



**T.C**  
**KARABÜK ÜNİVERSİTESİ**  
**LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**Buse Yaren TEKİN**

**1928126503**

**Dr. Öğr. Üyesi Caner ÖZCAN**

**BSM754 – Uzaktan Algılama ve Uygulamalar Dersi Vize Raporu**

# İÇİNDEKİLER

<b>1. GİRİŞ .....</b>	<b>3</b>
1.1. MRI El Bilek Görüntülerinin İncelenmesi .....	3
1.2. Python ve Matlab Platformlarında Çalıştırma .....	3
1.3. Öznitelik Çıkarımı İncelenmesi .....	5
<b>2. ÖZNİTELİK ÇIKARIMI (FEATURE EXTRACTION) TEKNİKLERİ .....</b>	<b>6</b>
2.1. RGB, HSV, LAB Renk Kanalları ve GLCM Matris İncelenmesi .....	6
2.2. Standart Sapma, Ortalama, Özellik Vektörü (Feature Vector) Oluşturma .....	10
<b>3. KLASİK MAKİNE ÖĞRENİMİ ALGORİTMALARI .....</b>	<b>11</b>
3.1. SVM (Destek Vektör Makineleri) .....	11
3.2. Random Forest Sınıflandırma Algoritması .....	11
<b>4. KAYNAKÇA .....</b>	<b>13</b>

# 1. GİRİŞ

Günümüzde bilgisayarlı görü, görüntü işleme ve yapay zekâ projelerinde sıkça kullanılan bir yöntem olan öznelik çıkarımı, ham veri üzerinde boyut azaltma yöntemidir [1]. Makine öğrenimi ise son zamanlarda çarpıcı gelişmeler yaşamaktadır. Bu durum makine öğreniminin endüstri, akademi ve popüler kültürden büyük ilgi görmesini sağladı. Son zamanlarda sağlık alanında makine öğreniminin ve derin öğrenme modellerin uygulamaya alınmasıyla birlikte akıllı sistemler, birçok hastalığı önceden tespit edebilmekte veya uzman bir kişinin göremediği ayrıntıları gözden kaçırmamaktadır [2]. Tıbbi tedavilerde sıkça rastlanılan MRI görüntüler üzerinde hastalığı tespit edebilecek farklı bölgeler bulunmaktadır. Bu bölgelerde yoğunlaşmak üzere özellik seçimi ve öznelik çıkarımı gerçekleştirilerek elde edilen sonuçlar çeşitli algoritmalara yansıtılarak insanın algıladığı hastalıkları makinenin tespit edilmesi sağlanır.

## 1.1. MRI El Bilek Görüntülerinin İncelenmesi

Bu projede 557 adet bireyden alınan el ve bilek MR görüntüleri çeşitli ön işleme aşamalarından geçirilerek verinin iyileştirilmesi sağlanmıştır.



Şekil 1’de görüldüğü üzere arka plan çıkarımı, kontrast iyileştirme, Mean, Median ve Sobel filtresi, Açma ve Genişleme filtreleri kullanılarak görüntü ön işleme aşamalarından geçirilmiştir.

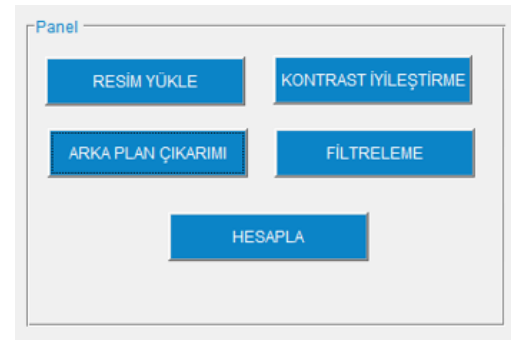
Şekil 1. Ön işleme Aşamalarından Geçirilmiş MR Görüntüsü

## 1.2. Python ve Matlab Platformlarında Çalıştırma

Projede görüntü işleme aşamaları başlangıçta MATLAB platformunda gerçekleştirilmiş olup aynı zamanda numpy kullanımı sayesinde sayısal veriler ile çalışılacağı için Python programlama dili ile de geliştirilmeye devam edilmiştir.

### MATLAB:

```
background = imopen(image,strel('disk',100));
background = uint8(background);
image = imsubtract(image,background);
image=adapthisteq(image);
image = uint8(image);
H=ones(3)/9;
image=imfilter(image,H);
mask = fspecial('disk',1);
image = imfilter(image,mask,'replicate');
gaussmask= fspecial('gaussian',[3,3],1);
image = imfilter(image,gaussmask,'replicate');
image = edge(image,'canny');
axes(handles.axes1)
set(handles.resimadi_edit,'String',filename)
```



Şekil 2. MATLAB Gui Paneli

## PYTHON:

```
def create_training_data():
    for category in CATEGORIES:
        path = os.path.join(DATA_DIRECTION, category)
        class_num=CATEGORIES.index(category)
        for img in os.listdir(path):
            try:
                img_array = cv2.imread(os.path.join(path, img), cv2.IMREAD_GRAYSCALE)
                #new_array = cv2.resize(img_array,(WIDTH,HEIGHT))
                imagenumpyarray=np.array(img_array)
                imagelist=np.ndarray.tolist(imagenumpyarray)
                print("Np array: ",imagenumpyarray)
                print("List:",imagelist)
                plt.imshow(img_array,cmap="gray")
                plt.title("El bilek")
                plt.show()
                cv2.waitKey(0)
                cv2.destroyAllWindows()
                training_data.append([img_array, class_num])
            except Exception as e:
                pass
```

