

Şubat 2024

Melanoma

MELANOMA ANALYSIS

Cilt benlerinden yapay zeka ile kanser riski
hesaplama web arayüz projesi

Hazırlayanlar

Muhammed Hükümdar	- 02210201501
Selami Çalışkan	- 02210201505
İbrahim Erkan	- 02195076044

İÇİNDEKİLER

01

Proje Hakkında

02

Geliştiriciler

03

Problem - Çözüm -
Teknolojiler- Yöntem

04

Veri Seti

05

Proje Adımları

07

Görseller



hkmdrmuhamd/
MelanomaAnalyze

3

Contributors

0

Issues

1

Star

1

Fork



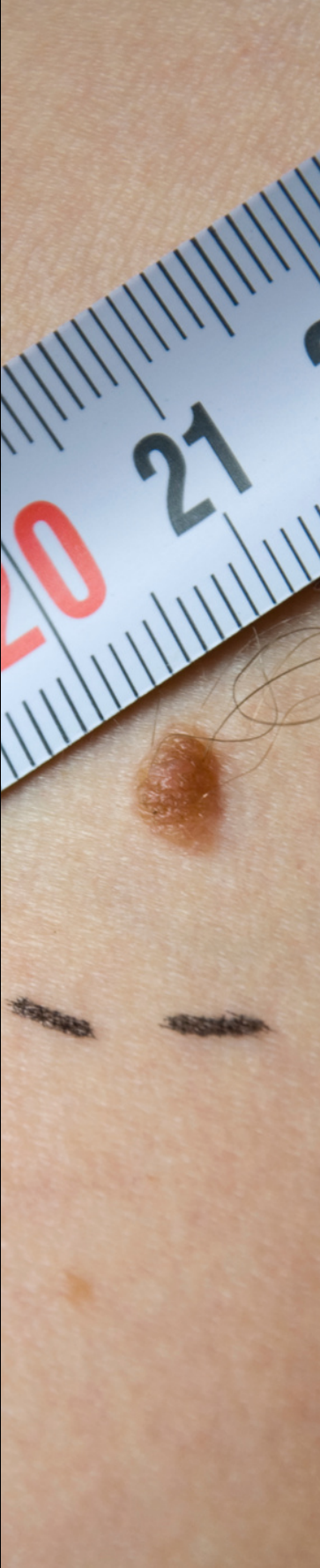
hkmdrmuhamd/MelanomaAnalyze

Contribute to hkmdrmuhamd/MelanomaAnalyze development by creating an account on GitHub.

 GitHub

PROJE HAKKINDA

Melanoma Analysis



Bu proje vücuttaki benleri görüntüleyerek kanser riskini ve zararlı benleri değerlendirmeye yardımcı olan bir web sitesi geliştirmeyi amaçlamaktadır. Web sitesi, kullanıcıların kendi benlerini evde kolayca ve güvenle takip edebilmelerine yardımcı olabilir. Bu erken teşhis ve tedavi ile kanser riskini azaltmaya yardımcı olabilir.

Vücuttaki benlerin iyi huylu veya kötü huylu olduğunu belirlemek genellikle bir uzmanın değerlendirmesini gerektiren karmaşık bir süreçtir. İyi huylu ve kötü huylu benler genellikle benzer görünebilir, bu nedenle sadece gözlemlenerek bir tanı koymak zor olabilir.

Projemizde ileri seviyede veri analizi ve yapay zeka teknikleri kullanarak web arayüzünde kullanıcı ile ürünümüzü buluşturuyoruz. Projemizde amaç ben tipine göre kanser riskini hesaplayıp kullanıcıyı tıbbi müdahale için yönlendirmektir.

