Year 2 Project

Image processing on cuttlefish with light compensation

By: Yaoming Zhang (201601262), Zhihao Yu (201601188)

Zhijie Yan (201601118), Zenan Zhu (201601386)

Supervised by: Joseph Spencer

Department of Electrical Engineering and Electronics

13 March 2022

Abstract

This document describes the image processing on cuttlefish with light compensation. It is known that cuttlefish could change their surface colour according to the living environments. The project aims to find their regular changing patterns. Fish injected with different types of chemical substances are photographed in advance. The essential idea of the experiment is to transfer images' RGB values into chromatic values HLS (hue, luminance and saturation) by programming, which shows the features of images in a new dimension. By regrouping the chromatic values, fish distributions under different chemical injections can be obtained. Besides, luminance is chosen to research detailly since the light intensity is more likely to change and affect the image processing result. The light compensation method is introduced to achieve the same light intensity of the same fish (photographed at different times), making the fish distribution more consistent. This project is helpful to monitor the condition of the cuttlefish efficiently and could have applications in the cuttlefish farming industry and fish conversation organization. Further work should focus more on quantifying the disparities of different fish clusters in the chromatic figures after using light compensation, which is not completed due to time limit.

Declaration

I confirm that I have read and understood the University's definitions of plagiarism and collusion from the Code of Practice on Assessment. I confirm that I have neither committed plagiarism in the completion of this work nor have I colluded with any other party in the preparation and production of this work. The work presented here is my own and in my own words except where I have clearly indicated and acknowledged that I have quoted or used figures from published or unpublished sources (including the web). I understand the consequences of engaging in plagiarism and collusion as described in the Code of Practice on Assessment (Appendix L).

Contents

List of Figures

List of Tables

Introduction

Image processing has been increasingly prevailing in recent years, which can be applied in versatile fields. It refers to using computers to process digital images through an algorithm [1]. It could be used to find relations from pictures and model them in the form of a multidimensional system.

Image processing is a popular topic with many different processing methods and algorithms [2, 3, 4, 5, 6]. One focus is to find an algorithm that could be widely applied in different situations and efficient for feature extraction. Also, for images under different light intensities, the extracted data will be considerably affected and cause inaccuracy in the result. To reduce the error, algorithms could be found to process the image to reduce the effect of light intensity which is called light intensity compensation. As a result, practical algorithms and light intensity compensation are essential research directions in image processing field.

For previous studies, some research has already investigated the feelings of cuttlefish injected with chemicals [7]. The primary methodology that the project use is HLS transformation which is a common method that has already been developed for many years and has proved to be helpful [2, 3].

The report will initially introduce the materials used in the project, including hardware and software. Based on these experimental tools, concrete experimental procedures have been designed to implement the project. Ahead of designing the project, methods will be introduced to decide how to manipulate every section of the experiment. However, problems will occur during the experiment inevitably. Therefore, the report will then demonstrate problems encountered in the experiment and point out the corresponding solutions. The next step is to record the data generated in the experiment. As the project is divided into two parts, the report will demonstrate data in a separate section. The report will analyse the result and then give corresponding error analysis regarding experimental data recorded in the experiment. The final part of this report concerns findings discovered in the project. Furthermore, the reflection of the report and the future expectation are also involved in the conclusion.

The project's main purpose is to study the relationship between the pixels on the cuttle fish back and the condition of the cuttlefish. The condition of cuttle fish is decided by the concentration of chemicals that have been injected in the cuttlefish. To achieve the major objective, the experiment will be divided into two parts and then merged. Therefore, studying the approach of image processing and light compensation is also objective involved in the project.

Materials and Methods

2.1 Material

The hardware applied in the project:

- Inspection Lighting ×1 (Type No. DN1410   Make: CANYON)
- Webcam ×1 (Type No.CNE-CWC5   Make: DAYLIGHT)
- Trace paper ×n

The software applied in the project: PyCharm Community Edition

2.2 Theory

$$\begin{aligned} H &= 240 - 120.g/(g+b) \quad r = 0 \\ &= 360 - 120.b/(b+r) \quad g = 0 \\ &= 120 - 120.r/(r+g) \quad b = 0 \\ L &= (R+G+B)/3 \end{aligned}$$

$$S = [\max(R, G, B) - \min(R, G, B)]/[\max(R, G, B) + \min(R, G, B)]$$

where
$$\begin{aligned} r &= R - \min(R, G, B) \\ g &= G - \min(R, G, B) \\ b &= B - \min(R, G, B) \end{aligned}$$

Figure 1: Basic H, L, S Transformation [2]

Figure 1 shows the core algorithm that will be used in the project. The nonorthogonal processors R, G, B of the figure will be transformed into chromatic parameters H, L S, where H stands for Hue, L stands for Luminance, and S stands for Saturation [2]. The method has advantages of low computation and high accuracy. It is a fundamental transformation method well established in color science and easy to extract color features from an image, so it is conveniently adapted for chromatic monitoring and processing [2, 3].

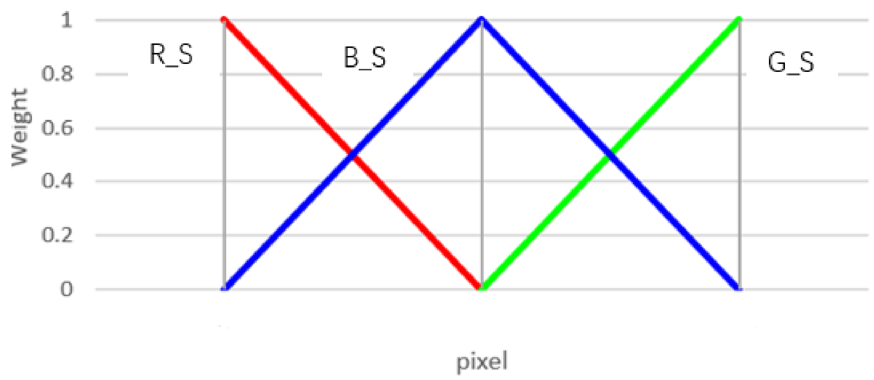

Figure 2: Second chromatic values processing

Figure 2 is another algorithm worth mentioning, which is the truncated triangular filter method [2]. The basic idea is to get a new set of RGB values by multiplying the filter with the corresponding HLS values and summing them. R_S, G_S and B_S are three filters. The x-axis represents the pixel and the y-axis represents the weight of the filters. In this project, the weight is assumed to be between 0 and 1. The equations are shown below [2].

$$R_s(S) = S_1 \times R_{S1} + S_2 \times R_{S2} + \cdots + S_n \times R_{Sn} = \sum_{k=1}^{n} S_k \times R_{Sn}$$

$$R_s(H) = \sum_{k=1}^{n} H_k \times R_{Sn}$$

$$R_s(L) = \sum_{k=1}^{n} L_k \times B_{Sn}$$

$$G_s(S) = \sum_{k=1}^{n} S_k \times G_{Sn}$$

$$G_s(H) = \sum_{k=1}^{n} H_k \times G_{Sn}$$

$$G_s(L) = \sum_{k=1}^{n} L_k \times G_{Sn}$$

$$B_s(S) = \sum_{k=1}^{n} S_k \times B_{Sn}$$

$$B_s(H) = \sum_{k=1}^{n} H_k \times B_{Sn}$$

$$B_s(L) = \sum_{k=1}^{n} L_k \times B_{Sn}$$

Here n means the length of the filter.

During the experiment, these RGB can be transferred to HLS again for further analysis.

2.3 Procedure

Cuttlefish image processing:

The chromatic transform on the fish images is the essential idea in this project. Its detailed procedure will be elaborated on first.

Figure 3 is the example fish "sham" used. It behaves normally without chemical injections. The figure is captured at t = 0.

Figure 3: Sham1068625 t=0

In the beginning, the head and tail of the fish were identified. Their middle points were selected to form a red line as shown in figure 4. Two sets of data shown on the figure are coordinates of the points respectively. The line can represent the features of the whole fish.

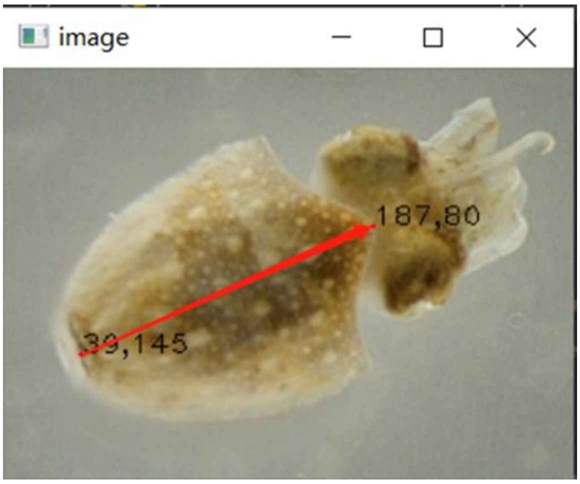

Figure 4: The chosen line representing the fish features

Then RGB values of each pixel on this line were obtained by programming. The corresponding first HLS values were available by applying the basic HLS transform algorithm equations in figure 1. These values were then normalized to reduce variation. Figures of RGB and first HLS values were plotted.

The third step was applying triangular linear filters on the first HLS. As mentioned in the theory part, the filters transferred the first HLS and made them be variables in the new RGB. In other words, R values will be determined by H, L and S respectively (R(S), R(L), R(H)), so are G and B values. As a result, 9 new RGB were obtained.

The fourth step was applying the basic HLS transformation again. 9 new RGB can be transferred to 9 second HLS, whose variables are H, L and S. The second HLS values were combined in pairs randomly and they can form 36 combinations. Every pair formed a point in a two-dimensional space. The second chromatic transformation reduced the features of the whole fish into one point. All two-dimensional combinations were plotted.

