EMS420U LAB REPORT

Microscopic image analysis

Essa Raby Youssouf [210230071]
Kirijan Nalvelnathan[210229949]
Manvydas Andriekus [210232857]
Senthur Sarveswaran [210926565]
Yunus Khan [210229927]

Introduction

The purpose of the experiment is to obtain microscopic images of the blood samples of fish, frogs, and birds and to identify them using a light microscope. These images are then used to measure the different nuclear areas and shapes of the animals using the coding software Python, which is also used to compare the three different animal nuclear areas statistically (through box plots). Most importantly the aim is to build upon microscopy skills learnt during the first semester, for more proficiency in the usage of microscopes.

Method

Apparatus

-Microscope

-Ruler (0.01mm resolution)

In-lab Procedure

The pixel size was identified by taking a screenshot of the 1mm ruler which can be used in processing to find the size of each pixel in standard units. The Microscope was calibrated such that any images we take, have a legend showing the scale of the images. Images of bird, fish, and frog blood were taken at 20x magnification. 5 images were taken of each sample. Images were kept track of by writing down the order in which the images were taken, and the sample it was taken from. The images were saved to a USB drive and renamed for convenience in processing.

Post-lab Procedure

Once the data had been attained, it had to be run through a Python program which would analyse the images and find the nucleus of each cell as well as determine its perimeter, area, solidity, and eccentricity. Multiple libraries were first imported [1] followed by the path to the images being defined [2]. The images were then read to the program in turn [3] and processed through a method of trial and error. The initial step was to determine an accurate factor to multiply by the threshold, which was obtained by using the yen algorithm [4]. The factor was generally between 0.2 and 0.6; this affected how much of the cell was considered as the nucleus, starting with the lowest possible value, and working up in increments of 0.1 until the processed image showed most of the nuclei being selected and not more than what was necessary [5]. The next factors that required altering were the masks, one to remove small objects [6] and the other to remove small holes [7]. The small holes mask removes holes smaller than a specified size, which in this case usually remained at 5000. The small objects mask removes objects which are smaller than the specified size, this ranged from 100 to 700, and initially tested from 100 and with each incrementation, this value was increased by 100 until an optimum image was obtained whereby the output image had all the nuclei selected and had minimal noise.

The data taken from these images were then processed through another Python program, which analysed the data by applying various methodologies.
After importing the necessary libraries and reading the comma-separated files, arrays corresponding to the set animals' blood i.e., bird; fish and frog, were created [8] and each file

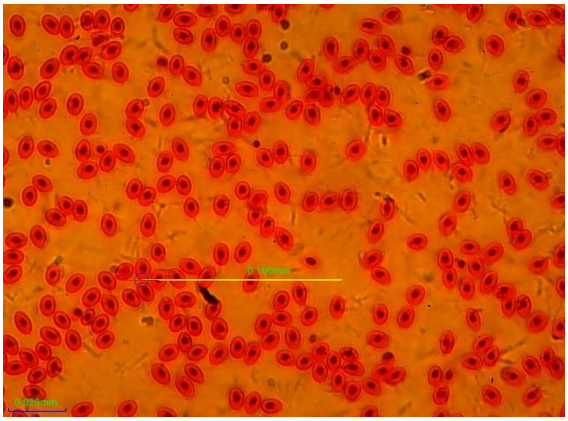
was concatenated to the named array. Another set of arrays was created which were named after each parameter of the data: area; perimeter; eccentricity; and solidity [9].
A conversion factor was required for converting the measured area and perimeter values' units from pixels to micrometres. This number was obtained by placing an image of a stage micrometre under the microscope [10] through ImageJ, an application for processing and analysing scientific images. The given conversion factor was then multiplied by each value in the array [11].
Boxplots were then produced, which can be found in Figure 4.

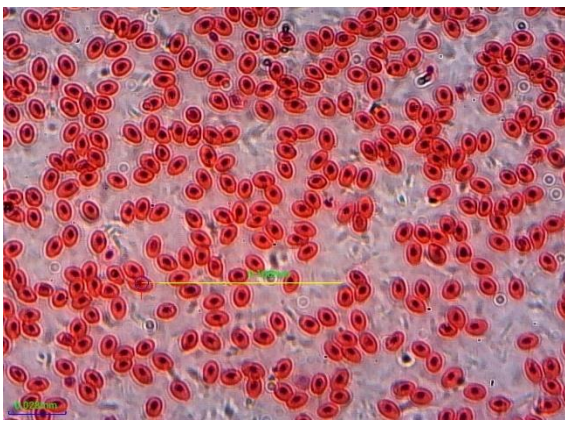
2 tests were run to analyse the data: an ANOVA test and a post-hoc test; of which results can be found in Figure 5 and Figure 6, respectively.