Şekil 3. Python' da El Bilek Görüntülerinin ve Sayısal Değerlerin Numpy Dizisi Olarak Yazdırılması

Şekil 3' de gösterildiği gibi boş oluşturulan **training\_data\_list** adlı listeye alınan her görüntü **append** komutu ile eklenmektedir. Toplamda 557 adet eklenen görüntü liste olarak Şekil 4' de görüldüğü gibi 0-556 arasında sayısal değerler ile ifade edilmektedir.

training\_data - List (557 elements)

Index	Type	Size	Value
546	list	2	[Numpy array, 0]
547	list	2	[Numpy array, 0]
548	list	2	[Numpy array, 0]
549	list	2	[Numpy array, 0]
550	list	2	[Numpy array, 0]
551	list	2	[Numpy array, 0]
552	list	2	[Numpy array, 0]
553	list	2	[Numpy array, 0]
554	list	2	[Numpy array, 0]
555	list	2	[Numpy array, 0]
556	list	2	[Numpy array, 0]

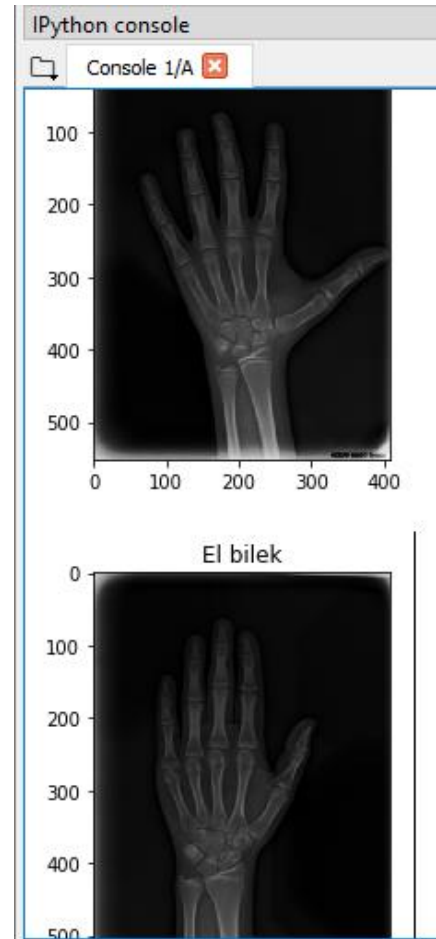
Şekil 4. Görüntülerin Eklendiği Listenin Görünümü

	0	1	2	3	4
0	185	180	176	177	175
1	182	177	174	175	170
2	181	177	174	171	164
3	181	175	169	165	159
4	178	171	164	158	153

Şekil 5. Sayısal Verilerinin İfade Edilmesi

```
List: [[185 180 176 ... 189 191 192]
[182 177 174 ... 182 186 188]
[181 177 174 ... 163 171 179]
...
[204 207 210 ... 240 242 237]
[209 209 209 ... 238 234 237]
[214 210 209 ... 232 231 243]]
```

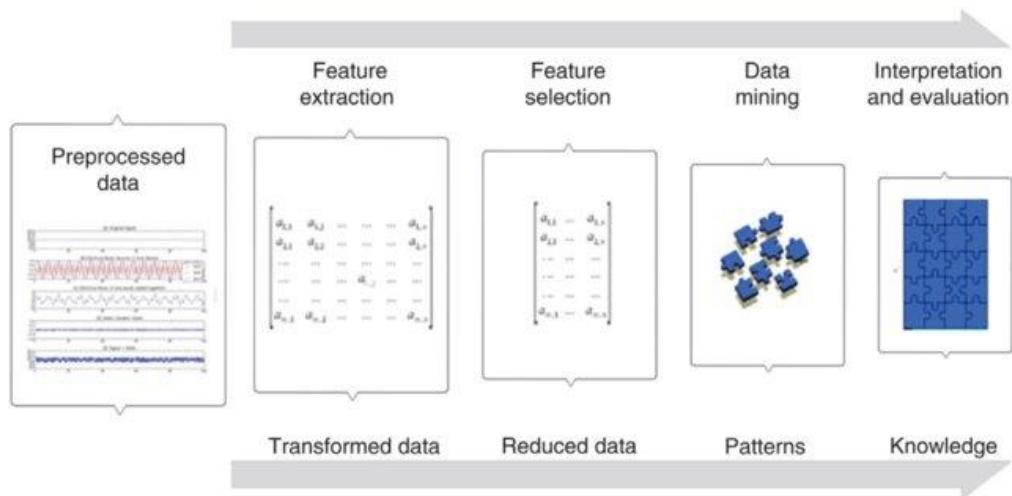
Şekil 6. `np.asarray` komutu ile listelenmiş hali



Şekil 7. Python Console Gösterimi

### 1.3. Öznitelik Çıkarımı İncelenmesi

Projedeki performansı olumlu yönden etkilemek amacıyla boyut indirgeyerek özellik çıkarımı yapılması hedeflenmektedir. Makine öğrenimi, örüntü tanıma ve görüntü işleme alanlarında kullanılan öznitelik çıkarımı (özellik çıkarımı), girdi olarak verilen ölçülmüş verileri kullanarak türetilmiş değerler (öznitelikler) oluşturur [3]. Makine öğrenmesindeki bir özellik (feature), gözlemlenen bir verinin bireysel olarak ölçülebilir özelliğidir. Özellikler, bir tahmin veya sınıflandırma yapılabilmesi için makine öğrenme modeline beslenen girdilerdir [4].



