



ISSN: 0067-2904

Controlling Image Color Programming and Thinking Outside the Box to Achieve Additional Results for the Image

Khaleel I. Abood, Ahmad Asaad Zaeen*, Laith Aziz Jawad

Remote Sensing Unit, College of Science, University of Baghdad, Iraq

Received: 14/12/2024

Accepted: 6/5/2025

Published: 30/6/2026

Abstract

This research is primarily an educational applied study, where a step-by-step MATLAB program has been written (unlike other ready-made programs). This allows researchers to modify, add, or remove any steps that suit their work in the program and what they find most appropriate for their tasks. The program is based on the spectral signature of the colors present in the map; it selects an area composed of 30×30 pixels or 20×20 pixels in a desert region, for example. The program recognizes its basic colors to identify all similar areas throughout the map and other regions, whether they are fertile areas, semi-desert regions, or water bodies. This research is a guide on applying for a program in MATLAB step by step to reach wide applications in image processing. It is not a compiled program, but it was written by the researchers manually so that users have wide options for entering equations as desired and without restrictions. Users may choose the same equations in the program by changing some constants to change results. Each map may be taken indoors to a specific area or even to the entire Earth by satellites in a unique pattern chosen by users. All that has to be done is input values defining the regions (desert regions, semi-desert regions, fertile agricultural regions, etc.), as has been done in this research. Through image processing, different regions were identified and accurately determined, and the areas of those regions were calculated using a map of Iraq using the Google Earth tool 530×530 pixels (68 KB). The color gradation of the three RGB colors in each region was taken, and they were determined by the minimum and maximum values for each color; the colors will be automatically chosen. These conditions were generalized to the entire map and subsequently identified automatically by the program. Good values were obtained from this method, as seen in the research where the desert lands represented 29.85% of Iraq's area, semi-deserts were almost 7.6 %, the lands which can be reclaimed were nearly 8.7 %, wetlands with deep groundwater were roughly 13.9%, wetlands with water close to the surface were almost 15.7 %. Finally, fertile agricultural lands represented roughly 14.6% of Iraq's areas.

Keywords: Color, google earth, image processing, remote sensing,

التحكم في برمجة ألوان الصورة والخروج عن المألوف لتحقيق نتائج إضافية للصورة

خليل ابراهيم عبود , احمد اسعد زعين* , ليث عزيز جواد

وحدة الاستشعار عن بعد، كلية العلوم، جامعة بغداد، العراق

*Email: ahmed.asaad@sc.uobaghdad.edu.iq

الخلاصة:

ان هذا البحث هو بحث تطبيقي تعليمي بالدرجة الاساس , حيث تم فيه كتابة برنامج ماتلاب خطوة بخطوة (وليس كالب برامج الجاهزة الاخرى) , حيث يتيح للباحثين تعديل وأضافة أو حذف أي خطوات تناسب عملهم في البرنامج ومايراه مناسباً اكثر لعمله , فهو يعتمد على البصمة الطيفية للالوان الموجودة في الخارطة , حيث يتم اختيار منطقة مكونة من 30×30 بكسل 20×20 بكسل في المنطقة الصحراوية مثلاً , فيتعرف البرنامج على الوانها الاساسية ليحدد كل المناطق المشابهة لها في جميع الخارطة , وهكذا باقي المناطق الاخرى , سواء كانت مناطق خصبة أو شبه صحراوية أو مسطحات مائية , هذا البحث دليلاً أساسياً لتطبيق برنامج في الماتلاب خطوة بخطوة للوصول إلى تطبيقات واسعة في معالجة الصور , وهو ليس برنامجاً معد مسبقاً بل تم كتابته بواسطة الباحث بحيث يكون للباحثين خيارات واسعة في إدخال المعادلات بالشكل المرغوب ودون قيود. وقد اختيرت نفس المعادلات في البرنامج بتغيير بعض الثوابت ليرى التغيير الناتج , وقد يتم أخذ كل خريطة إلى الداخل لمنطقة محددة أو حتى إلى الأرض بأكملها بواسطة الأقمار الصناعية بنمط فريد يختاره الباحث. كل ما على المتدرب فعله هو إدخال قيم تحدد المناطق (المناطق الصحراوية، المناطق شبه الصحراوية، المناطق الزراعية الخصبة، إلخ). من خلال معالجة الصور تم تحديد مناطق مختلفة وتحديدها بدقة، حسب مساحات تلك المناطق باستخدام خريطة العراق باستخدام أداة جوجل إيرث بقياس 530×530 بكسل (68 كيلو بايت)، وتم أخذ تدرج الألوان للألوان الثلاثة RGB في كل منطقة، وتم تحديدها بالقيم الدنيا والقصى لكل لون. والتي يتم تحديدها بشكل آلي، من ثم تم تعميم هذه الشروط على الخريطة بأكملها لتحديدها لاحقاً آلياً بواسطة البرنامج. اعطت هذه الطريقة نتائج بقيم جيدة جداً، حيث بلغت الأراضي الصحراوية 29.85% من مساحة العراق وشبه الصحراوية ما يقارب 7.6% والأراضي القابلة للاستصلاح ما يقارب 8.7% والأراضي الرطبة ذات المياه الجوفية العميقة ما يقارب 13.9% والأراضي الرطبة ذات المياه القريبة من السطح ما يقارب 15.7% وأخيراً الأراضي الزراعية الخصبة ما يقارب 14.6% من مساحة العراق.

1. Introduction

Generally, an image consists of colors as its primary content, and there is a secondary component, edge detection [1], that can also be replaced by colors to detect the edges of the internal components of the image [2]. The colors used in all CCD cameras are Red, Green, and Blue [3]. Every image is simply three layers of color applied to each other [4], where each pixel in the image consists of three colors (three wavelengths), and each color has a gradient ranging from 0 to 255 in intensity. The relationship between the number of photons and the intensity gradient is a direct linear relationship [5]. In Figure 1, each slice in (the RGB matrix) means a two-dimensional matrix of x and y, and each element in this matrix is one pixel.

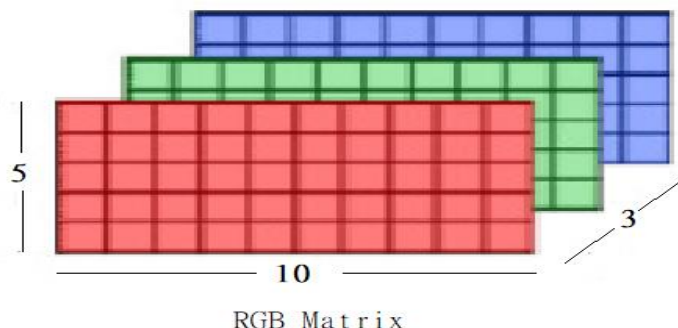


Figure 1: Contents of any color image are three matrices: Red, Green, and Blue [6].

Three two-dimensional arrays form a three-dimensional array [7], where each pixel for each color consists of a brightness ranging from 0-255[8]. Therefore, each pixel consists of

three numbers for each color [6]; for example, 0, 0, and 0 mean that all colors have zero brightness, resulting in black. If the pixel contains numbers 255, 255, and 255, the result is white light [9]. Mixing the three colors, red, green, and blue, in ratios from 0 to 255 in all possibilities will result in a total of 255^3 colors, which equals 16,581,375 visible colors [10] within the visual wavelength range, with limits ranging from 350-750 nm[11], Figure 2.

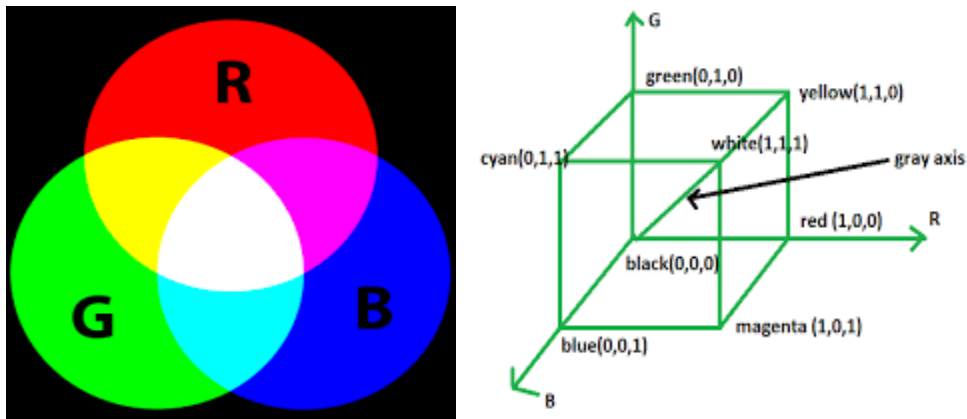


Figure 2: The mix of colors from black to white [12].

The image is read using the “imread” command in MATLAB will read all three matrices of the image simultaneously. If one wants to display only the red color matrix, one should write " image name (: , : , 1) " to select the first matrix that contains red color values, then green color " image name (: , : , 2) " and blue color " image name (: , : , 3) ", Figure 3 [13].

