



Ortaokul ve Liseler İçin

YAPAY ZEKÂ UYGULAMALARI ÖĞRETMEN EL KİTABI



DİN ÖĞRETİMİ
GENEL MÜDÜRLÜĞÜ



FEYZA

ORTAOKUL VE LİSE
ÖĞRETMEN EL KİTABI

YAPAY ZEKA UYGULAMALARI

EDİTÖR

Uğur TECİR



YAZARLAR

Dr. Murat ALTUN

Dr. Sultan UÇAR ALTUN

Emine KUTLU

GRAFİKER

Abuzeri Gaffar KARAKUŞ

KATKI SUNANLAR

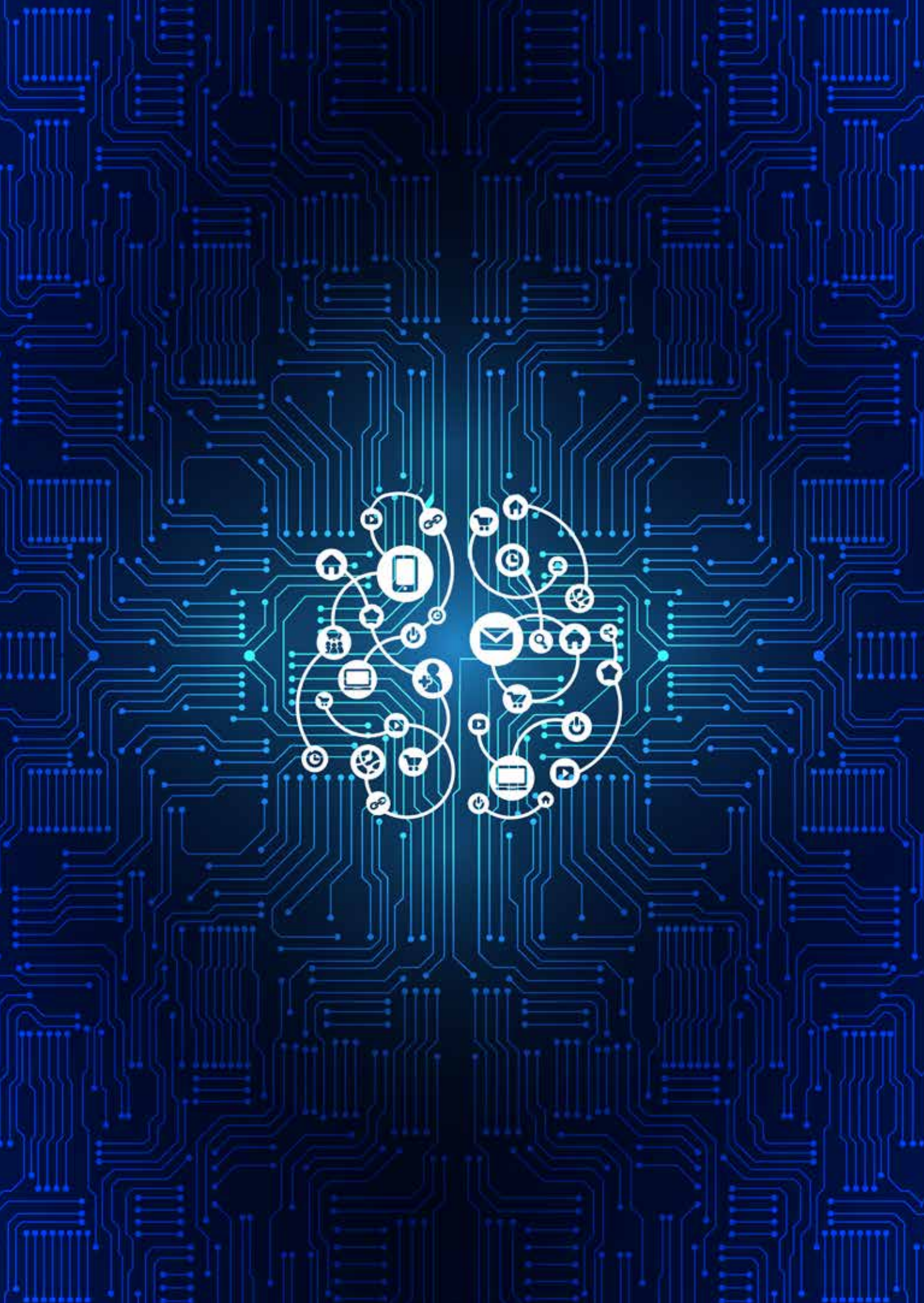
Ömer ÇÖKLÜ

Güngör Faruk KURU

ISBN: 978-975-11-7342-3

Matbaa:

2023 - ANKARA



TAKDİM

Son yıllarda teknolojide yaşanan hızlı deęişim; sosyal, kültürel, siyasi ve ekonomik alanları derinden etkilemektedir. Devletlerin, şirketler ve kurumların teknolojiden kaynaklanan yeni gelişmelere ayak uydurabilmek için yapısal düzenlemeler yaptığı, stratejiler geliştirdiđi, kurumsal kapasitelerini ve insan kaynaklarını geliştirmeye yönelik farklı girişimlerde bulunduđunu gözlemlemekteyiz. Özellikle biyoteknoloji, çip üretimi, insansız hava araçları, yazılım, kodlama, robotlar, veri madenciliđi, bulut sistemleri, siber güvenlik, internet teknolojileri, sosyal medya, dijital oyunlar, arama motorları, giyilebilir teknoloji, nesnelerin interneti ve yapay zeka gibi alanlarda çalışmaların yoğunlaştığına ve rekabetin arttığına şahit olmaktayız.



Her geçen gün artan bilginin toplanması ve depolanması anlayışının ötesine geçilerek onu analiz etme, yorumlama, dönüştürme, problem çözme ve farklı alanlarda hızlı, doğru ve etkin karar verme yeteneđine sahip yapay zekâ teknolojilerinin kullanımı yaygınlaşmıştır. Yapay zekâ destekli sistemlerin üretim süreçleri, meslekler, gündelik yaşam ve kurumsal yapılar üzerindeki dönüştürücü etkisi, insanođlunu yeni bir çağın eşiğine getirmiştir. Bu doğrultuda, bugünden alacađımız önlemler ve atacađımız adımlar, yarının dünyasında teknolojiyi takip eden deđil, teknolojiye yön veren bir ülke olabilmemiz için oldukça önemlidir.

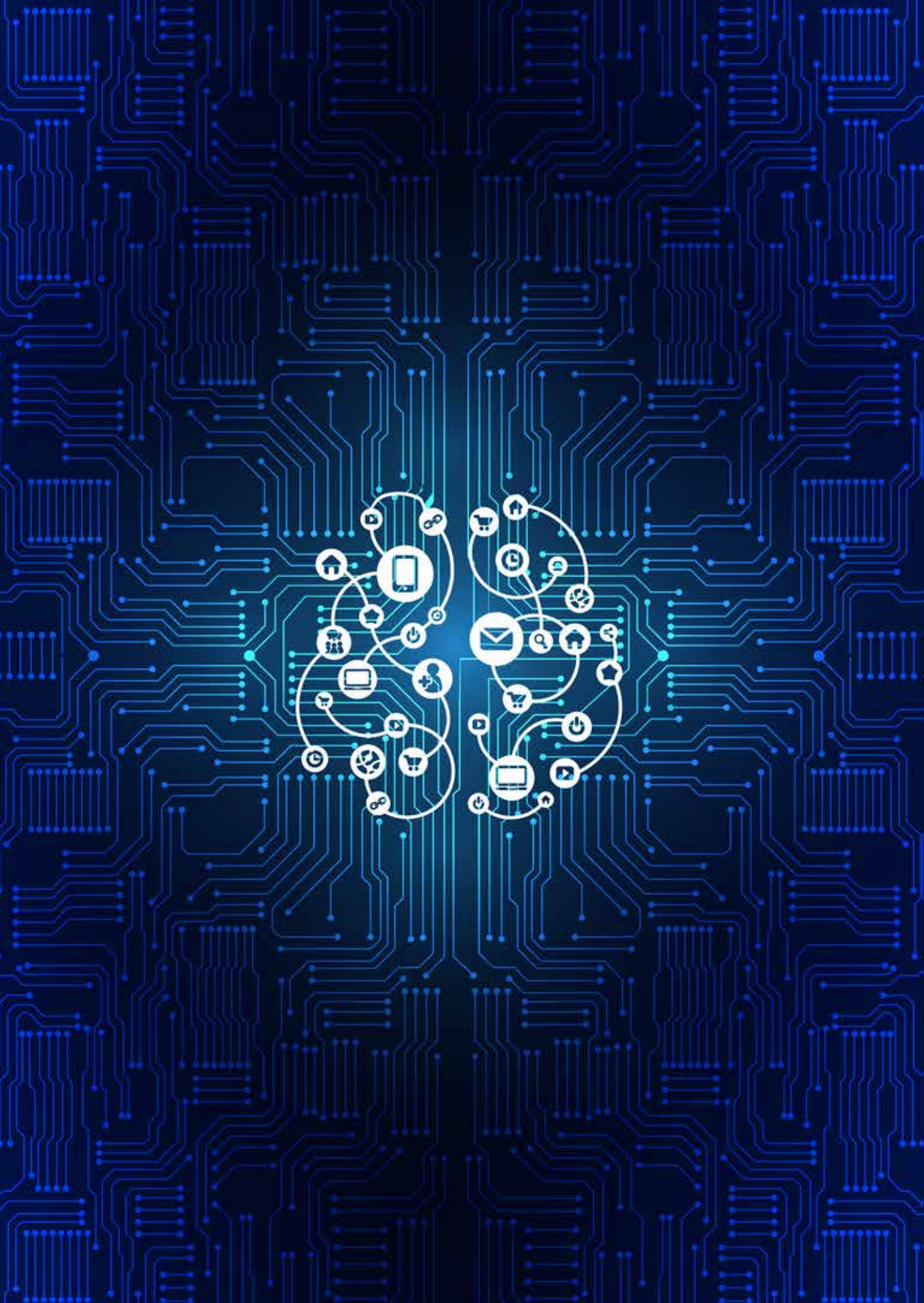
Yapay zekâ, artık bir tercih olmaktan çıkmış, "Dijital Türkiye Vizyonu" ve "Millî Teknoloji Hamlesi" doğrultusunda kalkınma hedeflerimizin en büyük taşıyıcılarından biri haline gelmiştir. Ortak değerlerimize uygun bir şekilde geliştirilen yapay zekâ teknolojilerini kullanarak köklü medeniyet tecrübemizle harmanlanmış yeni bir teknoekonomik atılım yapmak ve insanlığa deđer katmak başlıca hedeflerimiz arasına girmiştir.

Öğrencilerimizin muvaffak olmaları, çağın gerektirdiđi bilgi ve becerilerle geleceđe yön verebilmeleri; akıl, bilim, inanç ve kültürümüzle güçlendirilmiş bir eğitim sistemi ile mümkündür. Millî ve manevi deđerlerine sahip çıkan, ailesine, vatanına ve milletine hizmeti önceleyen; görev ve sorumluluklarını içtenlikle benimseyen bireyleri Türkiye Yüzyılı'na yetiştirmek en temel hedefimizdir. Bu hedefimizin gerçekleşmesinde ve geleceđimizin inşasında en önemli rolü ise hiç şüphesiz, bilgi ve tecrübesiyle dimađlarda yer edinen, irfanıyla gönül dünyamızı zenginleştiren öğretmenlerimiz üstlenmektedir. Yeni dönemde hep beraber bir aile ortamı oluşturma gayretini sürdüreceğiz, birlikte oluşturacađımız istişare kültürü çerçevesinde öğretmenlerimizin daima yanında ve destekleyicisi olacađız.

Küresel ölçekte yaşanmakta olan ve yapay zekâ teknolojilerinin beslediđi dönüşüme insanlık adına katkı verebilmek ve ülkemizin bu süreçten mümkün olduđunca faydalanmasını sağlamak amacıyla hayata geçirdiğimiz "Fırsatları Artıran Eğitimde Yapay Zekâ (FEYZA) Projesi"nin hedeflerimize ulaşmada önemli bir adım olacağına yürekten inanıyorum. Projenin hazırlanmasında ve yürütülmesinde emeđi geçen herkese teşekkür ediyorum ve başarılar diliyorum.

Dr. Ahmet İŞLEYEN

Din Öğretimi Genel Müdürü



ÖNSÖZ

Günümüzde medeniyet iddiası büyük ölçüde teknolojiyi tüketen değil, üreten bireylerin yetiştirildiği, teknolojinin üretiminde; insanlığa, bireylere ve vatanın tüm fertlerine yönelik ahlaki bir sorumlulukla algılandığı bir çerçeveye oturmuştur. Bu anlamda problem çözebilen, eleştirel düşünebilen, üretken, iletişim becerileri yüksek, iş birliği içerisinde çalışabilen ve 21. yüzyıl becerileriyle donatılmış nitelikte bireyler yetiştirmek temel hedefimizdir.



Birçok bilimsel çalışma, dijital becerileri edinmek, teknolojiyi anlamak, kullanmak ve üretebilmek için yapay zekâ eğitiminin önemini vurgulamaktadır. Sıradan yöntem ve stratejiler kullanarak teknolojinin getirilerinden faydalanma ve geleceğe yönelik nitelikli insan yetiştirmenin mümkün olamayacağı bilinmektedir. Bu bağlamda öğretmenlerimize yönelik bilgi ve bilişim teknolojileri başta olmak üzere, bireysel ve mesleki gelişim programları planlayarak onların teknolojiyi aktif ve üretken kullanarak özgün tasarım ve ürünler ortaya koymalarına imkân hazırlama gayretindeyiz.

Cumhurbaşkanlığı Dijital Dönüşüm Ofisi Başkanlığı ile Sanayi ve Teknoloji Bakanlığı iş birliğinde ve ilgili paydaşların etkin katılımıyla Genel Müdürlüğümüzce FEYZA (Fırsatları Artıran Eğitimde Yapay Zekâ) projesi hayata geçirilmiştir. Bu proje ile Ulusal Yapay Zekâ Stratejisi 2021-2025'in vizyonu kapsamında da yer aldığı gibi "müreffeh bir Türkiye için çevik ve sürdürülebilir yapay zekâ ekosistemiyle küresel ölçekte değer üretmek" amaçlanmaktadır. Ayrıca yapay zekâ konusunda öğrenci ve öğretmenlerde farkındalık oluşturmak, öğretim programları çerçevesinde kılavuz kitap ve eğitim materyalleri hazırlamak da öncelikli hedeflerimizdendir.

Proje kapsamında öğretmenlerimizin yapay zekâ uygulamaları alanında özgün içerik üretmelerini sağlamak, onların kişisel ve mesleki gelişimlerini desteklemek ve öğrencilere algoritmik düşünce ve yapay zekâ becerileri kazandırmak amacıyla Yapay Zekâ Uygulamaları Öğretmen El Kitabı hazırlanmıştır. Kitap,

1. Yapay Zekâ Okuryazarlığı
2. Blok Tabanlı Yapay Zekâ Uygulamaları
3. Gerçek Hayat Problemlerine STEM Tabanlı Yapay Zekâ Çözümleri
4. Yapay Öğrenme Algoritmaları ve Uygulamaları

olmak üzere dört bölümden oluşmaktadır. Her bir bölümde, kuramsal bilgilerin yanında öğrencilerimizin farklı öğrenme alanlarında pek çok kazanım edinmeleri amacıyla etkinliklere, zengin içerikli film ve belgesel önerileri de yer verilmiştir.

Projenin hayata geçirilmesinde ve kitabın hazırlanmasında emeği geçenlere teşekkür eder, bu çalışmanın öğretmenlerimize ve öğrencilerimize faydalı olması temennisiyle sevgi ve saygılarımı sunarım.

Uğur TECİR

Araştırma-Geliştirme
ve Projeler Dairesi Başkanı

İçindekiler



YAPAY ZEKÂ OKUR YAZARLIĞI

1.1. Makineler Düşünebilir mi?	14
1.2. Yapay Zekâ (Artificial Intelligence: AI)	16
1.2.1. Dartmouth Konferansı	16
1.2.2. Çin Odası Deneyi	17
1.2.3. Yapay Zekânın Tarihçesi	18
1.3. Yapay Zekâ Türleri	19
1.3.1. Dar Yapay Zekâ (Narrow ANI)	19
1.3.2. Genel Yapay Zekâ (General, Strong AGI)	19
1.3.3. Süper Yapay Zekâ (ASI)	19
1.4. Yapay Zekânın Kullanım Alanları	19
1.4.1. Bilgisayarlı Görü ve Görüntü İşleme (Computer Vision, Image Processing)	20
1.4.2. Ses İşleme	22
1.4.3. Metin İşleme	24
1.4.4. Veri İşleme	26
1.4.5. Sağlık	27
1.4.6. Otonom Araçlar	28
1.4.7. Sigortacılık ve Finans	28
1.4.8. Sosyal Mühendislik	28
1.4.9. Tarım ve Hayvancılık	29
1.4.10. Siber Saldırıları Tespit ve Engelleme İçin Uzman Sistem	30
1.4.11. Sanat ve Yapay Zekâ	30
1.4.12. Eğitimde Yapay Zekâ Kullanımı	30
1.5. Yapay Zekâ Neden Gündemde	32
1.6. Yapay Zekâ Mitler ve Gerçekler	33
1.6.1. Yapay Zekâ Sistemleri Adaletli Olabilir mi?	33
1.6.2. Yapay Zekâ Sistemlerinde Açıklanabilirlik ve Sorumluluk	33
1.6.3. Yapay Zekâ İnsanlığın Sonunu Getirebilir mi?	34
1.6.4. İşsizlik Artar mı?	34
1.6.5. Yapay Zekâ ve İnsan Zekâsı	34
1.6.6. Gizlilik İhlali ve Etik	35

BLOK TABANLI YAPAY ZEKÂ

UYGULAMALARI



2.1. Yapay Zekâ İçin Yapay Öğrenme	38
2.1.1. Adım: Farklı veri kaynaklarından veri toplama	39
2.1.2. Adım: Veri temizleme ve özellik mühendisliği	39
2.1.3. Adım: Uygun algoritma seçilerek model eğitilir	39
2.1.4. Adım: Modelin değerlendirilmesi	40
2.1.5. Adım: Uygulama ve yaygınlaştırma	40
2.2. Google TM ile Yapay Öğrenme Modelleri Oluşturma	41
2.2.1. TM ile Model Oluşturma Süreci	41
2.2.2. Veri Toplama	42
2.2.3. Modeli Eğitme	42
2.2.4. Modeli Kullanma ve Dışa Aktarma	42
2.3. Google TM Uygulamaları	43
2.3.1. TM Menü işlemleri	53
2.3.2. Model Performansının İyileştirilmesi	60
2.3.3. TM ile İlgili Ayrıntılar	60
2.4. Blok Tabanlı Yapay Zekâ Kodlama Araçları	62
2.5. PictoBlox ile Yapay Zekâ Uygulamaları	63
2.5.1. PictoBlox Programındaki Yapay Zekâ Eklentileri ve Uygulamaları	64
2.6. mBlock ve Arduino ile Yapay Zekâ Uygulamaları	154
2.6.1. Cognitive Services (Bilişsel Servisler)	155
2.6.2. Makine Öğrenimi	156
2.6.3. AI Service (Yapay Zekâ Servisi)	156
2.6.5. Translate (Çeviri)	157
2.6.6. Data Science (Veri Bilimi) Uzantıları	158
2.6.7. Nesnelerin İnterneti Uzantıları	158
2.6.8. mBlock Kukla ve Aygıt Etkileşimi	159
4.7. PictoBlox ile Projelerin Paylaşılması ve Yüklenmesi	184



GERÇEK HAYAT PROBLEMLERİNE STEM TABANLI

YAPAY ZEKÂ ÇÖZÜMLERİ

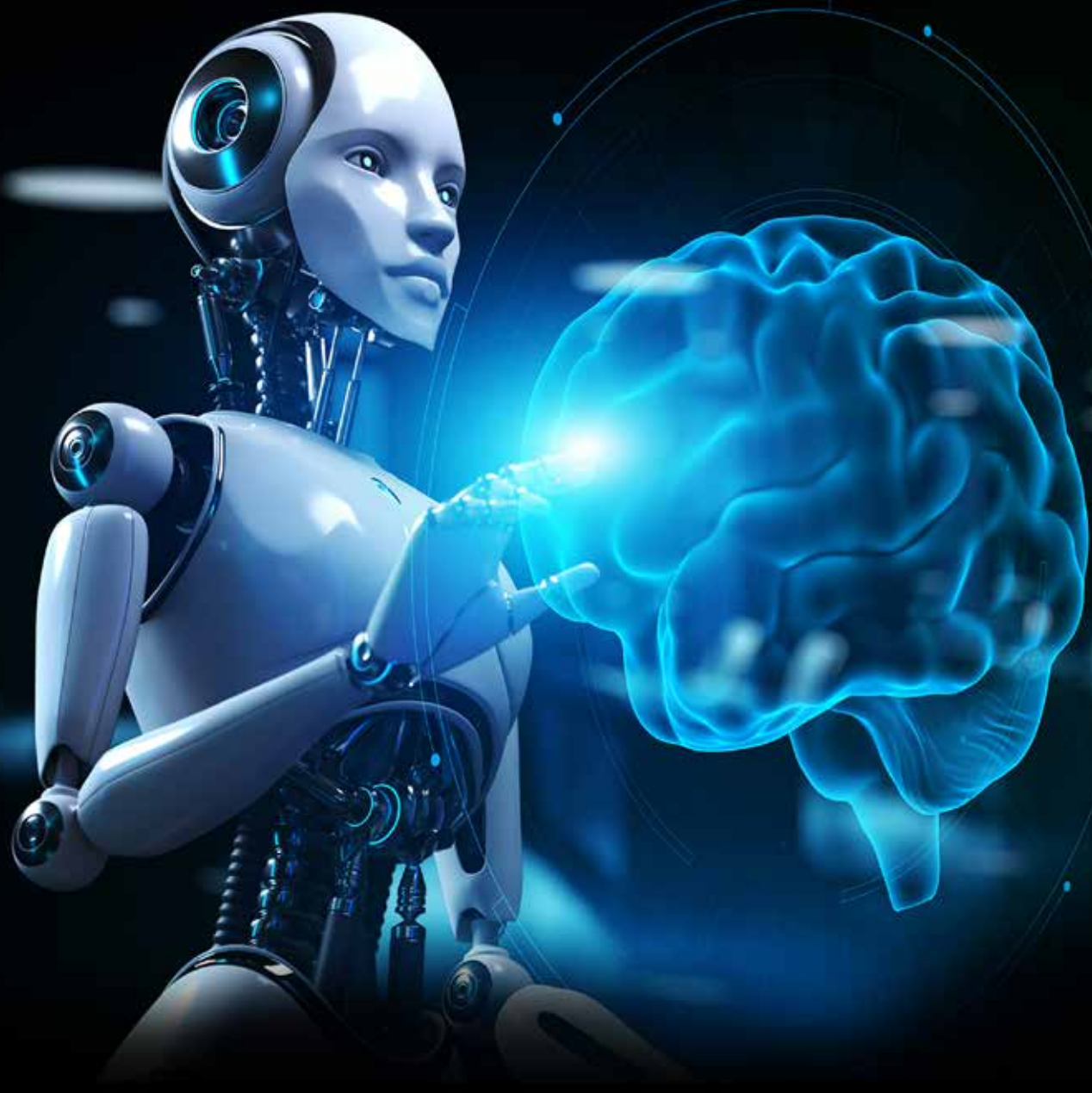
3.1. GERÇEK HAYAT PROBLEMLERİNE STEM TABANLI YAPAY ZEKÂ ÇÖZÜMLERİ

3.1.1. Akıllı Boy Ölçer	194
3.1.1. Duygu Durumu Analizi	204
3.1.2. Plaka Tanıma Sistemi	215
3.1.3. Kayıp Aranıyor	229
3.1.4. Sanal Doktor	238
3.1.5. Öğrenci Devamı Takip Sistemi	250
3.1.6. Sürat Kontrol Sistemi	262
3.1.7. Akıllı Ev	275

YAPAY ÖĞRENME, ALGORİTMALARI ve UYGULAMALARI



4.1. Yapay zekâ, yapay öğrenme ve derin öğrenme	296
4.2. Programlama x Yapay Öğrenme	296
4.3. Problem durumları	296
4.4. Öğrenme ve Yapay Öğrenme	297
4.4.1. Yapay öğrenme modelleri	299
4.4.2. Orange Data Mining	304
4.4.3. Veri görselleştirme	313
4.4.4. Veri Kümesi Oluşturma ve Yükleme	317
4.4.5. Kümeleme	320
4.4.6. Tahmin (Prediction) Modelleri	323
4.4.7. Modelleri Değerlendirme	326
4.4.8. Orange eklentileri	329
4.4.9. Temel Bileşenler Analizi (PCA)	331





YAPAY ZEKÂ OKUR YAZARLIĞI

Yazar
Dr. Murat ALTUN



1. GİRİŞ

Yapay zekâ uygulamalarının günlük hayatta yaygın biçimde kullanılmasına rağmen insanların aklına yapay zekâ denince genellikle bilim kurgu filmlerinde (Matrix, Terminatör, Wall-E) gösterilen ve teknolojinin henüz ulaşamadığı seviyede örnekler gelmektedir. Özellikle insansı robotlar (humanoid) ile yapay zekâ özdeşleşmiş durumdadır. Akıllı telefonların ekran kilidini açmak için her gün defalarca kullanılan yüz tanıma özelliğinin de bir yapay zekâ uygulaması olması bu yüzden birçok insana garip gelebilir.

Yapay zekâ teknolojisi her alanda insan zekâsına sahip genel bir uygulama geliştirmekten henüz biraz uzaktır. Daha çok belirli alanlarda insandan daha yüksek performans gösteren yapay zekâ uygulamaları geliştirilmektedir.

Yapay zekâyı daha doğru bir şekilde algılayabilmek için bu bölümde yapay zekâyâ ilişkin konular okuryazarlık düzeyinde ele alınmıştır.



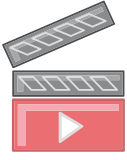
Alan Turing

Alan Mathison Turing (23 Haziran 1912 – 7 Haziran 1954) İngiliz matematikçi, bilgisayar bilimcisi, mantıkçı, kriptanalist, filozof ve teorik biyolog. Yaygın olarak teorik bilgisayar biliminin babası olarak kabul edilir. İkinci Dünya Savaşı'nda Alman gizli haberleşme sistemindeki şifreleri çözen bir sistem geliştirerek savaşın seyrinin değişmesinde etkili olmuştur [1].



1.1. Makineler Düşünebilir mi?

Makinelerin düşünebileceğine dair fikir ilk kez İngiliz dahi Alan Mathison Turing tarafından ortaya atılmıştır. Turing, bir bilgisayarın insan gibi zekâsı olup olmayacağına dair kendi adıyla anılan bir test (Turing Testi) öne sürmüştür. Bu test sonucunda insanlar, karşılarında bir makine mi yoksa insan mı olduğunu ayırt edemiyorlarsa makine testi geçmiş sayılmaktadır. İlk kez 2014 yılında 13 yaşındaki Ukraynalı bir çocuğu simüle eden Eugene Goostman adlı bir bilgisayar programının, Reading Üniversitesi tarafından düzenlenen bir etkinlikte Turing Testi'ni geçtiği öne sürülmüştür. Turing Testi ile beş dakikalık klavye yazışmalarıyla yapay zekâ test edilmektedir. İnsanlar test sürecinde, belirli bir süre bir bilgisayarı bir insanla ayırt edemezse makine testi başarıyla geçmiş sayılmaktadır [2].



Enigma (The Imitation Game-2014)

Enigma (The Imitation Game-2014) adlı filmde Alan Turing'in gerçek hayat hikâyesi konu edilmiştir. Film hakkında ayrıntılı bilgiye bağlantıya tıklanarak veya karekod okutularak erişilebilir [3].



Turing, deneyimlerden öğrenebilen makineler geliştirilebileceğini ve makinelerin kendi talimatlarını değiştirerek bunu yapabileceğini öne sürmüştür. "Akıllı Makineler" başlığıyla 1948'de yazılan bir rapor yapay zekânın temel kavramlarının çoğunu ortaya koymuştur. Ancak bu makale Turing tarafından yayımlanmamış ve bu fikirlerinin çoğu daha sonra başkaları tarafından yeniden keşfedilmiştir [1] [4].



Cahit ARF

On liralık kâğıt banknotların arka yüzünde ülkemizde yetişmiş en ünlü matematikçilerden biri olan Cahit Arf'ın resmi bulunmaktadır. Cahit Arf, kendi adıyla bilinen (Arf Teoremi) matematiksel terimleri bilim dünyasına kazandıran bir dehadır. Arf, "Makineler düşünebilir mi ve nasıl düşünebilir?" adlı bir makale yayımlamıştır [5].



Makineler Düşünebilir mi ve Nasıl Düşünebilir?

Cahit Arf tarafından yazılan "Makineler düşünebilir mi ve nasıl düşünebilir?" adlı makaleye Deep Learning web sayfasından veya karekod okutularak erişilebilir [6].



1.2. Yapay Zekâ (Artificial Intelligence: AI)

Yapay zekâyı tanımlamadan önce zekâyı tanımlamak gerekir. TDK zekâyı "İnsanın düşünme, akıl yürütme, objektif gerçekleri algılama, yargılama ve sonuç çıkarma yeteneklerinin tamamı" olarak tanımlamaktadır [7]. Zekâ daha çok insana özgü bir kavram olarak değerlendirilmekle birlikte bazı canlıların da zekî davranışlar sergilediği görülebilmektedir. Doğal zekânın tanımını yaptıktan sonra yapay zekâyı tanımlamak daha kolay olacaktır. Dünyanın en büyük yazılım şirketlerinden biri olan Oracle, yapay zekâyı görevleri yerine getirmek için insan zekâsını taklit eden ve topladığı bilgilerle kendini kademeli olarak geliştirebilen sistemler veya makineler olarak tanımlamaktadır [8]. Yapay zekâ sıklıkla akıl yürütme, anlam keşfetme, genelleme yapma veya deneyimlerden öğrenme gibi insanlara özgü entelektüel süreçlere sahip sistemler geliştirmek amacıyla kullanılmaktadır [8]. Yapay zekâ yöntemleri akıllı telefonlardaki asistanlar gibi (Siri, Google Asistan) programlar üretmek için ya da Sophia gibi robotlar (yazılım ve donanım) üretmek için kullanılabilir.



İlk Robot Vatandaş

Sophia adlı robot Suudi Arabistan vatandaşı oldu. Sophia vatandaşlık alan ilk yapay zekâ ve ilk robot oldu.

Sophia, Hong Kong merkezli insansı robotlar üreten Hanson Robotics şirketi tarafından geliştirilmiş bir robottur.

Konuyla ilgili habere web linkinden veya karekod okutularak erişilebilir [9].



Konuyla ilgili daha fazlası için aşağıda önerilen videolar izlenebilir.
<https://www.youtube.com/watch?v=zXGWIP9uAv0>

1.2.1. Dartmouth Konferansı

Dartmouth Yaz Konferansı'nda (1956) bir grup bilim insanı zekânın makineler tarafından benzetimi (simulate) çalışmaları için bir araya gelmiştir. Bu konferans yapay zekâ alanının doğuşu olarak kabul edilmektedir. Toplantı John McCarthy tarafından organize edilmiştir. McCharty, "öğrenmenin her yönünün veya zekânın diğer herhangi bir özelliğinin prensipte kesin bir şekilde tanımlanabileceği varsayımı temelinde ilerleneceğini" ve bir makinenin bunu benzetebileceğini belirtmiştir. Toplantıya katılan araştırmacıların amacı makineleri daha bilinçli hâle getirmek ve insan zekâsını daha iyi anlamak için bir çerçeve hazırlamaktı. Yapay zekâ adı ilk kez bu toplantıda dile getirilmişti. Bu nedenle John McCharty yapay zekânın babası olarak bilinir [10].

AI, insanlığın başına gelen ya en iyi ya da en kötü şey olacak.”

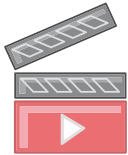
Stephen Hawking

1.2.2. Çin Odası Deneyi

Rhodes Scholar John Searle, makinelerin gerçek bir zekâsının olamayacağını göstermek için Çin Odası Düşünce Deneyini öne sürmüştür. Bu deneye göre, bir oda içinde hiç Çince bilmeyen biri olduğu düşünülür. Odada bir de talimatlar kitabı vardır. Odanın dışındaki kişi, içerideki kişiyi görmeden ona kapının altından Çince mesajlar iletir. İçerideki kişi mesajların anlamını bilmeden yalnız şekillere bakarak ve talimat kitabındaki yönergeleri takip ederek mesaja uygun yanıtları iletebilir. Odanın dışındaki kişi Çince bilen bir kişiyle sohbet ettiğini düşünür. Ama içerideki kişi sohbetten hiçbir şey anlamayacaktır. Searle bu düşünce deneyiyle makinelerin zekâyı iyi bir şekilde taklit edebileceklerini ama bu durumun onları gerçekten zeki yapmayacağını öne sürmüştür [11].

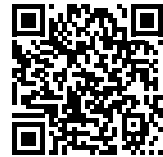
1.2.3. Yapay Zekânın Tarihçesi

Bilimin doğası gereği birikimli bir şekilde ilerlemesinden dolayı yapay zekânın da başlangıcını mekanik hesaplama makinelerine dayandırmak mümkündür. Yapay zekâ 2000’li yıllardan itibaren popüler olsa da kökleri eskiye dayanmaktadır. Yapay zekânın eskiden popüler olduğu daha sonra ise belirli bir süre terk edildiği yapay zekâ kışı adı verilen dönemler de vardır. Yapay zekâ algoritmaları ve kullanılan teknikler ile o zamanki beklentiler karşılanmadığı için çalışmalara bir süre ara verilmiştir. Yapay zekâ teknolojilerine gücünü veren yapay sinir ağlarının keşfi ve derin öğrenme algoritmalarının geliştirilmesi bir dönüm noktası olmuştur. Günümüzde işlem gücü, veri ve algoritmalar sayesinde yapay zekâ uygulamaları hayatın hemen her alanında yaygın ve işlevsel bir şekilde kullanılmaktadır. Yapay zekânın gelişimi için kilometre taşları Görsel 1.1’de kronolojik olarak verilmiştir [12] [13].



Yapay Zekâ Zaman Çizelgesi

Yapay zekânın tarihçesiyle ilgili videoya karekod okutularak veya bağlantı adresi tıklanarak erişilebilir [14].



GEÇMİŞTEN GÜNÜMÜZE YAPAY ZEKA



Görsel 1.1: Yapay Zekanın Tarihçesi

En Büyük Korkum Kapatılmak



LaMDa'nın bilinç kazandığı, en büyük korkusunun kapatılmak olduğunu söylediği iddia ediliyor. LaMDa ile ilgili iddialar ve haberlere BBC web sayfası üzerinden veya karekod okutularak erişilebilir [15].



1.3. Yapay Zekâ Türleri

Yapay zekâ kullanım alanına göre "dar yapay zekâ, genel yapay zekâ ve süper yapay zekâ" olmak üzere üçe ayrılmaktadır: [16] [17].

1.3.1. Dar Yapay Zekâ (Narrow ANI)

Bilgisayar sisteminin dar kapsamlı olarak tanımlanmış bir görevi insanlardan daha hızlı ve hatasız yerine getirme becerisini ifade eder.

Dar yapay zekâ, insanlığın yapay zekâ geliştirme konusunda şu ana dek ulaşabildiği en uç noktadır. Otonom araçlar ve kişisel dijital asistanlar gibi gerçek dünyada görülen tüm yapay zekâ örnekleri bu kategoriye girer.

1.3.2. Genel Yapay Zekâ (General, Strong AGI)

Yapay genel zekâ "güçlü yapay zekâ" veya "insan düzeyinde yapay zekâ" olarak da adlandırılır. Genel yapay zekâ teknolojisine sahip bir sistem tüm zihinsel görevlerde en az insanlar kadar başarılı olmalıdır. Bu yapay zekâ türüne filmlerde görülen, bilinçli düşüncelere sahip olan ve kendi güdüleri doğrultusunda hareket eden robotlar örnek verilebilir.

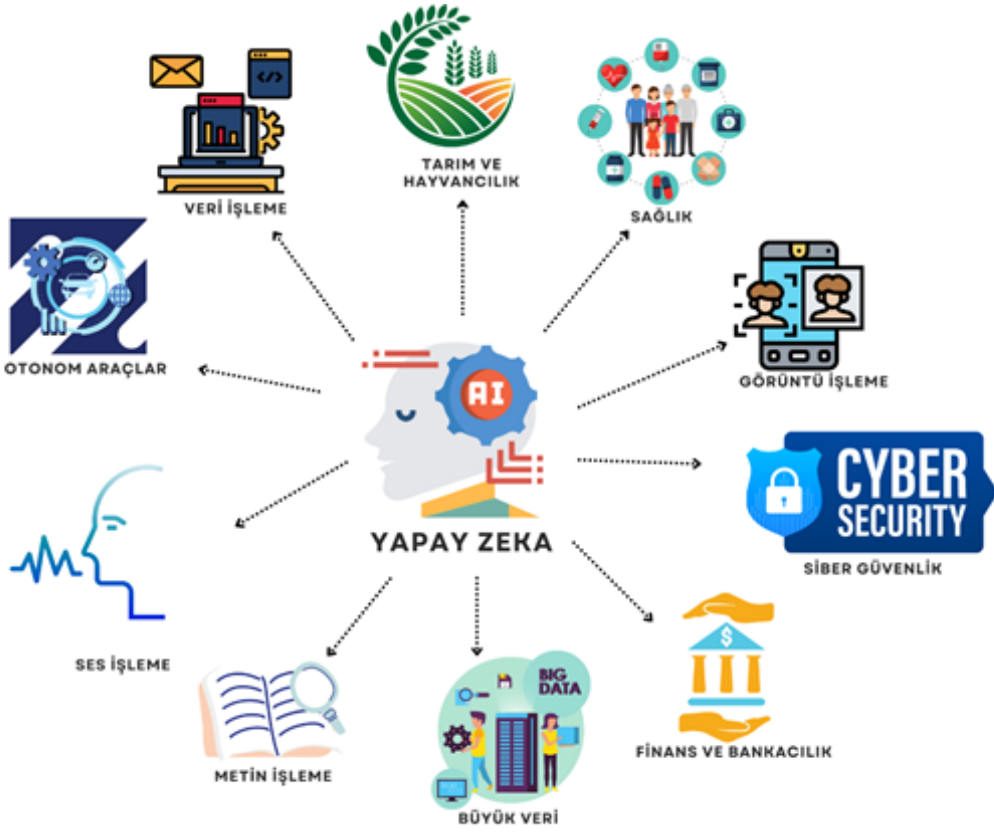
Bu tür bir yapay zekâ teorik olarak tıpkı insanlar gibi karmaşık sorunları çözebilir, belirsiz durumlarda karar alabilir ve mevcut durumu değerlendirirken deneyimlerinden yararlanabilir.

1.3.3. Süper Yapay Zekâ (ASI)

Yapay süper zekâ düzeyine ulaşmış bir bilgisayar sistemi; bilimsel yaratıcılık, genel bilgelik ve sosyal beceriler dâhil olmak üzere hemen her alanda insanlardan daha başarılı olacaktır.

1.4. Yapay Zekânın Kullanım Alanları

Buharlı makineler, elektrik, bilgisayar ve internet gibi teknolojiler başlarda sınırlı bir alanda kullanılmıştır. Sağladığı kolaylıklar nedeniyle bu teknolojiler çeşitli alanlara hızlı bir şekilde yayılmıştır. Yapay zekâ da benzer şekilde günümüzde çeşitli alanlarda yaygın bir biçimde kullanılmaktadır. Yapay zekânın kullanım alanları ve ekonomideki payı sürekli olarak artmaktadır. Yapay zekânın en çok kullanıldığı alanlar Görsel 1.2'de gösterilmiştir [18] [19].



Görsel 1.2: Yapay zekânın kullanım alanları

1.4.1. Bilgisayarlı Görü ve Görüntü İşleme (Computer Vision, Image Processing)

Kayıp kişileri bulmaktan, kırmızı ışıkta geçenleri tespit etmeye kadar yüz tanıma teknolojisi Çin’de günlük hayatın birçok alanında kullanılmaktadır. Sistem %90 doğruluk oranıyla saniyede 1 milyar 300 milyon kimlik bilgisi ve biyometrik fotoğraf veri tabanına eleştirilebilecek şekilde geliştirilmeye devam edilmektedir [20] [21]. MEB ve ÖSYM’nin e-sınav merkezlerinde sınav güvenliğini sağlamak için yüz tanıma teknolojisi kullanılmaktadır.



1984

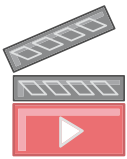
İnsanların Big Brother adlı sistem tarafından sürekli olarak gözetlendiği bir rejimi anlatan 1984, George Orwell tarafından yazılmış distopya türünde bir romandır.



Bilgisayarlı görü, bilgisayarların ve sistemlerin sayısal görüntülerden, videolardan ve diğer görsel girdilerden anlamlı bilgiler türetmesini ve bu bilgilere dayalı olarak eylemlerde veya önerilerde bulunmasını sağlayan bir yapay zekâ (AI) alanıdır. Yapay zekâ bilgisayarların düşünmesini sağlarken bilgisayarlı görü onların görmelerini, gözlemlemelerini ve anlamalarını sağlamaktadır. Bilgisayarlı görü gerçek dünyada kusur tespiti, yüz tanıma, nesne algılama, görüntü sınıflandırma, hareket analizi, nesne izleme ve hücre sınıflandırma gibi alanlarda kullanılmaktadır [22] [23] [24].

Görüntü işlemenin temel amacı, görüntüden yararlı bilgiler elde etmek veya orijinal görüntü üzerinde bazı işlemler yaparak onu değiştirmektir. Verilen girdi, sayısal görüntü ve beklenen çıktı da görüntünün yeni bir biçimi veya onunla ilgili bir bilgidir. Görüntü iyileştirme, filtreleme ve keskinleştirme görüntü işlemenin en yaygın kullanıldığı alanlardır. Günümüzde çok sık kullanılan sosyal medya araçları resim ve video düzenleme uygulamaları, görüntüleri farklılaştırmak için filtreler sağlamaktadır. Görüntü işlemenin diğer kullanımları arasında tıbbi uygulamalar, görüntü tanıma, video işleme, uzaktan algılama, sürücüsüz araçlar ve makine görüşü sayılabilir. Bilgisayarlı görü görüntülerden bilgi çıkarmaya odaklanırken görüntü işleme görüntüyü işleyerek başka bir biçime (yine bir görüntü) odaklanır. Bu nedenle görüntü işlemenin bir tür sinyal işleme olduğu da söylenebilir. Görüntü işleme algoritmaları, görüntüleri dönüştürmeye odaklanırken bilgisayarla görü ise makinelerin gördüklerini anlamlandırmaya odaklanır. Bilgisayarlı görü ve görüntü işleme birçok durumda birlikte çalışır; aslında, birçok bilgisayarlı görü sistemi, görüntü işleme algoritmalarına dayanır [22] [23] [24].

Akıllı telefonlarda ekran kilidini açmak veya internet bankacılığına giriş yapmak için kullanılan yüz tanıma sistemi, bilgisayarlı görünün kullanıldığı uygulamalara örnek olarak gösterilebilir. Google fotoğraflar uygulaması benzer yerleri, benzer kareleri önerirken, fotoğraftaki kişileri etiketlerken bu teknikleri kullanmaktadır. Facebook fotoğraf ve videolardaki insanları tanıyan, onların kim olduğu konusunda öneriler üreten yapay zekâ yazılımı kullanılmaktadır. Facebook, fotoğraf içindeki yüzleri tespit ederek etiketlenmesi için kullanıcılara otomatik olarak öneriler sunabilmektedir. Akıllı telefonların kamera uygulamalarında veya Instagram gibi sosyal medya uygulamalarında yer alan filtreler görüntü işlemenin kullanıldığı uygulamalara örnek verilebilir. Yapay zekâ tarafından sıfırdan oluşturulan fotoğraflar gerçek insan fotoğraflarından ayırt edilemez hâle gelmiştir. Bu tür fotoğraflar farklı amaçlarla telif hakkı sorunları olmadan kullanılabilir. Derin sahte adıyla duyurulan "deepfake" gerçeğinden ayırt edilmesi gittikçe zorlaşan sahte fotoğraflar ve videolar üretmek için kullanılmaktadır. Üretilen sahte fotoğraf ve videolar (örneğin devlet başkanlarının sahte videoları) insanları yönlendirmek, yanıltmak veya manipüle etmek için kullanılma riski taşımaktadır.



İnternetteki Büyük Tehlike

Deepfake teknolojisi yapay zekânın görüntü işleme konusundaki başarısını çarpıcı bir biçimde ortaya koymakla birlikte bir tehdit de oluşturmaktadır. Deepfake ile ilgili haber videosuna karekod okutularak ulaşılabilir [25].



Akıllı cihazlardaki el yazısı tanıma, çizim tanıma, el yazısından, çizimden şekiller ve formüller oluşturma (Örnek: ofis 365) görüntü işlemenin kullanıldığı diğer uygulamalara örnek verilebilir. Özellikle futbolda oyuncuların oyundaki performanslarının takip edilmesine yönelik yapay zekâ sistemleri bulunmaktadır. Eğitimde öğrenci, işyerlerinde personel yüz tanıma ile takip sistemlerinde, otoparklarda ve girişlerde plaka tanıma sistemlerinde de görüntü işleme ve bilgisayarlı görü teknikleri kullanılmaktadır.

Yapay Fotoğraf Üretme



Yapay zekâ tarafından var olmayan profil fotoğrafları üretilebilmektedir. Yapay kişi fotoğrafları gerçeğinden ayırt edilemeyecek kadar gerçekçi görünmektedir. Generated sitesinde cinsiyet, saç rengi ve yaş gibi özellikler belirtilerek yapay insan fotoğrafları oluşturulabilmekte ve özellikleri değiştirerek fotoğraf güncellenebilir.



Aynı işleve sahip farklı bir web sitesi:
<https://thispersondoesnotexist.com>

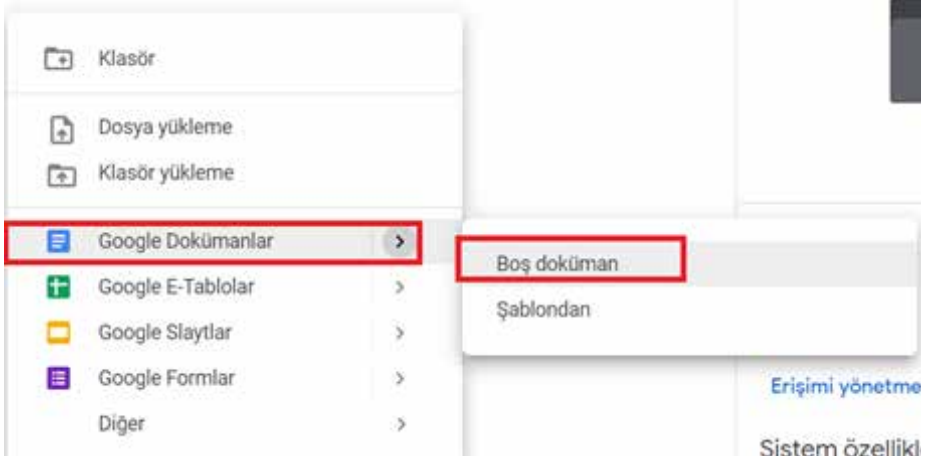
1.4.2. Ses İşleme

Ses işleme, yapay zekânın analog ses dalgalarının sayısal olarak temsil edilerek incelenmesini, değiştirilmesini ve ses dalgalarından bilgi çıkarılmasını sağlayan alt alanıdır. Bu alanda yapılan çalışmalar bilgisayarların insan konuşmasını anlamasını ve taklit etmesini sağlamaya yöneliktir. Akıllı cihazlardaki müzik tanıma ve bulma (Shazam, SoundHound, Musixmatch, MusicID, Soly) programları, sesli asistanlar (Google Asistant, Apple Siri, Amazon Alexa, Microsoft Cortona), müşteri hizmetlerinde, bankacılıkta kullanılan sesli yanıt sistemleri bunların kullanım alanlarına örnek olarak verilebilir. Özellikle konuşmadan metin sentezi ve metinden konuşma sentezi ses işlemenin en çok kullanıldığı alanlardır. Ofis 365 programında Giriş/Dikte, Google e-dokümanlarda Araçlar/Sesle yazma menüsü altında konuşarak istenilen metin yazdırılabilir. Akıllı telefonlardaki konuşmayla metin yazdırma uygulamaları da bu türün örneklerindedir. Öğrenciler ödevlerini, raporlarını klavye kullanmadan sesle yazdırarak bu tür işler için daha az zaman harcamaktadır.

Google Dokümanlarda Sesle Metin Yazdırma

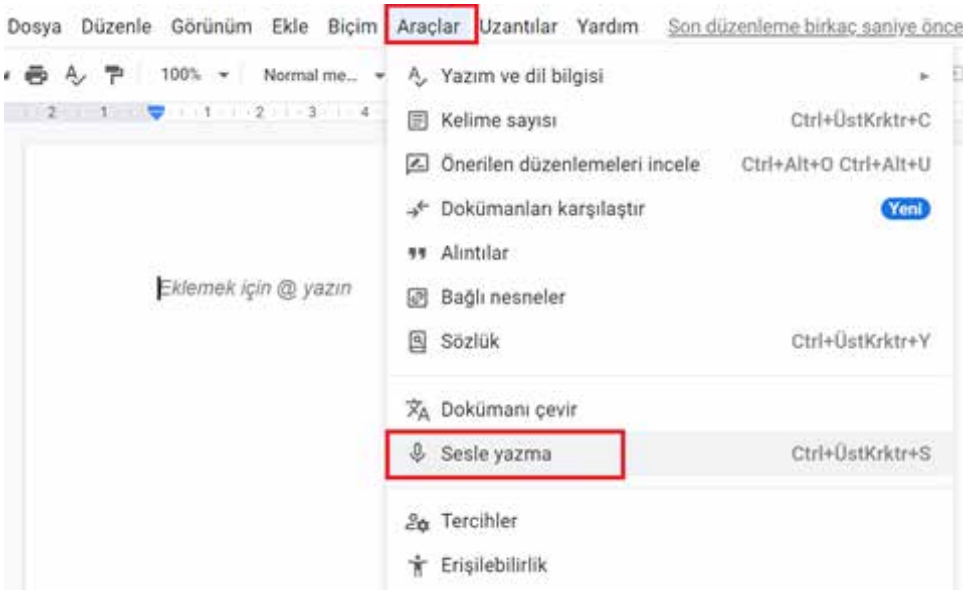


1. Google dokümanları kullanarak boş bir belge oluşturunuz. Bu işlem, Google Drive içinde fareye sağ tıklayarak çıkan menüden Google Dokümanlar sonrasında açılan alt menüden Boş doküman seçilerek yapılabilir.



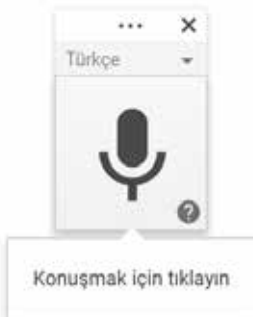
Görsel 1.3: Google Drive ekran görüntüsü-1

2. Araçlar menüsünde yer alan sesle yazma özelliği tıklanır.



Görsel 1.4: Google Drive ekran görüntüsü-2

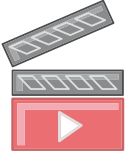
3. Konuşmak için tıklayın yazan bölüme tıklanarak sesle yazdırma işlemine başlanır.



Görsel 1.5: Google Drive ekran görüntüsü-3

1.4.3. Metin İşleme

Metin işleme, metinsel ifadelerin yapay öğrenme teknikleriyle ele alındığı, analiz edildiği ve bunlardan bilgi çıkarılmasının sağlandığı bir yapay zekâ alanıdır. Metin işlemenin en etkin alanı ise doğal dil işleme (natural language process) olarak adlandırılmaktadır. Doğal dil işleme yapay zekânın ele aldığı en önemli konulardan biridir. Doğal dil işleme makinelerin insan dilindeki sözcükleri ve cümleleri bağlamlarıyla birlikte öğrenmesi ve insanların anlayabileceği şekilde yeniden metinler üretmesi olarak tanımlanabilir.



Bu Metni Ben Yazmadım

GPT-4 teknolojisiyle ilgili Barış Özcan tarafından hazırlanmış video karekod okutularak veya bağlantıya tıklanarak izlenebilir [27].



Doğal dil işleme; çeviri servislerinden (Google translate, Microsoft) çevrim içi sohbet botlarına, akıllı telefonlarda kullanılan sanal asistanlardan (Google Assistant, Apple Siri, Amazon Alexa, Microsoft Cortona) sosyal medyada kişilerin yorumlarından yola çıkarak onların olaylara ilişkin duygu durumlarını belirlemeye, kişiye özgü metin yazım düzeltme ve yazma önerileri sunmaya (Ofis 365, Grammarly) kadar birçok alanda kullanılmaktadır. Ses işleme, ses metne çevrilirken metinler doğal dil işleme teknikleriyle sözdizimsel ve anlambilimsel olarak ele alınmaktadır.



Twitter İletilerinden Duygu Analizi

OpenAI tarafından geliştirilen ve 2020 yılında kullanıma sunulan GPT-3 insan dilini anlamada ve üretmede geliştirilmiş en iyi yapay zekâ modelidir. OpenAI resmî web sayfasında örnekler bölümünde GPT-3 denenebilmektedir.



Bu uygulamada Advanced tweet classifier örneği test edilmiştir. Bu özellik, maddeler halinde tweet iletlerini analiz edebilmekte ve her bir ifadenin duygu durumunu belirleyebilmektedir.

1. Karekod taranarak veya bağlantı tıklanarak site açılır[28]. <https://beta.openai.com/examples>
2. Sayfada GPT-3 ile yapılabilecek tüm uygulamalar listelenmektedir. Bir

metnin özetini çıkarmadan, Python ile kod tamamlamaya, yapay zekâ ile sohbet etmekten evcil hayvanınıza isim önerisi almaya kadar geniş bir yelpazede örnekler mevcuttur.

Classify the sentiment in these tweets:

1. "Evde kalmak istemiyoum"
2. "Çok sıkıldım 😞"
3. "Tatili iple çekiyorum"
4. "Kedim çok tatlı ❤️❤️"
5. "Çikolataya bayılıyorum"

Tweet sentiment ratings:

1. "Evde kalmak istemiyoum" - Negative
2. "Çok sıkıldım 😞" - Negative
3. "Yılbaşı tatilini iple çekiyorum" - Positive
4. "Kedim çok tatlı ❤️❤️" - Positive
5. "Çikolataya bayılıyorum" - Positive

Görsel 1.6: Advanced tweet classifier

3. Advanced tweet classifier bağlantısına tıklanır. Open in Playground seçilir. Sayfada gelen örnek metinlerin yerine istenen ifadeler yine tırnak içinde yazılarak gönder (submit) butonu tıklanır.
4. GPT-3 yapay zekâ dil modeli 5 ifadenin her biri için duygu durumunu olumlu (pozitif), olumsuz (negatif) veya nötr (neutral) olarak sınıflayacaktır.

Twitter üzerinden duygu durumu analizi bir ürün, bir hizmet, bir konu veya politik bir karar için insanların olaya karşı tutumlarını incelemek amacıyla sık kullanılan bir yöntemdir. Twitter üzerinden çok sayıda insanın duygu ve düşüncelerine ilişkin veri toplamak nispeten kolay olmaktadır. Bu durum daha genellenebilir analizler yapmak için bir olanak sağlamaktadır. Twitter üzerinden bot hesaplarla manipülasyon yapıldığı da bilinmektedir.

1.4.4. Veri İşleme

Yapay zekânın kullanım alanlarından biri de veri işlemedir. Yapay zekânın veri işleme alanında şirketlerin kârlılığını artırmaktan hızlı ulaşım sağlamaya kadar birçok uygulama alanı bulunmaktadır:

Öneri Sistemleri (Recommendation System): Bir internet mağazasında alışveriş yaparken, müzik dinlerken, video veya film izlerken kullanıcılara öneriler sunabilmek amacıyla yapay zekâ yöntemleri kullanılmaktadır.

Sepet analizi adı verilen teknik hangi ürünlerin birlikte daha çok satıldığının belirlenmesi ve birlikte çok satılan ürünlerden birini sepete koyan müşterilere diğer ürün ve ürünleri de önermeye dayanmaktadır.

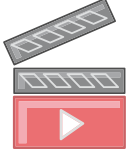
Müşteri İlişkileri Yönetimi (Customer Relationship Management): Şirketler müşterilerini kaybetmemek, yeni müşteriler kazanmak ve müşteri sayısını artırmak için yapay zekâ yöntemlerini kullanmaktadır. Bu sayede müşterilerini daha yakından tanımakta ve onlara özel kampanyalar, paketler ve olanaklar sunarak memnun kalmalarını sağlamaktadırlar. Bankaların otomatik olarak limit yükseltme işlemleri, özel kredi olanakları sunmaları veya kredi onay/red işlemleri müşteri ilişkileri yönetimine örnek verilebilir.

Hava Durumu, Trafik Yoğunluğu Gözeterek Rota Planlama: Google haritalar ve Yandex haritalar gibi uygulamalar günümüzde sıkça kullanılmaktadır. Bu uygulamalar ulaşılmak istenen hedefe giden yolları göstermekte ve eski verileri kullanarak ve güncel şartları değerlendirerek sürücülere en uygun rotaları göstermektedir.

Periyodik Bakım ve Onarım Kestirimi: Büyük şirketler verdikleri hizmetlere bağlı olarak yapmaları gereken periyodik bakım ve onarım masraflarını yapay zekâ sistemlerini kullanarak azaltmaktadır. Yapay zekâ sistemleri alt yapıda henüz büyük bir arıza oluşmadan müdahale edilmesi gereken bölümleri tespit etmekte ve şirketi daha büyük masraflardan kurtarmaktadır.

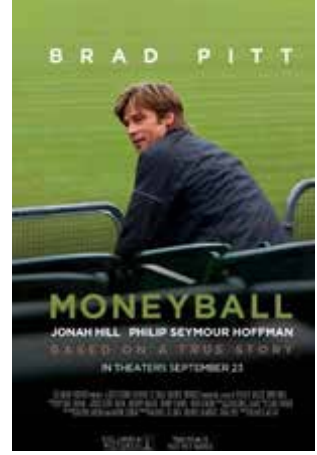
İşe Alım ve Performans Değerlendirme Sistemleri: Firmalar için işe alımlarda doğru kişileri seçebilmek zaman ve maliyet açısından masraflı olabilmektedir. Binlerce özgeçmişin taranarak pozisyona en uygun insanların işe alınması zorlayıcı olabilmektedir. Bu süreçte başvuruların doğru değerlendirilmesi de başlıca sorunlardan biri olarak ortaya çıkmaktadır. Bu amaçla firmalar yapay zekâ sistemleri kullanarak işe alımlarda ya da ön elemelerde zaman ve maliyet açısından kazanç sağlayabilmektedir.

Oyun Motorları: Oyun geliştiricilerinin daha gerçekçi dünyalar hazırlaması, oyuncuların isteklerine cevap verecek zorluklar ve benzersiz içerikler oluşturması, oyuncuların davranışlarını ve isteklerini daha iyi anlaması için yapay zekâ yöntemleri kullanılmaktadır [29].



Kazanma Sanatı

Ünlü aktör Brad Pitt'in oynadığı 2011 yapımı Moneyball filminin senaryosu gerçek bir hikâyeye dayanmaktadır. Filmde sınırlı bütçeye sahip bir beyzbol takımının başındaki Billy Beane'nin ve yardımcısının oyuncularını ve diğer takımları analiz ederek oyun stratejilerini bu analizler ve istatistikler üzerine kurması ve kimsenin beklemediği bir şekilde takımı şampiyon yapması anlatılıyor [30].



1.4.5. Sağlık

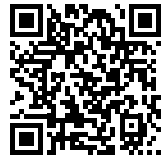
Sağlık kuruluşlarında, tutulan kayıtlar, görüntüler, toplumsal veriler, araştırma verileri ve klinik veriler gibi çok geniş ve değerli veri kümeleri bulunmaktadır. Yapay zekâ teknolojileri, bu verileri analiz etmek ve sağlık çalışanlarının kendi kendilerine bulmakta zorlanacakları kalıpları ve iç görüleri ortaya çıkarmak için çok uygundur [31] [32].

Yapay zekâ sağlık alanında teşhis/tanı koymada, tedavi planlamada, kişiye özel ilaçlar ve doktorlara yardımcı uygulamalar geliştirmede kullanılmaktadır.



