Team:16
Version:1
Date:08/06/2023

Experimental Validation Form

I. **Protocol Title:** Evaluation for Validating the Effectiveness of Normalization Technique for RGB Values in Image Processing.

II. **Hypothesis:** Hypothesize that the normalization process effectively scales and adjusts the overall RGB values of the test strip images, ensuring consistency and comparability between different images.

III. **Types of Data:** The Type of Data used in the evaluation includes RGB values obtained from image processing. The RGB values represent the level of intensity of the color red, green, and blue color channels and later standardize those values by performing a normalization technique.

IV. **Controls:** Appropriate controls are used to ensure the validity of the normalization technique. This includes the obtained and known reference images whose RGB values are known and can be validated during the normalization process.

V. **Number of Samples:** A total of 6 test strip images are samples that will be used to extract data and perform data analysis from the results. Creating 3 Test cases.

VI. **Brief Description of Protocol:** RGB calculation and bar plot verification protocol aims to ensure the efficiency and accuracy of RGB calculation and generation of bar plots for the test strips. This EVP outlines the validation process that is used for the normalization technique applied to RGB values obtained from test strip images. The EVP goes into the detailed steps involved in normalization, and the hypothesis to be tested. Further materials/software are required for the procedure for normalization and data analysis, experimental justification, and the limitations. Two test cases are been shown for comparison of normalized RGB values.

VII. **Experimental Justification:**

1. Normalization is a critical step in RGB data analysis as it ensures consistency and comparability across different test strip regions and also across different test strip images. By running a validation for the normalization process we can measure the accuracy and reliability of the obtained data. This also enhances the data analysis leading to more accurate, robust, and meaningful results.
2. Further, Normalization aims to ensure consistency and give a standardized RGB value across images. Up to a greater extent, it also takes into account variations in lighting conditions camera settings, and image quality. This also ensures that obtained values are comparable across images.
3. Normalization will not eliminate but surely minimize biases such as lighting variations providing more accurate and reliable results.

4. By performing Normalizaion it ensures a deep commitment to quality assurance in the data analysis process. Guaranteeing that results are robust, accurate, and reproducible which makes the findings very reliable.

VIII. Experimental Limitations

1. The effectiveness of normalization may change depending on the quality and characteristics of the test strip images. This impact can be seen during variations in capturing images under different lighting conditions.
2. Variability in the test strips can cause uncertainty in results and this could influence the measurements or outcomes.
3. The performance of the test strip can be influenced by reagent sensitivity and stability. Factors such as variations in reagent quality or storage conditions might also affect the test results.
4. Image Analysis Accuracy and pixel calculator can be influenced by the lighting condition, resolution, and algorithm limitation (linear scale). This may introduce the inherent limitations in the normalization process.
5. The sample size was small and limited and due to this statistical power and reliability of the results are restricted to the performed experiments.
6. Some other limitations are the lack of a control group, assumptions, equipment limitations, and human error. Furthermore, image resolution and noise may also influence the outcome of the normalization process

Team:16
Version:1
Date:08/06/2023

I. **Protocol Title:** Evaluation for Validating the Effectiveness of Normalization Technique for RGB Values in Image Processing.

II. **Hypothesis:** Hypothesize that the normalization process effectively scales and adjusts the overall RGB values of the test strip images, ensuring consistency and comparability between different images.

   A. **Null Hypothesis(H0):** There is no significant color change or effect on RGB values of images due to normalization.
   B. **Alternative Hypothesis(Ha):** There is a significant effect on the RGB values of the image due to normalization.

III. **Materials:**

- Test strip images saturated with modified Scotts Reagent
- Alcohol Solution containing Benedryl (substitute for ketamine)
- Smartphone Camera or imaging device for capturing test strip images
- Anaconda: Python programming environment
- Jupyter Notebook: Data Analysis and Visualization
- Required Python Libraries: os, cv2, numpy, matplotlib.pyplot

IV. **Procedure:**

A. **RGB Calculation:**

1. Star by setting the working directory to the folder that has all the reaction test strip images.
2. Import the required/necessary libraries such as OpenCV, NumPy, and Matplotlib.
3. Load the images using the cv2.imread() function and store them in separate variables.
4. Now compute the average pixel values for Red, Green, and Blue color channels in each image using cv2.mean() and store them in separate variables.
5. Convert the average pixel values to a list and Using the round() function round them to two decimal places.
6. Print the average pixel values for each color channel for each image.

