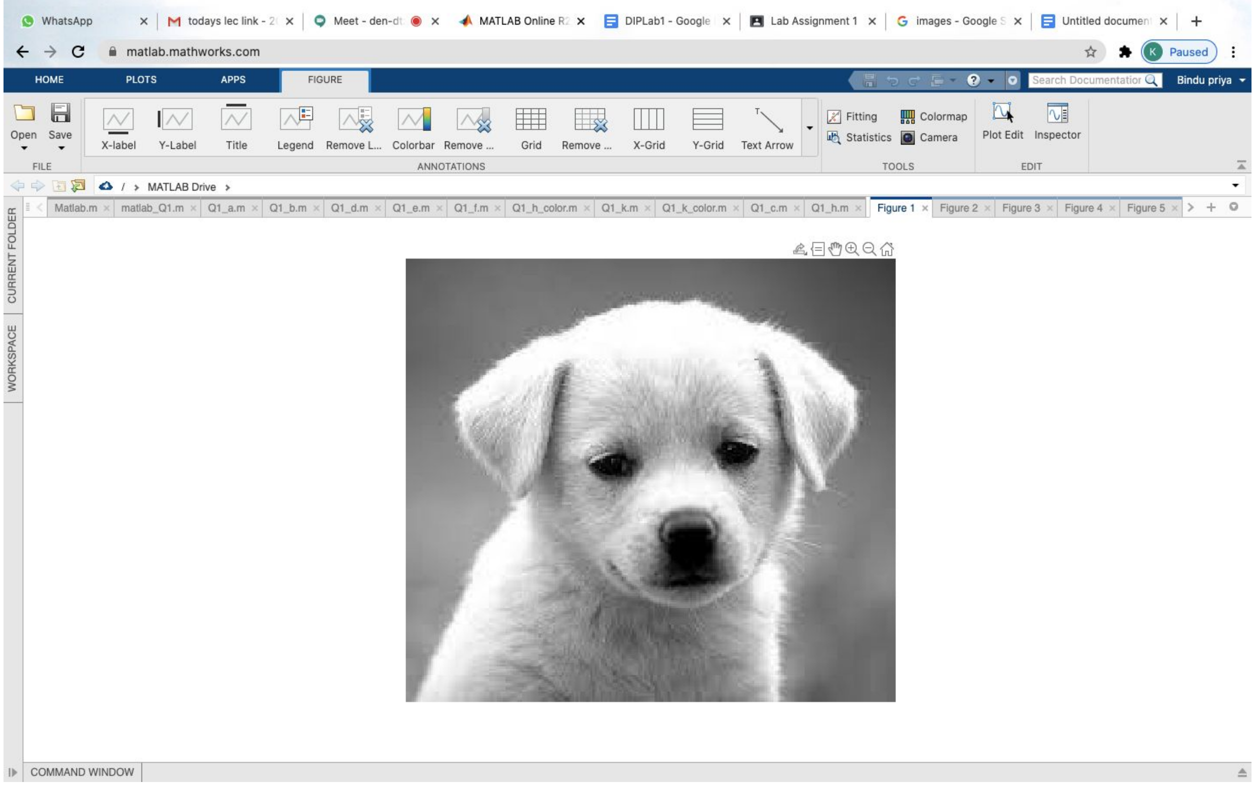
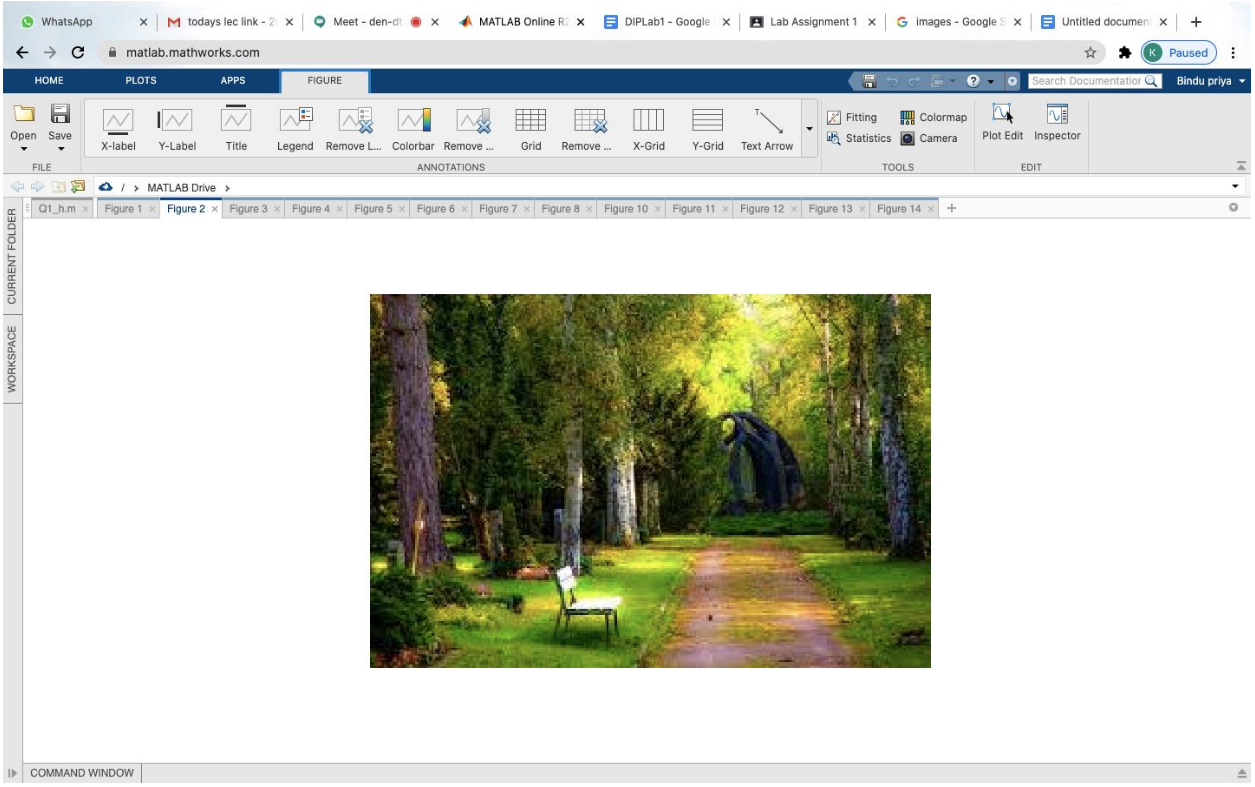
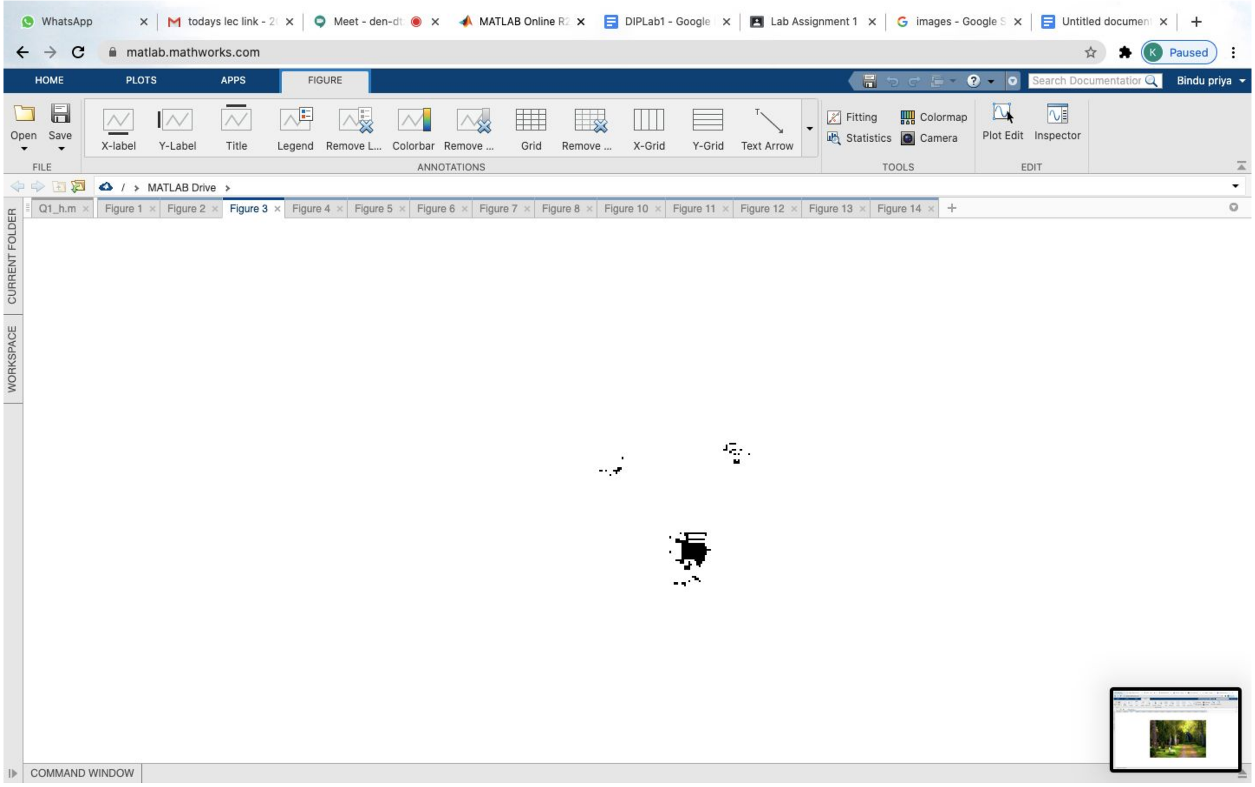
```
% a. read an image
im_grey = imread("grey_img.jpg");
im_color = imread("color.jpg");


%Display an image
figure(1)
Im_grey = imshow(im_grey);
```



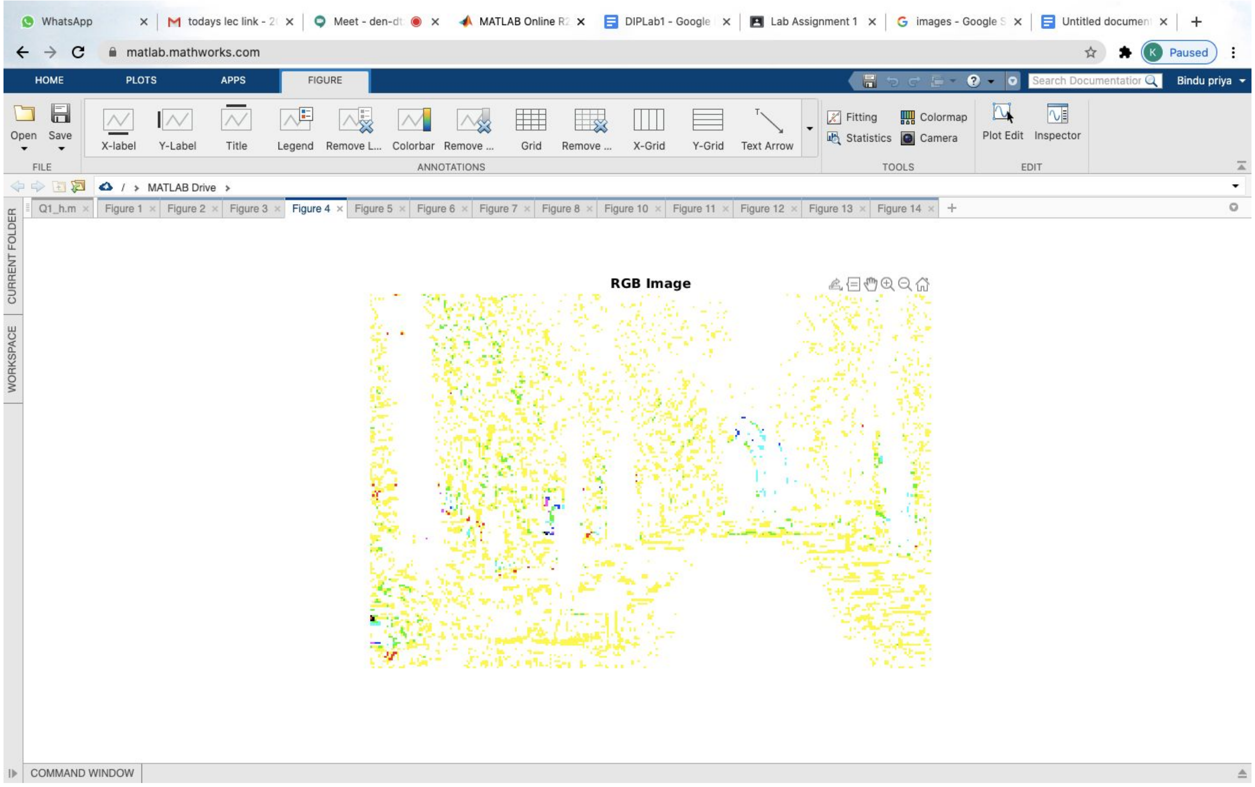
```
figure(2)
Im_color = imshow(im_color);
```



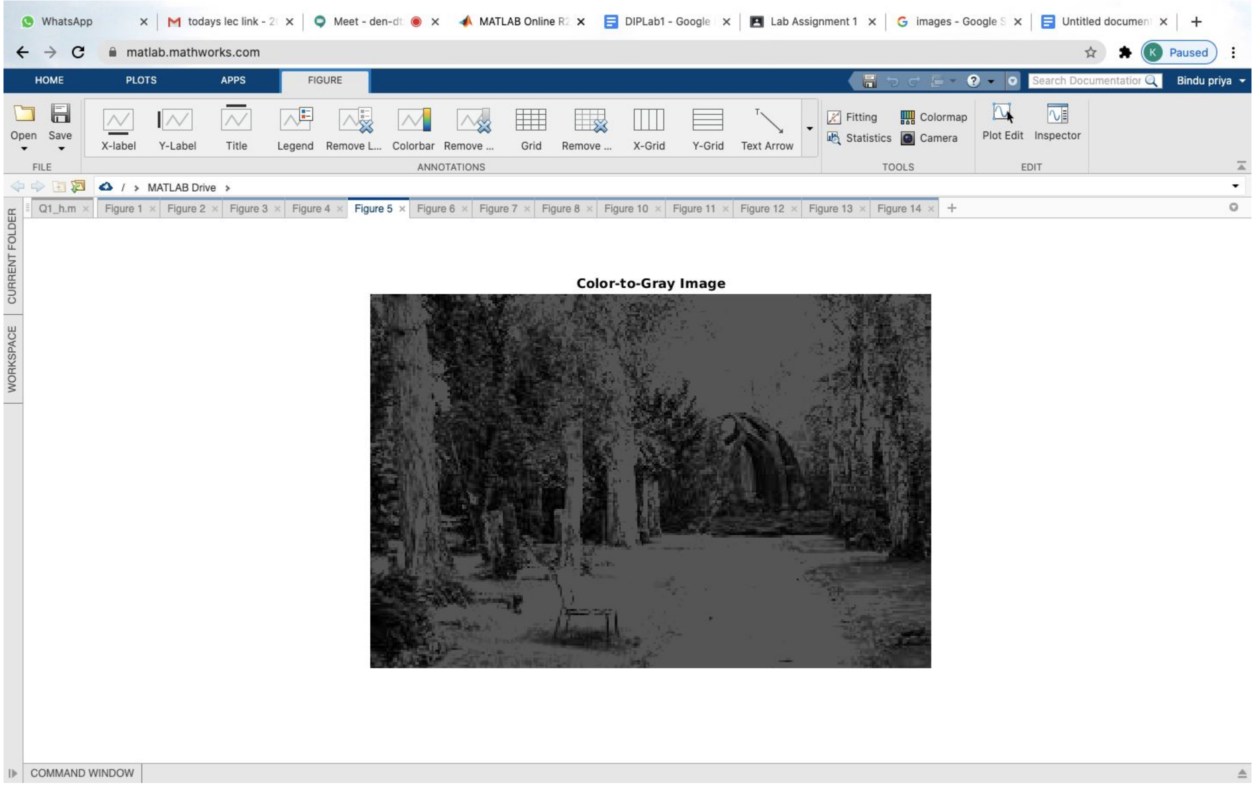
```
% c. write an image with another name
imwrite(im_grey, "Grey_img.jpg");
imwrite(im_color, "Color_img.jpg");

Im_grey = imread("Grey_img.jpg");
Im_color = imread("Color_img.jpg");


% d. display information of image
Info_grey = imagemodel(Im_grey);
Info_color = imagemodel(Im_color);


%e.Convert data class of image for greyscale