Results

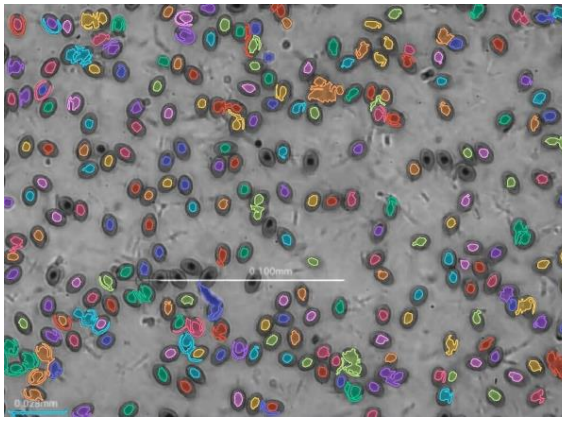
Bird blood:

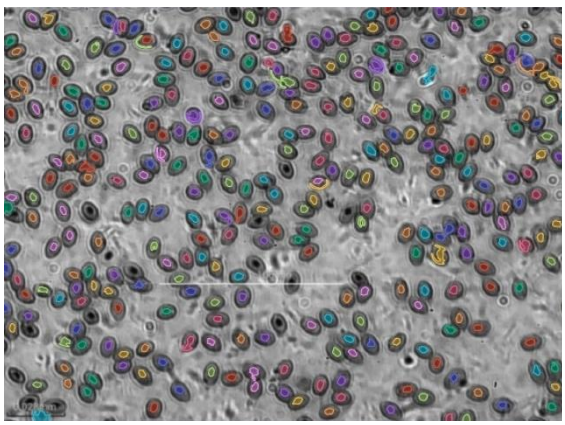
Raw images:





Overlay images:





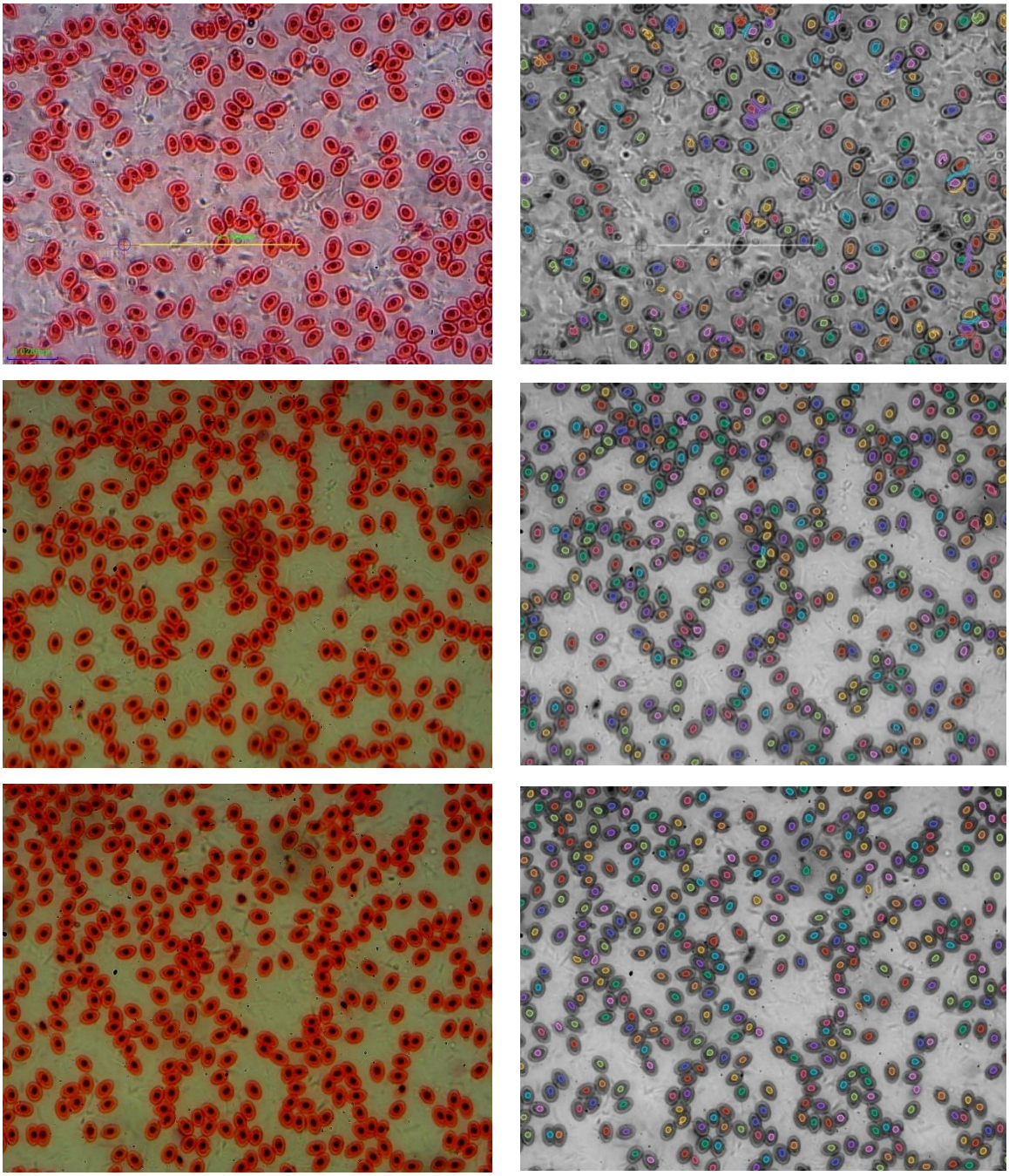
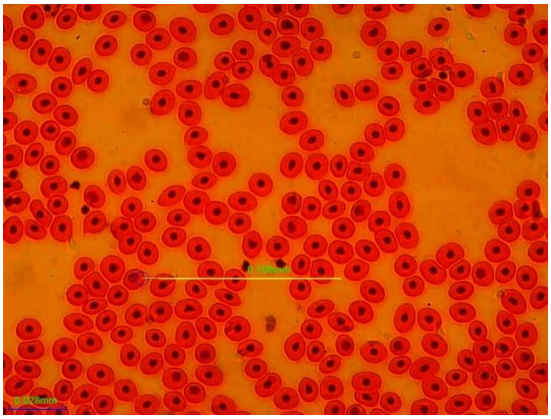