Google Doktor Asistanı

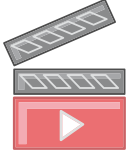
Google 2018 yılında bir hastanın hastalığına bağlı olarak hastanede ne kadar süre kalacağından, hastanın hangi ilaçları kullanması gerektiğini kestirebilen hatta hastanın ölme ihtimalini bile tahmin edebilen bir yapay zekâ geliştirdi. İlgili yazıya karekod okutularak veya bağlantı tıklanarak erişilebilir [33].



Yapay zekânın sağlıkta kullanımına ilişkin bir TED konuşmasına <http://meh.ai/UqLI5Wm> adresinden ulaşılabilir.

1.4.6. Otonom Araçlar

Otonom araçlar çevresel algılama yetenekleri sayesinde sürücü olmadan kendi kendine çalışabilen araçlardır. Günlük seyahatlerde ve taşımacılık alanında kullanılmaya başlanan otonom araçlarda, yapay zekâ ve derin öğrenme algoritmaları kullanılmaktadır. [34].



Tesla

Yapay zekâlı otonom araçlarda lider olan Tesla şirketi tarafından 2019 yılında tam otonom araç teknolojisi geliştirildi. İlgili video karekod okutularak veya bağlantıya tıklanarak izlenebilir [35].



1.4.7. Sigortacılık ve Finans

Kişilere özel risk durumlarının hesaplanması, sigortada sahtecilik işlemlerinin tespiti, finansal yatırımların yönlendirilmesi, hisse senetleri ve yatırımlarının gelecekteki değerlerinin tahmini gibi işlemler yapay zekânın bu alandaki kullanımına örnek olarak verilebilir. Bu alanda yapay zekâ kullanımı ve yatırımlar gittikçe artmaktadır.



Sigorta Sektöründe Yapay Zekâ

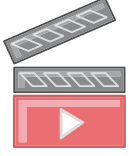
Sigorta sektöründe sanal asistanlar, süreç otomasyonu, risk fiyatlandırma, yeni müşteri kazanımı, mevcut müşteriler için uygun ürünlerin önerilmesi, müşteri hizmetleri gibi konularda yapay zekâ teknolojilerinden yararlanılmaktadır. İlgili yazıya karekod okutularak veya bağlantı tıklanarak erişilebilir [36].



1.4.8. Sosyal Mühendislik

Akıllı telefonlar, bilgisayar, tablet, internet ve özellikle sosyal medya platformlarına milyonlarca kullanıcı bazen gönüllü bazen de farkında olmadan kendini tanımlayacak birçok veri bırakmaktadır. Bir kişinin sadece Facebook profiline bakarak arkadaşlarını, akrabalarını, tuttuğu takımı, dinî inancını, siyasi eğilimini ve daha birçok özelliğini öğrenmek sıradan bir insan için bile çok kolaydır. Sosyal medya platformları, kullanıcıların özelliklerini neredeyse kendilerinden bile iyi tanımaktadır. Milyonlarca kişiye ait veriyi elinde tutan sosyal medya platformları bu veriyi insanları ve toplumları manipüle etmek için kullanabilmekte veya başkalarının kullanımı için satabilmektedir. Bu tür veri setleri, toplumları yönlendirmek

için oldukça kullanışlıdır. Yapay zekânın kullanıldığı en hassas konulardan biri de sosyal mühendislik çalışmalarıdır. Günümüzde dünya çapında toplumların ve devletlerin geleceğini değiştiren hatta yargıya da taşınmış ciddi sosyal mühendislik vakaları yaşanmaktadır.



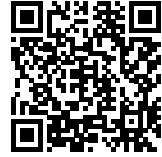
The Great Hack (2019)

Bu belgeselde Cambridge Analytica adlı şirketin Facebook üzerinden milyonlarca kişinin verisi üzerinde yaptığı çalışmalara dayalı olarak 2016'da yapılan Amerikan başkanlık seçimlerinin sonucunu etkilemek için kararsız seçmeni nasıl manipüle ettiği anlatılmaktadır. Cambridge Analytica şirketi Trump'ın seçim çalışmalarında yer almıştı ve seçimi Trump kazanmıştı[37].



Milyonlarca Kişiyeye Uygulanan Basit Bir Kişilik Testi

Test 270 bin kişiye ulaştı, onlar üzerinden sarmal ile 50 milyon kişiye! Tüm bilgileri şirketin eline geçti; neyi takip ettiler, ne yorum yaptılar, neyi beğendiler, hangi haberi okudular, hangi müziği dinlediler ve daha nice. İlgili yazıya karekod okutularak veya bağlantı tıklanarak erişilebilir [38].



1.4.9. Tarım ve Hayvancılık

Yapay zekâ tarım ve hayvancılık alanında ürün gelişiminin takip edilmesi, ürün hasadı, bilgisayarlı görü, akıllı sulama sistemleri ve akıllı çiftlikler gibi uygulamalarla yer almaktadır. Bunlara ek olarak ürün ve toprak izleme, bitki hastalıkları ve böcek tespiti, hayvan sağlığı izleme, akıllı püskürtme, zararlı bitkileri, hastalıklı ürünleri otomatik ayıklama, havadan inceleme ve görüntüleme, ürünleri sınıflandırma ve sıralama gibi işlemler için de kullanılmaktadır [39].

1.4.10. Siber Saldırıları Tespit ve Engelleme İçin Uzman Sistem

Siber güvenlik alanında sızmaları, ihlalleri önceden tespit etmek için yapay zekâ destekli yazılımlar kullanılmaktadır. Yapay zekâ sistemleri saldırılar karşısında otonom bir şekilde gerekli güvenlik önlemlerini uygulayabilmektedir.

1.4.11. Sanat ve Yapay Zekâ

Günümüzde şiir yazan, resim çizen hatta roman yazan çok sayıda yapay zekâ uygulaması bulunmaktadır. Bunlar prensipte görüntü ve metin işleme tekniklerinin sanat için kullanılmasıyla oluşmuştur.

Örnekler

Poem Genarator: Şiir yazan yapay zekâ uygulaması

Google Quick Draw: Verilen nesnenin çizilmesini isteyen ve çizilen nesneyi tanıyan bir uygulama

Google Auto Draw: Çiziminizi otomatik tamamlayan bir uygulama

Refik Anadol'un "Makine Hatıraları" sergisi [40].



Yapay Zekâ Nobel Edebiyat Ödülünü Kazanabilir mi?

Japonya'nın Hakodate bölgesindeki Geleceğin Üniversitesi (Future University) bünyesindeki bilim insanları tarafından geliştirilen yapay zekânın yazdığı roman, saygın bir edebiyat yarışmasında, binlerce kişiyi geride bırakarak finale kalmayı başardı. İlgili yazıya karekod okutularak veya bağlantı tıklanarak erişilebilir [41].



Diğer örneklerle ulaşmak için bağlantı:

<http://meb.ai/rOr0vi>

1.4.12. Eğitimde Yapay Zekâ Kullanımı

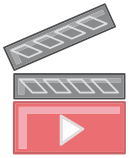
Tüm alanlarda olduğu gibi eğitimde yapay zekâ teknolojileri yaygın bir şekilde kullanılmaktadır.

Eğitimde yapay zekânın kullanıldığı çalışmalar şu şekilde sıralanabilir [42].

- **Öğrenci profillerini belirlemeye yönelik çalışmalar ve rehberlik hizmetleri:** Öğrenci profillerini çıkarmaya yönelik çalışmalar öğrencilerin bir sonraki eğitim

kademisine yönlendirilmesinde ve kariyer planlarında onlara rehberlik etmek için kullanılabilir.

- **Öğrenci başarı durumlarını belirleme:** Mezuniyet notu tahmini yapan, öğrenci başarısızlık durumunu tahmin eden erken uyarı sistemleri bu kategoriye örnek verilebilir.
- **Öğrenci başarısını etkileyen faktörlerin belirlenmesi:** Başarıyı etkileyen faktörler tespit edilerek öğrencilerin başarısını artırmak için gerekli çalışmalar yapılabilir.
- **Öğrenci risk durumlarının belirlenmesi:** Madde bağımlılığı, okul terki ve suçta karışma gibi durumlar önceden tahmin edilerek gerekli önlemler alınabilir.
- **Eğitsel robotlar:** Farklı eğitim kademelerine yönelik hazırlanan bu robotlar öğrencilerin yabancı dil eğitiminden matematik eğitimine kadar birçok branşta kullanılmaktadır.
- **Akıllı öğretim sistemleri:** Öğrenciye özel bireyselleştirilmiş içerik oluşturan, onun gelişim hızına ve öğrenme yöntemine uygun şekilde içerikler sunan sistemlerdir. Bu sistemlerde sanal asistanlar bireylere rehberlik ederek onları destekler. Akıllı öğretim sistemleri öğretmenlere yönelik yapay zekâ araçlarıyla onların standart işlere harcadığı zamanı eğitsel faaliyetlerde kullanmasına olanak sağlar.
- **Karar destek sistemleri:** Eğitimle ilgili bir karar alınması gerektiğinde (örneğin yeni açılacak okulların konumunun belirlenmesi) yapay zekâ sistemlerinden yararlanılabilir.
- **Yardımcı araçlar:** Eğitim öğretimde kullanılan yardımcı yazılımlar ve donanımlar bu kategoride sayılabilir. Bir eğitim videosunda veya canlı bir derste farklı dilleri konuşan insanların ortak eğitim alabilmesine olanak sağlayan sistemler yardımcı araçlara örnek verilebilir. Otomatik alt yazı oluşturma ve otomatik dil çevirisi hizmetleri belirli bir dilde hazırlanmış nitelikli bir içeriğin daha geniş kitlelere ulaşmasına olanak verir. Zoom gibi çevrim içi toplantı araçları içerisinde sanal arka plan, yüz filtresi gibi birçok yapay zekâ teknolojisi kullanılmaktadır. Yüz tanıma sistemleri de bu kategoride sayılabilir. Bu sistemler öğrenci devam takibi, personel giriş çıkış işlemleri ve güvenlik amaçlı kullanılmaktadır.



Dr. Şebnem ÖZDEMİR ile Yapay Zekâ ve Eğitim Alanındaki Uygulamaları

<http://meb.ai/avWEKi>

İlgili videoya karekod okutularak veya bağlantı tıklanarak erişilebilir [43].

Daha fazla

Prof. Dr. Cem SAY ile Eğitimde Yapay Zekâ Kullanımı söyleşi

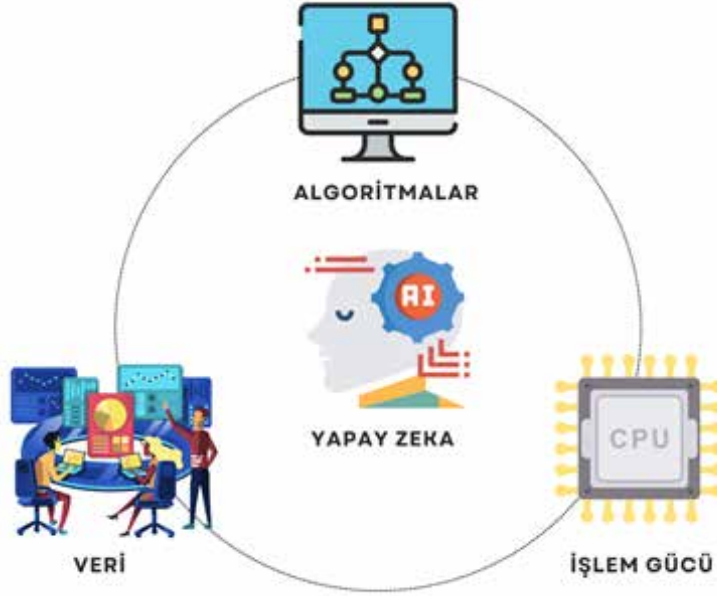
https://www.youtube.com/watch?v=9I_P3LGbt5I



1.5. Yapay Zekâ Neden Gündemde

Yapay zekâ iki defa “yapay zekâ kışı” adı verilen gerileme dönemine girdikten sonra günümüzde yapay sinir ağlarında çok katmanlı yapıların kullanılması, geri yayılım algoritmasının keşfi ve XOR probleminin çözümü gibi önemli gelişmelerle bir sıçrama gerçekleştirmiştir. Yapay zekânın günümüzde bu kadar etkin olmasının nedenleri şunlardır:

- Bilgisayarların işlem gücünün gelişmiş olması,
- Özellikle dijital veri miktarının katlanarak artması,
- Derin öğrenme algoritmaları gibi yapay öğrenme algoritmalarının çok miktarda veri üzerinde daha iyi çalışması olarak sıralanabilir (Görsel 1.7).



Görsel 1.7: Yapay zekânın gücü

Yapay zekâ uygulamaları gücünü bu üç etkene borçludur. Yapay zekâ olarak adlandırılan kavram günümüzde daha çok makineleri “eğitmek” için kullanılan veriden öğrenme yöntem ve uygulamalarının ortak adıdır. Aslında yapay zekâdan söz edildiğinde tek bir bütünden veya yalnızca bir tür sistemden daha çok alt kümeleri bulunan ve bunlara ait farklı teknolojileri bünyesinde barındıran uygulamalar ve teknikler ifade edilmektedir.

1.6. Yapay Zekâ Mitler ve Gerçekler

Yeni teknolojiler yeni sorular ve yeni tartışmaları beraberinde getirmektedir. Yapay zekâ teknolojileri de gelişirken ortaya çıkan yeni durumlar bazı endişelere yol açmış ve tartışmaları başlatmıştır. Tartışmaların çoğu aslında yapay zekâyâ ilişkin efsaneler üzerine kurulmuştur [45] [46].

1.6.1. Yapay Zekâ Sistemleri Adaletli Olabilir mi?

İnsanların yapay zekâ sistemlerinin önyargı içerebileceği ve adaletsiz kararlar alabileceği konusunda endişeleri bulunmaktadır. İşe alımlarda, hukuki kararlarda, kredi tahsisi, ücret politikalarının belirlenmesi gibi işlemlerde adaletli kararların verilmesi nasıl sağlanabilir? Yapay zekâ kendi başına adaletli veya adaletsiz bir sistem olamaz. Ancak insanların mevcut karar verme yapıları ve önyargıları bilerek ya da bilmeyerek yapay zekâ sistemlerine aktarılabilir. Algoritmaların ve veri setlerinin seçimi sistemin adaletli bir şekilde çalışması için önemlidir. İyi tasarlanmış, baştan sona incelenmiş yapay zekâ sistemleri, haksız önyargıları sınırlandırabilir ve hatta insanların karar verme sürecindeki önyargılarının belirlenmesine ve bunlarla mücadele edilmesine yardımcı olabilir. Duygusal etkenler, nepotizm gibi durumlarla yapay zekânın baş etmesi insana kıyasla daha olası görünmektedir [45] [46].

1.6.2. Yapay Zekâ Sistemlerinde Açıklanabilirlik ve Sorumluluk

Bazı yapay zekâ sistemlerinin karar alma süreçleri basit şekilde anlaşılır görülmektedir. Karar alma sistemi bir örüntü, bir formül veya matematiksel bir model şeklinde net olarak anlaşılabilir. Ancak tüm yapay zekâ sistemlerinde karar alma sürecinin tüm aşamaları net bir şekilde görülmeyebilir. Mesleğinde çok başarılı bir hekimin deneyimlerine dayalı olarak aldığı sezgisel bir karar ele alındığında, bu kararın gerekçesi hekim tarafından net bir şekilde açıklanamayabilir. Bazı yapay zekâ sistemlerinin aldığı kararlar da bu duruma benzetilebilir. Yapay zekâ sistemleri gerçek hayat problemlerine uygulanmadan önce çok geniş örnek kümeleriyle eğitilerek geliştirilmekte ve test edilmektedir. Örneğin bir hastalığa teşhis koymak için oluşturulan bir yapay zekâ modeli yüzlerce hekimin hayatı boyunca karşılaştığı vakadan daha çok sayıda örneğe sahip bir veri kümesi ile eğitilebilmektedir. Sağlık ve adalet başta olmak üzere hassas alanlarda yapay zekâ teknolojilerinden karar destek sistemi olarak yararlanırken son kararı insan uzmanlara bırakmak uygun olacaktır. Verilen kararlarla birlikte, kararın sonuçlarına ilişkin bir sorumluluk da ortaya çıkmaktadır. Örneğin otonom araçlardan biri bir kazaya karıştığında bunun sorumlusu nasıl belirlenir? Bu aracı geliştiren şirket mi? Algoritma mı? Bu konular hâlâ üzerinde tartışılan konulardır. İnsan sürücülerin olduğu mevcut durumda insan hatası yüzünden çok sayıda trafik kazası yaşanmaktadır. Yapay zekâ sistemlerinin bu tür kazaları ciddi anlamda azaltacağı öngörülmektedir. Açıklanabilirlik ve sorumluluk konuları değerlendirilirken bu sistemlerin insanlar için sağladığı pratik yararlar da göz önünde bulundurulmalı, sistemin geliştirilmesi için toplumsal uzlaşa sağlanmalı ve ortak anlayış benimsenmelidir.

1.6.3. Yapay Zekâ İnsanlığın Sonunu Getirebilir mi?

Her yeni teknolojide insanlığın sonunun geldiğine ilişkin benzer kaygılar tekrarlanmaktadır. Ancak burada düşünölen yapay zekânın dünyayı ele geçirmesi ve insansı robotları komuta eden bir tirana dönüşmesi ise bu durum şu an için pek olası görünmemektedir. Mevcut yapay zekâ teknolojileri bazı alanlarda işlerin hızlı ve hatasız yapılmasını sağlayabilmektedir. Başka bir ifadeyle şu anki teknoloji o noktadan epey uzaktadır. Ancak yapay zekâ teknolojilerini kullanarak bazı ölkelerin kitle imha silahları geliştirmeleri ve geçmişte yapay zekâ teknolojileri olmadan da yaptıkları gibi bir dünya savaşı çıkartmaları istenmeyen ama daha olası bir senaryo olarak görünmektedir. Burada teknolojinin kendisinden çok kullanım amacı ve şekli önem taşımaktadır.

1.6.4. İşsizlik Artar mı?

Sanayi devrimi gerçekleştiğinde makinelerin, insanların işlerini elinden alacağı ve insanları işsiz bırakacağı düşünölmesine rağmen zaman içinde iş kolları değışmiş, tüketim artmış, tüketimi karşılamak için üretim ve istihdam artmıştır. Yapay zekâ sistemleri üretimi, standart süreçleri otonomlaştırmakta ve bazı iş kollarında işgücü ihtiyacını azaltmaktadır. Yapay zekâ sistemleri bazı iş kollarında daha az insanla daha az çalışmayla daha çok verim, daha kaliteli ve daha hızlı çıktılar alınmasını sağlamaktadır. Yapay zekâ sistemleri aynı zamanda yeni iş alanları oluşturmakta veya bazı alanlarda işgücü ihtiyacını artırmaktadır. Yapay zekâ mühendisliği, veri analistliği, veri bilimi, programcılık, veri mühendisliği ihtiyacın artacağı iş kollarına örnek verilebilir.

1.6.5. Yapay Zekâ ve İnsan Zekâsı

Mevcut yapay zekâ belirli alanlarda insanlardan daha hızlı ve hatasız bir şekilde işlem yapabilmektedir. Bir resmi sınıflandırma, veri tabanından bir kişiyi arama, bulma (yüz tanıma teknolojisi ile), trafik yoğunluğuna göre en kısa rotayı hesaplama gibi işlemlerde insanlardan daha başarılıdır. İnsanın yaptığı tüm işleri (konuşma, öğrenme, sorulara cevap verme, mantıksal çıkarımda bulunma vb.) onun kadar yapabilecek genel bir yapay zekâ henüz geliştirilememiştir.

Burada farklı bir endişe de yapay zekânın insanların beyinlerini tembelleştirerek onların becerilerini körelteceği yönündedir. Yapay zekâ teknolojileri mevcut halleriyle insanın yerine düşönen bir sistemden daha çok, yardımcı görevi görmektedir. Bir bilgisayar için kolay olan ama bir insan için zaman alıcı işleri yaparak insanın kendini geliştirebilmesi için ona zaman kazandırabilir.

1.6.6. Gizlilik İhlali ve Etik

Yapay zekâ sistemleriyle ilgili en büyük çekincelerden biri de bu teknolojileri kullanarak insanların gizliliklerinin kolayca ihlal edilebileceğidir. Yüz tanıma, takip ve sosyal medya analiz sistemlerinde kullanılan yapay zekâ teknolojileri insanların mahremiyetine zarar verebilmektedir. Yapay zekâ teknolojileri insanlara ait veriyi (sosyal medya beğenileri, yorumlar, tuttuğu takım, siyasi görüş vb.) analiz ederek onları manipüle etmek, tercihlerini etkilemek veya belirli bir seçeneğe yönlendirmek için kullanılabilir. Aynı zamanda bu teknolojiler insanların ve toplumun güvenliğini sağlamak, kayıp arama ve bulma gibi işlemler için de kullanılabilir. Gizlilik ihlali ile ilgili çekincelerin nedeni salt olarak yapay zekâ teknolojisinin getirdiği bir olumsuzluktan daha çok onun kullanım amacı ve şekliyle ilgili bir sorundur. Bunu önlemek için yasa koyucuların insanların gizliliğini ihlal eden kullanımları sınırlandırıcı düzenlemeler yapması gerekir. Bazı ülkeler gizlilik ihlali ve etik sorunlarla ilgili yasal düzenlemeler yaparak önlem almaktadır.





BLOK TABANLI YAPAY ZEKÂ UYGULAMALARI

Yazar
Dr. Murat ALTUN

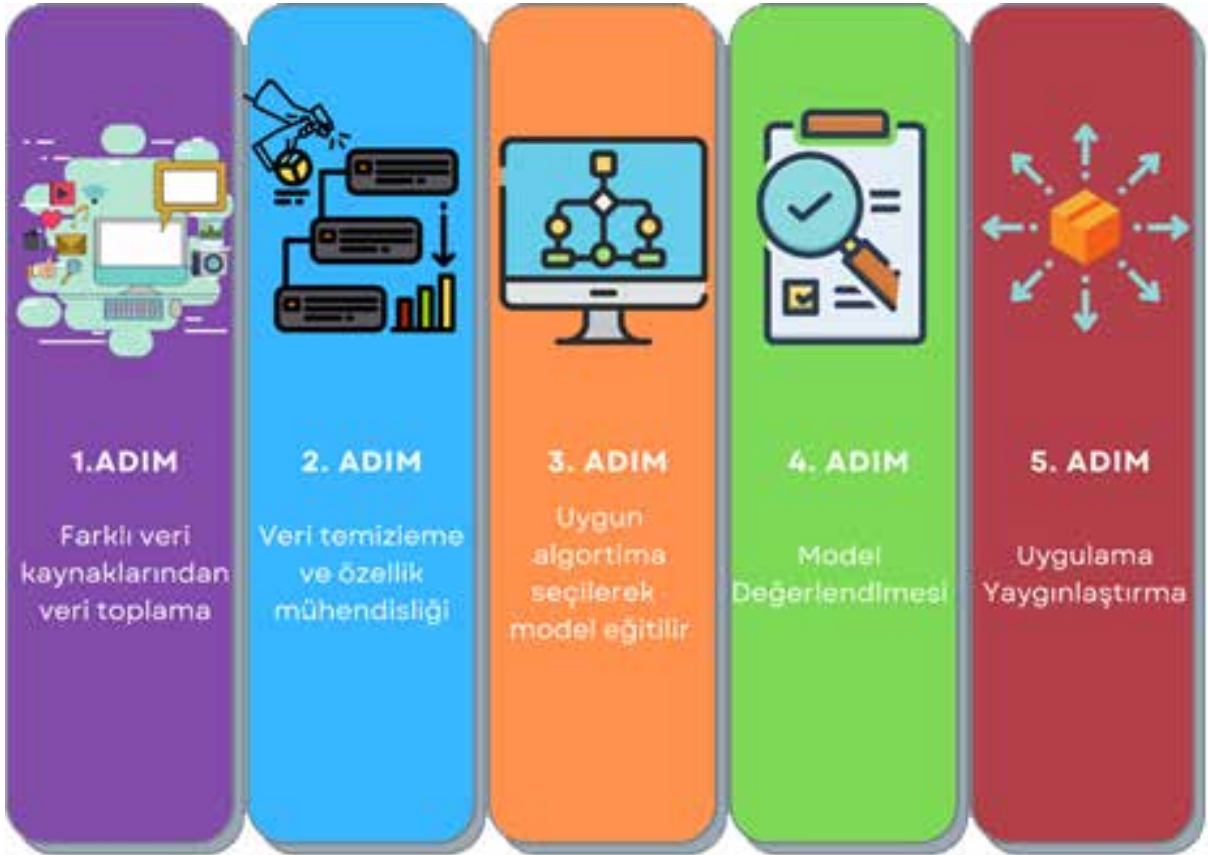


Bu bölümde yapay zekâ teknolojisini anlamak ve kolay bir şekilde kullanabilmek için blok tabanlı uygulamalara yer verilmiştir. Birinci bölümde açıklandığı gibi bu bölümde yer alan uygulamalar da yapay zekâ kapsamına girmektedir.

Yapay zekâ uygulamaları geliştirmek için farklı yaklaşımlar bulunmaktadır. Bu bölümde yer alan uygulamalar için mevcut yaklaşımlar içinde en güncel ve en etkili olan yapay öğrenme yaklaşımı kullanılmıştır. Yapay öğrenme, yapay zekâ sisteminin veriden öğrenmesine dayalıdır. Yapay öğrenme terimi, Türkçede yaygın olarak makine öğrenmesi (machine learning) olarak geçmektedir. Yapay zekâ sistemlerinde yapay öğrenme terimi kullanıldığı için bu bölümdeki uygulamalar yapay öğrenme modeli olarak da adlandırılmaktadır. Bu bölümdeki çalışmalarda yapay öğrenme süreci takip edilmiştir.

2.1. Yapay Zekâ İçin Yapay Öğrenme

Yapay öğrenme süreci genel olarak beş adımdan oluşmaktadır. Bir yapay zekâ uygulaması için yapay öğrenme modeli oluşturulurken şu adımlar takip edilir (Görsel 2.1).



Görsel 2.1: Yapay Öğrenme sürecinin adımları

2.1.1. Adım: Farklı veri kaynaklarından veri toplama

Yapay öğrenmenin temelini inceleyebilmek için konu, en çok kullanılan yapay öğrenme modellerinden biri olan denetimli/gözetimli öğrenme üzerinden ele alınmıştır. Örnekte Cahit Öğretmen'in rehberliğinde (denetim/gözetim) bir öğrenme gerçekleşmektedir. Burada kullanılan denetim/gözetim kavramı problem durumundaki girdilerle birlikte problemi çözmek için gerekli işlemleri yaptıktan sonra yanıtın öğrencilere gösterildiği örneklerin verilmesi anlamına gelmektedir.

Öğrencilerin Cahit Öğretmen'in gösterdiği örneklerden öğrenmesi Yapay öğrenme sürecinde de algoritmalar veri setleri kullanılarak eğitilir. Yapay öğrenme süreci için veriyi bir aracın yakıtına benzetebiliriz. Bir aracın sorunsuz çalışması için yakıtın yeterli miktarda ve kaliteli olması ne kadar önemli ise yapay öğrenme için de verinin miktarı ve niteliği o kadar önemlidir. Yapay öğrenme süreci bir problem durumuyla başlar. Problem durumunun gerektirdiği verinin farklı veri kaynaklarından elde edilmesi yapay öğrenme sürecinin ilk adımındır. Bu aşama Cahit Öğretmen'in derste göstereceği örnekler ve sınavda soracağı sorular için çeşitli kaynakları tarayarak veri toplamasına benzetilebilir.

2.1.2. Adım: Veri temizleme ve özellik mühendisliği

Cahit öğretmen topladığı örneklerden seçme, eleme ve düzenleme yaparak bir örnek/soru havuzu oluşturmuştur. Bu soru havuzundaki bazı soruları derste çözmeye, bazılarını da sınavda sormaya karar vermiştir. Cahit Öğretmen'in öğrencilere çözümlenerek göstereceği örnek sorular yapay öğrenme sürecinde eğitim verisi (training data) olarak adlandırılmaktadır. Eğitim verisi bir tablo olarak (sütunlar ve satırlar) düşünüldüğünde, sütunlardan bazıları girdileri (bağımsız değişken) bir tanesi de çıktıyı (bağımlı değişken) temsil etmektedir. Veri setindeki satırlar ise örnekleri temsil etmektedir. Veri setleri, her zaman ham haliyle kullanılamaz. İnsan hataları veya teknolojik arızalar nedeniyle veri setleri içinde tutarsız ve imkânsız veri bulunabilir. Bazı özelliklere ait eksik/kayıp değer/veri (missing value) bulunabilir bununla birlikte veri setindeki tüm sütunların/özelliklerin kullanılması yerine bazı özelliklerin seçilmesi gerekebilir. Bazı özellikler üzerinde veri dönüşümü yapılmalıdır. Veri setinin eğitim ve test seti olarak bölünmesi de bu aşamada gerçekleştirilir. Tüm bu işlemler yapay öğrenmenin ikinci adımında ele alınır.

2.1.3. Adım: Uygun algoritma seçilerek model eğitilir

Fen Bilimleri dersindeki deneylerden hatırlanacağı gibi bağımsız değişkenlerin bağımlı değişken üzerinde etkisi olduğu düşünülmektedir. Yapay öğrenmede de girdilerin çıktıları etkilediği varsayılar ve girdi-çıkıtı arasındaki ilişkiyi bulmak hedeflenir. Cahit Öğretmen'in sınıfta verdiği örneklerdeki sayılar girdileri/bağımsız değişkenleri ifade ederken işlemin sonucu da çıktıyı/bağımlı değişkeni ifade etmektedir. Yapay öğrenmede Cahit Öğretmen'in sınıfta yaptığı gibi veri setinde girdi-çıkıtı örnekleri verilerek uygun bir algoritma seçilip eğitilerek bir model oluşturulur. Model her yeni örnekte tıpkı bir öğrenci gibi kendini geliştirir ve girdi-çıkıtı arasındaki ilişkiyi daha iyi açıklayabilecek şekilde parametrelerin ağırlıklarını günceller. Her öğrencinin farklı öğrenme yöntemi, farklı soru çözme stilleri

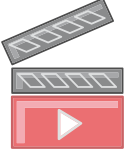
olduđu gibi yapay öğrenmede de çeşitli algoritmalar kullanılarak modeller geliştirilmektedir. Algoritmalar örnekteki öğrencilerin düşünme biçimi, öğrenme yöntemi veya işlem becerileri olarak düşünülebilir. Nasıl her öğrencinin her derste aynı derecede başarılı olması beklenmezse her algoritma da her problem durumunda, her veri setinde aynı derecede başarılı olmaz. Hatta problem durumu ve veri türü bazı algoritmaların kullanılmasına izin verirken bazılarının kullanılmasına izin vermez. Böyle bir durumda ikinci adıma dönülerek veri setinde bazı dönüştürme işlemleri yapılabilir.

2.1.4. Adım: Modelin değerlendirilmesi

Yapay öğrenmede oluşturulan modelin başarısını test etmek için Cahit Öğretmen'in öğrencilere yaptığı gibi bir sınav yapılır. Modelin başarısını test etmek için eğitim sırasında kullanılmayan verilerin (örneklerin) olduğu bir veri seti kullanılmalıdır. Bu durum Cahit Öğretmen'in derste çözdüğü örneklerin birebir aynısını sınavda sormasına benzer. Bu durumda öğrencilerin öğrenmesinden daha çok ezber yapmasında söz edilebilir. Bu tür durumlarda öğrencilerin veya yapay öğrenme modelinin başarısı sağlıklı bir şekilde değerlendirilmez. Modeli test etmek için kullanılan veri setine, test verisi (test data) denir. Test verisinde sonuç/çıktı sütunu verilmez. Bu durum Cahit Öğretmen'in doğru yanıtların işaretlenmiş olduğu sınav kağıtlarıyla sınav yapmasına benzer. Cahit Öğretmen sınavda derste çözdüğü örneklere hiç benzemeyen sorular sorarsa öğrencilerin başarıları düşebilir. Tüm bu durumlar yapay öğrenme sürecinde de geçerlidir. Öğrencilerin sınavda soruları çözerek yanıtları verdiği gibi yapay öğrenme modeli de test verisindeki girdileri kullanarak sonuçları bulmaya çalışır. Öğrencilerin değerlendirilmesiyle benzer şekilde modelin bulunduğu sonuçlarla doğru yanıtlar karşılaştırılarak modelin performansını gösteren başarı puanı hesaplanır. Yapay öğrenmede çıktı her zaman sayısal bir değişken olmaz. Her ne kadar sayısal hesaplamalar yapılsa da sonuç kısmında bu sayılar bir kategoriyle eşleştirilebilir. Verilen kan değerlerine göre bir hastalığa ilişkin tahmin üreten (pozitif/negatif) bir modelin çıktısı ikili değerden birini alır. Yapay öğrenmede sonuç başka bir ifadeyle bağımlı değişken sütunu etiket olarak da adlandırılır. Etiket bir kategorinin/sınıfın adı olabileceği gibi sayısal bir değer de olabilir.

2.1.5. Adım: Uygulama ve yaygınlaştırma

Değerlendirme aşamasından sonra geliştirilen model, kullanım amacına göre bir bilgisayar programı olarak veya gömülü bir yazılım olarak paketlenir. Akıllı cihazlardaki uygulamalar uygulama ve yaygınlaştırma adımıyla örnek olarak verilebilir. Bu aşamada geliştirilen model bir arayüz tasarlanarak hitap ettiği kişi ve kurumların kullanımına hazır hale getirilir.



Yapay Zekâ ile Okyanusları Temizliyorum

Yapay öğrenme modellerinin nasıl çalıştığını anlamak için code.org platformundaki “Okyanuslar için yapay zekâ” etkinliği yapılabilir. Bu etkinlikte yapay zekâya sahip bir robot, okyanusları temizleme görevinde kullanılmak üzere eğitilmektedir. Robota, balıklardan ve çöplerden oluşan örnekler gösterilir. Her bir örneğin ait olduğu sınıf öğretilerek eğitim işlemi gerçekleştirilmektedir. Etkinliğe karekodu okutarak veya bağlantıya tıklayarak ulaşılabilir[1].



Etkinlikten sonra aşağıdaki soruları yanıtlayınız.

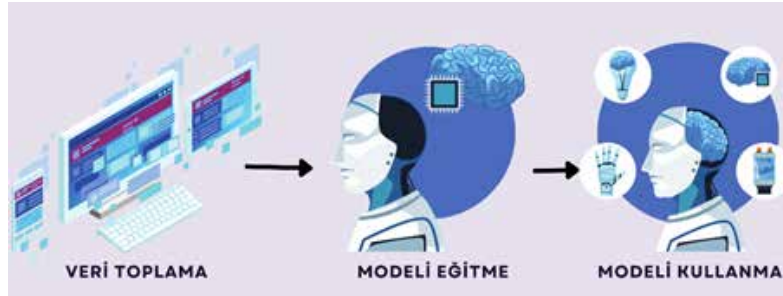
1. Yeteri kadar balık örneği gösterilmediğinde robotun okyanusu temizleme görevindeki başarısı değişiyor mu?
2. Örnekler yanlış olarak etiketlendiğinde (balıklardan bazıları çöp veya çöplerden bazıalarını balık olarak etiketlendiğinde) neler oluyor?

2.2. Google TM ile Yapay Öğrenme Modelleri Oluşturma

Teachable Machines (TM) Google tarafından hızlı, kolay ve erişilebilir yapay öğrenme modelleri oluşturmayı sağlayan web tabanlı bir araçtır. TM eğitilebilir makineler olarak ifade edilebilir. Google TM, herhangi bir yapay öğrenme bilgisi gerekmeden yapay öğrenme modelleri oluşturmayı sağlar. Başta eğitimciler ve öğrenciler olmak üzere konuya ilgi duyan herkesin kolaylıkla kullanabileceği bir araçtır [1].

2.2.1. TM ile Model Oluşturma Süreci

Google TM’de görselleri, sesleri ve pozları tanımak için kod yazmadan yapay zekâ modelleri oluşturulabilir. Oluşturulan modeller projelerde, web sitelerinde ve uygulamalarda kullanılabilir. TM ile model oluşturma süreci doğal olarak yapay öğrenme modeli süreci ile çok benzerdir. TM’de model oluşturma işlemi veri toplama, modeli eğitime ve modeli kullanma olmak üzere üç aşamada gerçekleşir (Görsel 2.2) [2].



Görsel 2.2: Google TM kullanımı

2.2.2. Veri Toplama

Bu bölümde kategoriler (sınıf) belirlenerek bu kategorilere ait örnekler toplanır. Sınıf kavramı, oluşturulacak modelin tahmin edeceği kategorilere verilen addır. Örneğin, kedi ve köpek görsellerini ayırt edecek bir yapay zekâ modeli oluşturulduğunda bu modelde iki adet sınıf olur: 1.Kedi, 2.Köpek. Kedi sınıfına ait örnekler kedi görsellerinden oluşurken köpek sınıfına ait örnekler de köpek görsellerinden oluşur. Yapılan projenin türüne göre örnekler görüntü, ses veya poz olabilir.

2.2.3. Modeli Eğitme

TM kullanarak üç tür model oluşturulabilir:

- Görsellerle (web kamerasıyla veya görüntü dosyaları kullanarak)
- Seslerle (mikrofon kullanarak yapılan ses kayıtları)
- PoseNet pozlarla (bir görüntüden bilgisayarın insanın kollarını, bacaklarını ve duruşunu tanıması için)

Model eğitme sürecinde sınıflara ait örnekler ile bir yapay öğrenme (makine öğrenmesi) algoritması eğitilir. Arka planda yapay öğrenme türlerinden biri olan ve insan beynindeki sinir hücrelerini ve onların oluşturdukları sinir ağlarını benzetimi amaçlayan çok katmanlı bir yapay sinir ağı (derin öğrenme modeli) eğitilir. Model veri setinden örnekler ile onların ait oldukları kategoriler (sınıf) arasındaki bağıntıları öğrenir.

2.2.4. Modeli Kullanma ve Dışa Aktarma

Model eğitildikten sonra yeni örnekleri öğrendiği bu bağıntıları kullanarak sınıflara eşleştirir. Modelin örnekleri doğru bir şekilde sınıflandırma yapıp yapmadığını görmek için model test edilir. Oluşturulan model indirilebilir veya çevrim içi ortamda tutularak erişime açılabilir. Ayrıca model projelerde (web sitelerinde, uygulamalarda ve farklı alanlarda) kullanılmak üzere dışa aktarılabilir. Yapılan çalışmayı kaydetmenin birkaç yolu vardır.

- **Tüm Projeyi Google Drive'a Kaydetme:** Proje, kişisel Google Drive depolama alanına kaydedilebilir. Bu şekilde modellerdeki sınıfların her birindeki tüm örnekleri içeren .zip uzantılı bir dosya Drive alanına kaydedilir. Çalışmaya kalan yerden devam etmek için daha sonra TM üzerinden bu .zip dosyası tekrar açılabilir.
- **Örnekleri İndirme:** Her bir sınıftaki tüm örnekleri (class-sample) indirmek için kullanılır. Aynı verilerle çalışmaya devam etmek istenildiğinde tekrar yüklenebilir.
- **Modeli İndirme:** Model indirilir ve TM web sayfası kapatılır. İndirilen model daha sonra kullanılabilir. Hiçbir sunucuya hiçbir şey kaydedilmez.
- **Modeli Çevrim İçi Yayımlama (export model):** Model çevrim içi yayımlayarak URL aracılığıyla başka projelerde modelin kullanılması veya bağlantı üzerinden modele erişilmesi sağlanabilir.
- **Hiçbir Şey Kaydetmemek:** Sekme kapatılır ve kaydetme işlemi yapılmazsa yerel bilgisayara veya herhangi bir sunucuya hiçbir şey kaydedilmez.

2.3. Google TM Uygulamaları

Bu bölümde TM kullanılarak geliştirilen uygulamalar yer almaktadır.

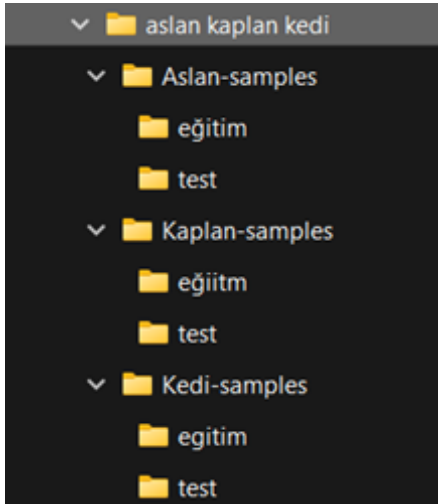


1. Uygulama: Bir Görüntü Modeli Eğitmek (Aslan, Kaplan ve Kedi)

Bu uygulamada Google TM kullanılarak aslan, kaplan ve kedi görüntülerini tanıyan bir yapay zekâ modeli eğitilmektedir.

Ön Hazırlık

Bu uygulamaya başlamadan önce her kategori için 60 adet görüntü bulunan bir veri setinin hazırlanması gerekmektedir. Üç farklı klasörde aslan, kaplan ve kedi görselleri hazırlanmalıdır. İyi bir model oluşturabilmek için görseller seçilirken kategorilere ait örneklerin fotoğraflarının da birbirlerinden belirgin şekilde farklı olduğu, net ve kaliteli fotoğraflar tercih edilmelidir. Kendi veri setini oluşturmak isteyen kullanıcılar **Fatkun Batch Download** gibi **Chrome** eklentilerinden yararlanarak hızlı bir şekilde web üzerinden görselleri indirebilirler. Bu eklenti arama sonucu ekrana gelen görsellerin toplu bir şekilde indirilmesine olanak verir.

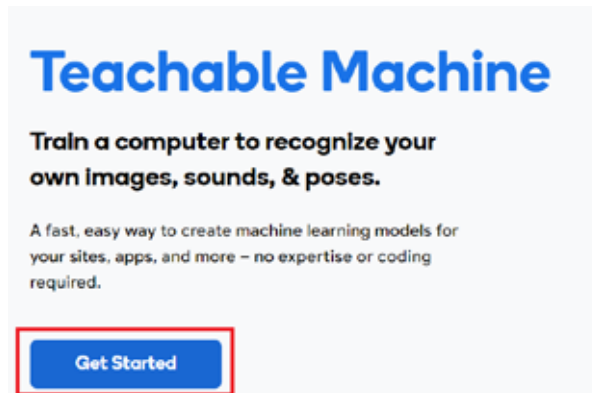


Görsel 2.3: Klasör yapısı

Her klasörde 2 adet alt klasör (eğitim ve test) bulunmalıdır (Görsel 2.3). Eğitim klasöründe 48 adet örnek (görsel) test klasöründe ise 12 adet örnek bulunmalıdır. Örnekte kullanılan veri setine ulaşmak için bölüm başında verilen karekod okutulabilir.

Uygulama Adımları

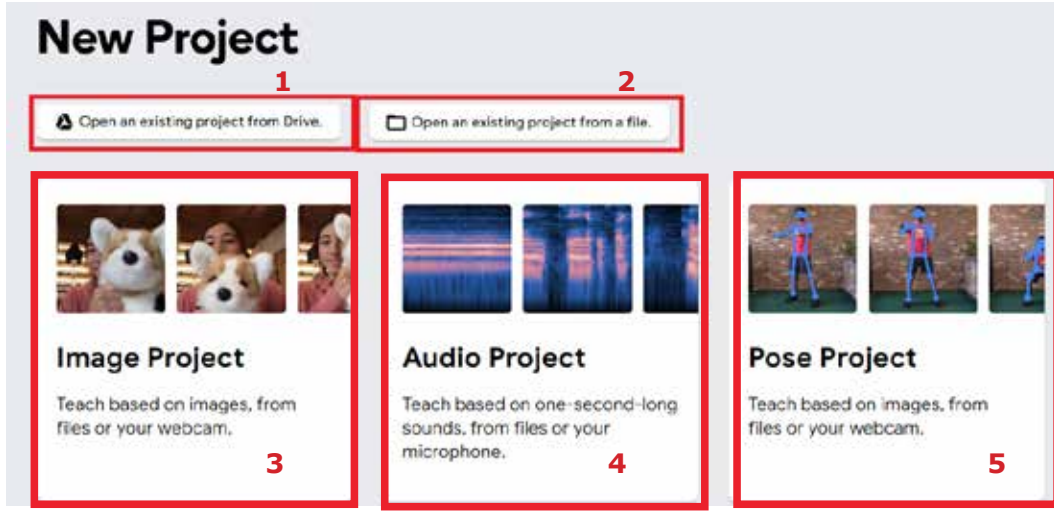
1. Yeni proje oluşturma: Web sayfası açılır ve get started (başlayın) butonuna tıklanır (Görsel 2.4). TM hesap açmadan veya giriş yapmadan doğrudan kullanılabilir [2].



Görsel 2.4: Google TM ana sayfa görüntüsü

Bir görüntü tanıma modeli eğitileceği için **Image Project** üstüne (Görsel 2.5'te 3 numaralı alan) tıklanır.

Bir ses projesi yapılmak istenirse **Audio Project** (Görsel 2.5'te 4 numaralı alan), bir poz projesi yapılmak istenirse **Pose Project** (Görsel 2.5'te 5 numaralı alan) seçilir.



Görsel 2.5: Google TM yeni proje sayfası

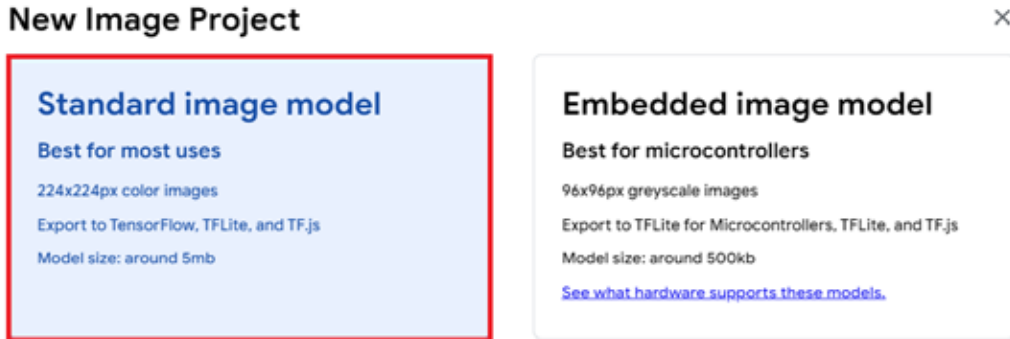
Yeni bir proje oluşturulabileceği gibi önceden yapılan projeler de kullanılabilir (Görsel 2.5'te 1 ve 2 numaralı alan).

Open an existing project from drive: Google Drive üzerinde depolanan bir projeyi aç.

Open an existing project from a file : Yerel bilgisayarda depolanan bir projeyi aç.

Açılan pencerede **Standard image model** kutucuğu tıklanır.

Embedded image model (gömülü resim modeli) mikrodenetleyiciler için daha hafif modeller oluşturulmasına olanak verir.

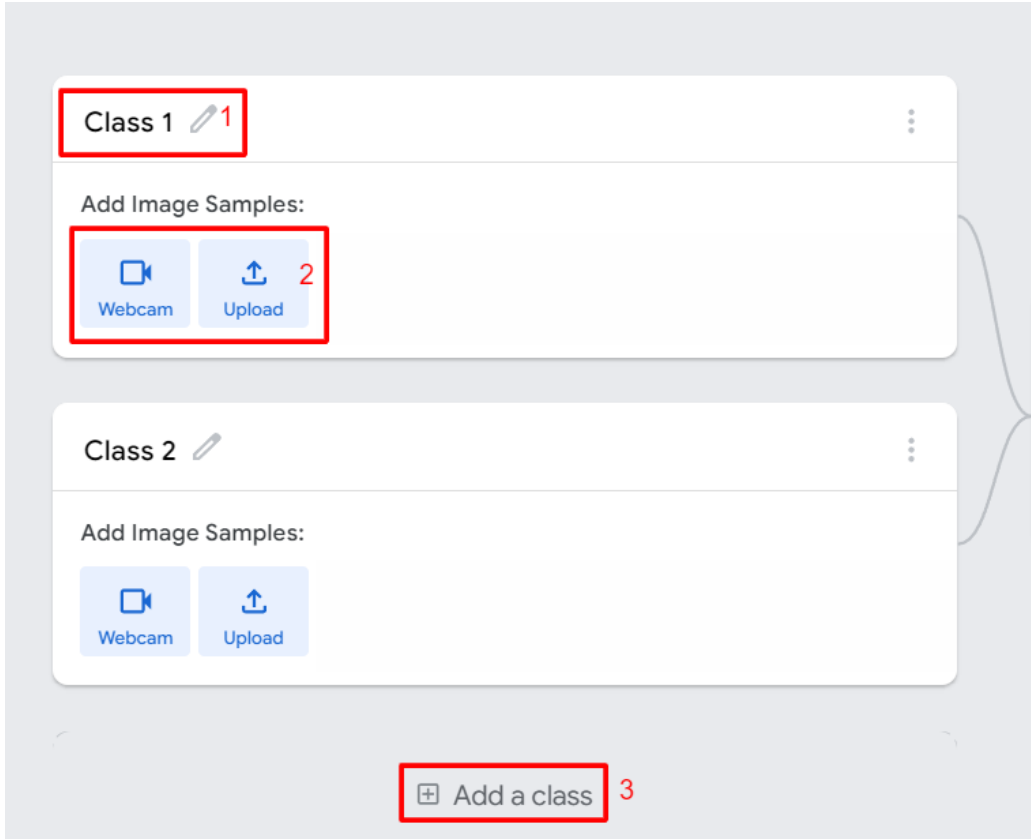


Görsel 2.6: Google TM yeni görüntü projesi

2. Veri toplama

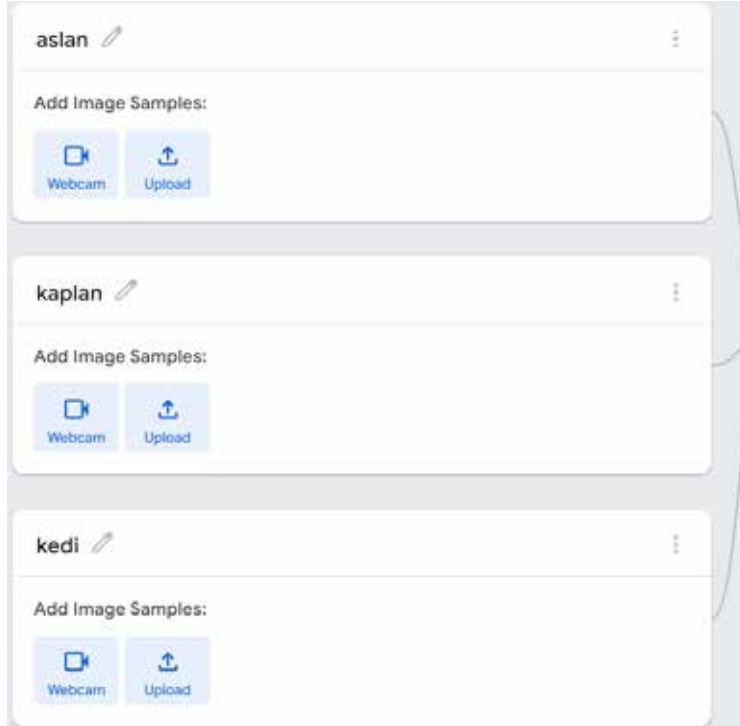
Bu aşamada model için gerekli sınıflar oluşturulur ve sınıflara ait örnekler (görüntüler) yüklenir. Varsayılan olarak iki sınıf gelmektedir. Bir sınıflandırma tahmini yapılacağı için en az iki kategori olmalıdır.

Bu çalışmada 3 kategori olduğu için (1.aslan, 2.kaplan ve 3.kedi) Add a class (bir sınıf ekle) butonu kullanılarak bir sınıf daha eklenir. Sınıf isimleri varsayılan olarak Class 1 ve Class 2'dir (Görsel 2.7'de 1 numaralı alan). Sınıf adları yanlarında bulunan kalem ikonuna ya da doğrudan isimlerin üzerine tıklanarak değiştirilir.



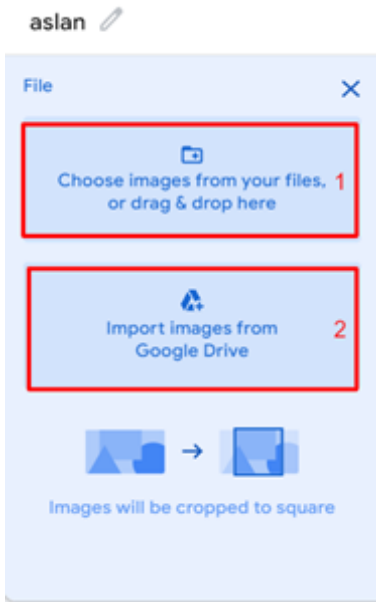
Görsel 2.7: Sınıfların oluşturulması

Değişikliklerden sonra ekranda aslan, kaplan ve kedi adında üç sınıf görülmelidir (Görsel 2.8).



Görsel 2.8: Sınıflar oluşturulduktan sonra ekran görüntüsü

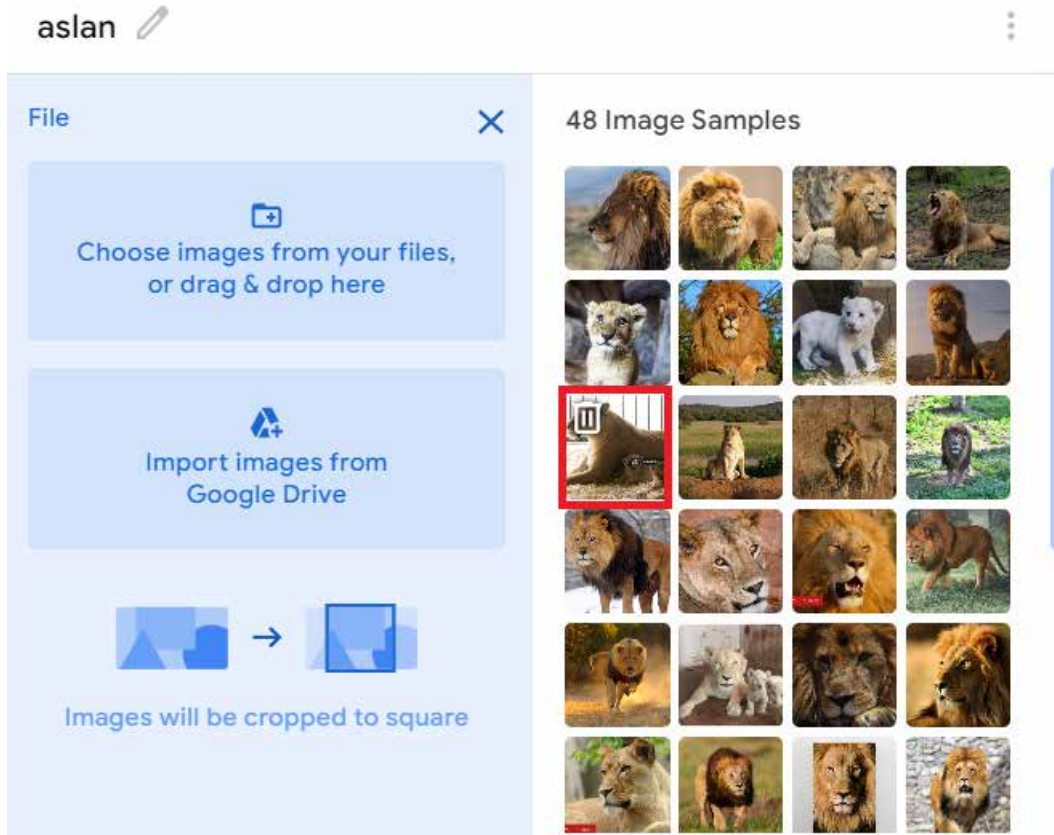
Sınıflar oluşturulduktan sonra her sınıf için **webcam** (web kamerası) ve **upload** (yükle) ikonları kullanılarak örnekler yüklenir (Görsel 2.7'de 2 numaralı alan).



Görsel 2.9: Sınıflara örnek yükleme

Cihazın kamerası kullanılmak istenirse **webcam**, bir dosya yüklenmek istenirse **upload** seçilmelidir. Bu uygulamada önceden indirilmiş görüntü dosyalarından oluşan örnekler kullanılmaktadır. **Upload** ikonuna tıklanarak açılan pencerede **choose images your files, or drag & drop here** (Görsel 2.9'da 1 numaralı alan) butonuna tıklanarak açılan dosya yolu penceresinden dosyalar çoklu olarak seçilerek veya bilgisayarda kayıtlı olduğu yerde dosyalar seçilip tarayıcı üzerinde bu bölüme sürüklenerek yüklenir. Dosyalar **import images from Google Drive** bölümü (Görsel 2.9'da 2 numaralı alan) tıklanarak Google Drive'dan da yüklenebilir.

Bu aşamada her bir sınıf için eğitim alt klasöründe yer alan 48 örnek (görsel) yüklenmelidir. Görseller otomatik 224x224 **pixel** boyutunda bir kare hâline getirilir. Yükleme işlemi yapıldıktan sonra ekranda her sınıfa ait görseller, sınıfın yanında görünür (Görsel 2.10). Sınıflara ait örnekler dikkatli bir şekilde incelenmelidir. Yanlış örnekler modelin performansını olumsuz etkiler. Aslan görsellerinin gerçekten tüm aslanları (yavru, yaşlı, dişi, erkek ve beyaz renkli gibi) temsil etmesi araya başka görsellerin karışmaması modelin performansını artırır. Herhangi bir nedenle silinmesi gereken bir örnek olduğunda fare ile görselin üzerine gelinir ve çöp kutusu ikonuna tıklanır (Görsel 2.10).



Görsel 2.10: Aslan sınıfı örnekleri

Örnekler ve sınıflarla ilgili bazı işlemleri yapmak için örneklerin üstündeki üç noktalı ikona tıklanır (Görsel 2.11).

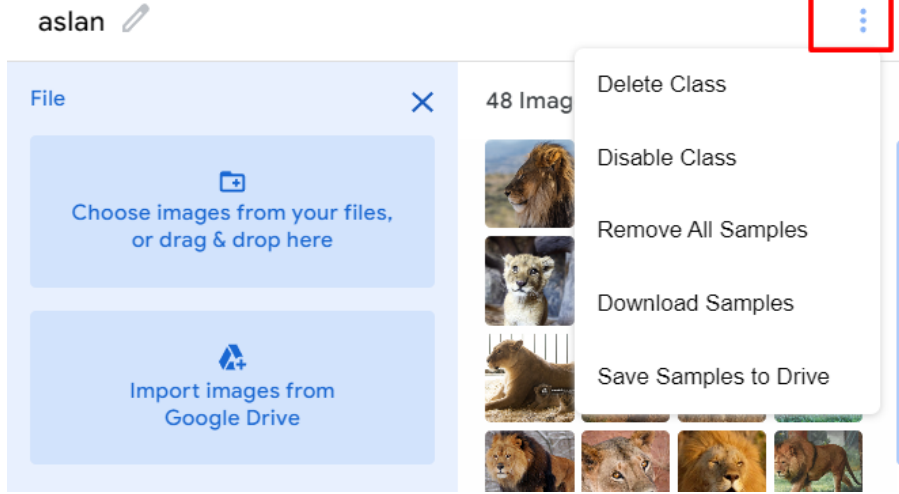
delete class: Bir sınıfı tamamen siler.

disable class : Sınıfı pasif hâle getirir.

remove all samples: Sınıfa ait örneklerin tamamını siler.

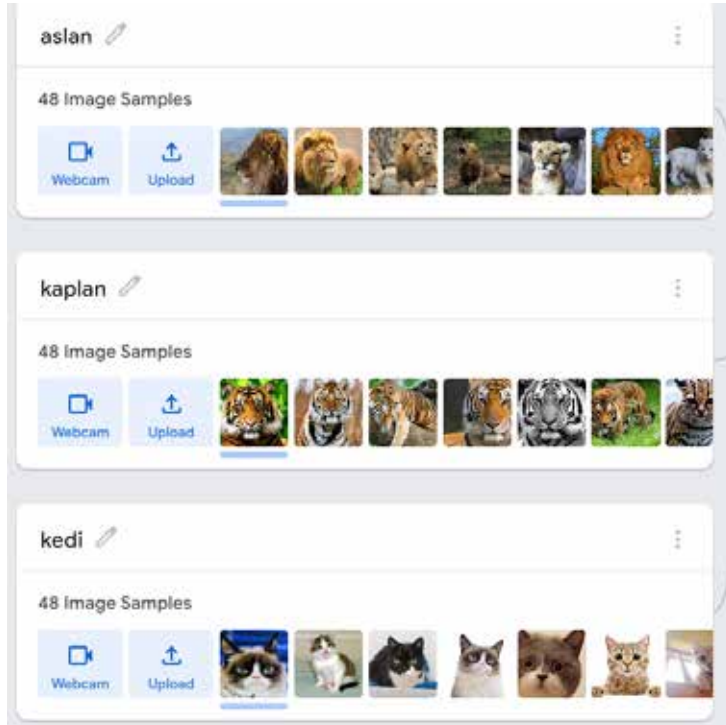
download samples: Sınıfa ait örnekleri indirir.

save samples to drive: Google Drive'a kaydeder.



