

T.C.
YOZGAT BOZOK ÜNİVERSİTESİ
MÜHENDİSLİK MİMARLIK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ

Bilgisayar Görmesi Dersi

Konu: İşlenen son derse kadar gösterilmiş olan görüntü iyileştirme tekniklerinin uygulanması ve açıklanması.

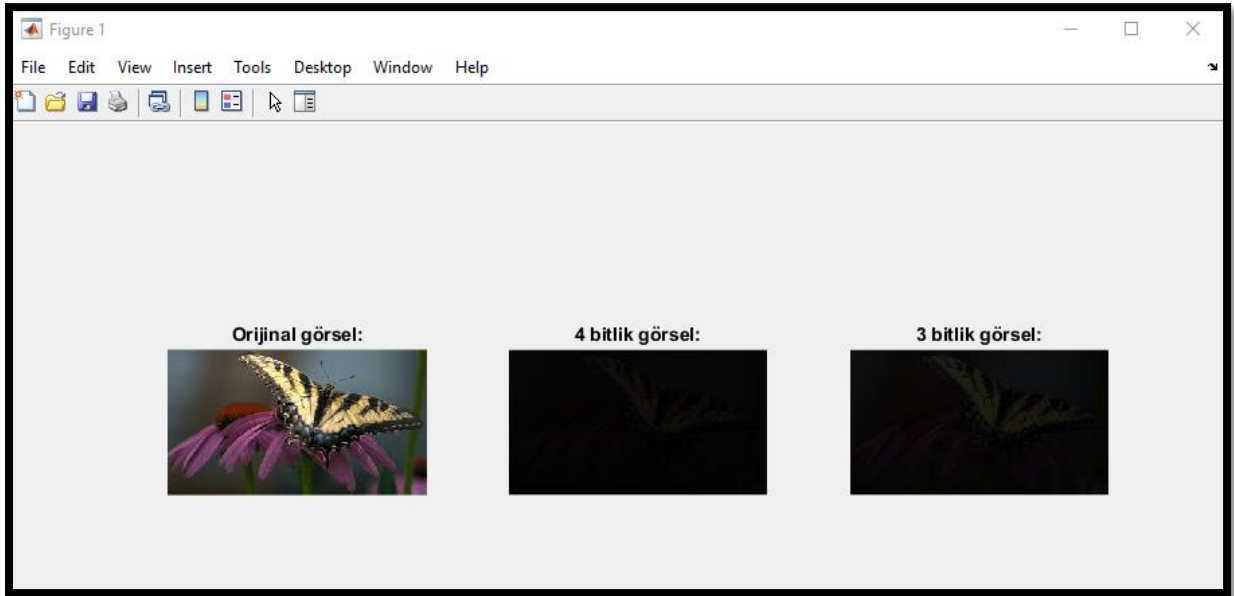
Hazırlayan

Adı Soyadı: Esra Yüce

Öğrenci No: 16008117051

Bit Çevirme

```
main.m X +
3 %% Bit Çevirme
4 X = imread('image.jpg'); % imread komutu ile dizindeki image adlı görsel alındı ve X değişkenine atandı.
5 image = X./16; % X eğişkenindeki görsel 4 bitlik görüntüye çevirildi.
6 image2 = X./8; % X eğişkenindeki görsel 3 bitlik görüntüye çevirildi.
7
8 subplot(1,3,1);
9 % tek tek pencere açmak yerine aynı pencere üzerinde birden fazla görsel...
10 % görüntüleyebilmek için subplot komutu kullanıldı.
11 imshow(X);
12 title('Orişinal görsel:')
13
14 subplot(1,3,2);
15 imshow(image,[]);
16 title('4 bitlik görsel:')
17
18 subplot(1,3,3);
19 imshow(image2,[]);
20 title('3 bitlik görsel:')
```



imread komutu ile dizindeki image adlı görsel alındı ve X değişkenine atandı.

Yukarıdaki şekilde X eğişkenindeki görselin 4 ve 3 bitlik görüntüye çevirilmiş halleri görülmektedir. Görsellerin ekrana basılmasında subplot komutu kullanıldı.

Subplot, Tek tek pencere açmak yerine aynı pencere üzerinde birden fazla görsel ve histogramı görüntüleyebilmek için kullanılan bir komuttur.