convert_grey = cast(im_grey, 'double');
figure(3)
image_grey = imshow(convert_grey);
```



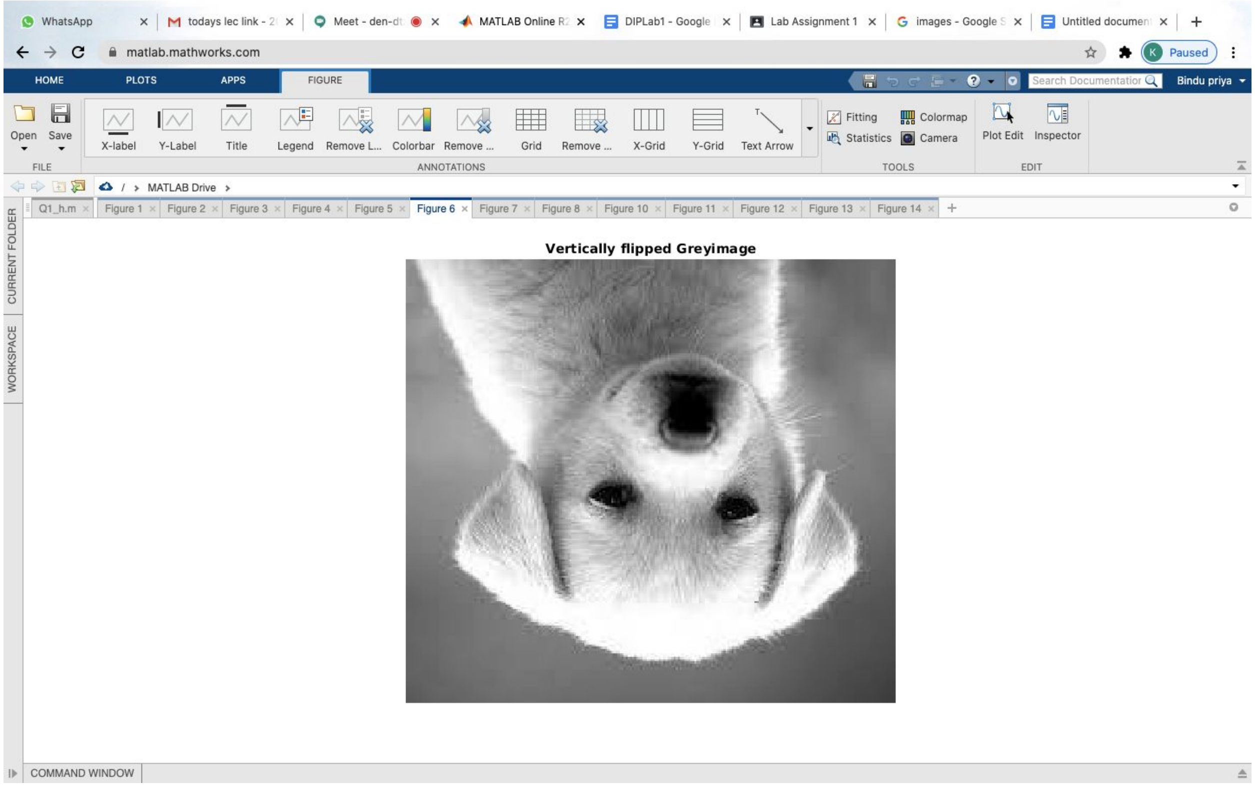
```matlab
%Convert data class of image for greyscale
convert_color = cast(im_color, 'double');
figure(4)
image_color = imshow(convert_color);
```



```
% f. convert types of image from one to another
% for grey image
title('RGB Image');
R = im_color(:, :, 1);
G = im_color(:, :, 2);
B = im_color(:, :, 3);
avg = (R + G + B) / 3;
figure(5)
imshow(avg); title('Color-to-Gray Image');
```



```
% g. flip an image
%for grey scale
[x, y, z] = size(Im_grey);
for plane = 1 : z
   len = x;
   for i = 1 : x
       for j = 1 : y
           if i < x/2
               temp = Im_grey(i, j, plane);