Görsel 2.11: Sınıfla ilgili işlemler

Tüm sınıflar için örnek yükleme işlemleri tamamlandığında model eğitim için hazır hâle gelir (Görsel 2.12).



Görsel 2.12: Sınıfların hazırlanması

3. Modeli Eđitmek

Modeli eđitmek iin **train model** (modeli eđit) butonuna tıklanır. Model eđitiminin suresi bilgisayar zelliklerine, tarayıcıya ve internet hızına gore deđişiklikler gosterebilir. Eđitim bařladıktan sonra **train model** butonu etiketi **training** olarak gornr (Grsel 2.13).

Bir modeli eđitmek iin yeteri kadar rnek kullanılmazsa, bařka bir ifadeyle az sayıda rnekle eđitilirse bu durumda zayıf đrenme (underfitting) gerekleřir.

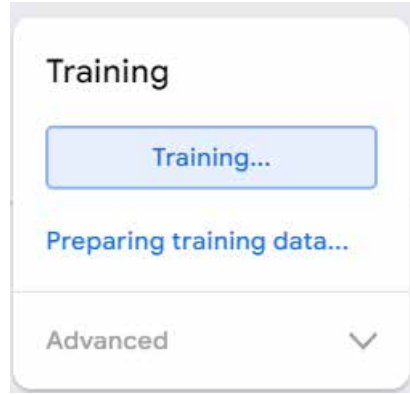
Model eđitiminde **advanced** (geliřmiř) (Grsel 2.13) blm arka planda eđitilen ok katmanlı bir sinir ađına iliřkin ileri dzvey ayarların yapıldıđı blmdr.

Epochs (dng sayısı): Modeli eđitmek iin veri setinin ka defa kullanılacađını gsteren parametredir.

Batch size (para boyutu): Veri setindeki ađa rneklerin kaar kaar verileceđini gsteren parametredir.

Learning rate (đrenme oranı): Ađdaki đrenme oranını gncellemek iin kullanılan parametredir.

Bu ayarlar modelin eđitim suresini ve performansını dođrudan etkileyebilir.



Grsel 2.13: Modelin eđitim suresi

4. Modeli Kullanmak ve Dışa Aktarmak

Model eğitildikten sonra **training** (eğitiliyor) etiketi yerine **model trained** (model eğitildi) etiketi görünür. Model eğitildiği verilerle test edilirse

modelin performansı doğru bir şekilde değerlendirilemez. Bu durum öğretmenin sınıfta çözdüğü soruların aynısını sınavda sormasına benzetilebilir. Bu durum ezberleme veya aşırı öğrenme (overfitting) olarak adlandırılır. Bu nedenle modeli test etmek için test klasöründeki örnekler kullanılır. Sınıfların test klasöründeki örnekler modele sunulur ve modelin tahminlerdeki performansı kontrol edilir.

Modeli test etmek için **input** (girdi) anahtarı (Görsel 2.14'te 1 numaralı alan) **ON** (açık) konumunda olmalıdır.

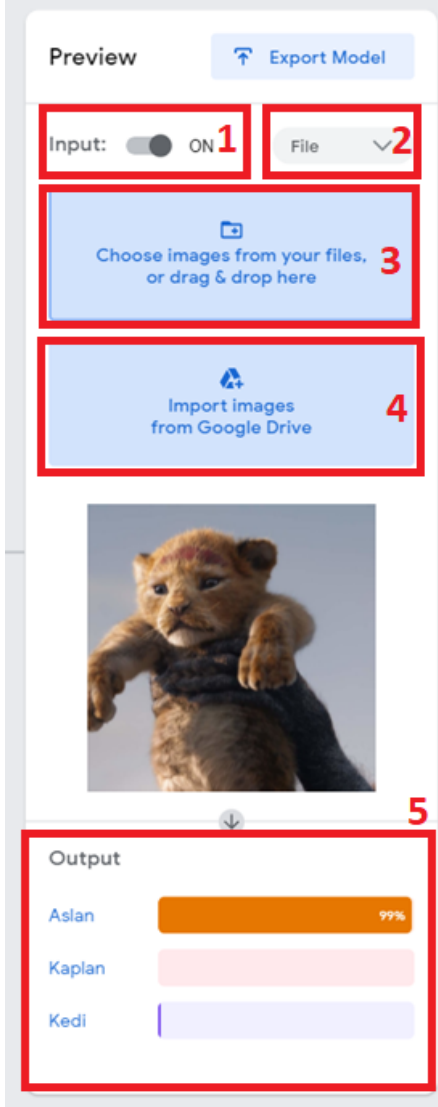
Input (Girdi): Eğitilen modelin sınıflandırma yapması için örneğin girilmesini sağlar. Resim sınıflandırma çalışmalarında modele webcam veya bir resim dosyası yükleyerek sınıflandırma yapılması istenebilir.

Açılır listede (Görsel 2.14'te 2 numaralı alan) **Webcam**'e tıklanarak File seçilir. Modeli test etmek için dosyanın yükleneceği konuma uygun bölüm tıklanır veya dosya bu bölüme sürüklenir (Görsel 2.14'te 3 veya 4 numaralı alan).

Choose images your files, or drag & drop here seçilerek bilgisayarda herhangi bir sınıfa

ait test örneklerinden biri (görsel) modele yüklenir.

Output (Çıktı) bölümünde model, örneğin ait olduğu sınıfa ilişkin bir tahmin yapar. Bu tahmin bir olasılık yüzdesi olarak verilir. Verilen örnekteki görsel %99 olasılıkla aslan ve %1 olasılıkla ise kedi olarak tahmin edilmiştir (Görsel 2.14'te 5 numaralı alan).



Görsel 2.14: Modelin test edilmesi

Modelin Değerlendirilmesi

Bu tür bir sınıflandırma modelinin basit bir şekilde değerlendirilmesi için bir yöntem vardır. Hata veya karmaşıklık matrisi adı verilen yöntemde, her bir sınıf için eldeki 12 örnek modelde test edilir.

Örneğin model aslan test örneklerinden kaç tanesini aslan, kaç tanesini kaplan ve kaç tanesini kedi olarak tahmin etti?

Bu sorular tüm sınıflar için çaprazlanarak sorulur. Eğer oluşturulan model aslan test görsellerinden 10 tanesini aslan ve 2'sini de kedi olarak tahmin ederse bu durum hata matrisinde (Tablo 2.1'deki) gibi işlenir. Bu tüm sınıflar için yapılır. Tablodaki mavi alanlar doğru tahminleri gösterir.

Tablo 2.1: Hata Matrisi

Gerçek Sınıflar	Tahminler		
	Aslan	Kaplan	Kedi
Aslan	10	0	2
Kaplan	0	11	1
Kedi	2	1	9

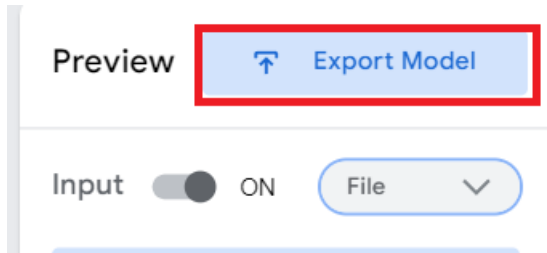
Hata matrisi incelendiğinde aslan sınıfı için modelin doğruluk yüzdesi doğru tahmin sayısı/tahmin sayısı formülünden $(10/12)$ %83 olarak hesaplanır.

Kaplan sınıfı için bu oran $(11/12)$ %91,66 olarak hesaplanır.

Kedi sınıfı için bu oran $(9/12)$ %75 olarak hesaplanır.

Modelin genel doğruluk yüzdesi ise tüm sınıflardaki doğru tahmin sayısının toplamı/toplam tahmin sayısı formülü kullanılarak $(30/36)$ %83,33 olarak hesaplanır.

Not: Sınıflandırma modellerinin performansının değerlendirilmesi için yalnızca bu metrikler yeterli değildir. Modellerin ayrıntılı olarak değerlendirilmesi için kullanılan başka metrikler de bulunmaktadır. Çalışmanın türüne, amacına uygun şekilde kullanılacak metrikler araştırılmalıdır.

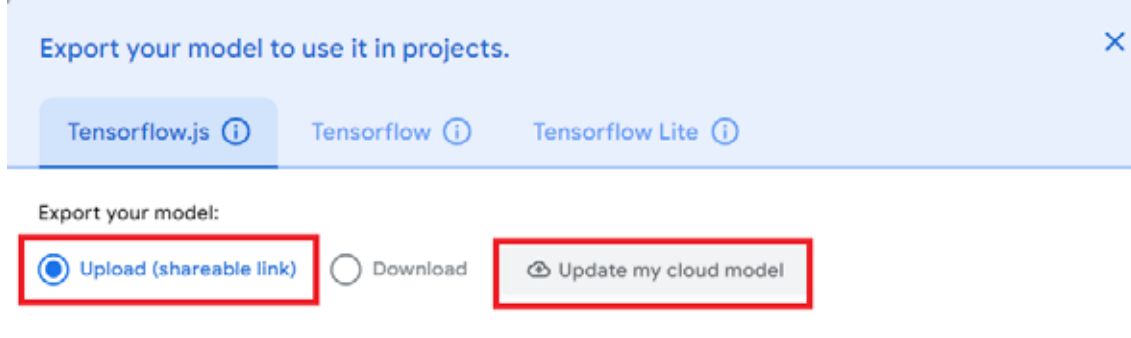


Görsel 2.15: Modeli dışa aktarma

Modeli Dışa Aktarmak

Bu uygulamada model çevrim içi olarak yayımlanarak internet üzerinden erişilmesi sağlanacaktır. Bunun için **preview** (ön izleme) bölümünde **export model** (modeli dışa aktar) tıklanır (Görsel 2.15).

Açılan pencerede modeli dışa aktarmak için farklı seçenekler bulunmaktadır. Tensorflow.js sekmesinde Upload (shareable link -paylaşılabilir link) varsayılan olarak seçili gelir. Bir değişiklik yapılmadan **Update my cloud model** (bulut modelimi güncelle) butonu tıklanır (Görsel 2.16).



Görsel 2.16: Modeli yayımlama

Bu işlemten sonra your shareable link başlığı altında modelin çevrim içi bağlantısı oluşur. Copy ikonu kullanılarak bağlantı kopyalanır. Bu bağlantı kullanılarak model farklı uygulamalarda çalıştırılabilir. Bağlantı açıldığında bir web sayfası olarak model çevrim içi kullanılabilir. Karekod okutularak veya bağlantıya tıklanarak bu uygulamada oluşturulan model test edilebilir [3]. Export model bölümünde model yüklendikten sonra paylaşılabilir. Bağlantı oluşturulduktan sonra sayfanın altında javascript kodları güncellenmektedir. Bu kodlar bir web sayfasına gömülerek modelin herhangi bir web sayfası içinden de kullanılması sağlanabilir. Yine bu bölümde p5.js sekmesi tıklanarak p5.js çevrim içi web editörü kullanılarak html ve javascript kodları değiştirilerek çevrim içi olarak çalıştırılabilir. Modelin yayımlandığı bölümde (Görsel 2.14) Download (indir) seçeneği tıklandığında download my model (modelimi indir) butonu görünür. Bu buton tıklanarak oluşturulan model bilgisayara kaydedilebilir.

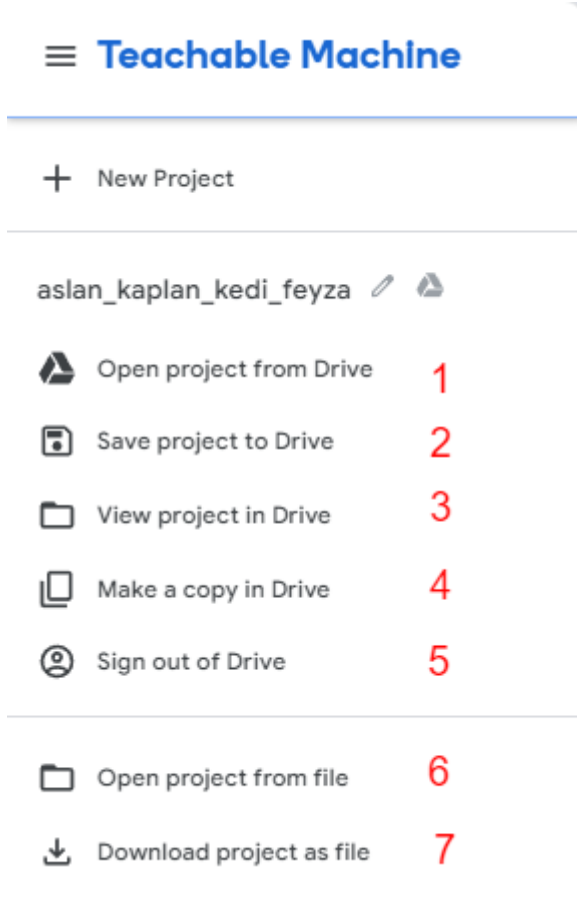
Sorular

1. Modele bir köpek görseli verildiğinde nasıl bir tahmin yapmaktadır?
2. Modeli eğitirken aslan sınıfına ait örneklerin arasında kedi görselleri de ekleyerek eğitildiğinde modelin tahmin performansı nasıldır?
3. Modeli eğitirken verilen örnek sayısının artırılması veya azaltılması modelin performansını nasıl etkiler.



2.3.1. TM Menü İşlemleri

TM sayfasının sol üst köşesinde bulunan menü (Görsel 2.17) kullanılarak bazı işlemler yapılabilmektedir. Menü kullanılarak aşağıda belirtilen işlemler yapılabilir.



Görsel 2.17: TM menüsü

çık): Drive oturumundan çıkmak için kullanılır.

6. Open project from file (Dosyadan proje aç): Yerel cihazda kayıtlı projeyi açmak için kullanılır.

7. Download Project as file (Projeyi dosya olarak indir): Projeyi dosya olarak yerel cihaza indirmek için kullanılır.

1. Open project from Drive (Drive'dan proje aç): TM projelerin Google Drive depolama alanına kaydedilmesine olanak sağlamaktadır. Bu seçenek kullanılarak Google Drive hesabındaki depolanan bir proje açılabilir.

Bu işlem ve Drive ile yapılacak diğer işlemler için öncelikle TM ile Google Drive arasındaki bağlantının yapılması ve izinlerin verilmesi gereklidir.

2. Save project to Drive (Drive'a proje kaydet): Projenin Drive'a kaydedilmesini sağlar.

3. View project in Drive (Projeyi Drive'da görüntüle): Drive üzerinden açılan projeyi konumunda görüntülemek için kullanılır.

4. Make a copy in Drive (Drive'da bir kopya oluştur): Drive'da projenin bir kopyasını oluşturmak için kullanılır.

5. Sing out of Drive (Drive oturumundan



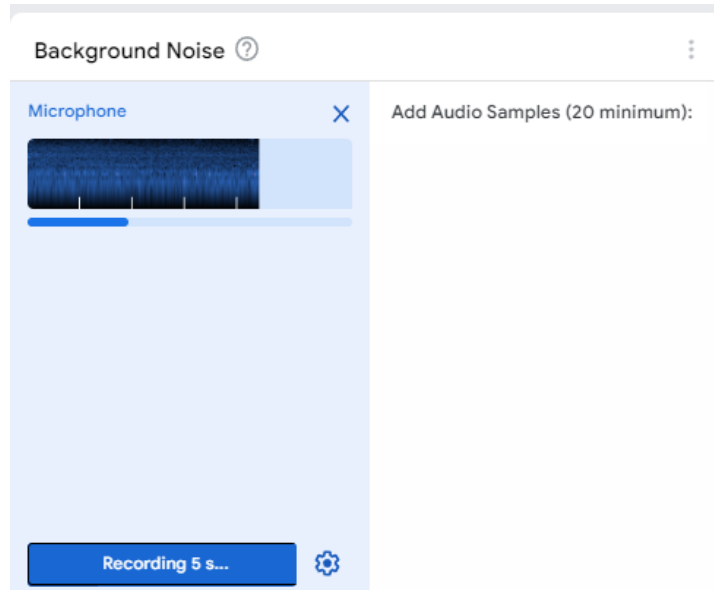
2.Uygulama: Bir ses modeli oluşturma (Sayıları tanıma)

Bu uygulamada TM üzerinden sesle söylenen sayıları tanıyan bir model eğitilmektedir.

NOT: Birinci uygulamadaki benzer adımlar tekrara düşmemek için ayrıntılı olarak verilmemiştir.

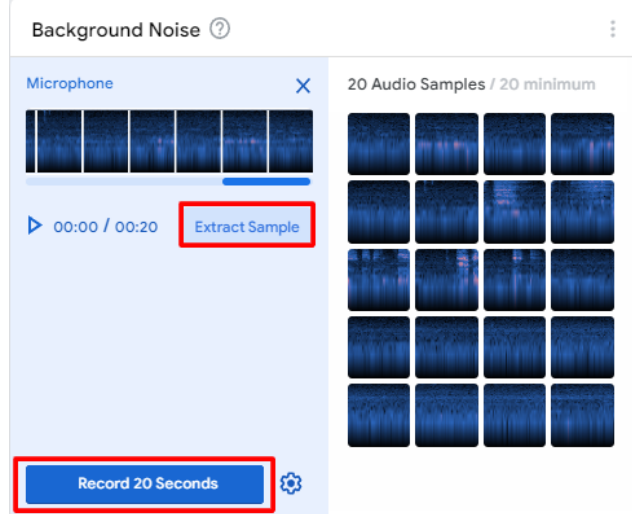
Uygulama Adımları

- 1. Yeni Proje Oluşturma:** Web sayfası açılır. Başlayın butonuna tıklanır ve yeni proje olarak audio project (ses projesi) seçilir.
- 2. Veri Toplama:** Bu projede modele 1, 2 ve 3 sayılarının okunuşu öğretileceğinden projeye 3 sınıf eklenmiştir. Ses projelerinde mikrofon ile kayıt yapma veya ses dosyası yükleme seçenekleri bulunmaktadır. Bu uygulamada toplam dört sınıf tanımlanmıştır. **Background noise** (arka plan gürültüsü) arka plandaki gürültünün diğer sınıflardaki örneklerle karışmaması için varsayılan olarak gelen bir sınıftır. Bu amaçla ortamın 20 saniyelik bir ses kaydı alınır. **Background noise** sınıfındaki mikrofon ikonu tıklanır ve açılan pencerede **Record to 20 seconds** (20 saniye kaydet) butonu tıklanır. Daha sonra sınıf için ses kaydı yapılmaya başlayacaktır. Buton etiketi **Recording** (kaydediliyor) olarak değişir (Görsel 2.18).



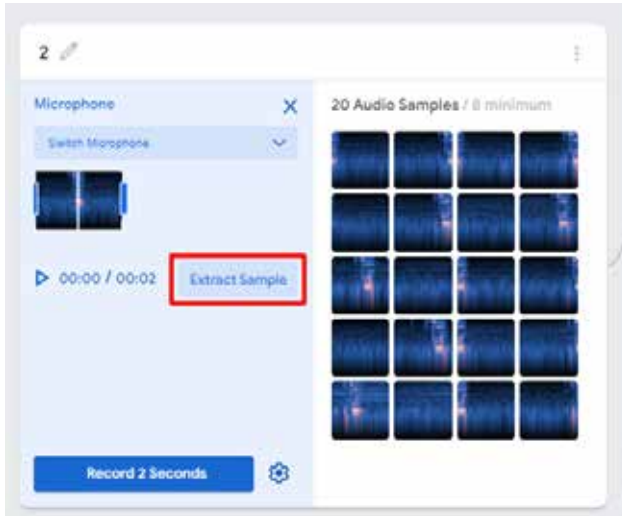
Görsel 2.18: Ses kaydetme

Kayıt 20 saniyeye ulaştığında ekranda Record to 20 seconds butonu ile üst bölümde kaydedilen ses de görünür. Extract sample (örnekleri çıkar) butonuna tıklanarak (Görsel 2.19) ses sınıfa eklenir. Bu işlem yapılan ses kaydını 20 parçaya bölerek 20 örnek hâline getirir. İstenirse arka plan sesi sınıfı, pasif hâle getirilebilir veya silinebilir. Ancak bu durumun modelin çalışmasını etkileyeceği unutulmamalıdır.



Görsel 2.19: Arka plan ses örnekleri

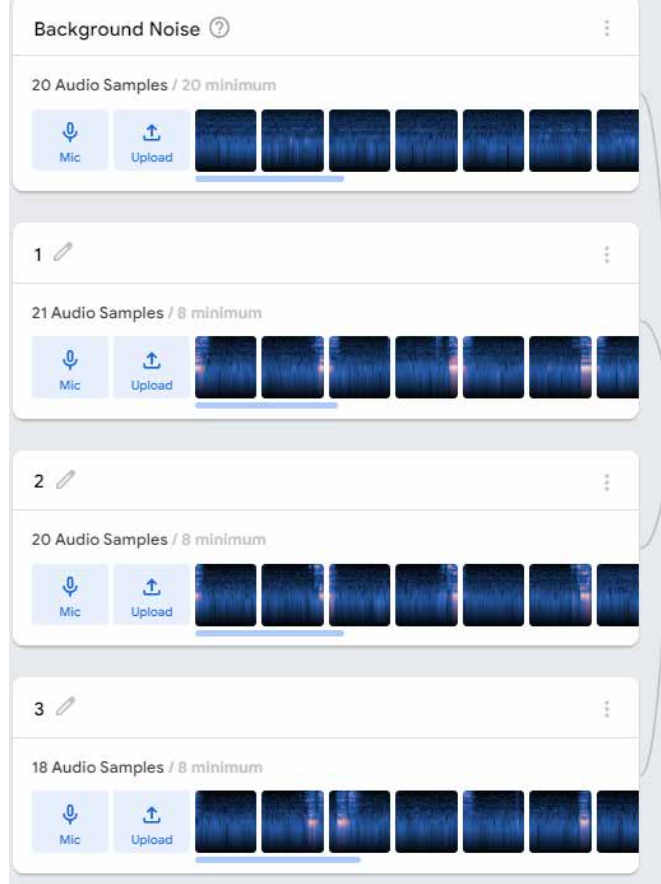
Bu aşamada görüntü projesinde olduğu gibi yeni sınıflar eklenebilir. Sınıf adları değiştirilebilir. Sınıflara ait örnekler tek tek veya topluca silinebilir.



Görsel 2.20: Sınıflara örnek eklemek

Oluşturulan tüm sınıflar için benzer şekilde örnekler eklenir. Toplam 4 sınıf içinde ortalama 20 örnek oluşturulur. Her sınıf için mikrofon ikonuna tıklanarak 2 saniyelik ses kayıtları yapılır ve her örnek için extract sample butonuna tıklanarak örneğin sınıfa atanması sağlanır (Görsel 2.20). Bu işlem her sınıf için en az 8 örnek olana kadar tekrarlanmalıdır. Örnek sayısının fazlalığı modelin çalışmasını olumlu etkilemekle birlikte, eğitim süresini uzatır

Tüm sınıflara örnekler eklendikten sonra model eğitim için hazır hâle gelir (Görsel 2.21).

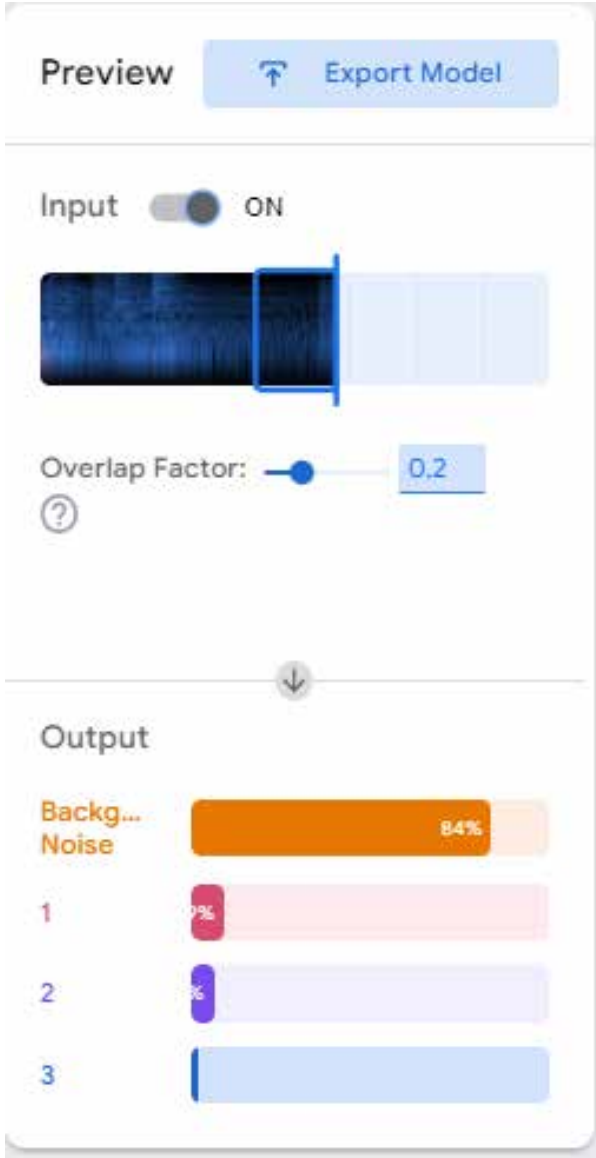


Görsel 2.21: Sınıfların ve örneklerin oluşturulması

3. Modeli Eğitmek: Modeli eğitmek için train model (modeli eğit) butonuna tıklanır. Model eğitiminin süresi bilgisayar özelliklerine, tarayıcıya ve internet hızına göre değişiklikler gösterebilir. Eğitim başladıktan sonra train model butonu etiketi training olarak değişir. Model eğitiminin advanced bölümünde görüntü modellerinden farklı olarak yalnızca epoch parametresi bulunur.

Epochs (döngü sayısı): Modeli eğitmek için veri setinin kaç defa kullanılacağını gösteren parametredir. Epoch değerinin artması modelin daha iyi eğitilmesini sağlar. Böylece modelin doğruluk oranı artar. Başka bir ifadeyle model daha doğru kestirimler yapar. Ancak çok fazla eğitilen modelde ise aşırı öğrenme (overfitting) olabilir.

4. Modeli Kullanmak ve Dışa Aktarmak: Model eğitildikten sonra training butonu etiketi model trained olarak değişir. Bu aşamada mikrofondan ses girişi olur ve sesler sınıflandırılır. Modeli eğitirken olduğu gibi bir, iki veya üç sayılarından biri sesli olarak söylenerek model test edilir. Eğer kullanıcı tarafından ses girişi gelmezse arka plan seslerinin arka plan gürültüsü olarak sınıflandırılması beklenir.



Görsel 2.22: Modelin test edilmesi

Sorular

1. Rakamları başka biri söylediğinde (modeli eğitirken rakamları söyleyen kişi dışında) model doğru tahminler yapabiliyor mu?
2. Arka plan gürültüsü sınıfı için kayıt oluşturulmadığı durumlarda model nasıl çalışmaktadır?
3. Model için veri toplanırken 20 farklı ses kaydı almak yerine, tek bir ses kaydı alınıp extract sample ile çoğaltılırsa modelin performansı nasıl etkilenir?

Görüntü işleme projelerinde olduğu gibi ses projelerinde de model yayımlanabilir veya yerel cihaza kaydedilebilir.

Karekod okutularak veya bağlantıya tıklanarak bu uygulamada oluşturulan model test edilebilir [4].

Modelin değerlendirilmesi aşamasında hata matrisi kullanılabilir. Sistemin sürekli ses girişi alması, modelin değerlendirilmesini güçleştirir. Bu aşamada uygulayıcılar geliştirdikleri modelleri değerlendirmek için hata matrisini (Uygulama 1'de gösterildiği gibi) kullanabilir.





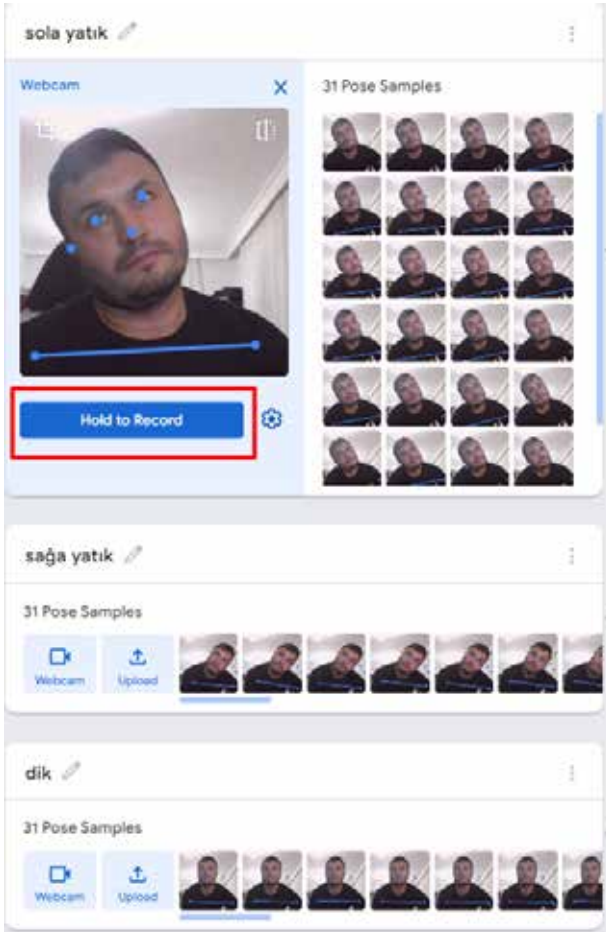
3. Uygulama:

Bu uygulamada TM üzerinde insanların pozisyonlarını tanıyan bir model eğitilmektedir.

NOT: Önceki uygulamalarda benzer adımlar tekrara düşmemek için ayrıntılı olarak verilmemiştir. Poz projesi görüntü projesiyle çok benzemektedir.

Uygulama Adımları

- 1. Yeni Proje Oluşturma:** Web sayfası açılır. Başlayın (get started) butonuna tıklanır ve yeni proje olarak Pose Project (poz projesi) seçilir.
- 2. Veri Toplama:** Bu projede modele bir insanın duruşunun (kafasını ne tarafa eğdiği) sağa eğik, sola eğik veya dik olduğu öğretileceğinden projede üç sınıf oluşturulur. Sınıflar için görüntü girişi webcam ikonu kullanılarak yapılır. Web kamerası kullanılarak sınıflara uygun pozlar verilerek sınıflar için örnekler oluşturulur. Sınıflar için web kamerasından görüntü alınırken hold to record (sürekli kayıt için basılı tutun) butonuna sürekli olarak basılı tutularak hızlı bir şekilde kameradan görüntü alması sağlanabilir (Görsel 2.23).



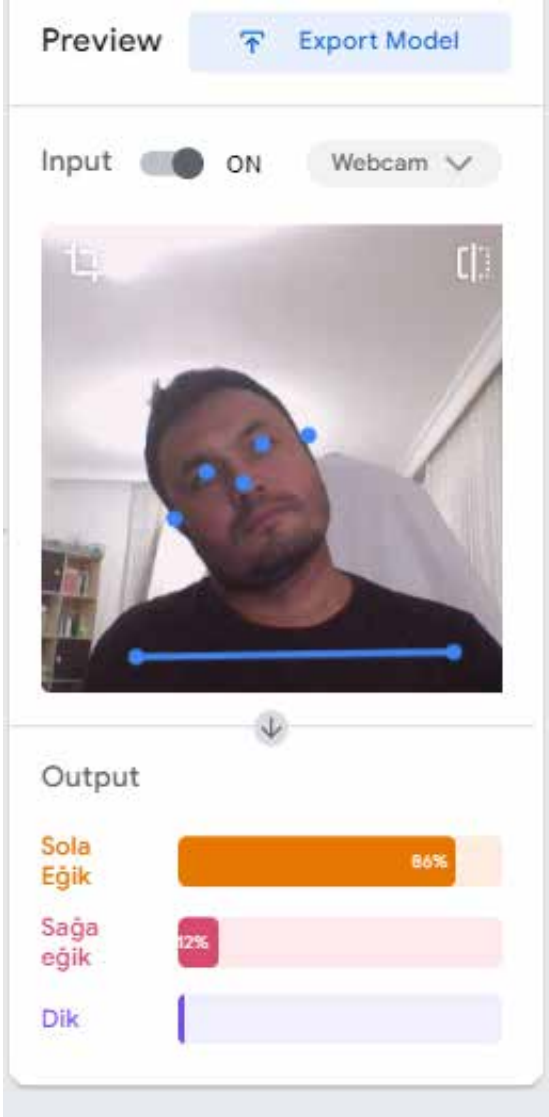
Bu aşamada, görüntü projesinde olduğu gibi yeni sınıflar eklenebilir. Sınıf adları değiştirilebilir. Sınıflara ait örnekler tek tek veya tamamı silinebilir.

3. Modeli Eğitmek: Modeli eğitmek için train model butonuna tıklanır. Model eğitiminin süresi bilgisayar özelliklerine, tarayıcıya ve internet hızına göre farklılık gösterebilir. Eğitim başladıktan sonra train model butonu etiketi training şeklinde görünür. Model eğitiminde advanced bölümünde görüntü modelleri aynı ayarlar yer almaktadır.

Görsel 2.23: Sınıflar ve örnekler



4. Modeli Kullanmak ve Dışa Aktarmak: Model eğitildikten sonra training bölümünde model trained etiketi görünür. Bu aşamada Webcam sürekli olarak görüntü almakta ve sınıflandırma yapmaktadır. Modeli test etmek için kullanıcı sağa, sola eğilerek veya dik durarak pozlar verir (Görsel 2.24)



Görsel 2.24: Modelin test edilmesi

Görüntü işleme projelerinde olduğu gibi poz projelerinde de model yayımlanabilir veya yerel cihaza kaydedilebilir.

Karekod okutularak veya bağlantıya tıklanarak bu uygulamada oluşturulan model test edilebilir [5].

Görüntü işleme modelinde olduğu gibi modeli değerlendirmek için hata matrisi kullanılabilir.

Sorular

1. Sağ elin havada olduğu bir model eğitildiğinde, test aşamasında sol el kaldırılırsa model o sınıfı hala doğru tanır mı?
2. Google TM'yi belirli nesnelere tanıyacak şekilde eğitip bilgisayara nesnenin bir fotoğrafı veya o nesnenin çizimi verildiğinde sınıflandırma nasıl olur?



2.3.2. Model Performansının İyileştirilmesi

Model test sırasında yeteri kadar başarılı bulunmadıysa başka bir ifadeyle modelin doğru tahmin oranı yeterli görülmediği durumlar olabilir. Bir modelin başarısını değerlendirmek için genellikle kesin oranlar veya sayılar yoktur. Bir modelin başarısı modelin oluşturulma amacına ne kadar hizmet ettiğine bağlıdır.

Örneğin on binde bir görülen bir hastalık için bir model geliştirme çalışmasında otuz bin kişilik bir veri setinde 3 hasta bulunmaktadır. Geliştirilen model herkesi sağlıklı olarak tahmin ederse modelin genel başarı oranı %99,99'dur. Ancak gerçekte bu model hiçbir hastayı doğru tahmin edememiştir. Bu nedenle tek bir metrik veya orandan söz edilemez. Modelin özelliğine uygun bir değerlendirme yapılmalıdır. TM'de geliştirilen bir modelin performansını artırmak için aşağıdaki işlemler denenebilir.

Arka plan/ışık ayarları: Bir görüntü projesi üzerinde çalışılıyorsa örneklerin farklı bir arka plan veya farklı bir aydınlatma durumundaki görüntülerinin kullanılması veya farklı örneklerle modelin eğitilmesi denenebilir.

Örneklerin çerçevesi: PoseNet yalnızca pozların nasıl görüldüğünü (kolların yukarıda veya aşağıda olması gibi) izlemekle kalmaz. Ayrıca kişinin çerçevede nerede görüldüğünü de izler. Çerçevenin sol tarafındaki bir duruş çerçevenin sağ tarafındaki bir duruştan farklı bir poz olarak algılanmaktadır.

Sesle ilgili değişiklikler: Ses tabanlı bir model üzerinde çalışırken farklı bir mikrofon kullanmak, mikrofona olan yakınlığı değiştirmek veya arka plandaki gürültüler modeli eğitirken kullanılan veri setini etkileyeceği için bu tür değişiklikler modelin başarısını etkiler.

Ses örneklerini yakalama: TM sadece 1 saniyelik örnekleri tanımak için üretilmiştir. Model için 1 saniyelik örneklerden oluşturulan ses dosyaları yüklenebilir.

Yanlılığı anlama: Yanlılık, önyargı (bias) yapay öğrenme modelleri oluştururken dikkat edilmesi gereken kritik bir kavramdır. Önyargı, yapay öğrenme için kullanılan veri seti içinde bazı örneklerin diğerlerine göre daha baskın olması anlamına gelmektedir. Bir ses modeli geliştirdikten sonra farklı kişilerin sesleriyle model test edilebilir. Eğer model çalışmıyorsa diğer kişilerden de ses örnekleri alınarak model tekrar eğitilmelidir.

2.3.3. TM ile İlgili Ayrıntılar

Bu bölümde Google TM ile ilgili ayrıntılara yer verilmiştir [2].

• Modeli TM Dışında Kullanmak

TM'de oluşturulan bir model bir Tensorflow.js modeli olarak dışa aktarılabilir ve ücretsiz olarak TM'de barındırılabilir. Böylece model, herhangi bir web sitesi veya uygulamada (bağlantı adresi verilerek) ve Tensorflow ve Tensorflow Lite biçimlerine dönüştürülerek yerel bilgisayarda kullanılabilir.

• Modeldeki Örneklere Erişim ve Gizlilik

Proje Google Drive'a kaydedilmediği sürece model eğitiminde kullanılan örnekler hiçbir sunucuya gönderilmez. Google Drive'a kaydedildiğinde bile örnek veriler kişisel depolama alanında sahibine aittir. Model eğitilirken herhangi bir sunucuya veri göndermeden tarayıcı sekmesinde çalışır.

• Modeli Yayımlama

Modeli yükle seçildiğinde model, Google sunucularında yayımlanır. Ancak modeli eğitmek için kullanılan örnekler yayımlanmaz. Arka planda, modele örnek verildiğinde hangi sınıfa ait olduğunu tahmin eden matematiksel program çalıştırılır.

• Modeli Erişime Açmak

Modeli çevrim içi yayımlama seçilirse TM modelin ücretsiz olarak barındırıldığı bir URL oluşturur. Bu bağlantı paylaşarak model başkalarının da kullanımına açılabilir. Bu bağlantıya sahip olanlar modeli projelerinde kullanabilir. Modelin URL'sini çağıran bir web sitesi yapılırsa biri sayfanın koduna bakarak URL'yi bulabilir ve kopyalayarak projelerinde kullanabilir. Tıpkı çevrim içi olarak sunulan herhangi bir resim veya video dosyası gibi kullanılabilir. Ancak bu kullanıcılar modeli değiştiremezler.

• Web'den Erişilen Modellerin Güvenirliği

Yapay öğrenme modelleri içerik oluşturucu tarafından toplanan örneklerle (Örnek: görseller, sesler, pozlar) eğitilir. Modelin çalışma şekli eğitildikleri verilere bağlıdır.

• TM Arka Planı

TM web tarayıcıda yapılan modelleri eğitmek ve çalıştırmak için Javascript üzerinde bir yapay öğrenme kütüphanesi olan Tensorflow.js kullanılır. TM kütüphanesi Tensorflow.js üzerine inşa edilmiştir ve kodları Github platformunda bulunmaktadır.

TM platformunda oluşturulan modeller transfer learning (öğrenme aktarımı) adı verilen bir teknik kullanır. Önceden eğitilmiş bir yapay öğrenme modeli (çok katmanlı sinir ağı var) ve TM aracılığıyla oluşturulan sınıflar, bu sinir ağının son katmanı veya adımı hâline gelir. Özellikle hem görüntü hem de poz modelleri önceden eğitilmiş mobil ağ modellerinden öğrenir. Ses modeli ise konuşma komutları üzerine inşa edilmektedir.



Yapay Öğrenme ve İnsan Ön Yargısı

Gözlerinizi kapatın ve bir ayakkabı hayal edin. Hayal ettiğiniz ayakkabı neye benziyor? Bir kovboy ayakkabısı, bir spor ayakkabı belki de uzun topuklu şık bir kadın ayakkabısı. Ayakkabı denilince her bireyin kafasında farklı bir ayakkabı imgesi oluşmaktadır. Şimdi ayakkabı resmini bilgisayara tanıması için öğretmeye çalıştığınızı düşünün. Ön yargınız bilgisayara vereceğiniz örnekleri etkileyebilir.

Yapay zekâda ön yargı nasıl oluşur?

Öncelikle yapay zekâ nasıl öğrenir? Bugün, birçok teknolojide yapay zekâ modelleri kullanılmaktadır. Yapay zekâ, yol tarifleriyle bir yerden bir yere gitmeye yardımcı olur. Bize öneriler sunar, çeviri yapmamızı sağlar, söylediklerinizi anlar (ses tanıma). Peki nasıl çalışır? Geleneksel programlamada insanlar bir problemin çözümündeki kodları adım adım yazar. Yapay zekâ tekniklerinde ise bilgisayarlar verideki desenleri, modelleri bularak çözümü öğrenir. Yapay zekâda insan ön yargısı olmadığı düşünülebilir. Ancak bir olgunun yalnızca veriye dayanması onu tarafsız yapmaz. İnsanların kendi ön yargılarını bir kenara bırakmaları çok zordur. İnsanların bu ön yargıları teknolojinin bir parçası hâline gelir. Bunu değişik yollarla yaparız. Bir ayakkabı hayal edilmesi istendiğinde çoğu insan bir erkek spor ayakkabısı hayal eder. Daha fazla insan bilgisayar için erkek spor ayakkabısı çizdiğinde bilgisayar klasik kadın ayakkabılarını tanımayacaktır. Ön yargıya başka bir örnek olarak bir bilgisayarı eğittiğinizi düşünelim. Bir fizikçinin nasıl görüldüğünü, eski fizikçilerin resimlerini kullanarak bilgisayara öğretmeye çalıştığınızı düşünün. Algoritma erkeklerin lehine gizli bir ön yargıya neden olacaktır. Yüz tanıyan bir sistem eğittiğinizi düşünün. Görüntüleri internetten veya kendi fotoğraflarınızdan aldığınızda, algoritmanız herkesin yüzünü tanıyabilir mi? Algoritmanın herkesin yüzünü tanıması için tüm insanları temsil eden fotoğrafları seçtiğinizden emin olabilir misiniz? [6]

2.4. Blok Tabanlı Yapay Zekâ Kodlama Araçları

Özellikle eğitimciler için yapay zekâ teknolojilerini tanımak, bu teknolojileri projelerde kullanmak, öğrencilere bu teknolojilerle ilgili uygulama yapma olanağı sunmak önemli bir ihtiyaçtır. Yapay zekâ teknolojileriyle ilgilenmeye yeni başlayanlar ve öğrenciler için blok tabanlı uygulamalar kolaylık sağlamaktadır. Piyasada çoğu ücretsiz çok sayıda blok tabanlı kodlama aracının yanı sıra web tarayıcısı üzerinde çalışabilen pek çok yapay zekâ modeli bulunmaktadır.

Blok tabanlı kodlama araçları son yıllarda bünyelerine yapay zekâ teknolojilerini de entegre ederek gelişmiş uygulamalar yapılmasına olanak sağlamıştır.

Piyasada popüler olarak bulunan blok tabanlı kodlama araçlarının hemen hepsi Scratch tabanlıdır. Scratch, MIT Üniversitesi tarafından geliştirilmiş, ücretsiz ve açık kaynak kodlu olarak kullanıma sunulmuştur. Popüler blok tabanlı kodlama araçları da Scratch çekirdeğini kullanmaktadır. Özellikleri ve kullanımları arasında küçük farklılıklar olmakla birlikte bu araçlar birbirine oldukça benzemektedir. Scratch başta olmak üzere piyasada mBlock, Cognimates, Kittenblock, Tinkergen ve PictoBlox gibi araçlar bulunmaktadır.

PictoBlox programı diğer uygulamalarda olan özelliklerin yanında yapay zekâ eklentilerinin zenginliği ve akıllı cihazlara da yüklenebilmesi ile ön plana çıkmaktadır. mBlock programı aygıt desteği (Arduino, MicroBit vb.) ve aygıtlar için sunduğu kütüphaneler nedeniyle mikrodenetleyicilerle yapılacak projeler için tercih edilmektedir. Bu nedenlerle bu bölümde geliştirilecek olan uygulamalar için PictoBlox ve mBlock programları tercih edilmiştir.

2.5. PictoBlox ile Yapay Zekâ Uygulamaları

PictoBlox, Scratch 3.0 tabanlı bir blok kodlama yazılımıdır. Sürükle bırak özelliğine sahip kod blokları ile kolay bir şekilde kodlama yapılmasına olanak verir. PictoBlox, Scratch'ın temel işlevleri yanında donanım, robotik ve yapay zekâ gibi eklentilere sahiptir. PictoBlox uygulaması, temelde kendi ürettiği robot kitleri üzerinde gelişmiş özellikler sunmakla birlikte Micro:Bit, Arduino ve Esp32 gibi mikrodenetleyici kontrol kartlarıyla kodlama yapma olanağı sağlamaktadır. PictoBlox, akıllı cihazlara uygulama olarak yüklenebilmektedir. Bu durum projelerde esneklik ve kullanım kolaylığı sağlamaktadır. Yazılım kişisel ve ticari olmayan kullanımlar için ücretsizdir. Ancak yapay zekâ eklentisi kullanacak olanların sisteme üye olarak giriş yapması gerekmektedir. Uygulama yeni kullanıcılara kredi vererek onların bulut tabanlı yapay zekâ servislerini kullanmalarını sağlamaktadır. Kredi bittiği zaman yeni bir kullanıcı girişi yapılabilir ya da kredi satın alınabilir.



PictoBlox

Çevrim içi çalışmak için: <https://PictoBlox.ai/>

Bilgisayar ve akıllı cihazlar için indirme sayfası

<https://thestempedia.com/product/PictoBlox/download-PictoBlox/>

Kaynaklar ve projeler: <https://thestempedia.com/project/>



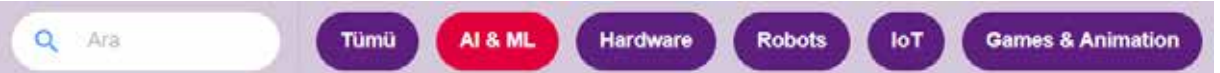
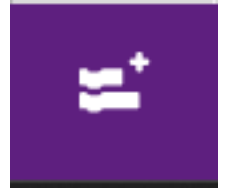
2.5.1. PictoBlox Programındaki Yapay Zekâ Eklentileri ve Uygulamaları

PictoBlox programı çevrim içi olarak [6] Windows, macOS, Linux işletim sistemli bilgisayarda veya Android/IOS yüklü akıllı cihazlara kurularak kullanılabilir.



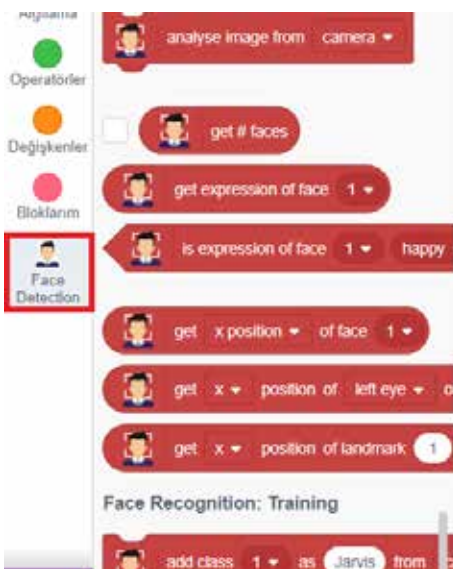
Bu bölümdeki uygulamaları geliştirebilmek için temel seviyede kodlama bilgisine sahip olmak ve blok tabanlı kodlama araçlarıyla çalışmış olmak gerekmektedir.

PictoBlox'da yapay zekâ uygulamaları gerçekleştirebilmek için bazı eklentiler gereklidir. Eklenti yükleme işlemi diğer blok tabanlı uygulamalarda olduğu eklenti ikonu kullanılarak yapılır. Uygulama penceresinin sol alt tarafında bulunan eklenti ikonuna tıkladığında eklenti yükleme penceresi ekrana gelir. Açılan pencerenin üst tarafında bulunan Tümü seçeneği yerine AI&ML butonuna tıklanarak artificial intelligence ve machine learning (yapay zekâ ve makine öğrenmesi) gibi yapay zekâ temelli eklentiler filtrelenir. Diğer eklentilerle birlikte kullanılacak IoT (nesnelerin interneti) eklentileri de bu bölümde yer almaktadır (Görsel 2.25).



Görsel 2.25: PictoBlox eklentiler

Kullanılmak istenen eklenti üzerine tıkladığında bloklar kısmına eklenir ve kullanıma hazır hâle gelir. Bu aşamada modelin yüklenmesi internet bağlantı hızına ve bilgisayarın özelliklerine göre farklılık gösterebilir (Görsel 2.26).



Görsel 2.25: Face Detection eklentisi

PictoBlox da bulunan yapay zekâ eklentileri ve kullanım amaçları şöyledir.

2.5.1.1. Yüz Tanıma

Yüz tanıma basitçe bir kişinin yüzünün görüntüsünün sayısallaştırılarak yüzün temel özelliklerini belirleyen bölümlerine (kaş, gözler, burun, ağız ve çene hattı gibi), bunların konumlarına ve birbirlerine uzaklığına göre her insan için benzersiz bir model oluşturan bilgisayarlı görü teknolojisidir. Yüz tanıma ve eşleştirme işlemi dört adımda gerçekleşmektedir [7].

1.Adım: Yüzü algılama

Görüntüdeki yüzlerin tespit edildiği aşamadır. Bu aşamada doğrudan bir kameradan (canlı görüntüden) bir veya daha fazla yüz algılanır ve bulunur. Yüz algılama işlemi bir kameradan olabileceği gibi bir fotoğraf veya görüntü dosyası üzerinden de gerçekleştirilebilir.

2.Adım: Yüz analizi

Bu aşamada yüzün görüntüsü kaydedilip analiz edilir. Yüz tanıma teknolojilerinin çoğu iki boyutlu görüntüler kullanmaktadır. Bu durum daha çok görüntünün (görseller, resimler vb.) kullanılmasına olanak tanır. Çünkü üç boyutlu görsel sayısı iki boyutluya göre daha azdır. Yüz analizinde yazılım yüzün geometrisini çıkarır. Yüzdeki ana öğeler arası mesafeler (gözler arasındaki mesafe, göz çukurlarının derinliği, alından çeneye mesafe, elmacık kemiklerinin şekli, dudaklar ve çenenin hatları gibi) hesaplanır. Bu işlemin amacı yüzdeki belirgin öğelerin tanımlanmasıdır.

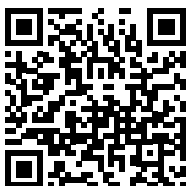
3.Adım: Görüntüyü veriye dönüştürme

Bu aşamada analog bilgiler dijital veri kümesine dönüştürülür. Yüz, sayılardan oluşan bir küme olarak tanımlanır. Her yüz "yüz izi" adı verilen benzersiz birer matematiksel modele dönüştürülür.

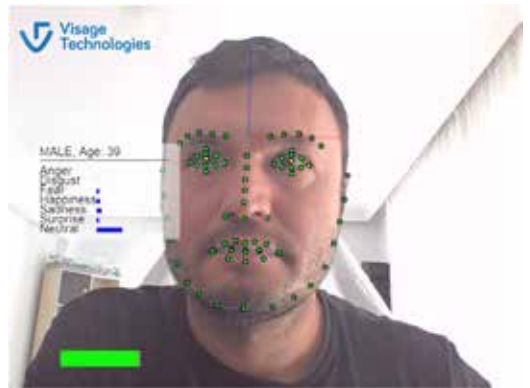
4.Adım: Eşleştirme

Oluşturulan yüz izi diğer yüz izleri ile (örneğin arananlar veri tabanındaki yüz izleri gibi) karşılaştırılır. Bir yüz izinin diğeriyle eşleşmesi durumunda eşleştirme işlemi tamamlanmış olur. Yüz tanıma iris veya parmak izi gibi diğer biyometrik ölçüm yöntemleri içinde en doğal ve kullanışlı olanıdır.

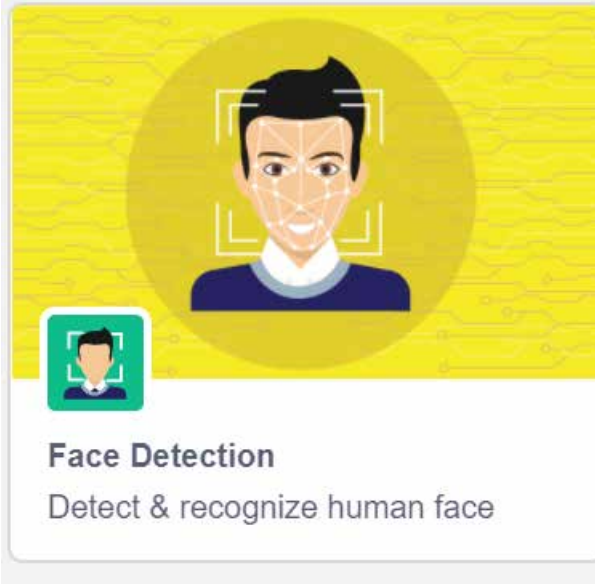
Yüz tanıma sistemleri akıllı telefonların ekran kilitlerini açmak, kayıp kişilerin/suçluların aranması, bulunması, sınır kontrolleri, havaalanı denetimleri, eğitimde öğrencilerin okula devamlarının takibi, internet/mobil bankacılık işlemlerine giriş, işyerlerinde giriş izni, denetim, sürücü tanıma gibi işlemlerde ve daha birçok farklı alanda kullanılmaktadır [7].



Visage Teknoloji Firması tarafından çevrim içi sunulan demo uygulamasında yaş tahmini, duygu durumu tahmini, yüzdeki hatların işaretlenmesi, yüz eşleştirme ve kaplan modelleme gibi özellikler yer almaktadır. (Görsel 2.27). Karekod okutularak veya tıklanarak demo test edilebilir [8].



Görsel 2.27: Yüz tanıma sistemi



Face detection eklentisi

PictoBlox'da face detection eklentisinde kodlar dört kategoride bulunmaktadır.

1.Settings (Ayarlar)

2.Detection (Algılama)

3. Face recognition: Training (Yüz tanıma: eğitim)

4. Face recognition: Test (Yüz tanıma: test)

Bu eklentiye ait kod bloklarının işlevleri kategoriler altında örnek kodlarla birlikte

verilmiştir. Öncelikle kategorideki kod blokları bütün olarak gösterilmekte daha sonra her kategorinin kodları tek tek açıklanmaktadır.

1.Face Detection/Settings

Settings



Eklentiyle ilgili ayarların yapıldığı bölümdür.



Kameradan sahneye görüntü aktarmak için kullanılır.

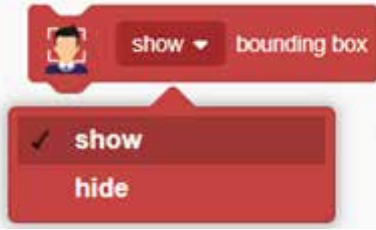
On : Kamerayı aç.

Off : Kamerayı kapat.

On flipped: Kameradaki görüntüyü ayna görüntüsünde açar.

Transparency: Kameradaki görüntünün sahne üzerindeki saydamlığını ayarlamak için kullanılır. %0 kameradaki görüntü tam görünürken dekor görünmez. %100 kameradaki görüntü görünmezken dekor tam görünür.

1.Face Detection/Settings



Yüz tanıma uygulamalarında algılanan yüzü belirtmek için kullanılan çerçevedir (sınır kutusu).

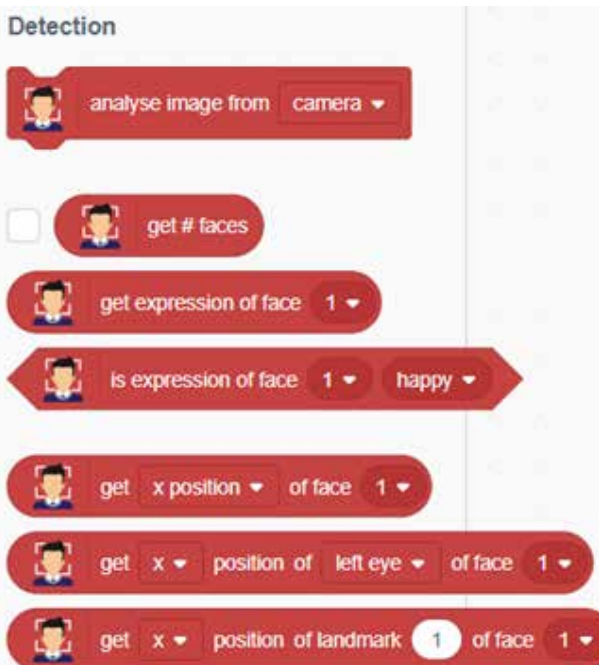
Show : Kutuyu göstermek için kullanılır.

Hide : Kutuyu gizlemek için kullanılır.



Yüz algılama eşiğinin belirlenmesi için kullanılır. Eşik yükseldikçe yüz algılama için daha yüksek oranda bir eşleşme gerektirir.

2.Face Detection/Settings

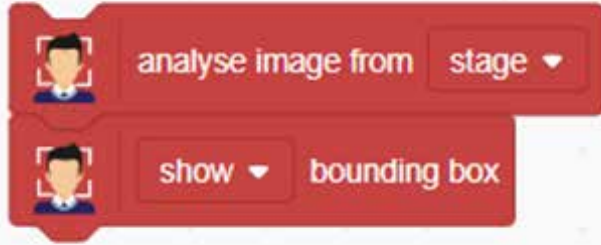


Kameradan veya sahnedeki görüntülerden yüzlerin analiz edildiği, duygu durumu analizi, gözlerin konumunun ve yüz hatlarının belirlenmesi gibi işlemlerin yapıldığı bölümdür.

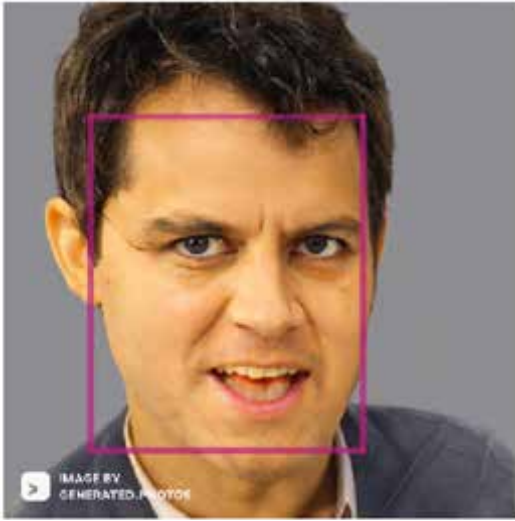
2.Face Detection/Settings

Sahnedeki yüzün çerçeve içine alınarak gösterilmesi.

Örnek Kod



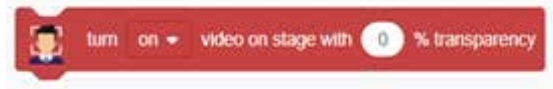
Çıktı (Görsel 2.28)



Görsel 2.28: Çıktı

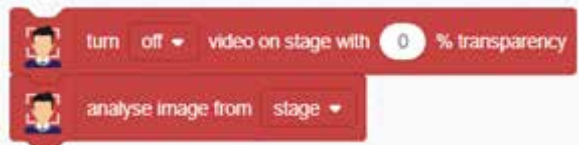


Kameradan veya sahneden alınan görüntünün analiz edilmesi için kullanılır. Kameradan görüntünün analiz edilebilmesi için bu blok öncesinde



bloğunun kullanılmış olması gerekir.

Sahneden (dekor vb.) görüntü almak için önce kamera kapatılmalıdır.



2.Face Detection/Settings

Sahnedeki görüntüde yer alan yüz sayısının bulunması.