Colormap

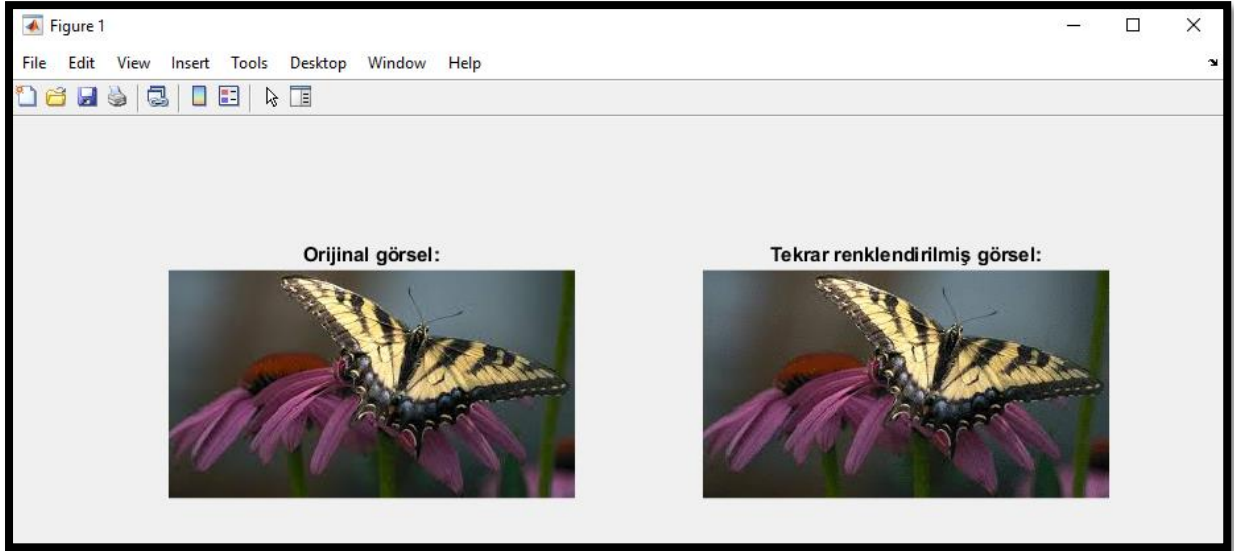
colormap map mevcut şeklin renk haritasını önceden tanımlanmış renk haritalarından birine ayarlar. Şeklin renk haritasını ayarlarsanız, şekildeki eksenler ve grafikler aynı renk haritasını kullanır. Yeni renk haritası, mevcut renk haritasıyla aynı uzunluktadır (renk sayısı). Bu sözdizimini kullandığınızda, renk eşlemesi için özel bir uzunluk belirleyemezsiniz.

Aynı şekilde mevcut bir görselden de renk haritası oluşturulup başka görseller için kullanılabilir.

```

23 %% Colormap 1
24 I = imread('image.jpg'); % imread komutu ile dizindeki image adlı görsel alındı ve I değişkenine atandı.
25 [indexed,map] = rgb2ind(I,255);
26 % I değişkenindeki görsel rgb2ind komutu ile indeksli hale dönüştürüldü ve map komutu ile renk haritası(10 renk) alındı.
27 I2 = ind2rgb(indexed,map);
28 % indeksli görsel ind2rgb komutu ile renkli görüntüye çevrilerek map'in içerisindeki renklerle renklendirildi ve I2 değişkenine atandı.
29
30 subplot(1,2,1); % tek tek pencere açmak yerine aynı pencere üzerinde birden fazla görsel görüntüleyebilmek için subplot komutu kullanıldı.
31 imshow(I); % orjinal görsel ekrana basıldı.
32 title('Orijinal görsel:');
33
34 subplot(1,2,2);
35 imshow(I2); % tekrar renklendirilen görsel ekrana basıldı.
36 title('Tekrar renklendirilmiş görsel:');

```



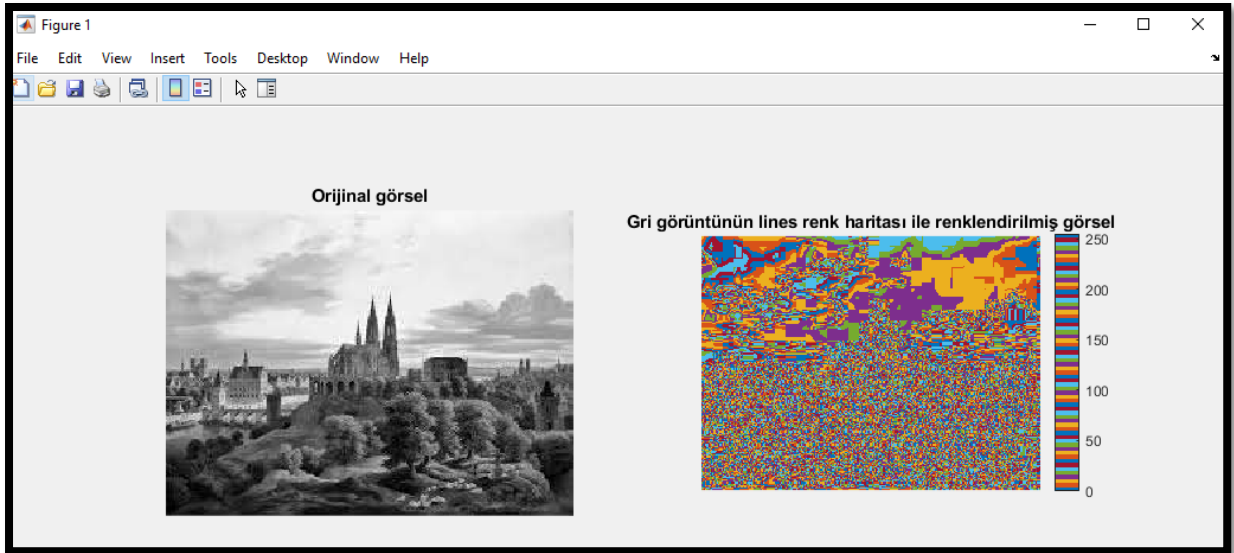
Yukarıdaki şekilde orjinal görsel rgb2ind komutu ile indeksli hale dönüştürüldü ve map komutu ile renk haritası (10 renk) alındı.

