Data

Once all the generations have been completed you will be presented with the option to run an analysis or to keep playing for 5 more generations. If you click analyse, it will take 1-2 minutes per generation for CamoEvo to go back through each slide of the game and measure the luminance (CIE LAB Δμ luminance), contrast (CIE LAB ΔσL) and colour (CIE LAB Δμ A and B) difference between the target and the local background (McCamy, 1992, Troscianko and Stevens, 2015, Troscianko et al., 2016, Troscianko et al., 2017). Where the local background is a sub sample of the background within a circle of a diameter equal to 2x the diameter of the target. In addition, the level of edge-disruption (CIE LAB GabRatL) of the target against the background is also recorded.



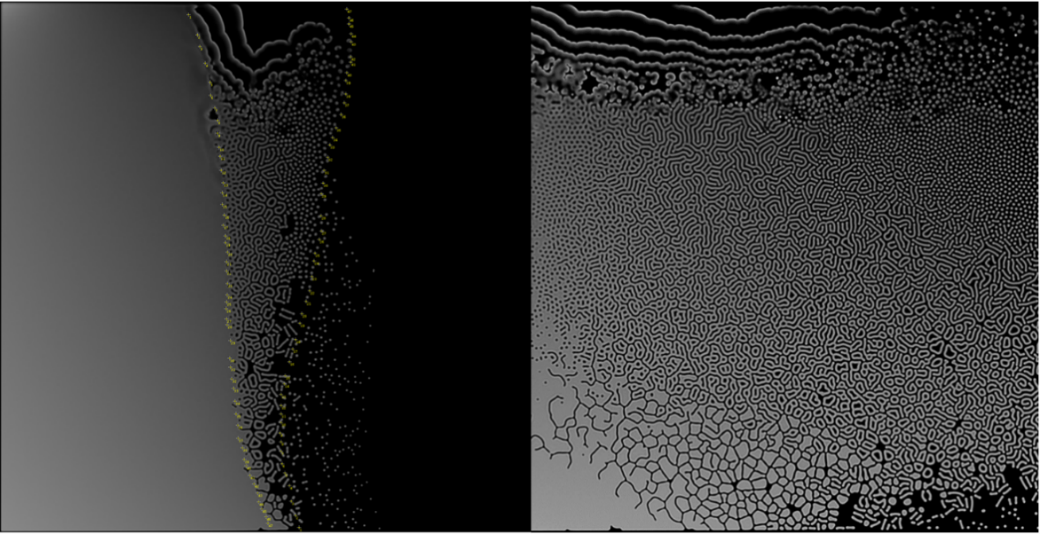
***Figure 3****. Example analysis output from a population of toad targets on boggy grass after 5 generations. (Left) the capture time for each generation and (right) the camouflage statistics with the contrast match for each generation on display.*

Once analysis is complete a results menu with the top 9 individuals from the most recent generation and plots for capture time, luminance match, contrast match, colour match and GABRAT will become available (Figure 3). The plots display the mean and standard error for the population at each generation. These measures can then be extracted as a .txt file for analysis in R or another similar programs. Within the output file there are three types of capture time recordings present. (Capture Time) the time taken for the target to be clicked on, (Reaction Time) the time taken for the cursor to be moved from the centre of the screen, and (Survival Time) which is equal to Reaction Time if , else it is equal to Capture Time. Along with the camouflage recordings the target ID, generation number, background image ID, target X coordinate, target Y coordinate, target rotation and target flip are all recorded and the genome for each individual within the population is also concatenated to the output .txt file. This allows users to run analyses for changes in camouflage phenotype and genotype with generation, the influence of generation on capture time and to account for random factors such as background image and target location. The µL, µA, µB and õL values for the local background and the targets are also output.