The fifth step was to repeat the procedure above on the same fish "sham" when time equals 5 and 10. Hence three points can be drawn in every combinational relationship of the second HLS. Three points were then averaged to one point. This point is thought to represent the fish status "sham."

Finally, the processes were replicated on all fish images of different status, 0.5%AA, 0.5%AA+2%L, 2%AA and 2%AA+3%L. Their performance on different second HLS relationships was observed to find their distribution.

The experimental steps below further investigate the relationship between RGB values and light intensity. Light compensation was achieved to make the fish image processing more accurate.

Light intensity compensation:

The mood of the cuttlefish is analyzed through image processing of the images taken. However, due to some external factors, the light intensity of each photo cannot be guaranteed to be consistent. Therefore, to make the experiment more convincing, the compensation function is used to ensure the fish is exposed to the same light intensity.

Blue was selected to verify the function.

First, the LED lamp is fixed on the table, and the tube is bent so that the light falls directly on the image. Then, a piece of transparent paper is placed around the LED lamp to reduce unnecessary reflections and to keep the saturation of the resulting photo at about 0.5, which reduces the error of analysis. Finally, the camera is clipped to the lamp and the LED light intensity is adjusted to take pictures.

Figure 5: Blue image under light condition 1

Figure 6: Blue image under light condition 2

Figure 7: Blue image under light condition 3

First of all, blue photos under three light intensities were captured by the camera as figure 5, 6 ,7 and processed by the program.

In the first part, two pixels were selected from the inputted image and a straight line is determined through these two pixels. The program automatically read the values of R, G and B on this straight line and output these values. Then, the graph of R, G, B values were plotted.

Through the HLS transformation algorithm, the figure for H, L, S was plotted.

The fixed area selected in the image was a line with a length of ten pixels. After reading H, L and S on the line, the program needs to sum up these data and take an average value to obtain a stable value, which will be used for subsequent analysis.

After the previous calculation, the mean value of H, L, S under three different light intensities were obtained, and the line graph of H-L and S-L were drawn respectively.

Cuttlefish processing after light compensation:

This part of the experiment is to apply the result from light compensation to the cuttlefish images.

The first step was to identify two points on the background of the image. RGB values on the line formed by these two points were obtained. Lightness (L) was then available by basic HLS transformation.

The second step was repeating the method above on the fish under the same injection (time equals 0, 5 and 10). Their difference of L was calculated.

The third step was to achieve the same lightness of fish by programming.

Finally, same procedures mentioned in the cuttlefish image processing part were implemented again to find the distribution of fish after light compensation.

2.4 Problem occurred

Cuttlefish image processing

Regarding image processing on the cuttlefish, some problems come from the bugs of programming itself. The definition and use of the triangular linear filter are not very clear initially, making the coding progress slow.

The most challenging part is data analysis after implementing all coding parts. One type of useful relationship should be chosen among all second HLS relationships.

Sometimes a particular fish condition cluster is very ambiguous due to outliers. Besides, the project requires to choose the fish's head and tail by manual, which increases the contingency of the result.

Light intensity compensation:

In the light compensation part, one of red, green, and blue colors should be selected for subsequent experiments. When these three colors are directly exposed to light, red and green have a more serious reflective problem.

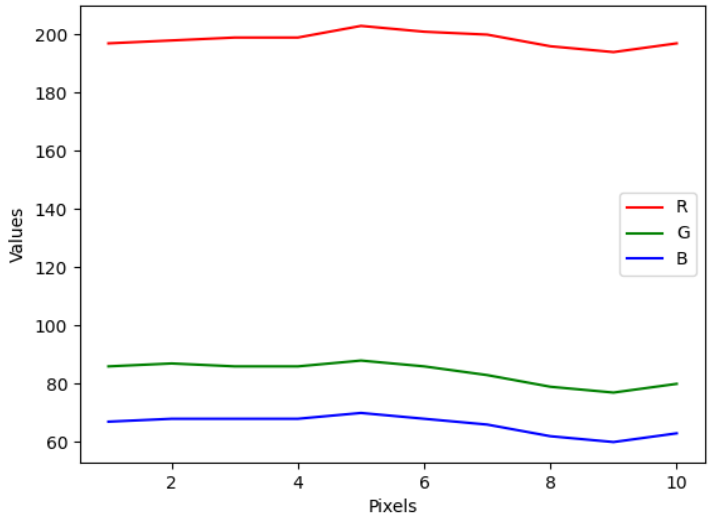

Figure 8: R, G, B values for Red

It can be observed from figure 8 that these three values are unstable, and this phenomenon also appears in green, which is shown in figure 9.

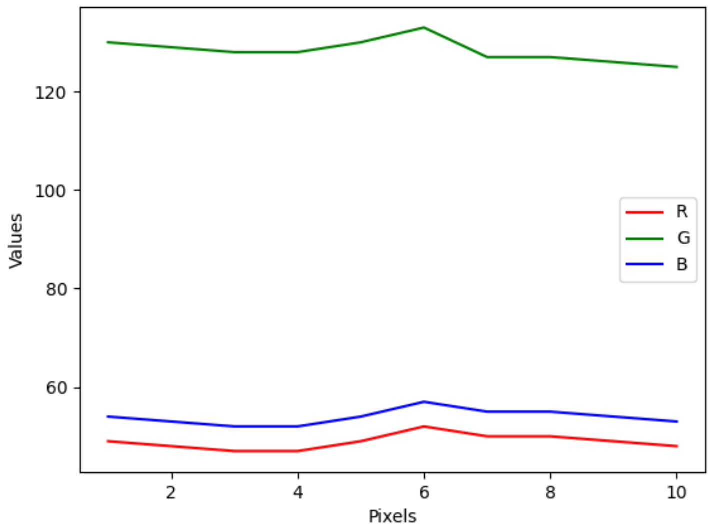

Figure 9: R, G, B values for Green

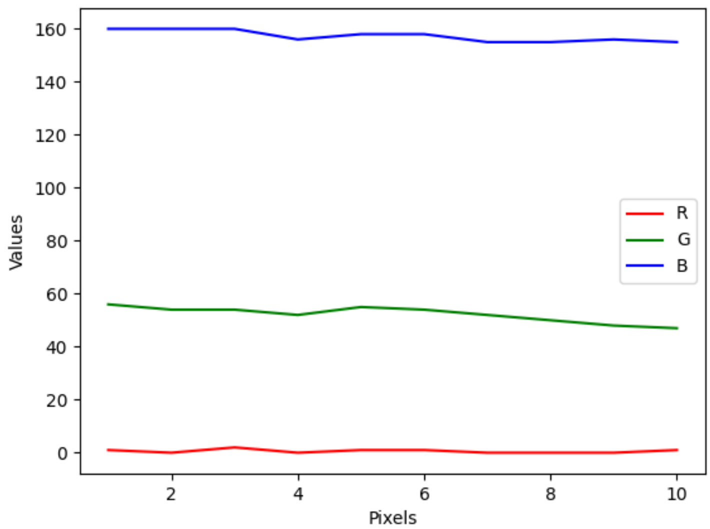

Figure 10: R, G, B values for Blue

It can be obtained from figure 10 that the values are more stable. Therefore, blue is chosen to carry on the subsequent research to make the result more reliable.

In addition, when it comes to the part of making the hypothesis, due to the limited shooting conditions, the photos taken have a great degree of uncertainty, which leads to the irregular fluctuation of H, L and S values within a certain range. To solve this problem, more photos should be taken to obtain the specific range of changes in H, L and S, and then analyze the average value within this range, so as to obtain a reasonable hypothesis.

Furthermore, problems also occurred during the experiment. It is difficult to keep the camera stable, which leads to poor focusing. To prevent photos from being blurred, the camera is placed on a stand to shoot a clear photo. Additionally, the light generated by the lamp did not spread evenly on the paper. To achieve demanding requirements, trace paper is used to make the light distribute on the paper uniformly.

Results

3.1 Results

Cuttlefish image processing

In terms of the fish image processing part, the results will be shown below step by step.

After the second step:

Figures 11 illustrates RGB values obtained from the line in figure 4. The x-axis represents each pixel, and the y-axis represents the concrete RGB values.

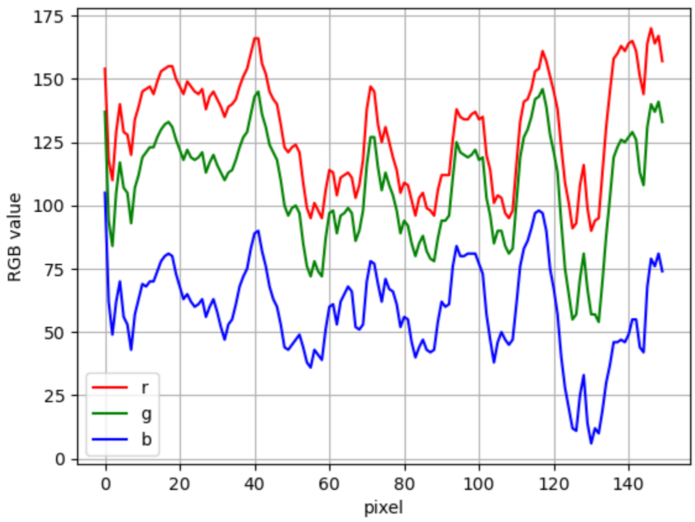

Figure 11: RGB values of sham

Figure 12 illustrates first HLS values after applying basic chromatic transform. Similar with figure 4, the x-axis represents each pixel. This time, the y-axis represents first HLS values.