İndeksli görsel ind2rgb komutu ile renkli görüntüye çevrilerek map'in içerisindeki renklerle renklendirildi. Görsellerin ekrana basılmasında subplot komutu kullanıldı.

```

38 %% Colormap 2
39 A = imread('image4.jpg'); % imread komutu ile dizindeki image4 adlı görsel alındı ve A değişkenine atandı.
40
41 subplot(1,2,1); % tek tek pencere açmak yerine aynı pencere üzerinde birden fazla görsel görüntüleyebilmek için subplot komutu kullanıldı.
42 imshow(A); % orjinal görsel ekrana basıldı.
43 title('Orijinal görsel:');
44
45 A = rgb2gray(A); % true color görüntü rgb2gray komutu ile gray görüntüye dönüştürüldü.
46
47 subplot(1,2,2);
48 imshow(A); % gri görüntü ekrana basıldı.
49 title('Gri görüntünün lines renk haritası ile renklendirilmiş görsel:');
50
51 colormap(lines); % açılan yeni pencerede gri görüntü lines renk haritasıyla renklendirildi.
52 colorbar;

```



Yukarıdaki şekilde true color görüntü `rgb2gray` komutu ile gray görüntüye dönüştürüldü.

Açılan yeni pencerede gri görüntü lines renk haritasıyla renklendirildi.

Görsellerin ekrana basılmasında subplot komutu kullanıldı.

Lines renk haritası, önceden belirlenmiş bir renk haritasıdır. Aşağıdaki renkleri içermektedir.



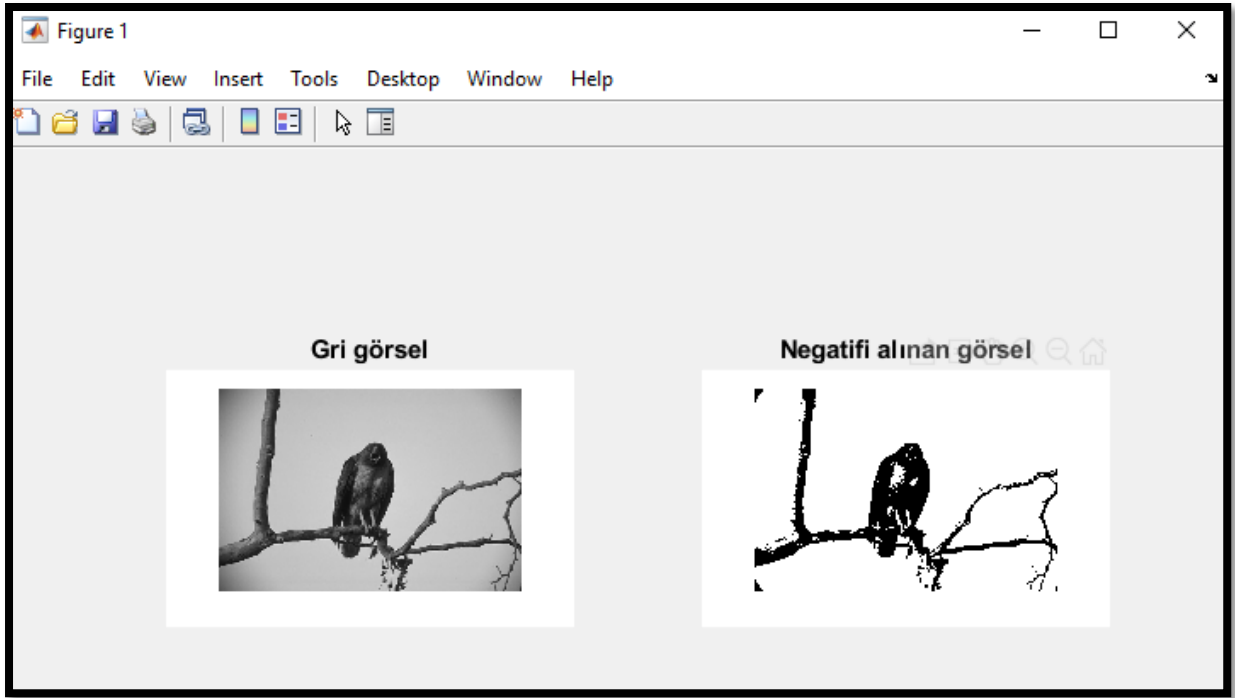
Görselin Negatifini Alma

Görüntünün bize daha fazla bilgi vermesi ve daha anlaşılır olması açısından negatifini alırız.