# How Targets are Generated

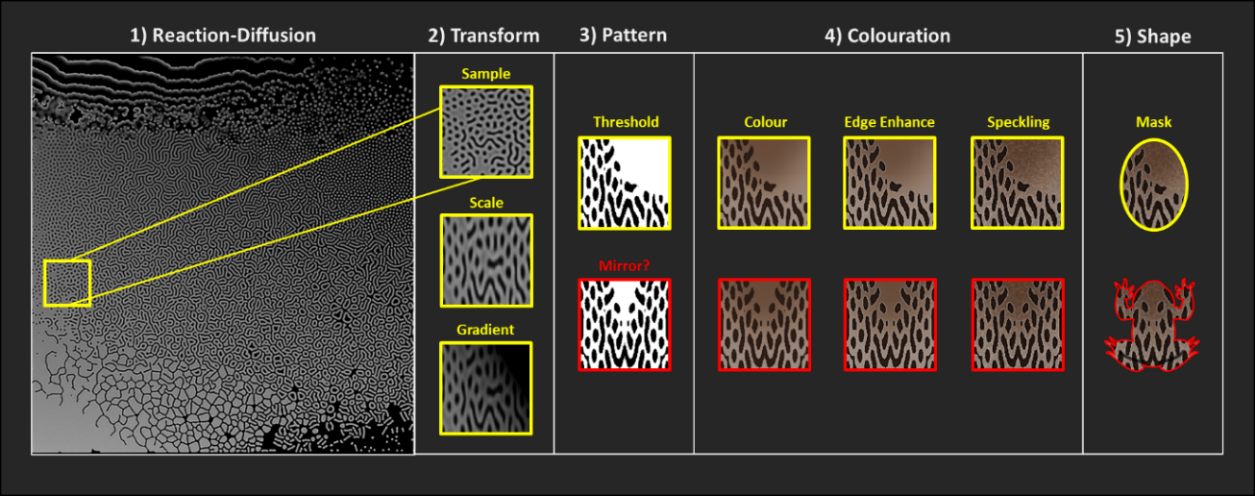
## Animal Maculation (Reaction-Diffusion Method)

For all target types barring the egg targets (see below), target maculation is generated using Gray-Scott reaction diffusion (Allen et al., 2010, Kondo and Miura, 2010). Given that most of the reaction-diffusion space (F and K values) result in blank black or grey spaces, our system uses a non-linear selection of the space. We did this by manually selecting (subjectively) the bounds of interesting patterns, then creating a model (4th degree polynomial in this case) to work out K-min and k-max from F. From these values a 2D pattern plot is generated and used to provide the gamut of reaction-diffusion patterns used by CamoEvo (Figure 4). New plots can be generated using " \CamoReq\Reaction\_Diffusion\_Pattern\_Generator\_gradient\_smlap\_adaptivek.java".



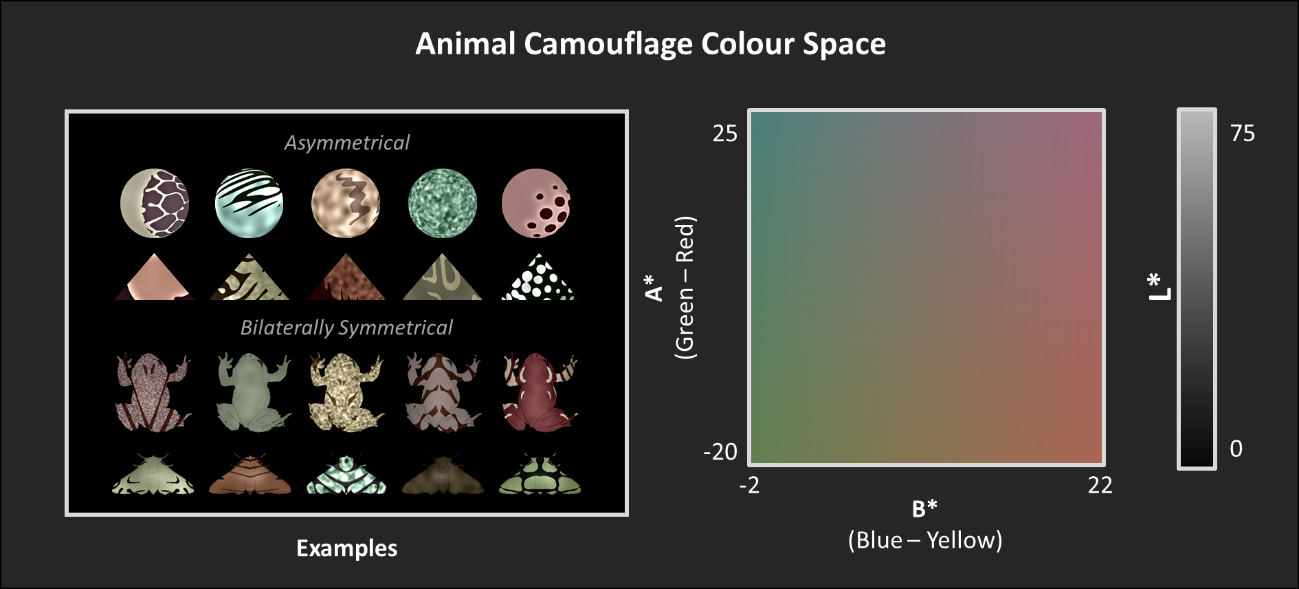
***Figure 4****. Left: the generated Gray-Scott reaction-diffusion space and the polynomial selected points shown in yellow. Right: the output 2D pattern space for the polynomial K and linear F.*