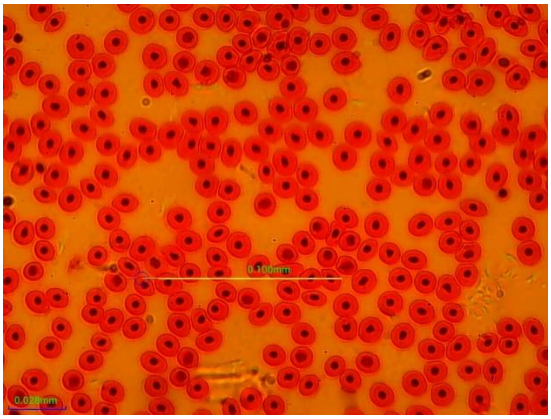
Figure 1. A set of images showing the blood smear of a bird, and the analysed counterpart.

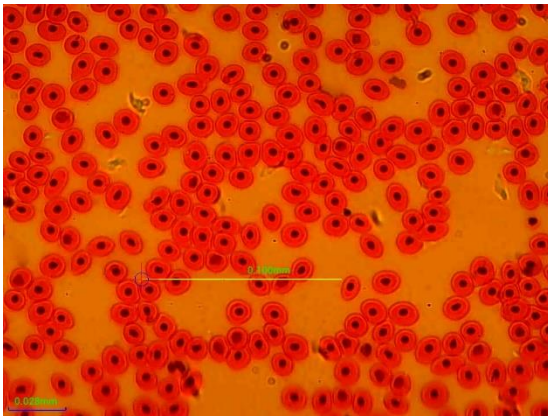
As shown in the images, the cells are similar in size but differ in terms of how they cluster together. However, the overlay images are not the most ideal as they miss some cells although this is as close to perfection as reachable with the technology that was provided.