Resmin negatifi resmin piksel değerlerinin 255-maksimum koyuluk değeri ile farkının alınmasıdır.

Negatifi alınmış resimler bazı resimlerde insan algısı için çok net ifadeler oluşturmaya da işe yarar şekilde kullanılabilir. Örneğin, bazı resimlerde algılanması güç olan ayrıntılar negatif alma işleminden sonra daha kolay algılanabilir hale geliyor veya bazı kısımları daha anlaşılır olmasını sağlıyor.

```
54 %% Görselin Negatifini Alma
55 B = imread('image7.png'); % imread komutu ile dizindeki image7 adlı görsel alındı ve B değişkenine atandı.
56 B = rgb2gray(B); % true color görüntü rgb2gray komutu ile gray görüntüye dönüştürüldü ve B değişkenine atandı.
57 F = B>100; % 100'den büyük değerler 1, 100'den küçük değerler 0 yapıldı. Bu işlemle görüntünün negatifi alınmış oldu.
58
59 subplot(1,2,1); % tek tek pencere açmak yerine aynı pencere üzerinde birden fazla görsel görüntüleyebilmek için subplot komutu kullanıldı.
60 imshow(B); % gri görsel ekrana basıldı.
61 title('Gri görsel');
62
63 subplot(1,2,2);
64 imshow(F); % negatifi alınan görsel ekrana basıldı.
65 title('Negatifi alınan görsel');
```



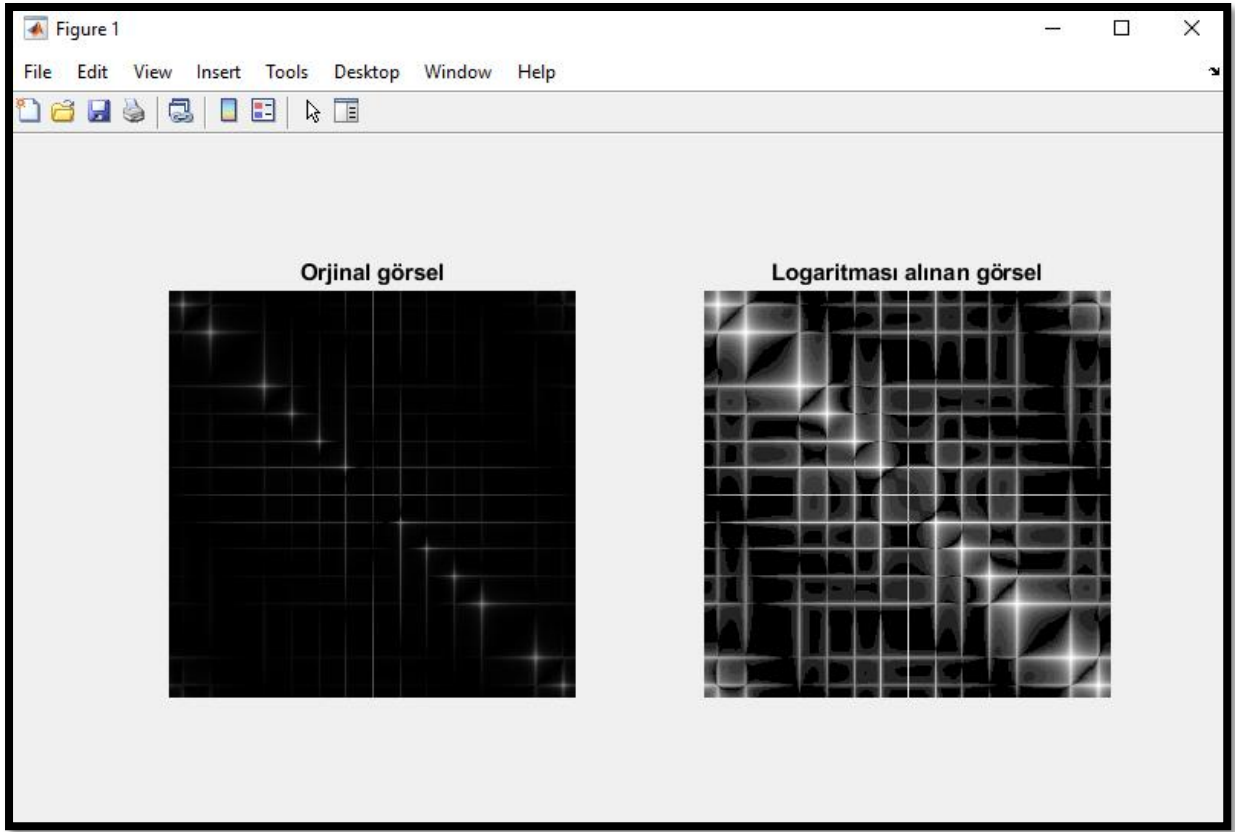
Yukarıdaki şekilde true color görüntü `rgb2gray` komutu ile gray görüntüye dönüştürüldü. 100'den büyük değerler 1, 100'den küçük değerler 0 olacak şekilde görüntünün negatifi alınmış oldu. Görsellerin ekrana basılmasında `subplot` komutu kullanıldı.

