

T.C.
ESKİŐEHİR OSMANGAZİ ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĐİ BÖLÜMÜ

2022-2023 Eğitim-Öğretim Yılı

152117011- TASARIM SÜREÇLERİ

PROJE RAPORU

Proje Başlıđı

“Bitki hastalıklarını tespit eden yapay zeka”

Projeyi Hazırlayanlar:

“Ferdî İslam Yılmaz, 152120191055, Bilgisayar Mühendisliđi”

“Hüseyin Emir Leylek, 152120191067, Bilgisayar Mühendisliđi”

“Bünyamin Taşkın, 152120191046, Bilgisayar Mühendisliđi”

Ders Grubu: A

Danışman

Öğr.Gör.Dr. Yıldırav ANAGÜN

İçindekiler

1. GİRİŞ	4
2. PLANLAMA.....	4
2.1. BİLGİ GEREKSİNİM BELİRLEME, PROBLEMİN TANIMLANMASI	4
2.1.1. Amaç	4
2.1.2. Konu ve Kapsam	4
2.1.3. Literatür Özeti.....	5
2.2. Fizibilite(Yapılabilirlik) Raporları	5
2.2.1. Kurumsal ve Kültürel Fizibilite	5
2.2.2. Teknolojik Kaynak Fizibilitesi	5
2.2.3. Yasal Fizibilite	5
2.2.4. Finansal Fizibilite	5
2.2.5. Zaman Fizibilitesi	5
2.3. BEKLENEN FAYDA	5
2.4. YÖNTEM	6
2.5. ARAŞTIRMA OLANAKLARI	6
2.6. ÇALIŞMA TAKVİMİ	7
2.6.1. GANNT şeması	7
2.6.2. Kişi iş paylaşımı	7
3. ANALİZ	8
3.1. Sistem Gereksinimlerini Ortaya Çıkarma Yöntem ve Teknikleri.....	8
3.1.1. Yazılı Basılı Belge İnceleme	8
3.1.2. Yüz Yüze Görüşme	8
3.1.3. Gözlem.....	8
3.1.4. SWOT Analiz Formu	8
3.1.5. İş Akış şeması	9
3.2. Gereksinimler	10
3.2.1. Fonksiyonel Gereksinimler	10
3.2.2. Fonksiyonel Olmayan Gereksinimler	10
4.TASARIM	11
4.1.1 Kullanıcı Veri Girişi Ara Yüzleri.....	11

4.1.2 Raporlama/Bilgilendirme Ara Yüzleri	12
4.3 Test Tasarımı	13

1. GİRİŞ

Yapacağımız projenin hedefi tarım sektöründe ürünlerin yaprak hastalıklarının daha önceden tespit edilip müdahale edilmesi. Bunu yerleştireceğimiz kamera sistemleriyle aldığımız görüntüyü yapay zeka modelimize aktararak yapacağız.

Günümüzde yaprak hastalıkları nedeniyle gelirlerde yüzde 7 veya 9'luk bir kayıp söz konusudur. Projemiz bu hastalıkların önüne daha erken geçerek gelirlerdeki kayıpları yüksek oranda azaltacaktır.

2. PLANLAMA

2.1. BİLGİ GEREKSİNİM BELİRLEME, PROLEMİN TANIMLANMASI

2.1.1. Amaç

Dünya üzerinde milyonlarca bitki hastalıklar nedeniyle çürümeye mahkum kalıyor. Bu durum gerek ekosistem için gerek zirai faaliyetler için çok büyük zararlara neden oluyor. Bu projemizde amacımız ise ekosisteme büyük zarar veren aynı zamanda büyük üretim kayıplarına neden olan zirai hastalıkları yapay zeka desteğiyle önceden tespit edip bu zirai hastalıkları aynı zamanda zirai hastalıkların neden olduğu zirai kayıpları önlemek.