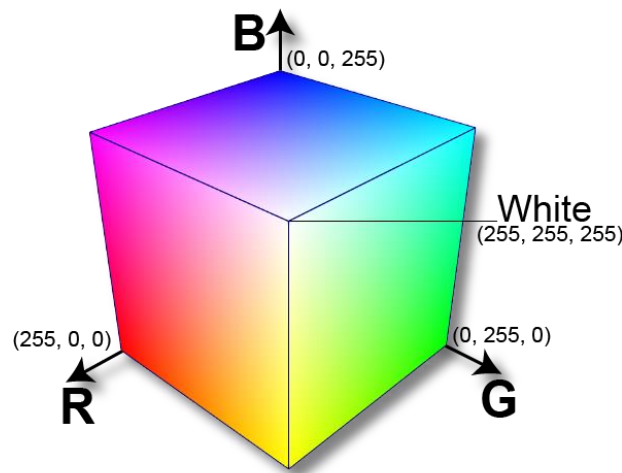
Şekil 8. Veri Analizinin Adımları [5]

## 2. ÖZNİTELİK ÇIKARIMI (FEATURE EXTRACTION) TEKNİKLERİ

Öznitelik çıkarımı, mevcut olan verilerden yeni özellikler oluşturarak ve ardından orijinal özellikleri atarak veri kümesindeki özellik sayısını azaltmayı amaçlamaktadır [6]. Bu bilgiler doğrultusunda, MATLAB ortamında görüntülerin renk kanalları öncelikle kontrol edilmiştir. Ardından belirtilen renk kanalına göre RGB ve Gray-Level bilgisi tutularak renk dönüşümleri sağlanmış ve bu sonuçlardan sayısal değerler elde edilmiştir. Bu sayısal değerler daha sonra makine öğreniminde kullanılacak olup öncelikle manuel olarak aynı sınıflara ait olup olmadığı kontrol edilmiştir.

### 2.1. RGB, HSV, LAB Renk Kanalları ve GLCM Matrisi İncelenmesi

- **RGB Renk Kanalı :** RGB en yaygın kullanılan renk alanıdır. Bu renk modelinde her renk kırmızı, yeşil ve mavi ana spektral bileşenleri olarak görülür. Bu modelin altyapısında Kartezyen Koordinat Sistemi bulunmaktadır. İlgilenilen renk alt uzayı, görüntü işlemede sıkça kullanılan bu küp olarak incelenir [7].



Şekil 9. RGB Renk Küpü [8]

Projede öncelikle RGB görüntü olup olmadığı kontrol edildi ve daha sonra Red (Kırmızı), Green (Yeşil) ve Blue (Mavi) kanalları ayrı ayrı seçildi.

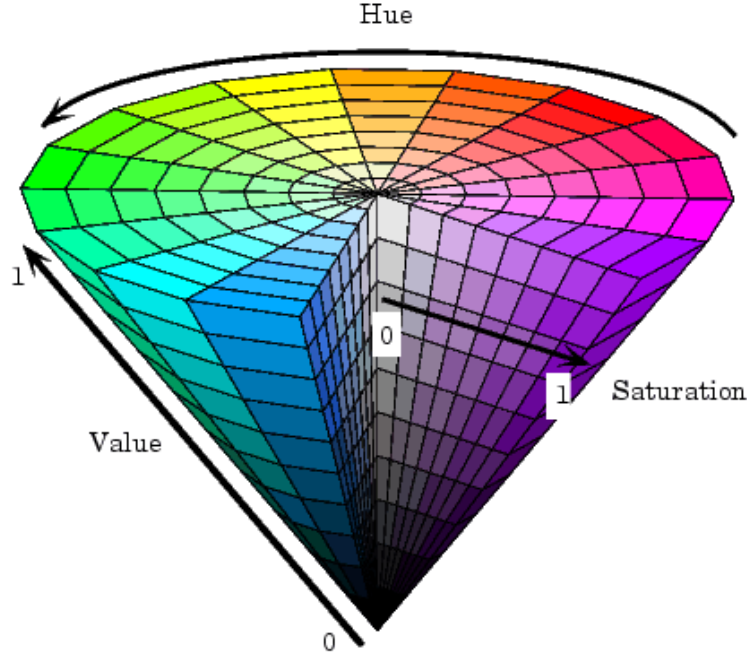
```
% Görüntü RGB ise;  
if size(image,3)==3  
    redChannel = image(:, :, 1);  
    greenChannel = image(:, :, 2);  
    blueChannel = image(:, :, 3);
```

