Practical 1 Part 2 Image Processing

March 9, 2021

Jaya Parmar, MATLAB filename P1_ex4_rework.m

1 Introduction

Our aim is to make a contrast enhancing algorithm that works for all images.

2 Method

2.1 Study the images and their histograms

1. **Read in images**
   We start the study by reading in all 5 images,

$$dive2 = imread('dive2.jpg') \tag{1}$$

$$dive3 = imread('dive3.jpg') \tag{2}$$

$$dive4 = imread('dive4.jpg') \tag{3}$$

$$dive31 = imread('dive31.jpg') \tag{4}$$

$$dive39 = imread('dive39.jpg') \tag{5}$$

2. **Image histograms**
   Then we study the histograms of these images by plotting them with Matlab command 'imhist'

$$subplot(3,2,1), imhist(dive2), title('dive2') \tag{6}$$

$$subplot(3,2,2), imhist(dive3), title('dive3') \tag{7}$$

$$subplot(3,2,3), imhist(dive4), title('dive4') \tag{8}$$

$$subplot(3,2,4), imhist(dive31), title('dive31') \tag{9}$$

$$subplot(3,2,5), imhist(dive39), title('dive39') \tag{10}$$

$$sgtitle('Original\ images\ histograms') \tag{11}$$

These histograms are shown in **Figure1** in results section. We can see that the histograms of every image covers the entire range of x-axis with more pixels on the darker intensity than the brighter intensity.

3. **Seperate the RGB channels from each image**
   To understand more about these contrast in these images, we need to seperate their red, green and blue channels and study them individually. We know that red is the first channel, green second and blue third. Below code uses this information for the first image dive2. We write below code for all images in our script,

$$red2 = dive2(:,:,1) \tag{12}$$

$$green2 = dive2(:,:,2) \tag{13}$$

$$blue2 = dive2(:,:,3) \tag{14}$$

We also pick the size of the image and store the row and column size in m and n variables. Below code shows it for dive2,

$$[m2, n2] = size(dive2); \tag{15}$$

4. **Red, Green and Blue Histograms**
   Now is the time to plot the red, green and blue histograms since we have the reds, greens and blues stored in seperate variables.

$$subplot(3,2,1), imhist(red2), title('dive2') \tag{16}$$

$$subplot(3,2,2), imhist(red3), title('dive3') \tag{17}$$

$$subplot(3,2,3), imhist(red4), title('dive4') \tag{18}$$

$$subplot(3,2,4), imhist(red31), title('dive31') \tag{19}$$

$$subplot(3,2,5), imhist(red39), title('dive39') \tag{20}$$

$$sgtitle('Red\ histograms') \tag{21}$$

The results are shown in **Figure2** of the results section. All the histograms are spread in the darker intensity range and gathered at one place. This shows a bad contrast in red images.

$$subplot(3,2,1), imhist(green2), title('dive2') \tag{22}$$

$$subplot(3,2,2), imhist(green3), title('dive3') \tag{23}$$

$$subplot(3,2,3), imhist(green4), title('dive4') \tag{24}$$

$$subplot(3,2,4), imhist(green31), title('dive31') \tag{25}$$

$$subplot(3,2,5), imhist(green39), title('dive39') \tag{26}$$

$$sgtitle('Green\ histograms') \tag{27}$$

The results of green histograms are shown in **Figure3** . These are better in contrast than the red histograms since they are spread across the range of x-axis and cover the brighter intensities also.

$$subplot(3,2,1), imhist(blue2), title('dive2') \tag{28}$$

$$subplot(3,2,2), imhist(blue3), title('dive3') \tag{29}$$

$$subplot(3,2,3), imhist(blue4), title('dive4') \tag{30}$$

$$subplot(3,2,4), imhist(blue31), title('dive31') \tag{31}$$

$$subplot(3,2,5), imhist(blue39), title('dive39') \tag{32}$$

$$sgtitle('Blue\ histograms'); \tag{33}$$

Results in **Figure4** show that the blue histograms are spread from darker to mid range intensities.

We can say that red channel will contribute to a poor contrast in the overall image, the blue contributes to an average contrast and the green channel will contribute to the best contrast in the overall image.