Bugün, belki de ilk ekilen buğdaydan neredeyse on bin yıl kadar sonra, tarım alanında sahip olduğumuz bilgi birikimi ve teknoloji inanılmaz boyutlarda. Böylesine bir gelişim, mevcut ekilebilir arazideki, hatta artık su, küvez gibi toprak dışında da, üretim kapasitesini tarihte eşi benzeri olmayan bir seviyeye taşımış durumda. Ancak ne yazık ki bu yüksek potansiyelin tamamını hayata geçiremiyoruz. Zirai hastalıklar, yeni yeni gündemimize girmeye başlayan iklim değişikliği ve kuraklık gibi sorunların ötesinde ve onlardan çok daha uzun zamandır, tüm teknolojik ve kültürel birikimimizi bir kalemde silebiliyor.

2.1.2. Konu ve Kapsam

Projemiz konu ve kapsam itibarıyla, tarım sektöründeki verim kaybını azaltarak alınabilecek maksimum verimde üretim sağlamak aynı zamanda ekosistemimize büyük zarar veren zirai hastalıkları erkenden tespit ederek ekosistemimizi korumak.

Bu projede zirai hastalıkları sınıflandırmak için derin öğrenme modeli tasarlanıp kullanılacaktır. Derin öğrenme modelimizin eğitim ve test aşamaları için gerekli olan fotoğraflar verileri internet üzerinden hazır olarak alınacaktır. Veri setinin öğrenme ve test kısmı yapılacaktır. Derin öğrenme modeli sayesinde hastalığın, hangi zirai hastalığa ait olduğunu değerlendirerek çözüm üretecek bölüme gönderilmesi sağlanacaktır.

2.1.3. Literatür Özeti

Günümüzde zirai hastalıkların sınıflandırması gelişmiş ülkelerde çeşitli bölgeler üzerinde görevlendirilmiş bilim insanları tarafından gelişme açısından geride kalmış ülkelerde ise çoğunlukla manuel olarak yapılmaktadır. Bu durum gelişmiş ülkelerde her ne kadar bilim insanların gözetiminde ilerlese de söz konusu süreç insan faktöründen kaynaklı olarak hastalığın doğru değerlendirilememesi veya zamanında teşhis edilememesi gibi sorunları da beraberinde getirmektedir. Sınıflandırma işlemi bir bilgisayar yardımı ile yapılmak istendiğinde ise devreye akıllı sistemler girmektedir. Bu akıllı sistemler sayesinde zirai hastalıklar hem erken teşhis hem de doğru teşhis konusunda çok yardımcı olacağını düşünmekteyiz. Henüz bu konuda geliştirilen bir akıllı sistem örneği mevcut değil. Bu nedenle bu projede daha önceden hazır olarak bulduğumuz veri seti ile eğitilecek en iyi sınıflandırma yapan model kullanılacaktır.

2.2. Fizibilite(Yapılabilirlik) Raporları

2.2.1. Kurumsal ve Kültürel Fizibilite

Ekibimizdeki herkesin tarım alanında tecrübeleri vardır. Aslında bu projeyi , kendi tarım alanlarımızda da kullanabilme ihtimalimiz olup olmadığını düşünerek ortaya çıkardık. Bu yüzden kültürel fizibilitesi bize uygun bir projedir.

2.2.2. Teknolojik Kaynak Fizibilitesi

Projeyi kendi cihazlarımız üzerinden yapacağımız için teknolojik yeterliliğe sahibiz. Gereken teknolojik bilgi olarak da yeterliyiz.

2.2.3. Yasal Fizibilite

Yapacağımız projeyi engelleyen herhangi bir kanun bulunmamaktadır.

2.2.4. Finansal Fizibilite

Ekibimiz tamamen gönüllü çalışacağı için herhangi bir ödeme yapılmayacaktır. Gereken cihazlar da ekibimizin sahip olduğu imkanlarla sınırlıdır.

2.2.5. Zaman Fizibilitesi

Proje bir ay içerisinde bitirilecek olup ekibimizin zaman olarak bir sıkıntısı yoktur.