Şekil 10. Görüntünün R,G ve B Kanallarına Ayrıştırılması

redChannel					
481x582 uint8					
	1	2	3	4	5
1	108	103	65	0	0
2	108	103	65	0	0
3	108	103	65	0	0
4	108	103	65	0	0
5	108	103	65	0	0

Şekil 11. Örnek Red Kanalı Değerlerinin Gösterimi

- **HSV Renk Kanalı :** HSV uzayının ismi, renk tonu, doygunluk ve parlaklık kelimelerinin İngilizce karşılığı olan hue, saturation ve intensity kelimelerinin baş harflerinden gelmektedir. HSV renk uzayı Hue, Saturation ve Value terimleri ile rengi tanımlar. RGB de renklerin karışımı kullanılmasına karşın HSV de renk, doygunluk ve parlaklık değerleri kullanılır. Doymunluk rengin canlılığını belirlerken parlaklık rengin aydınlığını ifade eder. HSI uzayı, bir renkli görüntüdeki yeğınlik bileşenini renk taşıyan bilgiler olan renk tonu ve doymunluktan ayırıştırır [9].



Şekil 12. HSV Renk Paletinin Sembolleştirilmesi [10]

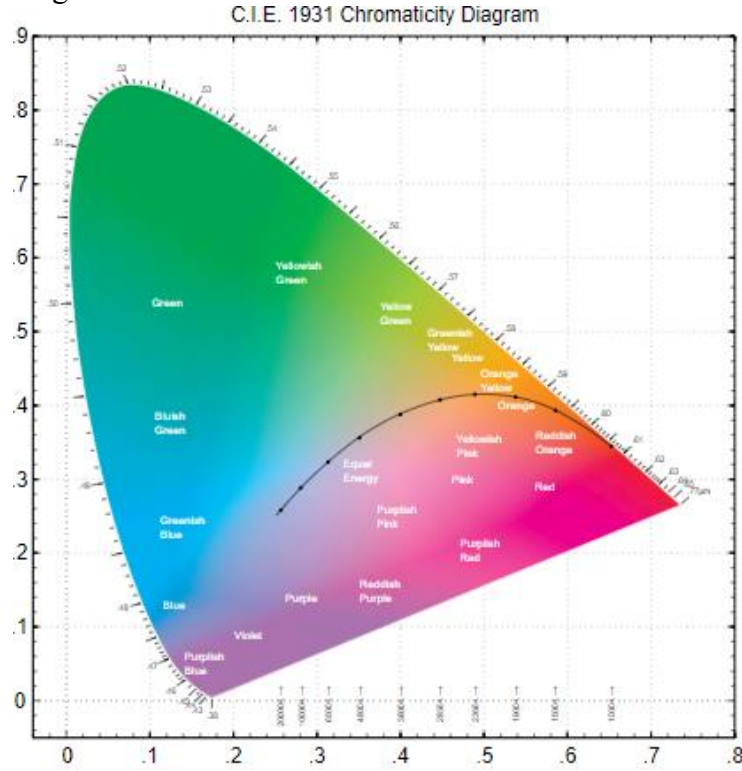
```
hsvImage=rgb2hsv(image);
hChannel = hsvImage(:,:,1); %H Component
sChannel = hsvImage(:,:,2); %S Component
vChannel = hsvImage(:,:,3); %I Component
```

Şekil 13. Görüntünün H,S ve V Özelliklerine Ayrıştırılması

	sChannel	hChannel	vChannel		
	481x582 double				
	1	2	3	4	5
1	0.4235	0.4039	0.2549	0	0
2	0.4235	0.4039	0.2549	0	0
3	0.4235	0.4039	0.2549	0	0
4	0.4235	0.4039	0.2549	0	0
5	0.4235	0.4039	0.2549	0	0
6	0.4235	0.4039	0.2549	0	0
7	0.4235	0.4039	0.2549	0	0
8	0.4235	0.4039	0.2549	0	0
9	0.4314	0.3961	0.2510	0	0
10	0.4314	0.4000	0.2549	0.0039	0
11	0.4314	0.4000	0.2549	0.0039	0

Şekil 14. Örnek V Kanalı Değerlerinin Gösterimi

- **CIE Renk Kanalı:** CIE 1931 renk uzayları , elektromanyetik görünür spektrumdaki dalga boylarının dağılımı ile insan renk görüşünde fizyolojik olarak algılanan renkler arasındaki ilk tanımlanmış kantitatif bağlantılardır . Bu renk uzaylarını tanımlayan matematiksel ilişkiler, renk mürekkepleri, ışıklı ekranlar ve dijital kameralar gibi kayıt aygıtlarıyla uğraşırken önemli olan renk yönetimi için gerekli araçlardır [11] . RGB görüntünün CIELAB kanallarına ayrıştırılması için `rgb2lab` komutu ile dönüşümünün gerçekleştirilmesi gerekir.