Log Transformation

Log dönüştürme, her bir x değişkenini bir $\log(x)$ ile değiştirdiği bir veri dönüştürme yöntemidir.

Orijinal sürekli verilerimiz çan eğrisini takip etmediğinde, bu verileri mümkün olduğunca "normal" hale getirmek için log işlemi uygulayarak dönüştürebiliriz, böylece bu verilerden elde edilen istatistiksel analiz sonuçları daha geçerli hale gelir.

```
66 %% Log Transformation
67
68 K = imread('image8.gif'); % imread komutu ile dizindeki image8 adlı görsel alındı ve K değişkenine atandı.
69
70 subplot(1,2,1);
71 % tek tek pencere açmak yerine aynı pencere üzerinde birden fazla görsel görüntüleyebilmek için subplot komutu kullanıldı.
72 imshow(K); % orjinal görsel ekrana basıldı.
73 title('Orjinal görsel');
74
75 K = K+1;
76 % logaritma 0 değerini almasın diye her piksel değerine 1 eklendi. 0 olan değerler 1 olmuş olup log(1) 0 değerine karşılık gelir.
77 K = im2double(K); % matematiksel işlemler için double veri türü uygundur. im2double komutu ile uint8'den double'a çevirildi.
78 K = log(K); % her değer logaritması alındı.
79 K = mat2gray(K); % matris 0 (siyah) ile 1 (beyaz) aralığında değerler içeren grayscale bir görüntüye dönüştürüldü.
80
81 subplot(1,2,2);
82 imshow(K); % Logaritması alınan görüntü ekrana yansıtıldı.
83 title('Logaritması alınan görsel');
```



Yukarıdaki şekilde log transformation işlemi uygulanarak gözle seçemediğimiz ayrıntılar daha anlaşılır hale getirildi.

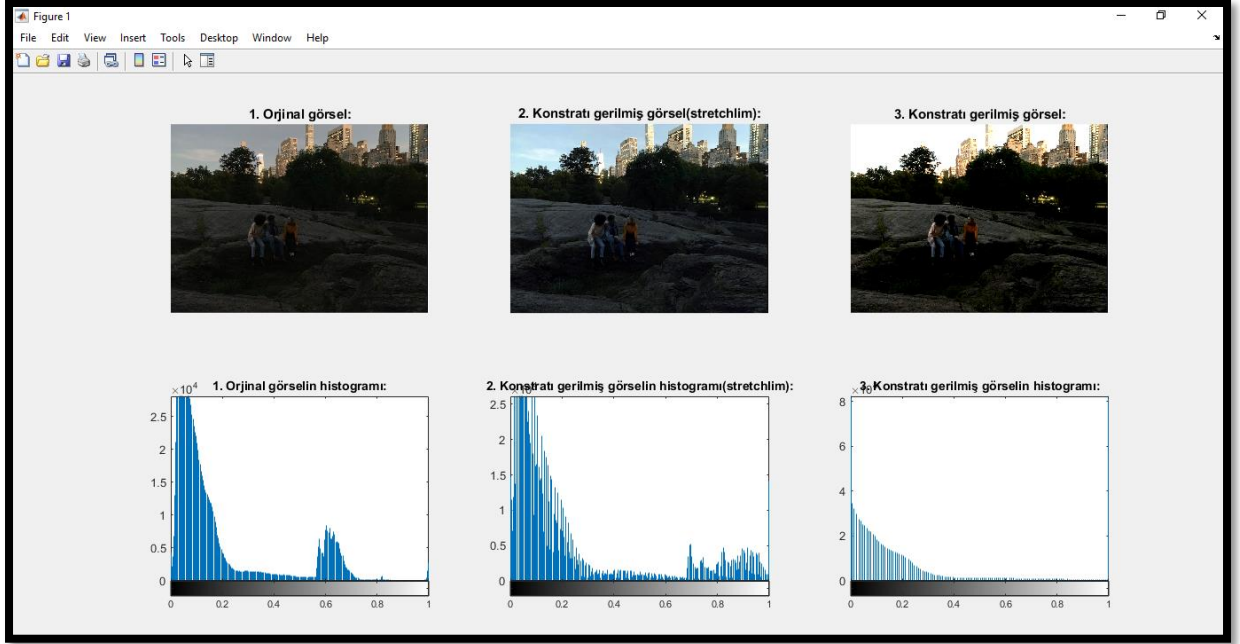
Logaritma 0 değerini almasın diye her piksel değerine 1 eklendi. 0 olan değerler 1 olmuş olup $\log(1)$ 0 değerine karşılık gelir.