GELİŞTİRİCİLER



Muhammed

Hükümdar

Full Stack Web Developer



Selami

Çalışkan

Artificial Intelligence Engineer



İbrahim

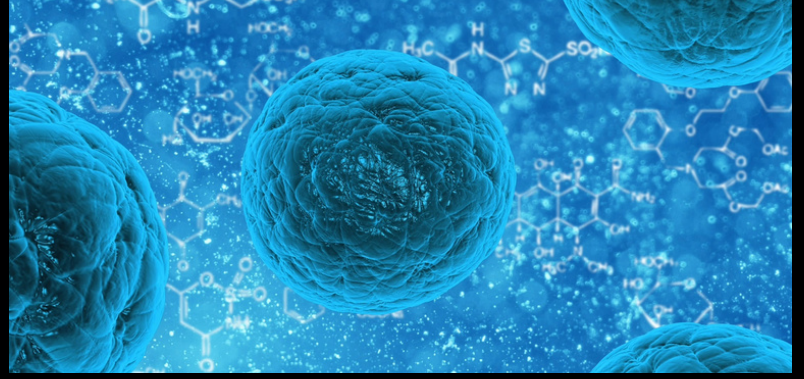
Erkan

Computer Vision Engineer



PROBLEM

Vücutta oluşan benlerin iyi huylu veya kötü huylu olduğunun tanısını koymak zordur.



YÖNTEM

Görüntü işleme tekniklerinin ardından yapay zeka modeli web arayüzünden gelen görüntüyü sınıflandırır ve kanser riskini gösterir

ÇÖZÜM

Yapay zeka destekli web tabanlı ürünümüz ile yalnızca bir fotoğraf ile erken tanı imkanı sağlamaktayız.



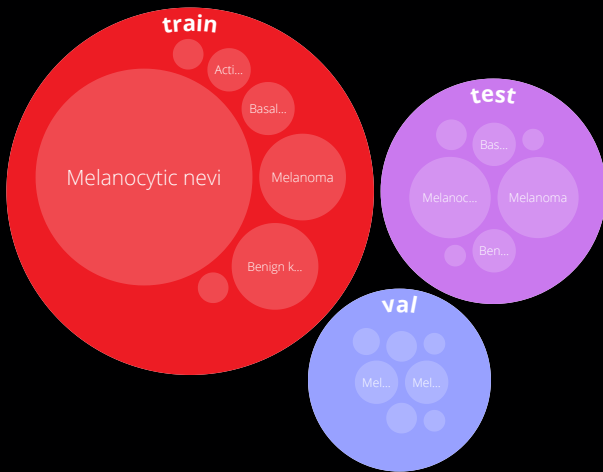
TEKNOLOJİLER

Tensorflow, openCV, Keras, Python, Java Spring, Maven, Data JPA, Spring WEB, React.JS, Bootstrap

VERİ SETİ

İnternet ortamından elde ettiğimiz görüntülerden oluşan veri setini train, test ve validation olarak ayırdık.

Train kısmı ile modelimizi eğitip, test kısmı ile eğitim sırasında başarı değerlendirmesi yaparken, validation kısmıyla nihai modelimizi değerlendirdik.



Melanocytic Nevi (Nevüs): Tanım: Deride melanosit hücrelerinin birikimi sonucu oluşan deri lekeleridir, genellikle kahverengi renkte ve genellikle zararsızdır.

Melanoma: Tanım: Cilt kanserinin agresif bir türüdür; melanositlerin kontrolsüz büyümesi ve yayılmasıyla karakterizedir.

Benign Keratosis-Like Lesions: Tanım: Cildin üst katmanında (epidermis) meydana gelen, genellikle keratinle ilişkili benign (zararsız) lezyonlardır.

Basal Cell Carcinoma: Tanım: Cilt kanserinin başka bir türüdür, genellikle güneşe maruz kalan bölgelerde ortaya çıkar ve genellikle yavaş büyür.

Actinic Keratoses: Tanım: Güneşin neden olduğu hasar sonucu ortaya çıkan precancerous (önceden kanseröz) lezyonlardır.

Vascular Lesions: Tanım: Ciltteki damarlarla ilişkili lezyonlardır, örneğin, hemangioma veya telanjiyektazi gibi.

Dermatofibroma: Tanım: Deride sıklıkla görülen, genellikle zararsız olan bir tür fibrohistiyositik tümördür.