B. **Normalization:**

1. The obtained RGB values above from image 1 and image 2 are now defined.
2. Now Calculate the average RGB values from each image by adding (sum) the RGB values and dividing by 3
3. Normalize the average RGB values by dividing them by 255 (255 is the maximum RGB value)
4. Print the normalized values to verify the results.
5. Libraries such as numpy and matplotlib.plot needs to be imported for data visualization.
6. Create a list of labels to identify the images. (e.g: “Non-test region of test strip” and “Tested Region of test strip”)
7. Create a list of normalized RGB values obtained from step 3.

Team:16
Version:1
Date:08/06/2023

8. Set the positions and width of the bars using the numpy arange function.
9. Create the figure and axis using the matplotlib.pyplot's subplots function.
10. Now use the matplotli.pyplot bar function to create the bar plot, specifying the positions, normalized values, width, color, and alpha.
11. Set the axis labels and title of the bar graph to provide meaningful information for the plot.
12. Set the axis ticks and tick labels using the earlier defined label tags and position.
13. Add annotation to the bar graph by putting the values on top of each bar to display the precise normalized RGB values.
14. Finally, display the plot using matplotli.pyplot show function.

C. Example Code Snippets:

The following code is for calculating the RGB values and then followed up by the code for normalized RGB values and the normalized bar plot will be in the Results section

```
import cv2

# Load the two images
img1 = cv2.imread('I2.png') #I2 is the second Image taken (Non-tested)
img2 = cv2.imread('Ben&Kombu_TS1.png') #(Tested Image)

print("Read") #Checkpoint
import numpy as np #import numerical python library for calculation

#Calculate the Average Pixel Values for color channels
mean1=cv2.mean(img1)
mean1=[round(val,2) for val in mean1[:3]]

mean2=cv2.mean(img2)
mean2=[round(val,2) for val in mean2[:3]]

#Print the Value of the R,G,B channels
print("Below we print the average pixel values of R, G, B component in image-1")
print(mean1[:3])

print("Below we print the average pixel values of R, G, B component in image-2")
print(mean2[:3])
```

```
Read
Below we print the average pixel values of R, G, B component in image-1
[126.43, 129.0, 135.02]
Below we print the average pixel values of R, G, B component in image-2
[127.57, 115.77, 86.97]
```

1. Explanation and Justification of RGB Values and Subtraction for Visual Representation

a. RGB values represent the intensity of red, green, and blue colors in an image. Each color channel has a value from 0 to 255, where 0 is the absence of that color and 255 represents the maximum intensity of that color.

Team:16
Version:1
Date:08/06/2023

b. In relation to image analysis, darker color tends to have lower RGB values because they absorb more light and brighter colors have higher RGB values as they reflect more light, while brighter colors have a much higher RGB value as they reflect more light. So as the color gets darker the RGB value decreases.
c. When plotting Bar graphs to represent the RGB values it is challenging to interpret results directly as color intensity is inversely related to RGB values. To resolve this issue and improve visual representation a simple transformation can be applied.
d. We subtracted the RGB values from 255 to invert the color intensity and now the darker colors get the higher values and the brighter colors get a lower value. This is better for a more intuitive interpretation of the bar graph, as higher values indicate a more intense color, and lower values indicate a less intense color.
e. This makes it easier to compare and draw conclusions based on the relative intensities of different colors in the test strip images.

Note: This transformation does not alter the relative difference between the colors; it simply enhances the visual representation for better interpretation.