               Im_grey(i, j, plane) = Im_grey(len, j, plane);
               Im_grey(len, j, plane) = temp;
           end
       end
       len = len - 1;
   end
end
figure(6)
imshow(Im_grey);
title('Vertically flipped Greyimage');
```

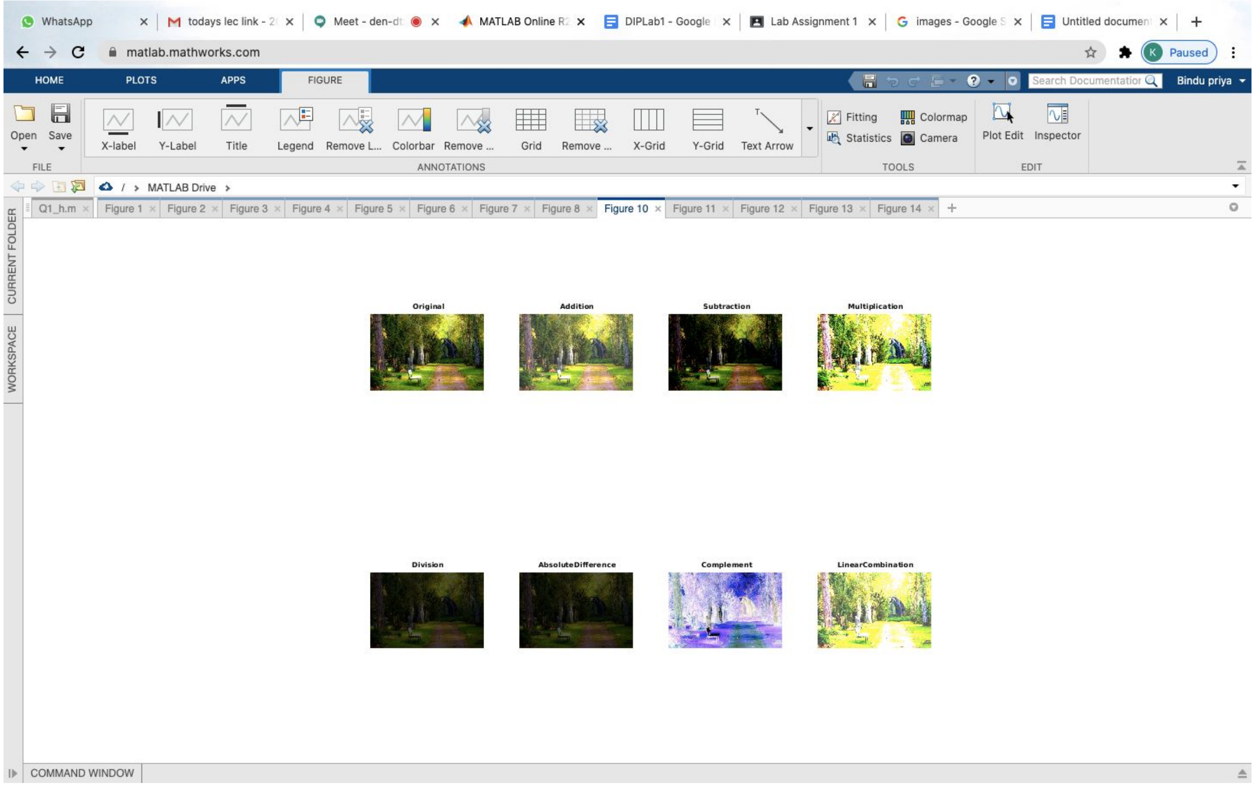


```
% g. flip an image
%for COLOR scale
[x, y, z] = size(Im_color);
for plane = 1 : z
   len = x;
   for i = 1 : x
       for j = 1 : y
           if i < x/2
               temp = Im_color(i, j, plane);
               Im_color(i, j, plane) = Im_color(len, j, plane);
               Im_color(len, j, plane) = temp;
           end
       end
       len = len - 1;
   end
end
figure(7)
imshow(Im_color);
title('Vertically flipped colorimage');
```

```