2.3. BEKLENEN FAYDA

Tarım sektöründe her yıl milyonlarca ürün zirai hastalıklar yüzünden telef olmaktadır. Bu durum tarım işçilerine ve tarım ekosistemine telafisi zor zararlar vermektedir. Çiftçilerin bekledikleri yıllık gelir beklenmedik şekilde düşüyor. Aynı zamanda çiftçilerin verdikleri emek boşa gidiyor ve moral bozukluğuna sebep oluyor. Çiftçiler yaşadıkları bu olumsuz olaylar dolayısıyla tarım sektöründen çekilmek zorunda kalabiliyor. Bireysel sorunların yanında toplumsal olarak günümüzde olduğu gibi gıda fiyatlarında yükseliş, bir gıda krizi veya ülke ihracatında düşüş gibi problemlere de yol açabiliyor. Yapmayı hedeflediğimiz projede zirai hastalıkları önceden tespit ederek ekilen ürünlerin daha verimli hasat edilmesini

sağlayacağız. Projemiz kullanımda yaygınlaştığı zaman bu sorunların beklenen hedefte önüne geçileceğini düşünüyoruz.

2.4. YÖNTEM

Kendi tecrübelerimizden ve sorduğumuz çiftçilerden aldığımız bilgilere göre ürünlerin kontrolü her gün yapılamıyor. Genelde bir veya iki haftada bir belki de ayda bir insan gözüyle kontrol ediliyor. Bu zaman aralıklarında hastalık çoktan bulaşmış ve yayılmış olabiliyor. Bizim yapacağımız sistem ile insan gözlemine gerek kalmadan anlık olarak kontrol edilebiliyor. Bu sayede hastalık ilerlemeden ve insan gözünden kaçabilecek detaylar atlanmadan erken teşhis edebileceğiz.

Tarımda son yıllarda teknolojinin yeri bir hayli artmış durumdadır. Yine de zirai hastalıkları önlemekte yeteri kadar başarılı olamamışlardır. Bizim yapacağımız sistemde, tarım arazisinin belirli noktalarına sabit veya hareketli kameralar kurularak bitki yaprakları anlık olarak gözlemlenecektir. Bu sayede olası bir hastalık henüz ilerlemeden teşhis edilip kullanıcıya gerekli bilgilendirme yapılacaktır.

2.5. ARAŞTIRMA OLANAKLARI

Bu bölümde projenin oluşturulması sırasında kullanılacak olan makine ve malzeme listesi aşağıda verilmiştir.

Projede Kullanılacak Mevcut Makine – Teçhizat Listesi	
Adı/Modeli	Projede Kullanım Amacı
Bilgisayar	Yazılım geliştirme ve yerel sunucu olarak kullanılması
Kamera	Derin öğrenme modelinin eğitilmesi, test edilmesi ve veri ön işleme için kullanılması

Projede Kullanılacak Cihazların Donanım ve Fiyat Bilgileri		
Adı/Modeli	Donanım Bilgileri	Fiyat

Bilgisayar	<ul style="list-style-type: none"> • Marka Model: Monster Notebook Abra A7 v12 • İşlemci: Intel® Coffee Lake Core™ i5-11000H • RAM: 8GB(1x8GB) DDR4L 1.2V 3100MHz SODIMM • Ekran Kartı: 4GB GDDR5 nVIDIA® GeForce® GTX1650 128-Bit DX12 • 512GB SSD 	15.000₺
Kamera sistemi	<ul style="list-style-type: none"> • Telefon kamerası 	7000₺

2.6. ÇALIŞMA TAKVİMİ

2.6.1. GANNT şeması



2.6.2. Kişi iş paylaşımı

- Ferdi İslam Yılmaz
 - Giriş,
 - Beklenen Fayda,
 - Gannt şeması,
 - Sunum hazırlanması
- Hüseyin Emir Leylek
 - Yöntem
 - Fizibilite raporu
 - Araştırma olanakları

- IV. Sunum hazırlanması
- Bünyamin Taşkın
 - I. Bilgi gereksinim belirleme
 - II. Problemin tanımlanması
 - III. Sunum hazırlanması

3. ANALİZ

3.1. Sistem Gereksinimlerini Ortaya Çıkarma Yöntem ve Teknikleri

3.1.1. Yazılı Basılı Belge İnceleme

Projemiz için internet üzerinde yazılı kaynaklarda bulunan Kew Kraliyet Botanik Bahçesi tarafından yayımlanan raporda, incelediğimizde dünya üzerinde bilinen yaklaşık olarak 390 nin türden bitki bulunmaktadır.Bu bitkilerin yaklaşık olarak %21 i hastalık,iklim değişikliği gibi sebeplerden ötürü yok olma tehlikesiyle karşı karşıyadır.Bu oranın tamamı hastalıklar olmasa da büyük ölçüde pay sahibidir.