Şekil 15. CIE Kromatiklik Diyagramı [11]

```
cieImage = rgb2lab(image);
cChannel = cieImage(:,:,1);
iChannel = cieImage(:,:,2);
eChannel = cieImage(:,:,3);
```

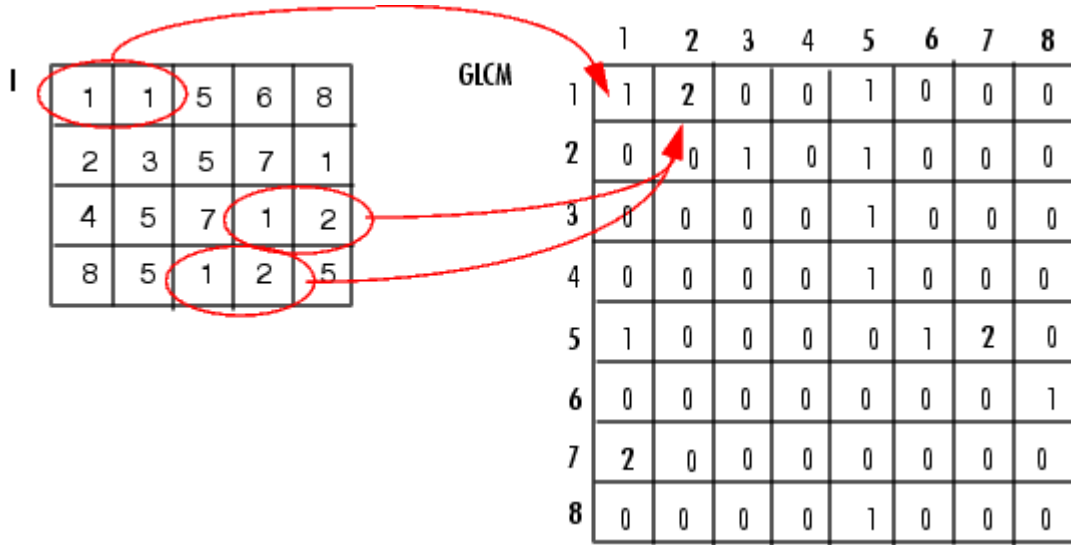
Şekil 16. Görüntünün C, I ve E Kanallarına Ayrıştırılması

	iChannel	cChannel	
	481x582 double		
	1	2	3
1	45.6287	43.6001	27.5347
2	45.6287	43.6001	27.5347
3	45.6287	43.6001	27.5347
4	45.6287	43.6001	27.5347
5	45.6287	43.6001	27.5347
6	45.6287	43.6001	27.5347
7	45.6287	43.6001	27.5347
8	45.6287	43.6001	27.5347
9	46.4355	42.7838	27.0934
10	46.4355	43.1923	27.5347

Şekil 17. Örnek C Kanalı Değerlerinin Gösterimi



- **GLCM (Gray-Level Co-Occurrence Matrix) :** Gri düzeyde eş oluşum matrisi ile bir dizi doku özelliği çıkarılabilmektedir. Doku filtresi işlevleri, görüntü histogramına dayalı olarak dokunun istatistiksel bir görünümünü sağlar. Bu işlevler bir görüntünün dokusu hakkında yararlı bilgiler sağlayabilir, ancak şekil, yani bir görüntüdeki piksellerin uzaysal ilişkileri hakkında bilgi sağlayamaz [12].



Şekil 18. Örnek GLCM Değerlerinin Hesaplanması [12]

```
gImage = rgb2gray(image);
glcms = graycomatrix(gImage);
statsg = graycoprops(glcms);
```

Şekil 19. Görüntünün Gri Seviye Eş Oluşum Matrisinin Oluşturulması

Şekil 19’ da görüldüğü üzere GLCM, RGB renk modelinde değil Gri Seviye boyutunda çalışmaktadır. Bu sebeple **rgb2gray** komutu model değişimi yapılmaktadır. **graycomatrix** komutu ile Şekil 18’ de görüldüğü gibi eş oluşum matrisi oluşturulmaktadır.

glcms

8x8 double

	1	2	3	4	5	6	7	8
1	202191	1051	362	2	0	0	0	0
2	1038	42844	1622	73	0	0	0	0
3	377	1627	21975	291	0	0	0	0
4	0	55	687	4808	26	0	0	0
5	0	0	32	90	239	2	0	0
6	0	0	0	1	38	30	0	0
7	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0

Şekil 20. GLCM (Eş Oluşum Matrisi) Oluşumu

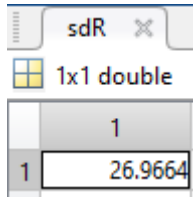
## 2.2. Standart Sapma, Ortalama, Özellik Vektörü (Feature Vector) Oluşturma

Makine öğreniminde görüntüler üzerinden özellik çıkarılmak istendiğinde öncelikle özellik seçimi için görüntülerin sayısal değerlerine ulaşım matematiksel özellikleri elde edilmektedir. Her renk modelindeki kanalların ayrı ayrı standart sapmaları, aritmetik ortalamaları ve entropileri hesaplanarak çözüme ulaşılmak istenmektedir.