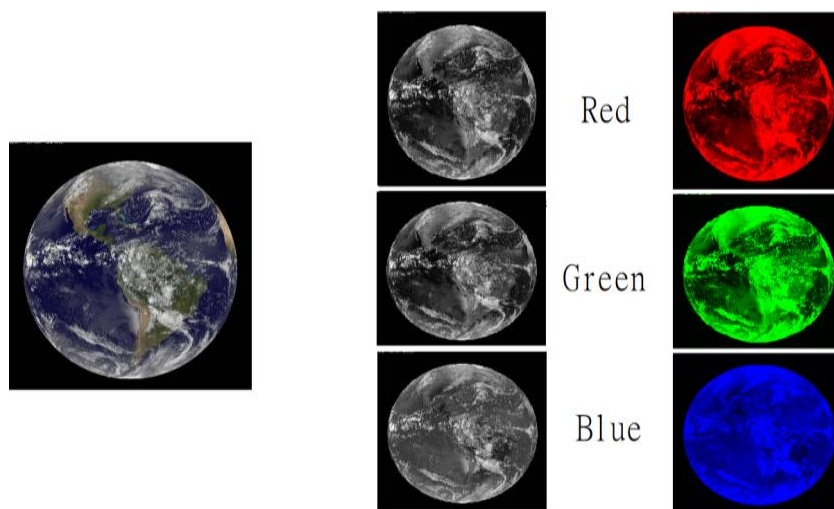


Figure 3: Any image in three colors can be translated to three images in gray color[13].

The colors on the right side in Figure 3 show the color in any color matrix, while the mid images represent the gray color of each matrix color, where the correlation between the colors depends on the image itself. The summation of each column in the matrix will be displayed as a curve in the x and y axes (Figure 4), where each point at x axis of any curve represents the sum of one column in the matrix color.

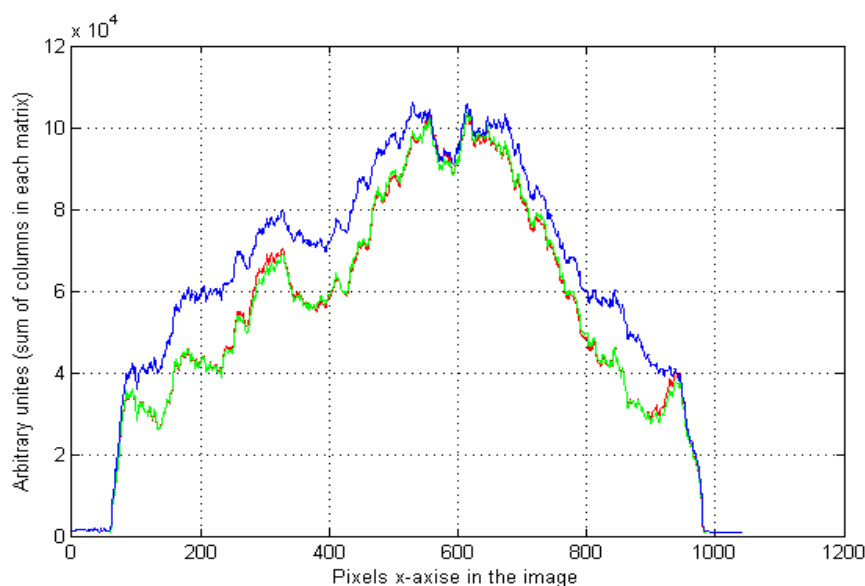


Figure 4: Three curves, R, G, and B, represent the correlation between the colors in this image[13].

Figure (4) represents the summation of each column in the image's matrix, which is done by the MATLAB program.

1.2 Some of the Applications of image processing by remote sensing are as follows:

- 1- Intelligent transportation technique used to recognize automatic number plates of cars[15],[17].
- 2- The remote sensing of spectral scanners, which represent sensors that capture the Earth's surface picture, is mounted on an aircraft [16],[18].
- 3- Tracking of object movement to monitor a motion and record the object moving [17],[19].
- 4- Defense monitoring is used to save all objects' direction motion on the land and oceans; it is a distribution of the spatial objects [18],[20].
- 5- Image processing in biomedical uses, different images by the X-ray devices, MRI, Ultrasound, etc. [20],[22].
- 6- An automatic visual inspection system improves factory product quality [19].

2. The aim of the research

This research eliminates restrictions by incorporating larger values, including negative and fractional ones. This enhancement allows for a broader spectrum of numbers, which helps provide more accurate information and detailed insights about the same image instead of being confined to a limited set of values. The technique also enables the use of trigonometric functions such as sine, cosine, and tangent, alongside logarithms and fractions for essential operations like multiplication, addition, subtraction, and division of image matrices; this represents the focus of the research.

3. Methodology

3.1. Determination of the map's area

The study used the Iraq map, where the image in the MATLAB program will appear, Figure (5). Cropping a map from a specific location can be done manually or, in some cases, using software like Photoshop. The area outside the map boundary should be filled with a single color (usually white) to calculate its area or remove it if needed easily. In this case, the map of Iraq was cropped from Google Earth maps, Figure 5.

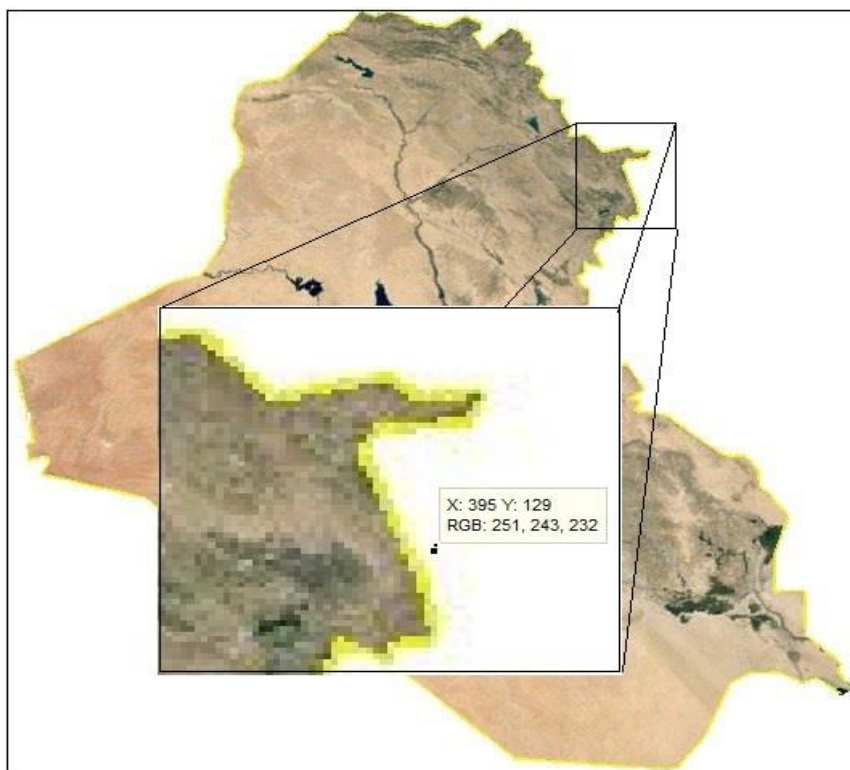


Figure 5: Iraq map, used from Google Earth tools, the height of the map is 9 cm, scale drawing $1/10^7$ [24].

The important point is that when taking this map as a portion of Google Earth with a resolution of 150 bytes per pixel, the image's size, usually kilobytes, is divided by the number of pixels to obtain the image resolution. This image is 42.6 kilobytes, with dimensions of 530×530 pixels. When dividing the image size in kilobytes by the number of pixels, the result is 151 bytes per pixel. If the results do not appear as expected in this research for any map, the map's resolution was less than 151 bytes per pixel. This issue must be carefully considered. To explain in more detail, each image has a specific resolution, and when worked with another image of lower clarity, the results were incorrect. Therefore, the results came out as expected in the research by taking an image with a higher resolution than the one worked with. When examining the image information, working with an image of 530×530 pixels, which had a size of 42.6 kilobytes. Thus, dividing the number of kilobytes (the size of the image) by the number of pixels, which is 530×530 pixels, gives us an average of 151 bytes per pixel. This represents the lowest byte count per pixel for any image that wants to work within this program, indicating the minimum clarity level the program can handle to produce good results. Therefore, if the image work has less than 151 bytes per pixel, the results will be incorrect; if it has more than 151 bytes per pixel, the results will be good, and the higher the byte count, the more accurate the results will be.

The map in the above figure consists of two main parts: the area outside the map boundary and the area within it. Therefore, it is necessary to calculate the entire image's total area (number of pixels), $n \times m$ (pixels of two dimensions along the x and y axes). Then, calculate the number of pixels of the external part of the map (in white color) where the condition in the software is 255,255,255. However, some pixels near the boundary map's lines may have less than 255 values because the transfer processes of the image (Figure 5) on the right side result in the pixels around map boundaries, making uncertain areas. Therefore, in the white area, the minimum value outside the map must be noted to be the condition in the MATLAB

program as the zoom in the right side of Figure 5, the RGB values are 251, 243, and 232, respectively, as indicated in Appendix B (lines from 1 to 18) in the program, where the number of map's pixels can be extracted from subtracting the number of black pixels from the total number of image's pixels, Figure 6. The figure scale is 7.2×7.2 cm.