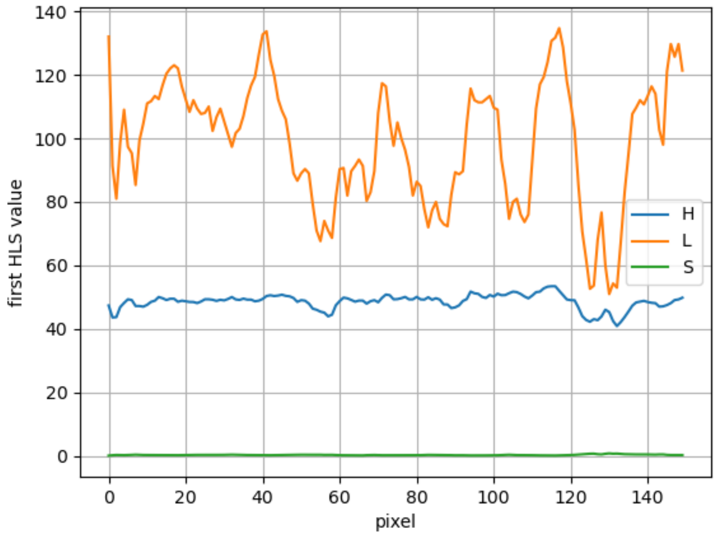

Figure 12: First HLS of sham

By utilizing normalization, the first HLS is shown in figure 13.

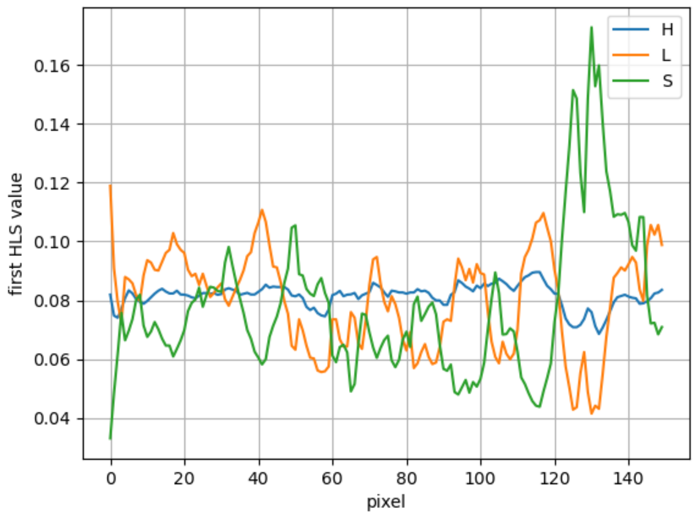

Figure 13: First HLS of sham after normalization

After the fourth step:

The second HLS values are calculated and are shown in table 1. Any two of them can form a point in the two-dimensional space. Hence 36 figures can be plotted.

Table 1: Second HLS of sham

Function	H(H)	L(H)	S(H)	H(L)	L(L)	S(L)	H(S)	L(S)	S(S)
values	241.34	4.06	0.34	250.02	3.99	0.32	214.72	3.88	0.32

Figure 14 shows one type of combination, which is H(H) and H(L). All figures can be found in appendix D.

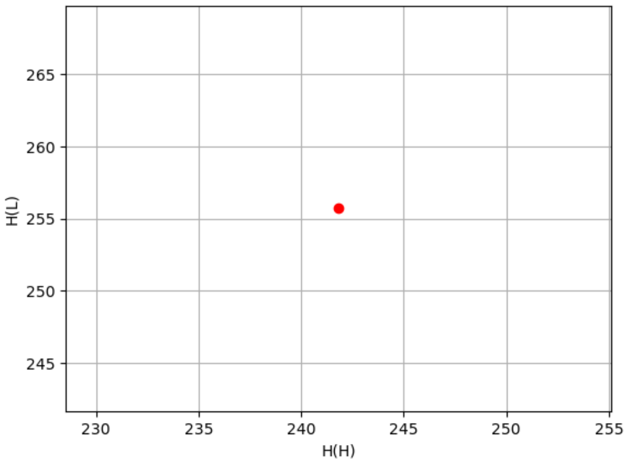

Figure 14: Sham fish (t = 0) in the 2-dimensional space (H(H) and H(L))

After the fifth step:

Figure 15 indicates the state of sham fish at time equals 0, 5 and 10. Still, the relationship between H(H) and H(L) is used as an example.

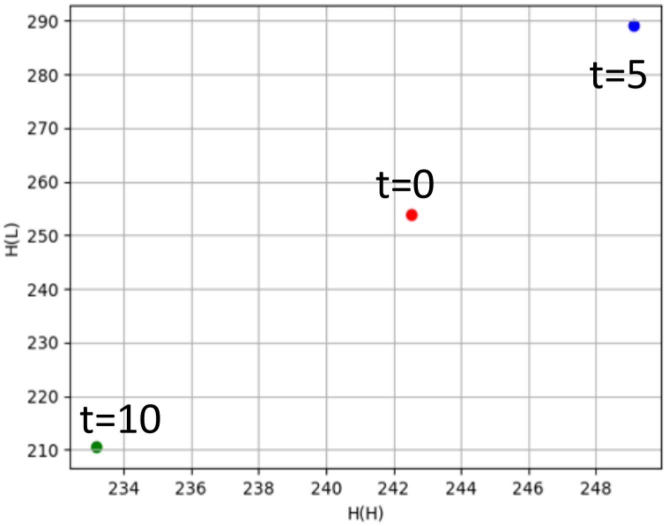

Figure 15 Sham fish (t = 0, 5, 10) between H(H) and H(L)

After the sixth step:

All possible relationships of second HLS are compared and figure 16 is chosen to indicate the cluster of all fish images. The reason for choosing the relationship between L(H) and L(S) is that the clusters of fish under different injections are most distinctive. Figure 17 reveals the clusters after averaging.

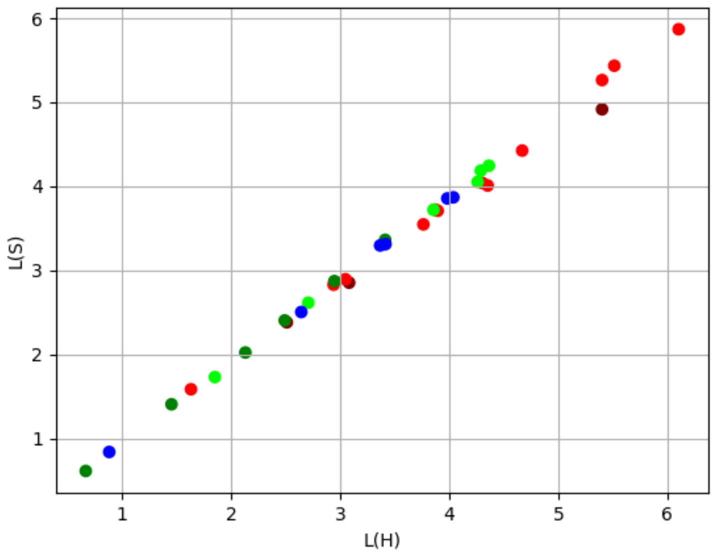

Figure 16 Clusters of all fish images before averaging

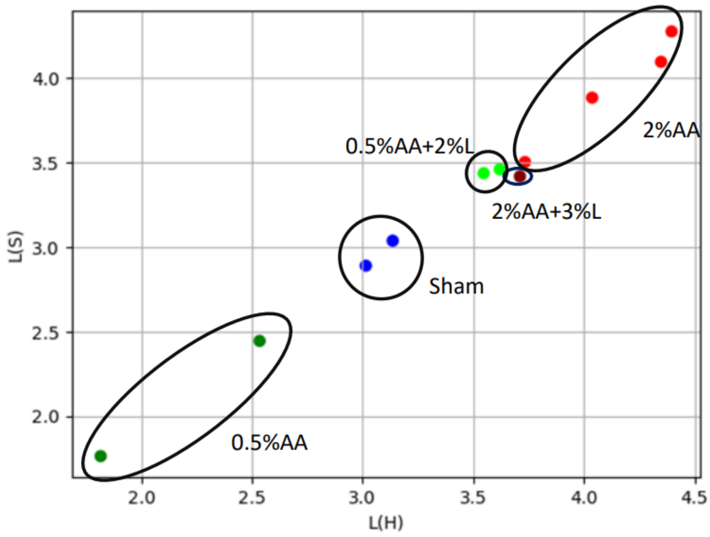

Figure 17: Clusters of all fish images

Light intensity compensation:

First of all, a blue photo under three light intensities is captured by the camera and processed by the program. The information of the image is shown in figures below.

It can be obtained from Figure 18 that R values are around 10, G values are around 36 and B values are around 45.

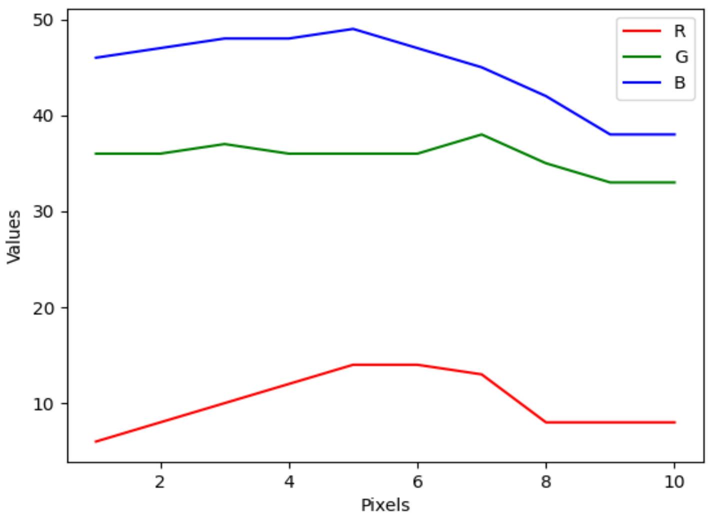

Figure 18: R, G, B values under light condition 1

The data can be obtained from figure 19 that the average hue is 189.14, the average saturation is 0.63 and the average light is 30.16.