2.2 Pre-processing operation

1. **Make reference image from best R,G and B histograms**
   Let us look at the original histograms in Figure 1. We can see that all the images peaks are inclined more to darker intensities than to lighter intensities. To make a reference image, we look for the color histograms having **lighter** intensity peaks so that the overall images shift to the center range on x-axis.
   The red and green histograms of dive3 are higher in contrast compared to other images since they are spread out in a wider range.
   The blue histogram of dive31 peaks more towards the center compared to dive3.

   We will use **Michaelson's contrast formula** in the min-max method to find which of dive3 and dive31 has higher contrast across each color histogram.

   Let us make dive3 as the reference image 'refim'.

$$refim = dive3; \tag{34}$$

2. **Find c and d for channels using min-max method**
   Use RGB from reference image to find c and d for min-max method. Since our images are 8-bit, our pixel value can be $2^8 = 256$. We choose the lower limit as a=0 and upper limit as b=255.
   Now to find the lowest (c) and highest (d) pixel values, we will have to find lowest and highes pixel value from each channel of our reference image. We do this as below,

$$c_{red} = min(min(red3)); d_{red} = max(max(red3)) \tag{35}$$

$$c_{green} = min(min(green3)); d_{green} = max(max(green3)) \tag{36}$$

$$c_{blue} = min(min(blue3)); d_{blue} = max(max(blue3)) \tag{37}$$

   I tried to re-run equations 35, 36 and 37 to image dive31 and calculate c and d values.

   The Michealson formula for each channel of two images gives below results.

   For image dive3, K_red = 199; K_green = 244; K_blue = 248.

   For image dive31, K_red = 92; K_green = 240; K_blue = 179.

   We see that dive3 shows a higher contrast than dive 31. This is the second reason we choose dive3 as the reference image.

   We now 'enhance' our reference image refim by scaling is using the function $new_{reference} = (refim - c)(b - a/d - c) + a$.

   Also, we need to apply this function across all rows and images of the refim which has 1536X2048X3 size.

   Running a loop 1536X6144 (2048X3=6144) times to scan three seperate channels,
   for i = 1:1536
   for j = 1:6144

$$new_{red}(i, j) = (refim(i, j) - c_{red}).*((b-a)/((0.95*d_{red}) - c_{red})) + a \tag{38}$$

$$new_{green}(i, j) = (refim(i, j) - c_{green}).*((b-a)/((0.95*d_{green}) - c_{green})) + a \tag{39}$$

$$new_{blue}(i, j) = (refim(i, j) - c_{blue}).*((b-a)/((0.95*d_{blue}) - c_{blue})) + a \tag{40}$$

end
end

Also, we take 95 percentile of the d values to remove the outlying pixels. This prevents outliers affecting the scaling.

Our min-max enhanced reference channels are now new_{red}, new_{green} and new_{blue}.

3. **Gamma Stretching in channels**
The images are dark in general. There is a need to tone down the images by choosing a $\gamma < 1$. Of personal choice, $\gamma = 0.6$ gives a good light shade.

Gamma stretching in Matlab is done using 'imadjust' command. See below,

$$red_{gamma} = imadjust(new_{red}, [c_{red}/255\ d_{red}/255], [0\ 1], 0.6) \tag{41}$$

$$green_{gamma} = imadjust(new_{green}, [c_{green}/255\ d_{green}/255], [0\ 1], 0.6) \tag{42}$$

$$blue_{gamma} = imadjust(new_{blue}, [c_{blue}/255\ d_{blue}/255], [0\ 1], 0.6) \tag{43}$$

$$ref_{gamma} = cat(3, red_{gamma}, green_{gamma}, blue_{gamma}); \tag{44}$$

The final enhanced image can now be made using reference gamma channels. We call this as ref_{gamma} image.

$$ref_{gamma} = cat(3, red_{gamma}, green_{gamma}, blue_{gamma}) \tag{45}$$

4. **Histogram Equalization**
One last step in pre-processing steps is to equalize the original images based on the reference histogram of ref_{gamma} image. We call this reference histogram as 'refhist'.

$$refhist = imhist(ref_{gamma}); \tag{46}$$

See $\mathbf{Figure'ReferenceHistogram'}$ in results section.