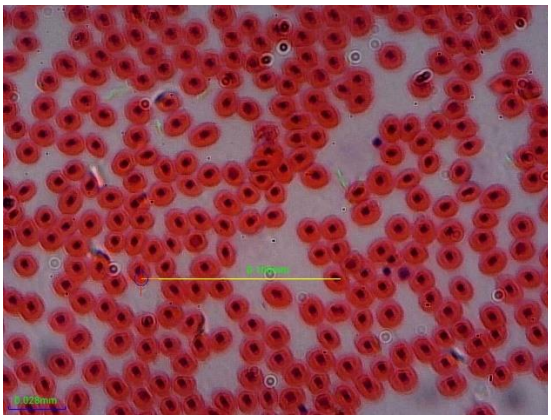
Fish blood:

Raw images:

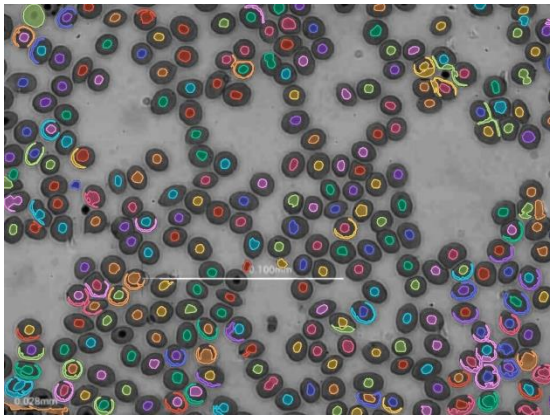


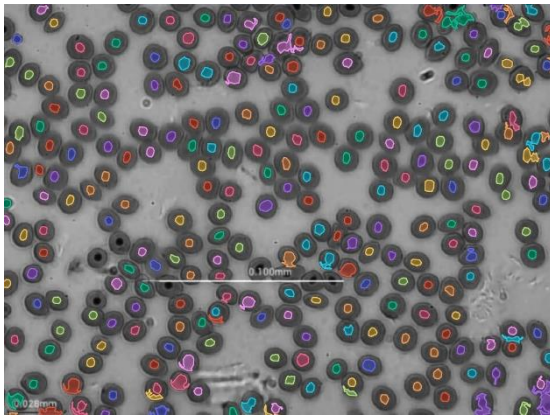


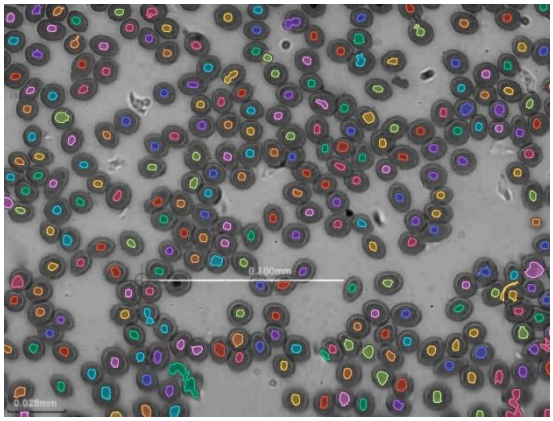


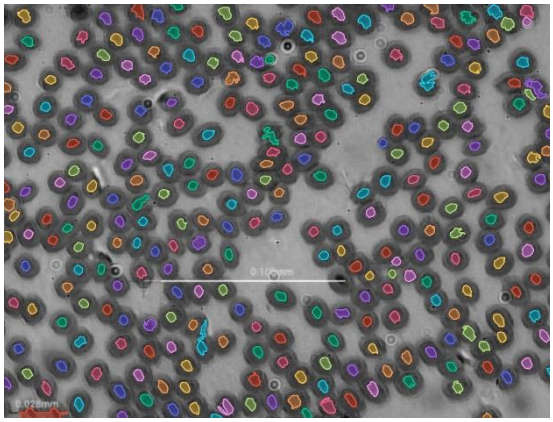


Overlay images:









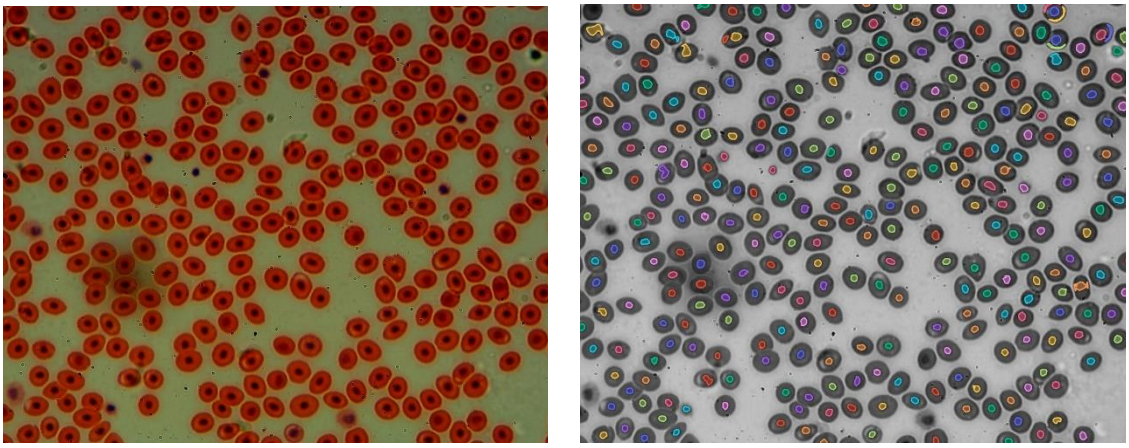
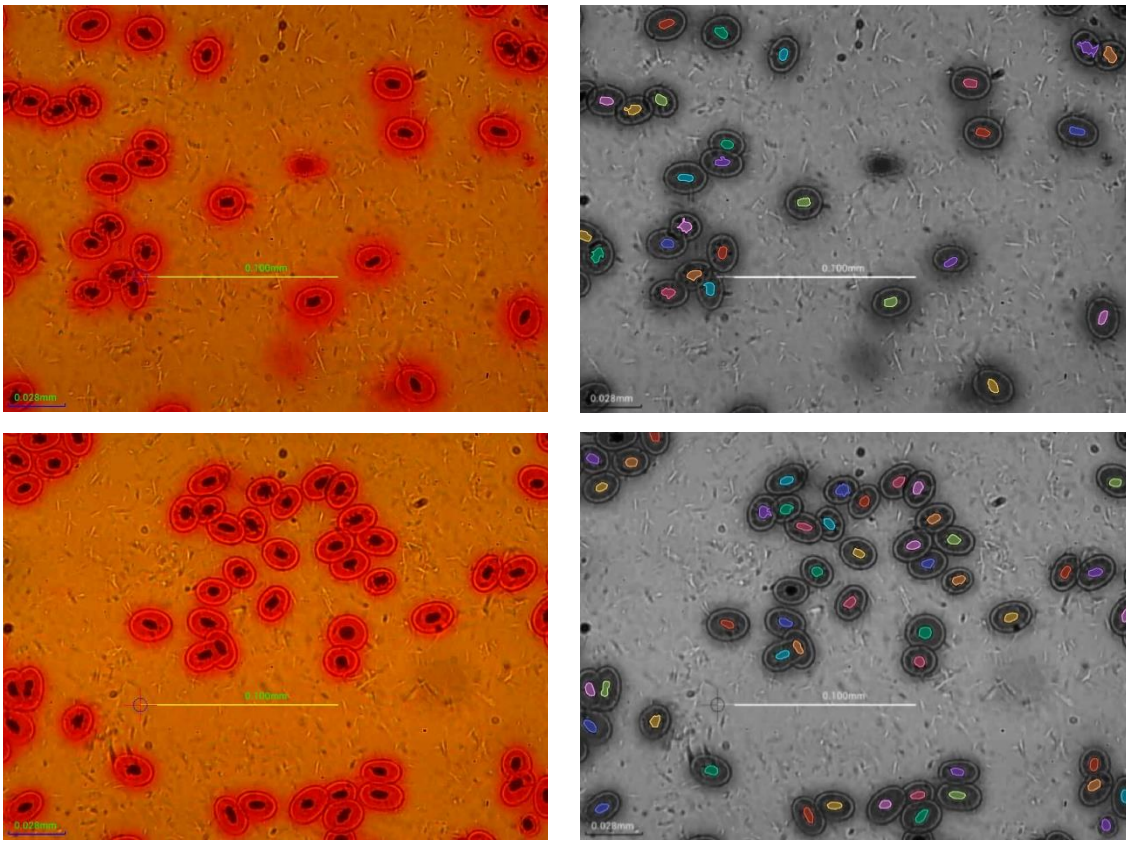


Figure 2. A set of images showing the blood smear of a fish, and the analysed counterpart.

Except for the first image, which has several anomalies, the sizes of the cells were consistent and there were a lot more cells clustered together compared to the other animals' red blood cells.

Frog blood:

Raw images: Overlay images:



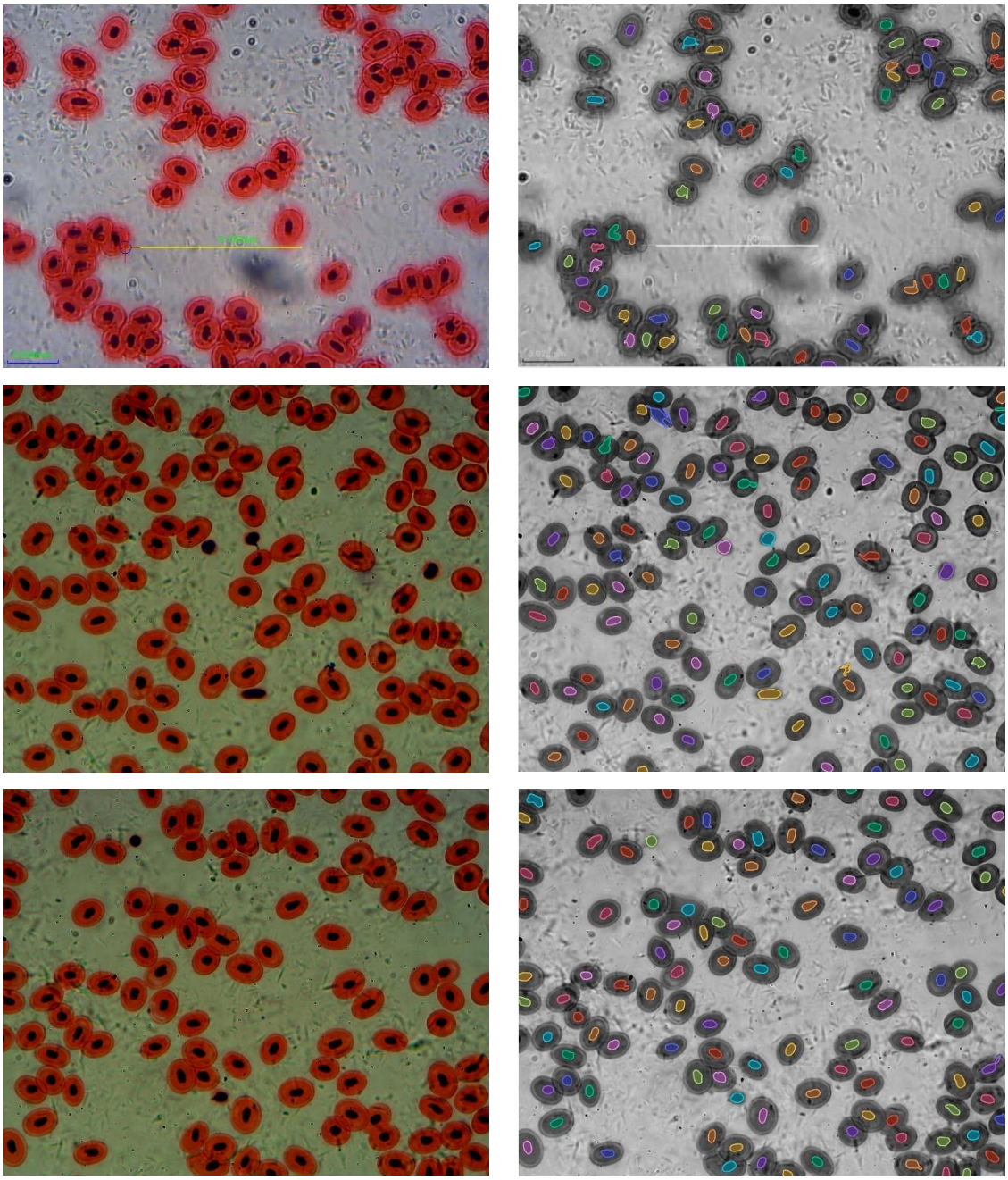


Figure 3. A set of images showing the blood smear of a frog, and the analysed counterpart.

As shown in the figure above, frogs' blood cells are significantly larger than other animals. This resulted in fewer data points being taken and may result in a less accurate result.

Boxplots:

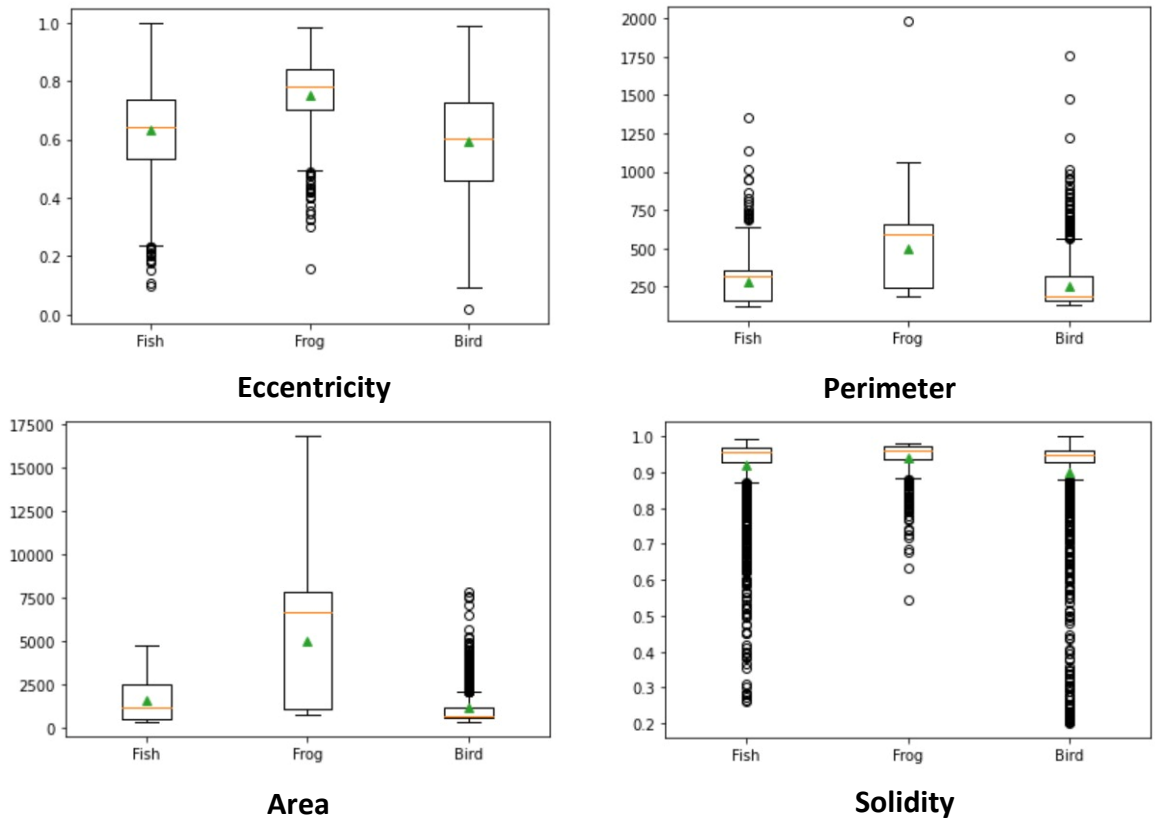


Figure 4. A set of boxplots showing the spread of data across 4 different conditions

As displayed in Figure 4, there is a large amount of anomalous data, especially in the solidity of the cells. However, this is most likely due to the program identifying small holes/ objects as cells, rather than actual cells having a solidity less than 0.9, therefore, it would make sense to disregard the anomalous data as these are most likely not actual blood cells.

Data tables:

BIRD	Area	Perimeter	Eccentricity	Solidity
Mean	449.993	79.735	0.631	0.920
Median	337	90.426	0.641	0.95
Standard Deviation	307.144	35.383	0.149	0.103
Count	1557	1557	1557	1557

Table 1. A table showing statistical data for the bird variant

FISH	Area	Perimeter	Eccentricity	Solidity
Mean	336.976	72.413	0.591	0.901
Median	201	54.749	0.603	0.948
Standard Deviation	289.882	40.923	0.179	0.145
Count	1476	1476	1476	1476

Table 2. A table showing statistical data for the fish variant

FROG	Area	Perimeter	Eccentricity	Solidity
Mean	1430.028	141.088	0.749	0.939
Median	1902.5	169.004	0.782	0.960
Standard Deviation	989.665	63.919	0.129	0.057
Count	390	390	390	390

Table 3. A table showing statistical data for the frog variant

ANOVA Values	Statistic (F)	P-value
Areas	913.0488622061338	$1.44764e^{-318}$
Perimeter	375.3168696553395	$4.122257214019081e^{-148}$
Eccentricity	150.18427839285366	$3.007174289967383e^{-63}$
Solidity	20.148633852191818	$1.9981204566121405e^{-9}$

Figure 5. A set of values showing the ANOVA for all blood types

As observable from Figure 5, the higher the F value returned from running an ANOVA test the higher the variation between sample means relative to the variation within the samples. From this, it can be judged that the area within each image varies quite significantly whereas, on the other hand, the solidity of the cells ranges less.