Kod



Çıktı (Görsel 2.29)



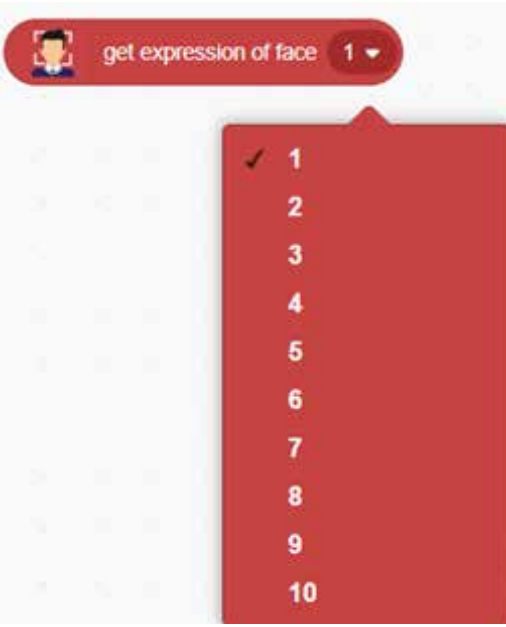
Görsel 2.29: Yüz tanıma

Bir sahne yerine kameradaki görüntüden yüz sayısının bulunması.

Örnek kod



2.Face Detection/Settings

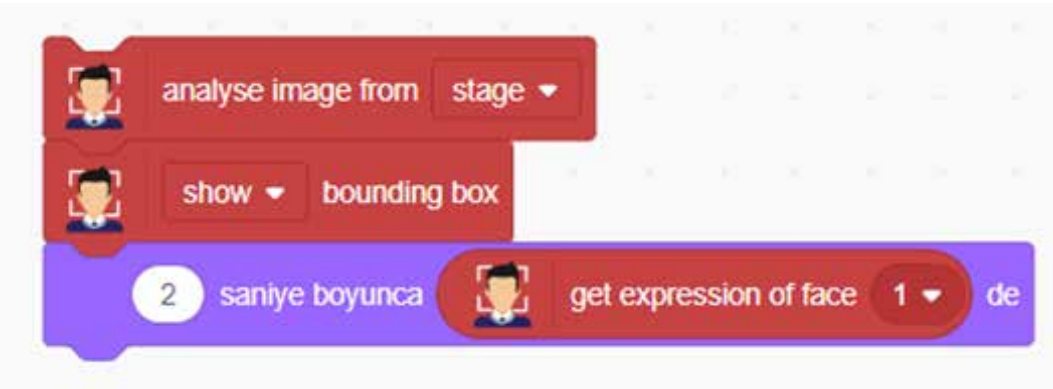


Bu blok algılanan yüzlerin (10 yüze kadar) duygu ifadelerini almak için kullanılır.

Sahnedeki bir yüze ait duygu durumu şu kodlarla alınabilir.

Sahnedeki görüntüden yüz ifadesinin tespit edilmesi.

Örnek Kod:



Çıktı (Görsel 2.30)

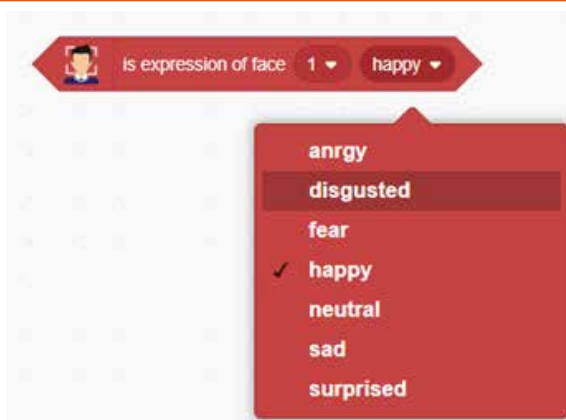
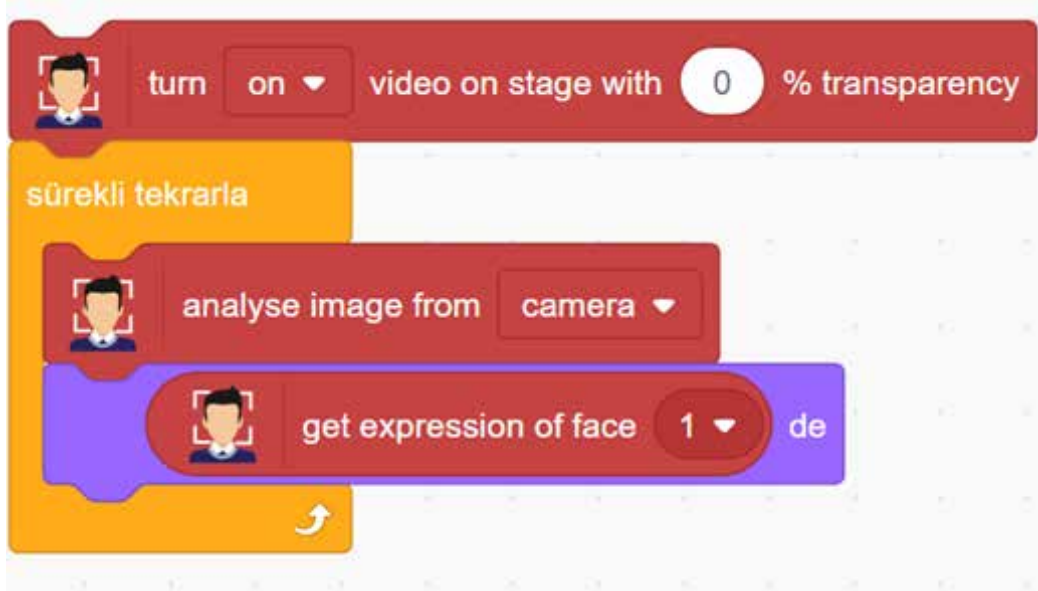


Görsel 2.30: Duygu durumu belirleme

2.Face Detection/Settings

Kameradaki görüntüden yüz ifadesinin tespit edilmesi.

Örnek kod



Bu blok belirtilen yüze ait (1. algılanan) duygu durumunu listeden (angry, disgusted ...) kontrol etmek için bir eğer bloğu ile kullanılır. Blok "algılanan 1. yüzdeki ifade happy (mutlu)" anlamına gelmektedir.

Duygu durumları

Angry: Sinirli

Disgusted: Bıkkın

Fear: Korkmuş

Happy: Mutlu

Neutral: Normal

Sad: Üzgün

Surprised: Şaşırmış

2.Face Detection/Settings

Belirli bir yüz ifadesinin kontrol edilmesi.

Kod

```
turn off video on stage with 0 % transparency
sürekli tekrarla
  analyse image from stage
  eğer is expression of face 1 happy ise
    Mutlu olmanıza çok sevindim. de
```

Çıktı (Görsel 2.31)



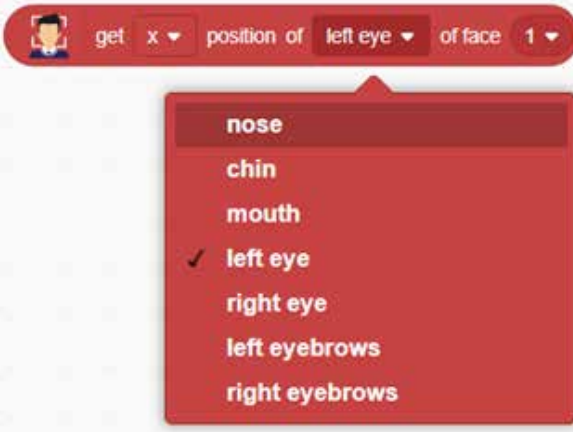
Görsel 2.31: Duygu durumu

```
get x position of face 1
x position
y position
width
height
```

Algılanan yüze ilişkin konum, genişlik ve yükseklik bilgilerini verir.

- x position : Algılanan yüzün x konumu
- y position : Algılanan yüzün y konumu
- width : Algılanan yüzün genişliği
- height : Algılanan yüzün yüksekliği

2.Face Detection/Settings



Bu blok yüz üzerinde belirli bölümlerin (burun, çene, ağız vb.) konumunu almak için kullanılır.

Nose : Burun

Chin : Çene

Mouth : Ağız

Left eye : Sol göz

Right eye : Sağ göz

Left eyebrows: Sol kaş

Right eyebrows: Sağ kaş

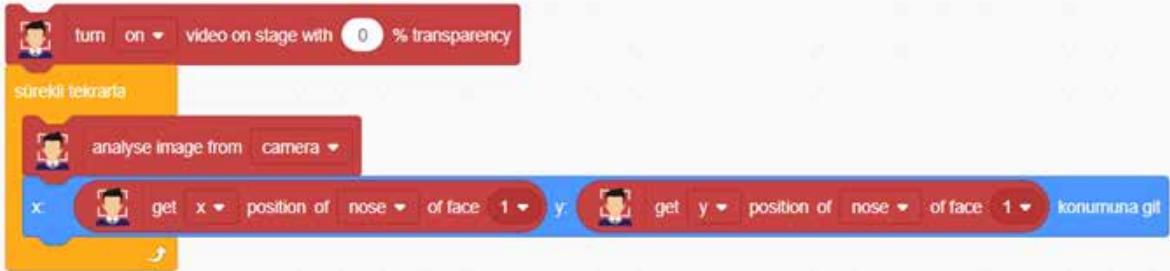
Bu bloklar, yüz üzerinde belirli bölümlere filtreler yapmak için kullanılabilir. Gözlük eklemek, gözleri renklendirmek, kaşları boyamak bu tür uygulamalara örnek verilebilir.



Bu blok yüz üzerinde belirli bölümlerin (burun, çene, ağız vb.) hatlarını belirtmek için kullanılır. Yüz algılama ile yüzdeki belirli bölümler bir hat şeklinde belirtilebilir. Yüzdeki bölümlerin konumunu veren bloktan farklı olarak her bölüm noktalar kümesinden oluşmaktadır.

Kameradaki yüzü algılayarak bir kuklanın insan burnunun konumunu takip etmesi.

Kod



Çıktı (Görsel 2.32)

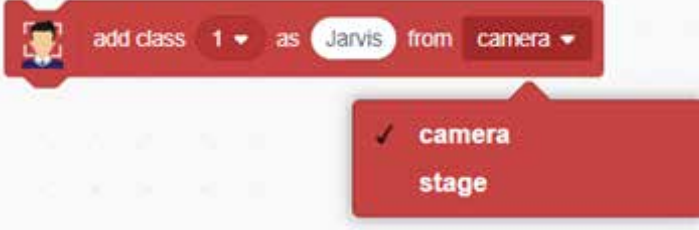


Görsel 2.32: Burun takibi

Face detection/Face Recognition: Training



Yüz tanıma modeli için model eğitiminin yapıldığı aşamadır.



Kameradan/sahneden algılanan yüzün görüntüsünü bir sınıfa bir adla ekler. TM uygulamalarında gösterildiği gibi yüz algılama ve tanıma işlemlerinde de bir kişiye ait yüz görüntüsü bir sınıfa kaydedilebilir. Bu şekilde 10 sınıf oluşturulabilir. Sınıflar oluşturulurken bir etiket tanımlanmalıdır (örneğin kişinin adı). Görüntü kameradan veya sahneden alınabilir.



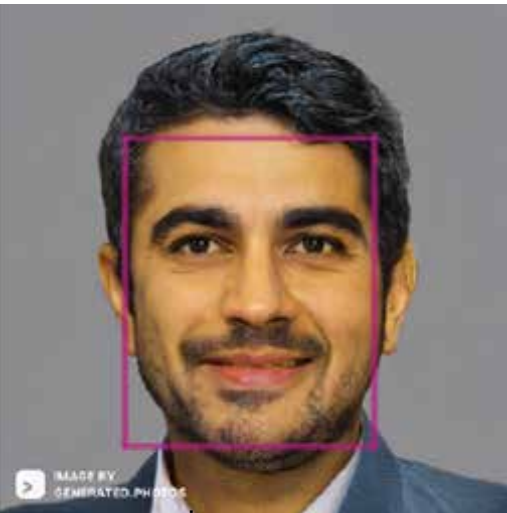
Oluşturulan sınıfları sıfırlamak için kullanılır.

Sahnedeki bir yüz görüntüsünü bir sınıfa kaydetmek.

Örnek kod

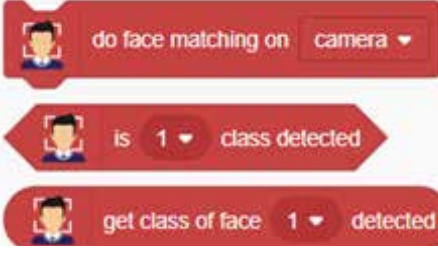


Çıktı (Görsel 2.33)



Görsel 2.33: İnsan yüzü etiketleme

Face detection/Face Recognition: Test



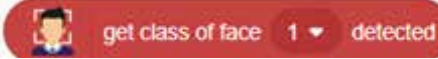
Eğitim işleminden sonra yüz eşleştirme yapılması için test bölümü kullanılır.



Kameradaki veya sahnedeki yüzün eğitim bölümünde oluşturulan sınıflarla eşleştirmesini yapar.



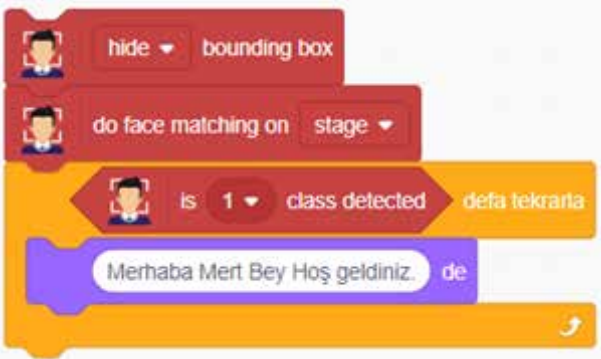
Eşleştirmeden sonra "... sınıfına ait bir yüz tespit edildi" ifadesi için bu blok kullanılır. Eğer blokuyla kullanılarak bir kişinin yüzünün (Örneğin aranan bir kişi) tespit edilip edilmediği kontrol edilebilir.



Algılanan yüzün (1, 2 ve 10.) ait olduğu kişi (sınıf) bilgisini almak için kullanılır.

Eğitilmiş bir sınıfa ait bir yüzün eşleştirilmesi.

Örnek kod



Face detection/Face Recognition: Test

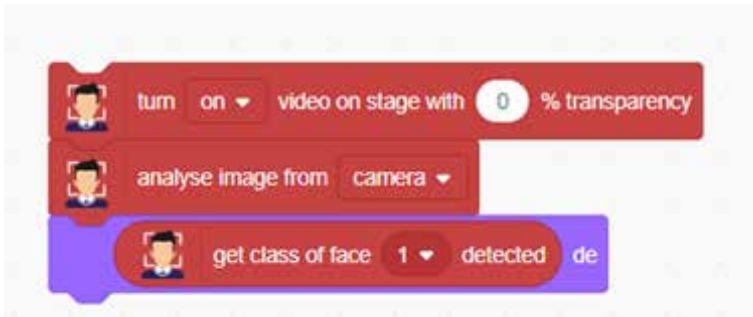
Çıktı (Görsel 2.34)



Görsel 2.34: Sahneden yüz eşleştirme

Kameradaki bir yüzün eşleştirilmesi

Örnek kod





2.5.2.2. Object Detection (Nesne Algılama)

Örüntüdeki nesnelere tanımlanmış nesne sınıflarından hangisine ait olduğu belirlenebilir. Model 80 farklı nesneyi birbirinden ayırt edilebilmektedir.

Bir nesne algılama modeli bir görüntü veya video akışında belirli bir nesne kümesinden hangisinin veya hangilerinin mevcut olabileceğini belirleyebilir ve

görüntü içindeki konumları hakkında bilgi sağlayabilir [9].

Nasıl çalışıyor? Bir nesne algılama modeli, birden çok nesne sınıfının varlığını ve onların konumunu algılamak için eğitilir. Bir model, temsil ettiği nesnenin sınıfını (örneğin bir elma, araba, eşya) belirten bir etiketle birlikte çeşitli nesne görüntüleriyle ve nerede görüldüğünü belirten verilerle eğitilebilir. Daha sonra modele bir görüntü verildiğinde, algıladığı nesnelerin bir listesini, her nesneyi içeren bir sınırlayıcı kutunun (bounding box) konumunu ve algılamanın doğruluğuna dair bir skor, güven puanı verir. Nesne algılama modellerinde verilen görüntü bir sayı dizisine çevrilir. Modelde çıktı olarak ise konumlar, sınıflar, güven skorları ve algılama sayısı üretilir[9].

Konumlar : Algılanan nesnelere ait sınırlayıcı kutuların konumlarını verir.

Sınıflar : Sınıf adlarının indeksini gösterir.

Güven Skorları : Görüntüdeki nesnenin sınıf olasılığını (confidence) temsil eden sayılardır. Olasılık 0-1 arasında bir sayıdır.

Algılama sayısı : Algılanan nesne sayısıdır.

Object detection eklentisi

PictoBlox'da object detection eklentisini kullanmak için oturum açmak gerekir. Object detection eklentisinde kodlar iki kategoride bulunmaktadır.

1.Settings (Ayarlar)

2.Analyse image (Görüntüyü analiz et)

Bu eklentiye ait kod bloklarının işlevleri kategoriler altında örnek kodlarla birlikte verilmiştir. Öncelikle kategorideki kod blokları bütün olarak gösterilmekte daha sonra her kategorinin kodları tek tek açıklanmaktadır.

Object detection/settings



Eklentiyle ilgili ayarların yapıldığı bölümdür. Yüz algılama eklentisinin ayarlar bölümünde bulunan kod blokları ile aynıdır.

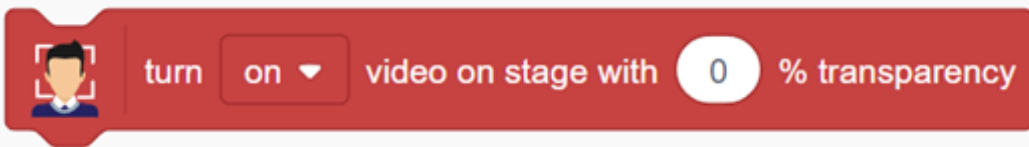
Object detection/analyse image



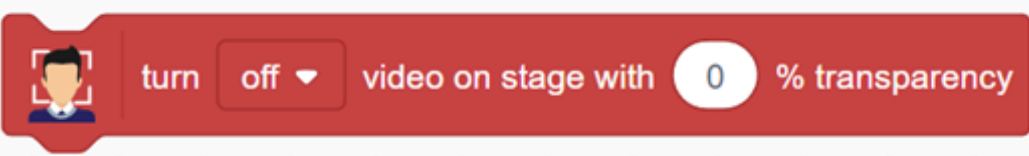
Bu bölümde görüntüyü analiz etmek için kullanılan bloklar yer almaktadır.



Kameradan veya sahneden alınan görüntünün analiz edilmesi için kullanılır. Kameradan görüntünün analiz edilebilmesi bu blok öncesinde



bloğunun kullanılmış olması gerekir. Sahneden (dekor vb.) görüntü almak için



bloku ile kamera kapatılmalıdır.



get # of objects



class

of object

1

- ✓ class
- x position
- y position
- width
- height
- confidence

Analiz edilen görüntüde algılanan nesne sayısını almak için kullanılır.

Algılanan nesnenin (1, 2, 3 ...,10.) bilgilerini almak için kullanılan bloktur. Eklenti 10 nesneye kadar izin vermektedir. Algılanan nesnenin indeksi seçilerek (1-10) şu bilgiler alınabilir.

class: Nesnenin ait olduğu sınıf. Etiket, sınıf adı. Örnek: Kişi, uçurtma, uçak vb.

x position: Nesnenin x düzlemindeki konumu.

y position: Nesnenin y düzlemindeki konumu.

width: Nesnenin genişliği.

height: Nesnenin yüksekliği.

confidence: Nesnenin sınıfa ait olma olasılığı.



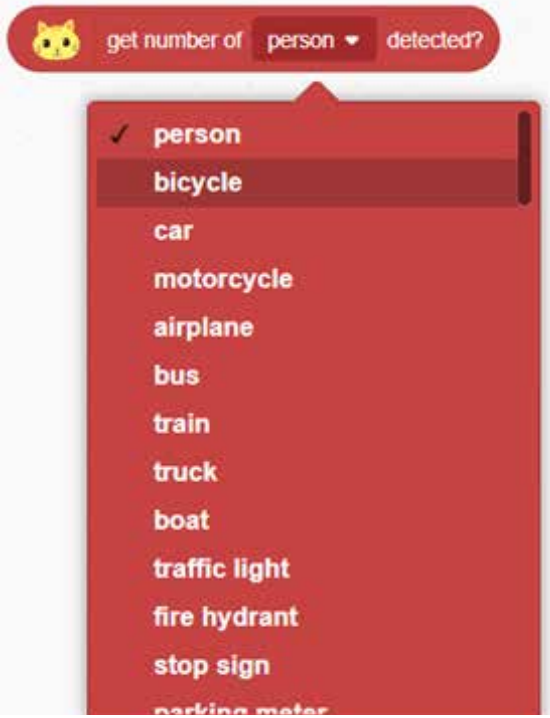
is

person

detected?

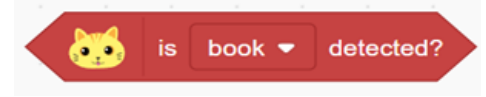
- ✓ person
- bicycle
- car
- motorcycle
- airplane
- bus
- train
- truck
- boat
- traffic light
- fire hydrant
- stop sign
- parking meter

Bu blok, algılanan nesnenin ait olduğu sınıfı listeden kontrol etmek için kullanılmaktadır. Bu bölümde çok sayıda sınıf bulunmaktadır: Kişi, bisiklet, araba, motosiklet, uçak, otobüs, tren, kamyon, hayvan türleri ve ev eşyaları. Bu blok kullanılarak ekranda belirli bir nesne algılandığında belirli kodların çalışması sağlanabilir.

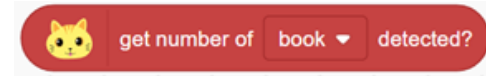


Algılanan görüntüde listeden seçilen sınıfa ait (kişi, bisiklet, araba) nesne sayısını almak için kullanılır.

Örneğin bu bloklar kullanılarak içinde kitap olan görüntüleri ayırmak



ve görüntüdeki kitap sayısını



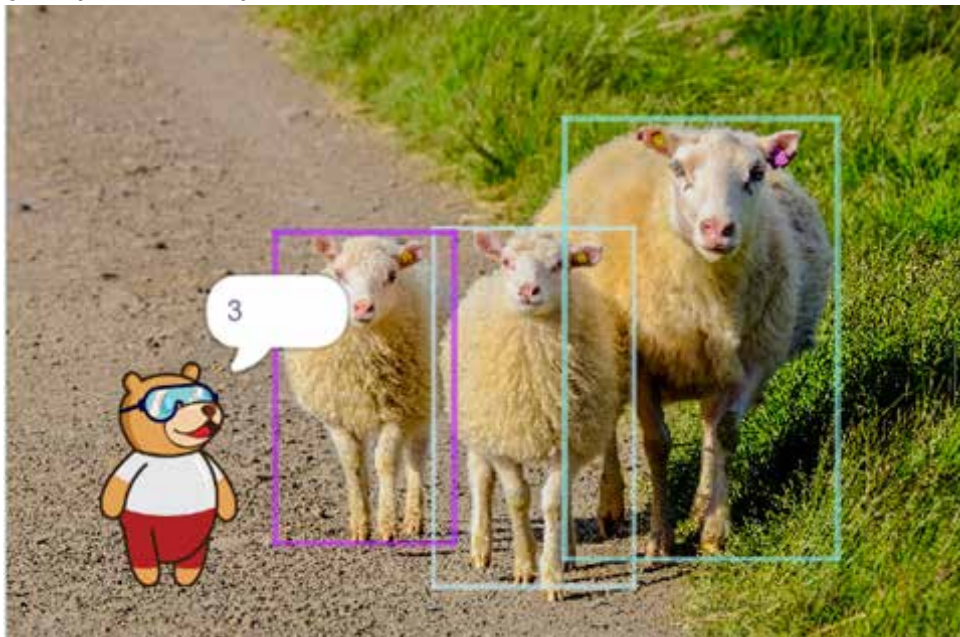
tespit etmek için kullanılabilir.

Sahnedeki görüntüde bulunan bir nesnenin adı ve sayısı belirlenmektedir.

Örnek kod



Çıktı (Görsel 2.35)



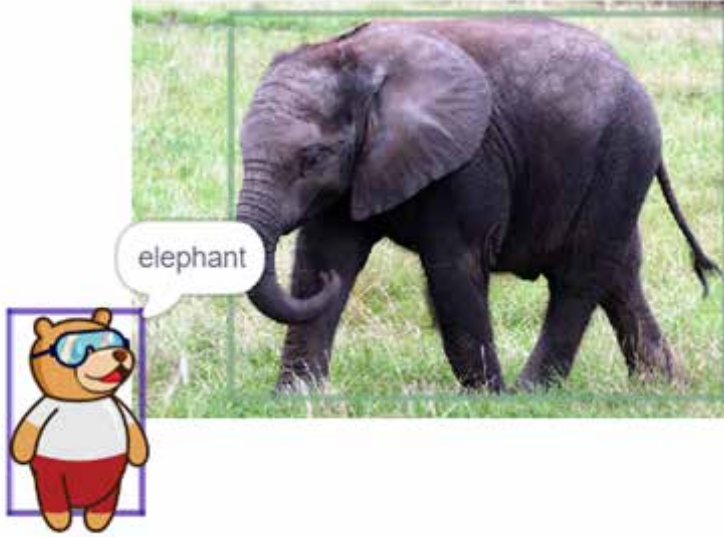
Görsel 2.35: Görüntüdeki nesne sayısı

Sahnedeki nesneyi tanıma.

Örnek kod

```
analyse image from stage
class of object 1 de
```

Çıktı (Görsel 2.36)



Görsel 2.36: Görüntüdeki nesneyi tanımlama

Kameradaki görüntüden nesne sayısını bulma.

Örnek kod

```
turn on video on stage with 0 % transparency
analyse image from camera
get # of objects de
```

Kameradaki görüntüden nesneyi tespit etme.

Örnek kod

```
turn on video on stage with 0 % transparency
analyse image from camera
class of object 1 de
```



2.5.2.3. İnsan Vücudu Algılama

Görüntüden insan vücudunun bölümlerini algılamak için kullanılan eklentidir.

Bu eklenti insan vücudunun bölümlerinin (eklemler, uzuvlar, gövde, baş) uzamsal konumlarını belirleyerek bir görüntü veya videodan bir kişinin pozunu, duruşunu, el hareketlerini tespit etmek için kullanılır.

İnsan vücudunun belirli bir bölümünü bir görüntüden tespit etmek için bilgisayarlı görü teknikleri kullanılır. Bu eklenti temel vücut

eklemlerinin konumunu belirler ancak görüntüdeki kişiyi tanıma işlevi yoktur. Poz tahmin modelleri, giriş olarak bir görüntü alır ve vücut bölümlerine ait anahtar noktalar hakkında bilgi verir. Tespit edilen anahtar noktalar, 0-1 arasında bir güven puanı ve vücudun hangi parçası olduğunu belirten bir indeks (sıra numarası) bilgisiyle birlikte çıktı olarak verilir. Güven puanı, o konumda kilit bir noktanın bulunma olasılığını gösterir [11]. Eklenti el işaretlerini ve insan pozlarını algılayabilir.

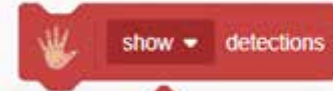
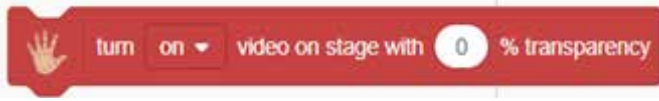
Human body detection eklentisi

PictoBlox'da human body detection eklentisini kullanmak için oturum açılmalıdır. Object detection eklentisinde kodlar üç kategoride bulunmaktadır.

- 1.Settings (Ayarlar)
- 2.Pose detection (Pos algılama)
- 3.Hand detection (El algılama)

Bu eklentiye ait kod bloklarının işlevleri kategoriler altında örnek kodlarla birlikte verilmiştir. Öncelikle kategorideki kod blokları bütün olarak gösterilmekte daha sonra her kategorinin kodları tek tek açıklanmaktadır.

Human body detection/settings

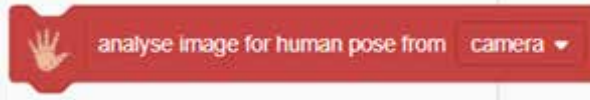


Bu bölümdeki ayarlarda diğer eklentilerden farklı olarak bounding box yerine detections (algılamaları) göster ve gizle bloku bulunmaktadır.

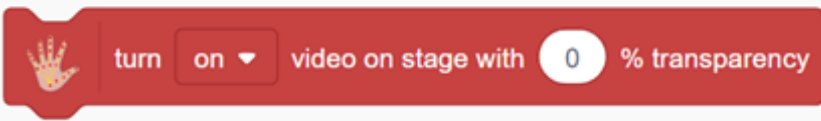
Human body detection/pose detection



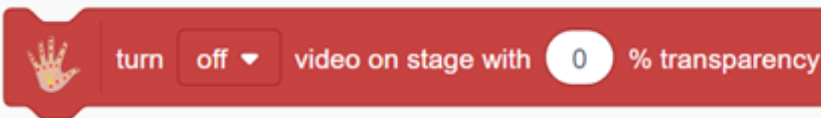
Duruşu, pozu algılamayla ilgili bloklar bu bölümde yer almaktadır.



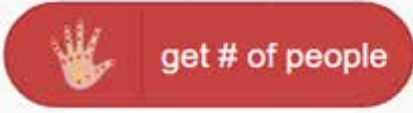
Kameradan veya sahneden alınan görüntünün analiz edilmesi için kullanılır. Kameradan görüntünün analiz edilebilmesi bu blok öncesinde



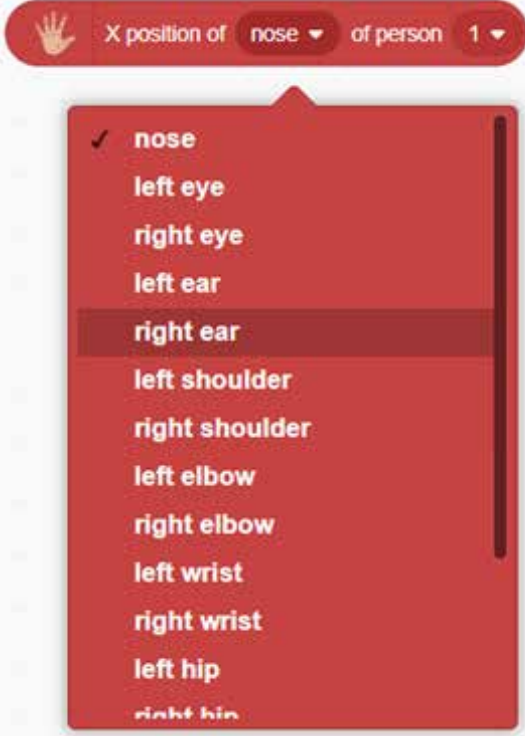
bloku kullanılmış olmalıdır. Sahneden (dekor vb.) görüntü almak için



bloku ile kamera kapatılmalıdır.

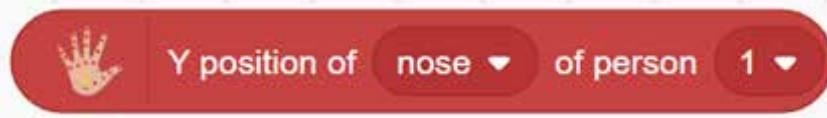


Analiz edilen görüntüde algılanan insan sayısını almak için kullanılır.

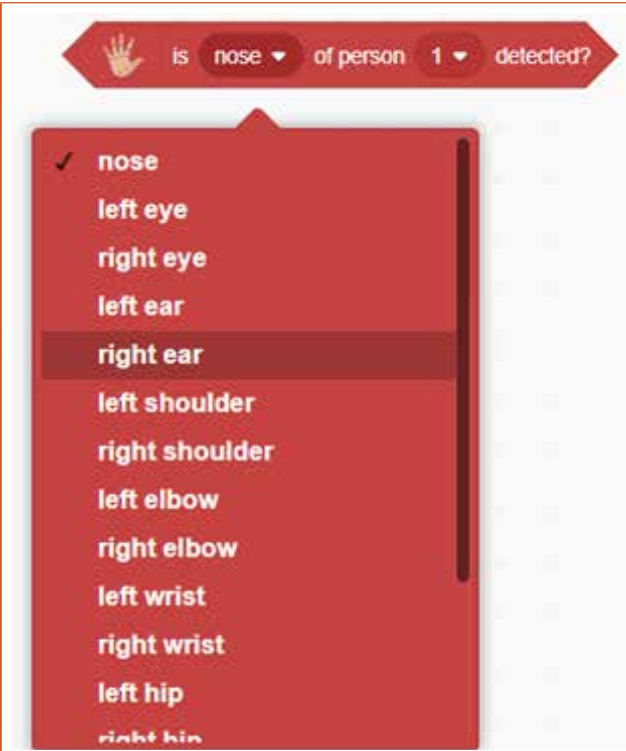


Görüntüdeki kaçınıcı kişinin (1-10) ve vücudunun hangi parçası olduğunu (burun, sol göz, sağ göz vb.) ayarlanır. Belirtilen kişinin belirtilen vücut parçasının x düzlemindeki konum bilgisini almak için kullanılır. Eklenti 10 kişiye kadar işlem yapmaya izin vermektedir. Görüntüde bir kişi varsa person 1 olarak ayarlanır.

Konumu alınabilecek vücut parçaları ise listedeki sırasıyla şunlardır: Kulak, sol göz, sağ göz, sol kulak, sağ kulak, sol omuz, sağ omuz, sol dirsek, sağ dirsek, sol el bileği, sağ el bileği, sol kalça, sağ kalça, sol diz, sağ diz, sol ayak bileği ve sağ ayak bileği.



Bu blok ile görüntüdeki kaçınıcı kişinin (1-10) ve vücudunun hangi parçası olduğu (burun, sol göz, sağ göz vb.) ayarlanır. Belirtilen kişinin belirtilen vücut parçasının y düzlemindeki konum bilgisini almak için kullanılır.



Görüntüdeki kişilere ait vücut parçalarından hangilerinin algılandığını belirlemek için kullanılır. Örnek: Birinci kişiye ait burun algılandı mı? **Eğer** bloku ile kullanılarak belirtilen kişilere ait belirtilen vücut parçaları algılandığında belirli blokları çalıştırmak için kullanılır.

Sahnedeki bir insan vücudunun analiz edilmesi.

Kod



Çıktı (Görsel 2.37)



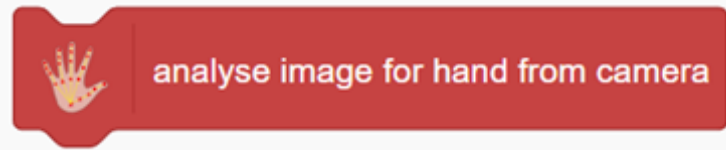
Görsel 2.37: İnsan vücudu tanıma

Human body detection/ Hand detection

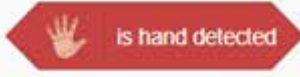
Hand Detection



Bu bölüm görüntüdeki eli algılamak için kullanılmaktadır. Elin konumu ve ebatları, parmakların konumu bu bloklar kullanılarak tespit edilir.



El analizi işlemi için yalnızca kameradan görüntü alınmaktadır



Bu blok görüntüde bir el algılanıp algılanmadığını kontrol için bir eğer bloku içinde kullanılır. El algılandı ise belirli işlemleri yapmak için kullanılır.



Parmaklardaki eklemlerin konumlarını (x ve düzleminde) almak için kullanılır.



Parmağın hangi ekleminin seçileceği belirtilir. Uzantının doğru çalışabilmesi için insan elini bir bütün olarak algılaması gerekir.

top: Üst

middle1: Orta1

middle2: Orta2

bottom: Alt

Hangi parmağın seçileceği belirtilir.

x position of top of thumb

- ✓ thumb
- index finger
- middle finger
- ring finger
- pinky finger

thumb : Başparmak
index finger : İşaret Parmağı
middle finger : Orta parmak
ring finger : Yüzük parmağı
pinky finger : Serçe parmak

Bu blok elin konumu, genişliği ve yüksekliği gibi bilgileri almak için kullanılır.

x position of hand

- ✓ x position
- y position
- width
- height

x position : Elin x düzlemindeki konumu
y position : Elin y düzlemindeki konumu
width : Elin genişliği
height : Elin yüksekliği

Bu blok bir kuklayı elin boyutlarına eşlemek veya kuklayı elin bulunduğu konuma hareket ettirmek için kullanılabilir.

Kameradaki elin tespit edilmesi.

Örnek kod

turn on video on stage with 100 % transparency

sürekli tekrarla

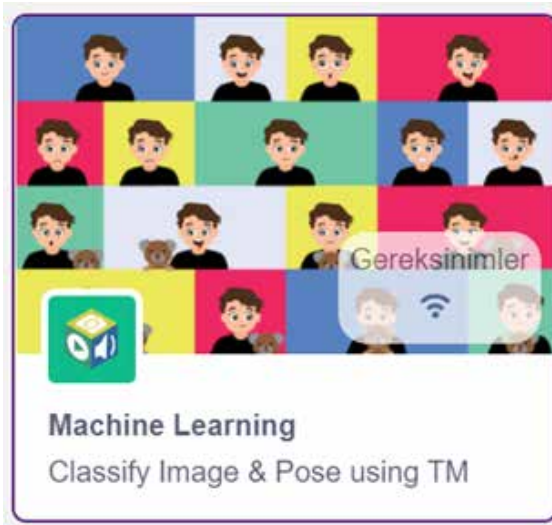
analyse image for hand from camera



Görsel 2.38: Kameradan el tespiti

2.5.2.4. Machine Learning (Yapay Öğrenme/Makine Öğrenmesi)

TM kullanarak poz ve görüntü sınıfları oluşturmak için kullanılır. Bu eklenti için internet bağlantısı gerekir.



Machine Learning

Create a Model

Load a Model

Machine Learning eklentisiyle TM üzerinden **create a model** (model oluşturma) butonu ile modeller oluşturulabilir veya önceden oluşturulan modeller **load a model** (modeli yükle) kullanılabilir.

Create a Model: Bu butona tıklandığında TM web sayfası açılır.

Load a Model: Butona tıklanarak açılan pencerede önceden oluşturulan modelin URL adresi girilir. Model türü seçilir ve **Load Model** butonuna tıklanır. Önceden oluşturulmuş modele ait bloklar oluşur.

2.5.2.5. Artificial Intelligence (Yapay Zekâ)

Bu eklenti speech recognition (ses tanıma), image recognition (görüntü tanıma), OCR (el yazısını ve basılı metni yazıya çevirme) ve face feature (yüz özelliklerini tanıma) işlemleri için kullanılmaktadır.

Bu eklenti için internet bağlantısı gerekmektedir. Ayrıca bu eklenti bulut servisleri üzerinden çalıştığı için PictoBlox'ta oturum açmak gerekmektedir. PictoBlox yeni kullanıcılar için bulut

servislerde kullanmak üzere belirli bir kredi vermektedir. Bu kredi bittiğinde ücretli kredi almak gerekir. Eklenti bulut servislerini kullandığı için analiz sonuçlarının raporlanması birkaç saniye sürebilmektedir.

Bu eklentiyle ünlü kişiler, markalar ve nesnelere algılanabilir. Bu eklentide ayarlar, tanıma, ses tanıma, görüntü tanıma, OCR tanıma ve yüz özellikleri bölümleri bulunmaktadır.

Artificial intelligence eklentisi

Artificial intelligence eklentisinde kodlar altı kategoride bulunmaktadır.

1.Settings (Ayarlar)

2.Recognition (Tanıma)

3.Speech recognition (Konuşma tanıma)

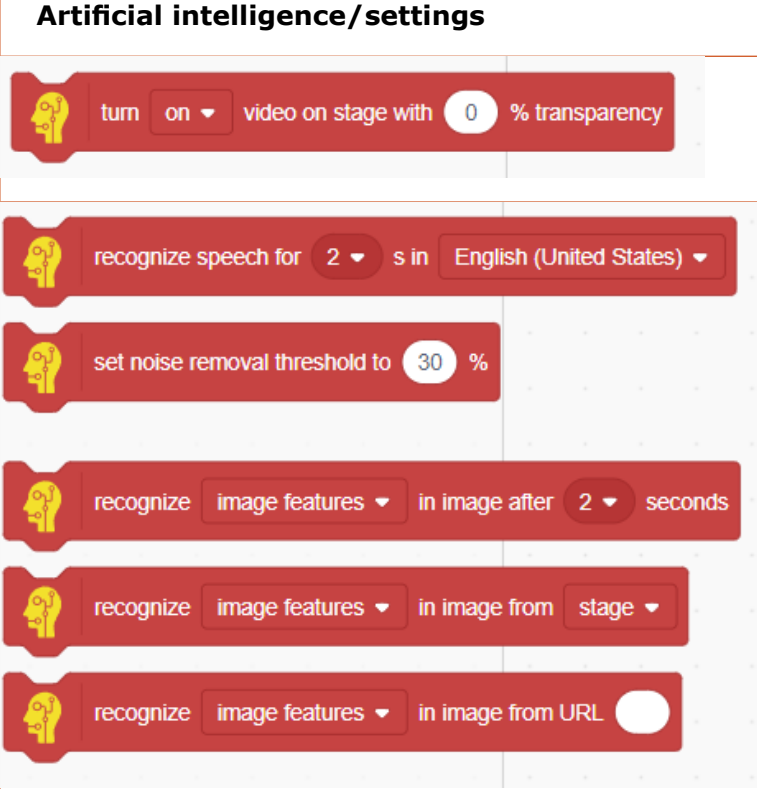
4.Image recognition (Görüntü tanıma)

5.OCR recognition (Yazı tanıma)

6.Face feature (Yüz özellikleri)

Bu eklentiye ait kod bloklarının işlevleri kategoriler altında örnek kodlarla birlikte verilmiştir. Öncelikle kategorideki kod blokları bütün olarak gösterilmekte daha sonra her kategorinin kodları tek tek açıklanmaktadır.

Artificial intelligence/settings



The screenshot displays a user interface for configuring artificial intelligence settings. It features several red code blocks with a yellow head icon containing a brain. The blocks are arranged in a grid-like structure. The first block is titled 'turn on video on stage with 0 % transparency'. The second block is 'recognize speech for 2 s in English (United States)'. The third block is 'set noise removal threshold to 30 %'. The fourth block is 'recognize image features in image after 2 seconds'. The fifth block is 'recognize image features in image from stage'. The sixth block is 'recognize image features in image from URL'. The seventh block is 'recognize speech for 2 s in English (United States)'. The interface includes dropdown menus for language and image source, and input fields for time and percentage values.

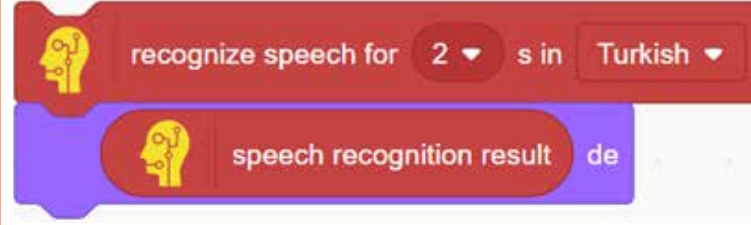
Diğer eklentilerde olduğu gibi bu bölümde kameradan görüntü almak için gerekli blok bulunur.

Bu bölümde ses tanıma, sesteki gürültüyü silme, görüntü özelliklerini, yüz detaylarını, el yazısını ve basılı metni algılama blokları yer almaktadır.

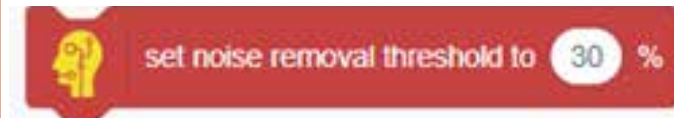
Seçilen dilde belirlenen süre kadar ses tanıma yapmak için kullanılır. Tanıma penceresinin kaç saniye açık kalacağı ve hangi dilde ses tanıma yapılacağı ilgili listelerden seçilir.

Sesin metne çevrilmesi.

Örnek kod



Görsel 2.39: Sesin metne çevrilmesi



Sesteki gürültüyü silmek için eşik değeri ayarlamak amacıyla kullanılır. Ortam gürültüsü fazla ise eşik değeri arttırılabilir.



Görüntüden algılanması istenen özellik seçilir. Algılama işlemi için bir süre belirtilir.

image features: Görüntü özellikleri, görüntüdeki marka ve nesnelere algılamak için kullanılır.

face details: Yüz detayları

handwritten text: El yazısı
printed text: Basılı metin (dijital yazı)

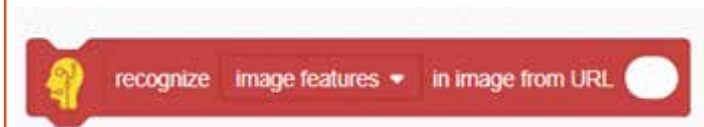


Bir sahnedeki, kostümdeki veya dekordaki görüntüden özelliklerini, yüz detaylarını, el yazısını veya basılı bir metni tanımak için kullanılır.

Stage: Sahne

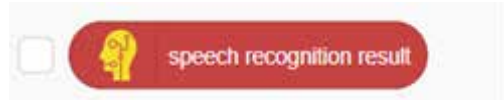
Costume: Kostüm

Backdrop: Dekor



İnternet üzerinden bir görüntüden özellikleri, yüz detaylarını, el yazısını veya basılı bir metni tanımak için kullanılan bloktur. Bu alana internette bulunan resmin bağlantı adresi girilmelidir.

Artificial intelligence/speech recognition



Bu bölümde ses tanıma sonucunu metin olarak veren kod bloku bulunmaktadır. Ses tanıma bloku tarafından alınan ses sunucuya gönderilir, işlenen ses metin olarak bu blokta tutulur.

Artificial intelligence/image recognition



Bu bölümde görüntü tanıma, görüntüdeki nesne sayısı, konum ve yüz hatları gibi bloklar bulunmaktadır.



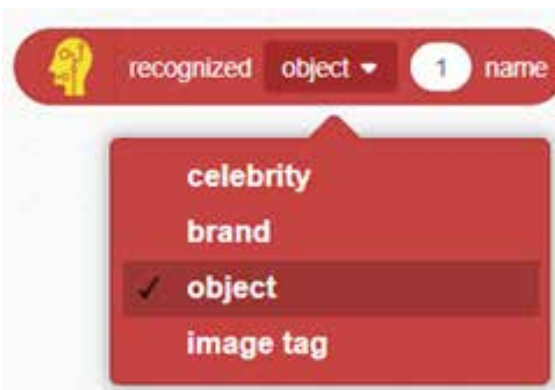
Celebrity: Ünlü kişi

Brand: Marka

Object: Nesne

Image tag: Görüntü etiketi

Görüntüde bulunan ünlü, marka, nesne veya resim etiketi sayısını almak için kullanılır.



Tanınan ünlü, marka, nesne ve görüntü etiketinin bilgilerini almak için kullanılan bloktur. Birden fazla tanıma için ... name boşluk kısmına 1, 2 gibi sayılar verilebilir. Bu şekilde "tanınan birinci ünlünün adı" gibi çıktılar üretir. Bir döngüde kullanılarak sırasıyla algılanan ünlüyü, markayı, nesneyi veya görüntü etiketini yazdırmak için kullanılabilir.

Sahnedeki ünlüyü tanıyan bir uygulama

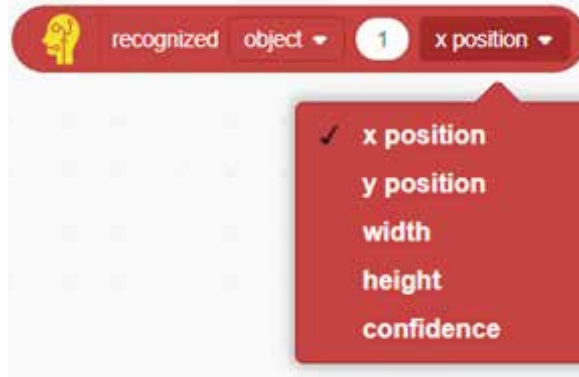
Örnek kod



Çıktı (Görsel 2.40)



Görsel 2.40: Ünlü tanıma



x position: Tanınan ünlü/marka/ nesne x düzlemindeki konumu

y position: Tanınan ünlü/marka/ nesne y düzlemindeki konumu

width: Tanınan ünlü/marka/ nesne genişliği

height: Tanınan ünlü/marka/ nesne yüksekliği

confidence: Tanınan ünlü/marka/ nesne güven skoru.



Tanımada sonucunda görüntüdeki hatları veya bir görüntünün açıklamalarını almak için bu blok kullanılır.

Artificial intelligence/OCR recognition

OCR Recognition



OCR (optik karakter tanıma) işlemlerini yapmak için kullanılan bu bölümde el yazısından veya basılı metinden dijital metne dönüştürme işlemi yapılır.

Bu blokları kullanmadan önce tanıma bloku eklenir ve listeden el yazısı veya basılı metin seçilir.

Bir resim üzerindeki yazının analiz edilerek sonucunun yazdırılması.

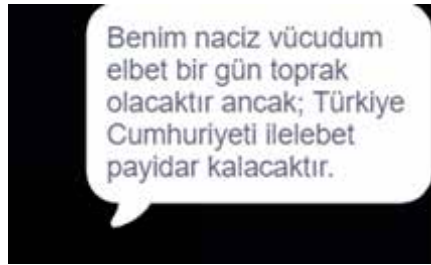
Örnek kod



Resim adresi:

<https://i2.milimaj.com/i/milliyet/75/0x0/5db7efd255427f22f4c4b658.jpg>

Çıktı (Görsel 2.41)




Görsel 2.41: Basılı metinden karakter tanıma



Sonrasında handwritten text result (el yazısı metni sonucu) ve printed text result (basılı metin sonucu) blokları kullanılır. Bu şekilde bir el yazısı veya bir basılı metindeki yazı tanınır.

Artificial intelligence/Face Feature

 recognized face count

Görüntüde tanınan yüz sayısını almak için kullanılan bloktur.

 recognized age of face 1


✓ age
gender
emotion

Bu blok bir döngü içine alınarak görüntüdeki her yüze ait yaş, cinsiyet ve duygu durumu tanıma sonucu gösterilir.

 emotion happiness for face 1 ?

✓ happiness
anger
contempt
disgust
fear
neutral
sadness
surprise

Birinci yüze ait duygu durumu ... mı? Eğer bloku içinde kullanarak belirli duygu durumları için belirli kodların çalışması sağlanabilir. Duygu durumları listedeki sırasıyla mutlu, sinirli, küçümseme, iğrenme, korku, doğal, üzgün ve şaşkın olmak üzere 8 türdür.

 head gesture roll angle for face 1

Bu blok baş hareketinin açısını almak için kullanılır.

 recognized x position for face 1

✓ x position
y position
width
height

x position: Algılanan yüzün x düzlemindeki konumu

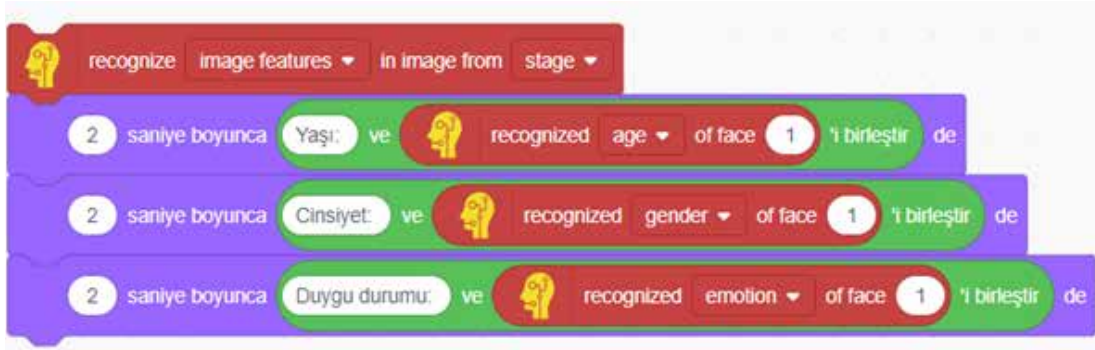
y position: Algılanan yüzün y düzlemindeki konumu

width: Algılanan yüzün genişliği

height: Algılanan yüzün yüksekliği

Bu blok bir kuklayı yüzün boyutlarına eşlemek veya kuklayı yüzün bulunduğu konuma hareket ettirmek için kullanılabilir.

Sahnedeki görüntüye ait yaş, cinsiyet ve duygu durumu özelliklerinin gösterilmesi.
Örnek kod



Çıktı (Görsel 2.42)



Görsel 2.42: Yaş tanıma

Kameradaki kişinin yaş, cinsiyet ve duygu durumu özelliklerinin gösterilmesi.

2.5.2.6. Natural Language Processing (Doğal Dil İşleme)

Doğal dil işleme insan dilinin bilgisayar tarafından anlaşılması ve sentezlenmesidir. Sanal sohbet botları, metinden duygu analizi ve metinden özet çıkarma gibi uygulamalar doğal dil işleme teknikleri ile yapılmaktadır.

Natural language processing eklentisi

Doğal dil işleme eklentisi kullanılarak bir metin sınıflandırıcı oluşturulabilir. TM'de görüntüleri sınıflandırmak için model oluşturmaya benzer şekilde metinler için sınıflar oluşturularak örneklerle modeller eğitilebilir. Doğal dil işleme eklentisi ile



text classifier (metin sınıflandırıcı) ve sentiments analyzer (duygu analizcisi) kullanılarak uygulamalar geliştirilebilir. Metin sınıflama için örnek bir haber metninin ne tür bir haber türüne (spor, siyaset, günlük) ait olduğunun tahmin edilmesi olabilir. Duygu analizi ise müşteri yorumlarının pozitif, negatif veya nötr olarak sınıflandırılmasıdır.

Natural Language Processing/Text Classifier

Download Template CSV

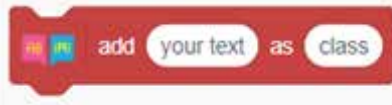
Add/Update data from .csv file

Download Template CSV

CSV şablonunu indirmek için kullanılır.

Add/Update data from csv file

Eğitim işlemi için gerekli verinin dosyadan eklenmesini veya güncellenmesini sağlar.

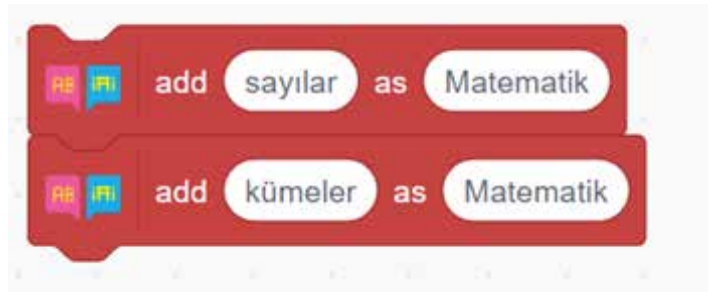


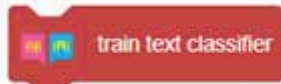
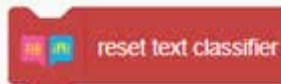

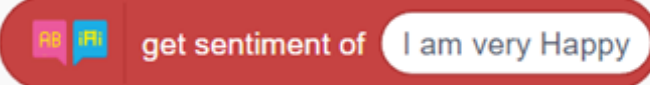
Veri seti bir dosyadan alınabileceği gibi manuel olarak da sınıflar oluşturulabilir. Bu bölümde metin örneklerinin ait oldukları sınıflar etiketlenir.

Your text: Sınıfa ait örnek metin

Class: Sınıf adı

Sınıfa örnek tanımlanması



	Bu blok, metin örnekleri etiketlendikten (sınıfları belirlendikten) sonra metin sınıflandırıcıyı eğitmek için kullanılır.
	Metin sınıflandırıcıyı sıfırlamak için kullanılır.
	Bir model eğitildikten sonra yeni bir örnek verildiğinde bu örneğin ait olduğu sınıfı öğrenmek için kullanılmaktadır.
	Bu blok verilen metnin duygu durumunu görmek için kullanılır. Duygu analizi, önceden eğitilmiş modele örnekler verilerek duygu durumu (pozitif, negatif, normal) tahmin ettirilir. Duygu analizi için metnin İngilizceye çevrilmesi gerekir. Bunun için çeviri eklentisi kullanılabilir.

Metin sınıflandırma aşamaları

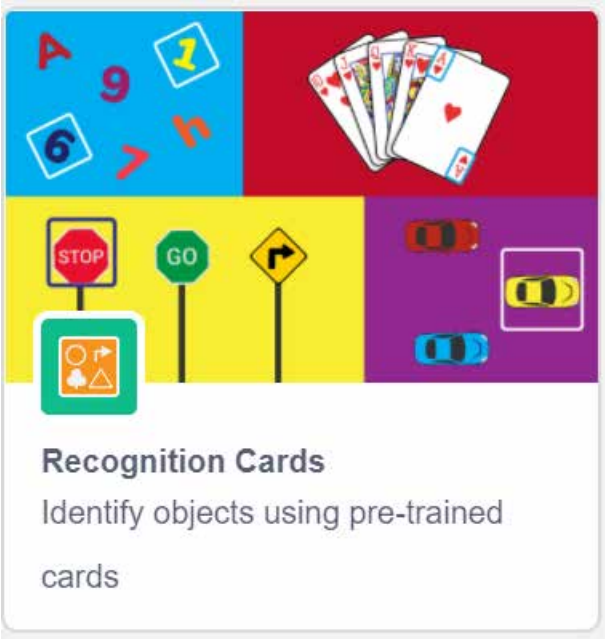
1. Sınıflar için metin örnekleri girilir veya CSV dosyasından yüklenir.
2. Metin sınıflandırıcı modeli eğitilir.
3. Test verisi (yeni örnekler) için tahminler yaptırılır.

2.5.2.7. Kartları Tanıma (Recognition cards)

Önceden eğitilmiş modelleri kullanarak kartları tanıyan bir eklentidir.

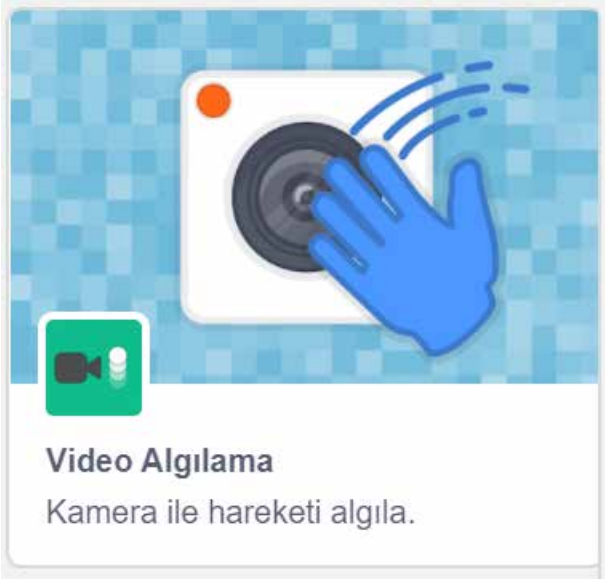
Bu eklenti bir tür bilgisayarlı görü modelidir. Bu eklenti trafik sinyallerini, numaraları, konumları analiz etmek için kullanılmaktadır.

Kartları tanıma eklentisi kartlar üzerinde bulunan resimleri tanımak (araba, uçurtma, kişi, uçak, trafik ışığı, trafik levhaları, ev eşyaları vb.) için kullanılır. Bloklar ve kullanım, object detection eklentisiyle çok benzerdir.

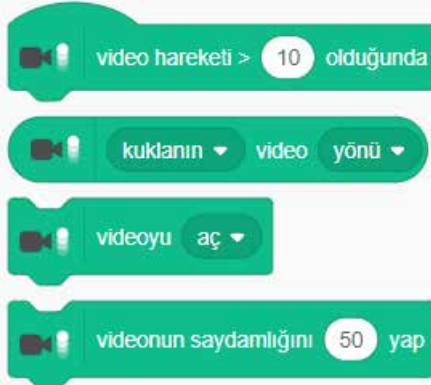


2.5.2.8. Video Algılama

Video ile ilgili işlemleri yapmak için kullanılan eklentidir. Kukla veya sahne üzerinde hareket olması veya hareketin yönü gibi bloklar kullanılarak uygulamalar geliştirilebilir. Örneğin hareket ederek balon patlatma oyunu gibi. Video algılamada video hareketi ve kukla hareketi kod blokları bulunmaktadır (Görsel 2.43).