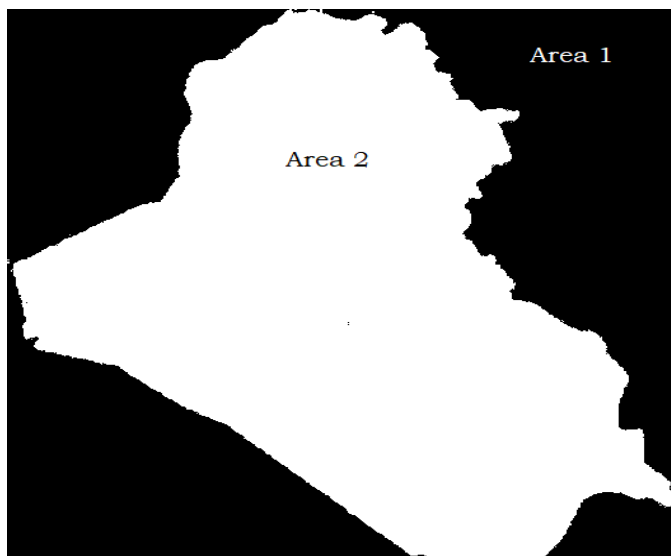


Figure 6: Area 2 came from subtracted outer black area1 from total image pixels $n \times m$.

Figure (6) can be found in the program (appendix B, steps from 35-52). The total image's pixels are $n \times m$ (dimensions of the image). The total pixels of this image are 530×530 pixels, where the outer pixels (black color) equal 13,3319 pixels and then the map's pixels (white color) equal 147581 pixels, where the resulting shape of the map will appear on the left side of Figure (7). However, notice some of the unlike pixels outside the map that must be removed by specifying more on conditions RGB colors' values by taking the values 220,220,220 instead of 251,243,232 (Figure 5), where in this case the number of pixels of the map will be 145,623 pixels. The result appears on the right side of Figure 7. Most figures in the research have rough measurements of 9×9 cm, where the accurate distance for every side of a square is equal to 900 km.

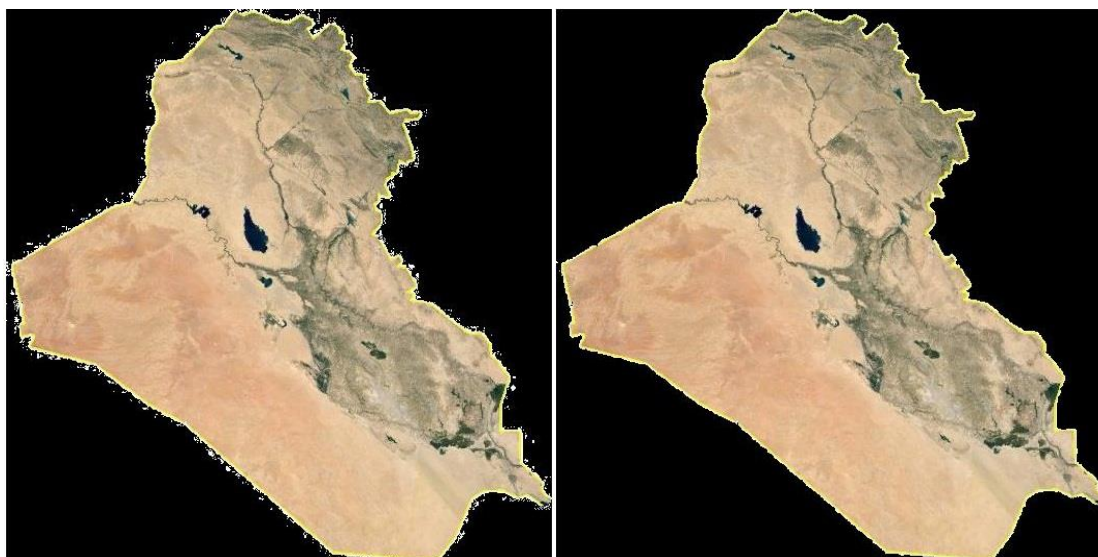


Figure 7: The ideal map limiting in the MATLAB program occurs by removing all remaining pixels; the map's height is 7.2cm, and the scale drawing is $8/10^6$.

The area of Iraq is about 438,317 km², while the map of Iraq has 145,623 pixels. Therefore, there is a map drawing scale that is calculated by dividing the number of pixels on the actual area in kilometers to determine the number of pixels per kilometer, or conversely, the number of kilometers per pixel is roughly 3.01 km²/pixel ~ 3 km²/pixels.

3.2 Three loops of three colors

This map is an aerial map of Iraq. The map considers raw images without processing, whereas the green regions represent plants, water, or both. It can distinguish dark green colors as water spots or watercourses by using only the eyes, but it cannot distinguish this watercourse when it has a low level and is close to green plants, as it will be in light green. The pale yellow, dark yellow, or orange-tinted color indicate deserts and dry sands.

The current map does not provide more detailed or additional information about the water or vegetation content, and the ratio of green parts to deserts cannot be calculated on the map. Therefore, the program was designed step by step in MATLAB (not ready-made software, but the program was designed by the researchers, and it is available in Appendix (B) of the research and can be used for other maps; users can place whatever mathematical variables they want between the three colors matrices, namely red, green, and blue, to give new results and new images of the same map.

The levels of colors were limited between 0-255 for each color (R, G, and B) since many mathematical operations between color matrices cannot be performed, such as logarithmic functions, exponents, or trigonometric functions (sine, cosine, tangent), because the rules of matrix numbers in the image only allow natural, positive, and non-fractional numbers limited with values only from 0 to 255 (integer and positive number). However, operations like multiplying exponents, addition and subtraction, or multiplying logarithms for matrices were done in this study. Therefore, the MATLAB program was created to break this barrier and to have free steps, as shown in Appendix (B), from 19-55 (loop e1 for three steps of three colors).

The first step in the program was to read the image to be processed, and the image should have a resolution above 530×530 pixels (68 KB) for more precise processing. To ensure that the program operates in the shortest time possible, the larger the image size, the longer it will take to process the program to extract results. There are huge images that can exceed 2000 x 2000 pixels, making it difficult for the program to analyze all this data quickly, and it may take several days to provide final results. After that, the program takes three counters (loops), one for each color channel, and each loop assigns a new name to the color matrix.

3.2- Choosing the suitable map from some mathematical equations

The resulting image matrices have the exact dimensions as the original, but if displayed using the “imshow” command, they would appear completely white without any features. This is because MATLAB did not interpret them as images but as mere numbers. Therefore, they must be normalized by dividing the matrix by its maximum value of 255. Alternatively, it can divide the matrix by a number greater than 255 to make the parts less faded and distinguish between the components of the image. This way, the image's color levels will be converted to a range between 0 and 1 (scale 0-1 for each color). As shown in the attached program, the first image displayed after taking its logarithm will be shown in Figure 8.

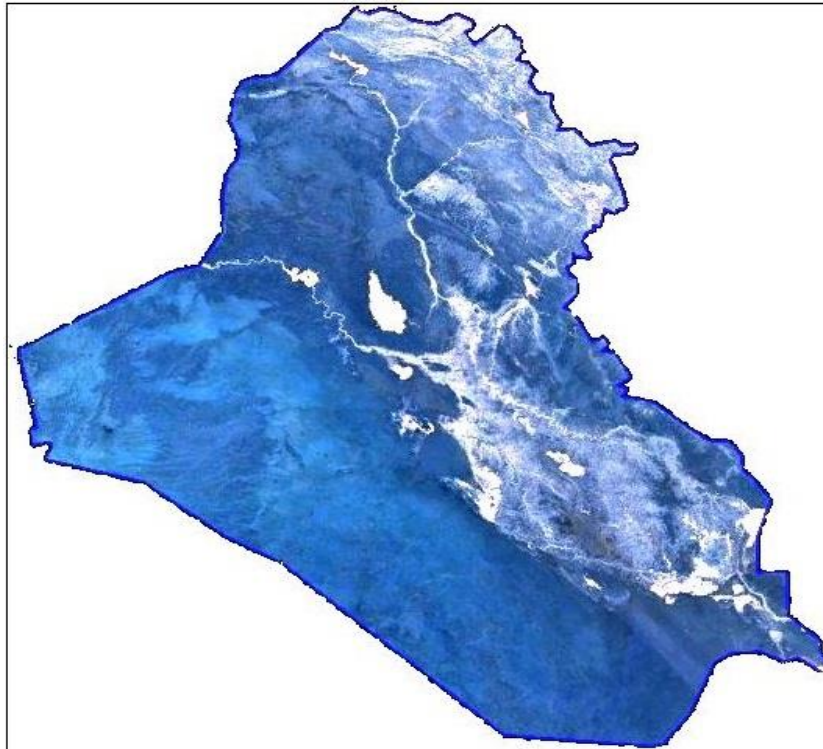


Figure 8: The map of Iraq uses a log (matrix); the map's height is 9 cm, and the scale drawing is $1/10^7$.

The simple color gradients in the above figure show distinct areas, particularly concerning water and green spaces. As illustrated, the white color indicates lakes, rivers, and water bodies, while the less white areas correspond to wetlands, and the areas with the least white indicate fertile regions suitable for agriculture. From this image, it is possible to calculate the volume of water relative to the desert areas or the total area of Iraq. Since the values in the matrices for this image range between 0 and 1, using the logarithm function will result in all negative numbers. Therefore, multiply the result by a negative sign before using the code (imshow) command. This can be done as shown in the following equation, which is found in Appendix (a).

$$p1 = -3\log(p) \quad \dots\dots\dots(1)$$

The name of the image in the MATLAB program is (p) where it represents three colors: $p(:, :, 1)$ is the matrix of red color, $p(:, :, 2)$ is the green, and $p(:, :, 3)$ is the blue color, where the parameter two before the logarithm is a personal choice which depends on the desire of the researcher.