It is from this pattern plot that a region is sampled using the genome and then rotated and scaled to produce additional maculation patterns. For example, increasing the aspect ratio converts spots into stripes (Figure 5). Using a pattern sheet, in this way, saves time as the lengthy reaction-diffusion pattern generation does not need to run for each individual and ensures that so long as the genotype for maculation is identical the pattern of maculation is perfectly heritable. If the target shape used is one of the animal targets, the pattern is mirrored bilaterally on the centre line of the animal’s shape.



***Figure 5****. Schematic of the animal pattern generation system used by CamoEvo. A section of the reaction-diffusion gamut is selected, scaled and shaded with a gradient before being thresholded. Depending on what target shape is selected the pattern is mirrored bilaterally. Colouration is then applied to both the background and the pattern in addition to edge enhancement and speckling. All these features are regulated by different genes.*

In addition to maculation, factors such as edge-enhancement, speckling, countershading, and colour in CIE LAB space are regulated by additional genes to produce the final target phenotype (McCamy, 1992). The full list of genes is given in table 1. The CIE LAB space is restricted to prevent highly saturated colours and blues (Figure 6). The L\*A\*B\* genes for the pattern, top and bottom of the image are also linked using ImageGAs targeted duplication system, allowing LAB genes to hop between maculation and the targets background (for more details see the ImageGA handbook).



***Figure 6****. Animal camouflage CIE LAB colour gamut.* *The animal colouration is a 3D space with genes regulating the level of luminance (L\*), green-red (A\*) and blue-yellow (B\*)*