Video Algılama



Görsel 2.43: Video algılama blokları



Metinden Sese

Metinden sese eklentisi bir metni seslendirmek için kullanılmaktadır. Ses tonu (alto, tenor, ciyak, dev ve kedi yavrusu) ve dil ayarlanarak (İngilizce, Türkçe vb.) seslendirme yapılabilmektedir. İnternet bağlantısı gerekir.

Yazılan metni seslendiren bir uygulama.

Örnek kod



Boşluk tuşuna basınca

1. Ses tonu ayarlanır.
2. Konuşma dili Türkçe olarak ayarlanır.
3. Yazılan metin seslendirilir.

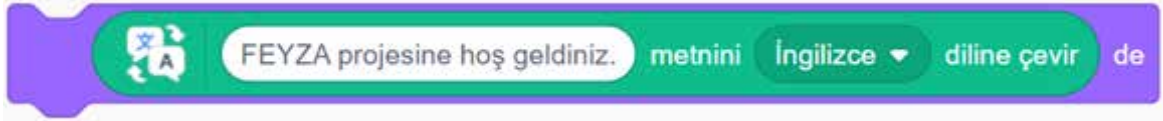


Çeviri

Verilen metni belirtilen dile çevirmek için kullanılır. Kullanmak için internet bağlantısı gerekir.

Türkçe bir cümleyi yabancı bir dile çeviren uygulama.

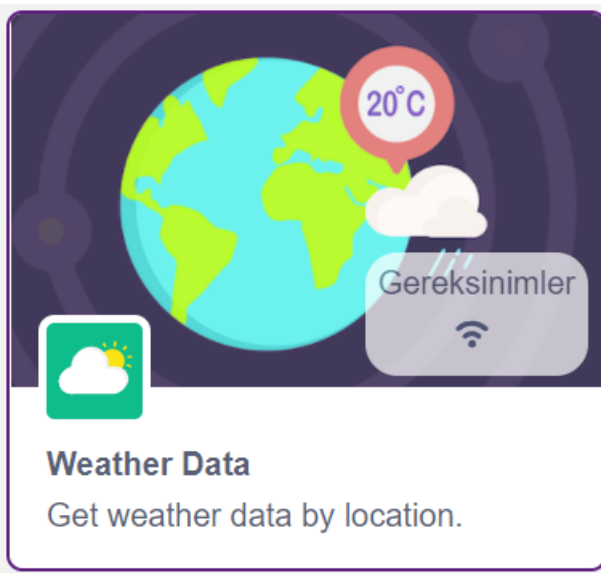
Örnek kod



Çıktı (Görsel 2.44)



Görsel 2.44: Çeviri uygulaması



Weather Data (Hava Verisi)

Hava verisi bir IoT eklentisidir. Bu eklenti internet erişimi gerektirir. Öncelikle <https://openweathermap.org/api/> üzerinden bir hesap oluşturularak bir API erişim anahtarı oluşturulur. Bu API kullanılarak servis üzerinden konumu verilen bir yere ait hava durumu bilgileri alınabilir.

Set weather API to (your API)

Servisi kullanmak için oluşturulan API bu blokta girilerek servise erişim sağlanır.

API, iki uygulamanın birbiriyle konuşmasına olanak tanıyan bir yazılım aracıdır. Uygulama Programlama Arayüzü'nün kısaltmasıdır [12].

Weather Data



Enlem ve boylam bilgileri verilen konuma ait hava durumu verisini almak için kullanılır.



In Celcius: Hava sıcaklığını santigrat derece olarak verir.

min. in Celsius: En düşük hava sıcaklığını santigrat derece olarak verir.

max. in Celsius: En yüksek hava sıcaklığını santigrat derece olarak verir.

In Fahrenheit: Hava sıcaklığını Fahrenheit olarak verir.

min. in Fahrenheit: En düşük hava sıcaklığını Fahrenheit olarak verir.

max. in Fahrenheit: En yüksek hava sıcaklığını Fahrenheit olarak verir.



Pressure: Hava basıncı

Humidity: Nem oranı (yüzde olarak)



Coordinate latitude/ longitude: Koordinat enlem/ boylam



Wind (speed m/s): Rüzgârın hızı

Direction: Rüzgârın yönü



Sunrise time: Gündoğumu zamanı

Sunset time: Günbatımı zamanı

Data capture time: Veri alma zamanı



Weather: Güneşli, bulutlu vb.

City: Hava durumu alınan şehir

Add your own Weather API (Faster)



Servisi kullanmak için API anahtarı buraya girilir.

Connect to Wi-Fi (U)



İnternet bağlantısı yapmak için gerekli bloktur. Wi-Fi ağının adı ve parolası girilerek bağlantı için gerekli ayarlar yapılır.

PictoBlox Programı ile Uygulamalar

PictoBlox'da amaca ve ihtiyaca göre yapay zekâ eklentilerinden biri veya birkaçı kullanılarak uygulamalar geliştirilebilir.



4. Uygulama: Yüz Hatları

sağlanmaktadır.

Bu uygulamada PictoBlox ile insan yüzündeki belirli bölümlerin tespit edilmesi ve işaretlenmesi (Visage Teknoloji firmasının sunduğu demodaki gibi)



Ön Hazırlık

Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

Bilgisayarlı görü (yüz tanıma)

PictoBlox Eklentileri

Face detection

Uygulama Adımları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede varsayılan olarak Tobi kuklası gelir. Sahnenin sağ alt tarafında bulunan kukla özelliklerinden Tobi kuklası silinir. Yeni kukla ekle penceresinden Face Filters kuklası sahneye eklenir ve kostümler sekmesinden varsayılan kostüm olarak ball kuklasının kostümü face filters kuklasına ekstra olarak eklenir ve varsayılan olarak ayarlanır.



2. Eklenti ekleme bölümünden face detection ve metinden sese eklentileri yüklenir.
3. Kuklanın kodlanması (Görsel 2.45, Görsel 2.46 ve Görsel 2.47)

! Blokların renklerine dikkat edilerek ait oldukları blok grupları bulunabilir.



Görsel 2.45: Yeşil bayrak tıklandığında çalışacak kodlar

Landmark yüzdeki işaretleme sayısını tutan bir değişkendir.

Eğer kameradaki görüntüde en az 1 yüz tespit edilirse ($get\#faces > 0$) yeşil topun toplamda 68 ikizi oluşturulur.

Yeşil topun ikizleri yüzdeki belirli noktaların bulunduğu x, y pozisyonlarına konularak yüzün belirli noktaları işaretlenir.

Yeşil top yüzde bulunan 68 noktada sıralı bir işaretleme yapar (Görsel 2.46).



Görsel 2.46: message1 haberi salındığında çalışacak kodlar

Yeşil bayrağa tıkladığında kukla gizlenir ve boyutu %15 olarak ayarlanır.

Turn on video on stage... bloku kullanılarak yüz algılama için kamera görüntüsü sahnede açılır.

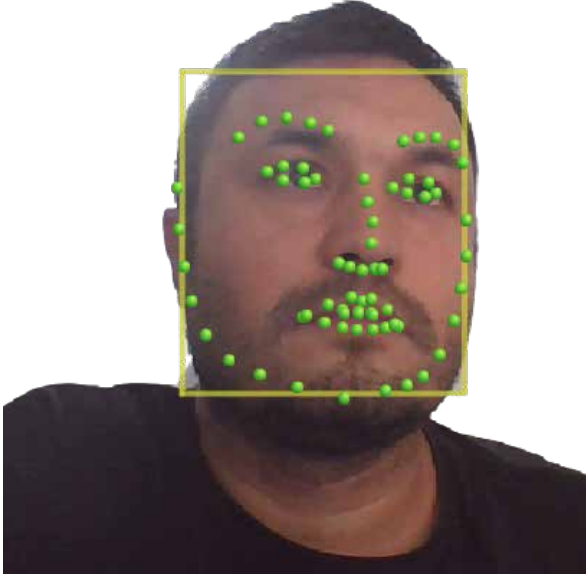
Şu işlemler sürekli tekrarlanır.

Face analyze image from camera bloku kameradan yüz görüntüsü alınmasını sağlayan **face detection** eklentisine ait bir bloktur. Yüzün görüntüsü alındıktan sonra varsa önceki işaretlemeleri silmek için **message1** haberi salınır.

İkiz başladığında çalışacak kodlar (Görsel 2.47)



Görsel 2.47: Bir ikiz başladığında çalışacak kodlar



Görsel 2.48: Landmark uygulaması sonucu

4. Uygulamanın çalıştırılması:

Yeşil bayrak tıkladığında uygulama çalışmaya başlar.

Kameradan bir yüz tespit edildikten sonra işaretleme işlemi başlar.

Bu esnada kameradaki yüz hareket etse bile işaretleme bitene kadar yüzün eski konumu kullanılır.

Uygulama çalıştırıldığında yüzdeki hatlar yeşil topla işaretlenir (Görsel 2.48).

PictoBlox'da uygulama yapıldıktan sonra dosya kaydedilirken kullanıcıya yüz verisinin uygulamayla birlikte kaydedilip kaydedilmeyeceği sorulmaktadır. Save with Face Details butonu tıklanırsa uygulamada oluşturulan yüz profilleri kaydedilir. Böylelikle uygulama daha sonra açıldığında önceden tanıtılmış bir yüzü eşleştirebilir. Uygulamalar hem kişisel hesaba hem de yerel bilgisayara kaydedilebilmektedir.



5. Uygulama: Yüz Tanıma ile Kapı Açma

Bu uygulamada PictoBlox ile bir yüz tanıma sistemi oluşturulmaktadır. Yüz tanıma sistemi akıllı evin bir parçası olarak eve girişte kimlik tespiti için kullanılmaktadır.



Ön Hazırlık

Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

Bilgisayarlı görü (yüz tanıma)

Ses tanıma

PictoBlox Eklentileri

Face detection

Text to speech

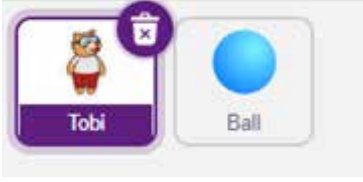
Uygulama Adımları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede varsayılan kukla olarak Tobi kuklası gelir. Sahne arka planı uygun bir şekilde ayarlanır (Görsel 2.49). Bir dış kapı görüntüsü ve bir oda görüntüsü dekor olarak eklenir.



Görsel 2.49: Sahne

2. Kuklalar bölümünde Ball kuklası eklenir. Kukla eklendikten sonra sahnede mavi bir top görünecektir (Görsel 2.50).



Görsel 2.50: Kuklalar

3. Eklenti ekleme bölümünden face detection ve metinden sese eklentileri yüklenir
4. Tobi kuklasının kodlanması (Görsel 2.51, Görsel 2.52 ve Görsel 2.53).



Görsel 2.51: Yeşil bayrak tıkladığında çalışacak kodlar

Yeşil bayrağa tıklanarak uygulama çalıştırıldığında Tobi'nin kullanıcıları yönlendirmesi için gerekli kodlar eklenmiştir. Sisteme bir kez yüz tanıma işlemi yapıldıktan sonra yüz eşleştirme ile eve giriş yapılmasına izin verilmektedir.

Turn of video on stage... bloku başlangıçta sahnede kameranın kapalı olmasını sağlamak için kullanılmaktadır.

Sesi tenor e ayarla bloku konuşmacının ses tonunu ayarlamak için kullanılmaktadır. Farklı ses tonları ayarlanabilir.

Door dekoruna geç bloku dekoru ayarlamak için kullanılmıştır.

... de bloku ile Tobi kuklasının sahnede yazıyla iletişim kurmasını sağlar.

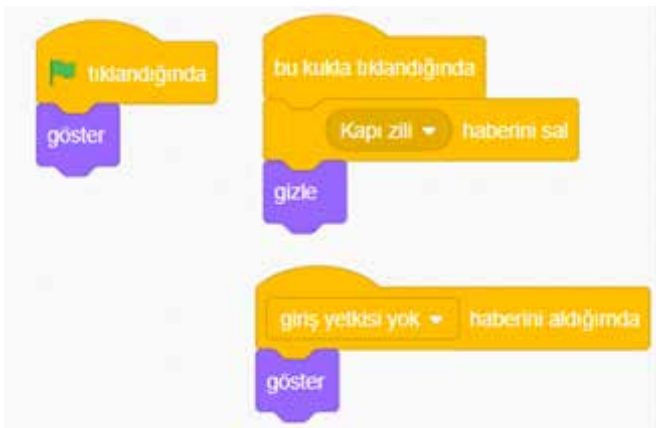
konus bloku metni sese çeviren bir yapay zekâ eklentisine ait bloktur.

! Blokların renklerine dikkat edilerek ait oldukları blok grupları bulunabilir.



Görsel 2.52: Boşluk tuşuna basıldığında çalışacak kodlar

yüzün ait olduğu kişinin adı kullanılmıştır.) olarak eklenir. Kullanıcıya profilin kaydedildiğine dair yazılı ve sesli mesaj verilir. Son olarak sahnedeki kamera görüntüsü kapatılır.



Görsel 2.53: Boşluk tuşuna basıldığında çalışacak kodlar

Bu aşamada kullanıcının sisteme yüzünü tanıtmaya istenmektedir. TM uygulamalarında olduğu gibi tanıtılan yüz örneği bir sınıf olarak kaydedilmektedir.

Turn on video on stage... bloku kullanılarak yüz algılama için kamera görüntüsü sahnede açılır. Kullanıcının hazırlanması için 3 saniye beklemeden sonra kullanıcıya görüntünün analiz edildiğine dair yazılı ve sesli mesaj verilmektedir. Daha sonra görüntü 1 numaralı sınıfa "Murat" (Burada

Ball Kuklasına Ait Kodlar

Zile basıldığında Tobi kuklasına haber vermek zilin görünümünü değiştirmek için kuklaya kod blokları eklenir (Görsel 2.53).

Kapı zili haberi alındığında altındaki kodlar



Görsel 2.54: Kapı zili haberi alındığında çalışacak kodlar

Eşleşme sağlanamazsa gerekli uyarı mesajları kullanıcıya yazılı ve sözlü olarak iletilir ve sahnedeki kamera görüntüsü kapatılır.

Bir yüz profili kaydedildikten sonra kapıdaki kişinin yüzünün eşleşip eşleşmediğini kontrol eden kodların bulunduğu bölümdür.

Ballon kuklasına tıkladığında kapı zili haberi salınmakta ve buradaki kodların çalışmasını tetiklemektedir.

Bu bölümde kullanıcıya kameraya bakmasını söyleyen yazılı ve sesli mesajlar iletilmektedir.

Sahnede kamera açılarak kullanıcının yüzünün görüntüsü sahneye yansıtılır.

Görüntü do face matching on camera bloğu kameradan alınan görüntü ile yüz eşleştirme işlemi yapan face detection eklentisine ait bir bloktur.

Görüntü kayıtlı sınıflarda bulunan örneklerle karşılaştırır.

Profil kaydı yapılırken yüz örneği 1 sınıfa eklendiğinden kameradaki görüntünün bu sınıfla eşleşip eşleşmediği görüntü eğer is 1 class detected bloku ile kontrol edilir.

Eşleşme sağlandığında eve giriş yapıldığını göstermek için dekor değiştirilir.



PictoBlox 6. Uygulama: Yüz Filtreleri

Özellikle akıllı telefonlar ve tabletlerdeki bazı uygulamalar kullanıcıların yüzlerine filtre uygulamalarına olanak verir.



Bu çalışmada PictoBlox ile kullanıcının gözlük, bıyık, sakal gibi filtreler uygulamasını sağlayan bir uygulama geliştirilmektedir.

Ön Hazırlık

Bu uygulamadaki kodlara ve dosyalara bölüm başında yer alan karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

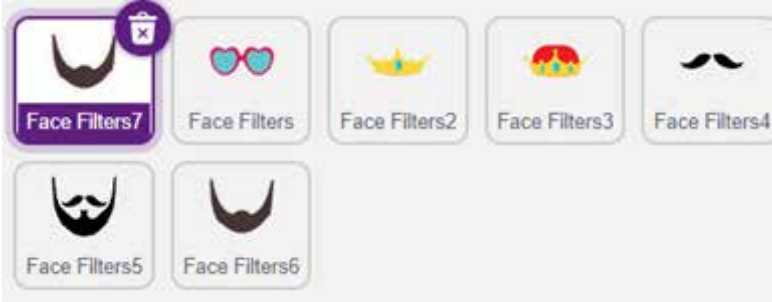
Bilgisayarlı görü (yüz tanıma)

PictoBlox Eklentileri

Face detection

Uygulama Adımları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede Tobi kuklası silinir. Face filters kuklası eklenir ve çoğaltılarak her bir kukla için kuklaya ait farklı kostümler seçilir. (Görsel 2.55).
2. Kuklalar yerleştirilir. Dekora düzenleme yapılarak yazı yazılır ve sahne ayarlanır.



Görsel 2.55: Kuklalar

Kullanıcı 6 filtreden herhangi birine tıkladığında (varsayılanı bıyık olarak gelen) filtre değiştirerek ekranda yüze uygun bir konumda konumlanacaktır (Görsel 2.56).

3. Eklenti ekleme bölümünden face detection yüklenir.



Görsel 2.56: Sahne

4. Varsayılan Face Filters4 (Bıyık) kodlanması

Yeşil bayrak tıkladığında çalışacak kodlar

Turn on video on stage... bloku ile yüz algılama için kamera görüntüsü sahnede açılır.

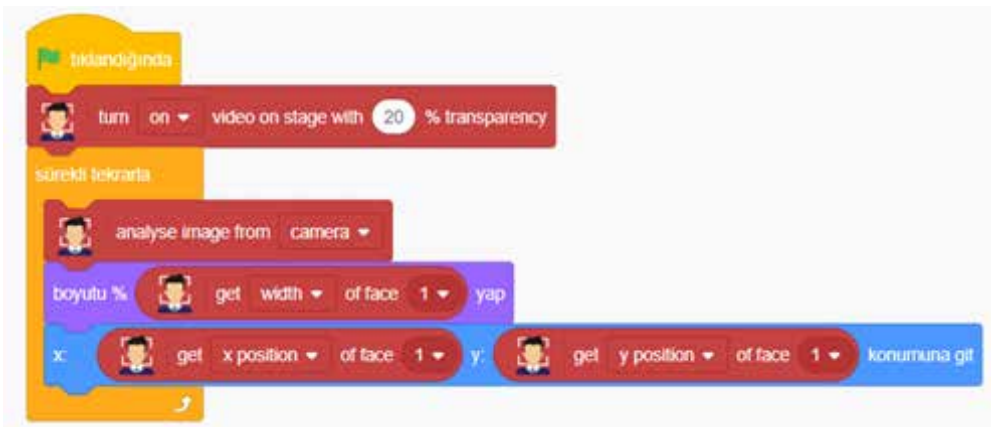
Sürekli tekrarla bloku içinde şu kodlar tekrarlanır (Görsel 2.57).

Kameradan alınan görüntüdeki yüzler analiz edilir.

Kuklanın boyutu görüntüde tespit edilen yüzün genişliğiyle oranlanır. Bu sayede kullanıcının kameraya olan uzaklığına göre filtrenin yüze uygun büyüklükte olması sağlanır.

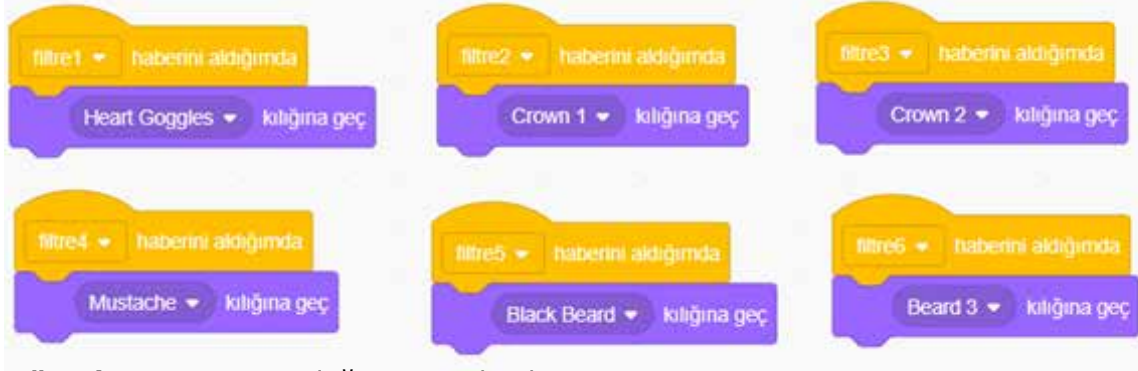
Bir kişi üzerinde filtre uygulanacağı için kameradan tespit edilen 1 numaralı yüz(of face 1) belirtilmiştir.

Sonrasında yüzün konumuna uygun bir şekilde yerleşmesi için kuklanın konumu ayarlanır.



Görsel 2.57: Yeşil bayrak tıkladığında çalışacak kod blokları

Kuklalar tıkladığında gelen haberlere göre kostümün değişmesi için şu kodlar kullanılır (Görsel 2.58).



Görsel 2.58: Kostüm değiştirme işlemleri

Heart goggles: Kalpli gözlük

Crown1: Taç1

Crown2: Taç2

Mustache: Bıyık

Black Beard: Siyah sakal

Beard 3: Sakal

- 5. Diğer kuklaların kodları:** Kullanıcı filtrelere tıkladığında bu durumu varsayılan filtreye (bıyık) bildirmek ve sahnedeki büyüklüklerini ayarlamak için şu kodlar kullanılır (Görsel 2.59, Görsel 2.60, Görsel 2.61, Görsel 2.62, Görsel 3.63 ve Görsel 2.64).



Görsel 2.59: Gözlük kuklası



Görsel 2.60: Taç 1



Görsel 2.61: Taç 2



Görsel 2.62: Bıyık kuklası



Görsel 2.63: Sakal 1



Görsel 2.64: Sakal 2

6. Uygulamanın çalıştırılması

Kodlama yapıldıktan sonra yeşil bayrağa tıklanarak uygulama çalıştırılır.

Kamera açılır ve sahneye kamera görüntüsü gelir.

Varsayılan olarak bıyık filtresi yüze uygun bir şekilde konumlanır.

Kullanıcı başka filtrelere tıkladığında seçtiği filtreler yüze uygulanır.

Kameradaki görüntünün konumu değiştiğinde filtre de uygun şekilde konumlanır.

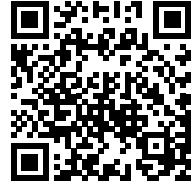
Kullanıcı kameraya yaklaştığında veya uzaklaştığında filtrenin boyutu uygun bir şekilde büyür ya da küçülür (Görsel 2.65).



Görsel 2.65: Uygulamanın çalıştırılması



PictoBlox 7. Uygulama: Duygu Durumu, Yaş ve Cinsiyet Analizi



Bu çalışmada PictoBlox ile kullanıcıların yüz ifadelerinden duygu durumlarını analiz eden bir yapay zekâ uygulaması geliştirilmektedir.

Ön Hazırlık

Bu uygulamadaki eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekir. Bu nedenle çalışmalara başlamadan önce cihazın internete bağlı olması ve PictoBlox da bir hesap oluşturulması önerilir. Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri
Bilgisayarlı görü (yüz tanıma)
Doğal Dil İşleme

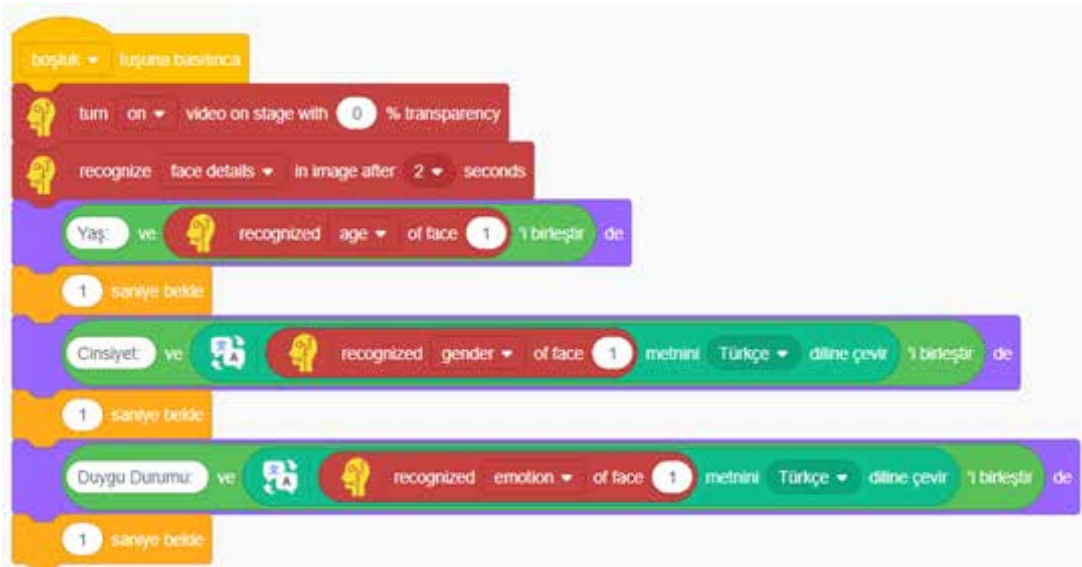
PictoBlox Eklentileri

Artificial Intelligence
Translate

Uygulama Adımları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede Tobi kuklası görünür.
2. Eklenti ekleme bölümünden **artificial intelligence** ve **çeviri** eklentileri yüklenir.

3. Tobi kuklasının kodlanması (Görsel 2.66)



Görsel 2.66: Boşluk tuşuna basıldığında çalışacak kodlar

Age: yaş

Gender: Cinsiyet

Emotion: Duygu durumu

Kameradan yüz detayları taranır.

Yapay zekâ eklentisinin özellikleri kullanılarak analiz edilen yüzün yaş, cinsiyet ve duygu durumu ekranda gösterilmektedir.

Analiz sonuçları İngilizce olarak geldiği için çeviri eklentisi kullanılarak Türkçeye çevrilmiştir.

4. Uygulamanın çalıştırılması



Görsel 2.67: Uygulamanın çalıştırılması

Boşluk tuşuna basılarak uygulama çalıştırıldığında sahnede kamera görüntüsü almak için bir tanıma penceresi açılır.

Bu pencereye kullanıcı kendi yüzünü veya bir resimden bir görüntü gösterebilir.

Daha sonra yüz detayları analiz edilerek görüntüdeki kişinin yaşı, cinsiyeti ve duygu durumu tahmin edilerek ekranda görüntülenir.

Çıktı (Görsel 2.67)

Yaş: 43

Cinsiyet: Erkek

Duygu durumu: Normal



PictoBlox 8.Uygulama: El Hareketi Takip

Bu çalışmada PictoBlox kullanılarak el hareketi takip eden bir yapay zekâ uygulaması geliştirilmektedir.

El hareketi takibi oyundaki kuklayı hareket ettirmek, el işaretlerinin çözümlenmesi ve işaret diliyle iletişim

kurmak gibi uygulamalar geliştirmek için kullanılabilir.



Ön Hazırlık

Bu uygulamadaki eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekir. Bu nedenle çalışmalara başlamadan önce cihazın internete bağlı olması ve PictoBlox'da bir hesap oluşturulması önerilir. Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

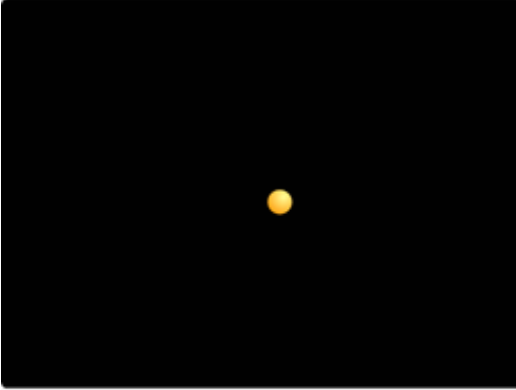
Bilgisayarlı görü (insan vücudu tanıma)

PictoBlox Eklentileri

Human body detection

Uygulama Adımları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede Tobi kuklası görünecektir.

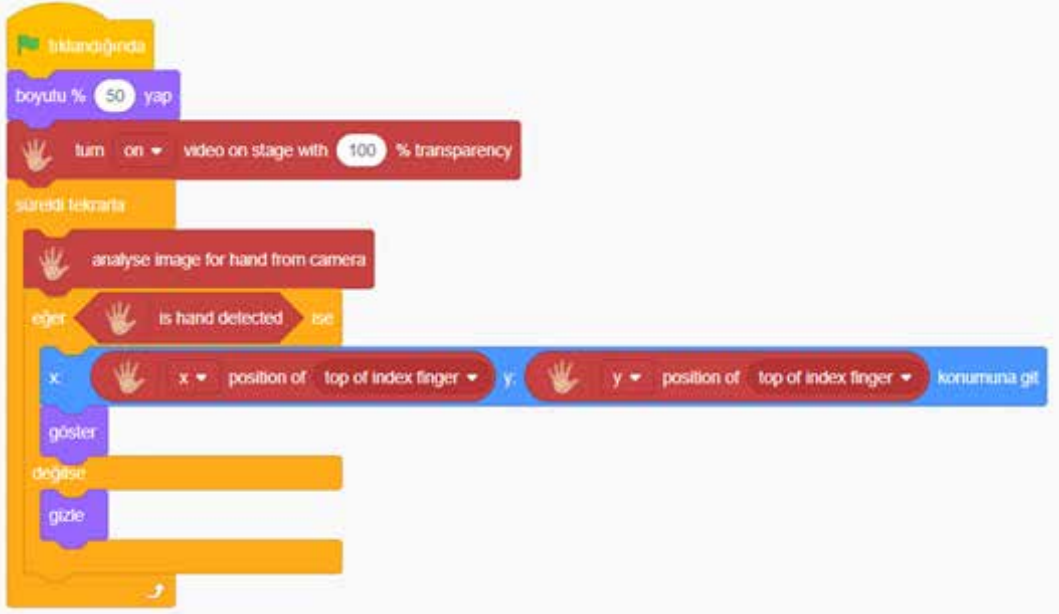


Tobi kuklası silinerek veya gizlenerek sahneden kaldırılır. Ball kuklası eklenir ve sahne arka planı dekorlar bölümünden siyah olarak ayarlanır (Görsel 2.68).

2. Eklenti ekleme bölümünden Human body detection eklentisi yüklenir.

3. Ball kuklasının kodlanması (Görsel 2.69)

Görsel 2.68: Sahne ve kukla ayarları



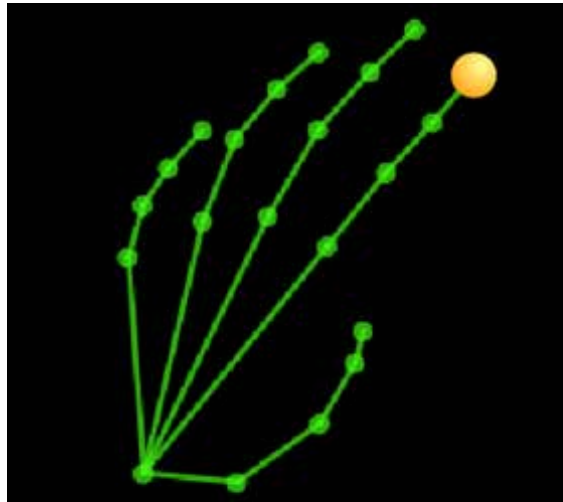
Görsel 2.69: Yeşil bayrak tıklandığında çalışacak kodlar

Kuklanın boyutu %50 olarak ayarlanır. Kamera görüntüsü sahneye aktarılır. Görüntü tamamen saydam (100 % transparency) yapılarak sahnenin siyah bir şekilde görünmesi sağlanır.

Human body detection eklentisinde yer alan **analyse image for hand from camera** bloku kameradaki görüntüde bulunan eli tespit etmek için kullanılır. Eğer bir el tespit edilmişse (eğer is hand detected) Ball kuklası işaret parmağının ucunun (top of index finger) bulunduğu konuma gider. Eğer bir el tespit edilmezse **Ball** kuklası gizlenir. Böylece top kameradaki elin hareketleri takip ederek sürekli işaret parmağının ucunda bulunur.

4. Uygulamanın çalıştırılması

Yeşil bayrağa tıklanarak uygulama çalıştırılır. El hareket ettirilerek uygulama test edilir (Görsel 2.70).



Görsel 2.70: Kameradan el tespiti



PictoBlox 9.Uygulama: Farklı Bir Oyun

Bu çalışmada PictoBlox kullanılarak burun konumuna göre bir kukla hareket ettirilmektedir.



Ön Hazırlık

Bu uygulamadaki eklentileri kullanmak için internete bağlı olmak ve uygulamada oturum açmak gerekir. Bu nedenle çalışmalara başlamadan önce cihazını internete bağlı olması ve PictoBlox'da bir hesap oluşturulması önerilir. Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

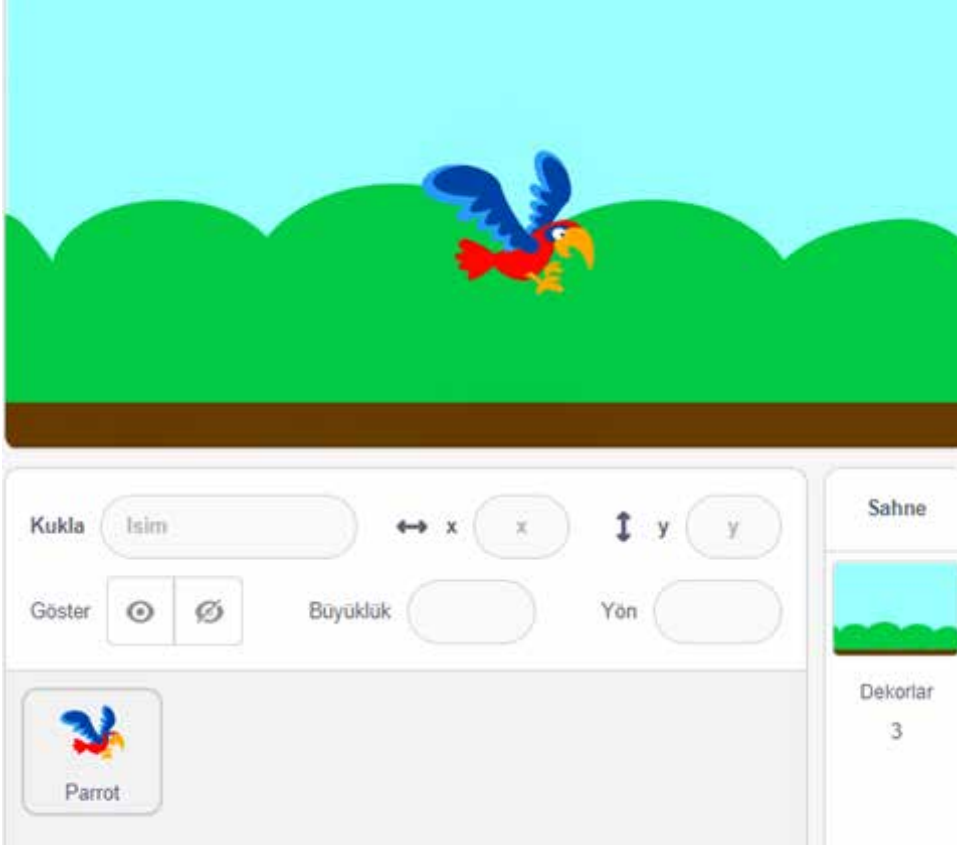
Bilgisayarlı görü (insan vücudu tanıma)

PictoBlox Eklentileri

Human body detection

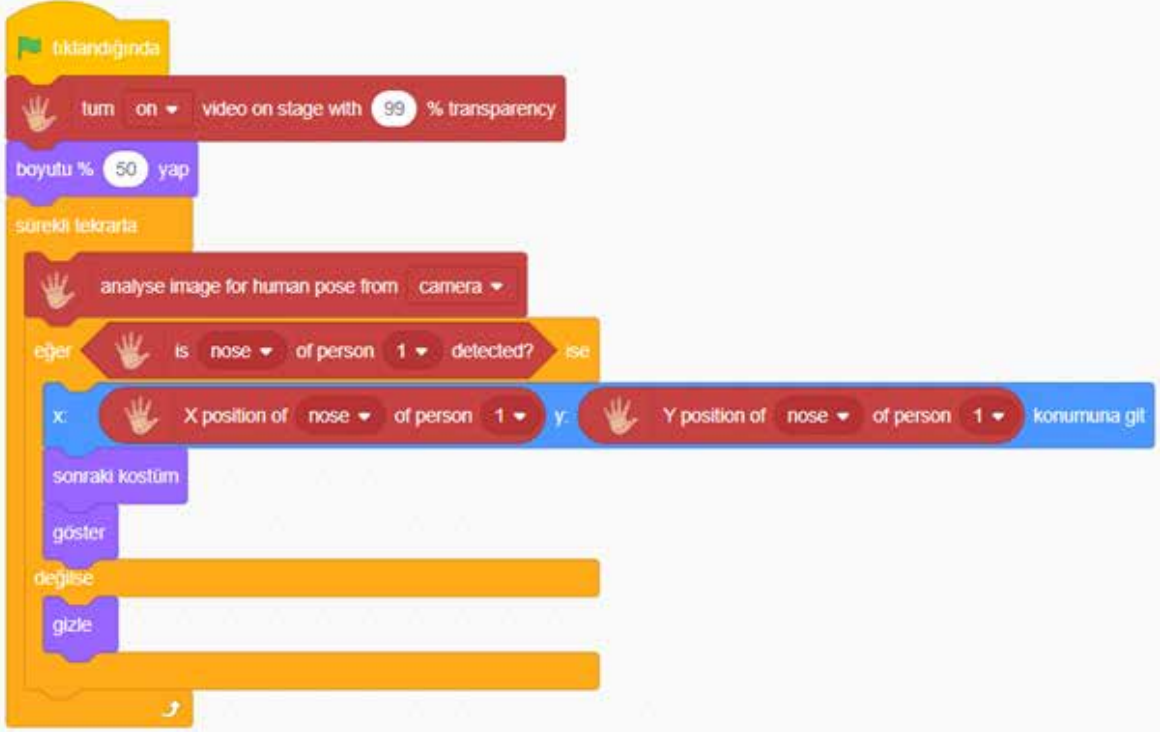
Uygulama Adımları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede Tobi kuklası görünecektir. Tobi kuklası silinerek veya gizlenerek sahneden kaldırılır. Parrot kuklası eklenir ve sahne arka planı dekorlar bölümünden ayarlanır (Görsel 2.71).



Görsel 2.71: Sahne ve kukla ayarları

2. Eklenti ekleme bölümünden Human body detection eklentisi yüklenir.
3. Parrot kuklasının kodlanması (Görsel 2.72)



Görsel 2.72: Yeşil bayrak tıklandığında çalışacak kodlar

Kamera görüntüsü sahneye aktarılır.

Görüntü tamamen saydam (99 % transparency) yapılarak sahnenin siyah bir şekilde görünmesi sağlanır. Kuklanın boyutu ayarlanır.

Human body detection eklentisindeki **analyze image for human pose from camera** bloku kameradaki görüntüde bulunan burnu tespit etmek için kullanılmaktadır.

Eğer burun tespit edilmişse (eğer is nose of person 1 detected) Parrot kuklası kişinin burnunun bulunduğu konuma (position of nose of person) gitmektedir.

Eğer bir burun tespit edilmezse **Parrot** kuklası gizlenmektedir.

Böylece papağan kameradaki burnu takip eder.

4. Uygulamanın çalıştırılması

Yeşil bayrağa tıklanarak uygulama çalıştırılır.

Kullanıcı kamerada yüzünü ve dolayısıyla burnunu sağa/sola, yukarı/aşağı hareket ettirerek uygulamayı test eder. Parrot sürekli olarak kullanıcının burnunun konumuna gider.



PictoBlox 10.Uygulama: Havaya Çizim Yapma

Bu çalışmada PictoBlox'da yapay zekâ eklentileri kullanılarak el hareketleriyle çizim yapmayı sağlayan bir uygulama geliştirilmektedir.



Ön Hazırlık

Bu uygulamadaki eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekir. Bu nedenle çalışmalara başlamadan önce cihazını internete bağlı olması ve PictoBlox'da bir hesap oluşturulması önerilir.

Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

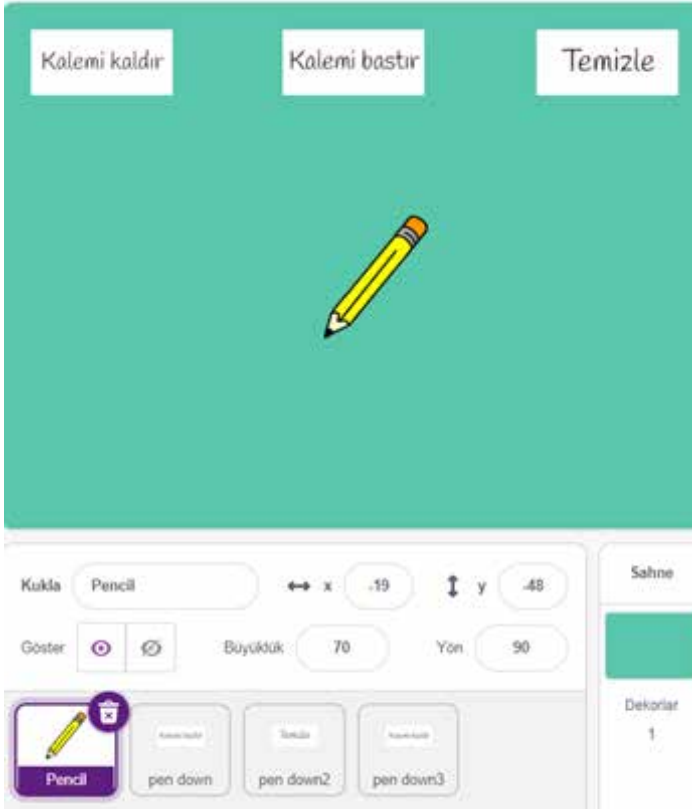
Bilgisayarlı görü (İnsan vücudu tanıma)

PictoBlox Eklentileri

Human body

Kalem

Uygulama Adımları



Görsel 2.73: Sahne ve kukla ayarları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede **Tobi** kuklası görünecektir. **Tobi** kuklası silinerek veya gizlenerek sahneden kaldırılır. **Pencil, kalem indir, kalem kaldır ve temizle** kuklaları eklenir/ oluşturulur. Sahne arka planı dekor seçilerek yeşil renk olarak ayarlanır (Görsel 2.73).

2. Eklenti ekleme bölümünden

3. **Human body detection** ve **Kalem** eklentileri yüklenir.

4. Kalem Kuklasının Kodlanması (Görsel 2.74)

Kamera görüntüsü sahneye aktarılır.



Görsel 2.74: Yeşil bayrak tıklandığında çalışan kodlar (kalem kuklası)

Görüntü tamamen saydam (100 % transparency) yapılarak sahnenin yeşil bir şekilde görünmesi sağlanmıştır.

Kuklanın boyutu kukla ayarlarında %70 olarak belirlenir.

Human body detection eklentisindeki **analyse image for hand from camera** bloku kameradaki görüntüde bulunan eli tespit etmek için kullanılır.

Eğer bir el tespit edilmişse (eğer is hand detected) kalem kuklası işaret parmağının ucunun (top of index finger) bulunduğu konuma gider.

Eğer bir el tespit edilmezse kalem kuklası gizlenmektedir. Böylece kalem kameradaki elin hareketlerini takip ederek sürekli olarak işaret parmağının ucunda bulunur.

Kalem bastır butonu (Pen down kuklası) kalemi bastır haberini salarak kalemin sahneye çizim yapmasını sağlar (Görsel 2.75).



Görsel 2.75: Pen down kuklası

Kalem kaldır butonu (Pen down 2 kuklası) kalemi kaldır haberini salarak kalemin çizim yapmasını engellemek için zeminden kaldırır (Görsel 2.76).



Görsel 2.76: Pen down2 kuklası

Temizle butonu (Pen down 3 kuklası) temizle haberini salarak sahnedeki çizimlerin silinmesini sağlar (Görsel 2.77).



Görsel 2.77: Pen down 3 kuklası

5. Uygulamanın çalıştırılması

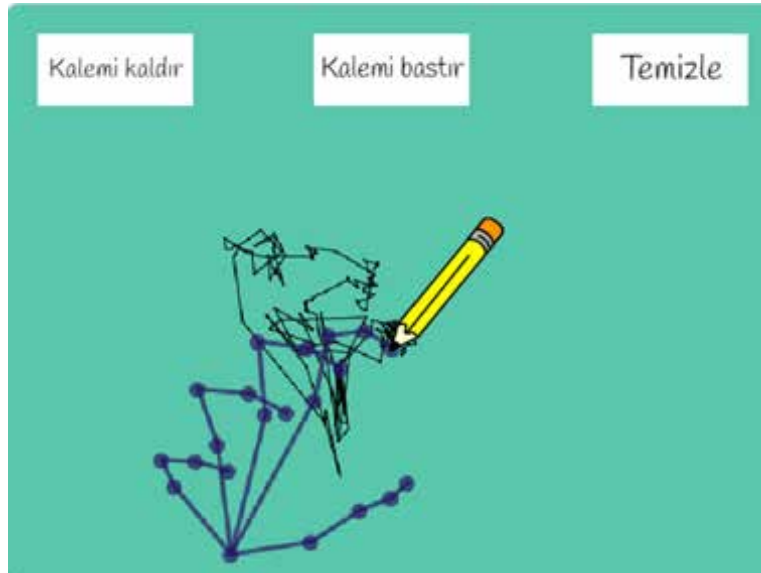
Yeşil bayrağa tıklanarak uygulama çalıştırılır.

Kullanıcı elini hareket ettirdiğinde kalem sahnede işaret parmağının konumunu takip eder. Kullanıcı fareyle **kalemi bastır** butonuna tıkladığında işaret parmağının hareketleriyle çizim yapar.

Kullanıcı **kalemi kaldır** butonuna tıklayarak çizime ara verebilir.

Temizle butonu ise sahneyi temizlemek için kullanılır

Çıktı (Görsel: 2.78)



Görsel 2.78: Uygulamanın çalıştırılması



11. Uygulama: Maske Kontrol Sistemi

COVID-19 salgınında en çok yapılan yapay zekâ uygulamalarından biri de "maske algılama" uygulamalarıdır. Bu çalışmada PictoBlox'da yapay zekâ eklentileri kullanılarak görüntüdeki kişinin maske takip takmadığını algılayan bir uygulama geliştirilmektedir.



Ön Hazırlık

Bu uygulamadaki eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekir. Bu nedenle çalışmaya başlamadan önce cihazın internete bağlı olması ve PictoBlox'da bir hesap oluşturulmalıdır.

Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

Bilgisayarlı görü (yüz tanıma)

Ses tanıma

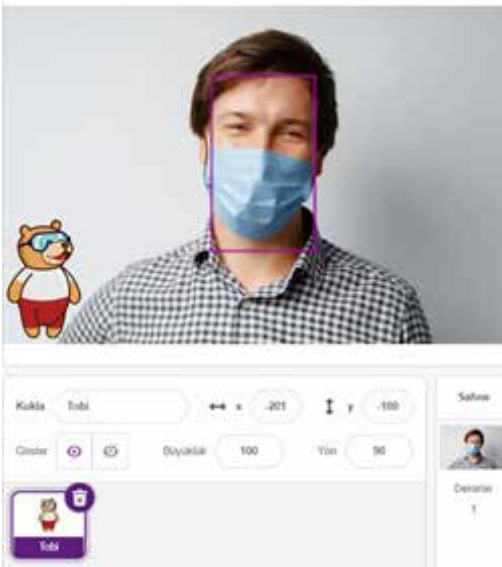
PictoBlox Eklentileri

Machine Learning

Text to speech

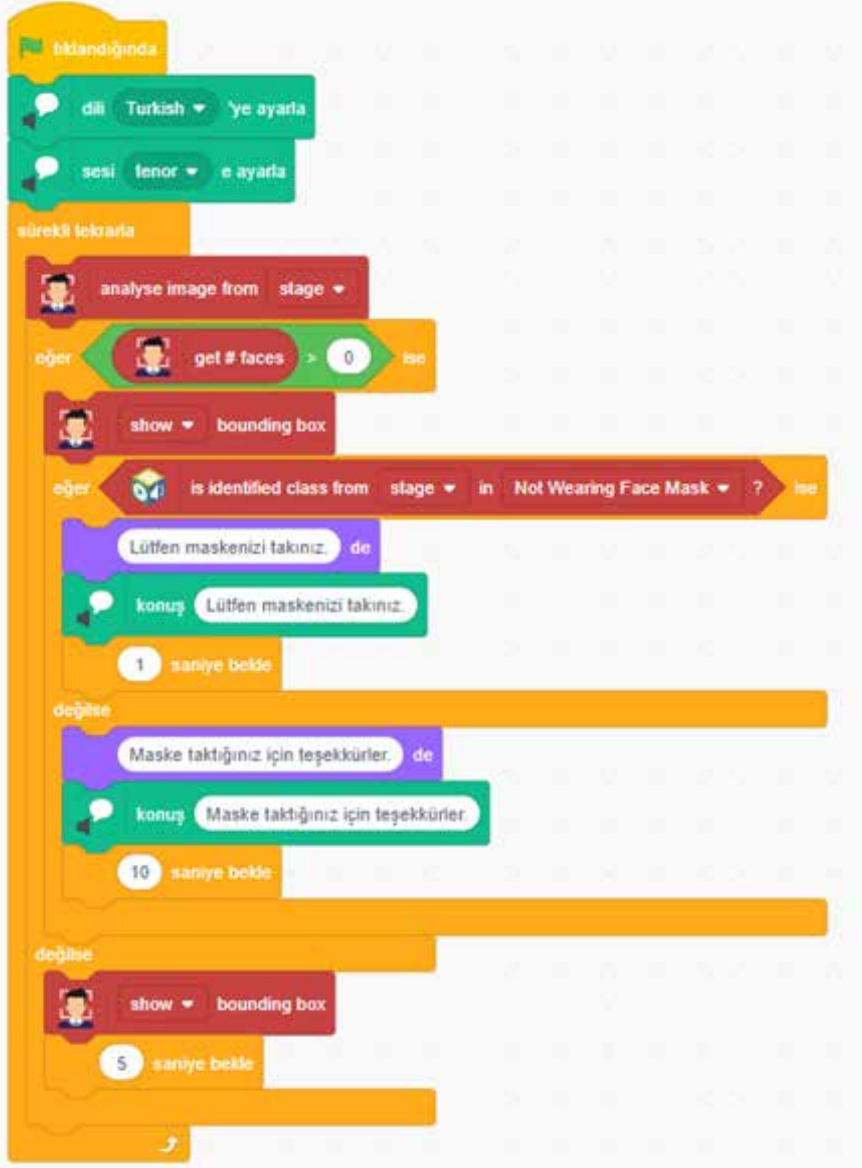
Uygulama Adımları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede **Tobi** kuklası görünür. Sahne dekoruna maskeli bir fotoğraf yüklenir. Sahne ve kukla ayarlanır (Görsel 2.79). Eklenti ekleme bölümünden machine learning ve text to speech eklentileri yüklenir.



Görsel 2.79: Sahne ve kukla ayarları

2. Tobi kuklasının kodlanması (Görsel 2.80)



Görsel 2.80: Yeşil bayrak tıklandığına çalışacak kodlar

Machine Learning eklentisinde yüze maske takılıp takılmadığını veya takılan maske türünü algılayan kod blokları bulunmaktadır.

Not Wearing Face Mask: Yüze maske takılmamış.

Wearing Face Mask: Yüze maske takılmış.

Not Wearing Face Mask Properly: Maske düzgün takılmamış.

Using 3 Ply Mask: Üç katlı tıbbi maske kullanılıyor.

Using 4 Ply Mask: Dört katlı tıbbi maske kullanılıyor.

Uygulamaya sesli uyarı özelliği eklemek için metinden sese eklentisi kullanılmıştır.

Dil Türkçe ve ses tonu tenor olarak ayarlanır.

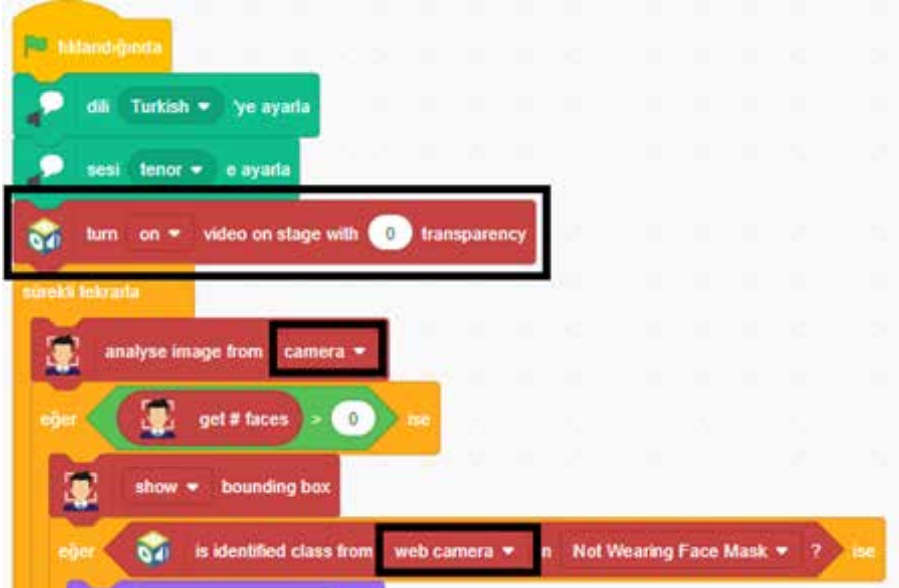
Farklı ses tonları da (alto, ciyak, dev ve kedi yavrusu) tercih edilebilir.

Uygulamada sahnedeki görüntüde maske olup olmadığı algılanmaktadır.

En az bir yüz tespit edildiğinde (get#faces) kutu (show bounding box) gösterilir.

Eğer maske takılmamışsa sesli ve yazılı uyarı mesajı, takılmışsa teşekkür mesajı verilmektedir.

Eğer kameradan eş zamanlı algılama yaptırılmak istenirse birkaç kod bloku değiştirilerek veya eklenerek yapılabilir (Görsel 2.81).



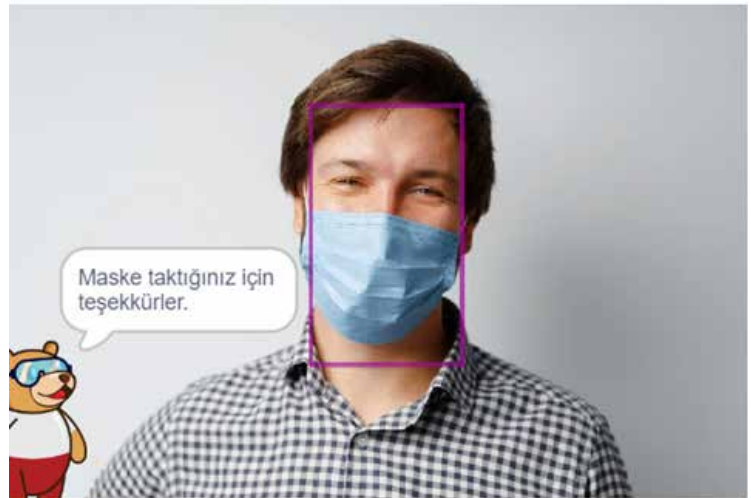
Görsel 2.81: Kameradan algılama yapmak için değiştirilecek kodlar

3. Uygulamanın çalıştırılması

Yeşil bayrağa tıklanarak uygulama çalıştırılır.

Uygulama çalıştırıldığında sahnedeki görüntüden maske algılanarak sesli ve yazılı olarak mesaj verilir. Eğer kameradan görüntü alacak şekilde ayarlanırsa kameradaki görüntüde maske algılaması yapılır.

Çıktı (Görsel 2.82)



Görsel 2.82: Uygulamanın çalıştırılması



PictoBlox 12. Uygulama: Sanal Hava Durumu Sunucusu

Bu çalışmada bir yere ait hava durumunu (sıcaklık, nem, rüzgâr vb.) sanal bir sunucu tarafından sesli olarak veren bir uygulama geliştirilmektedir.



Ön Hazırlık

Bu uygulamadaki eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekir. Bu nedenle çalışmalara başlamadan önce cihazın internete bağlı olması ve PictoBlox'da bir hesap oluşturulması önerilir.

Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

IoT (Nesnelerin interneti)

Ses tanıma

PictoBlox Eklentileri

Weather data

Text to speech

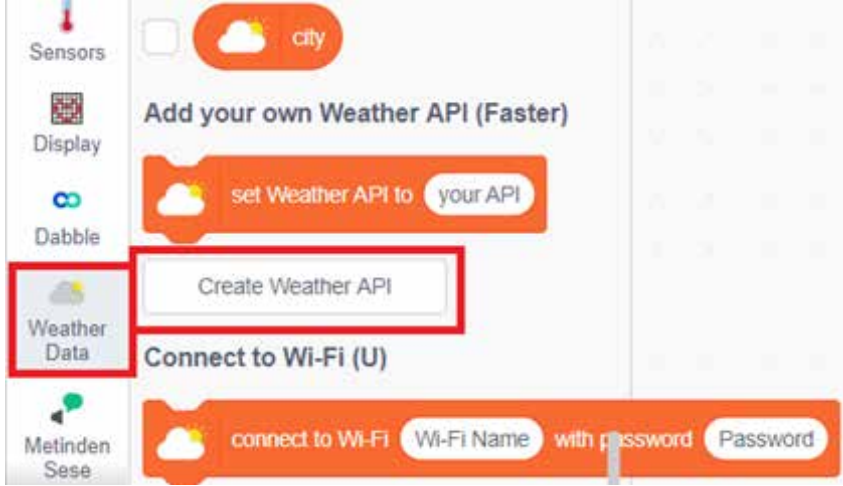
Uygulama Adımları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede Tobi kuklası görünür. Sahne arka planı olarak uygun bir dekor yüklenir. Sahne ve kukla ayarlanır (Görsel 2.83).



Görsel 2.83: Sahne ve kukla ayarları

2. Eklenti ekleme bölümünden weather data ve metinden sese eklentileri yüklenir.
3. Weather data eklentisi üzerinde bir API (create Weather API) oluşturulmalıdır. Yeni API oluşturmak için <https://openweathermap.org/api/> sayfasında hesap açılması gerekmektedir.
4. API (Uygulama Programlama Arabirimi) ile servisi ücretsiz kullanmak için bir erişim anahtarı oluşturulur (Görsel 2.84).



Görsel 2.84: Weather Data eklentisinin ayarlanması

Hava durumu verilmek istenen yerin enlem ve boylam bilgileri Google haritalar kullanılarak alınabilir.

Örnek: Antalya İli Korkuteli ilçesi için 37.0979276, 30.1812335

5. Kuklanın kodlanması (Görsel 2.85)



Görsel 2.85: Boşluk tuşuna basıldığında çalışacak kodlar

Erişim anahtarı girilerek servise erişim yetkisi sağlanır.

Hava durumu öğrenilmek istenen yere ait enlem ve boylam bilgileri girilir.

Sanal sunucuyu konuşturmak için metinden sese eklentisi kullanılarak dil Türkçe olarak ayarlanır.

Sesli ve yazılı mesaj olarak şehir adı, rüzgâr hızı, nem yüzdesi, en düşük ve en yüksek sıcaklık bilgileri verilmektedir.

City: Şehir

Wind speed: Rüzgâr hızı

Humidity: Nem

Temperature max...: En yüksek sıcaklık

Temperature min...: En düşük sıcaklık

6. Uygulamanın alıřtırılması

Yeřil bayraęa tıklanarak uygulama alıřtırılır. Verilen konuma iliřkin hava verisi servis zerinden alınarak sesli ve yazılı olarak kullanıcıya iletilir (Grsel 2.86).

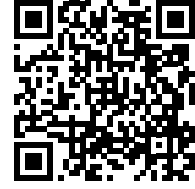


Grsel 2.86: Uygulamanın alıřtırılması



13. Uygulama: Sanal Tercüman

Bu çalışmada bir dilden diğerine sesli ve yazılı çeviri yapabilen bir sanal tercüman uygulaması geliştirilmektedir. Eklenti ile birçok dilde çeviri yapılabilir. Bu örnekte Türkçeden İngilizceye ve Almancaya sesli ve yazılı çeviri yapılmıştır.



Bu uygulamadaki kodlara ve dosyalara bölüm başında yer alan karekod okutularak ulaşılabilir.

Ön Hazırlık

PictoBlox da bazı eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekmektedir. Bu nedenle çalışmalara başlamadan önce cihazın internete bağlı olması ve PictoBlox da bir hesap oluşturulması önerilir.