Feed in the original images to Matlab command 'histeq',

$$eq2 = histeq(dive2, refhist) \tag{47}$$

$$eq3 = histeq(dive3, refhist) \tag{48}$$

$$eq4 = histeq(dive4, refhist) \tag{49}$$

$$eq31 = histeq(dive31, refhist) \tag{50}$$

$$eq39 = histeq(dive39, refhist) \tag{51}$$

2.3 Show processed images and their histograms

We will show the original image, processed image, original histogram and the equalized histogram using code below.

$$subplot(2,2,1), imshow(dive2), title('dive2') \tag{52}$$

$$subplot(2,2,2), imshow(eq2), title('Processed\ Image') \tag{53}$$

$$subplot(2,2,3), imhist(dive2), title(Original\ Histogram') \tag{54}$$

$$subplot(2,2,4), imhist(eq2), title('Equalized\ Histogram') \tag{55}$$

Same code is used on all other original images.

- See **Figure5** in results section for the resulting image and the equalized histogram. We can see there are no pixels in the 0-9 range. This is because the 'Reference Histogram' starts around pixel value 9. The shape of the equalized histogram follows the shape of the original histogram. Thus equalized histogram is shifted into a more middle zone.

- **Figure6** looks much lighter since the equalized histogram is shifted to a much brighter intensity range. The blue and red colors are more visible in the processed image.
- **Figure7** shows a yellow tinge on the top left side. This is because the red histogram which was prominent in the original dive4 image is now toned down. The green histogram which was prominent in the 150-200 range is retained in the equalized histogram. This combined effect makes the top left side yellow.
- **Figure8** has no high pixel intensities in the 0-14 range. This is because the prominent red histogram of dive31 is now shifted more towards the center.The shape of the original histogram is kind of distorted in 150-175 range. This is because there is a dip in the shape of original histogram whereas the reference histogram is relatively high intensity in this range.
- **Figure9** is the most unsatisfactory of all results. This is because the red histogram is peaking around the beginning of the histogram. This red is negligible after a value of 25 on x-axis. This effect can be seen in the processed image being much lighter than the original dive39.

  The blue histogram of dive39 is again peaking at x-value 25. It has a bell curve in the range 0-100 and negligible after that. Histogram equalization removes much of the blue effect from the image.

The green histogram of dive39 does not have high pixel intensities although spread in a broad range.

Due to histogram equalization, the final histogram begins at x-value 25. The red pick is visible at pixel value 25. The result is a faded image with shades of blue and green. The image also looks distorted due to noise appearing as dots on the water surface.