Post-hoc testing results:

AREAS	FISH	FROG	BIRD
FISH	1.000000	0.0	0.002339
FROG	0.000000	1.0	0.000000
BIRD	0.002339	0.0	1.000000

PERIMETER	FISH	FROG	BIRD
FISH	1.000000	0.0	0.000127
FROG	0.000000	1.0	0.000000
BIRD	0.000127	0.0	1.000000

ECCENTRICITY	FISH	FROG	BIRD
FISH	1.000000	0.0	$3.040997e^{-8}$
FROG	0.000000	1.0	0.000000
BIRD	$3.040997e^{-8}$	0.0	1.000000

SOLIDITY	FISH	FROG	BIRD
FISH	1.000000	0.000651	$5.129230e^{-14}$
FROG	$6.506290e^{-4}$	1.000000	$6.506290e^{-4}$
BIRD	$5.129230e^{-14}$	0.000000	1.000000

Figure 6. A set of data showing the post-hoc results

Shown in Figure 6 is the results from performing a Dunn's post hoc test. This is a non-parametric pairwise multiple comparisons procedure based on rank sums, often used following a rejection of an ANOVA test.

Conclusion

The boxplots computed show, for eccentricity and perimeter, the upper-lower quartile ranges for all blood types overlap, which suggests statistical significance. This is exceedingly so for solidity in which the boxplots all range from 0.7-1, however, they all have an alarmingly high number of abnormalities (outliers) which suggests issues with the overlay images/data and the programming for them. For area, the fish and bird blood were incredibly similar in terms of median and middle values however the birds' blood was the only sample with outliers, with the frog blood having an unusually high distribution in comparison to the two.

Appendix

[1]
```
#import libraries
import os #for changing path, etc
import imageio #for reading image files
import matplotlib.pyplot as plt #for displaying figure/image data
import pandas as pd #for data handling/analysis
import plotly #for interactive plots
import plotly.express as px #for displaying data on overlay
import plotly.graph_objects as go #for intearctive plots
import scipy.stats as ss # for statistical testing
import scikit_posthocs as sp # for posthoc testing
#import functions
from skimage import measure, morphology
from skimage.color import rgb2gray
from skimage.filters import (gaussian, threshold_yen)
from skimage.measure import regionprops_table
```

[2]

```
#Define Path and change directory
path = 'C:/Users/Yunus/Documents/AAA Uni/Experimental Design and Practise/6) Image analysis microscope'
os.chdir(path)
```

[3]

```
imagename = 'fish1'  #change here the file name
#read image - change image name here!
image = imageio.imread(imagename+'.jpg')
```

[4]

```
#image processing - convert to greyscale and perform gaussian filtering
img = rgb2gray(image)
img= gaussian(img, sigma=1)
#Show image
plt.imshow(img, cmap='gray')
# Binary image, post-process the binary mask and compute labels; change parameters here as needed
block_size = 51 # this is the size of the window for adaptive thresholding
threshold = threshold_yen(img, block_size) #change here the algorithm
```

[5]

```
mask = img < threshold*0.3 #change here the factor
#especially change here the parameters below for erosion and removing objects; comment out if not needed
#mask = morphology.erosion(mask)
```

[6]

```
mask = morphology.remove_small_objects(mask, 400)

```

[7]

```
mask = morphology.remove_small_holes(mask, 5000)
plt.imshow(mask, cmap='gray')
```

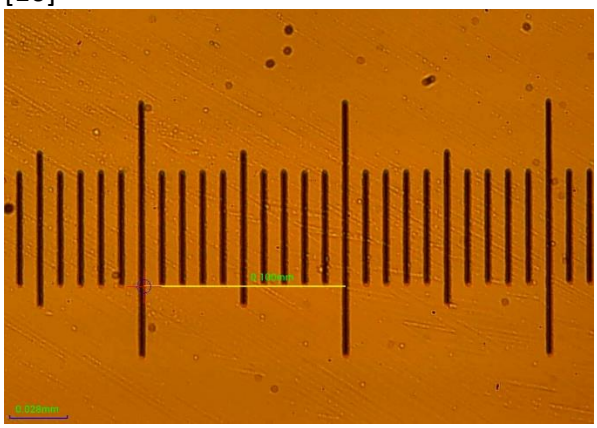
[8]

```
allBird=pd.concat([df1,df2,df3,df4,df5])
allFish=pd.concat([df6,df7,df8,df9,df10])
allFrog=pd.concat([df11,df12,df13,df14,df15])
```

[9]

```
Areas=[allFish["area"],allFrog["area"],allBird["area"]]
Perimeter=[allFish["perimeter"],allFrog["perimeter"],allBird["perimeter"]]
Eccentricities=[allFish["eccentricity"],allFrog["eccentricity"],allBird["eccentricity"]]
Solidity=[allFish["solidity"],allFrog["solidity"],allBird["solidity"]]
```

[10]



[11]

```
11 factor = 3.5
12
13 Areas_µm2 = [value * (factor**2) for value in Areas] #
14 Perimeter_µm = [value * factor for value in Perimeter]
```