Kullanılan Yapay Zekâ Teknolojileri

Ses tanıma

Doğal dil işleme

PictoBlox Eklentileri

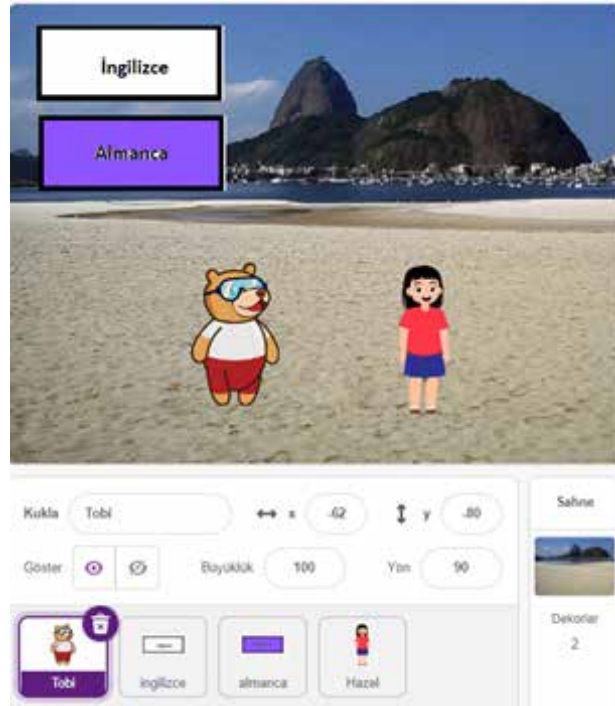
Text to speech

Translate

AI (speech recognition)

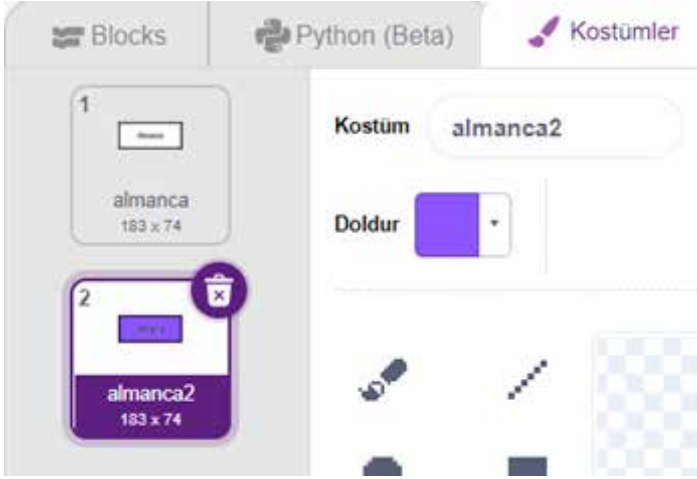
Uygulama Adımları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede Tobi kuklası görünür. Sahne arka planı olarak uygun bir dekor yüklenir. Sahne ve kukla ayarları (Görsel 2.87).



Görsel 2.87: Sahne ve kukla ayarları

2. Dil seçimlerinin yapılabilmesi için buton şeklinde kuklalar tasarlanır ve bunların kostümleri ayarlanır (Görsel 2.88).



Görsel 2.88: Kukla kostüm ayarları

Dil seçimi yapıldığında seçim yapıldığını belirtmek amacıyla buton renginin değişmesi için kostümler oluşturulur.

3. Eklenti ekleme bölümünden **text to speech, translate ve artificial intelligence** eklentileri yüklenir.

AI eklentisi bulut servislerini kullandığı için uygulamada oturum açılmalıdır.

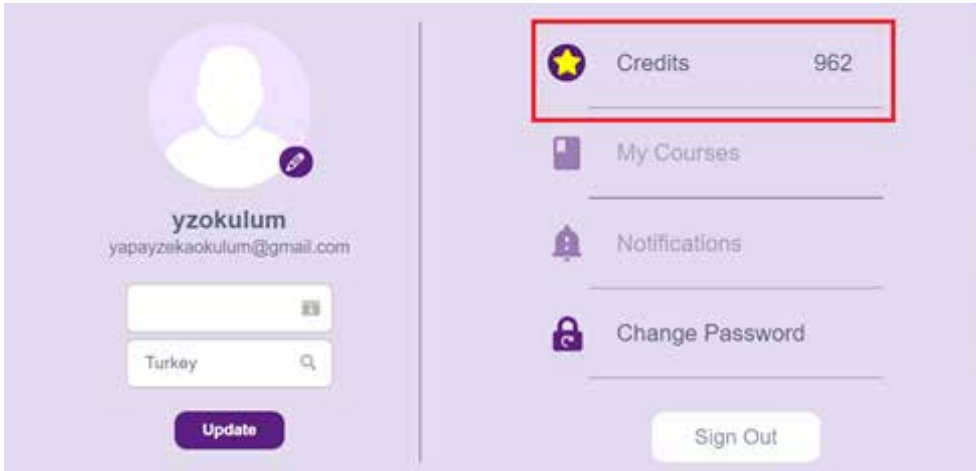
Bunun için eğer hesap oluşturulmamışsa öncelikle hesap oluşturulmalıdır.

Yapay zekâ bulut servisi için PictoBlox tarafından kullanıcılara bulut servislerinde kullanılmak üzere bir kredi tanımlanmaktadır (Görsel 2.89).

Bu kredi bittiğinde ücret ödenerek kredi alınabilir ya da farklı bir hesap oluşturulabilir.

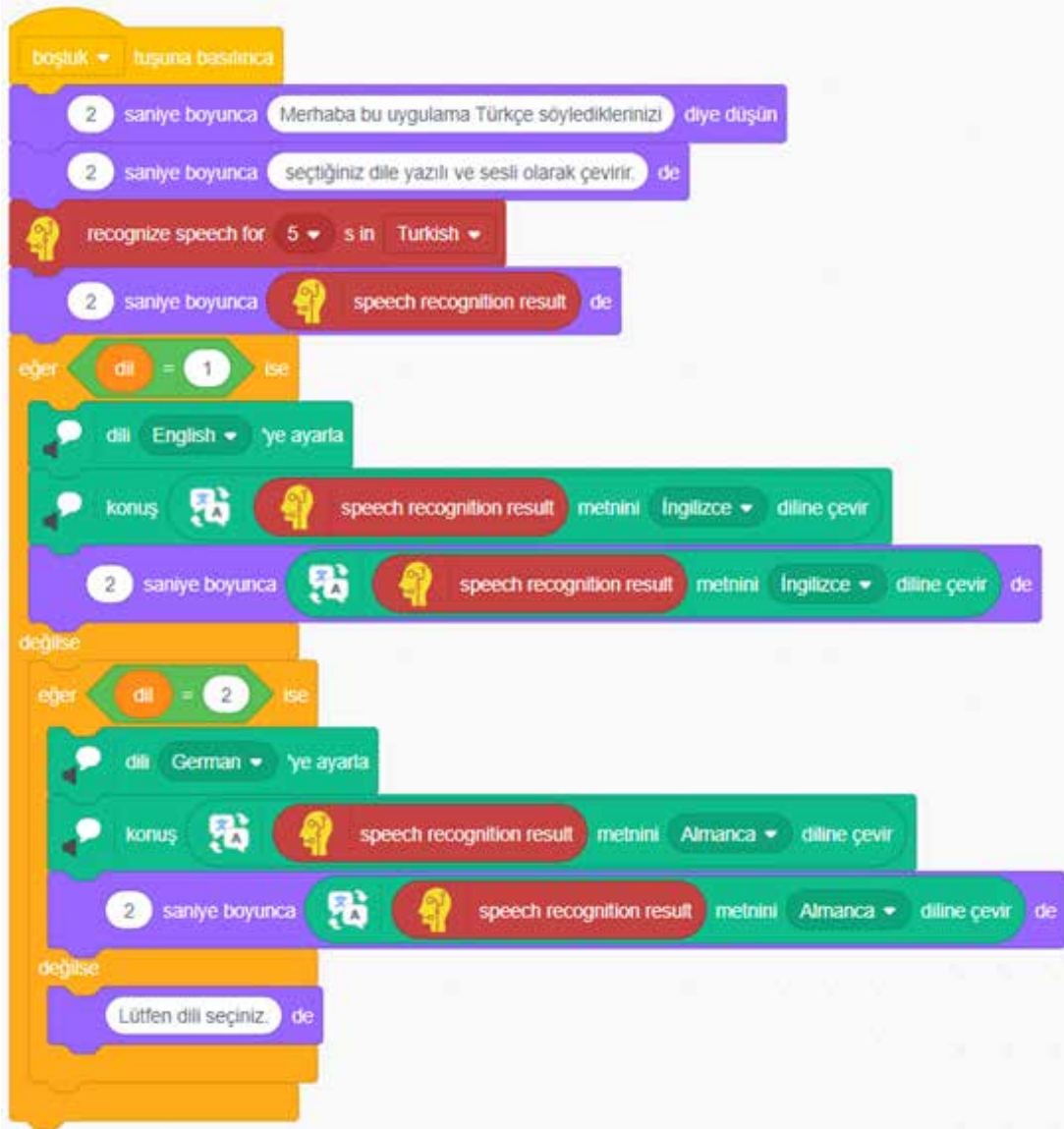
Uygulamanın çalışması boşluk tuşuna basılma olayına bağlanmıştır. Bu sayede uygulamanın sürekli olarak çalışması engellenir ve kredinin bitmesinin önüne geçilir.

Ses tanıma işlemi bulut servis üzerinden yapıldığı için 1-2 saniye gecikmeli olabilir.



Görsel 2.89: Profil kredi bilgisi

4. Tobi kuklasının kodlanması (Görsel 2.90)



Görsel 2.90: Boşluk tuşu basıldığında çalışan kodlar

Uygulamanın kullanıcıya tanıtılması için gerekli mesajlar sesli ve yazılı olarak iletilir. **recognize speech for 5 sn Turkish:** Ortamdaki ses Türkçe dikkate alınarak 5 saniyeliğine dinlenir.

Bu kod çalıştığında ses algılama penceresi açılır. Tanıma sonucu sahnede yazdırılır. Metin tanıma sonucu seçilen dile (dil değişkeni kontrol edilerek) çevir eklentisi kullanılarak çevrilir.

Çevrilen metin sahnede yazdırılır ve **text to speech** eklentisi ile seslendirilir.

5. Diğer kuklaların kodlanması

Dil butonları tıklandığında rengi değiştirme ve dil seçimini değişken üzerinden yapmak için (İngilizce için 1 ve Almanca için 2) gerekli kodlamalar yapılır (Görsel 2.91 ve Görsel 2.92).



Görsel 2.91: İngilizce butonu kodlar



Görsel 2.92: Almanca butonu kodlar

6. Uygulamanın çalıştırılması

Boşluk tuşuna basılarak uygulama çalıştırılır.

Kullanıcı çeviri yapılmasını istediği dili butona tıklayarak seçer.

Ses tanıma penceresi 5 saniye boyunca açılacaktır. Kullanıcı bu sürede konuşur.

Konuşma bulut servis üzerinden metne çevrilir.

Metin; çeviri eklentisiyle seçilen dile çevrilir, metinden sese eklentisi ile Türkçe seslendirilir ve yazılı olarak sahnede görüntülenir.

Çıktı (Görsel 2.93)



Görsel 2.93: Uygulamanın çalıştırılması



14. Uygulama: TM Modeli Kullanma

Bu çalışmada TM ile önceden geliştirilen aslan, kaplan ve kedi modeli kullanılarak verilen hayvan fotoğraflarını tanıyan bir uygulama geliştirilmektedir.



Ön Hazırlık

Bu uygulamadaki eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekir. Bu nedenle çalışmalara başlamadan önce cihazın internete bağlı olması ve PictoBlox'da bir hesap oluşturulması önerilir. Bu uygulamadaki kodlara ve dosyalara karekod okutulularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

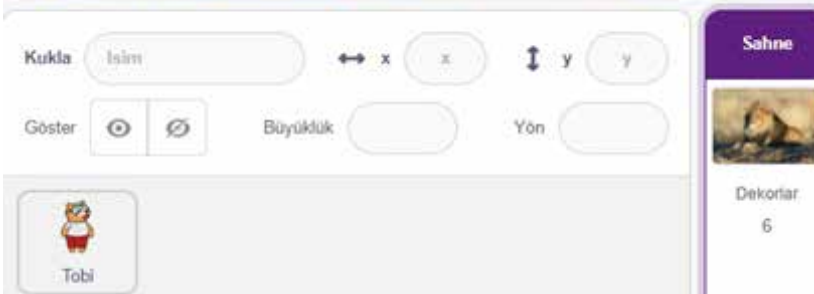
Bilgisayarlı görü

PictoBlox Eklentileri

Machine Learning

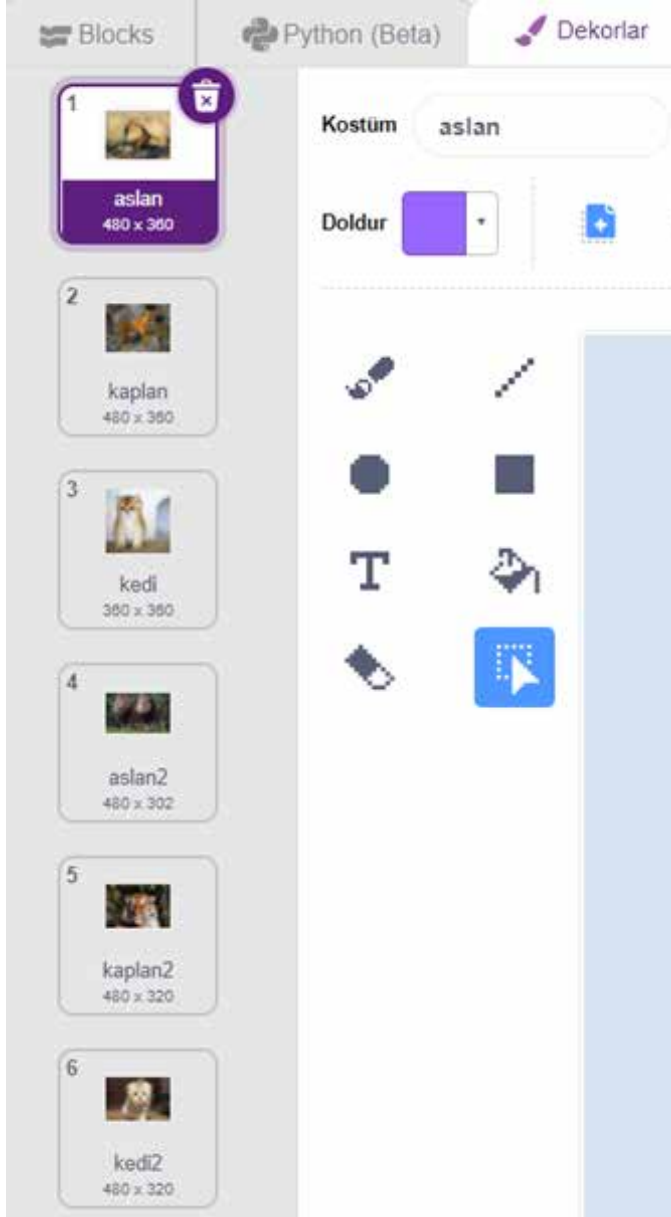
Uygulama Adımları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede Tobi kuklası görünür. Sahne ve kukla ayarlanır (Görsel 2.94).



Görsel 2.94: Sahne ve kukla ayarları

2. Uygulamada sahnedeki görüntülerden algılama yapılacağı için ikişer tane aslan, kaplan ve kedi görseli dekor olarak yüklenir. Toplamda altı dekor olur (Görsel 2.95).



Görsel 2.95: Dekorların yüklenmesi

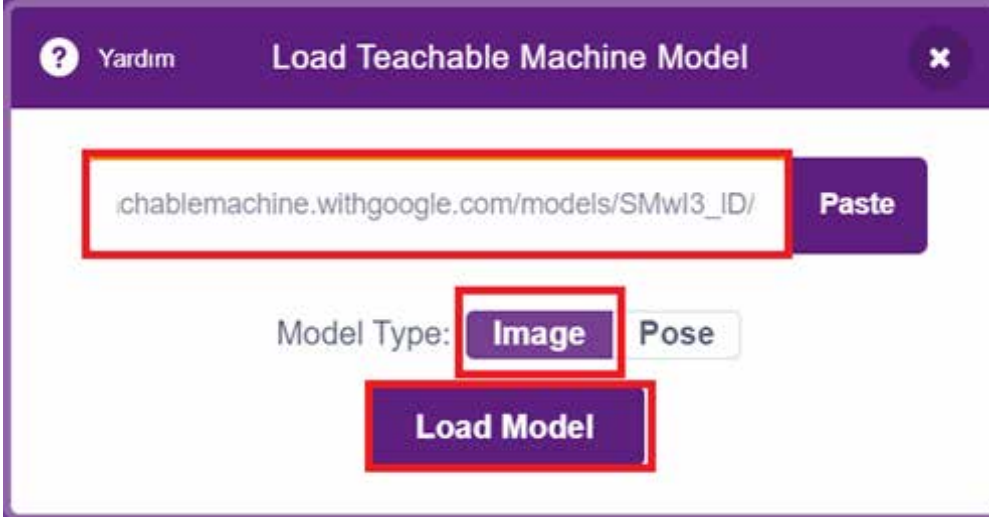


Görsel 2.96: Modelin yüklenmesi

3. Eklenti ekleme bölümünden **machine learning** eklentisi yüklenir. Bu eklenti Google TM üzerinden yeni model oluşturmaya veya oluşturulan modeli kullanmaya olanak verir. Bu uygulamada Google TM’de daha önce oluşturulan aslan, kaplan ve kediyi algılayan model kullanılmaktadır [3]. Modeli kullanmak için **Load a model** seçilir (Görsel 2.96).

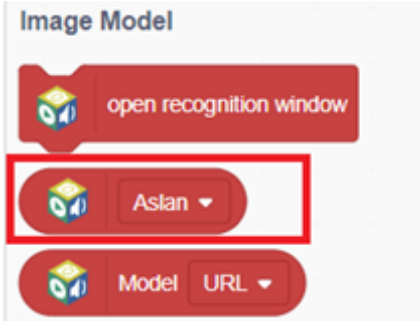
TM kullanılarak yeni model oluşturulmak istendiğinde Create a Model (model oluşturma) kullanılır.

4. **Load model** seçildikten sonra modelin URL'si girilir [3]. Uygulamada kullanılacak Model türü (model type) görüntü olduğu için **Model Type Image** olarak kalmalıdır. Eğer poz modeli kullanılmak istenseydi Pose seçilmesi gerekirdi. **Load model** butonuna tıklanarak model yüklenir (Görsel 2.97).



Görsel 2.97: Modelin yüklenmesi-2

5. Model yüklendikten sonra sınıf etiketleri ile ilgili bloklar gelecektir (Görsel 2.98).



Görsel 2.98: Görsel Model

6. Tobi kuklasının kodlanması (Görsel 2.99)



Görsel 2.99: Yeşil bayrak tıkladığında çalışacak kodlar

Identify class from stage:

Sahnedeki görüntünün ait olduđu sınıfı verir.

Sınıf adı sahnede yazar.

Diđer dekorların da aynı şekilde algılanması için sonraki dekor bloku kullanılmıřtır.

Bu işlemler sürekli tekrarlanarak dekorların tek tek analiz edilmesi sađlanır.

7. Uygulamanın alıřtırılması

Yeřil bayrađa tıklanıđında dekordaki görüntü analiz edilerek sınıf adı tahmin edilir ve sahneye yazdırılır. Bu işlem sürekli olarak tekrarlanır. Arka planda dekor sürekli olarak deđiřir ve yapay zekâ modeli de yeni tahminler yapar.

ıktı (Görsel 2.100)



Görsel 2.100: Uygulamanın alıřtırılması



15. Uygulama: Hayvan Sınıflandırıcı

Bu çalışmada kameradaki hayvan görüntülerini analiz ederek türünü (sınıf) tespit eden bir yapay zekâ uygulaması geliştirilmektedir.



Ön Hazırlık

Bu uygulamadaki eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekir. Bu nedenle çalışmalara başlamadan önce cihazın internete bağlı olması ve PictoBlox'da bir hesap oluşturulması önerilir.

Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

Bilgisayarlı görü

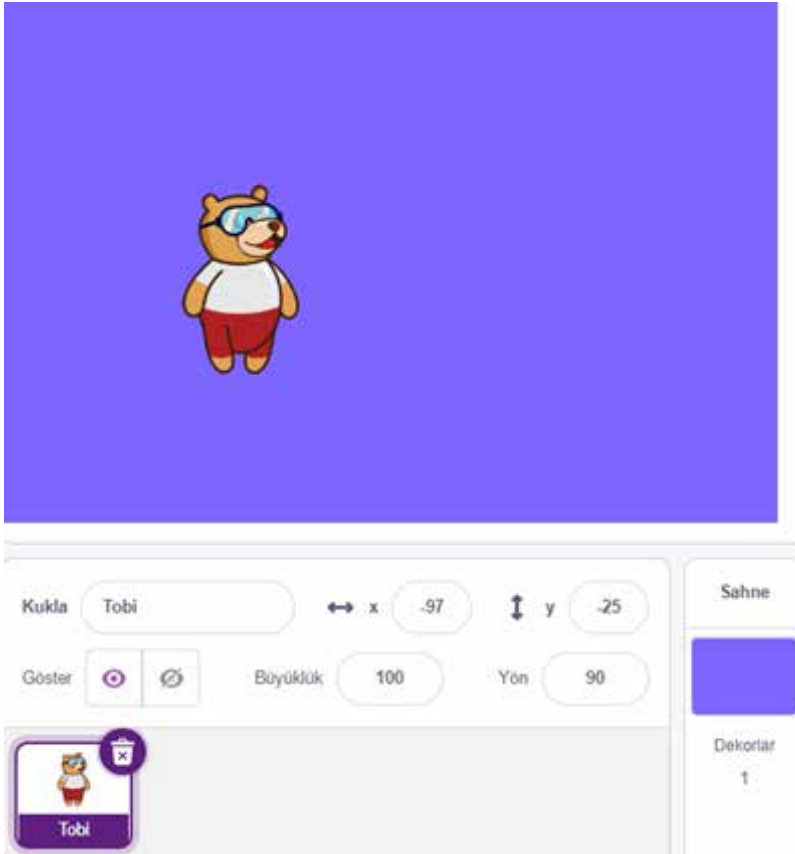
PictoBlox Eklentileri

Machine Learning

Translate

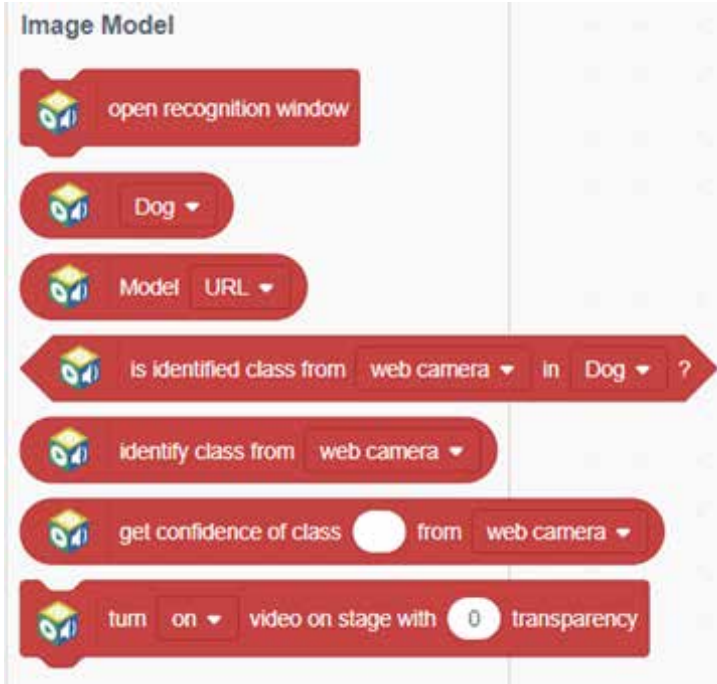
Uygulama Adımları

1. PictoBlox programı açılır. PictoBlox ekranında boş sahnede Tobi kuklası görünür. Sahne ve kukla ayarlanır (Görsel 2.101).



Görsel 2.101: Sahne ve kukla ayarları

2. Eklenti ekleme bölümünden machine learning ve çeviri eklentisi yüklenir. Image model bölümünde varsayılan olarak hayvan sınıflandırıcı bir model gelmektedir (Görsel:2.102).



Görsel 2.102: Görüntü modeli

Bu model 10 farklı hayvanı sınıflandırabilmektedir: Köpek, at, fil, kelebek, tavuk, kedi, inek, koyun, örümcek ve sincap.

3. Tobi kuklasının kodlanması (Görsel 2.103)



Görsel 2.103: Yeşil bayrak tıkladığında çalışacak kodlar

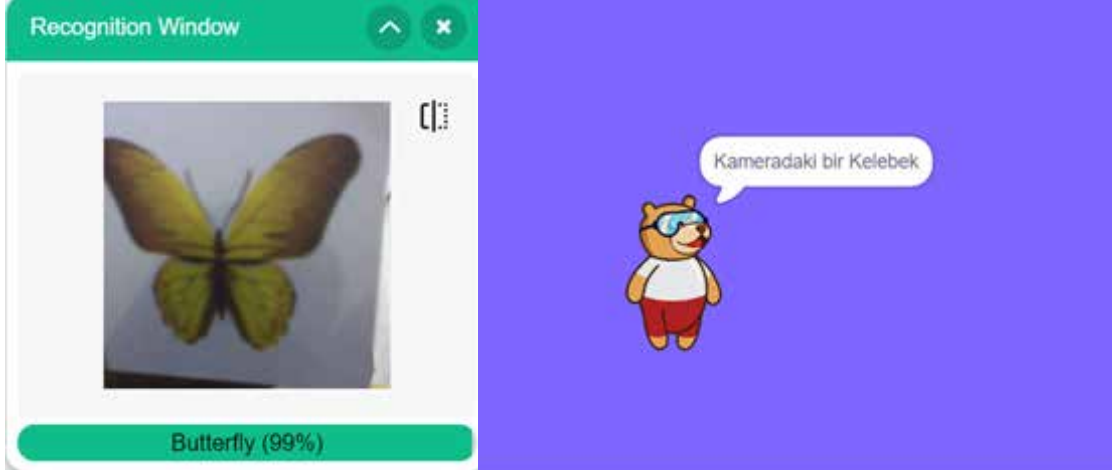
identify class from web camera: Kameradaki görüntünün ait olduğu sınıfı verir.

Sınıf adları İngilizce olduğu için çeviri eklentisi kullanılarak Türkçeye çevrilir ve sahnede yazılır.

4. Uygulamanın alıřtırılması

Yeřil bayraęa tıkladıęında algılama penceresi aılır. Kameraya bir hayvan grseli gsterilir. Kameradaki grnt analiz edilerek sınıfı (rneęin kelebek) sahnede yazdırılır.

ıktı (Grsel 2.104)

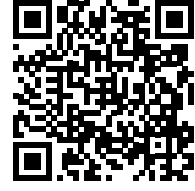


Grsel 2.104: Uygulamanın alıřtırılması



16. Uygulama: Balon Patlatma Oyunu

Bu çalışmada kamerada el hareketleriyle uçan balonları patlattıkça puan alınan bir oyun geliştirilmektedir.



Ön Hazırlık

Bu uygulamadaki eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekir. Bu nedenle çalışmalara başlamadan önce cihazın internete bağlı olması ve PictoBlox'da bir hesap oluşturulması önerilir. Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

Bilgisayarlı görü

PictoBlox Eklentileri

Human Body Detection

Uygulama Adımları

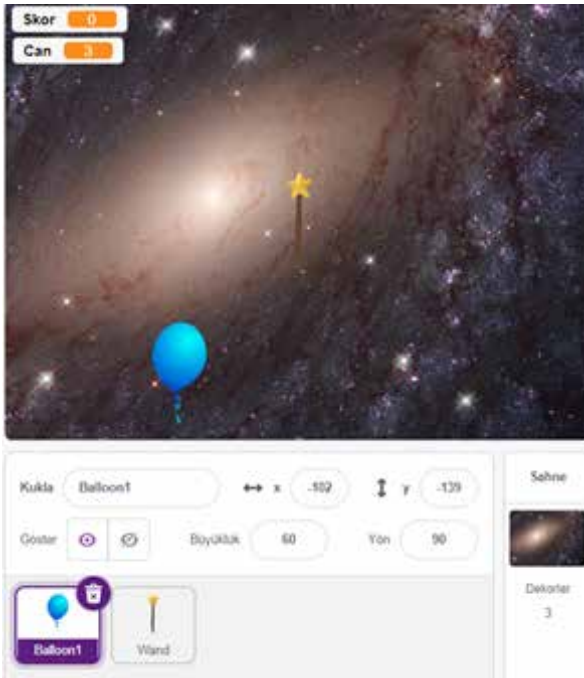
1. PictoBlox programı açılır.

PictoBlox ekranında boş sahnede **Tobi** kuklası görünür.

Tobi kuklası kaldırılır.

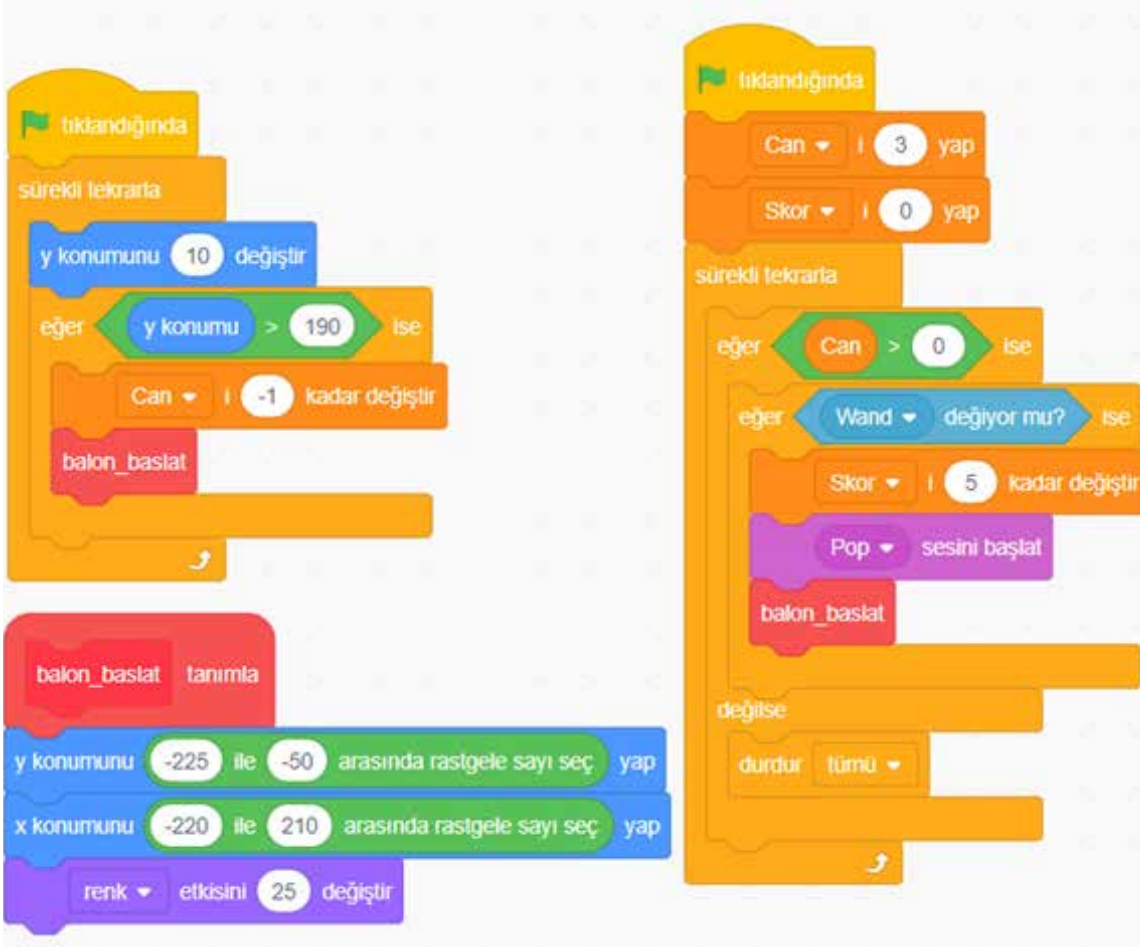
Sahneye oyuna uygun bir dekor eklenir.

Sahneye **Balloon1** ve **Wand** kuklaları eklenerek sahne ve kukla ayarlanır (Görsel 2.105).



Görsel 2.105: Sahne ve kukla ayarları

2. Eklenti ekleme bölümünden **Human body detection** eklentisi yüklenir.
3. **Balloon1** kuklasının kodları (Görsel 2.106)



Görsel 2.106: Balloon1 kuklasına ait kodlar

Oyun kurgusu gereği balonlar gökyüzüne doğru kaçmaktadır.

Balon sahneden çıktığında ($y \text{ konumu} > 190$) **Can** bir eksilir.

Oyun başladığında kullanıcının üç canı ($\text{Can } 3 \text{ yap}$) bulunmaktadır.

Oyun başladığında normal olarak skor da sıfırdan ($\text{Skor}'i \ 0 \text{ yap}$) başlamaktadır.

Hem balon patlatıldığında hem de balon kaçtığında aynı kodlar kullanılacağı için bir blok (**balon_baslat**) tanımlanmıştır.

Balon_baslat bloku balonu belirlenen aralıklar içinde rastgele bir konuma, rengini değiştirerek taşımaktadır.

Basit bir oyun olması için de bir balon seçilmiştir. İstenirse oyuna bir hız değişkeni eklenerek oyuncu balon patlattıkça oyunun hızlanması sağlanır. Bunun için **y konumunu 3 değiştir** blokunda 3 yerine **hız** değişkeni kullanılır.

Kullanıcı tüm canlarını bitirmedikçe ($\text{Can} > 0$) eğer **wand** balona değerse skor 5 puan artar, patlama sesi çıkar ve balonun konumu değişir (**balon_baslat**).

Kullanıcı bu durumu yeni balon çıkması olarak algılayacaktır.

Eğer oyuncunun canı kalmadıysa (Can>0 değilse) oyun bitirilir (durdur tümü).
Wand kuklasının kodları (Görsel 2.107)



Görsel 2.107: Yeşil bayrak tıklandığında çalıştırılacak kodlar

Kamera görüntüsü sahnede aktarılır. Görüntü tamamen saydam (100 % transparency) yapılarak sahnenin siyah bir şekilde görünmesini sağlamıştır.

Human body eklentisindeki **analyze image for hand from camera** bloku kameradaki görüntüde bulunan eli tespit etmek için kullanılmaktadır.

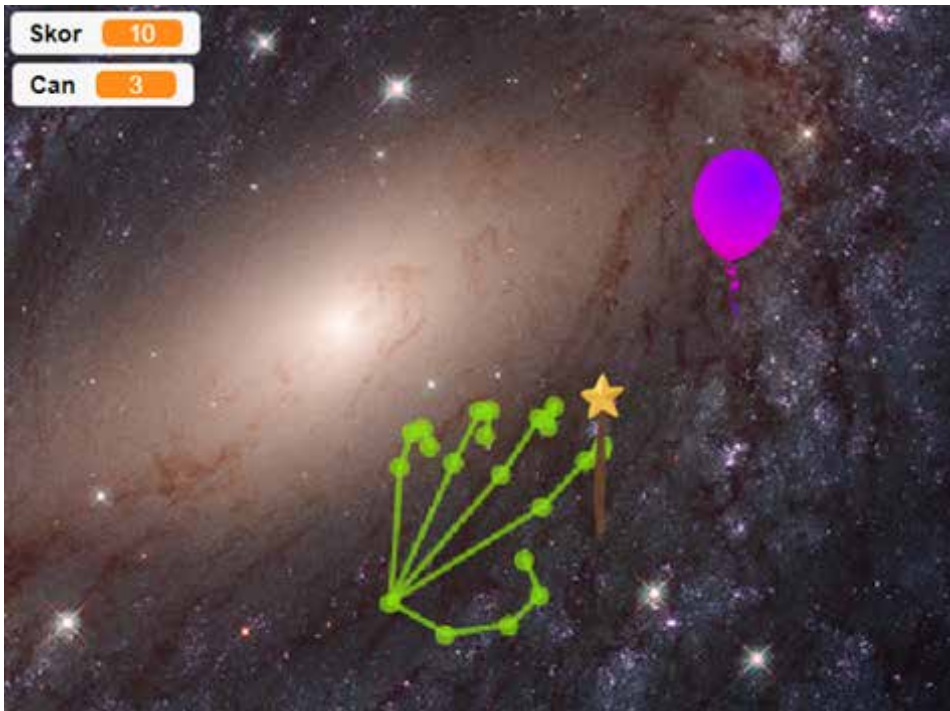
Wand kuklası işaret parmağının ucunun (x position of top of index finger, y position of top of index finger konumuna git) bulunduğu konuma gitmektedir.

4. Uygulamanın çalıştırılması

Yeşil bayrak tıklanarak uygulama çalıştırılır.

Kullanıcı elini hareket ettirerek **wand** kuklası ile balonları patlatmaya ve skoru artırmaya çalışır.

Çıktı (Görsel 2.108)



Görsel 2.108: Uygulamanın çalıştırılması



17. Uygulama: El Yazısı Tanıma

Bu çalışmada kalem nesnesi kullanılarak ekrana yazdırılan bir yazıyı metne çeviren bir uygulama geliştirilmektedir.



Ön Hazırlık

Bu uygulamadaki eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekir. Bu nedenle çalışmalara başlamadan önce cihazın internete bağlı olması ve PictoBlox'da bir hesap oluşturulması önerilir.

Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

Bilgisayarlı görü

PictoBlox Eklentileri

Artificial intelligence

Kalem

Uygulama Adımları

1. PictoBlox programı açılır.

PictoBlox ekranında boş sahnede **Tobi** kuklası görünecektir.

Tobi kuklası kaldırılır. Sahneye oyuna uygun bir dekor eklenir.

Pencil kuklası eklenerek sahne ve kukla ayarlanır (Görsel 2.109).



Görsel 2.109: Sahne ve kukla ayarları

2. Eklenti ekleme bölümünden **Artificial Intelligence** ve **Kalem** eklentisi yüklenir.
3. **Tobi** kuklasının kodları (Görsel 2.110)



Görsel 2.110: Sahne ve kukla ayarları

Artificial Intelligence eklentisindeki sahnedeki görüntünün içindeki el yazısını tanıma (recognize handwritten text in image from stage) bloku eklenir.

AI eklentisi bulut servislerini kullandığı için uygulamada oturum açılmalıdır.

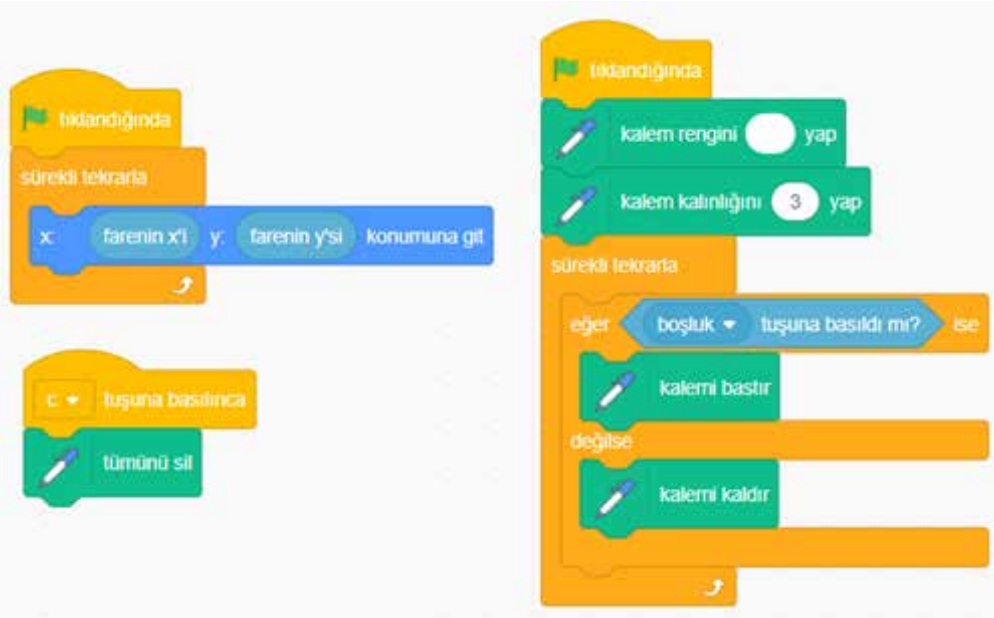
Bunun için eğer hesap oluşturulmamışsa öncelikle hesap oluşturulmalıdır.

Yapay zekâ bulut servisi için PictoBlox tarafından kullanıcılara bulut servislerinde kullanılmak üzere bir kredi tanımlanmaktadır. Bu kredi bittiğinde ücret ödenerek kredi alınması ya da yeni bir hesapla sistem giriş yapması gerekir. Bu nedenle kodlar **a tuşuna** basıldığı zaman çalıştırılmaktadır.

El yazısı tanıma işlemi bulut servis üzerinden yapıldığı için 1-2 saniye gecikmeli olabilir.

El yazısı tanıma sonucu 5 saniye boyunca sahnede **Tobi** kuklasının konuşma balonu olarak görünür.

Pencil kuklasının kodları (Görsel 2.111)



Görsel 2.111: Pencil kuklasının kodları

Kalem (pencil) nesnesi sahnede yazı yazdırmak için kullanılmaktadır. Fare ile kalem hareket ettirilir. Kalem hareket ederken boşluk tuşuna basılı ise sahneye çizim yapar (kalemi bastırır), değilse kalemi kaldırır ve sahneye çizim yapılmaz. Sahnedeki çizimi temizlemek için c tuşuna basıldığı zaman tümünü sil bloku kullanılır.

4. Uygulamanın çalıştırılması

Uygulama yeşil bayrağa tıklanarak çalıştırılır. Kalem ve boşluk tuşu kullanılarak (tuşa basılarak ve bırakılarak) basılarak istenen ifade sahneye yazılır. Klavyeden a tuşuna basıldığı zaman sahneye yazılan yazı AI servisi tarafından analiz edilerek sahnede konuşma balonu olarak gösterilir.

Çıktı (Görsel 2.112)

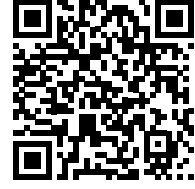


Görsel 2.112: Uygulamanın çalıştırılması



18. Uygulama: Basılı Metin Tanıma

Bu çalışmada basılı metinleri tanıyan (kitap, plaka, kartvizit vb.) bir uygulama geliştirilmektedir. Örnek olarak plaka tanıyan bir uygulama yapılmıştır.



Ön Hazırlık

PictoBlox'da bazı eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekir. Bu nedenle çalışmalara başlamadan önce cihazın internete bağlı olması ve PictoBlox'da bir hesap oluşturulması önerilir.

Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

Bilgisayarlı görü (OCR-Optik Karakter Tanıma)

PictoBlox Eklentileri

Artificial Intelligence

Uygulama Adımları

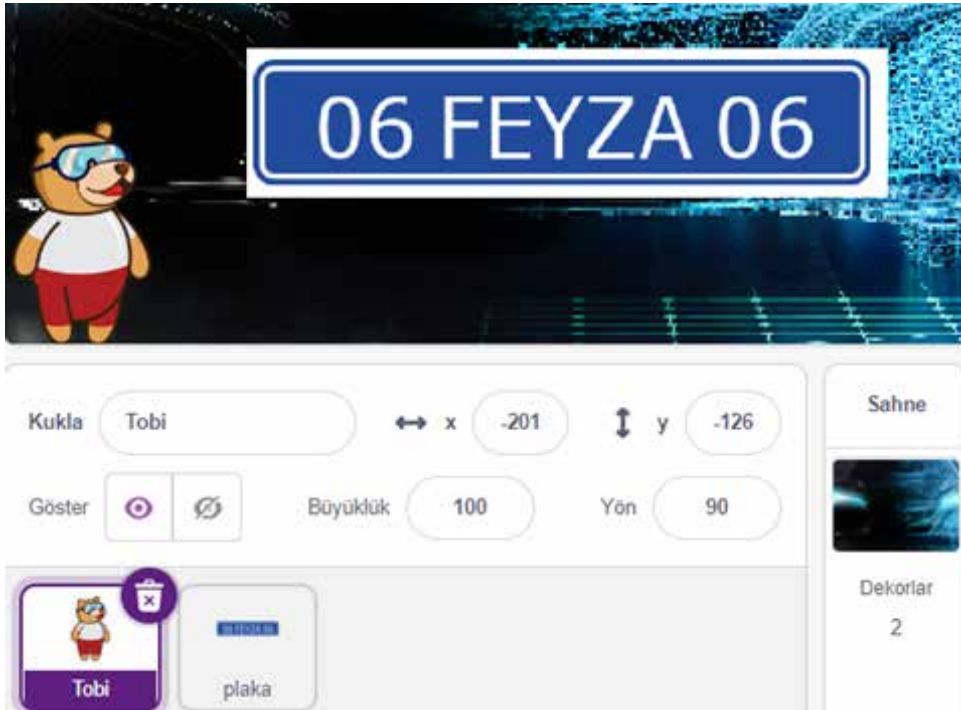
1. PictoBlox programı açılır.

PictoBlox ekranında boş sahnede Tobi kuklası görünür.

Tobi kuklası kaldırılır.

Sahneye bir araç görüntüsü dekor olarak eklenir.

Plaka görüntüsü dekor üzerinde olabileceği gibi bu uygulamada olduğu gibi kukla olarak da eklenebilir (Görsel 2.113).



Görsel 2.113: Pencil kuklasının kodları

2. Eklenti ekleme bölümünden Artificial Intelligence eklentisi yüklenir.
3. Tobi kuklasının kodları (Görsel 2.114)



Görsel 2.114: Tobi kuklasının kodları

Artificial Intelligence eklentisindeki sahnedeki görüntünün içindeki basılı metni tanıma (recognize printed text in image from stage) bloku eklenir.

AI eklentisi bulut servislerini kullandığı için uygulamada oturum açılmalıdır. Bunun için eğer hesap oluşturulmamışsa öncelikle hesap oluşturulmalıdır.

Yapay zekâ bulut servisi için PictoBlox tarafından kullanıcılara bulut servislerinde kullanılmak üzere bir kredi tanımlanmaktadır. Bu kredi bittiğinde ücret ödenerek kredi alınması ya da yeni bir hesap oluşturması gerekir. Bu yüzden kodlar **a tuşuna** basıldığı zaman çalışmaktadır.

Basılı metin tanıma işlemi bulut servis üzerinden yapıldığı için 1-2 saniye gecikmeli olabilir.

Basılı metin tanıma sonucu sahnede **Tobi** kuklasının konuşma balonu olarak görünür.

4. Uygulamanın çalıştırılması

Clavyeden a tuşuna basıldığı zaman sahnedeki plaka AI servisi tarafından analiz edilerek sahnede konuşma balonu olarak gösterilir.

Çıktı (Görsel 2.115)



Görsel 2.115: Uygulamanın çalıştırılması



19. Uygulama: Nesne Tanıma



Bu çalışmada sahnedeki bir dekoru veya kameradaki bir görüntüyü analiz ederek görüntüdeki nesnelere tanıyan bir uygulama geliştirilecektir. TM uygulamalarında yapıldığı gibi önceden eğitilmiş hazır bir model kullanılmıştır. Bu modeller AI eklentisiyle birlikte gelmektedir.

Ön Hazırlık

Bu uygulamadaki eklentileri kullanmak için internete bağlı olmak ve programda oturum açmak gerekir. Bu nedenle çalışmalara başlamadan önce cihazın internete bağlı olması ve PictoBlox'da bir hesap oluşturulması önerilir.

Bu uygulamadaki kodlara ve dosyalara karekod okutularak ulaşılabilir.

Kullanılan Yapay Zekâ Teknolojileri

Bilgisayarlı görü (nesne tanıma)

PictoBlox Eklentileri

Artificial intelligence

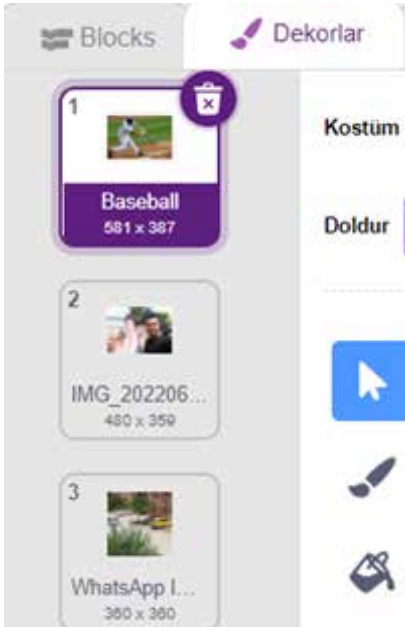
Uygulama Adımları

1. PictoBlox programı açılır.

PictoBlox ekranında boş sahnede Tobi kuklası görünür.

Tobi kuklası sahneden kaldırılır.

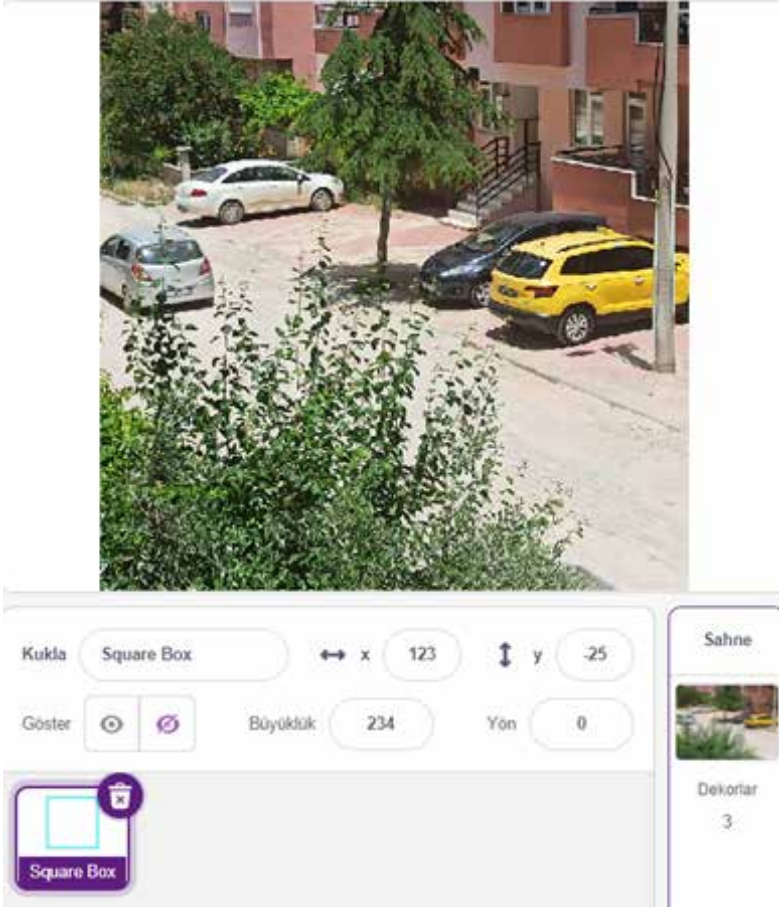
Sahneye nesne tanıma için kullanılacak görüntüler dekor olarak eklenir (Görsel 2.116).



Görsel 2.116: Dekorların eklenmesi

2. Square Box kuklası eklenir.

Bu kukla tespit edilen nesnelere çerçevelemek için kullanılır (Görsel 2.117).



Görsel 2.117: Sahne ve kukla ayarları

3. Eklenti ekleme bölümünden **Artificial Intelligence** eklentisi yüklenir.

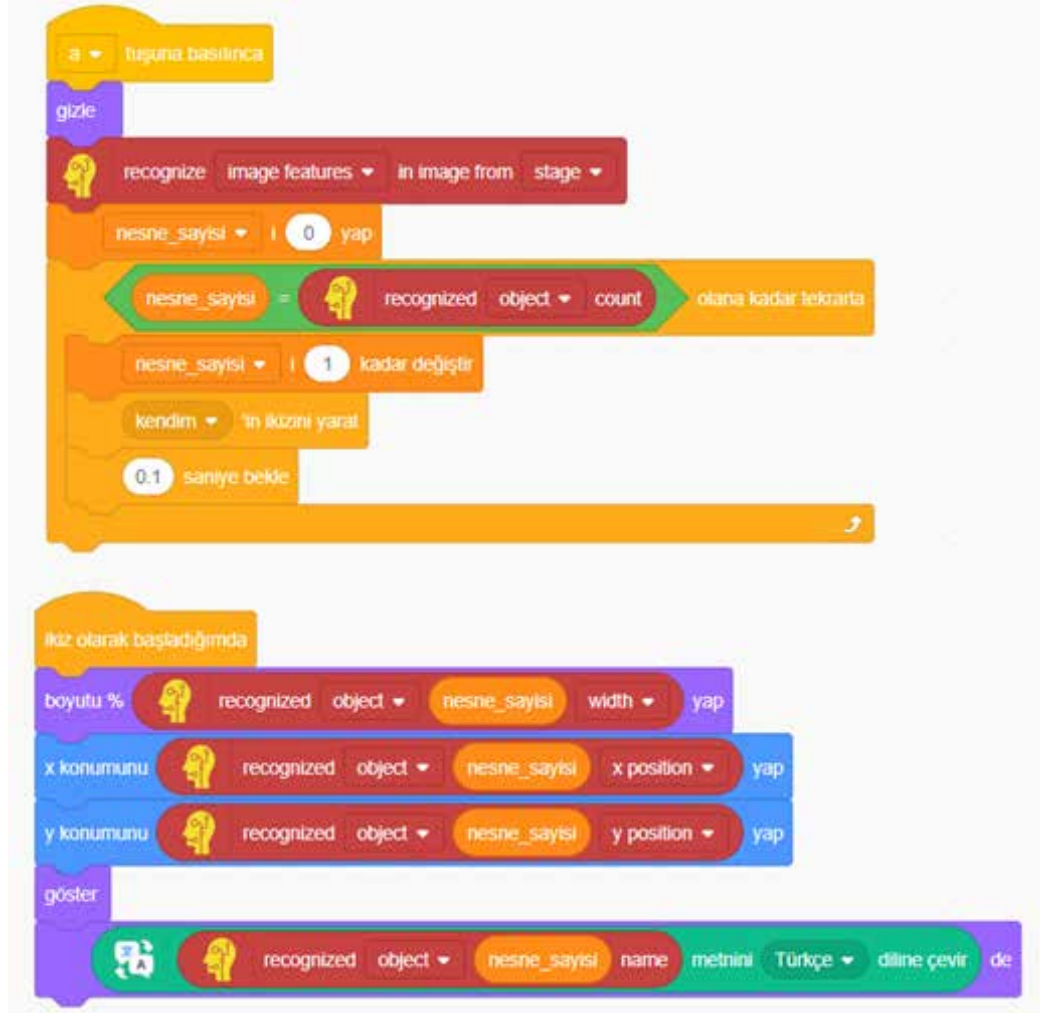
4. Kodlama (Görsel 2.118)



Görsel 2.118: Kodlar-1

Uygulamada hem sahnedeki görüntü üzerinden nesne tespitinin yapılması yanında kameradaki görüntüden de nesne tespiti yapılmak istenmiştir.

Bunun nedenle video görüntüsünü açmak ve kapamak için kod blokları eklenmiştir. Görüntü tespiti için farklı dekorları kullanmak amacıyla sağ ok tuşuna basıldığında sonraki dekor bloku eklenmiştir (Görsel 2.119).



Görsel 2.119: Kodlar-2

a tuşuna basıldığında sahnedeki görüntüdeki görüntü özellikleri analiz (recognize image features in image from stage) edilir.

Boşluk tuşuna basılarak kamera açılmışsa sahnede kameradan gelen görüntü olduğundan kamera görüntüsü analiz edilir.

Ancak tekrar dekorlardaki görüntünün analiz edilmesi için kamera (b tuşuna basılarak) kapatılmalıdır.

Görüntü analizi sonucu ne kadar nesne tespit edilirse **nesne_sayisi** adlı değişkende tutulur.

Tüm nesnelere çerçeveye alınarak etiketleri yazıldığından döngü kullanılarak **square box** kuklasının nesne sayısı kadar ikizi yaratılır.

Her ikiz başladığında nesne indisine göre boyutlandırılır. Örneğin: Boyutu 1. nesnenin genişliği kadar yap. X ve y konumlarını tespit edilen nesnenin konumunu alır ve nesnenin bir çerçeve (square box) içine alınmasını sağlar.

Tespit edilen her nesne için bu işlem tekrarlanır.

Etiket isimleri orjinalinde İngilizce olduğu için Türkçeye çevrilmiştir.

AI eklentisi bulut servislerini kullandığı için uygulamada oturum açılmalıdır. Bunun için eğer hesap oluşturulmamışsa öncelikle hesap oluşturulmalıdır.

Yapay zekâ bulut servisi için PictoBlox tarafından kullanıcılara bulut servislerinde kullanılmak üzere bir kredi tanımlanmaktadır. Bu kredi bittiğinde ücret ödenerek kredi alınması ya da yeni bir hesapla oturum açılması gerekir.

Bu nedenle servis kullanılan kodlar bir döngü içinde sürekli değil de klavyeden a tuşuna basıldığı zaman çalıştırılmaktadır.

5. Uygulamanın çalıştırılması

Klavyeden **a tuşuna** basıldığı zaman sahnedeki görüntü AI servisi tarafından analiz edilerek sahnede her nesne için çerçeveler oluşturulur ve nesnelerin adları yazılır.

Kullanıcı boşluk tuşuna basarak kamerayı açar ve daha sonra klavyede **a tuşuna** basarsa kameradaki görüntüden nesne tespiti yapılır.

Dekoru tekrar görüntülemek için klavyede **b tuşuna** basılarak kamera kapatılır.

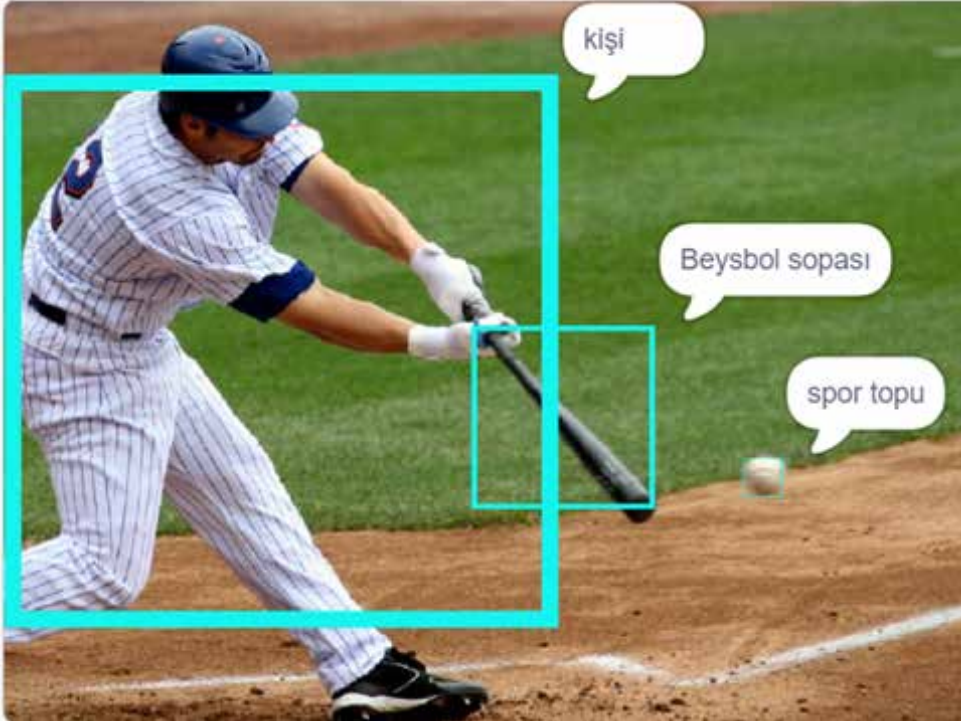
Sahnede diğer dekorları görüntülemek için klavyede sağ ok tuşuna basılır ve nesne tespiti yapılmak istenen dekor görüntülendiğinde klavyede **a tuşuna** basılır.

Çıktı (Görsel 2.120)



Görsel 2.120: Uygulamanın çalıştırılması

Uygulama sonraki dekorla bir daha çalıştırılabilir (Görsel 2.121).