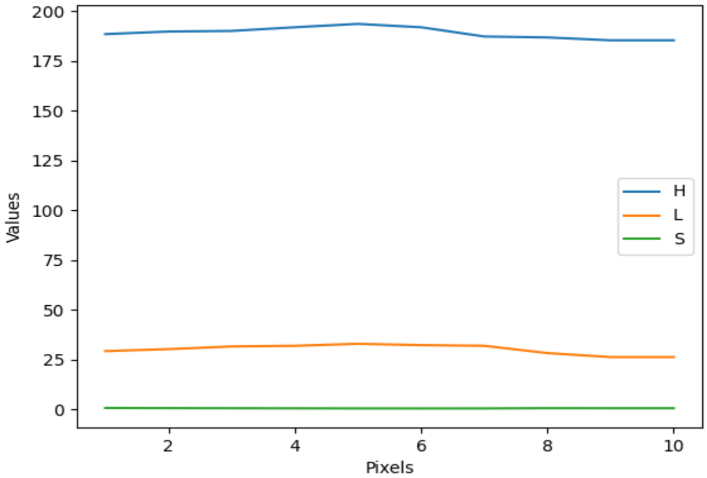

Figure 19: H, L, S values under light condition 1

Under light condition 2 in figure 20, R values are around 20 and G values are around 55, B values are around 67.

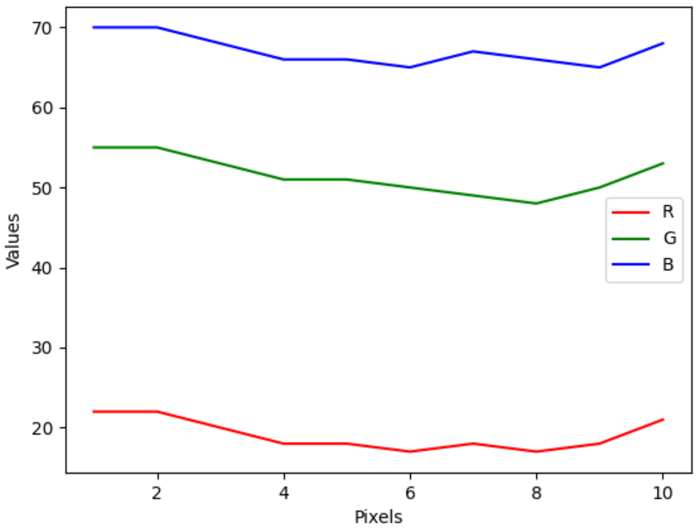

Figure 20: R, G, B values under light condition 2

The data can be obtained in figure 21 that the average hue is 191.64, the average saturation is 0.55 and the average light is 45.9.

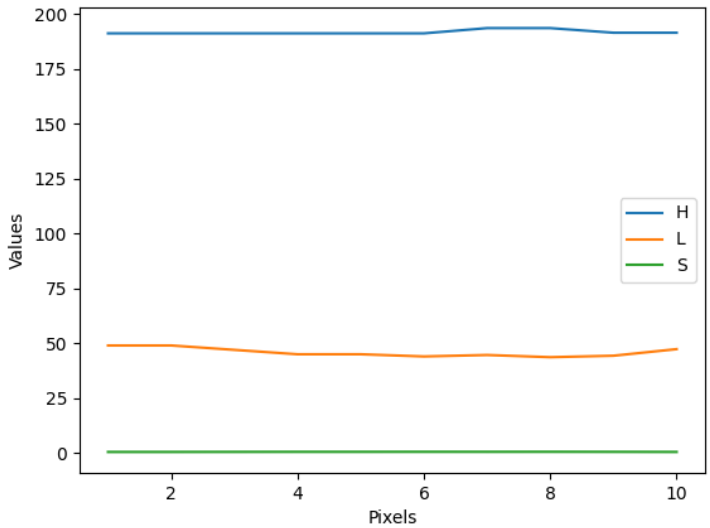

Figure 21: H, L, S values under light condition 2

Under light condition 3 in figure 23, R values are around 24 and G values are around 60, B values are around 80. It can be obviously observed that with the increasing of light intensity, R, G, B values also increase.

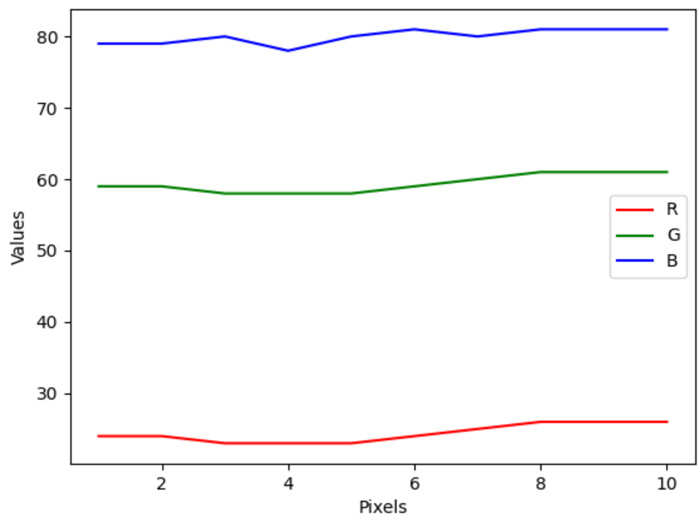

Figure 22: R, G, B values under light condition 3

The data can be obtained in figure 23 that the average hue is 193.63, the average saturation is 0.53 and the average light is 54.6.

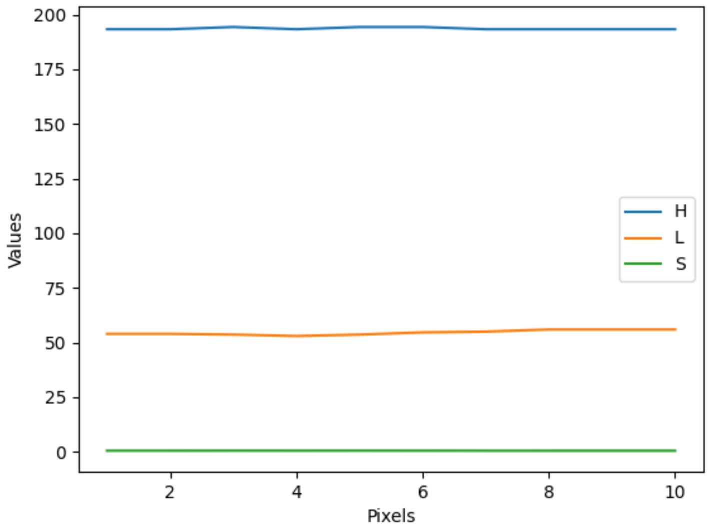

Figure 23: H, L, S values under light condition 3

Cuttlefish processing after light compensation:

After the third step:

The sham fish images can achieve light compensation. In figure 24, the upper 3 pictures are the original sham fish from time 0 to 5. The lower 3 pictures are the sham fish after light compensation, where the light intensities in these 3 pictures are the same.

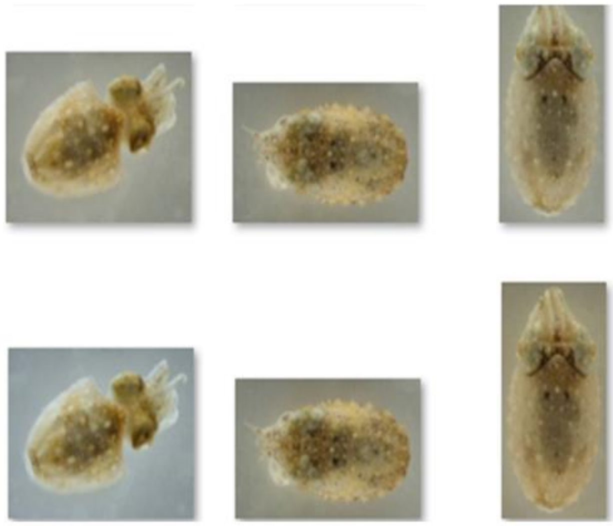

Figure 24: Sham fish before (upper) and after (below) light compensation

After the fourth step:

Figure 25 can be plotted illustrating the cluster of sham after light compensation.

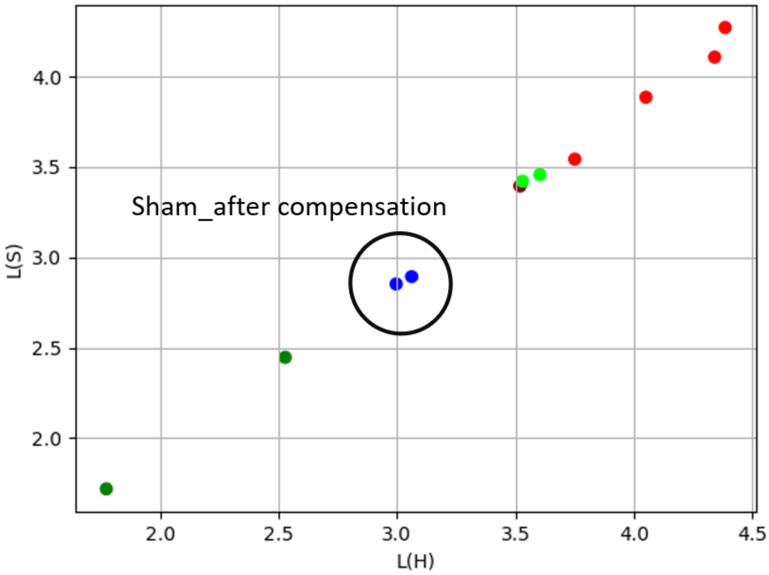

Figure 25: Clusters of sham after light compensation

3.2 Analysis

In this part, the resulting figures above will be analysed in detailed.