3.1.2. Yüz Yüze Görüşme

Proje kapsamında kendimiz de tarımla ilgilendiğimiz için çevremizdeki tarım sektöründe çalışan insanlarla görüştük ve hepsinden genel olarak şu sonuçlara vardık:

- bitki hastalıklarının kendilerinin emeklerini ve zamanlarını boşa götürdüğü,bütçelerini çok zorladığı sonucuna vardık.

3.1.3. Gözlem

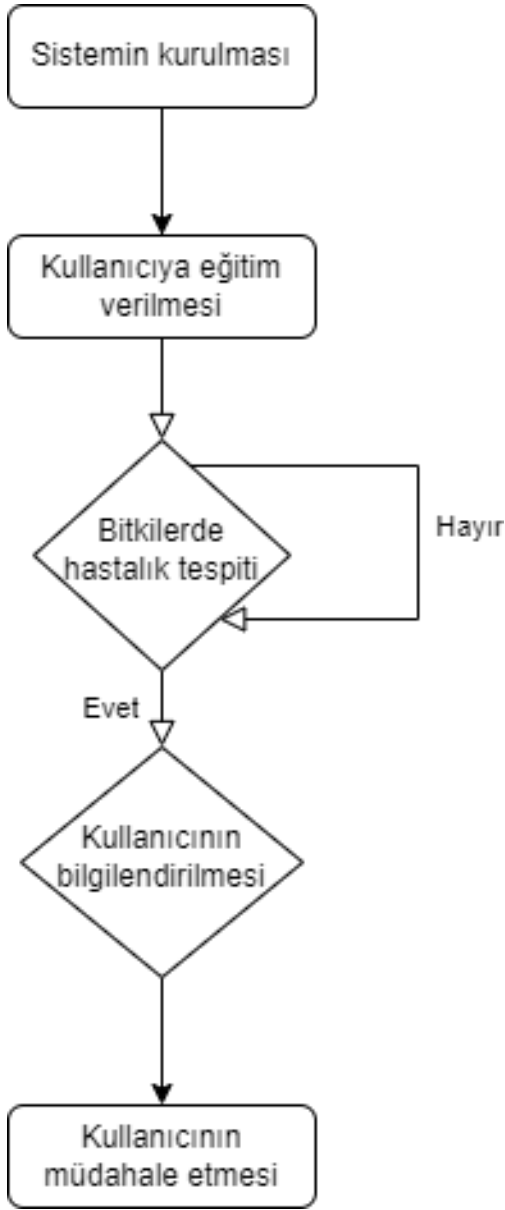
Kendi tarım arazilerimizden de gözlemlediğimiz kadarıyla hastalık başlangıcı insan gözüyle tespit edilemeyebiliyor.Hastalık ilerlediğinde ise yapılan müdahaleler için artık çok geç olabiliyor.Bu süreç içerisinde yayılan hastalığın doğurduğu zararlar ise üst düzeylere ulaşabiliyor.Bu yüzden sürekli insan gözüyle kontrole gerek olmayan hastalığı anında tespit edebilen bir teknoloji tarım sektörü için gayet faydalı olacaktır.

3.1.4. SWOT Analiz Formu

Güçlü Yanları	Zayıf Yanları
<ul style="list-style-type: none">• İnsan gözüyle denetime gerek kalmadan anlık olarak araziyi kontrol edecek.Olası bir hastalık tespiti durumunda kullanıcı durumdan haberdar olacak.	Yağmur,kırağı,kar gibi doğal olaylarda kamera sisteminin yaprak üzerindeki hastalık belirtilerini algılayamama ihtimali.Teknolojik cihazlar olacağından insan gözünden daha dayanıksız olması.

Fırsatlar	Tehditler
Tarım sektörü çalışanları için bitki hastalıklarının tespitini kolaylaştırması.	Projede kullanılan teknolojik cihazların çalınması
Hastalık tespiti için sürekli olarak kontrol etmeden zamandan tasarruf ettirmesi.	Sistemde meydana gelebilecek bir arıza nedeniyle hastalığın tespit edilememesi veya yayılması.
Erken tanıdan dolayı gelir kaybının indirgenmesi.	