3 Results

Figure 1 Image Histograms

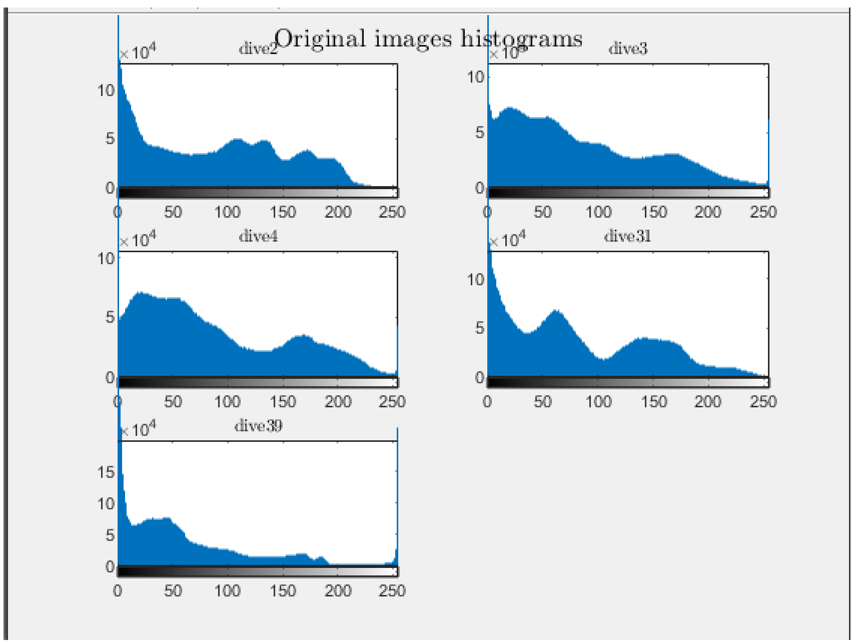


Figure 2 Red Histograms

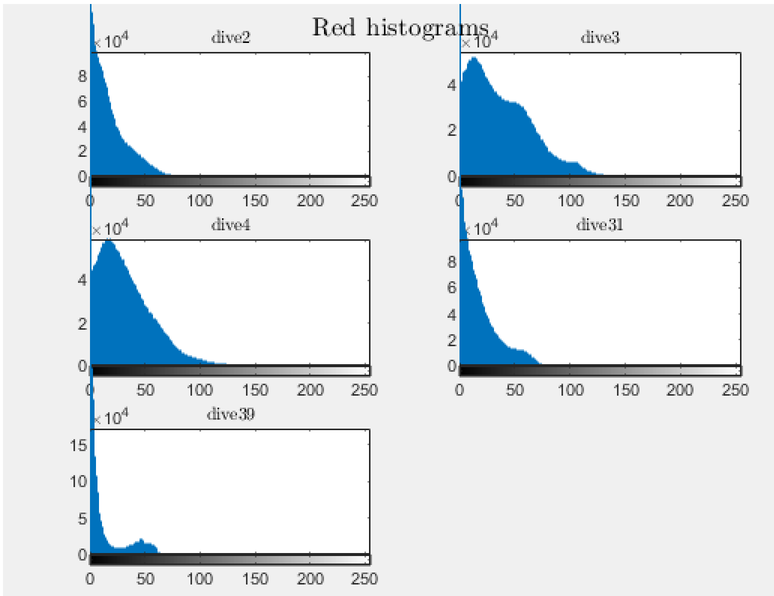


Figure 3 Green Histograms