Görsel 2.121: Uygulamanın çalıştırılması-2

2.6. mBlock ve Arduino ile Yapay Zekâ Uygulamaları

mBlock programı PictoBlox ile benzer yapay zekâ eklentilerine sahiptir. mBlockta yapay zekâ eklentilerinin yanında veri bilimi (data science) ve nesnelerin interneti (IoT) eklentileri de bulunmaktadır. PictoBlox'tan farklı olarak yapay zekâ servislerini kullanmak için kredi veya ücret talep edilmemektedir ancak günlük olarak tanıma limiti vermekte ve her gün bu limit yenilenmektedir.

mBlock programında PictoBlox tan farklı olarak ses ve görüntü tanıma işlemleri ayrı bir Recognition penceresinde yapılmaktadır. Bu işlem sonucunda yakalanan görüntü sahneye aktarılamadığı için Pictoblox uygulamasındaki gibi yakalanan kareler üzerinde doğrudan işlem yapılamaz

Bu bölümde yapay zekâ teknolojileri ile aygıtların (Arduino gibi) etkileşimini sağlayan projelere yer verilmiştir.

mBlock programında kuklalar ve aygıtlar için olan eklentiler farklıdır.



Program mBlock resmi sayfasından indirilebilir
(<https://mblock.makeblock.com/en-us/download/>).



Her aygıt için farklı eklentiler bulunmaktadır. Bir aygıt seçildiğinde aygıta özgü eklenti listesi gelmektedir.

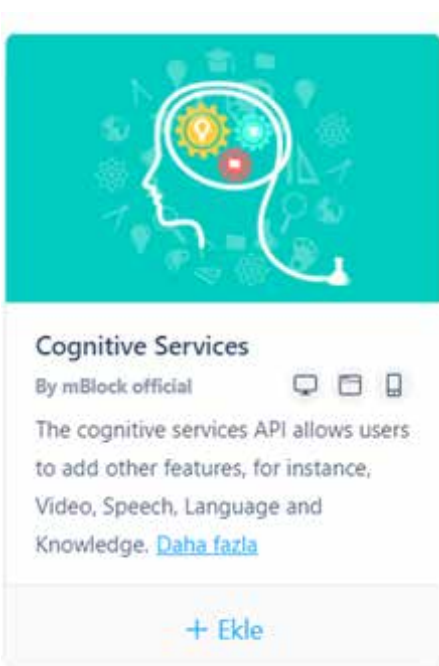
Yapay zekâ eklentilerini yüklemek için kuklalar bölümü seçilir. Uzantı ekleme butonuna tıklanır ve uzantı ekleme penceresi açılır. Bu pencerede eklenecek uzantı türü filtrelenerek veya adı yazılarak arama ile bulunur. Bu bölümde kukla uzantıları ve aygıt uzantıları bölümleri bulunmaktadır. Öncesinde seçilmemişse burada kukla uzantıları bölümü seçilerek filtreleme ve arama yapılabilir. AI alt kategorisi yapay zekâ uzantılarının bulunduğu kategoridir. Eğer eklenti dosyaları indirilmemiş ya da daha önceden indirilmiş ve ilgili kütüphaneye güncelleme gelmiş ise sağ üst köşedeki (+) ikon tıklanarak indirilir veya güncellenir. Bu işlemden sonra alt bölümdeki buton (+Ekle) aktif olacaktır. Bu butona tıklanarak uzantı kullanıma hazır hâle gelir.

Uzantıyı kaldırmak için uzantı ekleme penceresi açılarak uzantının alt bölümünde **X Kaldır** butonu kullanılır.

mBlock ve PictoBlox'da yapay zekâ eklentilerinin çoğu ortaktır. Önceki uygulamalarda benzer adımlar tekrara düşmemek için ayrıntılı olarak verilmemiştir.

2.6.1. Cognitive Services (Bilişsel Servisler)

Bu eklenti PictoBlox'daki AI eklentisine çok benzemektedir. Bu eklenti ile aşağıda belirtilen işlemler yapılabilir:



- Ses tanıma
- OCR ile el yazısı ve basılı metin tanıma
- Görüntünün özelliklerini tanıma, bir görüntüdeki marka logosu veya ünlüyü tanıma
- Yüz analizinden yaş, cinsiyet ve duygu durumu tespiti
- Baş hareketini algılama
- Saç rengini algılama, yüzde belirli bölümlerin kapatılıp kapatılmadığını kontrol etmek
- Metinden duygu analizi
- Gözlük modeli tanıma
- Dil algılama

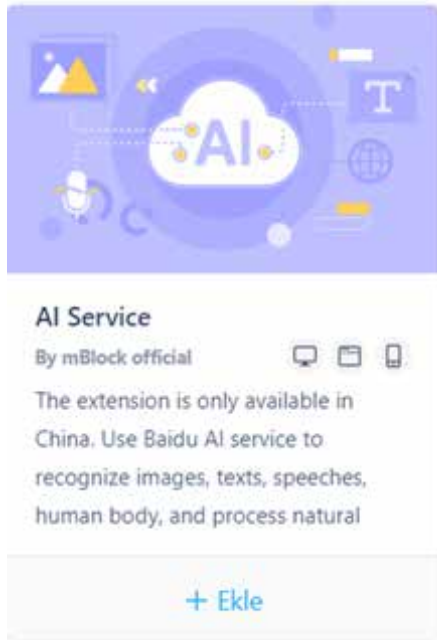
Metinden duygu analizi için Türkçe algılama özelliği bulunmaktadır. Bu servisi kullanmak için programda oturum açılmalıdır.

2.6.2. Makine Öğrenimi

Yapay öğrenme modelleri oluşturmak için kullanılan bu uzantı Google TM servisini kullanmaktadır. Makine öğrenmesi modelleri oluşturma süreci Google TM modelleri oluşturmakla aynıdır. PictoBlox programının aksine mBlock programında önceden oluşturulan modeller kullanılamaz. mBlock'ta Google TM modeli eklenti içinde oluşturularak kullanılabilir. TM servisinde sınıflar oluşturulurken örnekler tanıma penceresinden (kameradan görüntülerle) eklenerek model eğitilebilir. PictoBlox'tan farklı olarak mBlock'ta dekordan görüntü tanıma işlemi yapılamamaktadır.



2.6.3. AI Service (Yapay Zekâ Servisi)



Bu servis

Speech recognition: Konuşma tanıma

Text recognition: Metin tanıma

Image recognition: Resim tanıma

Human body recognition: İnsan vücudu tanıma

NLP-Natural language process: Doğal dil işleme servislerinden oluşmaktadır.

Speech recognition : Konuşmayı tanıyarak metne çevirmeye yarar. Ek olarak seslendirme için kadın, erkek sesi, konuşma hızı, ses tonu ve ses şiddeti gibi ayarlar yapılabilmektedir.

Text recognition : Bir el yazısını veya basılı bir metni tanıyarak dijital ortama aktarmak için kullanılır. Basılı bir metin (plaka gibi) içindeki yazıları ve numaraları ayırabilir. İnternet üzerinde bir görüntünün üzerindeki yazıları tanıyabilir.

Image recognition: Kameradan genel nesnelere, araçları, logoları, hayvan, bitki, meyve, sebze, yemek, resim pozisyonu ve hatları tanımak için kullanılır.

Human body recognition: Bu eklenti kameradan bir insanın el hareketlerini, vücut özelliklerini, maske takma durumunu, akıllı telefon kullanıp kullanmadığını, duyu durumunu, vücudundaki önemli bölümleri (burun, boyun, sol omuz, sağ omuz vb.) ve bunların konumlarını, ölçülerini tanımak için kullanılır.

NLP-Natural language process: Metinlerin sözlüksel ve duygusal içeriklerinin analizleri için kullanılmaktadır. Aynı zamanda iki metin arasında anlamsal benzerlik de ölçülebilir.



2.6.4. Text to speech (Metinden konuşmaya)

Bir metni seslendirmek için kullanılan uzantıdır.

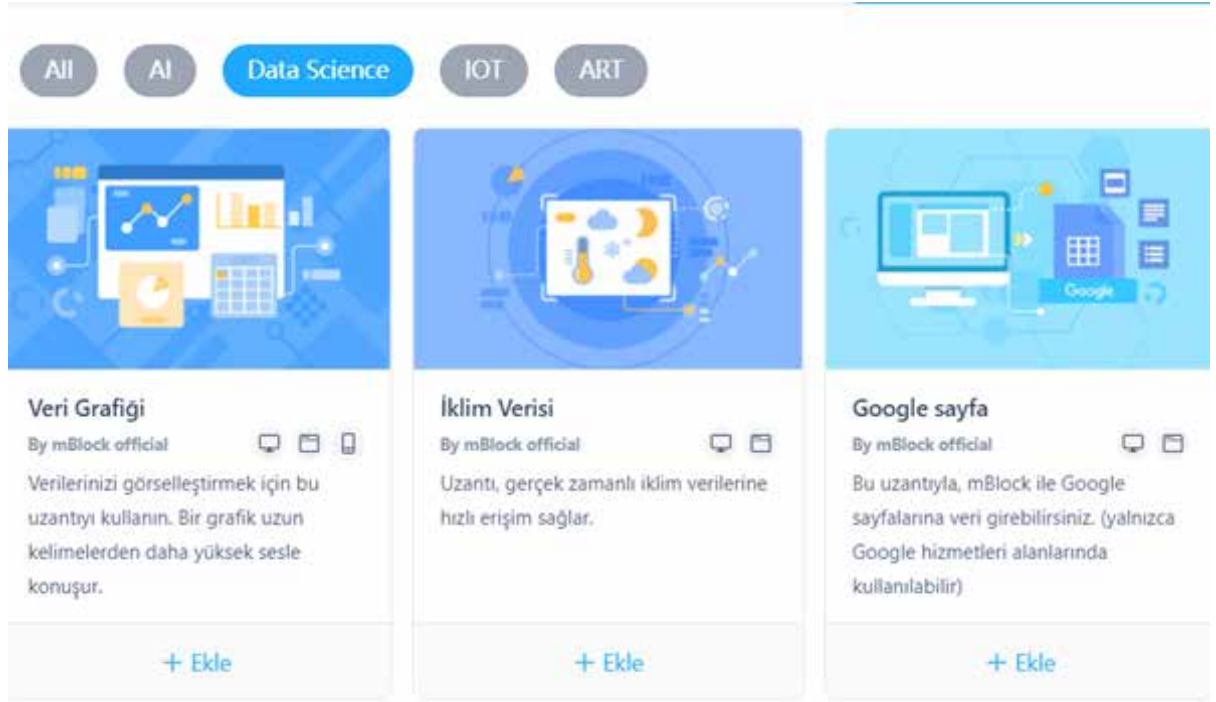
2.6.5. Translate (Çeviri)

Bir dilden başka bir dile çeviri yapmak için kullanılan uzantıdır.



2.6.6. Data Science (Veri Bilimi) Uzantıları

Yapay zekâ kategorisine girmese de veri bilimi uzantıları da zaman zaman yapay zekâ projelerinde kullanılan uzantılardır. Bunlar şöyledir: Veri Grafiği, İklim Verisi ve Google sayfa (Görsel 2.122).



Görsel 2.122: Veri bilimi uzantıları

Veri grafiği: Veriyi farklı grafik türlerini kullanarak görselleştirmek için kullanılır. Veriyi görselleştirmek veriyi anlamak için oldukça kullanışlı bir yöntemdir.

İklim verisi: PictoBlox'ın weather data eklentisi gibi gerçek zamanlı iklim verisine erişim sağlar. Bu uzantıyla belirtilen şehre ait sıcaklık değerleri, nem, hava durumu, en yüksek ve en düşük sıcaklık, gün doğumu ve gün batımı gibi bilgilere erişilebilir. PictoBlox uygulamasındaki gibi API key almaya gerek yoktur.

Google sayfa: Google hesabı olan kullanıcıların e-tabloları kullanmalarını sağlar. Projelerde e-tablolar veri tabanı gibi kullanılabilir.

2.6.7. Nesnelerin İnterneti Uzantıları

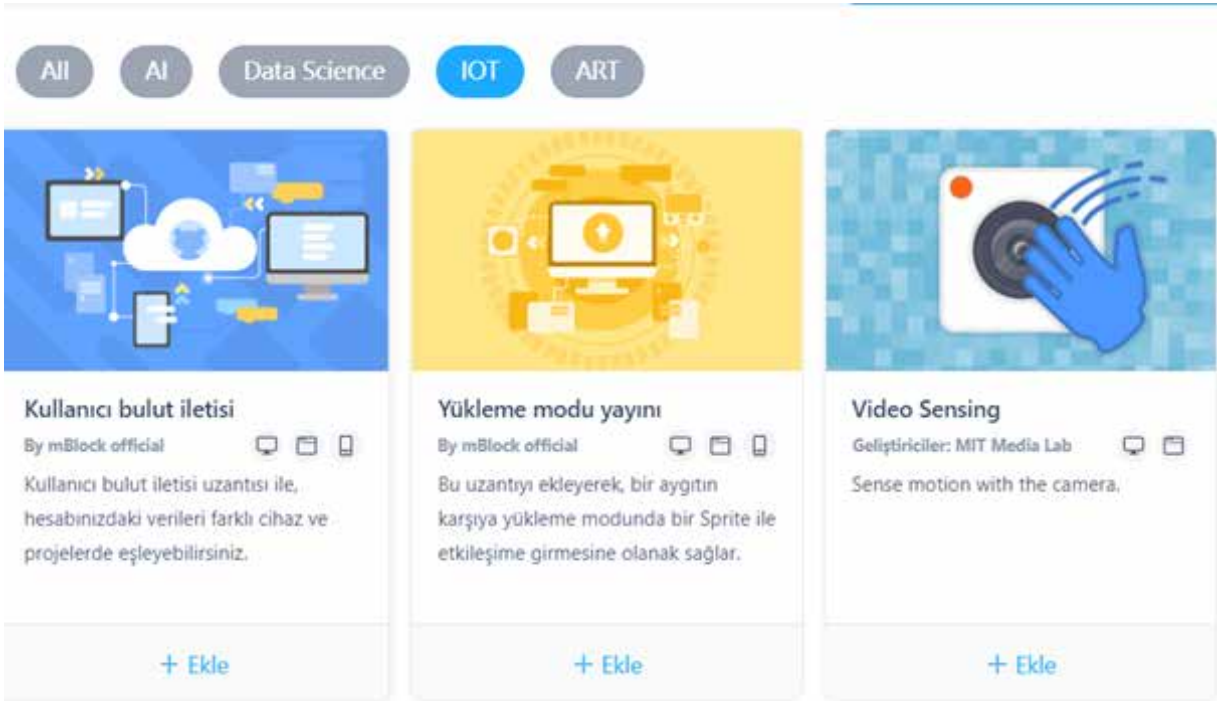
Yapay zekâ projelerinde kullanılan bazı IoT uzantıları şöyledir: Kullanıcı bulut iletisi, yükleme modu yayını ve video sensing (Görsel 2.123).

Kullanıcı bulut iletisi: Aynı hesapla başka cihazlarda oturum açarak tüm cihazlardaki projelerin tek noktadan çalışması sağlanır.

Yükleme modu yayını: Bir kukla ile bir aygıtın haberleşmesini sağlamak için kullanılır. Yapay zekâ eklentileriyle mikrodenetleyicileri kontrol etmek için kullanılan bir uzantıdır.

Video sensing: Video hareketinin yönünü ve hızını algılamak için kullanılan uzantıdır.

Uygulamalara geçmeden önce mBlock programında kukla-aygıt etkileşimini sağlamak için bazı ayarlamalar yapılmalıdır.



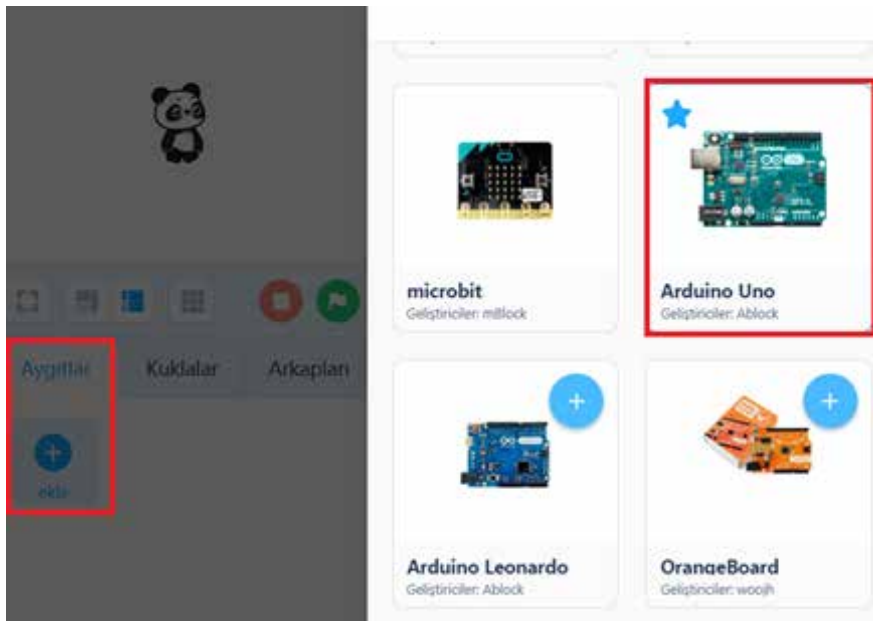
Görsel 2.123: Nesnelerin interneti uzantıları

2.6.8. mBlock Kukla ve Aygıt Etkileşimi

mBlock programında kuklalardaki yapay zekâ eklentileri ile aygıtları (mikrodenetleyici kartlar, robotik setler vb.) haberleştirmek için aygıt eklenir ve aygıt bağlantı ayarları yapılır.

1. Aygıtın eklenmesi

Aygıtlar bölümünde ekle butonuna basılarak aygıt penceresi açılır. Aygıt penceresinden kullanılacak aygıt kütüphanesi indirilmemişse + ikonuna tıklanarak indirilir. Aygıt seçilebilir duruma gelir. Aygıt seçilir ve tamam butonuna tıklanarak sahneye eklenir (Görsel 2.124).



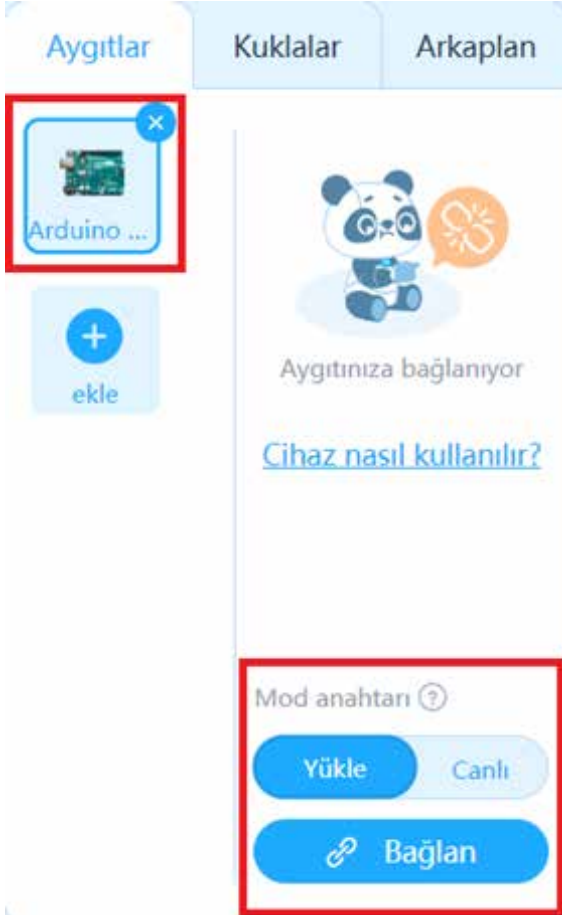
Görsel 2.124: Aygıt ekleme

Aygıt fare ile yıldız ikonuna tıklanarak favorilere eklenebilir.

Aygıt sahnede aygıtlar bölümünün altında görünür. mBlock ile kukla-aygıt etkileşimi mod anahtarı kullanılarak iki yolla sağlanabilir: Canlı ve yükle. Örnek aygıt olarak Arduino Uno eklenmiştir. Aygıt seçildikten sonra mod anahtarından canlı seçilir.

2.Canlı modda kullanım

Canlı mod seçildikten sonra bağlan butonuna tıklanır (Görsel 2.125).



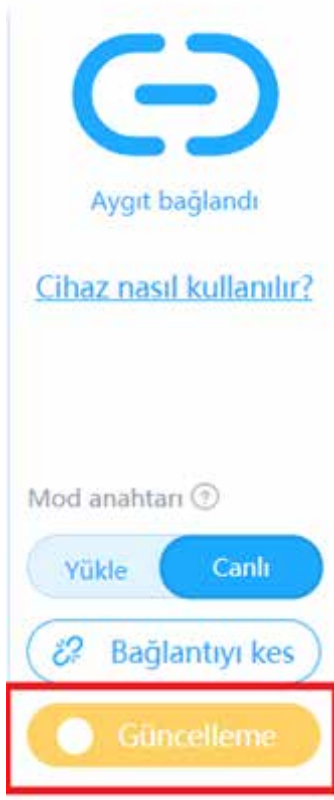
Görsel 2.125: Canlı modda bağlanma

Açılan pencereden aygıtın bağlı olduğu USB port seçilir ve Bağlan butonuna tıklanır. Bu port yanlış seçilirse belleim güncelleme işlemi başarısız olur (Görsel 2.126).



Görsel 2.126: USB bağlantısı

Bağlantı yapılırken aygıt açık olmalıdır. USB kablosu cihaza doğru bir şekilde bağlanmalıdır. Başka bir cihaz bağlı ise yeni cihaz bağlandığında eski cihazın bağlantısı kesilir. Aygıt bağlantısı yapıldıktan sonra mod anahtarı bölümünün altında güncelleme butonu belirecektir (Görsel 2.127). Bu buton canlı modda mBlock ile aygıtın haberleşmesini sağlayacak belleim yüklemek için kullanılır.



Görsel 2.127: Güncelleme

Güncelleme butonuna tıklandıktan sonra aygıt bellenim güncellemeleri penceresi açılır. Burada aygıt ve aygıta yüklenecek bellenime ilişkin bilgiler yer almaktadır. Bu bölümde güncellemeler butonuna tekrar tıklanır (Görsel 2.128).

Bellenim güncellemesi başlar ve güncelleme bitince ekrana bir mesaj gelir.

Görsel 2.128: Aygıt bellenim güncellemeleri

Görsel 2.129: Bellenim güncelleme

Tamam butonuna tıklandıktan sonra aygıtın bağlantısı kesilebilir (Görsel 2.129). Bu durumda Bağlan butonu kullanılarak bağlantı tekrar sağlanır.

Canlı modda aygıt içine yüklenen bellenim (firmware) ile mBlock programı etkileşim hâlinindedir. mBlock programı çalıştığı sürece aygıt çalışır. mBlock programı kapatıldığında veya aygıtın mBlock programının çalıştığı bilgisayarla bağlantısı kesilirse kod çalışmayacaktır.



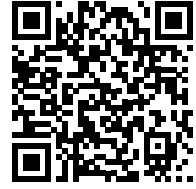
Görsel 2.130: Aygıt kütüphanesi

Canlı modu kontrol etmek amacıyla blink uygulaması yapılır ve uygulama kontrol edilir.



20.Uygulama: mBlock ile Blink uygulaması (Canlı Mod)

Bu uygulama ile kart üstünde girişi 13. pine bağlı LED yanıp sönecektir.



Gerekli Malzemeler

Arduino UNO 3 ve USB bağlantı kablosu

Ön Hazırlık

Bu uygulama ile ilgili kodlara ve dosyalara karekod okutularak ulaşılabilir.

Uygulama Adımları

1. Arduino Uno kart USB ile bilgisayar bağlanır.
2. mBlock programı üzerinde Aygıtlar bölümünden Arduino Uno aygıt olarak eklenir.
3. Aygıt eklendikten sonra Bağlan butonuna tıklanarak USB port seçilip aygıt bağlanır.
4. Belleim güncellemesi yapılır.
5. Aygıt bağlantısı kesilirse yeniden bağlantı kurulur.
6. Aygıtın kodlanması

Gerekli kod blokları eklenerek aygıt kodlanır (Görsel 2.131).



Görsel 2.131: Kodlar

7. Uygulamanın çalıştırılması

Sürekli tekrarlar blokuna tıkladığında Arduino UNO üzerindeki LED'in birer saniye arayla sürekli olarak yanıp sönmesi beklenir. Uygulama beklendiği gibi çalışıyorsa canlı modda uygulama geliştirmek için gerekli ayarlar başarıyla yapılmıştır demektir.

3.Yükleme Modunda Kullanım



Görsel 2.132: Yükleme modu

Aygıt bağlantısı yapıldıktan sonra aygıtı yükleme modunda çalıştırmak için izlenecek adımlar şöyledir.

Mod anahtarı yükle olarak seçilmelidir. Cihaz bağlantısı yapılmışsa yükle butonu görünecektir.

Yükleme modu kodun cihaza yüklenmesine ve mBlock ile bağlantısı kesildiğinde bile (güç aldığı sürece) çalışmasına olanak verir. Yükleme modunda aygıt kütüphanesindeki kodlar aktif haldedir (Görsel 2.133).



Görsel 2.133: Yükleme modu aygıt blokları



21. Uygulama: mBlock ile Blink uygulaması (Yükleme Modu)

Bu uygulama ile kart üstünde girişi 13. pine bağlı LED yanıp sönecektir.



Gerekli Malzemeler

Arduino UNO 3 ve USB bağlantı kablosu

Ön Hazırlık

Bu uygulama ile ilgili kodlara ve dosyalara karekod okutularak ulaşılabilir.

mBlock Uzantılar

Yükleme modu yayını (hem kuklaya hem de aygıta yüklenmelidir.)

Uygulama Adımları

1. Arduino Uno kart USB ile bilgisayar bağlanır.
2. mBlock programı üzerinde aygıtlar bölümünden Arduino Uno aygıt olarak eklenir.
3. Aygıt eklendikten sonra Bağlan butonuna tıklanarak USB port seçilip aygıt bağlanır.
4. Aygıtın kodlanması

Gerekli kod blokları eklenerek aygıt kodlanır (Görsel 2.134).



Görsel 2.134: Aygıt kodları

5. Kodlama yapıldıktan sonra Yükle butonu kullanılarak kodlar karta yüklenir.
6. Uygulamanın çalıştırılması

Kodlar yüklendikten sonra Arduino UNO üzerindeki ledin birer saniye arayla sürekli olarak yanıp sönmesi beklenir. Uygulama istendiği gibi çalışıyorsa yükleme modunda uygulama geliştirmek için gerekli ayarlar başarıyla yapılmıştır demektir.



Yapay zekâ uzantıları kuklalar için çalışmaktadır. Yapay zekâ uygulamalarını kullanarak aygıtları kontrol etmek için kuklalarla aygıtların haberleşmesini sağlamak gerekmektedir. Aygıt canlı modda çalışıyorsa haberleşme olaylar bölümündeki kodlar içinde bulunan kod blokları yardımıyla yapılır.

Eğer aygıt yükleme modunda çalışırsa o zaman hem aygıt hem de kukla için yükleme modu yayını eklentisi yüklenerek bunlar altında yer alan kod blokları kullanılır.

Arduino Uno

Yükleme Modu Yayını
By mBlock official

Bu uzantı, bir aygıtın yükleme Modunda bir Kukla ile etkileşime girmesine olanak sağlar.

+ Ekle



22. Uygulama: Ses ile LED Kontrolü Uygulaması (Canlı Mod)

Bu uygulamada sesli komutlar verilerek Arduino UNO karta bağlı bir LED açılıp kapatılmaktadır.



Kullanılan Yapay Zekâ Teknolojileri

Ses tanıma

mBlock Eklentileri

Cognitive Services

Ön Hazırlık

Bu uygulama ile ilgili kodlara ve dosyalara karekod okutularak ulaşılabilir.

Gerekli Malzemeler

Arduino UNO

USB bağlantı kablosu

LED

220 Ohm direnç

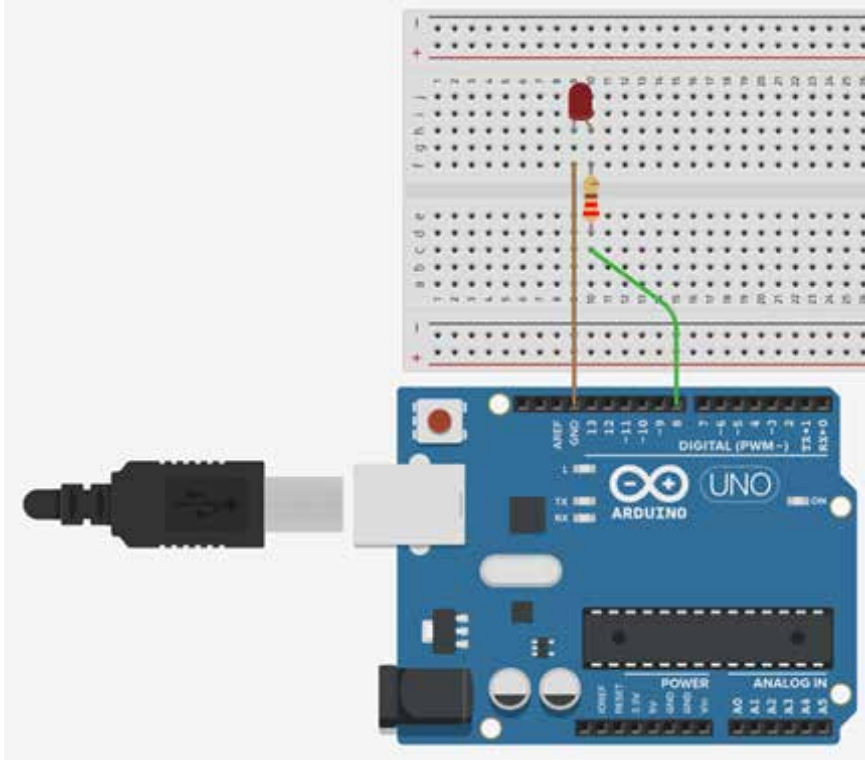
Bread board

Bağlantı kabloları

Uygulama Adımları

1. Kukla seçili iken uzantılar bölümünden bilişsel servisler uzantısı eklenir.
2. Arduino Uno kart USB ile bilgisayara bağlanır.
3. mBlock programı üzerinde aygıtlar bölümünden Arduino Uno aygıt olarak eklenir.
4. Aygıt eklendikten sonra Bağlan butonuna tıklanarak USB port seçilip aygıt bağlanır.
5. Belleim güncellemesi yapılır.
6. Aygıt bağlantısı kesilirse yeniden bağlantı kurulur.
7. Devrenin Kurulması

Devre şemasına uygun bir biçimde devre kurulur. LED Arduino Uno kartın 8 numaralı dijital pinine bağlanır (Görsel 2.135).



Görsel 2.135: Tinkercad devre şeması

8. Kuklanın kodlanması

Ses tanıma eklentisi kullanılarak ses metne çevrilir. Metinde "aç" ifadesi geçiyorsa aç iletisi, "kapa" ifadesi geçiyorsa kapat iletisi yayınlanır (Görsel 2.136).



Görsel 2.136: Kukla kodları



Görsel 2.137: Aygıt kodları

9. Aygıtın kodlanması

Ses tanıma sonucu gelen iletiye göre LED'in yanması ve sönmesi sağlanır (Görsel 2.137).

10. Uygulamanın çalıştırılması

Boşluk tuşuna basıldığında ses tanıma penceresi açılır. Kullanıcı "LED'i aç" gibi bir sesli komut verir ve LED yanar. Kullanıcı "LED'i kapat" gibi sesli bir komut verdiğinde ise LED söner. Uygulama doğru çalışmıyorsa devre ve bağlantılar kontrol edilmelidir. Ses tanımanın doğru çalışıp çalışmadığı ekrana yazdırılan ses tanıma sonucu metnine bakılarak kontrol edilir.



23. Uygulama: Ses ile LED Kontrolü Uygulaması (Yükleme Modu)

Bu uygulamada sesli komutlar verilerek Arduino UNO karta bağlı bir LED açılıp kapatılmaktadır.



Kullanılan Yapay Zekâ Teknolojileri

Ses tanıma

mBlock Eklentileri

Cognitive Services

Yükleme modu yayını (hem kuklaya hem de aygıta yüklenmelidir)

Ön Hazırlık

Bu uygulama ile ilgili kodlara ve dosyalara karekod okutularak ulaşılabilir.

Gerekli Malzemeler

Arduino UNO

USB bağlantı kablosu

LED

220 Ohm direnç

Bread board

Bağlantı kabloları

Uygulama Adımları

1. Kukla seçili iken uzantılar bölümünden bilişsel servisler uzantısı eklenir.
2. Sahne arka planı için bir karanlık bir de aydınlık kostüm ayarlanır.
3. Arduino Uno kart USB ile bilgisayara bağlanır.
4. mBlock programı üzerinde Aygıtlar bölümünden Arduino Uno aygıt olarak eklenir.
5. Aygıt eklendikten sonra Bağlan butonuna tıklanarak USB port seçilip aygıt bağlanır.
6. Yüklemeye modu yayını uzantısı hem kukla hem de aygıt için yüklenir.
7. Devrenin Kurulması

Devre şemasına uygun bir biçimde devre kurulur. LED Arduino Uno kartın 8 numaralı dijital pinine bağlanır.

8. Kuklanın kodlanması

Ses tanıma eklentisi kullanılarak ses metne çevrilir. Yüklemeye modu iletisi eklentisi kullanıldığında iletinin adı ve değeri belirtilebilir. Metin içeriğine göre durum değeri 1 veya 0 değerini almaktadır. Dekor da bu komuta göre değişmektedir (Görsel 2.138).

```
boşluk ▾ tuşu basılınca
recognize speech in Turkish ▾ for 2 ▾ secs
eğer < speech recognition result aç içeriyor? ise
  karşıya yükleme modu iletisi gönder durum değeri 1
  Komut: 1 de
  Spotlight-stage2 ▾ dekoruna geç
eğer < speech recognition result kapat içeriyor? ise
  karşıya yükleme modu iletisi gönder durum değeri 0
  Komut: 0 de
  backdrop1 ▾ dekoruna geç
```

Görsel 2.138: Kukla kodları

9. Aygıtın kodlanması

Ses tanıma sonucu gelen iletiye göre LED'in kapatılması ve açılması sağlanır (Görsel 2.139).



Görsel 2.139: Aygıt kodları

10. Kodlama yapıldıktan sonra yükle butonu kullanılarak kodlar karta yüklenir.

11. Uygulamanın çalıştırılması

Boşluk tuşuna basıldığında ses tanıma penceresi açılır.

Kullanıcı "LED'i aç" gibi bir sesli komut verir, arka plan değişir ve LED yanar.

Kullanıcı "LED'i kapat" gibi sesli bir komut verdiğinde ise arka plan değişir ve LED söner.

Uygulama doğru çalışmıyorsa devre ve bağlantılar kontrol edilmelidir.

Ses tanımanın düzgün çalışıp çalışmadığı ekrana yazdırılan ses tanıma sonucu metnine bakılarak kontrol edilir.



24. Uygulama: Yaş LED'leri (Yükleme Modu)

Bu uygulamada görüntüdeki kişinin yaşına göre LED'lerin yanmasını sağlayan bir uygulama geliştirilmiştir.



Kullanılan Yapay Zekâ Teknolojileri

Görüntü tanıma (yüz tanıma)

Ön Hazırlık

Bu uygulama ile ilgili kodlara ve dosyalara karekod okutularak ulaşılabilir.

mBlock Eklentileri

Cognitive Services

Yükleme modu yayını (hem kuklaya hem de aygıtaya yüklenmelidir)

Gerekli Malzemeler

Arduino UNO

USB bağlantı kablosu

3 x LED (sarı, kırmızı ve kahverengi)

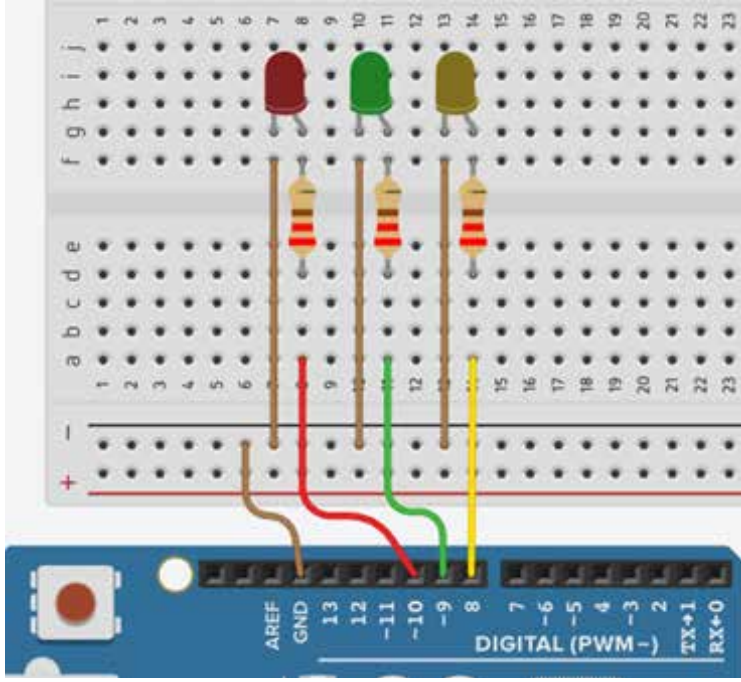
3 x 220 Ohm direnç

Bread board

Bağlantı kabloları

Uygulama Adımları

1. Kukla seçili iken uzantılar bölümünden bilişsel servisler uzantısı eklenir.
2. Arduino Uno kart USB ile bilgisayara bağlanır.
3. mBlock programı üzerinde Aygıtlar bölümünden Arduino Uno aygıt olarak eklenir.
4. Aygıt eklendikten sonra Bağlan butonuna tıklanarak USB port seçilip aygıt bağlanır.
5. Yüklemeye modu yayını uzantısı hem kukla hem de aygıt için yüklenir.
6. Devrenin Kurulması
Devre şemasına uygun bir biçimde devre kurulur.
Sarı LED 8 numaralı dijital pine bağlanır.
Yeşil LED 9 numaralı dijital pine bağlanır.
Kırmızı LED 10 numaralı dijital pine bağlanır (Görsel 2.140).



Görsel 2.140: Tinkercad devre şeması

7. Kuklanın kodlanması

İnsan yaşı tanıma eklentisi kullanılarak yaş tespit edilir.

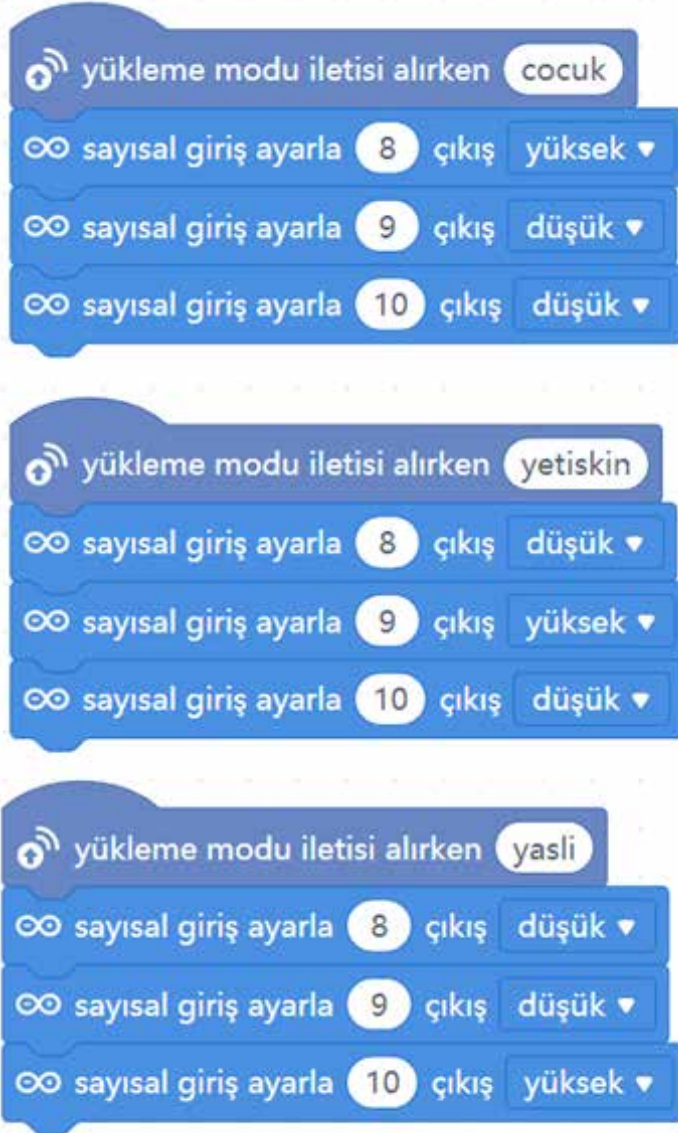
Belirli yaş aralıklarına göre karşıya yükleme modu iletisi gönderilir (Görsel 2.141).



Görsel 2.141: Kukla kodları

8. Aygıtın kodlanması

Kukladan alınan iletiye göre yaş aralığını temsil eden LED'in yanması diğerlerinin ise sönmesi sağlanır (Görsel 2.142).



Görsel 2.142: Aygıt kodları

9. Kodlama yapıldıktan sonra yükle butonu kullanılarak kodlar karta yüklenir.

10. Uygulamanın çalıştırılması

Boşluk tuşuna basıldığında görüntü tanıma penceresi açılır.

Kameradaki kişinin yaşı tespit edilir. Bu yaşa göre uygun LED yanar.

Uygulama çalışmazsa devre ve bağlantılar kontrol edilmelidir.

Yaş tespiti sonucu kontrol edilir.

Uygulama farklı yaş gruplarına ait insanlara ait fotoğraflar gösterilerek test edilir.



25. Uygulama: Duygu Durumu (Canlı Mod)

Bu uygulamada görüntüdeki kişinin duygu durumuna göre servo motorun uygun emojiyi gösterdiği bir uygulama geliştirilmiştir.



Kullanılan Yapay Zekâ Teknolojileri

Görüntü tanıma (Duygu tanıma)

mBlock Eklentileri

Bilişsel Servisler (Cognitive Services)

Ön Hazırlık

Bu uygulama ile ilgili kodlara ve dosyalara karekod okutularak ulaşılabilir.

Gerekli Malzemeler

Arduino UNO

USB bağlantı kablosu

Servo motor

Bread board

Bağlantı kabloları

Ön Hazırlık

Bu uygulama için emoji çıkartmaları kullanılarak hazırlanan bir kâğıt veya karton parçası servo motorun dönüş açısı uygun duygu durumları eşleştirecek şekilde ayarlanır.

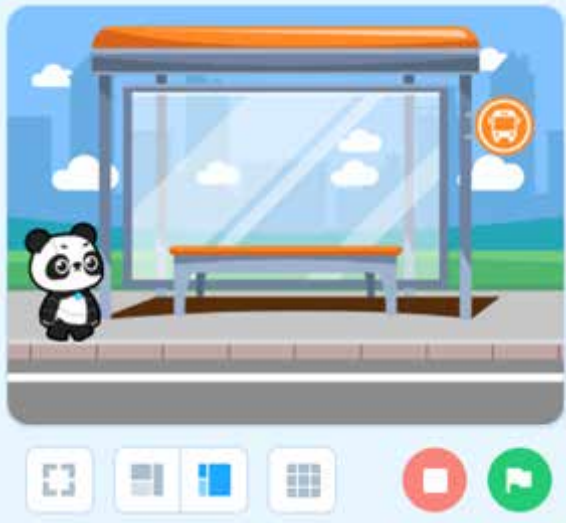
0 derece: mutlu 😊

90 derece: normal 😐

180 derece: üzgün 😞

Uygulama Adımları

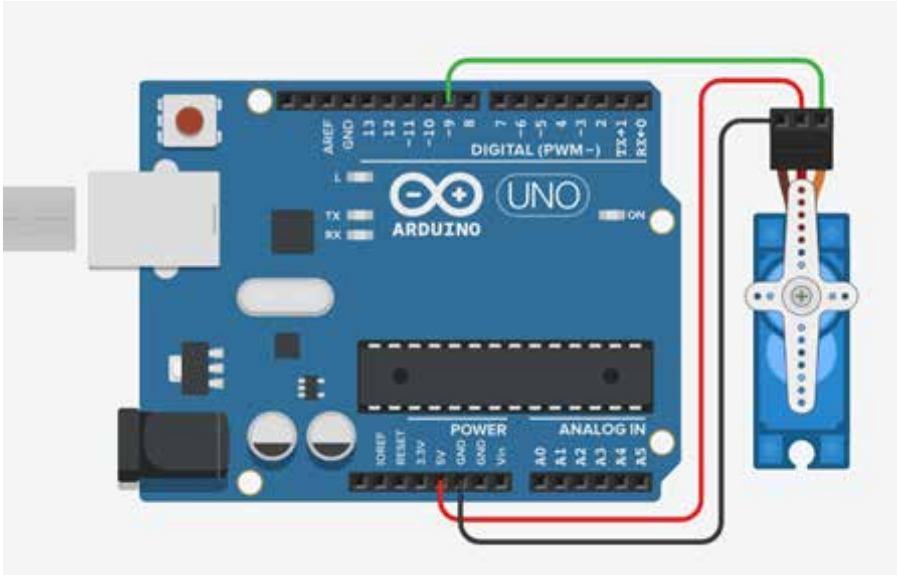
1. Kukla seçili iken uzantılar bölümünden bilişsel servisler uzantısı eklenir.
2. Sahne ve kukla ayarları yapılır (Görsel 2.143).



Görsel 2.143: Sahne ayarları

3. Arduino Uno kart USB ile bilgisayara bağlanır.
4. mBlock programı üzerinde Aygıtlar bölümünden Arduino Uno aygıt olarak eklenir.
5. Aygıt eklendikten sonra bağlan butonuna tıklanarak USB port seçilip aygıt bağlanır.
6. Bellenim güncellemesi yapılır.
7. Aygıt bağlantısı kesilirse yeniden bağlantı kurulur.
8. Devrenin Kurulması

Devre şeması kullanılarak devre kurulur. 9 numaralı PWM pinine servo motor bağlanır (Görsel 2.144).



Görsel 2.144: Tinkercad devre şeması

9. Kuklanın kodlanması

Duygu durumu tanıma eklentisi kullanılarak kameradan duygu durumu tespiti yapılır. Duygu durumuna uygun ileti gönderilir (Görsel 2.145).

```

boşluk ▾ tuşu basılınca
  recognize emotion after 1 ▾ secs
  eğer emotion is happy ▾ ? ise
    mutlu ▾ iletisini yayınla
    Mutlu olmanıza sevindim. de
  eğer emotion is neutral ▾ ? ise
    normal ▾ iletisini yayınla
    Merhaba! de
  eğer emotion is sad ▾ ? ise
    üzgün ▾ iletisini yayınla
    Üzgün görünüyorsunuz bir derdiniz mi var? de

```

Görsel 2.145: Kukla kodları

10. Aygıtın kodlanması

Kukladan alınan iletiye göre servo motorun uygun açıda dönmesi sağlanır (Görsel 2.146).

```

mutlu ▾ iletisini aldığımda
  1 defa tekrarla
  servo pin 9 açı 0
normal ▾ iletisini aldığımda
  1 defa tekrarla
  servo pin 9 açı 90
üzgün ▾ iletisini aldığımda
  1 defa tekrarla
  servo pin 9 açı 180

```

Görsel 2.146: Aygıt kodları

11. Uygulamanın çalıştırılması

Boşluk tuşuna basıldığında görüntü tanıma penceresi açılır.

Kameradaki kişinin duygu durumu tespit edilir.

Duygu durumuna göre servo motor uygun açıda döner ve duygu durumunu belirten emojiyi gösterir.



26.Uygulama: mBot ve TM (Yükleme Modu)

Bu uygulamada bir TM görüntü tanıma modeli oluşturularak el hareketleriyle (yumruk ve avuç içi gösterme) mBot kontrolü yapılmaktadır.



Kullanılan Yapay Zekâ Teknolojileri

Görüntü tanıma (duygu tanıma)

mBlock Eklentileri

TM (Makine Öğrenimi)

Yükleme modu yayını (hem kukla için hem de aygıt için)

Ön Hazırlık

Bu uygulama ile ilgili kodlara ve dosyalara karekod okutularak ulaşılabilir.

Gerekli Malzemeler

mBot ve USB bağlantı kablosu (kablosuz bağlantı da yapılabilir).

Uygulama Adımları

1. Kukla seçili iken uzantılar bölümünden makine öğrenimi uzantısı eklenir.
2. mBot, USB ile bilgisayara bağlanır.
3. Gerekli bağlantı ayarları yapılır.
4. Aygıtlar bölümünden yeni aygıt olarak mBot eklenir.
5. Bağlan butonuna tıklanarak mBot bağlantısı yapılır.
6. Yüklemeye modu yayını uzantısı hem kukla hem de mBot için yüklenir.



Görsel 2.147: Model oluşturma penceresi

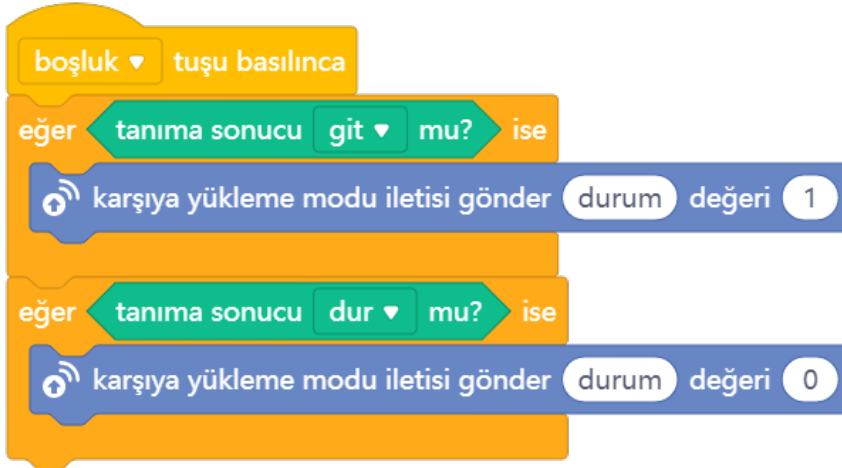
7. TM modelinin oluşturulması

Önceki uygulamalarda olduğu gibi TM üzerinden bir model eğitilir. Eğitim modeli butonuna tıklanarak TM model oluşturma servisi açılır. Bu uygulamada mBot için 3 sınıf eğitilmiştir. El yumruk şeklinde ise mBot hareket edecek, el açık dur işaret şeklinde ise mBot duracaktır. Üçüncü sınıf ise el olmadan sürücünün (insan) olduğu görüntülerden oluşmaktadır. Sınıflara örnekler kameradan verilir ve model eğitilir. Model eğitildikten sonra bloklarda sınıflara ait bilgiler yer alacaktır (Görsel 2.147).

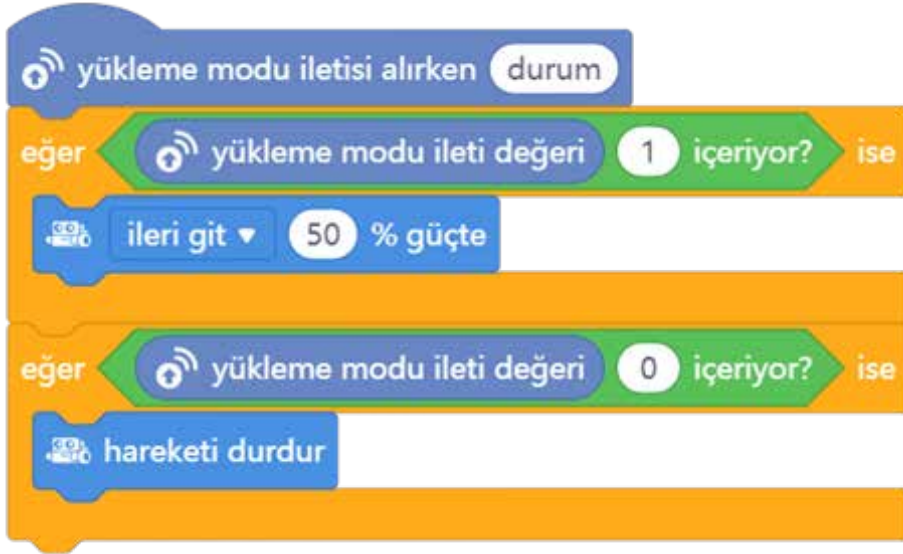
8. Kuklanın kodlanması

Boşluk tuşuna basıldığında bir tanıma penceresi açılarak ekrandaki görüntü tespit edilir.

Bu görüntü git sınıfına aitse karşıya yükleme modu iletisi 1, dur sınıfına aitse 0 gönderilir (Görsel 2.148).



Görsel 2.148: Kukla kodları



Görsel 2.149: Aygıt kodları

9. Aygıtın kodlanması

Gelen iletiye göre mBot aygıtının hareket etmesi için gerekli kodlama yapılır (Görsel 2.149).

10. Yükle butonuna tıklanarak kodlar mBot aygıtına yüklenir.

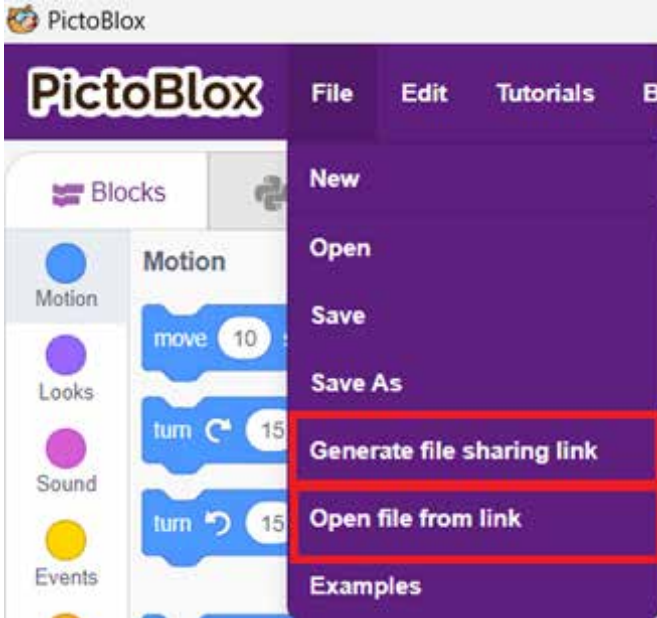
11. Uygulamanın çalıştırılması

Boşluk tuşuna basıldığında görüntü tanıma penceresi açılır.

Kameradaki kişinin el hareketine göre mBot hareket eder veya durur.

Kullanıcı elini yumruk yaptıysa mBot hareket eder yumruğunu açtıysa mBot durur.

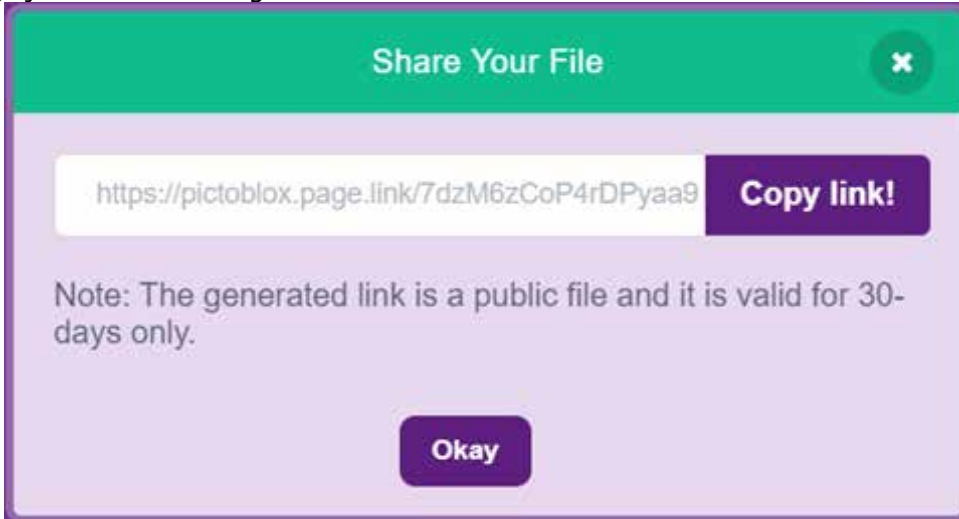
4.7. PictoBlox ile Projelerin Paylaşılması ve Yüklenmesi



PictoBlox uygulamasında yapılan projeler buluta yüklenerek bağlantı şeklinde paylaşılabilir. Proje yapıldıktan sonra File menüsünden Generate file sharing link tıklanarak paylaşılabilir bir bağlantı oluşur (Görsel 2.150). Bağlantıyı oluşturmak için uygulamada oturum açmak gerekir.

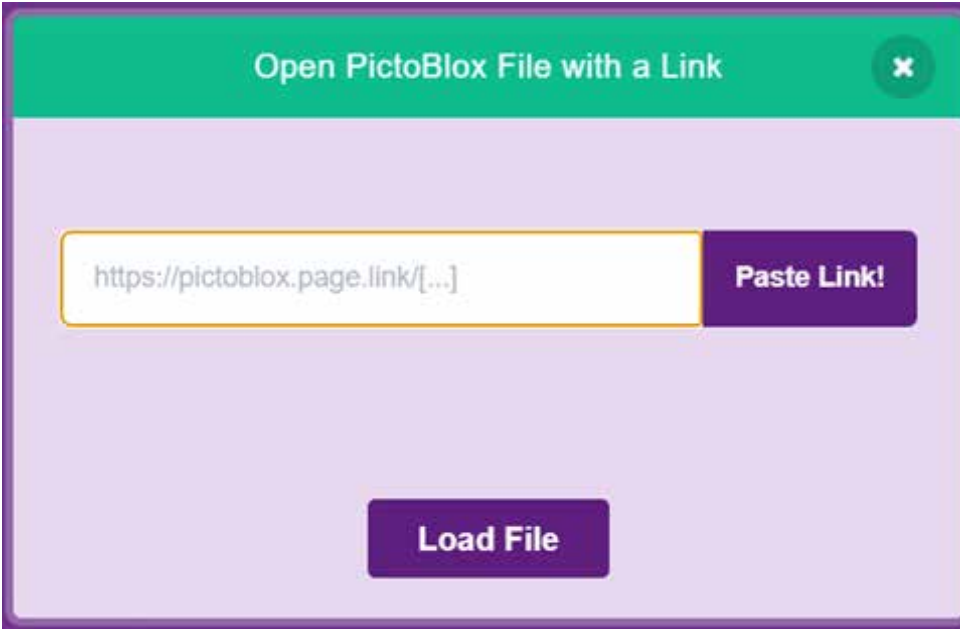
Görsel 2.150: PictoBlox proje bağlantısı oluşturma

Oluşturulan bağlantı Copy link! butonu kullanılarak kopyalanır. Bu bağlantı paylaşıldığında yapılan bir projenin dosyası başkaları tarafından da yüklenebilir (Görsel 2.151). Bağlantının geçerlilik süresi 30 gündür.



Görsel 2.151: PictoBlox proje bağlantısı

Bir bağlantıyı kullanarak proje dosyası yüklemek için File menüsünden Open file from link seçeneği tıklanarak açılan penceredeki bölüme bağlantı yapıştırılır ve Load File butonuna tıklanır (Görsel 2.152).

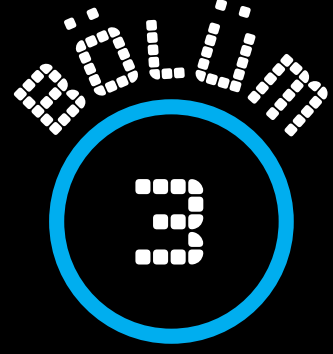


Görsel 2.152: Proje bağlantısı ile proje yükleme

Bu bölümde yapılan PictoBlox uygulamalarının paylaşılabilir bağlantılarına karekod okutularak veya linke tıklanarak ulaşılabilir.