3.1.5. İş Akış şeması



3.2. Gereksinimler

3.2.1. Fonksiyonel Gereksinimler

- Oluşturulan veri seti eğitilecek.
- Veri seti içerisinde hasta ve sağlıklı olmak üzere binlerce fotoğraf yer alacak.
- Kullanımı kolay ve anlaşılır bir ara yüz yapılacak.
- Eğitilen model ile ara yüz birbirine bağlanılacak.

3.2.2. Fonksiyonel Olmayan Gereksinimler

3.2.2.1. Kullanılabilirlik

- Drone ve kamera sistemi aktif çalışır durumda olmalıdır.
- Kamera sisteminden devamlı olarak kayıt alınmalıdır. ×

3.2.2.2. Güvenilirlik

- Kullandığımız derin öğrenme sayesinde hata oranı en düşük seviyede olacaktır.

3.2.2.3. Desteklenebilirlik

- Windows işletim sisteminde kullanılacaktır.

3.2.2.4. Arayüz

- Uygulama arayüzü kullanışlı ve anlaşılır olmalıdır.

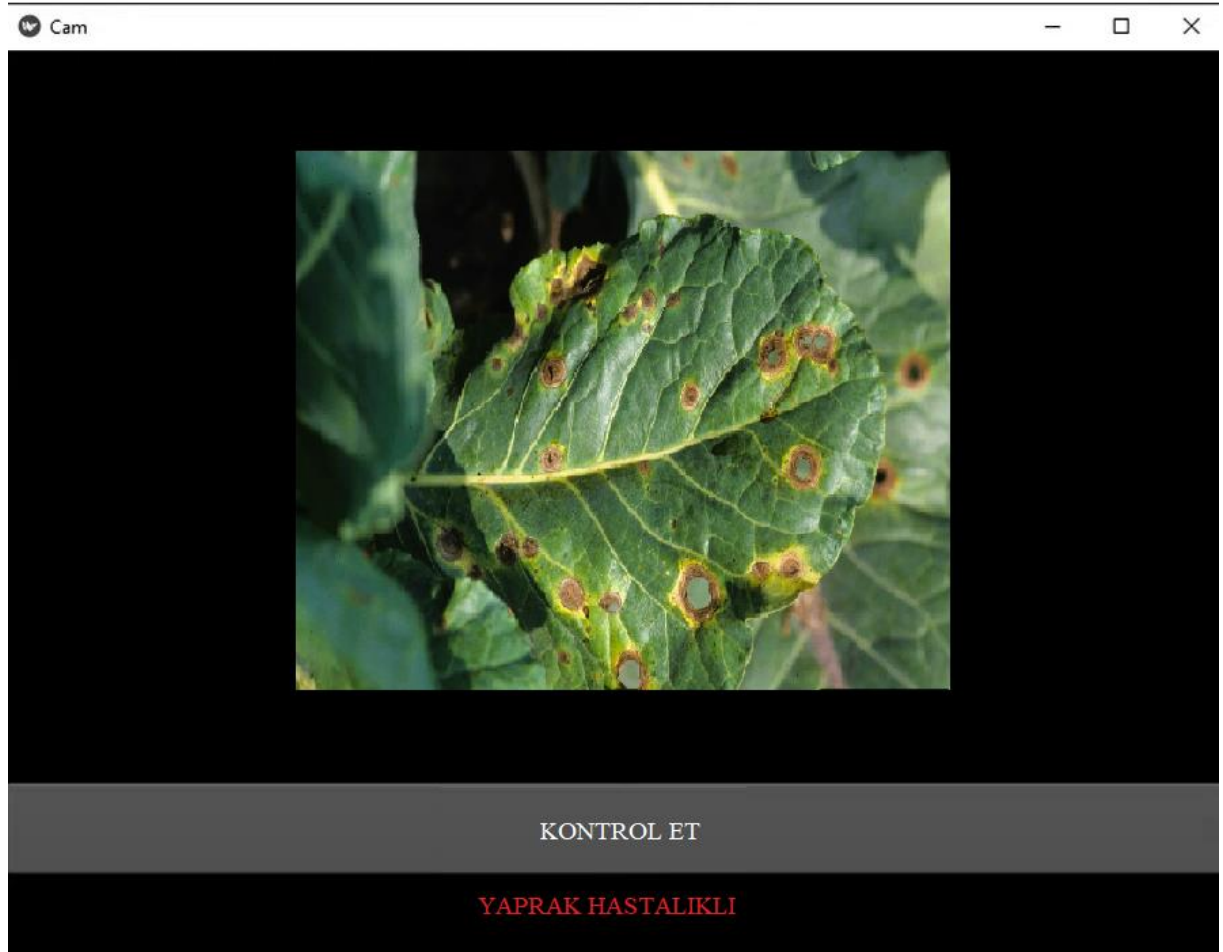
3.2.2.5. Gizlilik

- Kullanıcının herhangi bir bilgisi dış dünya ile paylaşılmayacaktır

4.TASARIM

4.1 Kullanıcı ve Sistem Ara yüzü Tasarımları

4.1.1 Kullanıcı Veri Girişi Ara Yüzleri



Şekilde görüldüğü üzere kivy üzerinden bir ara yüz yapılacaktır. Bu kısımda kullanıcıdan veri alınacaktır.

4.1.2 Raporlama/Bilgilendirme Ara Yüzleri

Kontrol et butonuyla beraber yaprağın hastalıklı olup olmadığı hesaplanacak ve kullanıcıya bir bilgilendirme yansıtılacaktır.

4.2 Yazılım Tasarımı