***Table 1****. The pattern generation genes for the reaction-diffusion pattern generation.*

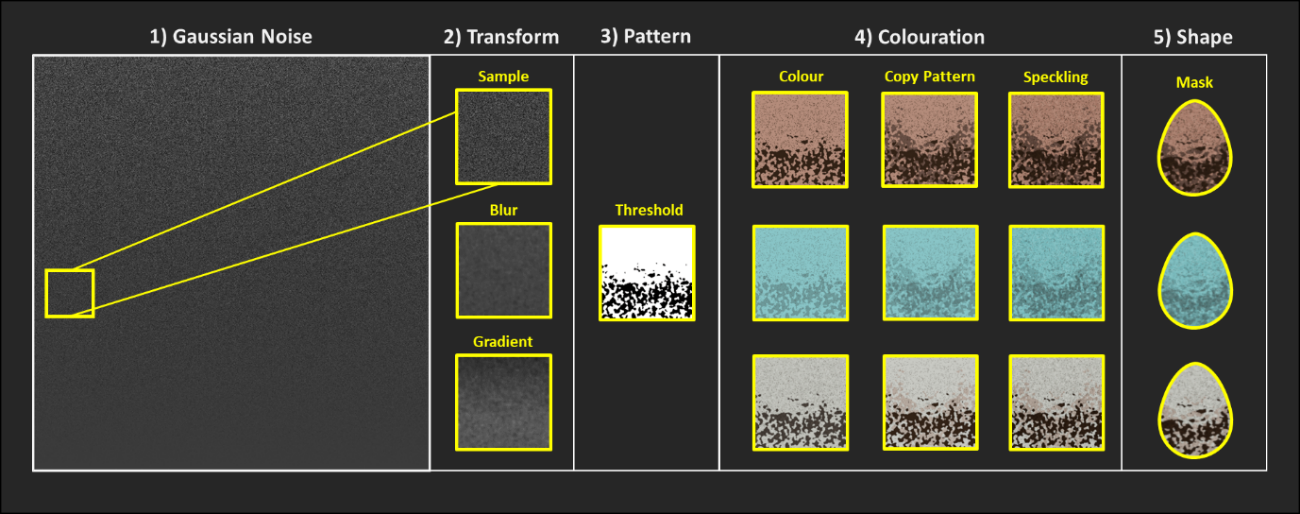
|  |  |
| --- | --- |
| Gene Name | Function |
| speckling\_intensity | Speckling noise (StdDev) |
| speckling\_sigma | Speckling sigma (level of Gaussian blurring) |
| pttn\_x | X coordinate that the pattern is sampled from. |
| pttn\_y  pttn\_width | Y coordinate that the pattern is sampled from.  Scale of the pattern (changes pattern size). |
| pttn\_aspect\_ratio | Aspect ratio (stretches patterns). |
| pttn\_angle | Pattern angle (pattern rotation). |
| pttn\_coverage | Pattern threshold (percentage of image covered by pattern). |
| pttn\_CS\_sigma | Pattern shading sigma (level of blurring). |
| pttn\_CS\_height | Pattern shading height/radius. |
| pttn\_CS\_angle | Pattern shading angle. |
| ee\_intensity | Level of edge enhancement. |
| ee\_sigma | Edge enhancement sigma (level of blurring). |
| unit\_C\_0 | Unit 0 for category C (colour) for targeted duplication. |
| pttn\_L | Pattern colour luminance value (CIE LAB). |
| pttn\_A | Pattern colour A channel value (CIE LAB, red-green). |
| pttn\_B | Pattern colour B channel value (CIE LAB, blue-yellow). |
| unit\_C\_1 | Unit 1 for category C (colour) for targeted duplication. |
| top\_L | Top colour luminance value (CIE LAB). |
| top\_A | Top colour A channel value (CIE LAB, red-green). |
| top\_B | Top colour B channel value (CIE LAB, blue-yellow). |
| unit\_C\_2 | Unit 2 for category C (colour) for targeted duplication. |
| bottom\_L | Bottom colour luminance value (CIE LAB). |
| bottom\_A | Bottom colour A channel value (CIE LAB, red-green). |
| bottom\_B | Bottom colour B channel value (CIE LAB, blue-yellow). |
| unit\_C\_N | End marker for the category C (colour). |
| col\_sigma | Sigma (blurring) between the top and bottom colour. |
| col\_height | Top colour height (where the bottom colour starts, and the top colour stops). |

## Egg Maculation (Gaussian Method)

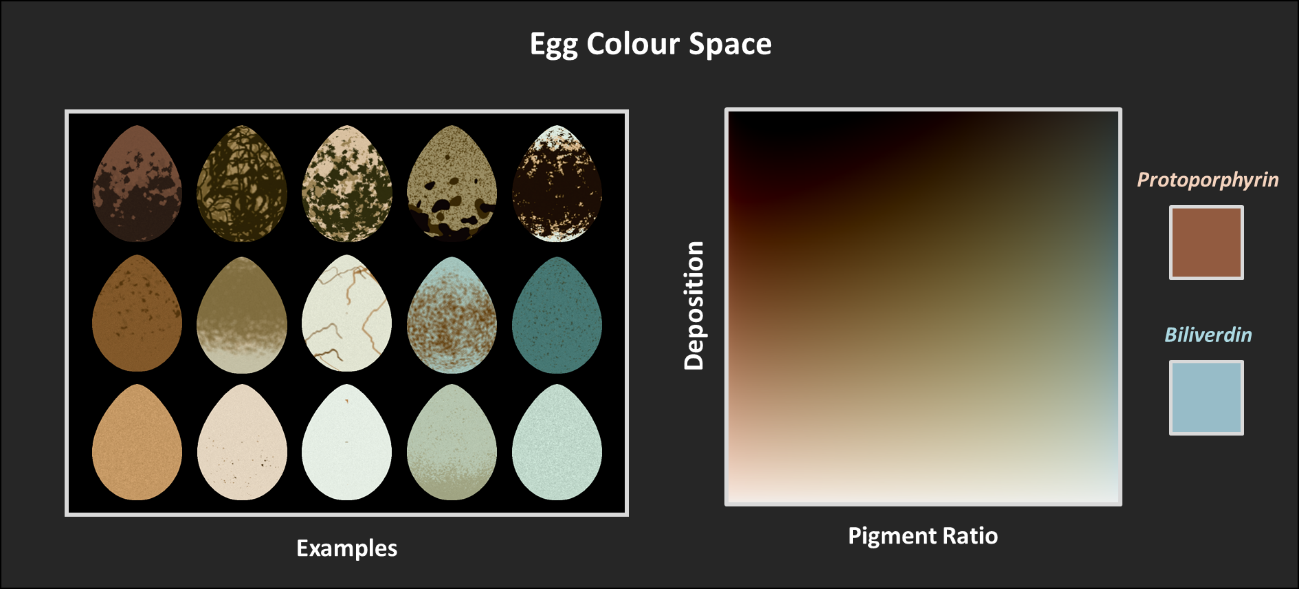
Unlike the other targets, eggs use their own distinct pattern generation system based on egg maculation and pigment colour theory (Pike, 2015, Pike, 2019). Instead of using reaction-diffusion to generate the maculation, thresholding Gaussian noise and Gaussian blurred images are used to create the speckled and spotted patterns of eggshells. Like the reaction-diffusion a pattern sheet is used to reduce generation time and to improve heritability (Figure 7). This pattern sheet consists of a 1-dimensional change in Gaussian noise where Gaussian noise increases along the y axis.