In Figure 8, the level of white color means either wet areas or water bodies or pure water in the max level of white color; despite this, Figure 8 may still have some confusion between wet (suitable for agriculture) and dry areas in the low-intensity white levels. There is also an effect of colors on each other in the RGB image. Therefore, a more precise form is used to clarify the difference between dry and wet areas. Equation (1) of color matrices, Figure 9, has more details than Figure 8 because it displays each color alone. Figure 9 can be applied in the program (appendix B) steps 51-70.

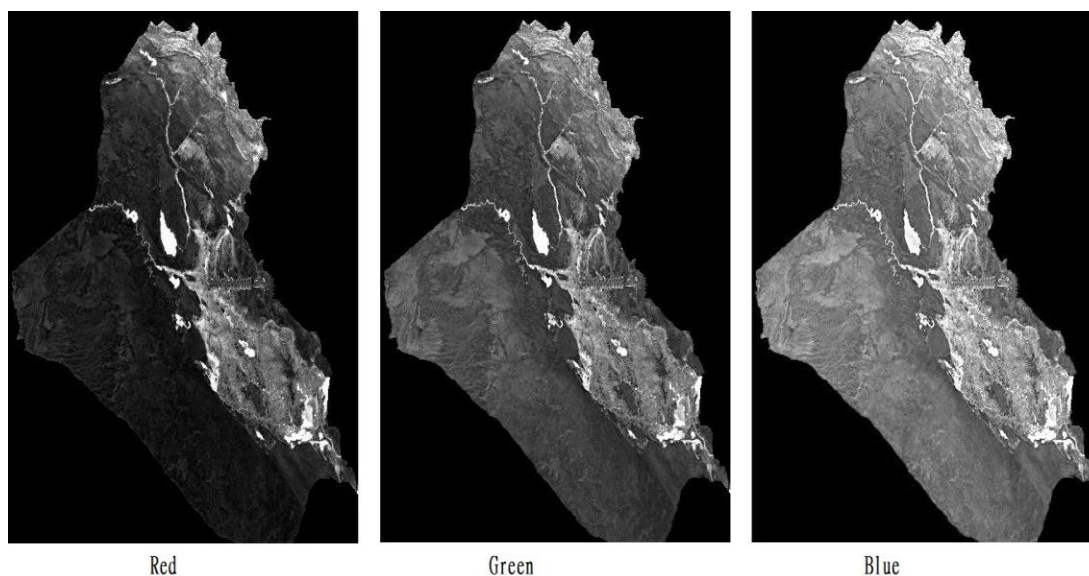


Figure 9: RGB colors using the logarithm function, the map's height is 6.5 cm, and the scale drawing is $7.2/10^8$.

The question may arise in equation (1): Why was a negative sign placed in this equation, and why was the number 3 specifically? The reason for converting the map scale from 0-255 to 0-1 is because taking the logarithm value of the values from 0-1 will always give us a negative value. The result must be multiplied by a negative sign to convert these values to positive numbers. As for number 3, it is because the map will appear very dark, causing many distinctive features to disappear. Multiplying it by 3 increases the clarity of the map on the screen. This number is arbitrary; choosing 2, 5, or any other number, but three was suitable for clarity. It was later measured and used to classify different regions. Choosing a number other than three would change everything after that, including all classifications of distinct numbers for different regions.

The red map is the most suitable due to the clear differentiation of light intensity between the wet and dry lands. Therefore, the red color map, Figure 10, contains intensities between 0-1 (left side), which is equivalent to 0-255 in the MATLAB program (right side), and the intensity light, which has a value more than 0.25 equivalent to 0.25×255 in this image which means the map of dry land and the water.

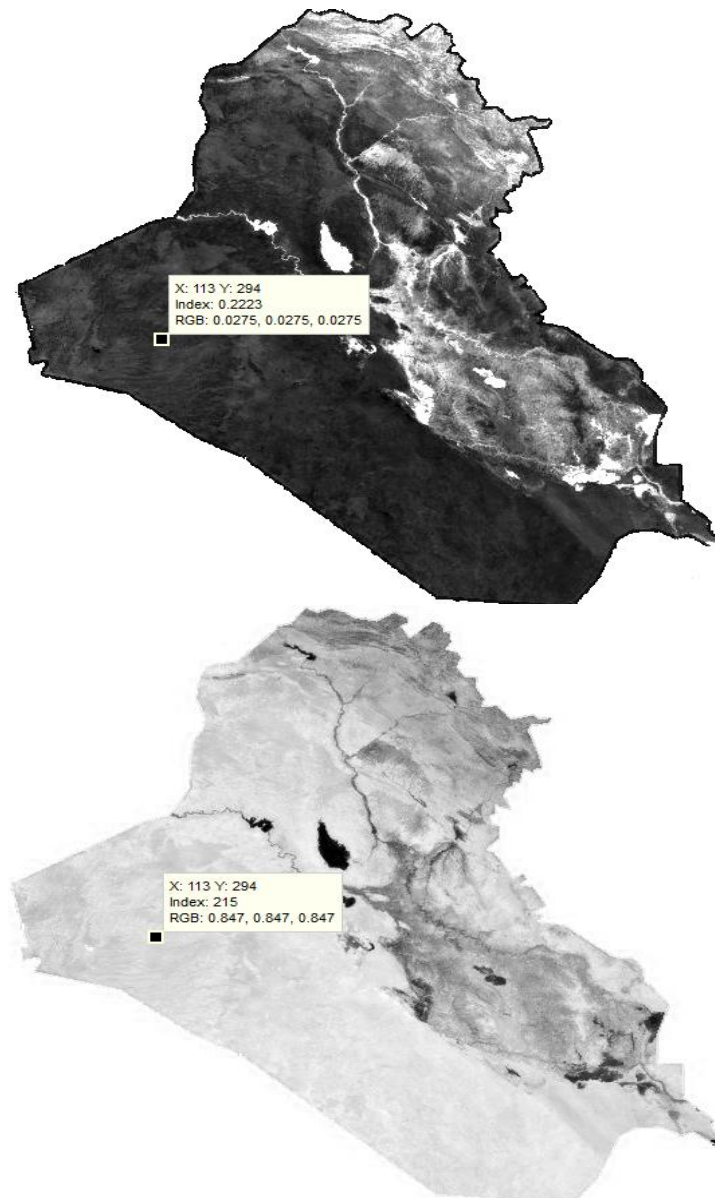


Figure 10: Red color using the logarithm function (left side), and red from the original image (right side). The map's height is 9 cm, and the scale drawing is $1/10^7$.

The map shows clear color variations on the left side of Figure 10 after using the logarithmic function. The map on the right side of the figure is the original image in red without any image processing, and it does not show clear variations between the areas or has very faint and subtle variations that cannot be easily distinguished between different areas. The percentage between non-humid and desert areas relative to the total area was calculated by comparing the number of pixels in the image of the map of Iraq with the number of pixels in non-humid (arid) areas. The ratio shows that there are 134,243 pixels with a brightness intensity less than 64 on a scale of intensity from 0 to 255 or a brightness intensity less than 0.25 on a scale of 0-1, which indicates the dark areas in the image and are the non-humid areas. On the other hand, there are 400,72 pixels with a brightness intensity more significant than 64 on a scale of 0-255, which indicates humid and agricultural areas and water sites. The ratio between the number of pixels from the first and second is that 69.67% of the Iraqi lands are in complete dryness and do not contain humid or agricultural areas, meaning they are semi-desert or desert areas. There are also 29.85% of a mix of humid and agricultural areas,

water bodies, river channels, and lakes. This result (percentage ratios) was adopted as a reference in this research from the data results of the Ministry of Planning [21-23, 25].

4-Other forms of map

Additional mathematical equations extract results for cities or country maps and infer additional information. The supplementary program in Appendix B (steps 64-70) used mathematical principles to create more accurate maps or maps with additional features. It will present users with the mathematical equations used for each. Users can follow the same steps for any image using the same equations presented in Figure 11.

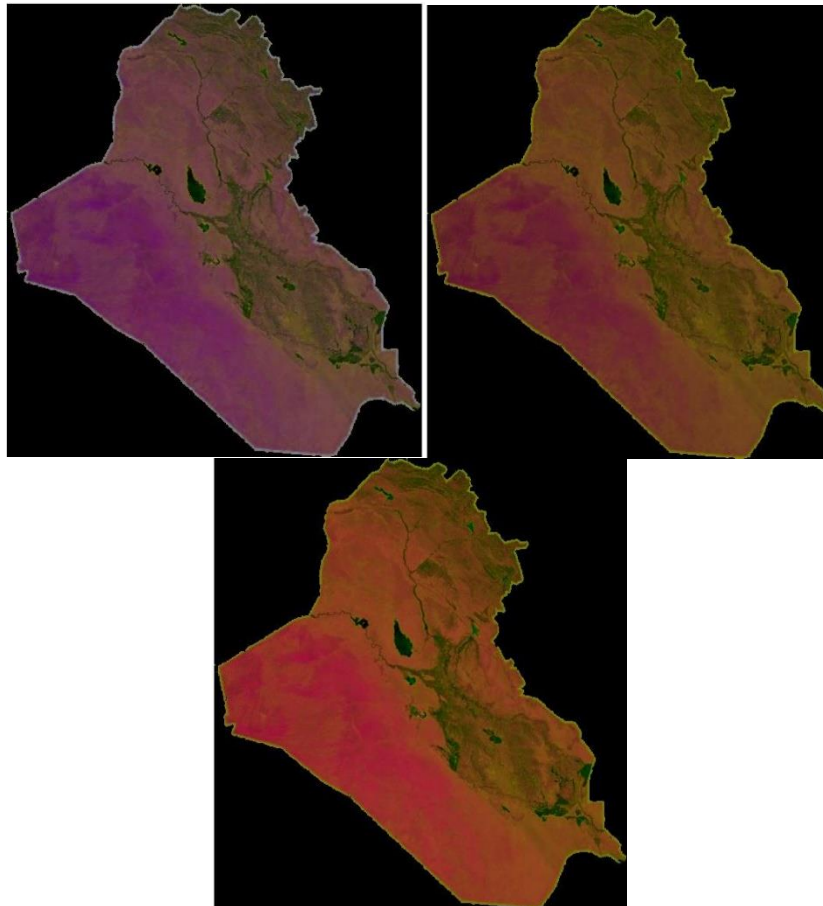


Figure 11: The natural map gives new forms of colors by using some equations, such as scale drawing $7/10^8$.