Cuttlefish image processing:

In figure 12, the basic chromatic transform algorithm transfers RGB to the first HLS. Three lines H, L and S are expected to represent the features of the fish. However, the first HLS values are still very messy. When the sham fish is replaced by 2%AA fish, figure 26 indicates its first HLS values. After comparing with figure 12, it is hard to distinguish some reasonable results. That is why the second HLS transformation is necessary.

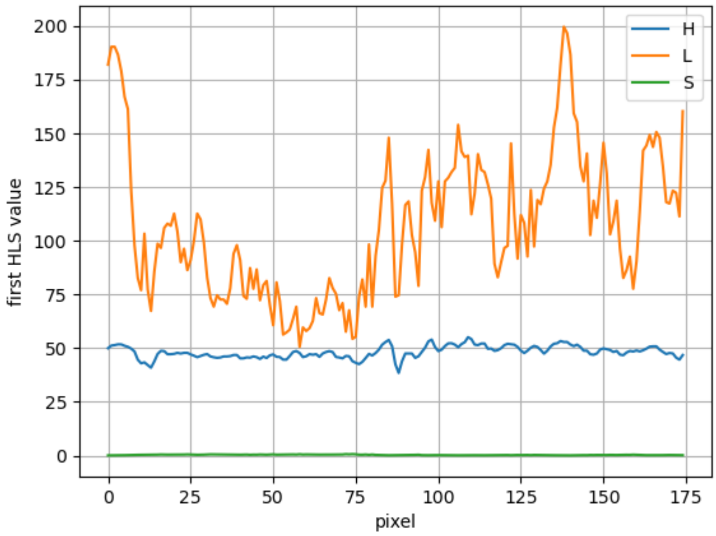

Figure 26: First HLS of 2%AA

It can be found from both figure 12 and 26 that H, L and S are in different regions. Variation should be reduced. Besides, the triangular linear filters have the weight from 0 to 1. That is why normalization is applied to first HLS.

After the second HLS transformation, features of the fish at a particular time are reduced from 3 continuous lines to 1 single point. When plotting the clusters, second

HLS values are averaging based on time and injection concentration since the experiment focuses on the overall performance of fish under the injection, rather than the difference among time. That is why figure 17 is used instead of figure 16.

From figure 17, it is easy to find that fish distribute differently when types and concentrations of injections are different. Their clusters are labelled in the figure. There exists an approximate linear relationship between L(H) and L(S), which is $y = x$. Sham fluctuates near the point (3, 3). It is speculated that when the concentration of AA increases, both L(H) and L(S) will increase linearly.

When the concentration is still low (0.5%AA), the cuttlefish feels one type of pain. If the concentration increases to 2%AA, the cuttlefish feels more pain.

The result can be applied when a new fish image is fed into the program. The type of pain the fish is enduring can be figured out based on their location.

Light intensity compensation:

After getting the values about H, L and S, line charts of H-L and S-L can be obtained. Therefore, reasonable assumptions can be made based on the charts.

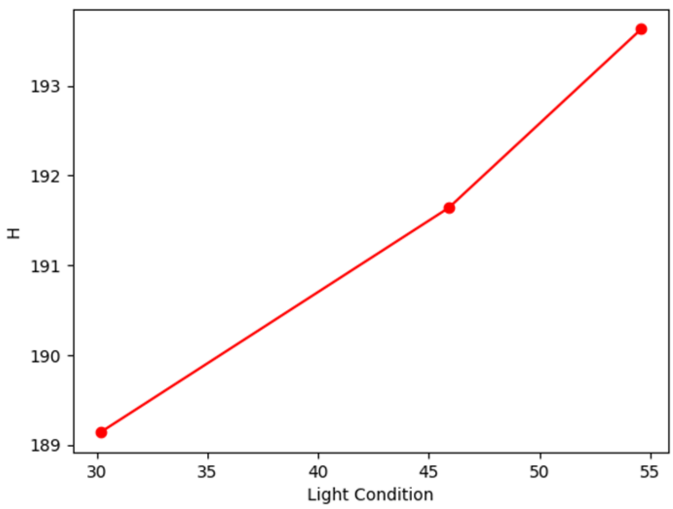

Figure 27: H with respect to L

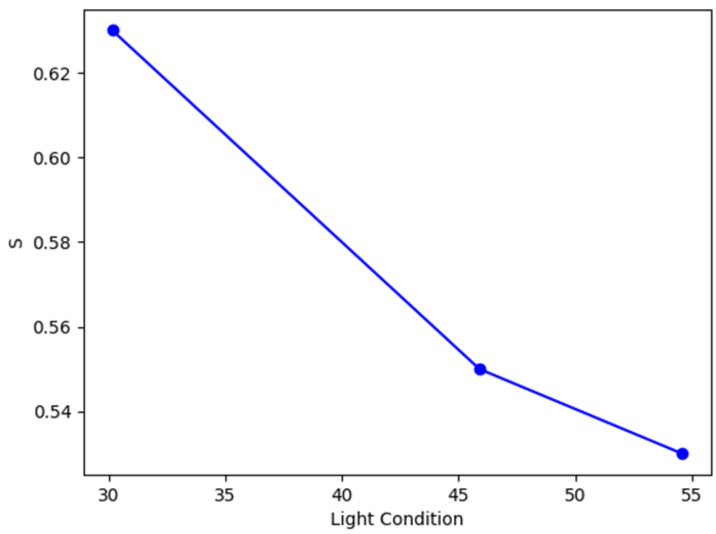

Figure 28: S with respect to L

First, assuming that the transformations of H and S with respect to L are linear, we can obtain expressions for two lines:

$$H = 0.184L + 183.58$$

$$S = -0.0041L + 0.75$$

According to the algorithms shown above,

When $r = 0 (r = R - \min(R, G, B))$

$$H = 240 - \frac{120 \cdot g}{(g+b)}$$

$$= 240 - 120 \cdot \frac{g+b-b}{(g+b)}$$

$$= 240 - 120 \cdot \left(1 - \frac{b}{(g+b)}\right)$$

$$= 120 + 120 \cdot \frac{b}{(g+b)}$$

So, $H = 120 + 120 \cdot \frac{B-R}{B+G-2R}$

And $S = \frac{B-R}{B+R}$

$$= \frac{B+R-2R}{B+R}$$

$$= 1 - \frac{2R}{B+R}$$

The R, G, B values at different pixels under different light conditions are shown in the following tables.

Table 2 shows the values when L = 30.16.

Table 2: R, G, B values under light condition 1

	1	2	3	4	5	6	7	8	9	10
R	6	8	10	12	14	14	13	8	8	8
G	36	36	37	36	36	36	38	35	33	33
B	46	47	38	48	49	47	45	42	38	38

Table 3 shows the values when L = 45.9.

Table 3: R, G, B values under light condition 2

	1	2	3	4	5	6	7	8	9	10
R	22	22	20	18	18	17	18	17	18	21
G	55	55	53	51	51	50	49	48	50	53
B	70	70	68	66	66	65	67	66	65	68

Table 4 shows the values when L = 54.9.

Table 4: R, G, B values under light condition 3

	1	2	3	4	5	6	7	8	9	10
R	24	24	23	23	23	24	25	26	26	26
G	59	59	58	58	58	59	60	61	61	61
B	79	79	80	78	80	81	80	81	81	81

Through calculation, the following assumptions can be made:

when the light intensity is weakened (when the difference is 10), it is necessary to increase G by 4.5% when H returns to the previous state; To return S to the previous state, we need to increase B by 12%.

For example, under the light condition 3, R = 24.4, G = 59.4, B = 80, increase G by 4.5% $G_2 = 62.073$ and B by 12% $B_2 = 89.6$.

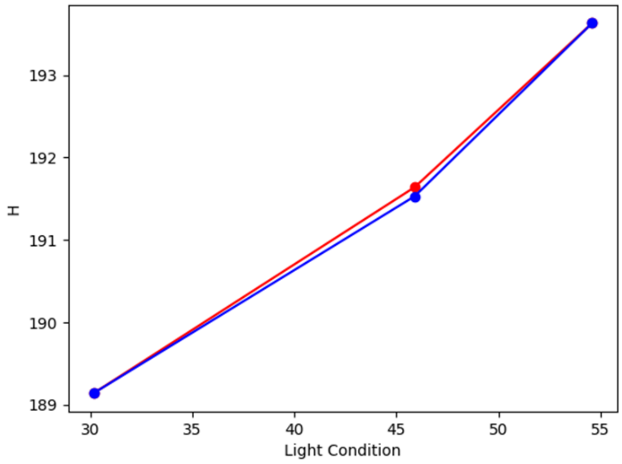

Figure 29: H values after light compensation

After getting H values after light compensation, the modified line is drawn. In Fig.29, the red line represents the original one and the blue line represents the modified one.

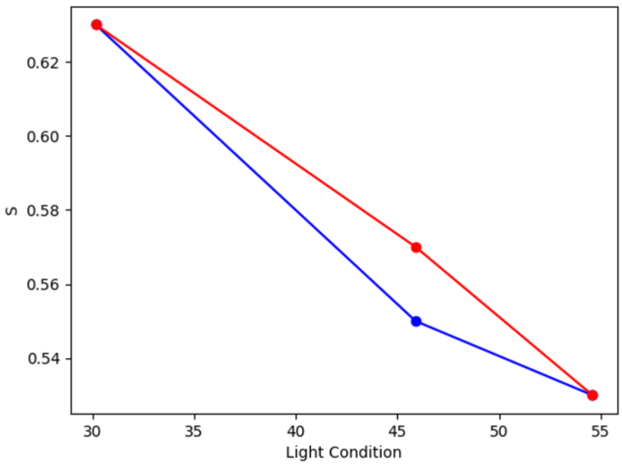

Figure 30: S values after light compensation

After getting S values after light compensation, the modified line is drawn. In Fig.30, the blue line represents the original one and the red line represents the modified one.

It can be seen from the charts that the numerical error between hue and Saturation after compensation and the original light intensity is within a controllable range, so the hypothesis is established.