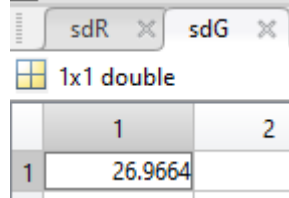
- **Standart Sapma:** Bir serideki sayıların serinin aritmetik ortalamasından farklarının karelerinin toplamının dizinin eleman sayısının bir eksiğine bölümünün kareköküdür [13].

```
sdR = std2(redChannel);  
sdG = std2(greenChannel);  
sdB = std2(blueChannel);  
  
avgRGB = mean(reshape(image, [], 3), 1);
```

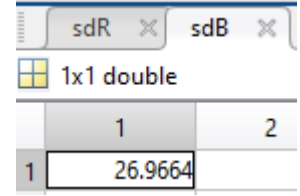
Şekil 21. Standart Sapma ve Ortalamanın Gösterilmesi



Şekil 22. R Kanalının Standart Sapması



Şekil 23. G Kanalının Standart Sapması

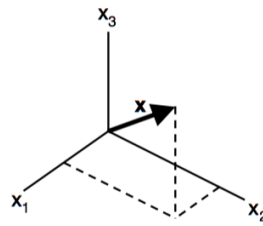


Şekil 24. B Kanalının Standart Sapması

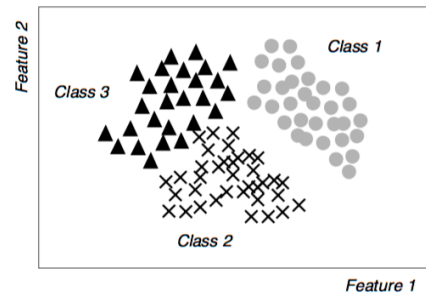
- **Özellik Vektörünün Oluşturulması:** Makine öğreniminde özellik vektörleri, bir nesnenin özellikler olarak adlandırılan sayısal veya sembolik özelliklerini matematiksel, kolayca analiz edilebilir bir şekilde temsil etmek için kullanılır. Makine öğrenimi ve görüntü işlemenin birçok farklı alanı için önemlidir. Makine öğrenimi algoritmaları, algoritmaların işleme ve istatistiksel analiz yapabilmesi için genellikle nesnelerin sayısal bir gösterimini gerektirir. Özellik vektörleri, doğrusal regresyon gibi istatistiksel prosedürlerde kullanılan açıklayıcı değişkenlerin vektörlerinin eşdeğeri [14].

$$\mathbf{X} = \begin{bmatrix} X_1 \\ X_2 \\ X_d \end{bmatrix}$$

Feature vector



Feature space (3D)



Scatter plot (2D)

Şekil 25. Örnek Feature Vector Oluşturma

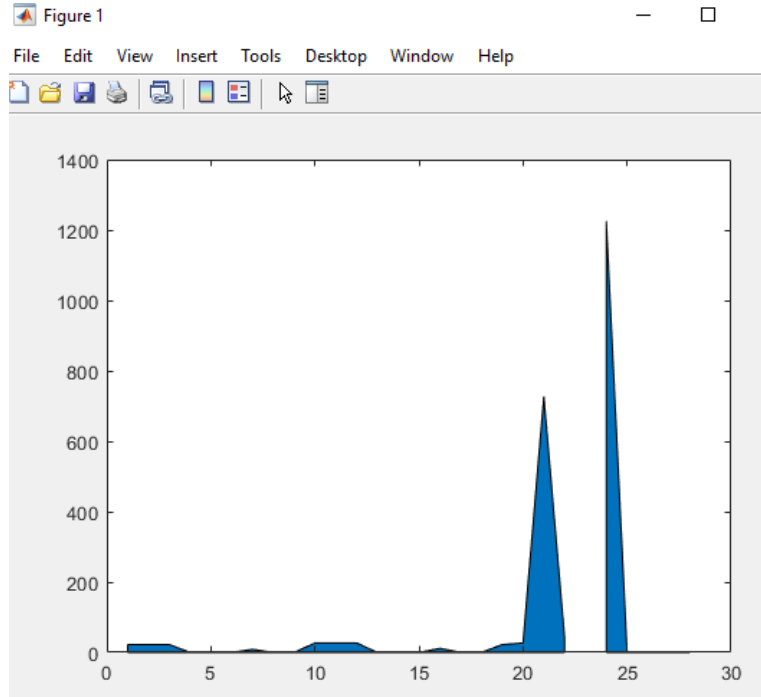
```
feature_vektor=[v1 v2 statsg.Contrast statsg.Correlation statsg.Energy statsg.Homogeneity];
```

Şekil 26. Özellik Vektörünün Oluşturulması

Oluşan özellik vektörü 1x28 boyutunda değerler içeren bir vektördür.

feature_vektor												
1x28 double												
	1	2	3	4	5	6	7	8	9	10	11	12
1	22.3250	22.3250	22.3250	0	0	0.0875	8.7005	1.6179e-15	9.8632e-17	26.9664	26.9664	26.9664