The three maps in Figure 11 represent RGB with all the colors, but the difference in each image is the addition or subtraction of a color matrix from another color matrix of the three colors by a certain percentage. Therefore, different percentages were applied to each color from the others, as shown in equations 1-9 in Appendix A. In the program, the name of the original image was (P), where (P) is three matrices of colors (RGB), the mathematical representation of the three images in Figure 11, which were put in the appendix (a), the equations from (1-9). Changing the colors in the shapes of Figure (11) is to increase the diversity of the research options. For example, if the question is about how to work with these shapes, desert areas are shown in light and dark pink colors. Therefore, the color blue can be used to indicate the gradient. As moving towards desert areas, the intensity of the blue color decreases first, Figure 12.

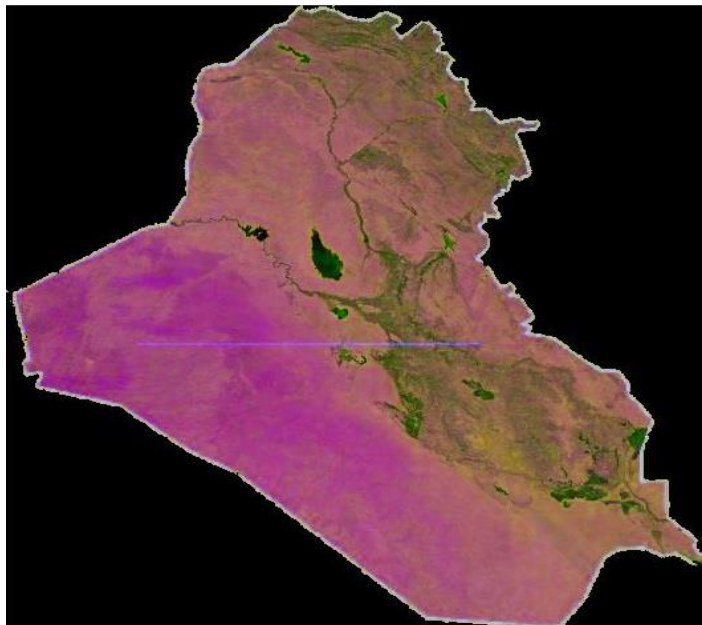


Figure 12: The Line indicates the range change in the scale blue matrix between different lands; the scale drawing is $6/10^8$.

The values of the blue matrix along the line will decrease in intensity as the movement was from arid regions to more humid land, where an intensity curve, Figure 13, can represent the pixels' values of the line.

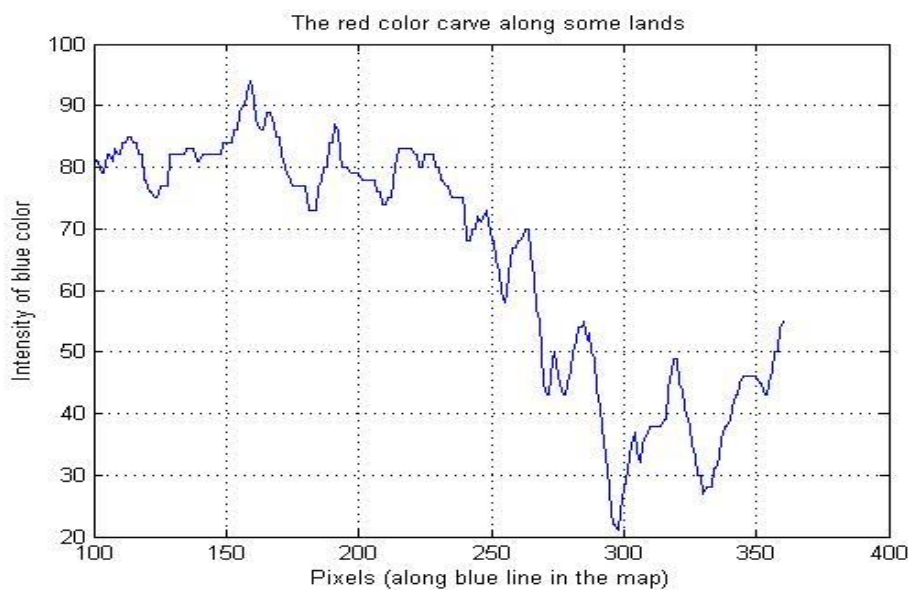


Figure 13: The intensity curve of the line in the map.

In general, the pixels in this map of blue matrix, the arid land has intensity values larger than humid land; therefore, the minimum values of blue color indicate land near the humid land, where the map was displayed in more detail, Figures 14-16.

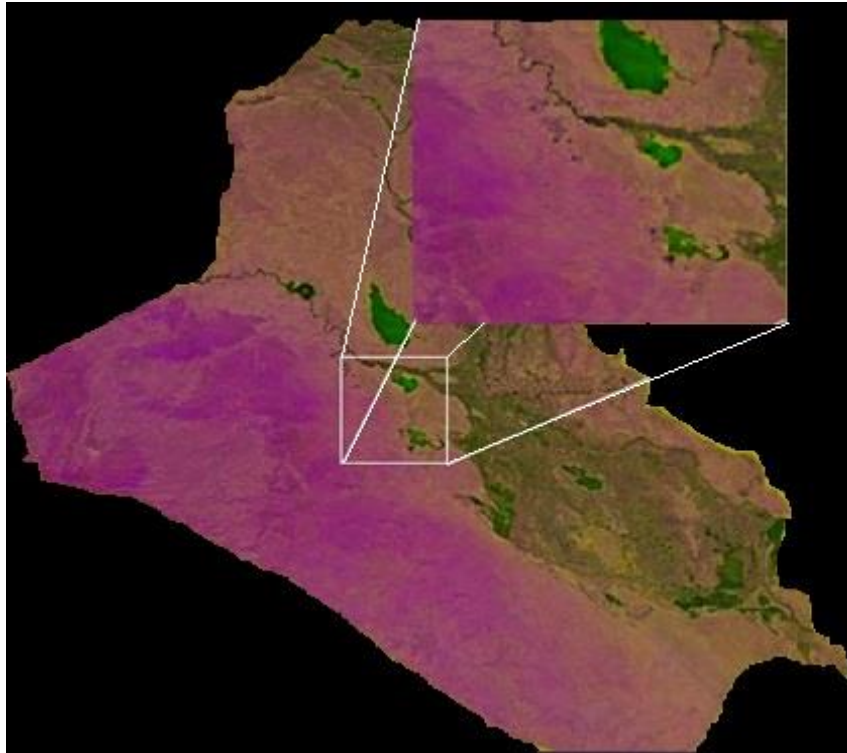


Figure 14: Part of the map explains the change in blue color along different lands; the scale drawing is $1.1/10^7$.

The enlarged portion of the map (test sample) will be selected to monitor its changes to decrease the blue color's value as it approaches the more humid areas, Figures 15-16.

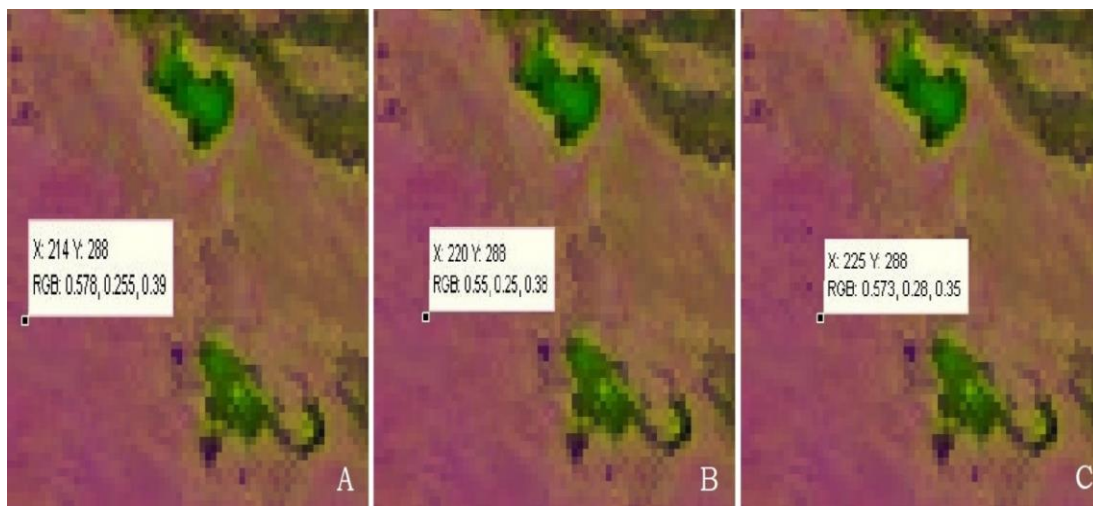


Figure 15: The intensity of the blue color decreases as moving toward the less arid areas; the scale drawing is $5.2/10^7$.