`im2double` komutu ile `uint8`'den `double`'a çevirildi. Her değerın logaritması alındı. Matris 0 (siyah) ile 1 (beyaz) aralığında değerler içeren grayscale bir görüntüye dönüştürüldü.

Görsellerin ekrana basılmasında `subplot` komutu kullanıldı.

Power Law (gamma) Transformation

```
main.m x +
85 %% Power Law (gamma) Transformation
86 image = imread('imgg.jpg'); % imread komutu ile dizindeki imgg.jpg adlı görsel alındı ve K değişkenine atandı.
87 image = im2double(image); % matematiksel işlemler için double veri türü uygundur. im2double komutu ile uint8'den double'a çevirildi.
88 lim = stretchlim(image,[0.01,0.99]);
89 % image değişkenine atanan görsel için stretchlim komutu ile limitler bulundu. kontrast germe işlemi için tüm range aralığı kullanılacağı belirtildi.
90 j = imadjust(image,lim,[]); % imadjust komutu ile kontrast germe işlemi uygulandı.
91 k = imadjust(image,[0.06,0.55],[]); % germek istenilen aralığın değerlerini el ile girilip kontrast germe işleminin uygulanması.
92
93 subplot(2,3,1);
94 % tek tek pencere açmak yerine aynı pencere üzerinde birden fazla görsel ve histogramı görüntüleyebilmek için subplot komutu kullanıldı.
95 imshow(image,[]); % görüntü ekrana yansıtıldı.
96 title('1. Orjinal görsel:'); % Görsele ait başlık yazıldı.
97
98 subplot(2,3,2);
99 imshow(j); % görüntü ekrana yansıtıldı.
100 title('2. Konstratı gerilmiş görsel(stretchlim):');
101
102 subplot(2,3,3);
103 imshow(k); % görüntü ekrana yansıtıldı.
104 title('3. Konstratı gerilmiş görsel:');
105
106 subplot(2,3,4);
107 imhist(image); % görüntünün yoğunluk dağılımlarını görmek için, imhist fonksiyonu çağırılarak bir histogram oluşturuldu.
108 title('1. Orjinal görselin histogramı:');
109
110 subplot(2,3,5);
111 imhist(j); % görüntü ekrana yansıtıldı.
112 title('2. Konstratı gerilmiş görselin histogramı(stretchlim):');
113
114 subplot(2,3,6);
115 imhist(k); % görüntü ekrana yansıtıldı.
116 title('3. Konstratı gerilmiş görselin histogramı:');
```



Görsel im2double komutu ile uint8'den double'a çevirildi. image değişkenine atanan görsel için stretchlim komutu ile limitler bulundu.

Stretchlim, görüntünün kontrastının sınırlarını bulmayı sağlayan komuttur.

Kontrast germe işlemi için tüm range aralığı(0.01, 0.99) kullanılacağı belirtildi ve imadjust komutu ile kontrast germe işlemi uygulandı.

İmadjust, görüntü yoğunluğu değerini ve renk haritasını ayarlayan komuttur.