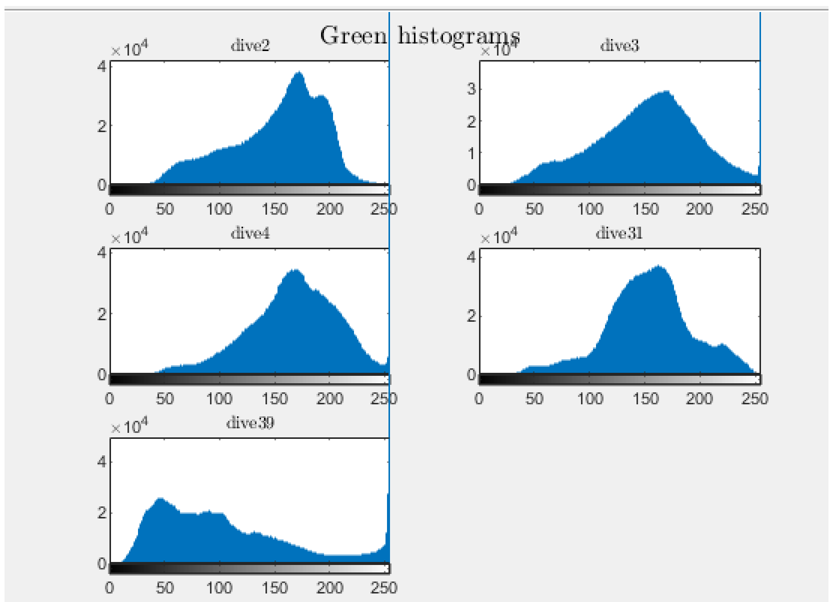


Figure 4 Blue Histograms

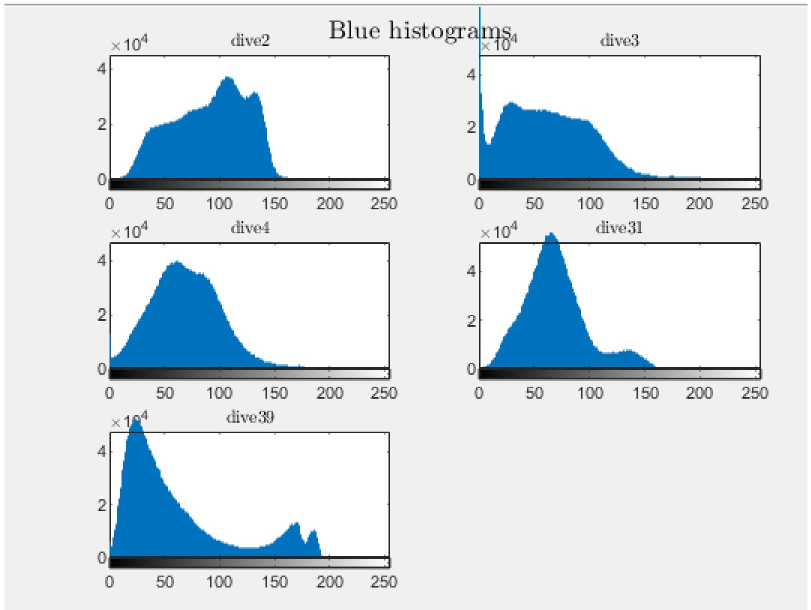


Figure Reference Histogram

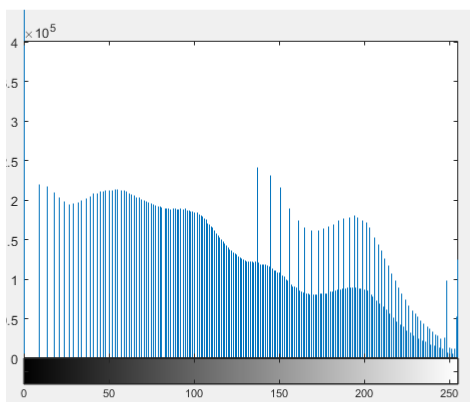