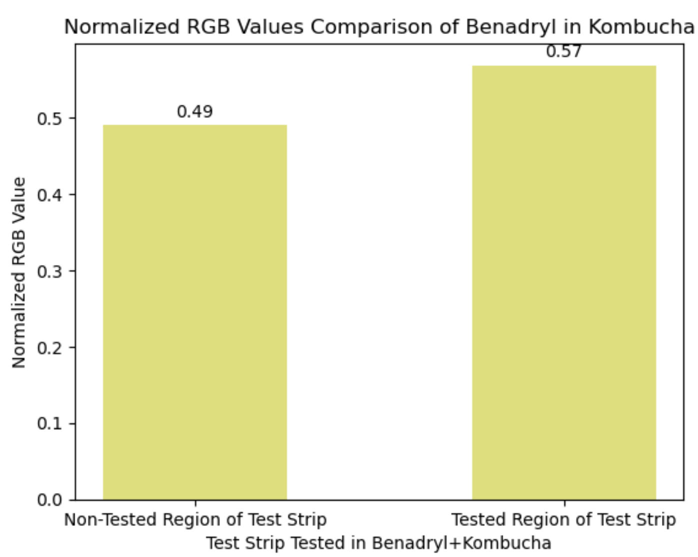
```python
# RGB values for image1 and image2
rgb_image3 = (128.57, 126, 119.98)
rgb_image4 = (127.43, 139.3, 168.03)

# Calculate the average RGB value for each image
average_rgb_image1 = sum(rgb_image3) / 3
average_rgb_image2 = sum(rgb_image4) / 3

# Normalize the average RGB values
normalized_value_image1 = average_rgb_image1 / 255
normalized_value_image2 = average_rgb_image2 / 255

# Print the normalized values
print("Normalized value for image1:", normalized_value_image1)
print("Normalized value for image2:", normalized_value_image2)
```

```
Normalized value for image1: 0.4896078431372549
Normalized value for image2: 0.568313725490196
```

Team:16
Version:1
Date:08/06/2023

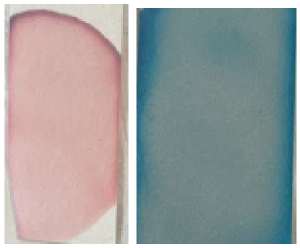
```python
import numpy as np
import matplotlib.pyplot as plt

# Normalized RGB values for image1 and image2
normalized_value_image1 = 0.4896078431372549
normalized_value_image2 = 0.568313725490196
# Create a list of labels for the images
labels = ['Non-Tested Region of Test Strip', 'Tested Region of Test Strip']

# Create a list of the normalized RGB values
normalized_values = [normalized_value_image1, normalized_value_image2]

# Set the positions and width of the bars
pos = np.arange(len(labels))
width = 0.5

# Create the figure and axis objects
fig, ax = plt.subplots()

# Create the bar plot
rects = ax.bar(pos, normalized_values, width, color='y', alpha=0.5)

# Set the axis labels and title
ax.set_xlabel('Test Strip Tested in Benadryl+Kombucha')
ax.set_ylabel('Normalized RGB Value')
ax.set_title('Normalized RGB Values Comparison of Benadryl in Kombucha')

# Set the axis ticks and tick labels
ax.set_xticks(pos)
ax.set_xticklabels(labels)

# Add the values on top of each bar
for rect in rects:
    height = rect.get_height()
    ax.annotate(f'{height:.2f}', xy=(rect.get_x() + rect.get_width() / 2, height), xytext=(0, 3),
                textcoords="offset points", ha='center', va='bottom')

# Display the plot
plt.show()
```

Output:

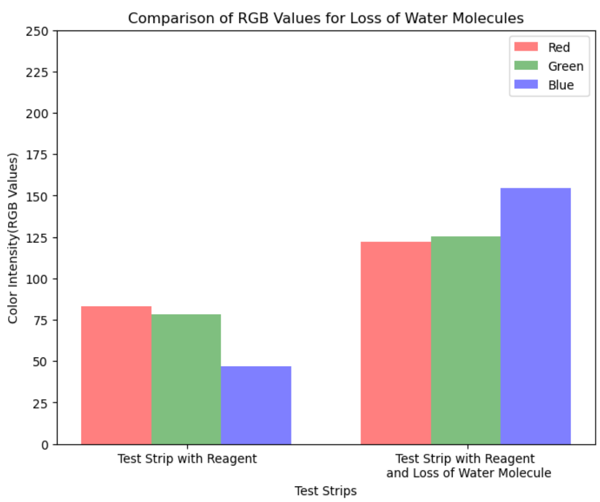


Team:16
Version:1
Date:08/06/2023
V. Result and Data Analysis:

Test Case 1:



Bar Graph Plot Figure 1a on the Left is before normalization and Bar Graph Plot Figure 1b on the Right is after normalization:



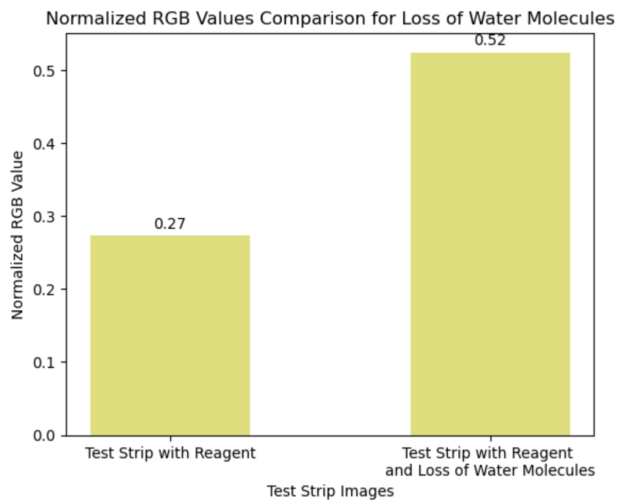


Fig 1a(on the LEFT) is the calculated RGB values that are visually represented by each color channel Red, Green, and Blue. Fig1a is a comparison of the Test Strip with Just Reagent and after 30mins of Reagent with loss of water molecules. Fig 1b. (on the RIGHT) is the Normalized calculated RGB values. It is the overall trend that there is a substantial rise of RGB values as there is a loss of water molecules. Here the the Test Strip with Regent is 0.27 and the test strip with Loss of water molecules is 0.57 which shows an upward trend.

Team:16
Version:1
Date:08/06/2023
Test Case 2:

Figure 2a

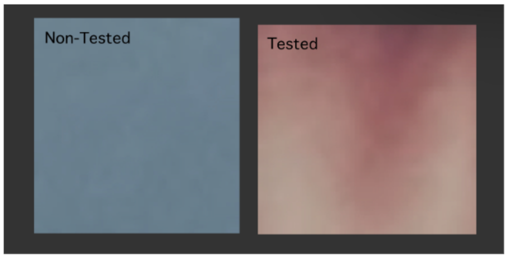


Figure 2b.

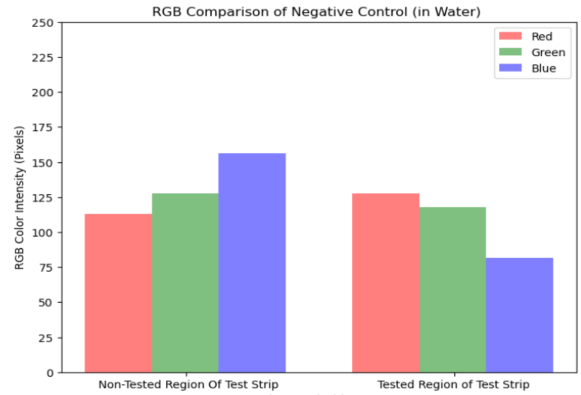
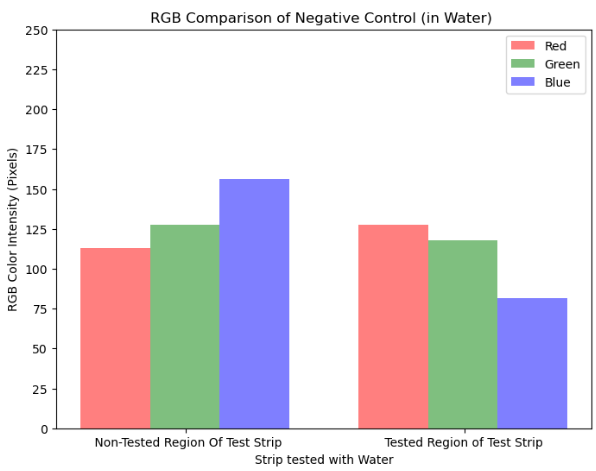


Figure 2a. Color change in the presence of diphenhydramine. Non-tested is darker blue due to a loss of coordinated water molecules. This pink test strip was rehydrated.
Figure 2b. RGB color intensity between the non-tested test strip and the tested strip.