% h. Perform Arithmetic operation (Addition, Subtraction, Multiplication,
Division, Absolute Difference, Complement, Linear combination)
function Arithmetic_grey()
   im = imread("grey_img.jpg");
   figure
   subplot(2,4,1)
   imshow(im);
   title("\fontsize{6} Original")
   subplot(2,4,2)
   imshow(im + 50)
   title("\fontsize{6} Addition")
   subplot(2,4,3)
   imshow(im - 50);
   title("\fontsize{6} Subtraction")
   subplot(2,4,4)
   imshow(im * 3);
   title("\fontsize{6} Multiplication")
   subplot(2,4,5)
   imshow(im / 3);
   title("\fontsize{6} Division")
   subplot(2,4,6)
   imshow(im / 3);
   title("\fontsize{6} AbsoluteDifference")
   subplot(2,4,7)
   imshow(255 - im);
   title("\fontsize{6} Complement")
   subplot(2,4,8)
   imshow(2 * im + 70);
   title("\fontsize{6} LinearCombination")
end
```

```
% h. Perform Arithmetic operation (Addition, Subtraction, Multiplication,
Division, Absolute Difference, Complement, Linear combination)
%color image
function Arithmetic_color()
```

```
   im = imread("color.jpg");
   figure
   subplot(2,4,1)
   imshow(im);
   title("\fontsize{6} Original")
   subplot(2,4,2)
   imshow(im + 50)
   title("\fontsize{6} Addition")
   subplot(2,4,3)
   imshow(im - 50);
   title("\fontsize{6} Subtraction")
   subplot(2,4,4)
   imshow(im * 3);
   title("\fontsize{6} Multiplication")
   subplot(2,4,5)
   imshow(im / 3);
   title("\fontsize{6} Division")
   subplot(2,4,6)
   imshow(im / 3);
   title("\fontsize{6} AbsoluteDifference")