The intensity of the blue color decreases from 0.39 on a scale of 0-1, which equals roughly 100 on a scale of 0-255, to 0.38, which equals roughly 97, and then with more movement, will be 0.35, equivalent to 89 on scale 0-255; These are some steps when following the intensity of the blue color towards the more humid and less arid lands.

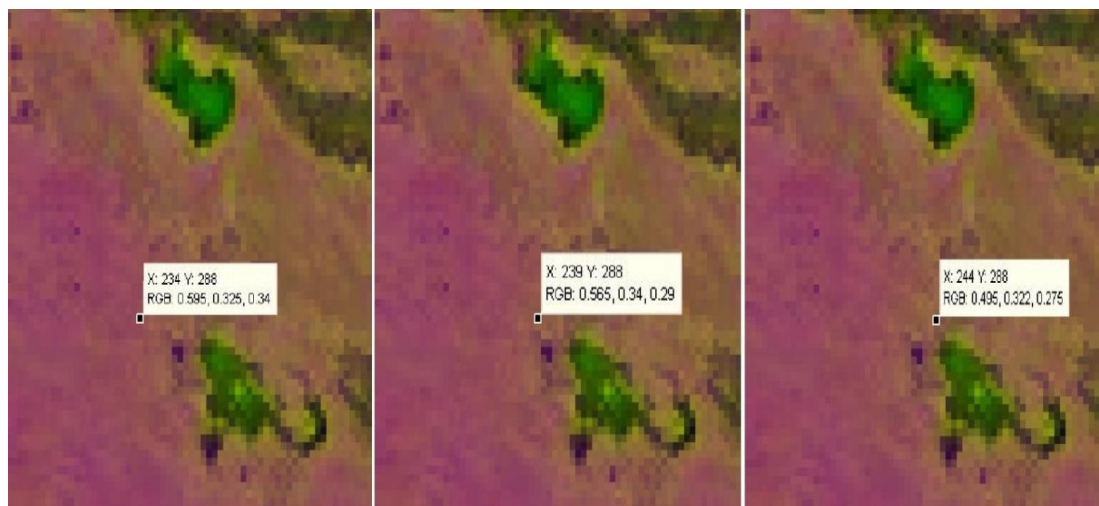


Figure 16: The intensity of the blue color has a minimum value near the humid land and water, and the scale drawing is $5.2/10^7$.

Much information can be extracted from this case, where the monitoring of the variables was so clear between the different lands that the accuracy of this method may arrive at 97%.

5-Results and Discussion

The lands were divided into nine regions according to their desertification rate or connection to humidity and water. There are desert areas, semi-desert areas, lands that can be reclaimed, semi-humid lands with deep groundwater (where the green color cannot be distinguished), wetlands with shallow groundwater, wetlands with water close to the surface where there is a very faint green color due to the presence of few wild plants, fertile agricultural lands with a clear green color, shallow water bodies, semi-deepwater bodies, lands without computing rivers, and lakes, Table 2.

Table 2: Some of the geographical features in Iraq are represented by RGB measurements.

No.	The land	Percent % to area of Iraq	Red color	Green color	Blue color
1	Desert	29.85	0.493 - 0.59 1	0.070 - 0.304	0.372 - 0.568
2	Semi-desert	7.569	0.555 - 0.601	0.307 - 0.357	0.362 - 0.387
3	Lands can be reclaimed	8.693	0.548 - 0.598	0.297 - 0.387	0.307 - 0.357
4	Wetlands with deep groundwater	13.87	0.475 - 0.593	0.294 - 0.387	0.266 - 0.331
5	Wetlands with water close to the surface	15.68	0.405 - 0.5477	0.221 - 0.457	0.226 - 0.307
6	Fertile agricultural lands	14.58	0.106 - 0.5251	0.156 - 0.440	0.005 - 0.216
7	Shallow water bodies	2.657	0.186 - 0.3769	0.201 - 0.377	0.070 - 0.166

The results in the table were put by dividing the blue line in Figure 12 into seven parts depending on the curve in Figure 13 when the range of curve values changed from 22 to 94 (on a scale of 0-255). These values change along the blue color in Figure 12 from desert areas (highest value 94) to fertile land (lowest value 22); the blue color values from the table for the desert region range from the highest value of 1 to the lowest value of 0.371 on a scale of 0-1. The total number of pixels on the map is 131797 (excluding the black color outside the map). The number of pixels indicated in white on the map for the desert areas is 39271

pixels. The percentage is 29.8%, representing desert areas. This percentage is considered a key indicator obtained from a reliable source for the Ministry of Planning, which was used to predict the lowest value of the blue color at 0.371 on a scale of 0-1. 97 on a scale of 0- 255, indicating the intensity of the blue color. Figure 12 was adopted in this research as a basis for determining geographical locations and their proportions, and the specific values for each color of the RGB colors in Table 1 refer only to this map. These values will change from one map to another. This color map was adopted as shown in the mathematical equations and program in Appendix (B), where the color values will be the same for each map. The desert land can be shown on the map on the left side of Figure 17. The white points indicate the desert region, which is indeed the western region of Iraq. This confirms the practical results of the work in the program of this study. The semi-desert areas will be shown on the right side of Figure 17. The results from Figures 17-20 appeared in the program (steps 70-94).

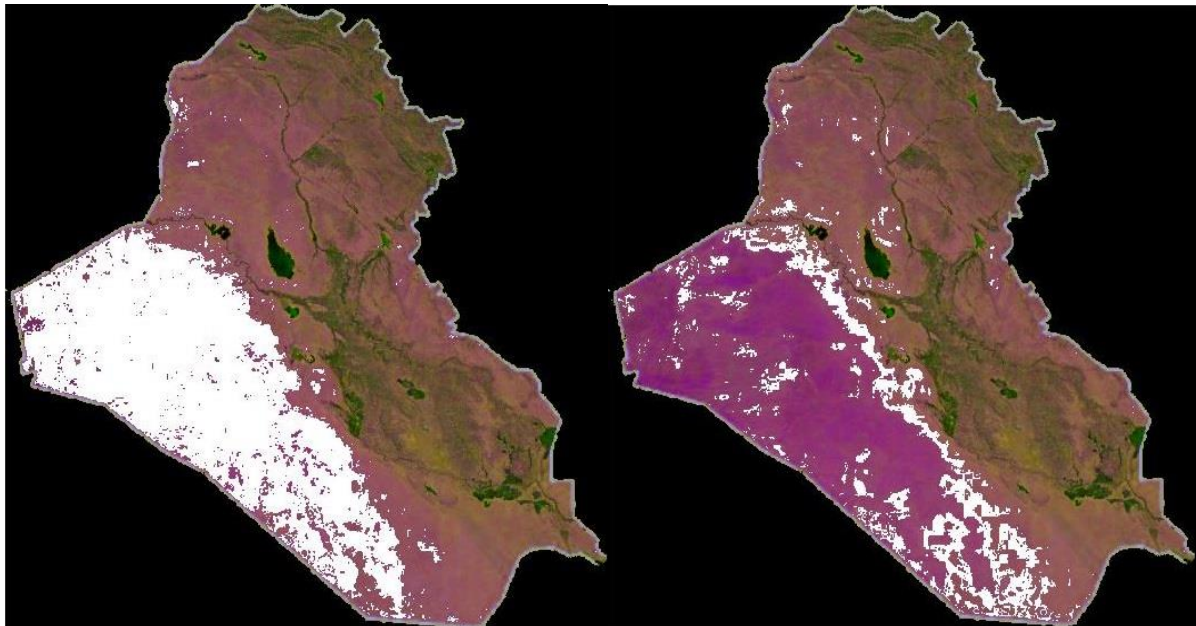


Figure 17: The desert land in Iraq is 29.85% (left side), and the right side is semi-desert, 7.57% of the total area.

The map shows that the semi-desert areas are adjacent to the desert regions. There are semi-desert areas around Lake Tharthar, and a few extend to the northwest of Iraq, as indicated by the white dots in Figure (17). The lands that can be reclaimed are shown in Figure 18. The lands that can be reclaimed are characterized by a clear green color, as previously mentioned. These wild plants can grow in areas with low water content but are not fertile and, at the same time, not completely dry. They can be reclaimed for agriculture. Wetlands with deep groundwater lands are shown on the right side of Figure 18.