Cuttlefish processing after light compensation:

It can clearly be found in figure 25 that the backgrounds of fish images have changed. After implementing the image processing part again and making comparison, the points for sham cuttlefish get closer, as shown in figure 26.

Due to the time limit, only sham fish have achieved the same light intensity. However, it is assumed that the light compensation makes the image processing more accurate.

3.3 Error Analysis

Cuttlefish image processing:

The most critical error comes from identifying the point. During the experiment, the straight line chosen for processing has to be picked manually in the program for each picture. It will cause the line picked every time to be slightly different for the same image. The characteristic RGB values extracted from the pixels on the selected line will change so that the imaging processing will be affected. Besides, the fish images the project used are in different sizes and scales, increasing the difficulty of choosing points.

Besides, although the project concludes the relations between the cuttlefish images and the condition of the cuttlefish, the number of images used for processing and getting the final conclusion is not enough. As a result, the conclusion of cuttlefish injected with different chemicals in different concentrations may have errors because of the limitation of the image number.

Light intensity compensation:

For the light compensation part, after processing three different blue images, figures show that R, G, B still have relatively large fluctuations at different positions, and this is due to limited experimental conditions. The light source covers a small area, and the scattering of light leads to different degrees of reflection in the image. This will have an impact on the experimental results: different light intensity will lead to different exposure degrees in different areas, which directly affects the values of R, G and B.

In addition, this greatly increases the experimental error, it can be obtained from Fig.30 that after modification, the margin of error is 3.6%, and some external factors can account for this error. First, only blue is selected for the experiment of light compensation, which led to our hypothesis being too one-sided, resulting in errors in verification. Second, the light source used scatters, which results in varying degrees of exposure on the image, and over or under exposure, color saturation is reduced. Therefore, further experiments are needed to complete the hypothesis.

Discussion and Conclusions

4.1 Conclusion

In conclusion, the project focuses on using image processing to find the relationship between the conditions of the cuttlefish and their location in the chromatic figure.

The main theories used are HLS transformation and Truncated triangular filters. The procedure of the whole project can be divided into image processing part and light intensity compensation part. The image processing includes extracting RGB values from pictures, doing HLS transformation on RGB values and further processing HLS values. The light intensity compensation contains taking photos of pictures under different light intensities, doing HLS transformation on RGB values and analysing the HLS values.

There are key findings in both parts. For image processing, the graph between L(S) and L(H) clearly shows the distribution of cuttlefish. It shows that cuttlefish with the same injection are in a similar region. For a new cuttlefish with unknown conditions, the image processing code can transfer the image to a point on the L(S) and L(H) graph. Hence the injection type and concentration can be obtained. In the light intensity compensation part, an algorithm is concluded to achieve the same luminous (L) value of different pictures by changing the original one's G and B. The algorithm is then applied to the sham fish in the fish image processing part. It can be found that sham fish's points get closer after the same process. It is assumed that the light compensation makes the image processing more accurate. These two findings are in line with the original objectives.

4.2 Achievements and self-evaluation

The year two project goes a tough but good way. In the project process, many factors influenced our progress such as the personnel change and time limitation that forced us to delay some further work in the future. Despite all the difficulties, there is an excellent outcome of our project. After getting the pictures of the cuttlefish, the software can do colour compensation in terms of the light intensity of the background and process the image data to find the condition of the fish. This product has various applications that will be discussed later.

During the four weeks, there is a good collaboration in the group. All the group members have good attendance in group discussion and communication. The whole project was separated into parts and allocated to group members. Everyone did their work well. If next time the work is allocated according to the advantages of different group members, there is no need to change work in the later weeks and the project could be done better.

4.3 Ethical consideration

As the software is related to fish, the regulation about animals should be concerned. According to Welfare of Farmed Animals (England) Regulations 2007 [8], the fish welfare is being protected. Although the project's software only needs the picture of the cuttlefish, some photos came from the injected cuttlefish with chemical substances in the past experiment. Moreover, unnecessary suffering to the animal is banned based on the Animal Welfare Act 2006 [8]. When taking photos, the exposure of light to fish should be decreased to improve fish welfare.

For the whole lifecycle of the product, it could give immediate and persistent benefit to sustainability as a sub-component of the fish monitoring system. The constraint is not needed for the typical users such as cuttlefish farm or fish protect organization. It is tough to use this software illegally since it is only applied to cuttlefish and only processes pictures.

4.4 Applications

The software the project developed is based on auto image processing that can evaluate the condition of the cuttlefish. After getting the pictures of the cuttlefish, the software could do colour compensation according to the light intensity of the background and process the image data to find the condition of the fish. As a result, the programme has two main functions: monitor the condition of the cuttlefish and the living condition of the cuttlefish. The product can be applied in various fields and have straight benefits to the industry, ecological environment, and society. For instance, aquariums can use this product to efficiently detect fish health or emotional condition at a low cost, which may discover potential risks of fish. Also, the cuttlefish farms can use the software to observe the breeding situation easily with some pictures. Furthermore, fish conversation organizations could also use software to protect marine biodiversity and the living environment. The core technique used in the product is image processing and light intensity compensation, which involves broader applications in the current camera and mobile phone market.

4.5 Future work

The future work can be divided into completing the unfinished part and improving current defects due to time limits. Regarding the unfinished part, the project only concludes the general relationship of the image and the conditions of cuttlefish. In the future, more specific quantitative relations can be investigated before and after the light compensation. Regarding the improvement, the pictures fed into the image processing part are not enough and they have different sizes and image scales, making it hard to get an accurate result. Hence it is significant to apply more sets of pictures with the same criterion in the future. Moreover, the LED used only has four different light intensities due to the cost limit. A better LED with more light modes should be arranged to get more data of RGB values so that the compensation will become more reasonable.

References

[1] M. M. P. Petrou and C. Petrou, *Image processing: the fundamentals*. John Wiley & Sons, 2010.

[2] G. R. Jones, A. G. Deakin, and J. W. Spencer, *Chromatic monitoring of complex conditions*. CRC Press, 2008.

[3] H. Levkowitz and G. T. Herman, "GLHS: A Generalized Lightness, Hue, and Saturation Color Model," *CVGIP Graph. Model. Image Process.*, vol. 55, no. 4, pp. 271–285, 1993, doi: https://doi.org/10.1006/cgip.1993.1019.

[4] D. F. Rogers, *Procedural elements for computer graphics*. McGraw-Hill, Inc., 1986.

[5] A. G. Deakin, Y. R. Kolupula, A. Koh, H. M. Looe, and J. Zhang, *Characteristics of chromatic H, L, S systems*, vol. 2. chapter, 2008.

[6] R. S. Berns, *Billmeyer and Saltzman's principles of color technology*. John Wiley & Sons, 2019.

[7] L. U. Sneddon, "Pain perception in fish: indicators and endpoints.," *ILAR J.*, vol. 50, no. 4, pp. 338–342, 2009, doi: 10.1093/ilar.50.4.338.

[8] GOV.UK. 2022. *Animal welfare.* [online] Available at: <https://www.gov.uk/guidance/animal-welfare> [Accessed 6 March 2022].

Appendix A project management forms

Role allocation (responsibility matrix)

Member Name	Title(s)
Yaoming Zhang	Project manager/designer
Zhijie Yan	Implementer/Developer
Zenan Zhu	Implementer/Developer
Zhihao Yu	Assistant

Responsibility Matrix

KEY:

R – Responsible (accountable) for completion of task. (Task can be delegated to this person.)

S – Supports task.

C – Requires communication about the task.

Title	Project Activity				Deliverables			
	Requirements/ Scope	Design	Implementing	Testing	Poster	Blog	Bench	Report
Project manager	R	R	R	S	R	R	R	R
Implementer/ Developer	R	R	R	S	R	R	R	R
Assistant	S	S	S	C	C	C	R	R

Contribution to project deliverables

Member name	Deliverable(s)	Comments
Yaoming Zhang	Bench, Poster, Report, blog	Excellent
Zhijie Yan	Bench, Code, Poster, Report, blog	Excellent
Zenan Zhu	Bench, Code, Blog, Poster, Report	Excellent
Zhihao Yu	Bench, Report	good

Attendance record

Member name	Attended the weekly meeting? (Yes/No)					Comments
	Week 1	Week 2	Week 3	Week 4	Week 5	
Yaoming Zhang	Yes	Yes	Yes	Yes	Yes	Active involvement
Zhijie Yan	Yes	Yes	Yes	Yes	Yes	Active involvement
Zenan Zhu	Yes	Yes	Yes	Yes	Yes	Active involvement
Zhihao Yu	Yes	Yes	Yes	Yes	Yes	Active involvement

Week 1

Date: 2022.1.31 Supervisor: Joseph Spencer

Project Title: Emotional behavior analysis of fish based on image processing

Student Names /Attendees:	1. Zhijie Yan	2. Yaoming Zhang
3. Zenan Zhu	4. Zhihao Yu	5.

Summary of week's activities:

1. Programming: Draw a line on the fish image based on its feature (from the head of the fish to the fish's tail). All RGB values of the pixels on the line should be obtained. Use the chromatic transform to change these RGB values to HLS for further analysing.

2. Experiment: Shine the LED on papers of different colours and take photos in terms of different light intensity. Analyse the relation between RGB values and light intensity.

Problem, issues and concerns:

1. When we shine the LED on the paper, the reflection may be too intense.

2. The environment may be unstable considering we need to take at least 15 photos.

3. Ways to control the camera to reduce the automatic balance itself.

Tasks for next week/Actions for next meeting:

1. Focus on finding the features of HLS figures. The conclusion may apply to more different fish images.

2. Solve the possible photographic problems when using the LED.

Supervisor use only

Progress Assessment: □ Unsatisfactory □ Satisfactory √ Good

Comments/Recommendations:

Project completed.