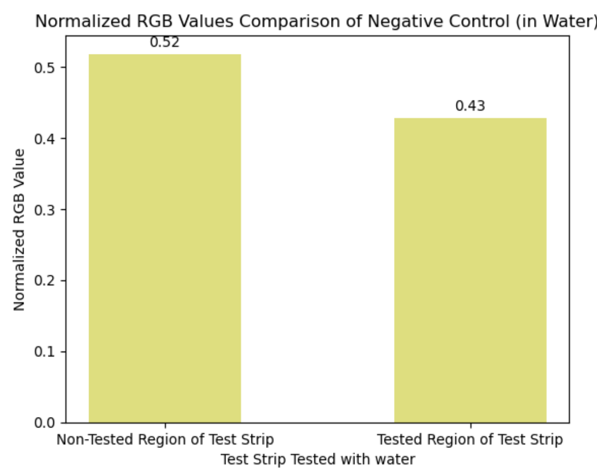


Fig 2a(on the LEFT) is the calculated RGB values that are visually represented by each color channel Red, Green, and Blue. Fig2a is a comparison of the Test Strip with Just Reagent and after 30mins of Reagent with loss of water molecules. In reality, the Non-tested has higher blue because of the loss of water molecules and not because of testing. Fig 2c. (on the RIGHT) is the Normalized calculated RGB values. Here the non-tested region of the Test Strip is 0.52 and the test region of the test strip is 0.43 which shows a downward trend.

Team:16
Version:1
Date:08/06/2023
Test Case 3:

Figure 3a

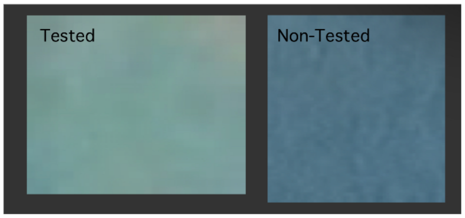


Figure 3b

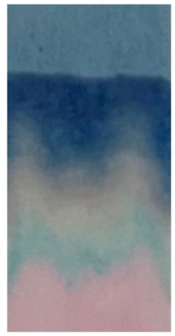


Figure 3c

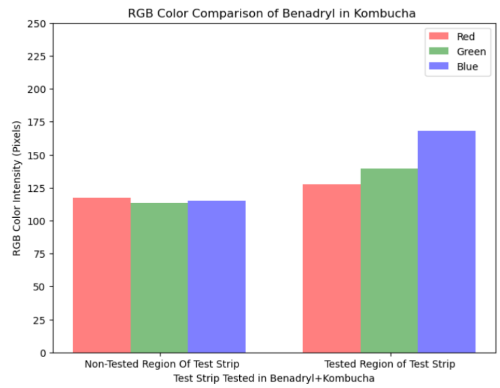
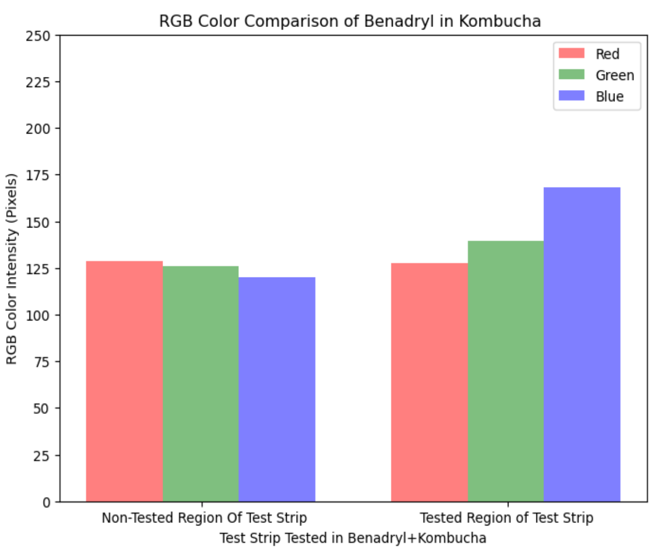