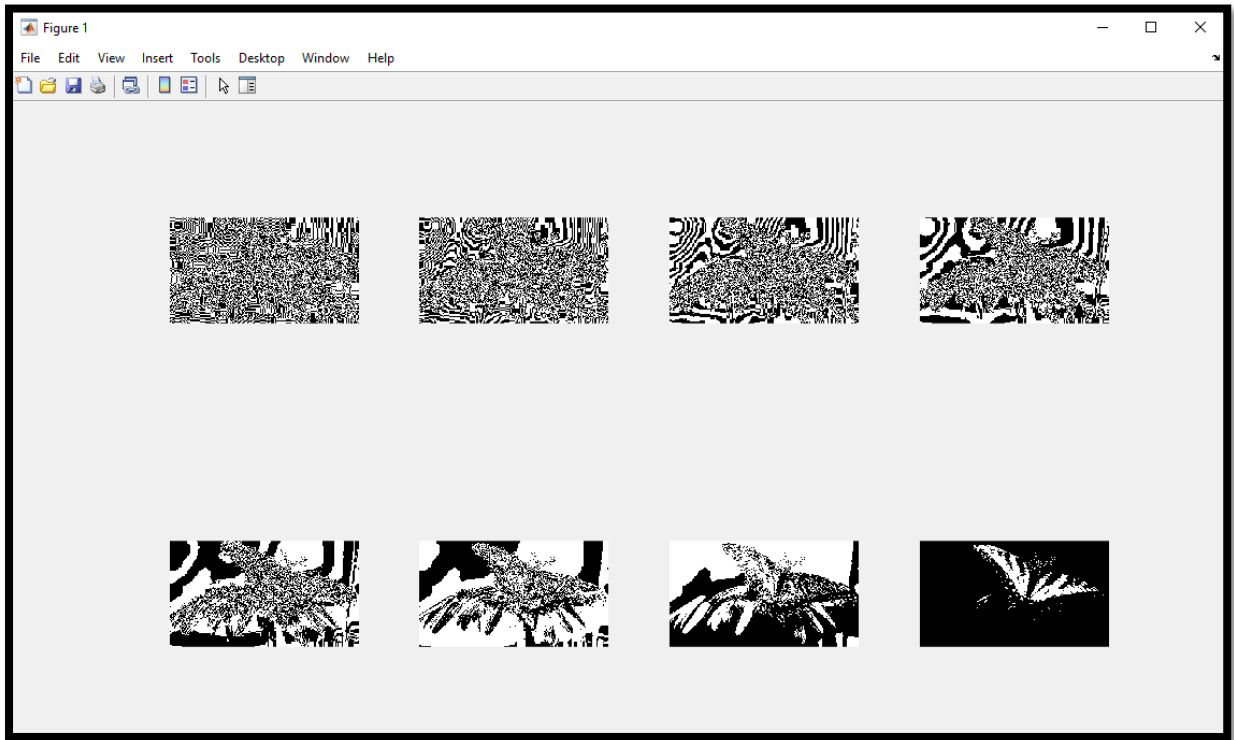
Ayrıyeten, germek istenilen aralığın değerlerini el ile girilip kontrast germe işlemi de gerçekleştirildi.

Görsellerin ekrana basılmasında subplot komutu kullanıldı.

BitPlane

BitPlane, bit düzlemi dilimlemenin gösterilmesi için kullanılır. Girdi olarak verilen görüntü 8 bitlik düzlemlere bölünür ve her düzlem görüntülenir. Ayrıca orijinal görüntü bu bit düzlemlerinden yeniden oluşturulur.

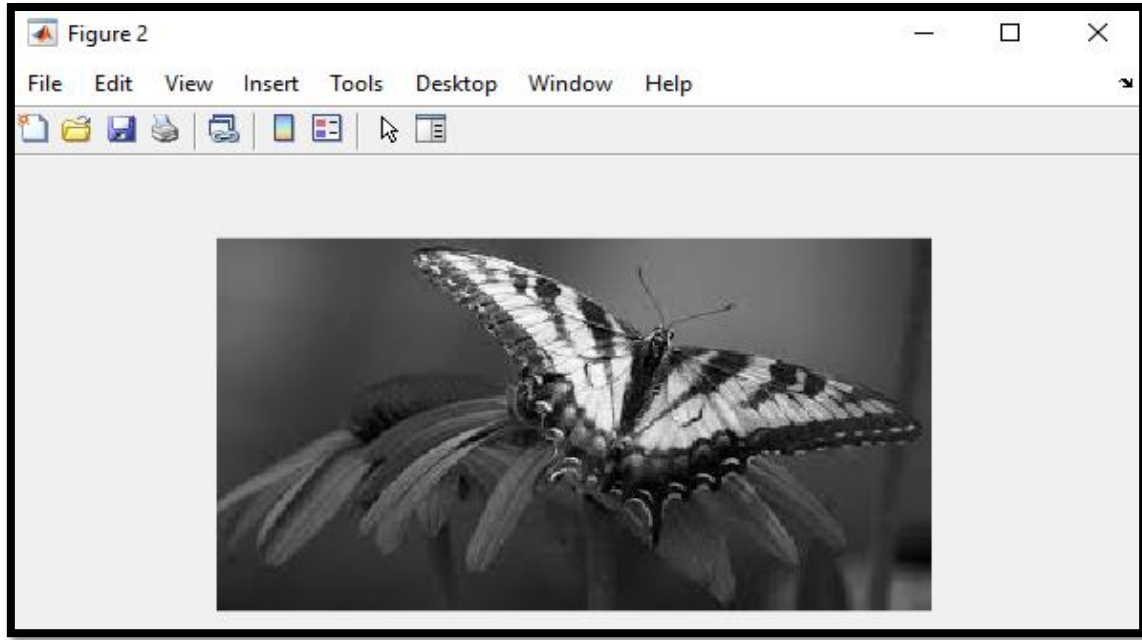
```
main.m x +
118 %% BitPlane
119 image = imread('image.jpg'); % dizindeki image.jpg gri görseli imread komutu ile alındı.
120 image = rgb2gray(image);
121 for i = 1:8
122     bitPlane(i) = bitget(image,i);
123     figure(1), subplot(2,4,i), imshow(logical(bitPlane(1,i)));
124     % bitget komutu ile görselin piksel değerlerinin ikili koddaki karşılığının 8 biti ayrı ayrı alındı ve subplot ile akrana basıldı.
125 end
```



Yukarıdaki şekilde görselin bitPlane'lerine ayrılmış hali görülmektedir.