Week 2

Date: 2021.2.7 Supervisor: Joseph Spencer

Project Title: Emotional behavior analysis of fish based on image processing

Student Names /Attendees:	1. Zhijie Yan	2. Yaoming Zhang
3. Zenan Zhu	4. Zhihao Yu	5.

Summary of week's activities:

1. Fish programming: We introduced three RGB filters and applied them to the first HLS transform values. The new obtained RGB values are entered to the HLS transfer function again to get the second HLS values. The procedure is repeated when t = 0, 5 and 10. The relationship curve can be plotted.

2. Primary colors experiment: We tried to solve the reflection and found the relationship between light intensity and RGB values further.

Problem, issues and concerns:

1. The relationship of HLS can hardly be concluded and represented by mathematical function since the images of the same fish under different time are few.

2. The effect may not be good enough although the primary color images are photographed under the dark environment.

Tasks for next week/Actions for next meeting:

1. Detailed relationships of HLS respectively should be concluded.

2. Apply the conclusion from RGB to the fish images and observe the results.

Supervisor use only

Progress Assessment: □ Unsatisfactory □ Satisfactory √ Good

Comments/Recommendations:

Project completed.

Week 3

Date: 2021.2.14 Supervisor: Joseph Spencer

Project Title: Emotional behavior analysis of fish based on image processing

Student Names /Attendees:	1. Zhijie Yan	2. Yaoming Zhang
3. Zenan Zhu	4. Zhihao Yu	5.

Summary of week's activities:

1. Fish inspection part: More fish images are fed into the programme to find their HLS relationship. We focus on two different states of fish, which are sham (normal) and 2%AA (with injection). Their clusters in the figures are observed.

2. Primary colour experiment: The light intensity is changed by covering with the opaque plastic foil on the LED. Considering the last week conclusion that saturation (s) is 1. We hope s can be changed to 0.5 for a better result.

Problem, issues and concerns:

1. s equals 0.5 may hard to achieve after calculation. The light intensity may have influence on RGB values but we cannot observe them clearly from the HLS.

Tasks for next week/Actions for next meeting:

Finish the combination between fish inspection part and primary colour part. The background of fish images should be detected. By using the conclusion from primary colour part, the whole fish image after a certain period (t = 5 or 10) should be compensated to return to the HLS before (t = 0).

Supervisor use only

Progress Assessment: □ Unsatisfactory □ Satisfactory √ Good

Comments/Recommendations:

Project completed.

Week 4

Date: 2022.2.21 Supervisor: Joseph Spencer

Project Title: Emotional behavior analysis of fish based on image processing

Student Names /Attendees:	1. Zhijie Yan	2. Yaoming Zhang
3. Zenan Zhu	4. Zhihao Yu	5.

Summary of week's activities:

The fish inspection part combines to the primary color experiment. The original fish image applies the conclusion and all pixels' RGB values are compensated. It is expected that after compensation the images will be clearer and the cluster accuracy will increase.

We start to prepare for the bench day. Blog, poster and presentation are inspected.

Problem, issues and concerns:

The final result may not be obvious after light compensation.

Tasks for next week/Actions for next meeting:

Show our final achievement to the supervisor.

Consider the possible direction for further research.

Supervisor use only

Progress Assessment: □ Unsatisfactory □ Satisfactory √ Good

Comments/Recommendations:

Project completed.

Appendix B breakdown of individual contributions to the project

Yaoming Zhang:

Project leader: Design project procedure, allocate works for group members and communications between groups. Date analysis of image processing part. Designer of the poster. Design the blog. Designer of bench inspection.

Report: design report structure, abstract, introduction, error analysis, whole discussion and conclusion part, appendix

Zhijie Yan

Implementer: Code and test for image processing part. Date analysis of image processing part. Code and test for final combination part. Blog and poster developer. Bench inspection speaker

Report: procedure, problem occurred part, result and analysis part, appendix

Zenan Zhu

Implementer: Taking photos, code and test for light intensity compensation part. Make assumption and test for the conclusion of light intensity compensation. Blog and poster developer. Bench inspection speaker

Report: theory, procedure, problem occurred part, result and analysis part, error analysis appendix

Zhihao Yu

Assistant: Provide help with the photo shot., Bench inspection speaker

Report: part of introduction, materials

Appendix C Python code

Cuttlefish image processing

First chromatic transform:

```
import cv2
import numpy as np
from PIL import Image
import matplotlib.pyplot as plt
from sklearn import preprocessing
from statistics import mean
import math


def hls_transfer(red, green, blue):
    rr = red - min(red, green, blue)
    gg = green - min(red, green, blue)
    bb = blue - min(red, green, blue)

    if rr == 0:
        h = 240 - 120 * gg / (gg + bb)
    if gg == 0:
        h = 360 - 120 * bb / (bb + rr)
    if bb == 0:
        h = 120 - 120 * rr / (rr + gg)

    l = (red + green + blue) / 3

    high = (max(red, green, blue) - min(red, green, blue))
    low = (max(red, green, blue) + min(red, green, blue))
    s = high / low

    return h, l, s


img = cv2.imread('/Users/DELL/Desktop/result/data/Origin/2%AA1069500/2%AA_
1069500-0.png')
a = []
b = []

# start and end point
def on_EVENT_LBUTTONDOWN(event, x, y, flags, param):
    if event == cv2.EVENT_LBUTTONDOWN:
        xy = "%d,%d" % (x, y)
        a.append(x)
        b.append(y)
        cv2.circle(img, (x, y), 1, (0, 0, 255), thickness=-1)
        cv2.putText(img, xy, (x, y), cv2.FONT_HERSHEY_PLAIN,
                    1.0, (0, 0, 0), thickness=1)
        cv2.imshow("image", img)
        # print(x, y)


cv2.namedWindow("image")
cv2.setMouseCallback("image", on_EVENT_LBUTTONDOWN)
cv2.imshow("image", img)
cv2.waitKey(0)

coordin = np.mat([[a[0],b[0]],
                  [a[1],b[1]]])
```

```python

imgg = Image.open('/Users/DELL/Desktop/result/data/Origin/2%AA1069500/2%AA_1069500-0.png')
redImage = imgg.convert('RGB')

value = []
if a[0] == a[1]:
    for yy in range(min(b[0],b[1]), max(b[0],b[1])):
        pix = list(redImage.getpixel((a[0],yy)))
        value.append(pix)

else:
    # slope of the line
    k = (b[1] - b[0]) / (a[1] - a[0])

    # rgb value
    for x in range(min(a[0], a[1]), max(a[0], a[1])+1):
        # line
        y = k * (x - a[1]) + b[1]
        pix = list(redImage.getpixel((x,y)))
        value.append(pix)

# print(value)

# draw the three lines
if a[0] == a[1]:
    x = list(range(0, abs(b[1]-b[0])))
else:
    x = list(range(0, 1+abs(a[1]-a[0])))

# original rgb
r = []
g = []
b = []

for i in value:
    r1 = i[0]
    r.append(r1)
    g1 = i[1]
    g.append(g1)
    b1 = i[2]
    b.append(b1)

# each pixel rgb value
p1 = plt.plot(x,r,'r-', label='r')
p2 = plt.plot(x,g,'g-', label='g')
p3 = plt.plot(x,b,'b-', label='b')
plt.xlabel('pixel')
plt.ylabel('RGB value')
plt.legend()
plt.grid()
plt.show()


# first chromatic transform
h = []
l = []
s = []
for m in value:
    rx = m[0]
    gx = m[1]
    bx = m[2]
    hh, ll, ss = hls_transfer(rx, gx, bx)

    h.append(hh)
    l.append(ll)
```

```
    s.append(ss)

# normalization
h = preprocessing.normalize([h])
h = [x for x in h[0]]
l = preprocessing.normalize([l])
l = [x for x in l[0]]
s = preprocessing.normalize([s])
s = [x for x in s[0]]

# print(h)
p4 = plt.plot(x,h, label='H')
p5 = plt.plot(x,l, label='L')
p6 = plt.plot(x,s, label='S')
plt.legend()
plt.xlabel('pixel')
plt.ylabel('first HLS value')
plt.grid()
plt.show()
```

second chromatic transform:

```
# second chromatic transform
mid = (len(x) - 1) / 2

# r_second
rs_s = 0
rs_h = 0
rs_l = 0
# three filters
k_rs = -1 / mid
# y_rs = k_rs * x + 1
for r_second in range(0, math.ceil(mid)):
    y_rs = k_rs * r_second + 1
    rs_s_temp = s[r_second] * y_rs
    rs_s = rs_s + rs_s_temp

    rs_h_temp = h[r_second] * y_rs
    rs_h = rs_h + rs_h_temp

    rs_l_temp = l[r_second] * y_rs
    rs_l = rs_l + rs_l_temp

# g_second
gs_s = 0
gs_h = 0
gs_l = 0

k_gs = 1 / (len(x) - 1 - mid)
# y_gs = k_gs * (x - mid)
for g_second in range(math.ceil(mid), len(x)):
    y_gs = k_gs * (g_second - mid)
    gs_s_temp = s[g_second] * y_gs
    gs_s = gs_s + gs_s_temp

    gs_h_temp = h[g_second] * y_gs
    gs_h = gs_h + gs_h_temp

    gs_l_temp = l[g_second] * y_gs
    gs_l = gs_l + gs_l_temp

# b_second
bs_s1, bs_h1, bs_l1 = 0, 0, 0
```

```python
bs_s2, bs_h2, bs_l2 = 0, 0, 0

k_bs1 = 1 / mid
# y_bs1 = k_bs1 * x
for b_second in range(0, math.ceil(mid)):
    y_bs1 = k_bs1 * b_second
    bs_s_temp = s[b_second] * y_bs1
    bs_s1 = bs_s1 + bs_s_temp

    bs_h_temp = h[b_second] * y_bs1
    bs_h1 = bs_h1 + bs_h_temp

    bs_l_temp = l[b_second] * y_bs1
    bs_l1 = bs_l1 + bs_l_temp

k_bs2 = -1 / (len(x) - 1 - mid)
# y_bs2 = k_bs2 * (x - mid) + 1
for b_second in range(math.ceil(mid), len(x)):
    y_bs2 = k_bs2 * (b_second - mid) + 1
    bs_s_temp2 = s[b_second] * y_bs2
    bs_s2 = bs_s2 + bs_s_temp2

    bs_h_temp2 = h[b_second] * y_bs2
    bs_h2 = bs_h2 + bs_h_temp2

    bs_l_temp2 = l[b_second] * y_bs2
    bs_l2 = bs_l2 + bs_l_temp2

bs_s = bs_s1 + bs_s2
bs_h = bs_h1 + bs_h2
bs_l = bs_l1 + bs_l2

h_h, l_h, s_h = hls_transfer(rs_h, gs_h, bs_h)
h_l, l_l, s_l = hls_transfer(rs_l, gs_l, bs_l)
h_s, l_s, s_s = hls_transfer(rs_s, gs_s, bs_s)

# show complete data
np.set_printoptions(suppress=True)
final = np.mat([[h_h, l_h, s_h],
                [h_l, l_l, s_l],
                [h_s, l_s, s_s]])

colors = np.array(["red"])
print(final)

# 1.relationship between h_h and h_l
x = np.array([h_h])
y = np.array([h_l])

# print(x,y)
plt.scatter(x,y,c=colors)
plt.grid()
plt.xlabel('H(H)')
plt.ylabel('H(L)')
plt.show()
```