Figure 3a. The left test strip displays a blue /teal color change caused by 40% alcohol containing 2% benadryl concentration. Right test strip demonstrates a darker blue color on a non-tested region with loss of coordinated water molecules.**Figure 3b** RGB color intensity between the non-tested strip and the tested strip 5 distinct colors from top to bottom: non tested loss of water molecules (blue), (darker blue) how high the solution reached, (white) separation between positive result and non testing region, (bright blue/teal) positive result, (pink) rehydration of reagent
Figure 3c. RGB color intensity between the non-tested strip and the tested strip.



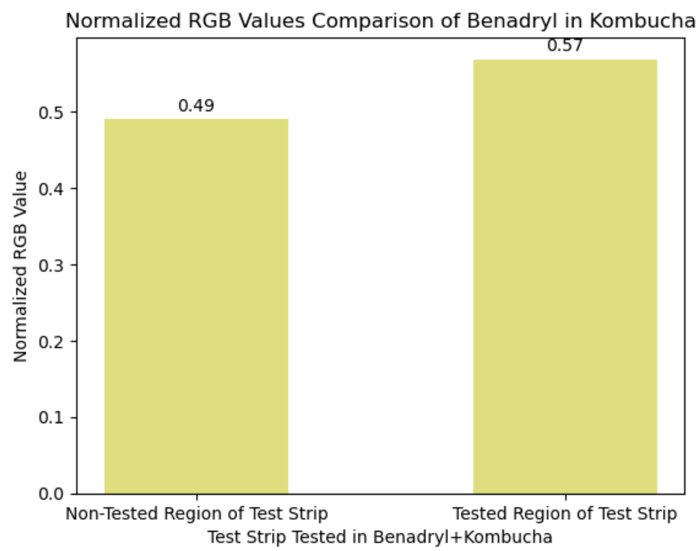


Fig 3c(on the LEFT) is the calculated RGB values that are visually represented by each color channel Red, Green, and Blue. There is a slight rise of the color green, and a significant rise of the color blue particularly on the tested region which indicates positive i.e. the drug Benedryl is present in the drink. Fig 3d. (on the RIGHT) is the Normalized calculated RGB values. It is the overall trend. Here the non-tested region of the Test Strip is 0.49 and the test region of the test strip is 0.57 which shows an upward trend indicating a positive result (The rise indicates benedryl is present and successfully detected).

Team:16
Version:1
Date:08/06/2023

VI. Future Work/Plan

1. More testing with more research leads to improvements that can address the above-mentioned limitation and enhance the robustness of the findings/results.
2. A robust Machine learning approach can be taken if more data is available. Such as Convolutional Neural Networks (CNN) if a larger dataset with diverse test strip images containing variations is available. For example, face recognition in smartphones. This occurs as the machine learning model has learned the facial stature of the user and can unlock the smartphone in any condition such as various light conditions (in the sun, clubs, bars, pubs, etc). Similarly if enough test strips are there indicating different shades of positive color(i.e. blue) then a Machine learning Model using Convolutional Neural Networks (CNN) can be developed so that users can take a picture of the test strip and immediately get the results on a mobile application.

VII. References

1. Prasannahariveeresh, “Implementing Lane Detection in OpenCV Python: A Step-by-Step Beginner’s Guide - Prasannahariveeresh.” *Prasannahariveeresh - Read My Tech Blogs Here*, 18 Jan. 2023, https://jrprasanna.com/2023/01/18/implementing-lane-detection-in-opencv-python-a-step-by-step-beginners-guide/
2. “Matplotlib Pie Charts.” *Vegibit*, vegibit.com/matplotlib-pie-charts/. Accessed 19 May 2023. https://vegibit.com/matplotlib-pie-charts/