- Kütüphaneler yüklenecek:

Tensorflow: TensorFlow, TensorFlow hesaplamalarının oluşturulmasını ve yürütülmesini sağlamak için belirli bir modül seti (Python, C ve C++ için API'ler içerir) kullanır. Bu hesaplamalardan elde edilen veri akışı grafikleri durum bilgisidir, yani programın etkileşim durumunu takip ettiği anlamına gelir.

Daha spesifik olmak gerekirse, TensorFlow, bir görüntü hakkında giderek daha karmaşık verileri ortaya çıkarmak için Düğüm (Nodes) adı verilen veri katmanlarını sıralar. TensorFlow, daha derin düğümlere daldıkça daha karmaşık sorular sorabilir.

OpenCV: Bilgisayarla görü, makine öğrenimi, görüntü işleme, video analizi gibi uygulamalar için kullanılan devasa bir açık kaynak kodlu kütüphanedir. Gerçek zamanlı işlemlerde oldukça önemli bir rol oynamaktadır

Matplotlib: Python programlama dilinde 2D grafikler için kullanılan bir çizim kitaplığıdır. Python komut dosyalarında, kabukta, web uygulama sunucularında ve diğer grafik kullanıcı arayüzü araç setlerinde kullanılabilir.

- Dosya yollarını oluşturacağız.
- Yaprak fotoğraflarını hasta veya değil olarak etiketleyeceğiz.
- Tensorflow kütüphanesini kullanarak tüm fotoğrafları ölçeklendireceğiz.
- Eğitim ve test kısmını kuracağız.
- Siamese Neural Network kullanacağız.