   subplot(2,4,7)
   imshow(255 - im);
   title("\fontsize{6} Complement")
   subplot(2,4,8)
   imshow(2 * im + 70);
   title("\fontsize{6} LinearCombination")
end
```



```
im = imread("grey_img.jpg");
[rows, cols] = size(im);
sum = 0;
for i = 1:rows
   for j = 1:cols
        sum = sum + im(i,j);
   end
end
avg = sum / 3;

figure(11)
imshow(avg); title('average value of an GreyImage');


%k.Compute average value of an image with help of conditional loop.
%for grey scale
im = imread("grey_img.jpg");
[rows, cols] = size(im);
sum = 0;
for i = 1:rows
   for j = 1:cols
        sum = sum + im(i,j);
   end
end
```

```
avg = sum / 3;

figure(11)
imshow(avg); title('average value of an GreyImage');



%k.Compute average value of an image with help of conditional loop.
%for color image
im = imread("color.jpg");
A = imread("color.jpg");
R = A(:, :, 1);
G = A(:, :, 2);
B = A(:, :, 3);

[rows, cols] = size(R);
sum = 0;
for i = 1:rows
   for j = 1:cols
       sum = sum + R(i,j);
   end
end
avg_R = sum / 3;

figure(12)
imshow(avg_R); title('average value of an RedImage');

[rows_G, cols_G] = size(G);
sum = 0;
for i = 1:rows_G
   for j = 1:cols_G
       sum = sum + G(i,j);
   end
end
avg_G = sum / 3;

figure(13)
imshow(avg_G); title('average value of an GreenImage');

[rows_B, cols_B] = size(B);
sum = 0;
for i = 1:rows_B
   for j = 1:cols_B
       sum = sum + B(i,j);
   end
end
avg_B = sum / 3;
```

```
figure(14)
imshow(avg_B); title('average value of an BlueImage');
```