Figure 5 dive2

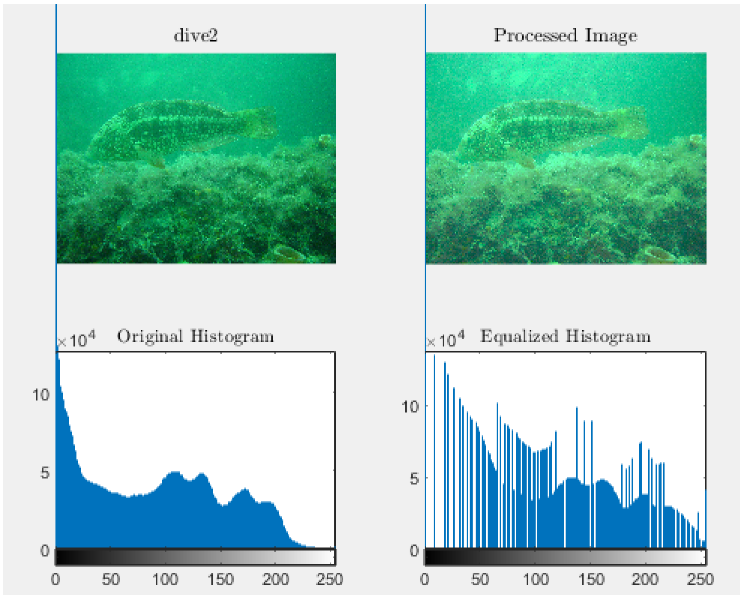


Figure 6 dive3

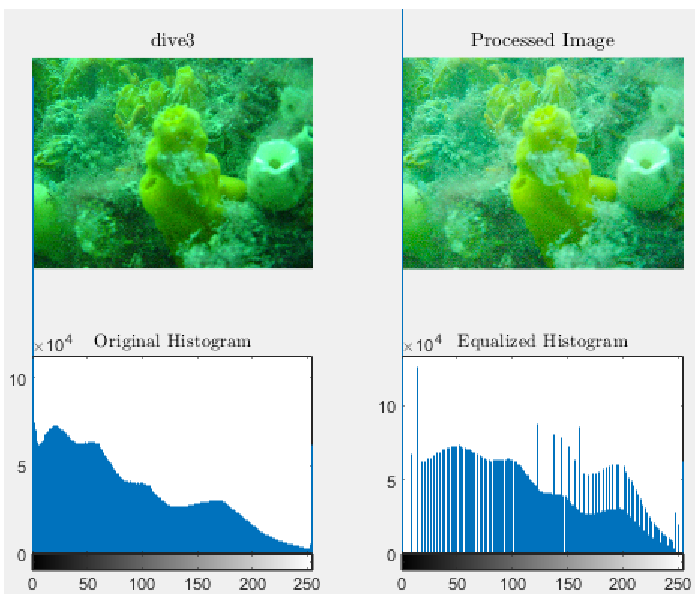


Figure 7 dive4

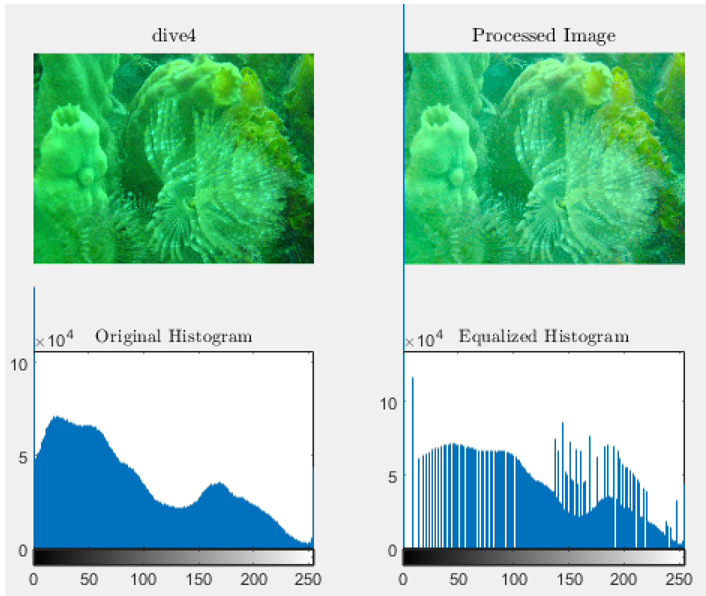