Şekil 27. Özellik Vektörü Değerlerinin Gösterimi

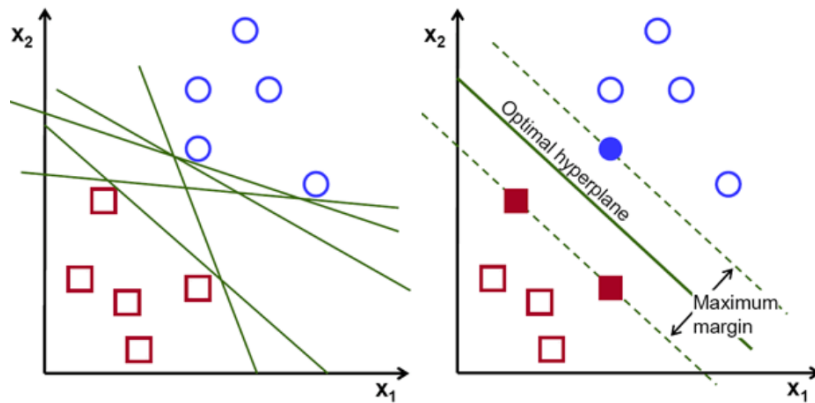


Şekil 28. Özellik Vektörünün Grafikleştirilmesi

### 3. KLASİK MAKİNE ÖĞRENİMİ ALGORİTMALARI

Elde edilen veriler ile makinenin öğrenmesini sağlamak için öncelikle belirli algoritmalar araştırıldı. Veri bilimi ve özellik çıkarımında sıkça kullanılan algoritmalar Destek Vektör Makinesi (SVM) ve Random Forest Sınıflandırma Algoritması olarak tespit edildi.

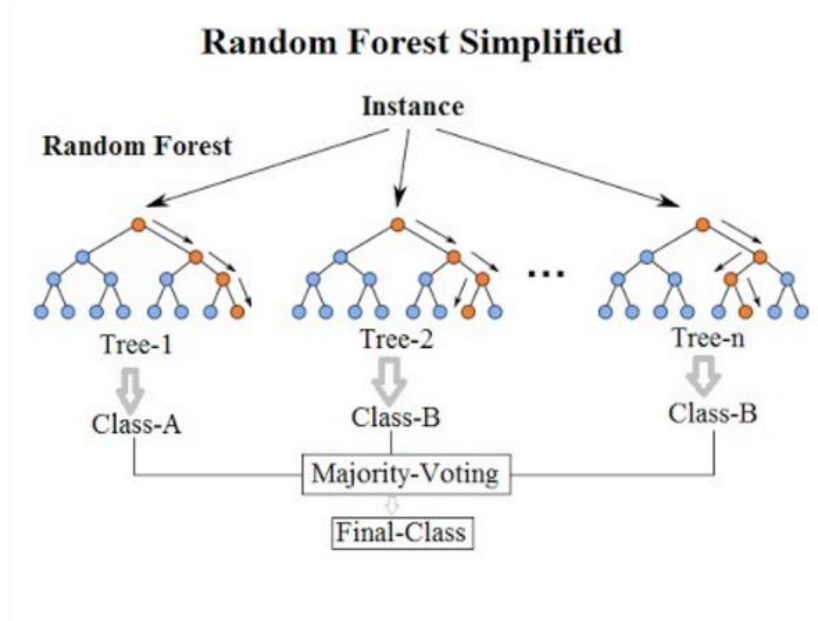
**3.1. SVM (Destek Vektör Makineleri):** SVM, temel olarak iki sınıfa ait verileri birbirinden en uygun şekilde ayırmak için kullanılır. Bunun için karar sınırları yada diğer bir ifadeyle hiper düzlemler belirlenir. Karar fonksiyonunda bir takım eğitim noktaları kullanılır. Dolayısıyla bellek verimli bir şekilde kullanılmış olur [15].



Şekil 29. Doğrusal SVM Grafiği [16]

Sınıflandırma (Classification) konusunda kullanılan oldukça etkili ve basit yöntemlerden birisidir. Sınıflandırma için bir düzlemde bulunan iki grup arasında bir sınır çizilerek iki grubu ayırmak mümkündür. Bu sınırın çizileceği yer ise iki grubun da üyelerine en uzak olan yer olmalıdır [17].

**3.2. Random Forest (Rastgele Orman) Algoritması:** Rastgele ormanlar veya rastgele karar ormanları, sınıflandırma, regresyon ve diğer görevler için, eğitim aşamasında çok sayıda karar ağacı oluşturarak problemin tipine göre sınıf (sınıflandırma) veya sayı (regresyon) tahmini yapan bir toplu öğrenme yöntemidir. Rastgele karar ormanları, karar ağaçlarının eğitim setlerine aşırı uyma problemlerini gidermektedir [18].



Şekil 30. Rastgele Orman Düzümlerinin Sembolik Olarak İfade Edilmesi [16]

Rastgele orman algoritması, eğitimdeki aşırı öğrenme (Overfitting) problemini çözmek için hem veri setinden hem de öznitelik setinden rassal olarak 10'larca 100'lerce farklı alt-setler seçiyor ve bunları eğitiyor. Bu yöntemle 100'lerce karar ağacı oluşturuluyor ve her bir karar ağacı bireysel olarak tahminde bulunmaktadır [19].