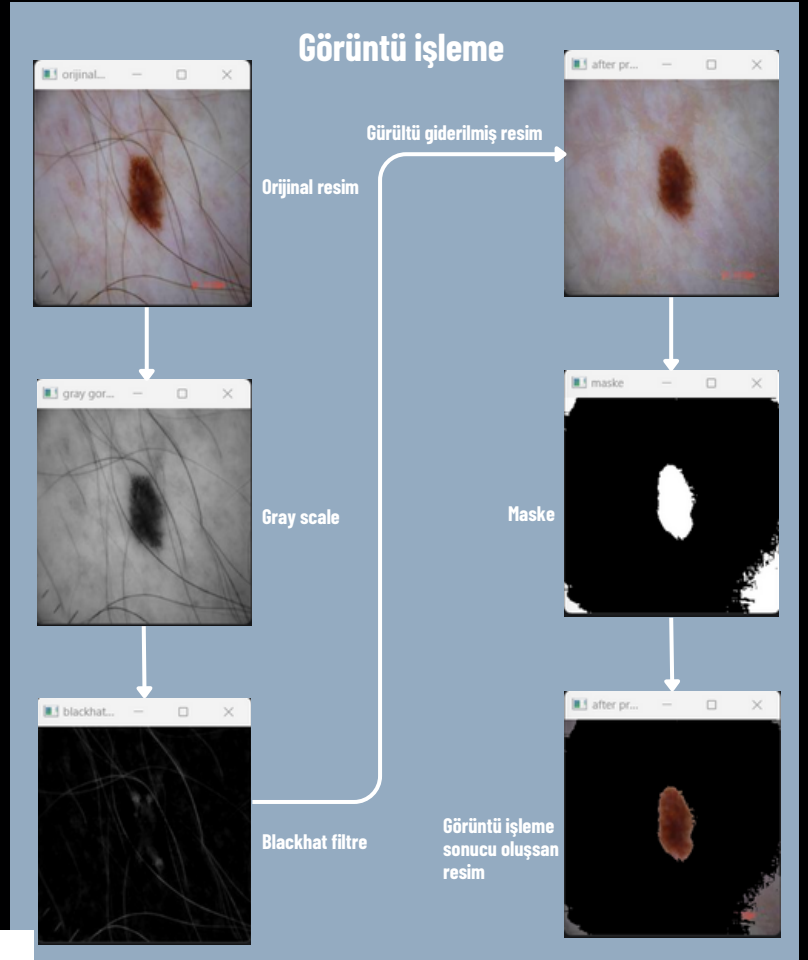
Veri seti linki : <https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000>

PROJE ADIMLARI

1- GÖRÜNTÜ İŞLEME

Train ve test veri setlerindeki görüntüleri, görüntü işleme adımlarından geçirip modelimiz için uygun hale getirdik. Bunlar:

- Görseli yeniden boyutlandırma
- Piksel ölçeklendirme Veri artırımı
- Çeşitli filtreleme yöntemleri



Model: "sequential"			
Layer (type)	Output Shape	Param #	
conv2d (Conv2D)	(None, 75, 100, 32)	896	
conv2d_1 (Conv2D)	(None, 75, 100, 32)	9248	
max_pooling2d (MaxPooling2D)	(None, 37, 50, 32)	0	
dropout (Dropout)	(None, 37, 50, 32)	0	
conv2d_2 (Conv2D)	(None, 37, 50, 64)	18496	
conv2d_3 (Conv2D)	(None, 37, 50, 64)	36928	
max_pooling2d_1 (MaxPooling2D)	(None, 18, 25, 64)	0	
dropout_1 (Dropout)	(None, 18, 25, 64)	0	
flatten (Flatten)	(None, 28800)	0	
dense (Dense)	(None, 512)	14746112	
dropout_2 (Dropout)	(None, 512)	0	
dense_1 (Dense)	(None, 256)	131328	
dropout_3 (Dropout)	(None, 256)	0	
dense_2 (Dense)	(None, 7)	1799	
Total params: 14944807 (57.01 MB)			
Trainable params: 14944807 (57.01 MB)			
Non-trainable params: 0 (0.00 Byte)			

2- MODEL

İlk olarak CNN yöntemi ile base model oluşturduk. Bu modeli oluşturmamızın amacı görüntüler üzerinde ve modelimiz üzerinde geliştirmeler yaptıktan sonra başarılarımızı görmektir.