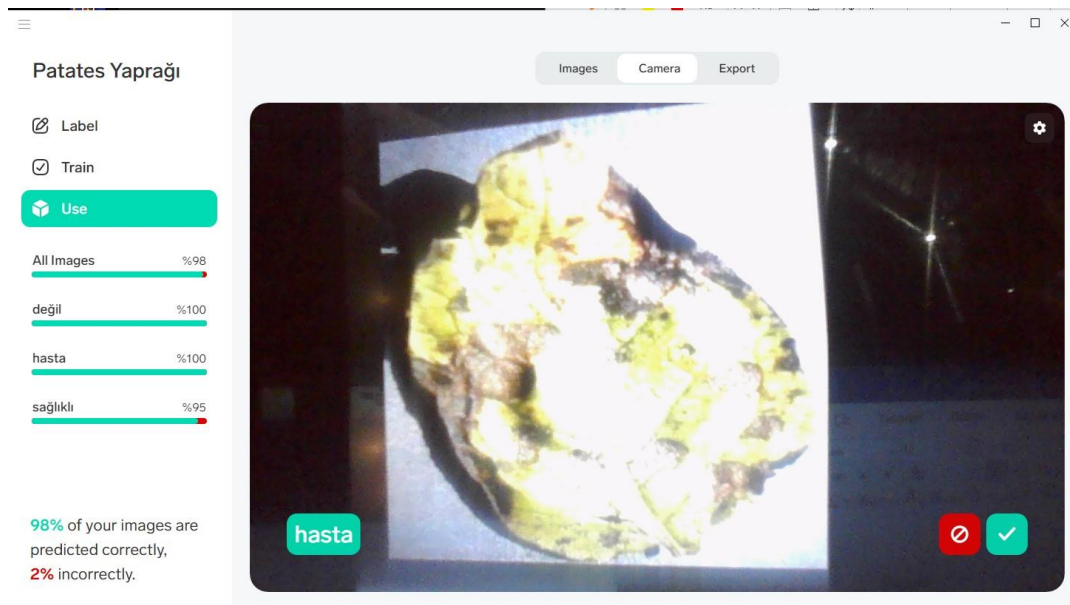
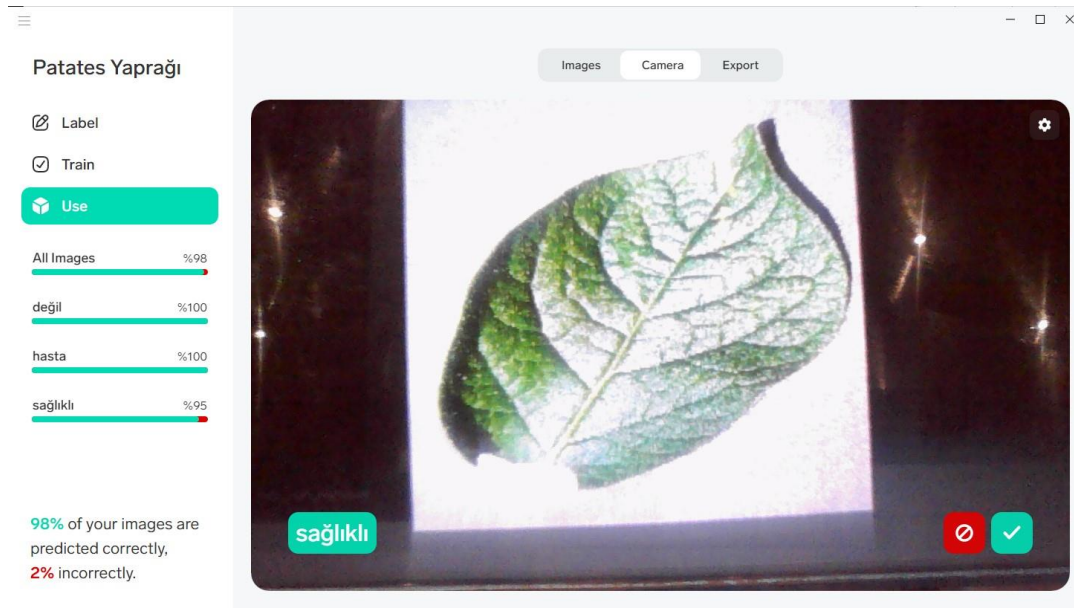
Siamese Neural Network karşılaştırılabilir çıktı vektörlerini hesaplamak için iki farklı girdi vektörü üzerinde birlikte çalışırken aynı ağırlıkları kullanan bir yapay sinir ağıdır.

- Verileri eğitim aşamasına geçeceğiz.
- Veriler eğitildikten sonra test edilecek.
- Başarı oranı yüksek bir model elde edersek modeli daha sonra programımızda kullanmak üzere kaydedeceğiz.
- Ara yüzü tasarlamak için Kivy kütüphanesini kullanacağız.
- Hastalık tespit uygulamasını geliştireceğiz.
- Sonra bu uygulama ile tensorflow modelimizi entegre edeceğiz.

- Daha sonra kameradan alacağımız görüntüyü openCV kütüphanesi metodları ile işleyeceğiz ve görüntülerin modelden geçmesini sağlayacağız.