## KAYNAKÇA

- [1] Şadi Evren Şeker, “Özellik Çıkarımı (Feature Extraction)”, 1 Aralık 2008, <http://bilgisayarkavramlari.sadievrenseker.com/2008/12/01/ozellik-cikarimi-feature-extraction/adresinden alınmıştır>.
- [2] M. Mert Tunalı, “MRI Görüntüleri Üzerinden Beyin Tümörü Tespiti Bölüm 1 (U-Net)”, Medium, <https://medium.com/deep-learning-turkiye/mri-g%C3%B6r%C3%BCnt%C3%BCleri-%C3%BCzerinden-beyin-t%C3%BCm%C3%B6r%C3%BC-tespiti-ec644a2ff0c9> adresinden alınmıştır.
- [3] From Wikipedi, Özgür Ansiklopedi, “Öznitelik çıkarımı”, 26 Mart 2020, [https://tr.wikipedia.org/wiki/%C3%96znitelik\\_%C3%A7%C4%B1kar%C4%B1m%C4%B1](https://tr.wikipedia.org/wiki/%C3%96znitelik_%C3%A7%C4%B1kar%C4%B1m%C4%B1).
- [4] MC.AI, The Computer Vision Pipeline, Part 4: Feature Extraction, October 2019, <https://mc.ai/the-computer-vision-pipeline-part-4-feature-extraction/>.
- [5] Javier Gonzalez-Sanchez, Mustafa Baydoğan, Maria-Elena Chavez-Echeagaray, Winslow Burleson, Affect Measurement: A Roadmap Through Approaches, Technologies, and Data Analysis, December 2017.
- [6] Pier Paolo Ippolito, “Feature Extraction Techniques”, Towards Data Science, <https://towardsdatascience.com/feature-extraction-techniques-d619b56e31be>.
- [7] C. Gonzalez, Rafael, E. Woods, Richard, Sayısal Görüntü İşleme, Palme Yayıncılık, (Ankara, 2014).
- [8] [https://favpng.com/png\\_view/light-rgb-color-space-rgb-color-model-light-png/BsYUHtec](https://favpng.com/png_view/light-rgb-color-space-rgb-color-model-light-png/BsYUHtec) adresinden alınmıştır.
- [9] Dr. Öğr. Üyesi Caner Özcan, Karabük Üniversitesi, BLM429 Görüntü İşlemeye Giriş, “Renkli Görüntü İşleme”.
- [10] <https://tr.pinterest.com/pin/391179917623338540/> adresinden alınmıştır.
- [11] From Wikipedi, Özgür Ansiklopedi, “CIE 1931 renk alanı”, 10 Nisan 2020, [https://en.wikipedia.org/wiki/CIE\\_1931\\_color\\_space](https://en.wikipedia.org/wiki/CIE_1931_color_space).
- [12] Matlab, Image Processing Toolbox User's Guide, “Using a Gray-Level Co-Occurrence Matrix (GLCM)”, <http://matlab.izmiran.ru/help/toolbox/images/enhanc15.html>.
- [13] <https://standart-sapma-hesaplama.hesabet.com/> adresinden alınmıştır.
- [14] Brilliant, “Feature Vector”, <https://brilliant.org/wiki/feature-vector/>, April 2020.
- [15] E. Kaan Ulgen, Medium, “Makine Öğrenimi Bölüm-4 (Destek Vektör Makineleri), 2017, <https://medium.com/@k.ulgen90/makine-%C3%B6%C4%9Frenimi-b%C3%B6l%C3%BCm-4-destek-vekt%C3%B6r-makineleri-2f8010824054> adresinden alınmıştır.
- [16] Arş. Gör. Dr. Rafet Durgut, BLM436 Örüntü Tanıma, Hafta 4 Destek Vektör Makinesi, [https://docs.google.com/presentation/d/1F2I3iljmXuLHSayf7xhCAwWqM3qAcp5cyw7bXh1J2og/edit#slide=id.g50271ce022\\_0\\_0](https://docs.google.com/presentation/d/1F2I3iljmXuLHSayf7xhCAwWqM3qAcp5cyw7bXh1J2og/edit#slide=id.g50271ce022_0_0) adresinden alınmıştır.

[17] Şadi Evren Şeker, “SVM (Support Vector Machine, Destekçi Vektör Makinesi)”, 1 Aralık 2008, <http://bilgisayarkavramlari.sadievrenseker.com/2008/12/01/svm-support-vector-machine-destekci-vektor-makinesi/> adresinden alınmıştır.

[18] From Vikipedi, Özgür Ansiklopedi, “Rastgele orman”, 16 Ocak 2020, [https://tr.wikipedia.org/wiki/Rastgele\\_orman](https://tr.wikipedia.org/wiki/Rastgele_orman) adresinden alınmıştır.

[19] Hakkı Kaan Şimşek, Data Science TR, Medium, “Makine Öğrenmesi Dersleri 5a: Random Forest (Sınıflandırma)”, 24 Mart 2018, <https://medium.com/data-science-tr/makine-%C3%B6%C4%9Frenmesi-dersleri-5-bagging-ve-random-forest-2f803cf21e07> adresinden alınmıştır.

[20] Juan Ram’ón Ant’ón Morillas, Irene Camino Garc’ía and Udo Zölzer, Faculty of Electrical Engineering, Helmut Schmidt University “Ship Detection Based on SVM Using Color and Texture Features”.