Light intensity compensation

```python
import cv2
import numpy as np
from PIL import Image
import matplotlib.pyplot as plt

```

```python

# slope of the line
k = (162 - 160) / (260 - 250)

imgg = Image.open("/Users/Zenon/OneDrive - The University of Liverpool/桌面/F1.jpg")
redImage = imgg.convert('RGB')

# rgb value
value = []
for x in range(250, 260):
    # line
    y = k * (x - 260) + 162
    pix = list(redImage.getpixel((x, y)))
    value.append(pix)


print(value)

# original rgb
r = []
g = []
b = []

# draw the three lines
x = np.mat(list(range(1, 11))).T

for i in value:
    r1 = i[0]
    r.append(r1)
    g1 = i[1]
    g.append(g1)
    b1 = i[2]
    b.append(b1)

# each pixel rgb value
l1 = plt.plot(x, r, 'r-', label='R')
l2 = plt.plot(x, g, 'g-', label='G')
l3 = plt.plot(x, b, 'b-', label='B')
plt.legend()
plt.xlabel('Pixels')
plt.ylabel('Values')
plt.show()


def hls_transfer(red, green, blue):
    rr = red - min(red, green, blue)
    gg = green - min(red, green, blue)
    bb = blue - min(red, green, blue)

    if rr == 0:
        h = 240 - 120 * gg / (gg + bb)
    if gg == 0:
        h = 360 - 120 * bb / (bb + rr)
    if bb == 0:
        h = 120 - 120 * rr / (rr + gg)

    l = (red + green + blue) / 3

    high = (max(red, green, blue) - min(red, green, blue))
    low = (max(red, green, blue) + min(red, green, blue))
    s = high / low

    return h, l, s


```

```
h = []
l = []
s = []
for m in value:
    rx = m[0]
    gx = m[1]
    bx = m[2]
    hh, ll, ss = hls_transfer(rx, gx, bx)

    h.append(hh)
    l.append(ll)
    s.append(ss)

print(h, l, s)

l4 = plt.plot(x, h, label='H')
l5 = plt.plot(x, l, label='L')
l6 = plt.plot(x, s, label='S')
plt.legend()
plt.xlabel('Pixels')
plt.ylabel('Values')
plt.show()

Lenh = len(h)
Lens = len(s)
lenl = len(l)
sumh = 0
sums = 0
suml = 0
for i in h:
    sumh = sumh + i
avgH = sumh/Lenh

for j in s:
    sums = sums + j
avgS = sums/Lens

for m in l:
    suml = suml + m
avgl = suml/lenl

print(avgH, avgS, avgl)
```

Cuttlefish processing after light compensation:

```
import cv2
import numpy as np
from PIL import Image
import matplotlib.pyplot as plt


def second_HLS_noselection(file_path,input_x, input_y):
    # file direction
    img = cv2.imread(file_path)
```

```
    a = []
    b = []
    a = input_x
    b = input_y

    coordin = np.mat([[a[0],b[0]],
                      [a[1],b[1]]])
    print(coordin)
    # slope of the line
    k = (b[1] - b[0]) / (a[1] - a[0])

    imgg = Image.open(file_path)
    redImage = imgg.convert('RGB')

    # rgb value
    value = []
    for x in range(min(a[0], a[1]), max(a[0], a[1])+1):
        # line
        y = k * (x - a[1]) + b[1]
        pix = list(redImage.getpixel((x,y)))
        value.append(pix)

    # print(value)

    # original rgb
    r = []
    g = []
    b = []

    # draw the three lines
    x = list(range(0, 1+abs(a[1]-a[0])))

    for i in value:
        r1 = i[0]
        r.append(r1)
        g1 = i[1]
        g.append(g1)
        b1 = i[2]
        b.append(b1)

    # each pixel rgb value
    p1 = plt.plot(x,r,'r-', label='r')
    p2 = plt.plot(x,g,'g-', label='g')
    p3 = plt.plot(x,b,'b-', label='b')
    plt.legend()
    plt.grid()
    plt.show()


    def hls_transfer(red, green, blue):
        rr = red - min(red, green, blue)
        gg = green - min(red, green, blue)
        bb = blue - min(red, green, blue)

        if rr == 0:
            h = 240 - 120 * gg / (gg + bb)
        if gg == 0:
            h = 360 - 120 * bb / (bb + rr)
        if bb == 0:
            h = 120 - 120 * rr / (rr + gg)

        l = (red + green + blue) / 3

        high = (max(red, green, blue) - min(red, green, blue))
        low = (max(red, green, blue) + min(red, green, blue))
        s = high / low
```

```python

        return h, l, s

    # first chromatic transform
    h = []
    l = []
    s = []
    for m in value:
        rx = m[0]
        gx = m[1]
        bx = m[2]
        hh, ll, ss = hls_transfer(rx, gx, bx)

        h.append(hh)
        l.append(ll)
        s.append(ss)

    p4 = plt.plot(x,h, label='H')
    p5 = plt.plot(x,l, label='L')
    p6 = plt.plot(x,s, label='S')
    plt.legend()
    plt.grid()
    plt.show()

    # sum
    Lenh = len(h)
    Lens = len(s)
    lenl = len(l)
    sumh = 0
    sums = 0
    suml = 0
    for i in h:
        sumh = sumh + i
    avgh = sumh / Lenh
    for j in s:
        sums = sums + j
    avgs = sums / Lens
    for m in l:
        suml = suml + m
    avgl = suml / lenl

    return avgl

def changeRGB1(file_path):
    omb = 0.12
    omg = 0.045

    imgg = Image.open(file_path)
    mage = imgg.convert('RGB')
    array = np.array(mage)
    cols,rows = mage.size

    red = []
    green = []
    blue = []
    for i in range(cols):
        for j in range(rows):
            pix = mage.getpixel((i,j))
            bb = pix[2] + pix[2] * omb * (14/10)

            blue.append(bb)
            gg = pix[1] + pix[1] * omg * (14/10)
            green.append(gg)
            red.append(pix[0])
    fina = []
    for o in range(len(red)):
```

```python
        fin = red[o],green[o],blue[o]
        tempp = list(fin)
        fina.append(tempp)

    f = 0
    while f < len(fina):
        for p in range(cols):
            for q in range(rows):
                array[q,p] = fina[f]
                f += 1
    new_img = Image.fromarray(array)
    new_img.show()
    # new_img.save("/Users/DELL/Desktop/result/sham_1068625_0_after.png")

def changeRGB2(file_path):
    omb = 0.12
    omg = 0.045

    imgg = Image.open(file_path)
    mage = imgg.convert('RGB')
    array = np.array(mage)
    cols, rows = mage.size

    red = []
    green = []
    blue = []
    for i in range(cols):
        for j in range(rows):
            pix = mage.getpixel((i,j))
            bb = pix[2] - pix[2] * omb

            blue.append(bb)
            gg = pix[1] - pix[1] * omg
            green.append(gg)
            red.append(pix[0])
    fina = []
    for o in range(len(red)):
        fin = red[o],green[o],blue[o]
        tempp = list(fin)
        fina.append(tempp)

    f = 0
    while f < len(fina):
        for p in range(cols):
            for q in range(rows):
                array[q,p] = fina[f]
                f += 1
    new_img = Image.fromarray(array)
    new_img.show()
    # new_img.save("/Users/DELL/Desktop/result/sham_1068625_10_after.png")
```

```python
import numpy as np

x1 = [30.16, 45.9, 54.6]

y1 = [189.14, 191.64, 193.63]
x3 = [30.16, 45.9, 54.6]
y3 = [189.14, 191.53, 193.63]
plt.figure()
plt.xlabel('Light Condition')
plt.ylabel('H')
plt.plot(x1, y1, '-o', color='red')
plt.plot(x3, y3, '-o', color='blue')

plt.show()
```

```

x2 = [30.16, 45.9, 54.6]
y2 = [0.63, 0.55, 0.53]
x4 = [30.16, 45.9, 54.6]
y4 = [0.63, 0.57, 0.53]
plt.figure()
plt.xlabel('Light Condition')
plt.ylabel('S')
plt.plot(x2, y2, '-o', color='blue')
plt.plot(x4, y4, '-o', color='red')
plt.show()
```