Figure 8 dive31

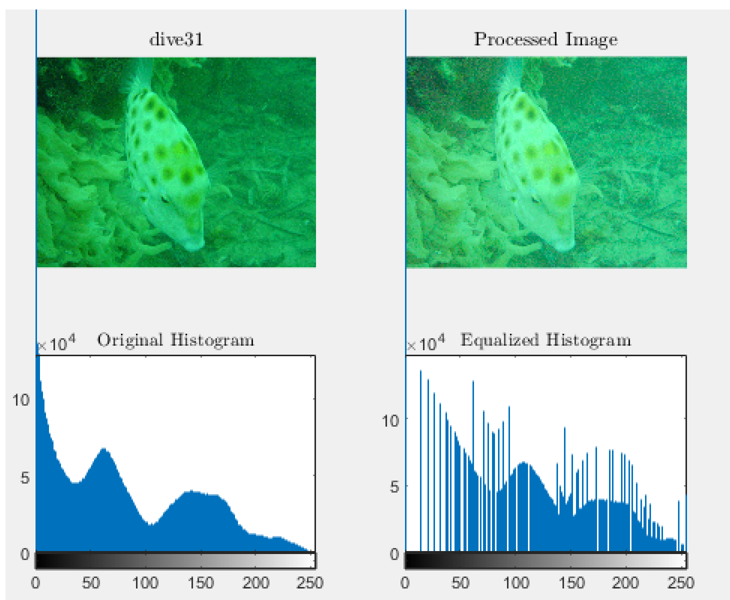


Figure 9 dive39

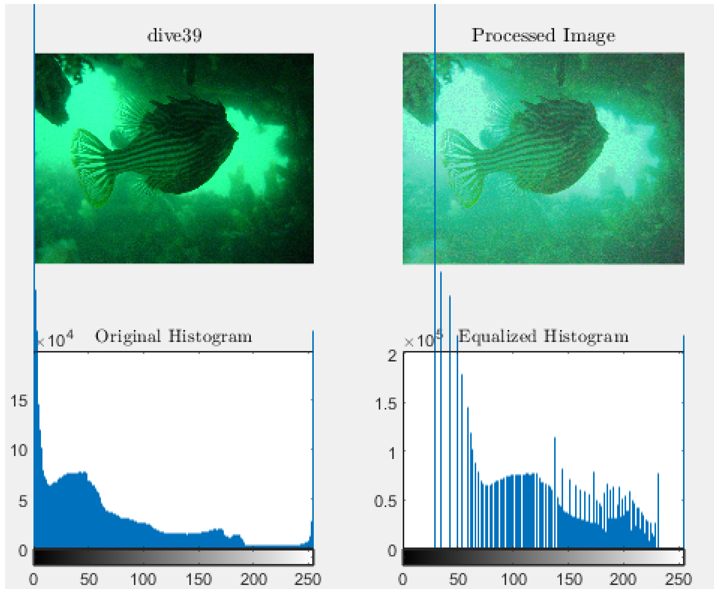
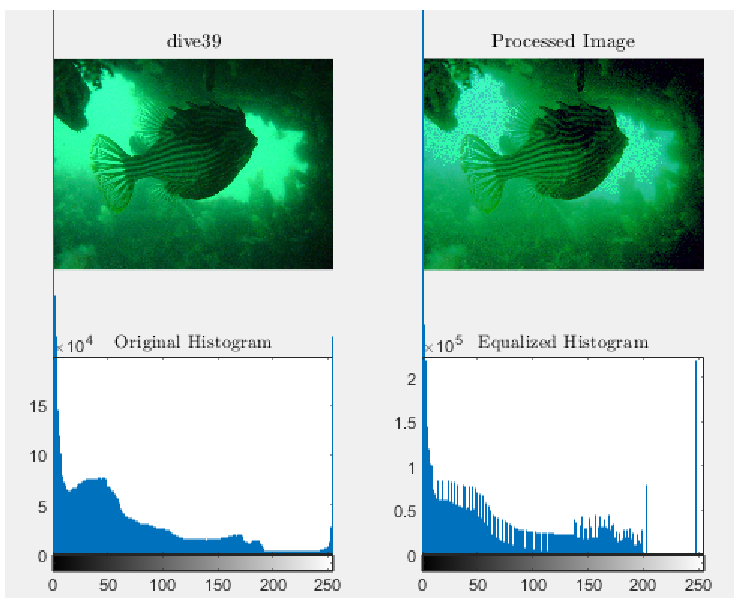