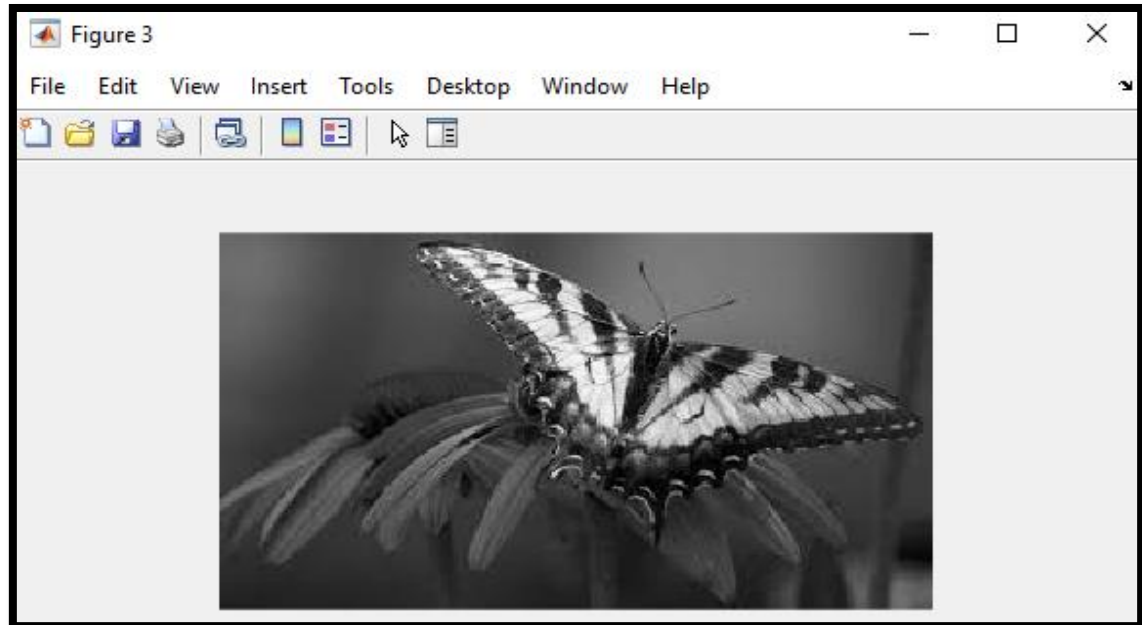
Dizindeki image.jpg gri görseli imread komutu ile alındı. bitget komutu ile görselin piksel değerlerinin ikili koddaki karşılığının 8 biti ayrı ayrı alındı ve subplot ile akrana basıldı.

```
126 %%
127 image2 = zeros(size(image));
128 for j = 1:8
129     image2 = bitset(image2,j,bitPlane(1,j));
130 end
131 figure(2), imshow(uint8(image2));
132 % bit plane'lerine ayrılan görsel bitset komutu ile birleştirildi ve ekrana basıldı.
```

Yukarıdaki şekilde bitPlane'lerine ayrılan görselin bitset komutu ile birleştirilmesi ve ekrana basılmış hali görülmektedir.

```
134 % bitset komutu kullanılmadan bitPlane'lerin birleştirilmesi.  
135 [m,n] = size(image); % Görselin boyutu alındı.  
136 for i = 1: m  
137     for j = 1: n % birleştirme için tüm pikseller gezildi.  
138         img(i,j) = 0;  
139         sayac = 8;  
140         for k = 1:8 % bitPlane bit değerleri gezildi.  
141             % bitPlane bit değerleri piksel değerlerinin bulunması için decimal değerlere çevrildi ve img değişkenine atandı.  
142             img(i,j) = img(i,j) + (double(bitPlane(1,k)(i,j))) * 2^(8-sayac);  
143             sayac = sayac - 1;  
144         end  
145     end  
146 end  
147 figure(3), imshow(uint8(img)); % Birleştirilen görsel ekrana basıldı.
```



Yukarıdaki şekilde görselin bitset komutu kullanılmadan bitPlane'lerin birleştirilmiş hali görülmektedir.

İşlem için görselin boyutu alındı. For döngüsü ile birleştirme için tüm pikseller gezildi. İkinci bir döngü ile bitPlane bit değerleri gezildi.

bitPlane bit değerleri piksel değerlerinin bulunması için decimal değerlere çevrildi ve img değişkenine atandı ve ekrana basıldı.