PROJE ADIMLARI

3- WEB ARAYÜZÜ

Kullanıcının arayüze yüklediği görsel ile görüntü işleme scriptine ve model nesnesine bir request gönderir. Dönen cevap sonucunda back-end'te yazılan API ile kullanıcıya response döndürülür.

BACKEND

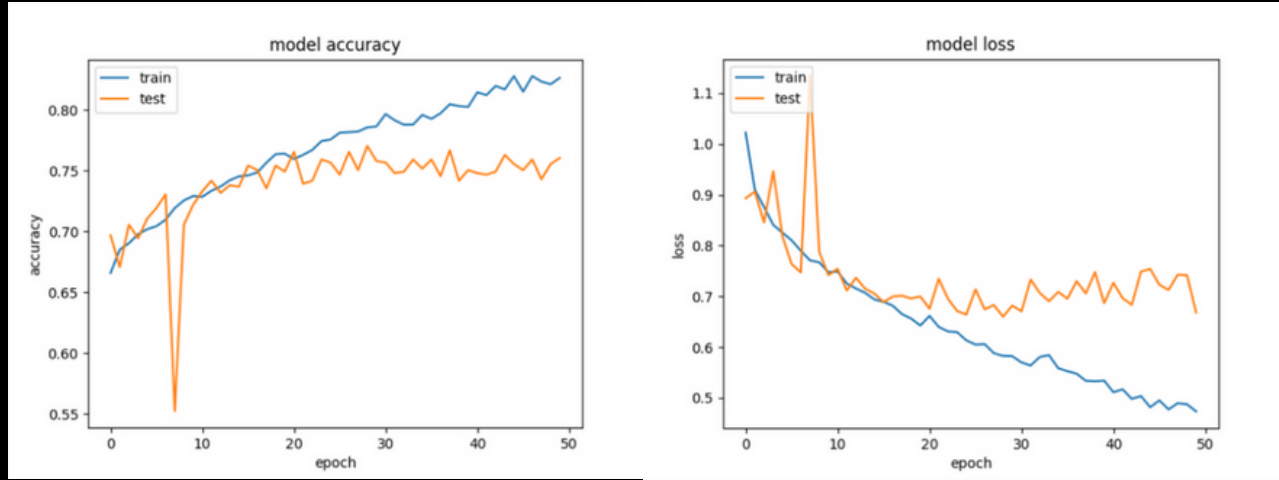
- Frontend ile iletişime geçilmesi için bir API yazılmıştır.
- Bu API'nin yaptığı işlem: frontend'teki görselin binary olarak bölünmesini ve bu binary verinin model tarafına iletilmesini sağlar. Sonra model tarafında veri model için uygun hale getirilir.
- Upload image API'nin AI tarafına gönderdiği response düzenlenmiştir.
- AI'dan dönen sınıflandırma sonucunu yakalayan API yazılmıştır.
- Back-end'e gelen sınıflandırma sonucunu front-end'e ileten API yazılmıştır.

FRONTEND

- Görüntünün yüklenebilmesi için file input component'i eklenmiştir.
- Görüntünün backende gönderilmesi için axios modülü kullanılmıştır.
- Yüklenen fotoğraf image component içerisinde gösterilmiştir.
- Model tarafından alınan sonuç kullanıcıya bir sütun grafiği ile sunulmuştur.
- Ben türleri ve detayları ile ilgili kullanıcıya bilgi sunmak için sayfanın en üstüne navbar eklenmiştir.
- UploadFile butonunun stil düzenelemeleri yapılmıştır.
- Barchart statik halden dinamik hale getirilmiştir.
- Axios ile API'dan gelen json yapısındaki veri ile grafik oluşturulmuştur.
- Sayfanın stil güncellemeleri detaylı bir şekilde yapılmıştır.



MODEL BAŞARI DEĞERLENDİRMESİ



WEB ARAYÜZÜ