A gradient is applied to each pattern prior to thresholding to allow for the belted and pole concentrated patterns of eggs seen in birds. Spiral patterns are generated by applying random walk and copying the pattern to different locations (Pike, 2015). Each egg can have up to 4 maculation patterns allowing for different patterns such as spots, speckles and spirals to exist concurrently. For simplicity patterns 3 and 4 are copies of patterns 1 and 2, respectively. These copies are shifted on the X-axis such that they are identical in terms of pattern generation, without being the exact same pattern. Large blotches are created by blurring and re-thresholding patterns 1 and 3. Spirals are created by applying random walk to patterns 2 and 4.

Egg background and maculation colour uses a CIELAB colour space and is controlled using genes for pigment ratio and pigment deposition. Pigment type determines the ratio of the red-yellow pigment protoporphyrin and the green-blue pigment biliverdin. While pigment deposition determines the amount of pigment added.(Hanley et al., 2015) (Figure 8). The CIELAB colour is calculated by multiplying the pigment L, A and B values by their ratio and deposition. Maculation colour is calculated by adding the deposition to that of the background (maculation are always darker) and by averaging the pigment ratio weighted by deposition. The maculation copies (patterns 3 and 4) have their pigment ratio taken from their copied maculation but have their own deposition. The luminance of maculation patterns and is calculated by subtracting the luminance from that of the background. Egg pore patterns are generated with the same speckling method as the reaction-diffusion generation system though at a reduced range. The full list of genes is given in table 2.



***Figure 7****. Schematic of the egg pattern generation system used by CamoEvo. A section of the Gaussian noise gamut is selected, blurred and shaded with a gradient before being thresholded. Colouration is then applied to both the background and the patterns.*



***Figure 8****. Egg Colouration Gamut.* *Deposition and Pigment Ratio operate in combination to change the CIE LAB colour of the eggs. Each pigment has a set CIELAB colour for when it is at its highest deposition. Deposition determines the amount of pigment used while pigment ratio determines the % of protoporphyrin and biliverdin pigment.*