Figure 10 dive39 reworked



4 Discussion

Focusing on original image dive39, we see that it is a dark image. The enhancement removes important contrast features from the image. Keeping the red and blue channel intact, we re-work only on the green image and see the results in Figure 10.
The image has better contrast now but the dots like noise on the surface of water is still present. It is the noise in the red channel which contributes to these dots.
See 'Appendix' section for the 'Rework on image dive39' for code details.

We can say that one doesnot need to stretch all the bands since it can remove the contrast features from an image. Also, all the three channels should not be contrast enhanced as in our dive39 example, the image can fade due to enhancement of all channels.

5 Appendix

```
set(0, 'defaulttextinterpreter','Latex');

%% Read in images
dive2 = imread('dive2.jpg');
dive3 = imread('dive3.jpg');
dive4 = imread('dive4.jpg');
dive31 = imread('dive31.jpg');
dive39 = imread('dive39.jpg');

figure(1),
subplot(3,2,1),imhist(dive2),title('dive2');
subplot(3,2,2),imhist(dive3),title('dive3');
subplot(3,2,3),imhist(dive4),title('dive4');
subplot(3,2,4),imhist(dive31),title('dive31');
subplot(3,2,5),imhist(dive39),title('dive39');
sgtitle('Original images histograms');

%%Seperate the RGB channels from each image
[m2,n2] = size(dive2);
red2 = dive2(:,:,1);
green2 = dive2(:,:,2);
blue2 = dive2(:,:,3);

[m3,n3] = size(dive3);
red3 = dive3(:,:,1);
green3 = dive3(:,:,2);
```

```
blue3 = dive3(:,:,3);

[m4,n4] = size(dive4);
red4 = dive4(:,:,1);
green4 = dive4(:,:,2);
blue4 = dive4(:,:,3);

[m31,n31] = size(dive31);
red31 = dive31(:,:,1);
green31 = dive31(:,:,2);
blue31 = dive31(:,:,3);

[m39,n39] = size(dive39);
red39 = dive39(:,:,1);
green39 = dive39(:,:,2);
blue39 = dive39(:,:,3);

%%Pick the best R,G and B histogram amongst all images
figure(2),
subplot(3,2,1),imhist(red2),title('dive2');
subplot(3,2,2),imhist(red3),title('dive3'); %best
subplot(3,2,3),imhist(red4),title('dive4');
subplot(3,2,4),imhist(red31),title('dive31');
subplot(3,2,5),imhist(red39),title('dive39');
sgtitle('Red histograms');

figure(3),
subplot(3,2,1),imhist(green2),title('dive2');
subplot(3,2,2),imhist(green3),title('dive3');
subplot(3,2,3),imhist(green4),title('dive4');
subplot(3,2,4),imhist(green31),title('dive31'); %best
subplot(3,2,5),imhist(green39),title('dive39');
sgtitle('Green histograms');

figure(4),
subplot(3,2,1),imhist(blue2),title('dive2');  %best
subplot(3,2,2),imhist(blue3),title('dive3');
subplot(3,2,3),imhist(blue4),title('dive4');
subplot(3,2,4),imhist(blue31),title('dive31');
subplot(3,2,5),imhist(blue39),title('dive39');
sgtitle('Blue histograms');

%% Make reference image from best R,G and B histograms
refim = dive3;
%% Use RGB from reference image to find c and d for
   min-max method
```

```
a=0;
b=255;
c_red = min(min(red3)); d_red = max(max(red3));
c_green = min(min(green3)); d_green = max(max(green3))
   ;
c_blue = min(min(blue3)); d_blue = max(max(blue3));

    for i = 1:1536                                  %
       m2=m3=m4=m31=m39=1536
        for j = 1:6144                              %
           n2=n3=n4=n31=n39=6144
            new_red(i,j)=(refim(i,j)-c_red).*((b-a)
               /((0.95*d_red)-c_red))+a;
            new_green(i,j) = (refim(i,j)-c_green).*((b
               -a)/((0.95*d_green)-c_green))+a;
            new_blue(i,j) = (refim(i,j)-c_blue).*((b-a
               )/((0.95*d_blue)-c_blue))+a;
        end
    end

%% Use these c and d values for gamma stretching in
   channels
%red_gamma = imadjust(new_red,[c_red/255 d_red/255],[0
    1],0.6);
red_gamma = imadjust(new_red,[c_red/255 d_red/255],[0
   1],0.6);
green_gamma = imadjust(new_green,[c_green/255 d_green
   /255],[0 1],0.6);
blue_gamma = imadjust(new_blue,[c_blue/255 d_blue
   /255],[0 1],0.6);
ref_gamma = cat(3,red_gamma,green_gamma,blue_gamma);

%% Make a reference histogram and use histogram
   equalization.
refhist = imhist(ref_gamma);
eq2 = histeq(dive2,refhist);
eq3 = histeq(dive3,refhist);
eq4 = histeq(dive4,refhist);
eq31 = histeq(dive31,refhist);
eq39 = histeq(dive39,refhist);

%% Show processed images and their histograms
figure(5),
subplot(2,2,1),imshow(dive2),title('dive2');
subplot(2,2,2),imshow(eq2),title('Processed Image');
subplot(2,2,3),imhist(dive2), title('Original
```

```
    Histogram');
subplot(2,2,4),imhist(eq2),title('Equalized Histogram'
    );

figure(6)
subplot(2,2,1),imshow(dive3),title('dive3');
subplot(2,2,2),imshow(eq3),title('Processed Image');
subplot(2,2,3),imhist(dive3), title('Original
    Histogram');
subplot(2,2,4),imhist(eq3),title('Equalized Histogram'
    );

figure(7),
subplot(2,2,1),imshow(dive4),title('dive4');
subplot(2,2,2),imshow(eq4),title('Processed Image');
subplot(2,2,3),imhist(dive4), title('Original
    Histogram');
subplot(2,2,4),imhist(eq4),title('Equalized Histogram'
    );

figure(8),
subplot(2,2,1),imshow(dive31),title('dive31');
subplot(2,2,2),imshow(eq31),title('Processed Image');
subplot(2,2,3),imhist(dive31), title('Original
    Histogram');
subplot(2,2,4),imhist(eq31),title('Equalized Histogram
    ');

figure(9),
subplot(2,2,1),imshow(dive39),title('dive39');
subplot(2,2,2),imshow(eq39),title('Processed Image');
subplot(2,2,3),imhist(dive39), title('Original
    Histogram');
subplot(2,2,4),imhist(eq39),title('Equalized Histogram
    ');

%% Rework on image dive39
greenhist = imhist(green_gamma);
dive39_green = histeq(green39,greenhist);

%bluehist = imhist(blue_gamma);
%dive39_blue = histeq(blue39, bluehist);

%redhist = imhist(red_gamma);
%dive39_red = histeq(red39, redhist);
```

```
dive39_eq = cat(3,red39,dive39_green,blue39);

figure(10),
subplot(2,2,1),imshow(dive39),title('dive39');
subplot(2,2,2),imshow(dive39_eq),title('Processed
   Image');
subplot(2,2,3),imhist(dive39), title('Original
   Histogram');
subplot(2,2,4),imhist(dive39_eq),title('Equalized
   Histogram');
```