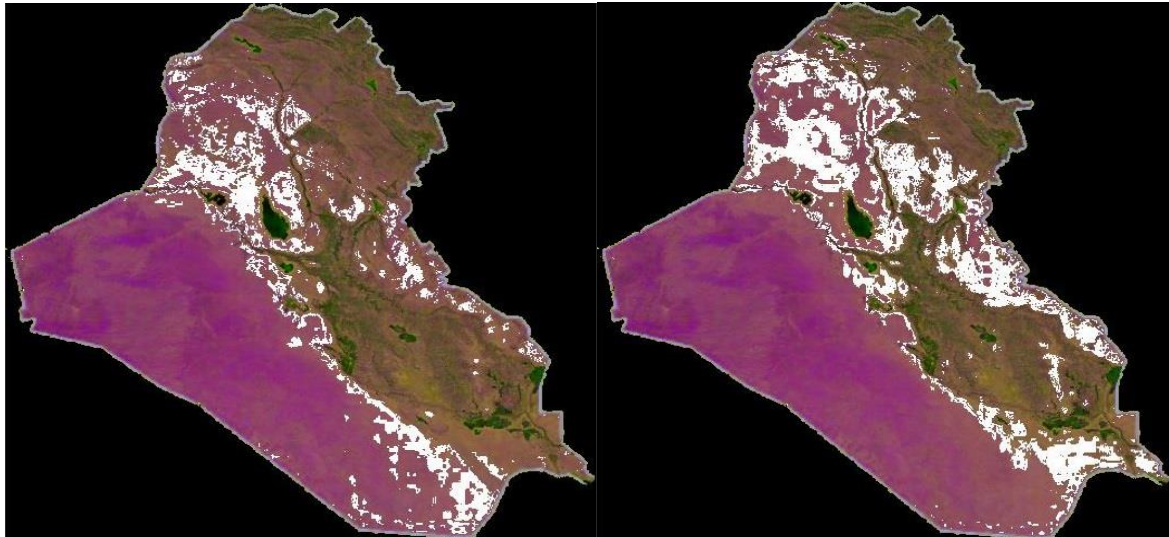


Figure 18: Lands can be reclaimed, 8.7% (left). Wetlands with deep groundwater constitute 13.87% of the total area (right), and the scale drawing is $9.6/10^8$.

The lands with deep groundwater are adjacent to green agricultural areas, while areas with shallow groundwater are included in the pale green agricultural areas on the left side of Figure 19. In principle, the lands with shallow groundwater are supposed to be particularly adjacent to surface water, such as rivers. However, the Euphrates River appears to be a very thin line, and the pixel colors of the areas near it seem to have overwhelmed it, so it did not give us precise results along the river, especially in the desert areas in western Iraq. This is because each pixel on the map equals $3 \text{ km}^2/\text{pixel}$, and the results would be evident if the map were enlarged to clarify the features more and more. The fertile agricultural lands appear on the right side of Figure 19.

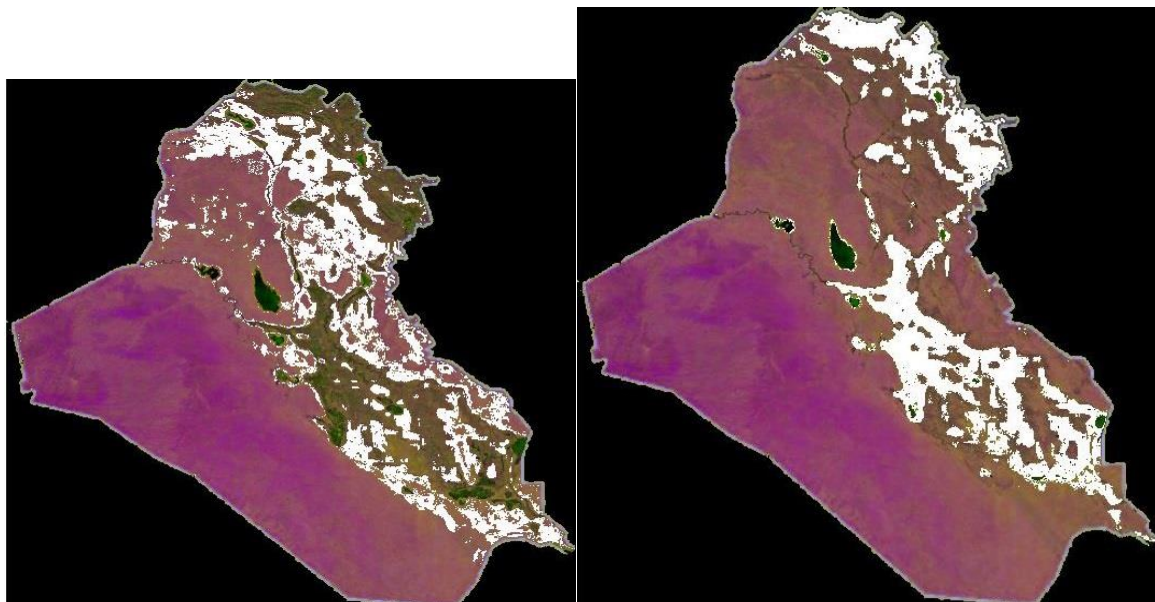


Figure 19: Wetlands with water close to the surface represent 15.677% (left). Fertile agricultural lands constitute 14.58% of the total area (right); the scale drawing is $9/10^8$.

Agricultural lands should certainly be dark green, which the program has identified and marked as white. They constitute 14.58% of the total area of Iraq. The shallow water bodies appear in Figure 20.

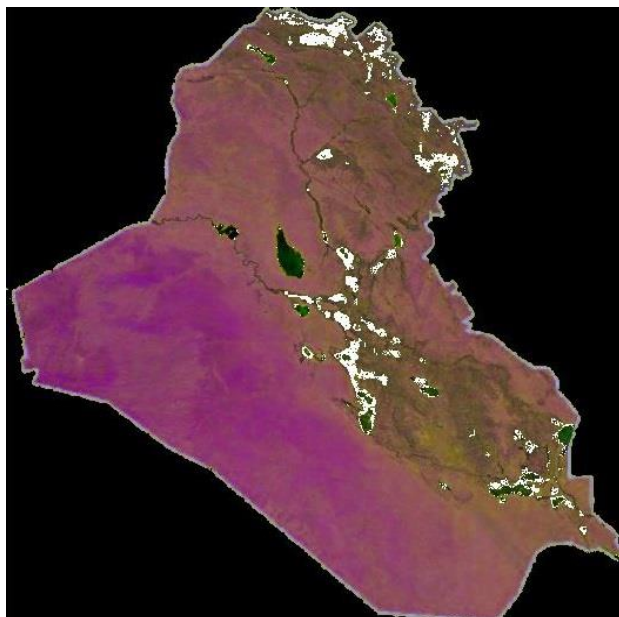


Figure 20: Shallow water bodies represent 2.265%, and the scale drawing is $9/10^8$.

The water bodies here are shallow and not deep, so they do not include lakes and rivers. It can be seen that there are abundant water bodies in the areas located in the far north of Iraq, close to the Turkish border, including large parts above the mountains that feed the springs and small waterfalls found in northern Iraq.

6. Conclusion

The MATLAB program currently available in appendix (B) allows researchers a wide range of options, depending on the researcher's preference in choosing the appropriate color scheme for the map to calculate what can be calculated from it, whether in percentage or real areas, provided that the total area of the map is known by dividing it by the number of pixels of the map itself to produce the area of one pixel on the ground. One pixel is approximately 3 km^2 , which was obtained by dividing the total area of Iraq by $438,317 \text{ km}^2$. The number of pixels for the map itself, not for the entire image, is 145,623 pixels. Knowing the area of one pixel can calculate the areas marked in the program. Even if the area is not known in real terms, it can compensate for this by using percentages and dividing the number of pixels for a specific desert region extracted in the program. A counter inside the loop calculates the number of pixels within the conditions of the desert region, which is divided by the total number of pixels on the map, thus deducing the percentage of this area from the total area of the map. Some conclusions were drawn as necessary steps for working with the map. Firstly, it is essential to crop the map before reading it in the program and place white color outside the map boundaries. By transferring the image and reading it in the program, it will become apparent that there are non-white color artifacts that have appeared near the image borders, which do not represent the white color (255,255,255) but rather RGB values lower than that in each actual color. The program has set conditions to remove these artifacts, which are set at a threshold of 220,220,220 for RGB values. If the RGB values are equal to or greater than 220, they should be set to 0, resulting in 0,0,0, which is black. This will turn the background from white to black outside the map boundaries. These processes ensure that the calculated area measurements for the number of pixels are accurate and do not affect the extracted results.

The extracted results of the areas and proportions of desert, semi-desert, agricultural, and fertile lands are simply statistics of the program's outcomes and secondary results. The main reason for conducting this research is to select the appropriate map shape based on its colors to identify the desired areas according to the researcher's preference, which is the foundation of the research. The equations were presented efficiently, and the program every researcher can use on their computer was included. They can work with it and start working on their own. They can also create other equations according to their preferences to derive different maps with different colors that may be more suitable for their work than those mentioned in this research.

7. Disclosure and conflict of interest

The authors declare that they have no conflicts of interest.”