***Table 2****. The pattern generation genes for the Gaussian egg pattern generation.*

|  |  |
| --- | --- |
| gene name | Function |
| pore\_intensity | Speckling noise (StdDev) |
| pore\_sigma | Speckling sigma (level of blurring) |
| mac1\_X | X coordinate that maculation 1 is sampled from. |
| mac1\_Y  mac1\_threshold | Y coordinate that maculation 1 is sampled from.  The threshold value for creating maculation 1. |
| mac1\_sigma | Gaussian blurring applied to the sampled gaussian noise. |
| mac1\_median | Median filter which makes patterns rounder. |
| mac1\_eNoise | Gaussian noise applied to the pattern creating broken edges |
| mac1\_cs\_sigma | Pattern shading sigma (level of blurring). |
| mac1\_cs\_height | Pattern shading height. |
| mac1\_cs\_length | Size of the gradient. |
| mac1\_cs\_intensity | Intensity of the gradient. |
| mac2\_x | X coordinate that pattern 1 is sampled from. |
| mac2\_y | Y coordinate that pattern 1 is sampled from. |
| mac2\_sigma | Gaussian blurring applied to the sampled gaussian noise. |
| mac2\_threshold | The threshold value for creating maculation 2. |
| mac2\_\_rWalk | Determines whether random walk is applied to maculation 2. |
| bg\_deposition | The amount of pigment added to the background. |
| unit\_D\_0 | Unit 0 for category D (deposition) for targeted duplication. |
| pttn1\_deposition | The amount of additional pigment added to pattern 1. |
| unit\_D\_1 | Unit 1 for category D (deposition) for targeted duplication. |
| pttn2\_ deposition | The amount of additional pigment added to pattern 2. |
| unit\_D\_2 | Unit 2 for category D (deposition) for targeted duplication. |
| copy1\_deposition | The amount of additional pigment added to pattern 3. |
| unit\_D\_3 | Unit 3 for category D (deposition) for targeted duplication. |
| copy2\_deposition | The amount of additional pigment added to pattern 4. |
| unit\_D\_N | End marker for the category D (deposition). |
| egg\_exposure | Adjustment to luminance of the egg. |
| egg\_smudge | Applies a mean filter to blend patterns together. |
| unit\_P\_0 | Unit 0 for category P (pigment) for targeted duplication. |
| bg\_pigment | The ratio of pigments used by the background. |
| unit\_D\_1 | Unit 1 for category P (pigment) for targeted duplication. |
| pttn2\_ deposition | The ratio of pigments added by pattern 1 and its copy. |
| unit\_D\_2 | Unit 1 for category P (pigment) for targeted duplication. |
| pttn2\_ deposition | The ratio of pigments added by pattern 2 and its copy. |
| unit\_D\_N | End marker for the category P (pigment). |

# How Slides are Generated

Each CamoEvo slide is generated using one of the background images and a number of targets equal to the group size selected. The sequence with which the targets and the backgrounds are selected is randomly generated. If the number of backgrounds provided is lower than the number of slides necessary to display all the targets, additional random sequences of backgrounds will be generated and concatenated until the sequence is equal to the number of slides required. This means that depending on the number of backgrounds provided, not all backgrounds will be displayed the same number of times.

When adding targets to each background the target is assigned a random x and y coordinate and rotation. The range of x and y coordinates are restricted such that the target cannot spawn within the target’s diameter of the centre. This prevents targets from being immediately located due to being at the observer’s point of focus (Talas et al., 2020). For each target, a random angle between -1800 and +1800 is assigned. This angle range can be adjusted for each target type by editing the "/1 CamoEvo/Targets/Target\_Settings.txt" file. By default, the triangle and moth targets use a range of-400 to +400 such that they are orientated facing upwards.

If the group size is greater than one, multiple targets will be added to each background. Each time a target coordinate is assigned, the distance between the target and existing targets is calculated and the coordinates adjusted to prevent target from spawning too close to each other.

# How to run your own Image Analyses

If you wish to run your own custom image analyses, the slides presented by CamoEvo are all saved as .tiff images within the population folder including an overlay for both the target and the local background. For slides with multiple targets, the overlays for all the targets and the local backgrounds are provided. You can convert these overlays to ROIs in ImageJ using the ‘To ROI Manager’ function allowing you to measure these regions using other ImageJ based image analysis tools.

# Additional CamoEvo features

## Phenotype Plotter

Allows you to select a CamoEvo population and generate a plot showing the phenotype (appearance) of individuals from each generation. The interface allows you to adjust the space between rows and columns, the number of individuals shown per generation, the range of generations used, the interval between generations, the ordering of individuals and the labels. The ordering can either be: ranked, individuals are shown in order of fitness; category, individuals are shown by category (Parents, Mutants, and Offspring) and age, oldest to youngest. *Note: the plot image is quite large and may need to be down-scaled before saving.*

## Evolution gif

Allows you to create a gif showing the change in phenotype for all individuals with each generation. You can adjust using the interface the ordering of individuals and the labels. The ordering can either be: ranked, individuals are shown in order of fitness; category, individuals are shown by category (Parents, Mutants, and Offspring) and age, oldest to youngest. To save the output as a .gif, simply use ImageJ’s “File/Save As/ Gif…”.

# How to Customise CamoEvo

## Changing the GA Settings

Using CamoEvo’s interface you can adjust the number of generations and the population size used by its genetic algorithm. However, if you wish to change other aspects of the algorithm e.g., mutation rate or crossover probability you will need to change the default settings that CamoEvo uses with ImageGA.