4.3 Test Tasarımı

Projemizin yapılabilirliğini test etmek için Microsoft'un geliştirdiği Lobe uygulamasını kullandık. Bu uygulamada veri setini kendisi eğitiyor ve bize test imkanı sunuyor. Test ettiğimiz görüntüler aşağıdaki gibidir.



Görüldüğü üzere veri seti doğru bir şekilde eğitilmiş ve doğru sonuçlar vermiştir.

5.UYGULAMA

6.SONUÇ VE ÖRNEKLER

Bu aşamada modelimizi eğittik fakat arayüze bağlayamadığımızdan ötürü bir uygulama formatında sunamadık.Fakat arayüzsüz modelimize ait bazı fotoğraflar aşağıdadır.

```
In [52]: def train(data, EPOCHS):
# Loop through epochs
for epoch in range(1, EPOCHS+1):
    print('\n Epoch {} / {}'.format(epoch, EPOCHS))
    progbar = tf.keras.utils.Progbar(len(data))

    # Creating a metric object
    r = Recall()
    p = Precision()

    # Loop through each batch
    for idx, batch in enumerate(data):
        # Run train step here
        loss = train_step(batch)
        yhat = siamese_model.predict(batch[2])
        r.update_state(batch[2], yhat)
        p.update_state(batch[2], yhat)
        progbar.update(idx+1)
    print(loss.numpy(), r.result().numpy(), p.result().numpy())

    # Save checkpoints
    if epoch % 10 == 0:
        checkpoint.save(file_prefix=checkpoint_prefix)
```

```
In [53]: EPOCHS = 50
```

```
In [54]: train(train_data, EPOCHS)
```

```
Epoch 46/50
50/50 [=====] - 29s 579ms/step
4.4227625e-05 1.0 1.0

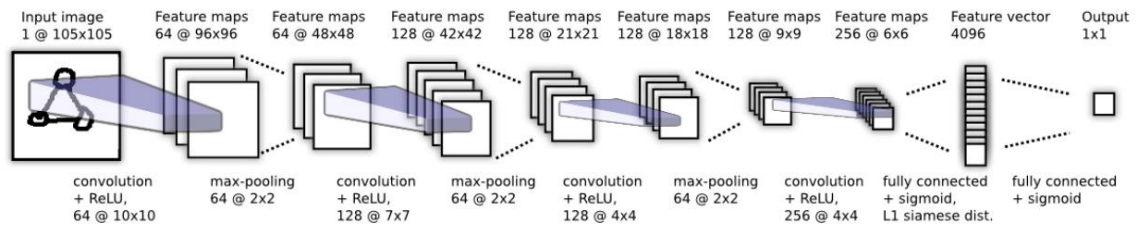
Epoch 47/50
50/50 [=====] - 29s 586ms/step
-0.0 1.0 1.0

Epoch 48/50
50/50 [=====] - 29s 588ms/step
2.920632e-06 1.0 1.0

Epoch 49/50
50/50 [=====] - 29s 586ms/step
-0.0 1.0 1.0

Epoch 50/50
50/50 [=====] - 29s 576ms/step
.....
```

Modelimizin eğitim aşamaları şekildekii gibidir.



Bizim modelimizde de şekildeki gibi alınan inputları pikselleyerek aşağıdaki gibi eğitilmiştir.

Model: "embedding"

Layer (type)	Output Shape	Param #
=====		
input_image (InputLayer)	[(None, 105, 105, 3)]	0
conv2d_4 (Conv2D)	(None, 96, 96, 64)	19264
max_pooling2d_3 (MaxPooling2)	(None, 48, 48, 64)	0
conv2d_5 (Conv2D)	(None, 42, 42, 128)	401536
max_pooling2d_4 (MaxPooling2)	(None, 21, 21, 128)	0
conv2d_6 (Conv2D)	(None, 18, 18, 128)	262272
max_pooling2d_5 (MaxPooling2)	(None, 9, 9, 128)	0
conv2d_7 (Conv2D)	(None, 6, 6, 256)	524544
flatten_1 (Flatten)	(None, 9216)	0
dense_1 (Dense)	(None, 4096)	37752832
=====		
Total params: 38,960,448		
Trainable params: 38,960,448		
Non-trainable params: 0		