References

- [1] O. Marques and G. B. Borba, “*Image Processing Recipes in MATLAB.*” CRC Press, 2024.
- [2] L. Chen and F. Su, "Towards Robust Video Text Detection with Spatio-Temporal Attention Modeling and Text Cues Fusion," in *2022 IEEE International Conference on Multimedia and Expo (ICME)*, 2022, pp. 1-6: IEEE.
- [3] A. Clark-Wilson, O. Robutti, and M. Thomas, "Teaching with digital technology," *Zdm*, pp. 1-20, 2020.
- [4] C. Woodhouse, *Capturing the Universe: A Photographer’s Guide to Deep-Sky*, Taylor & Francis, 2020, Pp220,220, ISBN:9781000078961, 1000078965
- [5] J. P. Mueller and J. Sizemore, *MATLAB for Dummies*. John Wiley & Sons, 2021.
- [6] J. Courtney, 2001, "Application of Digital ImageProcessing to Marker-free Analysis of Human Gait ", ResearchGate, https://www.researchgate.net/publication/267210444_Application_of_Digital_Image_Processing_to_Marker-free_Analysis_of_Human_Gait/figures?lo=1
- [7] D. P. Martin, A. Varsani, P. Roumagnac, G. Botha, S. Maslamoney, T. Schwab, Z. Kelz, V. Kumar, and B. Murrell, "RDP5: a computer program for analyzing recombination in, and removing signals of recombination from, nucleotide sequence datasets," *Virus Evolution*, vol. 7, no. 1, p. veaa087, 2021.
- [8] P. Dinesh, S. Mehta, *Handbook of Data Structures and Applications*, CRC Press, Pp 60,2018, ISBN:9781351645645, 1351645641
- [9] D. Koterov, *Creating Your Web Site with PHP*, A-List, LLC, Pp 319, 2002, ISBN:9781931769044, 1931769044
- [10] R. Hirsch, *Light and Lens: Thinking about Photography in the Digital Age*. Routledge, 2022
- [11] K. John, B. Paul, D. Kelter, and S. Hage, *Chemistry: An Industry-Based Introduction with CD-ROM*, 1st Edition ed. Boca Raton, 2000, p. 544. [Online]. Available.
- [12] R. Gupta, Paper Publications Study of Color Image Processing for Capturing and Comparing Images, *International Journal of Recent Research in Mathematics Computer Science and Information Technology*, Vol. 2, Issue 1, pp: (127-132), Month: April 2015 – September 2015,
- [13] J. Sanchez, *Space Image Processing*, CRC Press, Pp 75,2018, ISBN:9781351415439, 1351415433
- [14] R. F. Hanash, S. Z. Khalaf, and K. I. Abood, "Photometry Technique to Map Elements’ Distribution on Comets’ Nuclei Surfaces Using a New Method," *Iraqi Journal of Physics*, vol. 21, no. 3, pp. 77-91, 2023.
- [15] M. B. Kishore, *Sustainable Science and Intelligent Technologies for Societal Development: IGI Global*, Pp 61, 2023, ISBN:9798369311875
- [16] H. J. J. Chen, W. Yu, X. Tian, J. Huang, and J. Ma, "MDbFusion++: A Visible and Infrared Image Fusion Framework Capable for Motion Deblurring," *IEEE Transactions on Image Processing*, vol. 35, pp. 3647-3661.2026.
- [17] T. A. Touma, H. K. Abbas, A. H. Al-Saleh, and A. A. Al-Zuky, "Traffic light detection system using video images analysis technique," in *AIP Conference Proceedings*, 23, vol. 2414, no. 1: AIP Publishing.

[18] D. Galar, U. Kumar, D. Seneviratne, R. Drones, UAVs and UGVs for Operation and Maintenance, CRC Press, Pp162, 2020, ISBN:9780429839184, 0429839189

[19] S. Amirian, K. Rasheed, T. R. Taha, and H. R. Arabnia, "Automatic image and video caption generation with deep learning: A concise review and algorithmic overlap," *IEEE access*, vol. 8, pp. 218386-218400.2020.

[20] W. Burger and M. J. Burge, *Digital image processing: An algorithmic introduction*. Springer Nature, 2022.

[21] A. K. Tyagi, Computational Analysis and Deep Learning for Medical Care, Wiley, Pp121, 2021, ISBN:9781119785729, 1119785723

[22] K. H. Abbas Al-Aarajy, A. A. Zaeen, and K. I. Abood, "Supervised Classification Accuracy Assessment Using Remote Sensing and Geographic Information System," *TEM Journal*, vol. 13, no. 1, 2024.

[23] M. Hassaballah and A. I. Awad, *Deep learning in computer vision: principles and applications*. CRC Press, 2020.

[24] GoogleEearth.<https://earth.google.com/web/>

[25] A. F. Militino, M. Moradi, and M. D. Ugarte, "On the performances of trend and change-point detection methods for remote sensing data," *Remote Sensing*, vol. 12, no. 6, p. 1008.2020.

Appendix (A)

The image's equations on the left side in Figure (9) follow the equations:

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{red} = 0.5 \times P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{red}$$

....(1)

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{green} = 1.5 \times P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{green} -$$

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{red} \dots (2)$$

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{blue} = P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{red} -$$

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{blue} \quad \dots (3)$$

The result of the matrix (P) was multiplied by 1.5, while the image's equations in the middle are as follows:

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{red} = 0.5 \times P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{red}$$

....(4)

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{green} = 1.5 \times P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{green} -$$

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{red} \dots (5)$$

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{blue} = P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{red} -$$

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{green} \quad \dots(6)$$

The result of the matrix (P) was multiplied by 1.5, while the image's equations on the right side are as follows:

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{red} = 1.5 \times P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{red} -$$

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{green} \quad \dots(7)$$

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{green} = 1.5 \times P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{green} -$$

$$P \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ \vdots & \ddots & \vdots & & \\ m & \dots & (n, m) & & \end{pmatrix} \text{red} \dots(8)$$

\dots (9)

Appendix (B)

```

clc
clear all
close all
% steps of work
% 1- Calculate the map pixels by the pixels outside the map must be black color
% 2- processing the boundary of map to be blacked color
% 3- Convert the image from scale 0-255 to scale 0-1
% 4- Choose the map equations that is suitable for you
% 5- Return the image to standard value by divided on max value
% 6- There are boundaries colors values in the following table
% which refer to type of land (chosed from them to put them in the program loop)

% Type of land      Red color      Green color      Blue color

%1- Desert          0.5905 - 0.4925  0.3040 - 0.0704  0.5678 - 0.3719
%2- Semi-desert     0.6005 - 0.5553  0.3568 - 0.3065  0.3869 - 0.3618
%3- Lands can be reclaimed
%4- Wetlands with deep groundwater  0.5930 - 0.4749  0.3869 - 0.2940  0.3317 - 0.2663
%5- Wetlands with water close to surface  0.5477 - 0.4045  0.4573 - 0.2211  0.3065 - 0.2261
%6-Fertile          0.5251 - 0.1055  0.4397 - 0.1558  0.2161 - 0.0050
% agricultural lands
%7-Shallow water bodies  0.3769 - 0.1859  0.3769 - 0.2010  0.1658 - 0.0704
    
```

```

1) i=imread('iraq.jpg');
2) figure(1)
3) imshow(i)
4) % _____ 1- Calculation map pixels
5) % _____ 2- boundary of map to be blacked color

6) yy=530;xx=530;
7) k1=0;
8) for y=1:1:yy;
9) for x=1:1:xx;
10) if i(x,y,.)>220;
11) k1=k1+1;
12) i(x,y,.)=1; % We chosed avalue 1 instead of zero
i.% because when dividing by 0 in the normalization
ii.% Step, by dividing by the max. value, the value '
iii.% will be undefined
13) end;end;end
14) figure(2)
15) imshow(i)
16) allpixels=x*y;
17) outerpixels=k1;
18) mappixels=allpixels-outerpixels
19) %3- Convert the image from scale 0-255 to scale 0-1
20) k1=0;

21) for y=1:1:yy;
22) for x=1:1:xx;
23) u= i(x,y,1);
24) u1=0;
25) for e1=1:1:u
26) u1=u1+1;
27) end

28) u=i(x,y,2);
29) u2=0;
30) for e1=1:1:u
31) u2=u2+1;
32) end
33) u=i(x,y,3);
34) u3=0;
35) for e1=1:1:u
36) u3=u3+1;
37) end
38) p(x,y,1)=u1;
39) p(x,y,2)=u2;
40) p(x,y,3)=u3;
41) if p(x,y,1)==0
42) p(x,y,1)=1;
43) end
44) if p(x,y,2)==0

```

```

45) p(x,y,2)=1;
46) end
47) if p(x,y,3)==0
48) p(x,y,3)=1;
49) end;end;end
50) maxp=max(max(max(p)));
51) % Return the image to standard value by divided on maximum value
52) p=p./maxp; % Normalization
53) maxp=max(max(max(p)))
54) figure(3)
55) imshow(p)
56) figure(4)
57) imshow(-3*log10(p))
58) figure(5)
59) imshow(-3*log10(p(:, :, 1)))
60) figure(6)
61) imshow(-3*log10(p(:, :, 2)))
62) figure(7)
63) imshow(-3*log10(p(:, :, 3)))
64) p1=p;
65) p2(:, :, 1)=(p1(:, :, 1)*1.5-p1(:, :, 1));
66) p2(:, :, 2)=(p1(:, :, 2)*1.5-p1(:, :, 1));
67) p2(:, :, 3)=(p1(:, :, 1)-p1(:, :, 3));
68) figure(8)
69) imshow(p2)
70) % _____
71) % The following step is very important to return the map to true values
72) maxvalue=max(max(max(p2)));
73) standardmap=p2./maxvalue;
74) p3=standardmap;
75) figure(9)
76) imshow(standardmap)
77) maxstandardmap=max(max(max(standardmap)))
78) p4=p3;p5=p3;
79) k1=0;k2=0;
80) for y=1:1:yy;
81) for x=1:1:xx;
82) if p3(x,y,1)>0.49
83) if p3(x,y,1)<0.59
84) if p3(x,y,2)>0.07
85) if p3(x,y,2)<0.3
86) if p3(x,y,3)>0.37
87) if p3(x,y,3)<0.57
88) k2=k2+1;
89) p3(x,y,:)=1;
90) end;end;end;end;end;end;end;end;end
91) figure(10)
92) imshow(p3)
93) percentage=100*k2/mappixels
94) Desert =100-percentage

```