To prevent other users of ImageGA from interfering with or changing CamoEvo, we provide a separate settings storage .txt file for CamoEvo so that CamoEvo’s settings can be adjusted independently. We also provide an interface macro ‘/1 CamoEvo / Edit Algorithm.txt’ specifically for changing the settings used by CamoEvo that are not altered by the advanced settings (number of generations and population size). The genetic algorithm settings can also be manually adjusted by changing the “AlgorithmSettings.txt” file within the population folder.

By default, CamoEvo uses the following:

Deletion Pool = **50%**

Breeding Pool = **25%**

Mating System = **random**

Offspring Mutation Probability = **0.0010**

Offspring Mutation Distribution = **Normal**

Offspring Mutation Strength = **10**

Non-breeder Mutation Probability = **0.0500**

Non-breeder Mutation Distribution = **Normal**

Non-breeder Mutation Strength = **8**

Crossover Type = **Two Point**

Crossover Probability = **1.0000**

Recombination = **incomplete**

Mating System = **Random**

Alpha = **1.1**

Beta = **1.9**

Offspring Inversion Probability = **0.0000**

Offspring Duplication Probability = **0.0000**

Offspring Scramble Probability = **0.0000**

Offspring Targeted Duplication Probability = **0.001**

Non-breeder Inversion Probability = **0.001**

Non-breeder Duplication Probability = **0.001**

Non-breeder Scramble Probability = **0.0000**

Non-breeder Targeted Duplication Probability = **0.01**

Crowd Operator = **mutant-mate**

Delta threshold = **0.9**

Number Removed = **4**

## Creating Custom Target Shapes

As individuals using CamoEvo are likely to want to tailor the size and shape of targets to their own experiments we provide an interface for creating custom targets. Users can either import existing targets they’ve made for other experiments or create new targets. To make a target type you need an image of the shape you want. This image needs to be a binary black and white 400x400px image where the target shape is drawn in white. The target should fit entirely within a 400px diameter circle. We provide a template circle for the purposes of creating targets. Alternatively, you can draw a shape using the creator itself.

Once you have picked a shape you can then choose which pattern generation system the target will use (animal maculation, animal bilateral maculation or egg maculation) and the size of the target. To make target scaling easier the target will be displayed against a sample background from the background image gallery chosen.

If you do not want to use the creator, you can manually create new targets by copying one of the rows in the "/1 CamoEvo/Targets/Target\_Settings.txt" file and adjusting the values. You will also need to create a folder containing the target shape containing the target image titled “target.png”. The columns are organised as follows:

|  |  |
| --- | --- |
| *ID:* | *The unique ID for the target type distinguishing it from the others.* |
| *Shape:* | *The name of the folder from which the target shape is taken.* |
| *Size* | *The size in pixels of the target.* |
| *Pattern:* | *The type of pattern generation used (animal\_uni, animal\_bi and egg).* |
| *Centre:* | *The X coordinate of the sagittal plane for bilateral symmetry.* |
| *minAngle:* | *Minimum angle of rotation.* |
| *maxAngle:* | *Maximum angle of rotation,* |

# File and Code Organisation

For the purposes of making CamoEvo easier to edit, the code is organised as follows:

## 1 CamoEvo:

Folder containing all of the plugins intended to be used at the outset by CamoEvo. This includes the CamoEvo starting interface, the phenotype and evolution plotters and the GA settings interface:

* Interface/: *Contains the UI and a .pptx file with all the original slides used to make the UI.*
* Population/: *Contains the folders for all the populations created.*
* Targets/: *Contains all the Target types including the custom generated ones.*
* CamoEvo\_Game: *Runs CamoEvo.*
  + *Contains the UI for setting up a game and loading previous games.*
  + *Contains a built-in tutorial.*
* Edit Algorithm: *Allows the user to edit the settings of ImageGA that aren’t changeable using the built in UI (Mutation, Mating Systems and Crossover).*
* Get\_Evolution\_gif: *Creates an animation showing the evolution of the entire population.*
  + *Targets are ordered by fitness, top left = fittest, bottom right = least fit.*
* Get\_Phenotype\_Plot: *Customisable plot of ranked targets over multiple generation.*
  + *Customizable X and Y interval*
  + *Customizable column spacing*
  + *Targets are ordered by fitness, top left = fittest, bottom right = least fit.*

## 2 CamoReq:

Contains all the requisit files for running CamoEvo and a number of the .txt files used to store information. Most importantly among these are:

* Camo\_Evolution\_Game: *Runs the game for what ever population is listed in PopLocation.txt*
  + *Applies the phenotype generator*
  + *Saves the phenotype images*
  + *Runs the game*
  + *Runs the camouflage analyses*
* Colour Maps/: *Contains the code and corresponding maps for the CIELAB egg and animal spaces.*
  + colour\_map\_animal = the animal camouflage CIE LAB space
  + colour\_map\_egg = the egg CIE LAB space
* Patterns/: *Contains the reaction-diffusion and gaussian pattern sheets*
  + eggPatterns.jpg = egg patterns
  + pattern1.jpg = animal patterns
* Egg\_Gene\_Template: The gene template used for egg patterns.
* Gene\_Template: *The gene template for the animal patterns.*
* Generate\_Eggs: *The code for converting the egg pattern genes to their phenotype.*
  + *Output images are not saved using this script*
  + *As batch mode is true, output images aren’t visible*
* Generate\_Targets: *The code for converting the animal pattern genes to their phenotype.*
  + *Output images are not saved automatically*
* Phenotype\_Maker: *Allows the user to test out the phenotype for different genotypes and save the corresponding genotype and phenotype.*
* Reaction\_Diffusion\_Pattern\_Generator: *Creates reaction-diffusion patterns.*
* Reaction\_Diffusion\_Pattern\_Generator\_gradient\_smlap\_adaptivek: *Creates reaction-diffusion pattern sheets.*

# References

ALLEN, W. L., CUTHILL, I. C., SCOTT-SAMUEL, N. E. & BADDELEY, R. 2010. Why the leopard got its spots: relating pattern development to ecology in felids. *Proceedings of the Royal Society B: Biological Sciences,* 278**,** 1373-1380.

BONNEY, R., SHIRK, J. L., PHILLIPS, T. B., WIGGINS, A., BALLARD, H. L., MILLER-RUSHING, A. J. & PARRISH, J. K. 2014. Next Steps for Citizen Science. *Science,* 343**,** 1436.

GOLDBERG, D. E. & HOLLAND, J. H. 1988. Genetic Algorithms and Machine Learning. *Machine Learning,* 3**,** 95-99.

HANLEY, D., GRIM, T., CASSEY, P. & HAUBER, M. E. 2015. Not so colourful after all: eggshell pigments constrain avian eggshell colour space. *Biol Lett,* 11**,** 20150087.

KONDO, S. & MIURA, T. 2010. Reaction-Diffusion Model as a Framework for Understanding Biological Pattern Formation. *Science,* 329**,** 1616.

MCCAMY, C. S. 1992. Correlated color temperature as an explicit function of chromaticity coordinates. *Color Research & Application,* 17**,** 142-144.

NIU, Y., SUN, H. & STEVENS, M. 2018. Plant Camouflage: Ecology, Evolution, and Implications. *Trends in Ecology & Evolution,* 33**,** 608-618.

PIKE, T. W. 2015. Modelling Eggshell Maculation. *Avian Biology Research,* 8**,** 237-243.

PIKE, T. W. 2019. Quantifying the maculation of avian eggs using eggshell geometry. *Ibis,* 161**,** 686-693.

SCHNEIDER, C. A., RASBAND, W. S. & ELICEIRI, K. W. 2012. NIH Image to ImageJ: 25 years of image analysis. *Nature Methods,* 9**,** 671-675.

TALAS, L., FENNELL, J. G., KJERNSMO, K., CUTHILL, I. C., SCOTT-SAMUEL, N. E. & BADDELEY, R. J. 2020. CamoGAN: Evolving optimum camouflage with Generative Adversarial Networks. *Methods in Ecology and Evolution,* 11**,** 240-247.

TROSCIANKO, J., SKELHORN, J. & STEVENS, M. 2017. Quantifying camouflage: how to predict detectability from appearance. *BMC Evolutionary Biology,* 17**,** 7.

TROSCIANKO, J. & STEVENS, M. 2015. Image Calibration and Analysis Toolbox – a free software suite for objectively measuring reflectance, colour and pattern. *Methods in Ecology and Evolution,* 6.

TROSCIANKO, J., WILSON-AGGARWAL, J., STEVENS, M. & SPOTTISWOODE, C. N. 2016. Camouflage predicts survival in ground-nesting birds. *Scientific Reports,* 6**,** 19966.