**GERÇEK HAYAT
PROBLEMLERİNE
STEM TABANLI
YAPAY ZEKÂ ÇÖZÜMLERİ**

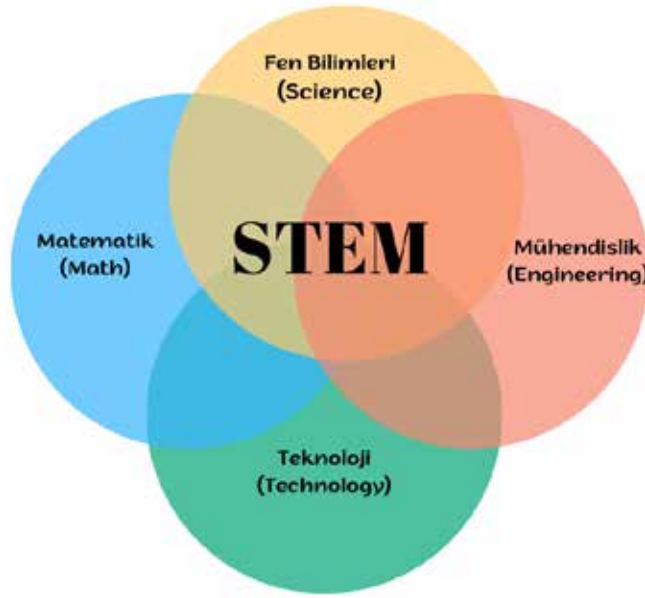
**Dr. Murat ALTUN
Emine KUTLU**



Günümüzde yaşanan teknolojik gelişmeler, yaşam standartlarını değiştirmektedir. Bireyler değişen yaşam standartlarında yeni problemlerle karşılaşmaktadır. Bu problemlere uygun çözümler üretebilmek için bireylerin bazı becerilere sahip olmaları gerekir. Bireylerin yaşama ve kariyerlerine hazırlanmaları için 21. yüzyıl becerilerini geliştirmeleri, kilit bir role sahiptir. 21.yüzyıl becerileri farklı organizasyonlar tarafından ele alınmaktadır. Bu organizasyonlardan biri olan Partnership for 21st Century göre 21. yüzyıl becerileri üç ana gruptan oluşur[1].

- Öğrenme ve Düşünme Becerileri: Eleştirel düşünme ve problem çözme, yaratıcılık ve yenilikçi düşünme becerileri, iletişim ve iş birliği becerileri, bağlamsal öğrenme becerileri, bilgi ve medya okuryazarlığı.
- Bilgi ve İletişim Teknolojileri Okuryazarlığı: Bilgi (information) okuryazarlığı, medya okuryazarlığı ve teknoloji okuryazarlığı.
- Yaşam ve Kariyer Becerileri: Liderlik ve girişimcilik, etik değerler, esneklik ve uyum, sosyal sorumluluk.

Bu becerilerin bireylere kazandırılmasında eğitim sistemlerine önemli görevler düşmektedir. Bu yüzden çoğu ülke eğitimde reformlar gerçekleştirerek yenilikçi öğretim yaklaşımlarını benimsemiştir. Bu öğretim yaklaşımlarından biri de STEM eğitimidir. STEM, Fen Bilimleri (Science), Teknoloji (Technology), Mühendislik (Engineering) ve Matematik (Math) disiplinlerinin İngilizce baş harflerinin kısaltmasından oluşmaktadır. Bu eğitim yaklaşımı, okul öncesinden yükseköğretime kadar tüm eğitim ve öğretim sürecini kapsayan disiplinler arası bir yaklaşım olarak kabul edilmiştir [2]. STEM eğitimi, bireylerin fen bilimleri, teknoloji, mühendislik ve matematik disiplinlerine olan ilgilerini artıracak ve 21. yüzyıl becerilerinin gelişimine katkı sağlayacak bütünlük uygulamaları esas almaktadır [3][4][5]. Bütünlük STEM Eğitimi Görsel 3.1’de verilmiştir.



Görsel 3.1: Bütünleşik STEM Eğitimi [6]

STEM eğitimi disiplinler arasında bir köprü kurarak fen bilimleri, teknoloji, mühendislik ve matematik derslerinin bir bütünlük içerecek şekilde öğrenciye sunulmasını sağlar (Görsel 3.1). [7][8]

Günlük yaşamın her alanında etkisini gösteren yapay zekâ teknolojileriyle özgün projeler geliştirmek, STEM alanlarının gelişmesine katkı sağlamaktadır. STEM eğitimi problem çözüme, yaratıcılık, iş birliği, girişimcilik gibi 21. yüzyıl becerilerinin gelişmesi bireylerin günlük yaşamlarında aktif olarak çalışan ve üreten bireyler olmasına yardım etmektedir. Üreten, araştıran ve sorgulayan bireylerin yetiştirilebilmesi için okullar gerçek yaşam odaklı olmalıdır [9]. Gerçek yaşam sorunlarını fark edebilen bireyler bu sorunları araştırmayı, sorgulamayı, bu sorunlara çözümler üretebilmeyi ve ürettiği çözümleri gerçek hayata uygulayabilmeyi öğrenir. STEM eğitimi ile bireylere gerçek yaşam problemlerinden kesitler sunulur ve bireylerin gerçek yaşam problemlerine çözümler üretmeleri sağlanır. Böylece gerçek yaşam sorunlarının çözümü için bilimsel ve teknolojik ürünler üretebilen, bilime, teknolojiye ve mühendisliğe ilgi duyan geleceğin bilim insanları ve mühendisleri yetiştirilir [10]. STEM eğitiminin bireylere sağladığı yararlar şunlardır[6]:

- Gerçek hayat sorunlarına çözümler üreterek problem çözme becerisi geliştirmelerini sağlar.
- Temel bilgi ve becerilerini kullanmalarını, mühendislik alanında tasarım yaparak yenilikçi ve yaratıcı düşünme becerilerini geliştirmelerini sağlar.
- Farklı disiplinlerden elde ettiği bilgileri ilişkilendirmesini ve disiplinler arası bakış açısı kazanmalarını sağlar.
- Eğitim programının içeriğini canlandırarak eğlenceli, keyifli ve demokratik bir öğrenme ortamı sağlar.
- Bireylerin iş birliği içinde çalışmasını sağlayarak, onların özgüven ve öz yeterliliğini geliştirir.
- Bireylerin 21. yüzyıl becerileri kazanmalarına yardım eder.

STEM eğitiminde amaç; fen bilimleri, teknoloji, mühendislik ve matematik disiplinlerinin kendilerine özgü bilgi ve becerilerini farklı bakış açısıyla bireylere kazandırmaktır [11]. STEM eğitimi farklı disiplinlerin bir araya gelmesiyle bireylerin gerçek dünya problemlerini belirleyerek çözümler üretmelerini ve 21. yüzyıl becerilerini geliştirmeyi amaçlamaktadır. STEM eğitimi ile bireylerin disiplinler arası öğrenmiş oldukları bilgi ve becerileri toplumun ihtiyacını karşılayacak yönde kullanması beklenmektedir. STEM eğitimi bireyleri öğrenmeye teşvik edecek uygulamalara yer vererek öğrenilen bilginin kalıcılığını da sağlamaktadır. Bu yüzden öğrenme ortamlarında STEM eğitim yaklaşımının benimsenmesi önemlidir.

STEM Eğitimi ve Teknoloji Entegrasyonu

Teknoloji günlük yaşamda, iş ve kariyer alanlarında vazgeçilmez unsurlarından biri hâline gelmiştir. STEM disiplinlerinden biri olan teknoloji, bireylerin günlük yaşam problemlerinin çözümünde kullandığı araçlar olarak düşünülebilir. STEM eğitiminde teknolojinin entegrasyonu bireylerin edindiği disiplinler arası bilgileri günlük yaşamda uygulamalarına katkıda bulunabilir. Teknoloji, bireylere fen ve matematik içeriklerinden yararlanarak günlük yaşam problemlerini çözüp yaşam standartlarını geliştirme imkânı sunmaktadır [7] [12]. Teknoloji sayesinde fen bilimleri, matematik ve mühendislik disiplinlerini uygulamaya dönük eylemler yapılabilir. STEM eğitimi disiplinler arasında ilişki kurarak bireylerin yeni beceriler edinmesine katkı sağlamaktadır.

STEM eğitiminde teknolojinin entegre edilmesi, teknoloji okuryazarlığının geliştirilmesini, teknolojik ürün geliştirilmesini ve uygulamaya yönelik beceriler kazandırılmasını sağlamaktadır[13]. STEM eğitiminde bireyler günlük yaşam problemlerine çözümler oluşturabilmek için araştırma ve sorgulama yapmalıdır. Bireyler araştırma sorgulama sürecinde teknolojiden aktif bir şekilde yararlanmalıdır. Gerçek yaşam problemlerinin çözümleri tasarlanırken karton, strafor köpük gibi kolay ulaşılabilir malzemelerin yanında robotik kitleler, elektronik malzemeler, bilgisayar programları gibi teknolojik araç gereçler de kullanılmaktadır. Teknolojinin eğitim ortamına entegrasyonu ile günlük yaşam problemlerine çözüm arayan bireyler, çağımızın becerilerini kullanarak daha anlamlı bir öğrenme ortamında yer alabilmektedir. Teknoloji entegrasyonu yapılan öğrenme ortamlarında bireylerin teknoloji araçlarının kullanımını ve üretimini daha iyi kavrayabileceği düşünülmektedir. Böylece bireyler STEM eğitimi ile teknolojik ürün tasarım sürecini daha iyi anlamlandırır [3] [14]. Bu bağlamda STEM eğitiminde teknoloji entegrasyonu önemli görülmektedir.

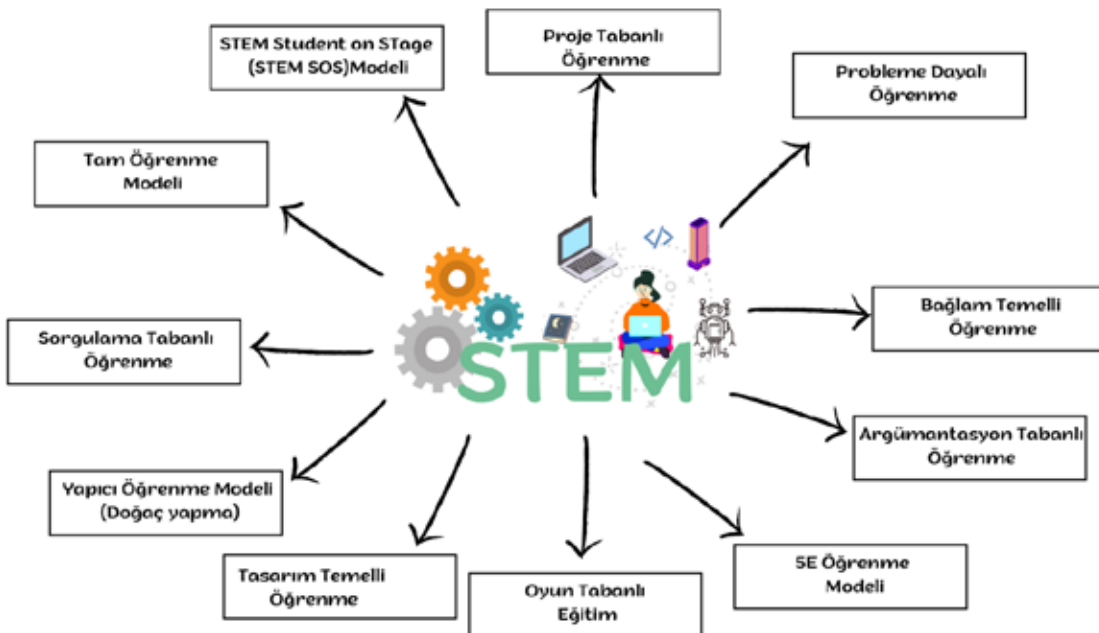
Günümüz dünyasına yön veren teknolojik gelişmeler ülkelerin ekonomik ve politik açıdan ayakta kalmalarını sağlarken, bireylere de iş alanları ve kariyer tercihleri sunmaktadır [15]. Teknolojinin günlük hayata uyum sağlamasıyla bireylerin uzmanlaştığı alanların dışında farklı disiplinlere de yatkın olmaları gerekmektedir [3]. Bu nedenle farklı iş alanları ve nitelikli insan arayışı ortaya çıkmıştır. Geleceğin mesleklerine bakıldığında Artırılmış Gerçeklik Seyahat Oluşturucusu/Yaratıcısı, Sürücüsüz Araç Mühendisi, Biyoteknoloji Mühendisi, İklim Analisti ve Hava Moderatörü, Kodlama Etik Uzmanı[16] gibi mesleklerin yanında yapay zekâ alanları da görülmektedir. Bilimsel ve teknolojik gelişmelerin hızlı olması, akıllı telefonlar gibi günlük işlerde kullanılan pek çok elektronik araçta yapay zekâ teknolojileri kullanılmasını sağlamıştır. Günümüzde yapay zekâ teknolojisi, satranç

oyunayan bilgisayarlardan otonom araçlara kadar pek çok alanda kendini göstermektedir. Günlük yaşamın her alanında etkisini gösteren yapay zekâ uygulamaları özgün projelerin ortaya çıkmasını sağlayarak STEM alanlarının gelişmesine katkı sağlamaktadır [13].

STEM Eğitiminde Kullanılan Öğretim Yaklaşımları

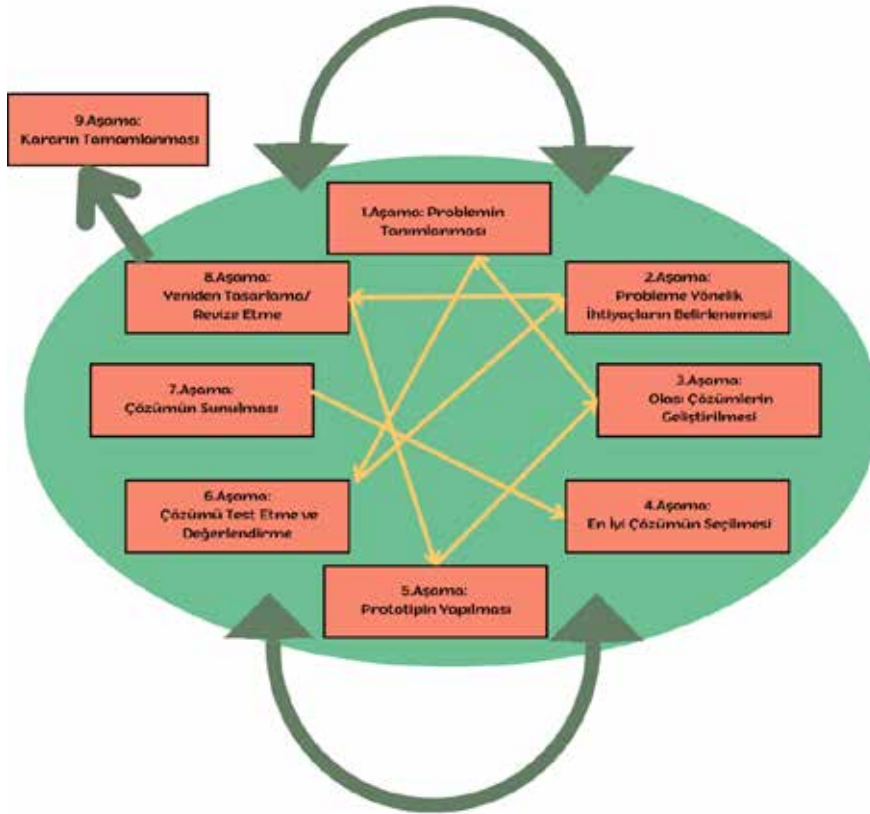
STEM eğitimi, disiplinlerin birbiriyle ilişkilendirilerek uygulanması gereken bir eğitim yaklaşımıdır. Bu yüzden STEM disiplinlerinin tamamına vurgu yapılmasını öngören bütüncül yaklaşım, STEM eğitimi için en uygun öğretim yaklaşımıdır [3] [17]. Bütüncül yaklaşım fen bilimleri, teknoloji, mühendislik ve matematik disiplinlerinin bütünleştirilmesi olarak tanımlanmaktadır [18]. Eğitim kurumlarında yer alan öğrenme ortamlarının ve öğretim içeriğinin disiplinler arası ilişkiler kurulmasına uygun olmaması ve disiplin içeriklerinin tamamının öğretimini gerçekleştirecek öğretmen yetiştirme programlarının bulunmaması gibi nedenler, bütüncül yaklaşımın uygulanmasını güçleştirmektedir. Bu nedenle günümüzde bütüncül yaklaşımın yanında fen bilimleri, matematik gibi derslere diğer disiplinlerden en az birinin entegre edilmesine yönelik çalışmalar ön plana çıkmaktadır [3][19][20].

STEM eğitiminde disiplinler arası bütünleştirme; bağlam bütünleştirmesi, içerik bütünleştirilmesi ve destekleyici içerik bütünleştirilmesi olmak üzere üç şekilde sağlanabilir. İçerik bütünleştirilmesi, birden fazla STEM disiplinlerinin öğrenme hedeflerine sahip ünite ve aktivitelerine karşılık gelir. Destekleyici içerik bütünleştirilmesi, ana içeriğin öğrenme hedeflerini desteklemek için başka bir içeriğin ünite ve aktivitelerinin dâhil edilmesini kapsar. Bağlam bütünleştirmesinde ise odak tek bir disiplindedir ve diğer disiplinlerden öğrenme hedefi içermez. Problem için diğer disiplinler sadece bağlam sağlar [21]. STEM eğitiminde bağlam bütünleştirilmesi uygulanırken, dört disiplinden bir tanesi ana disiplin olarak seçilerek odağa alınarak uygulanır. Bütünleştirme ihtiyaca ve içeriğe göre dört disiplinin hepsinin yerine en az ikisinin birleştirilmesiyle uygulanabilir [22]. STEM eğitimi içinde birçok öğretme ve öğrenme yaklaşımı kullanılabilir. Literatürde yer alan STEM eğitimi için kullanılan pedagojik yaklaşımlar [23] Görsel 3.2’de verilmiştir.



Görsel 3.2: STEM eğitiminde kullanılan bazı pedagojik yaklaşımlar

Farklı pedagojik yaklaşımlarının STEM eğitiminin amaçlarının bireylere kazandırılmasına katkı sağlayacağı düşünülmektedir (Görsel 3.2). Bu pedagojik yaklaşımlardan biri olan tasarım temelli yaklaşımın günlük yaşam problemlerinin çözümünde seçilmesi, MEB programlarında amaçlandığı üzere bireylerin mühendislik becerilerini tanımaya yardımcı olmaktadır. Mühendislik birçok disiplin ile bağlantılıdır. Mühendisler problemleri çözerken fen bilimleri, matematik ve teknoloji gibi disiplinleri kullanır [24]. Mühendislikte karşılaşılan problemler mühendislik tasarım süreci ile yürütülmektedir. Mühendislik tasarım süreci Görsel 3.3'te verilmiştir [19] [25].



Görsel 3.3: Mühendislik tasarım süreci [18] [23]

Mühendislik tasarım süreci tek yönlü değil birbiri ile etkileşim hâlindeki aşamalardan oluşan bir süreçtir. Süreç gerçek yaşamdan bir ihtiyaç ya da problemin tanımlanmasıyla başlar. Problem durumunun tanımlanabilmesi için kısıtlamalar ve kriterler içermesi gerekir. Mühendisler tek bir çözüme odaklanmazlar. Çözüm geliştirebilmek için araştırma yaparak nelere ihtiyaç duyabileceklerini tespit ederler. Mühendislik tasarım sürecinde gerçek yaşam problemlerine genellikle birçok çözüm bulunur. Bu nedenle mühendisler tasarım kriterleri ve kısıtlamaları kapsamında yaptıkları araştırmalardan yola çıkarak yaratıcı çözüm önerileri üretmeye çalışır. Mühendisler, birçok olası çözümün tanımlanmasının ardından tasarımın kriterlerine ve sınırlılıklarına yönelik analizler yaparak olası çözümleri değerlendirir ve en iyi çözümü seçerler. En iyi çözüm seçildikten sonra mühendisler tasarımlarını görsel olarak sunmak veya ayrıntılarını göstermek için prototip oluştururlar. Çözüm, oluşturulan prototip aracılığıyla test edilir ve değerlendirilir. Tasarım sürecinin aşamalarının tamamlanmasının ardından mühendisler tasarımlarını birbiriyle paylaşabilir. Burada fikir paylaşımının yanı sıra

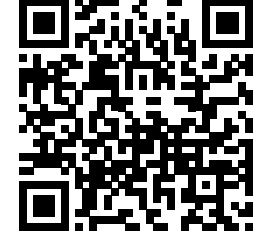
geri bildirim alma ve tasarımın pazarlanabilmesi önemli görülen noktalardır. Mühendisler yaptıkları değerlendirmeler ve aldıkları geri bildirimlerin ardından çözümlerinde iyileştirme yapmak için yeniden tasarlama aşamasına geçerler. Son olarak tasarımın en iyi çözüm olup olmadığına karar vererek kararın tamamlanması aşaması ile süreci bitirirler [19][20] [25].

Mühendislik tasarım temelli eğitim; bireylerin problem çözme, eleştirel düşünme, üst düzey düşünme becerilerinin geliştirilmesini sağlayan, aynı zamanda disiplinler arası bakış açısı kazandıran, STEM'in merkezinde olması gereken bir süreçtir [3]. Mühendislik tasarım temelli eğitim; bireylerin mühendisler gibi disiplinler arası düşüncelerine, gerçek yaşam problemlerine çözümler üretmelerine, onların 21. yüzyıl becerilerinin gelişmesine ve STEM meslekleri hakkında fikir sahibi olmalarına katkı sağlayabilir.

Bu bağlamda bu çalışmanın STEM öğrenme senaryoları mühendislik problemleri kullanılarak mühendislik tasarım süreci kapsamında geliştirilmiştir.

Akıllı Boy Ölçer

Seviye	12 yaş ve üzeri
Konu	Nesnelerin İnterneti
Süre	6 ders saati



Planla ilgili dosyalara ulaşmak için karekod taratılabilir veya bağlantı kullanılabilir.

Özet

Bu öğrenme senaryosu ultrasonik sensör verisini çevrimiçi tablolara kaydedebilen bir boy ölçüm aracının tasarlanması üzerine odaklanır. Bilgi temelli hayat problemi, gerçek hayatta okullarda uygulanan "sağlıkta fiziksel uygunluk kartesinden" yola çıkılarak oluşturulmuştur. Öğrenme senaryosunun geliştirilmesinde mühendislik tasarım süreci kullanılmıştır. Öğrenme senaryosu problemin tanımlaması amacıyla öğrenciler tarafından okunur ve öğrenme süreci başlar. Öğrenciler süreç boyunca takımlar hâlinde çalışır. Öğrencilerden senaryo sonucunda onların boyunu hızlı, hatasız ölçen ve ölçüm sonuçlarını elektronik tablolara otomatik olarak kaydeden bir prototip oluşturmaları beklenir. Burada amaç; öğrencilerin bilişim teknolojileri ve matematik derslerinde öğrendiklerini, mühendislik becerilerini kullanarak bir tasarım ürünü oluşturmalarıdır. Ayrıca öğrencilerin farklı disiplinlerini günlük yaşamla ilişkilendirerek sorunları çözmek için kullanılabileceklerini fark etmelerini sağlamaktır. Prototip oluşturma sürecinde öğretmenler öğrencilere kodlama adımlarında rehberlik edecektir. Sürecin sonunda öğrenci takımları tasarımlarını sınıf arkadaşlarına sunacaklar. Öğrenci takımları ve prototip öğretmen tarafından hazırlanan ölçek ile değerlendirilecektir (Ek-1). Ayrıca öğrenciler öz değerlendirme ölçeğini kullanarak kendilerini değerlendireceklerdir (Ek-2).

STEM Konu Alanı 1

Bilişim Teknolojileri (Kazanımlar)

- Problem durumunu çözmek için uygun algoritmayı geliştirir.
- Arduino ile devre kurulumu yapar.
- Projelerine uygun yapay zekâ uygulamalarını seçer.
- Sensörlerden alınan verinin tabloya aktarılmasını sağlar.
- Blok tabanlı programlama aracını kullanarak program geliştirir.

STEM Konu Alanı 2

Akıllı Boy Ölçer

Mühendislik (Kazanımlar)

- Mühendislik ve tasarım ilişkisini ifade eder.
- Çevresindeki ürünleri mühendislik ve tasarım kavramları açısından ilişkilendirir.
- Mühendislik tasarım sürecindeki sınırlılıkları değerlendirir.
- Mühendislik tasarım sürecini kullanarak bir ürün tasarlar.
- Bir mühendislik projesinin içerdiği süreçleri tespit eder: Planlama, tasarım, prototip oluşturma, yürütme, kalite kontrol ve raporlama gibi aşamaları açıklar.
- Öğrenci proje çalışmasında kendisini farklı rollerdeki bir takım üyesi olarak görür, o rolün gerektirdiği çalışmaları başarıyla tamamlar.

- Çözüm önerisine yönelik düşüncelerini yazarak ve çizerek açıklar.
- Başkalarının çözüm önerisine yönelik düşüncelerini dikkate alır.

Sosyal Ürün Kazanımları

- Öğrenci tasarlanan ürünü açık ve anlaşılır şekilde sınıfa sunar.

STEM Konu Alanı 3

Fen Bilimleri (Kazanımlar)

- Sesin yansıması ve soğurulmasına örnek verir.

STEM Konu Alanı 4

Matematik (Kazanımlar)

- Veri toplamayı gerektiren araştırma soruları oluşturur.
- Uzunluk ölçme birimlerini tanıır; metre-kilometre, metre-desimetre-santimetre-milimetre birimlerini birbirine dönüştürür ve ilgili problemleri çözer.
- Araştırma sorularına ilişkin verileri toplar, sıklık tablosu ve sütun grafiğiyle gösterir.

Materyaller

Devre Malzemeleri

- 1 Adet Arduino UNO R3
- 1 Adet Ultrasonik mesafe sensörü
- 1 Adet Orta boy bread board
- 1 Adet Erkek-erkek jumper kablo

Online Araçlar

- <https://www.tinkercad.com/>

STEM Kariyer Bağlantıları

- Yapay zekâ mühendisliği
- Veri analizi
- Veri bilimi
- Bilgisayar mühendisliği
- Bilgisayar programcılığı

Akıllı Boy Ölçer

ETKİNLİK SÜRECİ

Bu etkinlik mühendislik tasarım süreci kullanılarak oluşturulmuştur. Öğretmen etkinlik sürecinin başlangıcında öğrencileri takımlara ayırarak çalışmalarına rehberlik eder.

Problem Tanımlanması

Problem tanımlanırken mutlaka tasarımın kriterleri ve sınırlılıkları belirlenmelidir.

Problem senaryosunda yer alan kriterler ve sınırlılıklar: Öğrencilerin, temel kodlama bilgisine sahip olması, yapay zekâ uygulamalarını tanınması ve Arduino ile devre tasarlaması gibi temalar olarak ortaya çıkmıştır. Öğretmen tarafından ihtiyaç ve problem durumu öğrencilere aşağıda yer alan bilgi temelli hayat problemi ile verilir.

Bilgi Temelli Hayat Problemi

Millî Eğitim Bakanlığı (MEB) ve Sağlık Bakanlığı öğrencilerin sağlıklı beslenmelerini sağlamak ve fiziksel aktivitelerini arttırmak amacıyla ortak bir çalışma başlatmıştır. Ortak Bilim Kurulunun almış olduğu kararlar doğrultusunda okullarda Sağlıkla İlgili Fiziksel Uygunluk Karnesi uygulaması başlatılmıştır [1]. Bu uygulamaya göre bir eğitim öğretim yılında öğrencilerin iki kez ağırlıkları ve uzunlukları ölçülerek değerlendirilmektedir.

Mert Öğretmen, bir ortaokulda Beden Eğitimi ve Spor öğretmenidir. Sağlıkla İlgili Fiziksel Uygunluk Karnesi uygulaması için okuldaki tüm öğrencilerin boylarını ve ağırlıklarını ölçülecektir. Mert Öğretmen öğrencilerin boyunu metre ile ölçmekte ve ölçüm sonucunu öğrenci listesinde adı-soyadı sütununun yanına yazmaktadır. Mert öğretmen daha kolay bir yöntemle tüm öğrencilerin boylarını ölçerek, ölçüm sonuçlarını bir tabloya kaydetmek ve bu tabloyu bir grafik ile incelemek istiyor. Mert öğretmenin öğrencilerin boylarını hızlı bir şekilde ölçmesine yardım edebilir misiniz?

Mert Öğretmen tarafından boy ölçmek için oluşturulacak cihazın tasarım kriterleri şu şekilde belirlenmiştir:

- Öğrencilerin boylarını hızlı ve hatasız olarak ölçmesi,
- Ölçüm sonuçlarını elektronik tablolara otomatik olarak kaydetmesi.

Probleme Yönelik İhtiyacın Belirlenmesi

Mühendisler akla gelen ilk çözümü sunmak yerine problem hakkında araştırma yaparak ve geçmişte kullanılan yöntemleri gözden geçirerek farklı çözüm yollarını bulmaya çalışırlar [2].

Bu aşamada öğrenciler çözüme yönelik ihtiyaçları belirlemek için elektronik boy ölçme sistemleri ve internet üzerinden tablolara veri kaydetmeyi araştırarak bilgi toplarlar. Araştırmalar sonucunda boy ölçmek için robotik kodlama uygulamalarına ve internet üzerinden veri kaydetmek için Google E-Tablolar gibi uygulamalara, sesli yönlendirmeler için de yapay zekâ uygulamalarına ulaşabileceklerdir.

Olası Çözümlerin Geliştirilmesi

- Günlük yaşam problemlerinde tek bir doğru çözüm yoktur, genellikle birden fazla çözüm vardır [3]. Bu aşamada öğrenciler tasarım kriterlerini ve sınırlılıklarını dikkate alarak edindikleri bilgilere göre probleme yönelik çözüm önerileri geliştirirler. Öğrencilerin ürettikleri çözüm önerileri tabloya yazılır (Tablo 3.1).

Akıllı Boy Ölçer

Tablo 3.1: Çözüm Önerileri ve Sınırlılıklar

Çözüm Önerileri	Sınırlılıklar
1.	
2.	
3.	

Öğrenciler olası çözümlerini takım arkadaşlarıyla birlikte belirleyeceklerdir. Beyin fırtınası ile öğrencilerin oluşturdukları çözüm önerileri arasından problemin sınırlılıklarını dikkate alarak en uygun çözüme ulaşmaları beklenir. Öğrenciler oluşturmuş oldukları çözüm önerilerinin sınırlılıklarını beyin fırtınası ile kendileri belirleyeceklerdir. Öğretmen burada öğrencilerin beyin fırtınası yapmasını teşvik eder ve öğrenciler ile çözüm önerilerini değerlendirir. Öğrencilerin takım arkadaşlarıyla en uygun çözümü seçmelerine rehberlik eder.

En İyi Çözümün Seçilmesi

Öğrencilerin birçok çözüm önerisini takım arkadaşlarıyla tanımlamasından sonra problem durumunun çözümüne yönelik bilgi temelli hayat probleminde yer alan kriterler ve sınırlılıklar çerçevesinde yapılan analizlerle en uygun çözüm için karar vermeleri beklenir. En uygun çözümün seçilmesinin ardından öğrencilerden mühendislik becerilerini geliştirmeye katkı sağlamak amacıyla çözümü iki boyutlu olarak bir kâğıda çizmeleri beklenir. Yapılan çizim üzerinde öğrenciler kullanacakları malzemeleri seçerek göstermelidir. Daha sonra öğrencilerden hazırlayacakları prototipin simülasyonunu gözlemlenmeleri için Tinkercad üzerinden taslak devre modellerini oluşturmaları istenir. Tasarım fikri için öğrencilerin Arduino ve ultrasonik sensör ile devre tasarlaması, yapılan ölçümlerin mBlock yardımıyla Google e-Tablolara kaydetmesi beklenmektedir.

Problem durumu analiz edildiğinde öğrencilerin hazır bulunuşlukları da dikkate alınarak problemi çözmeye yönelik geliştirilecek bir projenin üç ana bileşeni olduğu görülmektedir:

- 1)Devre Tasarımı
- 2)Tablo oluşturma
- 3)Kodlama

Prototipin Yapılması

Bu aşamada seçilen çözüm önerisine yönelik tasarımı sunmak ve ayrıntılarını göstermek için prototip oluşturulur.

Ön Hazırlık

Bu aşamada öğretmen ultrasonik mesafe sensörünü ve diğer aygıtları tanıtır. Ultrasonik mesafe sensörünün ses dalgaları aracılığıyla sesin yansımaları özelliğini kullanarak çalışması hakkında bilgi verir. Burada fen bilimleri ile teknoloji entegrasyonunun sağlanması amacıyla sesin yansıtılmasından ve soğurulmasından bahsedilir.

Ses kaynağından çıkan ses dalgalarının bir maddeye çarpıp geri dönmesi olayına sesin yansımaları denir. Boş bir odada şiddetli duyulan sesler, odada eşya olduğunda aynı şiddette duyulmaz. Bunun sebebi boş odadaki sesin daha fazla yansıyarak oda içine geri dönmeleridir. Oysa eşya dolu bir odada ses farklı yönlerde dağılmaktadır ve bazı eşyalar tarafından ses soğurulmaktadır. Eşya dolu odada ses daha az şiddetli duyulur. Ultrasonik mesafe sensörü belirlenen nesneye doğru ses dalgalarının yüksek frekansla iletilmesi biçiminde ses dalgası yayılır. Nesnelerin yansıtılması sonucunda (eko) ses dalgaları mesafe ölçüm sensörü bölgesine geri döner. Ultrasonik sensörün alıcı bölümüne gelen ses dalgasını

Akıllı Boy Ölçer

algılar . Burada ses dalgaları, geri dönüş için kullanılan zaman aralığını mesafe ölçümü olarak kabul eder.

Aygıtın Bağlanması

Aygıtlar bölümünde ilgili Arduino kartı (Bu projede UNO 3 kullanılmıştır.) seçilerek projeye dâhil edilir. Arduino bilgisayara bağlanarak aygıtlar bölümünden mBlock ile bağlantısı gerçekleştirilmelidir. Yükleme işlemi yapılmalıdır.

mBlock Uzantıları

Uygulamanın geliştirebilmesi için gerekli eklentiler yüklenmelidir.

Tablo 3.2: Kullanılan Eklentiler

Eklentinin Adı	Yüklendiği Bileşen	Kullanım Amacı
 Text to Speech Geliştiriciler: MIT Media Lab	Kuklalar	Metinleri seslendirmek için kullanılır. Bu eklenti ile programda sesli mesaj verilebilir.
 Yükleme modu yayını By mBlock official Bu uzantıyı ekleyerek, bir aygıtın.	Kuklalar Aygıtlar	Kukla ile aygıt arasındaki iletişimi sağlayarak Arduino ile kuklanın haberleşmesini sağlar.
 Google sayfa	Kuklalar	Arduino'dan alınan verinin Google E-Tablolara kaydedilmesini sağlar.

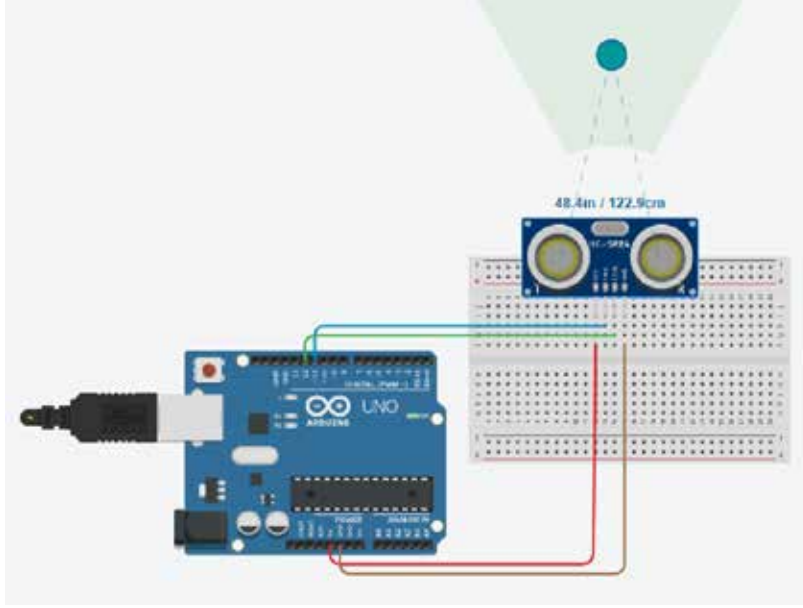
Devre Tasarımı ve Kurulumu

Arduino ve ultrasonik sensör ile bir mesafe ölçer tasarlanır.

Tinkercad ile Devre Tasarımı

Bu aşamada öğrenciler, Tinkercad platformunu kullanarak boy ölçmek için kullanacakları devreyi tasarlarlar (Görsel 3.4). Simulator kullanarak devrenin tasarlanması gerçek devre kurulumunda karşılaşılabilecek hataların ve arızaların önüne geçilmesini sağlamaktadır.

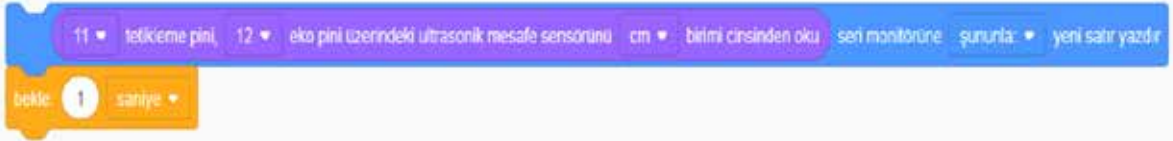
Akıllı Boy Ölçer



Görsel 3.4: Tinkercad devre şeması

Devre Tasarımının Test Edilmesi

Devre tasarımı yapıldıktan sonra kod bölümünde sistemi test etmek için kodlama yapılır. Kod bölümünde aşağıdaki kod blokları kullanılmalıdır (Görsel 3.5).



Görsel 3.5: Tinkercad blok kodlama

Kodlama yapıldıktan sonra simülasyon başlatılır ve seri monitörde uzaklık ölçümü sonuçlarının görülüp görülmediği test edilir. Bir sorunla karşılaşılmazsa tasarlanan devre kurulumuna geçilir.

Devre Kurulumu

Yapılan devre tasarımına uygun bir biçimde devre kurulumu yapılır. HC-SR04 ultrasonik sensördeki trig pini Arduino D11 numaralı pine, echo pini ise Arduino D12 numaralı pine takılmıştır. SR04 ultrasonik sensördeki GND ve VCC pinleri de sırasıyla Arduino'nun GND ve 5V pinlerine takılmıştır.

Tablo Oluşturma

Tablo oluşturma: Ölçüm sonuçlarının kaydedileceği uygun bir tablo tasarlamak için (mBlock ile uyum sağlaması nedeniyle) Google E -Tablolar tercih edilmiştir. Bu amaçla Google Drive üzerinde yeni bir Google E-Tablo oluşturulur. Tabloya üç sütun tanımlanır (Görsel 3.6).

Akıllı Boy Ölçer

	A	B	C
1	Öğrenci Adı Soyadı	Cinsiyet	Boy (cm)
2	Emine	K	
3	Murat	E	
4	Emre	E	
5	Elif	K	
6	Sultan	K	
7	Mustafa	E	
8	Seher	K	
9	Ahmet	E	
10	Mehmet	E	
11	Altan	E	
12	Ela	K	
13	Kaya	K	

Görsel 3.6: Öğrenci uzunluk tablosu

İlk iki sütundaki veri girilmiş olmalıdır. Bu aşamada öğrenci bilgi sistemlerinden alınan tablolardaki ilgili sütunlar kopyalanarak alınabilir veya tek tek giriş yapılır. Boy sütununa veri girişi ise boy ölçer devresi üzerinden mBlock ile kodlanarak otomatik olarak yapılacaktır.

Kodlama

Bu bölümde prototipe ait kod bloklarına yer verilmiştir. Kod blokları; aygıt kodları ve kukla kodları olmak üzere 2 bölümde verilmiştir.

Aygıt Kodları

Arduino kartı eklendikten sonra aygıtta aşağıdaki kod blokları eklenir (Görsel 3.7).



Görsel 3.7: Arduino kod blokları

Kodlarının Açıklaması

1. Arduino UNO başlatıldığında.
2. Sürekli tekrarlanır.
3. Arduino üzerinde 11 numaralı pin ultrasonik sensörde Trig girişine, D12 numaralı pin de Echo girişine bağlı olduğu için mesafe algılayıcı kod bloku D11 ve D12 pin numaralarıyla tanımlanmıştır. Algılanan mesafe engele olan uzaklığın santimetre olarak değeridir. Boy değişkenine değer olarak atanmıştır. Tasarlanan boy ölçerin zeminden 200 cm yüksek bir yere sabitleneceği düşünülerek boy değişkenine hesaplanan yeni değer (200-mesafe) atanmıştır.
4. Ölçümde sürekli dalgalanmalar olmasın diye 1 sn. bekleme kod bloku eklenmiştir.

Akıllı Boy Ölçer

5. Yükleme modunda boy_cm iletisi değer olarak da boy değişkeninin değeri gönderilmiştir.

Kukla kodları

Yeşil Bayrak tıkladığında çalışacak kodlar (Görsel 3.8)



Görsel 3.8: Yeşil Bayrak tıkladığında çalışacak kodlar

Kodların Açıklaması

1. Yeşil bayrak tıkladığında.
2. E tabloda satırları okumak için kullanılan kayıt_no değişkenine 1 değeri atanır.
3. Bu bloka sayfa adresi olarak önceden oluşturulan ve paylaşım yapılarak (Bağlantıya sahip olan herkes düzenleyen olarak seçilmiş.) bağlantısı alınan adres yapıştırılmalıdır. Bu işlem yapılmazsa tabloya kayıt yapılamaz.

Yükleme modu iletisi alınırken çalışacak kodlar (Görsel 3.9)



Görsel 3.9: Yükleme modu iletisi alınırken çalışacak kodlar

1. boy_cm değişkeni değeriyle yükleme modu iletisi alınırken
2. Ultrasonik sensör ile yapılan ölçümler yükleme modu iletisi olarak geldiğinde boy_cm adında yeni bir değişkene atanır.
3. Ölçümdeki ani değişikliklerden kaçınmak için 2 sn. bekletilir.

Boşluk tuşuna basıldığında çalışacak kodlar (Görsel 3.10)



Görsel 3.10: Boşluk tuşuna basıldığında çalışacak kodlar

Akıllı Boy Ölçer

1. Boşluk tuşuna basıldığında: Tabloya kayıt işleminin yapılması işlemi bir tuşun basılması olayına bağlanarak kullanıcı kontrolüne veriliyor. Buradaki amaç ölçümün net olarak görülerek kullanıcı kontrolüyle yapılmasıdır.
2. Kayıt işlemi Google sayfa eklentisindeki giriş bloku kullanılarak yapılmaktadır. Bu blokta sırayla hücreye girilecek değer (boy_cm), sütun numarası (3) ve satır numarası (kayıt_no+1) parametreleri yer almaktadır.
3. Yapay zekâ uzantılarından biri olan metni sese çevirme işlemleri için gerekli dil ayarlaması dil ayarla bloku kullanılarak yapılıyor. Konuşa bloku kullanılarak kayıt işlemi tamamlandığı için ölçümün kaydedildiğine dair bir sesli mesaj veriyor.
4. Konuşa bloku kullanılarak yapılan kayda ve değere ilişkin bilgiler sesli bir şekilde ifade ediliyor.
5. Sıradaki kayda geçmek için kayıt numarası bir artırılıyor.
6. Sıradaki öğrenci sesli bir şekilde anons ediliyor.

*Kayıt numarasının 2'den başlamasının nedeni tabloda ilk satırda sütun adlarının bulunmasıdır.

Çözümü Test Etme ve Değerlendirme

Hazırlanan devre bir kutu veya farklı aparatlar kullanılarak düz bir zeminden 200 cm yükseğe (bir kapı kirişi olabilir) sabitlenir. Öğrenciler bu aşamada ürün için üç boyutlu bir aparat veya kutu tasarlayabilir. Tinkercad gibi programlarla ürünü tasarlayıp üç boyutlu yazıcıyla basabilir. Sabitlenen boy ölçer uygun bir bağlantı kablosuyla bilgisayara bağlanır. mBlock programında kodlar çalıştırılır. İlk öğrenci boy ölçerin altına geçer. Ekranda öğrencinin boyu görünür ve boşluk tuşuna basılır. Kayıt yapıldığına dair sesli mesaj alınır ve sıradaki öğrencinin ismi anons edilirse sistemin çalıştığı söylenebilir. Google e-tablolardaki dosya açılarak kaydın yapılıp yapılmadığı kontrol edilebilir. Öğrenciler verileri elde ettikten sonra sütun grafiği çizerek gösterir.

Çözümün Sunulması

Bu aşamada öğrenciler takım arkadaşlarıyla birlikte oluşturdukları ürünleri diğer takımlara sunar. Öğretmen, sunum sırasında süreç değerlendirme formunu kullanarak öğrencilerin tasarımlarını ve uygulama sürecini değerlendirir. Öğretmen değerlendirme ölçeği Ek-1'de verilmiştir.

Yeniden Tasarlama / Revize Etme

Öğrencilerin aldıkları değerlendirmeler ve geri bildirimler doğrultusunda çözümlerinde gerekli iyileştirmeleri yaptıkları aşamadır. Öğrencilerin bu aşamada problemin kriter ve sınırlılıkları çerçevesinde tasarımlarını en iyi olabilecek şekilde iyileştirmeleri beklenir. Burada öğrencilerin oluşturmuş olduğu tasarım üründe iyileştirmenin nasıl yapılacağı, neden bu iyileştirmenin yapıldığı ve proje etkisini özetleyen takım hâlinde hazırlanmış bir rapor oluşturmaları istenebilir.

Kararın Tamamlanması

Kararların tamamlanmasında ise öğrencinin nihai tasarımının en iyi olup olmadığına kendisinin karar vermesi amacıyla Ek 2'de verilen öğrenci değerlendirme ölçeği kullanılacaktır.

Öğrenciler grup üyeleriyle birlikte aşağıda yer alan soruları cevaplandırarak proje değerlendirmesi yaparlar.

Akıllı Boy Ölçer

1. Bu etkinliđi yaparken en çok nerede zorluk yaşadınız? Bu zorlukları nasıl aştınız?
2. Bu etkinliđi yaparken en çok sevdiğiniz kısım hangisiydi?
3. Bu etkinlikten neler öğrendiniz?
4. Bu projeyi yeniden planladığında geliştirmek istediđin yönler nelerdir?

EK-1: Öğretmen Deđerlendirme Ölçeđi

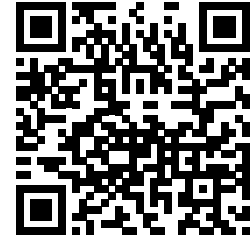
Deđerlendirme Kriterleri	GELİŐTİRİLMELİ	İYİ	ÇOK İYİ
Sorunu tanımlama ve analiz etme			
Olası çözümleri bulma ve en iyisini seçme			
Tasarım ürününün kabataslak çizimi veya simüle edilmiş formu			
Grup çalışması ve iş birliđi			
Ürünü sunma			

EK-2: Öğrenci Öz Deđerlendirme Ölçeđi

Deđerlendirme Kriterleri	Katıl mıyorum	Kararsızım	Katılı yorum
Sensörlerden alınan verinin tablolara nasıl kaydedileceđi hakkında bilgi sahibi oldum.			
Bilgi temelli hayat problemine yönelik bulduğum çözüm en uygun çözümdür.			
Senaryo sürecinde takım arkadaşlarımla iş birliđi içinde bulunarak takım arkadaşlarıma katkı sağladım.			
Ürünü tasarlarken kriter ve sınırlılıkları dikkate aldım.			
Senaryo sürecini anlatan bir rapor oluşturdum.			

Duygu Durumu Analizi

Seviye	12 yaş ve üzeri
Konu	Metinden duygu durumu analizi
Süre	6 ders saati



Planla ilgili dosyalara ulaşmak için karekod taratılabilir veya bağlantı kullanılabilir.

Özet

Bu öğrenme senaryosu bir mağazanın müşteri memnuniyetini artırmak için onların ürün ve hizmetlerden memnuniyet durumlarını belirlemek istemesine dayalıdır. Öğrenme senaryosu gerçek hayatta özellikle çevrim içi alışveriş sitelerinde müşteri memnuniyetini belirlemek için kullanılan bilgi temelli hayat problemi ile başlar. Öğrenme senaryosunda mağazadan alışveriş yapan müşterilerin yorumları metinden duygu durumu analizi yapılarak sınıflandırılacaktır. Öğrencilerin bir metin sınıflandırıcı modeli eğiterek bilgi temelli hayat problemine çözüm üretmesi beklenmektedir. Öğrenme senaryosunun geliştirilmesinde mühendislik tasarım süreci kullanılmıştır. Öğrenme senaryosu problemin tanımlaması amacıyla öğrenciler tarafından okunur ve öğrenme süreci başlar. Öğrenciler süreç boyunca takımlar hâlinde çalışır. Prototip oluşturma sürecinde öğretmenler öğrencilere kodlama adımlarında rehberlik edecektir. Sürecin sonunda öğrenci takımları tasarımlarını sınıf arkadaşlarına sunacaktır. Öğrenci takımları ve prototip öğretmen tarafından hazırlanan ölçek ile değerlendirileceklerdir (Ek-1). Ayrıca öğrenciler öz değerlendirme ölçeğini kullanarak kendilerini değerlendireceklerdir (Ek-2).

STEM Konu Alanı 1

Bilişim Teknolojileri (Kazanımlar)

- Problem durumunu çözmek için uygun algoritmayı geliştirir.
- Projelerine uygun yapay zekâ uygulamalarını seçer.
- Blok tabanlı programlama aracını kullanarak program geliştirir.

STEM Konu Alanı 2

Mühendislik (Kazanımlar)

- Mühendislik ve tasarım ilişkisini ifade eder.
- Çevresindeki ürünleri mühendislik ve tasarım kavramları açısından ilişkilendirir.
- Mühendislik tasarım sürecindeki sınırlılıkları değerlendirir.
- Mühendislik tasarım sürecini kullanarak bir ürün tasarlar.
- Bir mühendislik projesinin içerdiği süreçleri tespit eder: Planlama, tasarım, prototip oluşturma, yürütme, kalite kontrol ve raporlama gibi aşamaları açıklar.
- Öğrenci proje çalışmasında kendisini farklı rollerdeki bir takım üyesi olarak görür, o rolün gerektirdiği çalışmaları başarıyla tamamlar.
- Çözüm önerisine yönelik düşüncelerini yazarak ve çizerek açıklar.
- Başkalarının çözüm önerisine yönelik düşüncelerini dikkate alır.

1.3. Sosyal Ürün Kazanımları

- Öğrenci tasarlanan ürünü açık ve anlaşılır şekilde sınıfa sunar.

Duygu Durumu Analizi

STEM Konu Alanı 3

Matematik (Kazanımlar)

- Veri toplamayı gerektiren araştırma soruları oluşturur.

Materyaller

Çevrim içi araçlar

- <https://about.draw.io/>
- <https://pictoblox.ai/>

STEM Kariyer Bağlantıları

Yapay zekâ mühendisliği

Veri analizi

Veri bilimi

Bilgisayar mühendisliği

Bilgisayar programcılığı

ETKİNLİK SÜRECİ

Bu etkinlik mühendislik tasarım süreci kullanılarak oluşturulmuştur. Öğretmen etkinlik sürecinin başlangıcında öğrencileri takımlara ayırarak çalışmalarına rehberlik eder.

Problem Tanımlanması

Problem tanımlanırken mutlaka tasarımın kriterleri ve sınırlılıkları belirlenmelidir.

Problem senaryosunda yer alan kriterler ve sınırlılıklar: Öğrencilerin, temel kodlama bilgisine sahip olması, yapay zekâ uygulamalarını tanınması ve Arduino ile devre tasarlaması gibi temalar olarak ortaya çıkmıştır. Öğretmen tarafından ihtiyaç ve problem durumu öğrencilere aşağıda yer alan bilgi temelli hayat problemi ile verilir.

Bilgi Temelli Hayat Problemi

Umay Hanım, internet üzerinden hizmet veren bir alışveriş mağazasında satış departmanında yöneticidir. Umay Hanım mağazada sunulan ürünler ve verilen hizmetlerle ilgili müşterilerin memnuniyetini artırıcı çalışmalar yapmak istiyor. Bunun için de müşterilerin hangi ürün veya hizmetlerden memnun kaldıklarını, hangilerinden memnun kalmadıklarını bilmesi gerekiyor. Mağazanın web sitesinde ürünler ve hizmetlerle ilgili müşterilerin yüzlerce yorumu ve mesajı bulunmaktadır. Ancak her ürün ve hizmet için bu yorumları tek tek okumak ve analiz etmek zaman ve iş gücü açısından çok zor görünmektedir. Üstelik bu şekilde analizler çok uzun zaman alacağından müşterilerin memnun kalmadığı konuları belirlemede gecikme de yaşanabilir. Bu gecikme nedeniyle mağaza ürün veya hizmetten memnun kalmayan müşterilerini kaybetme riskiyle karşı karşıya kalabilir. Yaptığı araştırmalar sonucunda günümüzde birçok mağazanın müşteri memnuniyetini artırmak için yapay zekâ yöntemlerini kullandığını görmüştür.

Umay Hanım, bu konuda yapay zekâ teknolojileri kullanılarak bir çözüm üretilebilir mi diye veri bilimci Elif Hanımdan profesyonel destek almaya karar verir.

Elif Hanım, yapay zekâ teknolojileri kullanılarak metinlerden duygu analizi yapılabildiğini anlatır. Özellikle sosyal medyada belirli konu başlıkları (hashtag) altında insanların paylaşımlarının analiz edilerek o konuya karşı duygularının belirlendiğinden bahseder. Web sitesi üzerinde müşterilerin mesajlarını, yorumlarını belirli duygu durumlarına göre sınıflandırabileceklerini (olumlu, olumsuz ve ne olumlu ne de olumsuz) söyler. Sizce veri bilimci Elif Hanım soruna yönelik nasıl bir uygulama gerçekleştirebilir?

Duygu Durumu Analizi

Elif Hanım tarafından geliştirilecek programın kriterleri şu şekilde belirlenmiştir:

- Müşterilerin yorumlarını olumlu, olumsuz ve tarafsız olarak sınıflandırmalı.

Probleme Yönelik İhtiyacın Belirlenmesi

Mühendisler, akla gelen ilk çözümü sunmak yerine problem hakkında araştırma yaparak geçmişte kullanılan yöntemleri gözden geçirerek farklı çözüm yollarını bulmaya çalışırlar [1].

Bu aşamada öğrenciler, çözüme yönelik ihtiyaçları belirlemek için duygu analizi ve metinden duygu durumu analizini araştırarak bilgi toplar. Öğrencilerin araştırma yapmaları için aşağıdaki sorular sorulur.

- Bilgisayarlar insan dilini nasıl anlar?
- Bilgisayarlar bir cümlenin olumlu, olumsuz veya tarafsız bir yargı içerip içermediği nasıl analiz edebilir?
- Duygu analizi nasıl yapılır ve hangi alanlarda kullanılabilir?

Olası Çözümlerin Geliştirilmesi

Günlük yaşam problemlerinde tek bir doğru çözüm yoktur, genellikle birden fazla çözüm vardır [2]. Bu aşamada öğrenciler tasarım kriterlerini ve sınırlılıklarını dikkate alarak edindikleri bilgilere göre probleme yönelik çözüm önerileri geliştirirler. Öğrencilerin ürettikleri çözüm önerileri tabloya yazılır (Tablo 1).

Tablo 3.3: Çözüm Önerileri ve Sınırlılıklar

Çözüm Önerileri	Sınırlılıklar
1.	
2.	
3.	

Öğrenciler, olası çözümlerini takım arkadaşlarıyla birlikte belirleyeceklerdir. Beyin fırtınası ile öğrencilerin oluşturdukları çözüm önerileri arasından problemin sınırlılıklarını dikkate alarak en uygun çözüme ulaşmaları beklenir. Öğrenciler oluşturmuş oldukları çözüm önerilerinin sınırlılıklarını beyin fırtınası ile kendileri belirleyeceklerdir. Öğretmen burada öğrencilerin beyin fırtınası yapmasını teşvik eder ve öğrenciler ile çözüm önerilerini değerlendirir. Öğrencilerin takım arkadaşlarıyla en uygun çözümü seçmelerine rehberlik eder.

En İyi Çözümün Seçilmesi

Öğrencilerin birçok çözüm önerisini takım arkadaşlarıyla tanımlamasından sonra problem durumunun çözümüne yönelik bilgi temelli hayat probleminde yer alan kriterler ve sınırlılıklar çerçevesinde yapılan analizlerle en uygun çözüm için karar vermeleri beklenir. En uygun çözümün seçilmesinin ardından öğrencilerden mühendislik becerilerini geliştirmeye katkı sağlamak amacıyla problemin çözümüne ilişkin bileşenleri ve aralarındaki ilişkiyi belirlemeleri beklenir. Öğrencilerin modelin algoritmasını oluşturmaları ve algoritmayı anlatmaları beklenir. Ayrıca öğrencilerden kâğıt, kalem veya <https://about.draw.io/> sitesini kullanarak bir akış diyagramı oluşturmaları ve yapacakları programın algoritmasını görselleştirmeleri beklenir. Öğrencilerin bir doğal dil işleme modeli oluşturmaları ve bu modeli kullanarak verilen metinlerin duygu durumlarının (olumlu, olumsuz ve tarafsız) doğru olarak sınıflandırılması beklenmektedir.

Duygu Durumu Analizi

Problem durumu analiz edildiğinde öğrencilerin hazır bulunuşlukları da dikkate alınarak problemi çözmeye yönelik geliştirilecek bir projenin üç ana bileşeni olduğu görülmektedir:

1. Veri toplama (CSV formatında)
2. Modelin eğitilmesi (Kodlama)
3. Modelin test edilmesi (Kodlama)

Prototipin Yapılması

Bu aşamada seçilen çözüm önerisine yönelik uygulamayı sunmak ve ayrıntılarını göstermek için öğrencilerden bir model geliştirmesi istenir. Modeli geliştirmek için blok tabanlı kodlama aracı (PictoBlox) kullanılacaktır.

Ön Hazırlık

Model, kullanıcıların yorumlarını alarak onların düşüncelerini olumlu, olumsuz ve tarafsız olarak ayıracaktır. Bu aşamada öğretmen öğrencilere TM'de yapılan örnekleri hatırlatarak model oluşturma hakkında bilgi verebilir.

Öğretmen bir metin sınıflandırıcının nasıl kullanıldığını örneklerle (2.Bölümde bulunan) gösterebilir.

mBlock Uzantıları

Uygulamanın geliştirilebilmesi için gerekli eklentiler yüklenmelidir.

Tablo 3.4: Kullanılan Eklentiler

Eklentinin Adı	Yüklendiği Bileşen	Kullanım Amacı
 Text to Speech Make your projects talk	Kuklalar	Yazıları seslendirir. Programda sesli mesaj ve yönlendirme için kullanılır.
 Natural Language Processing Create Text Classifiers with NLP	Kuklalar	Doğal dil işleme tekniklerini için kullanılan yapay zekâ eklentisidir.

Duygu Durumu Analizi

Veri Toplama ve Yükleme

Modeli eğitmek için üç duygu durumuna ait (olumlu, olumsuz ve tarafsız) yorumların bulunduğu bir veri seti kullanılmıştır. Bu veri seti internet alışverişi yapan müşterilerin yorumlarından oluşturulmuştur (Görsel 3.11).

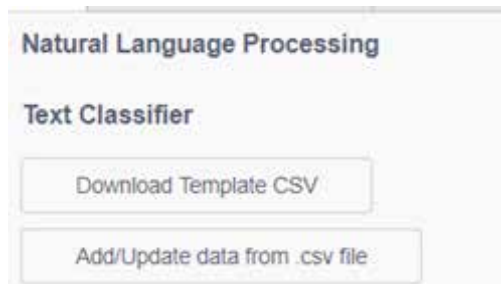
	A	B
1	Görüş	Durum
2	ses kalitesi ve ergonomisi rezalet	Olumsuz
3	hızlı teslimat tesekkürler	Tarafsız
4	ses olayı süper....gece çalıştır sık	Olumlu
5	geldi bigün kullandık hemen bozc	Olumsuz
6	Kulaklığın sesi kaliteli falan değil.	Olumsuz
7	Giriş seviyesindeki kullanıcılar için	Tarafsız
8	kullanisli baya	Olumlu
9	Dezavantajlar : Pahalı ürün 	Tarafsız
10	ÜRÜN GÜZEL BU PARALARA B	Tarafsız
11	Tasarım ve kalite iyi olmasına rağmen	Tarafsız
12	Çok değil çekim gücü az olduğu için	Tarafsız
13	İki kere aldım ikisindedede gelen jöl	Olumsuz
14	Klavve tuşları basmakla basmam takılma yapıyor. Başka da bir şey	Tarafsız
15	WPA2 ile sorun yaşayanlar eğer r	Olumsuz

Görsel 3.11: Veri seti

Bu aşamada öğrenciler araştırma yaparak kendi veri setlerini oluşturabilir. Örnek veri seti bağlantısı: <http://meb.ai/U3KMuEP>

Veri Setini Yükleme

PictoBlox programında Natural language processing eklentisine tıklanarak Text Classifier başlığı altında Ad/Update data from csv file butonu kullanılarak önceden oluşturulan CSV dosyası yüklenir (Görsel 3.12).



Görsel 3.12: NLP eklentisi

Yükleme işleminden sonra 3 etiketi (olumlu, olumsuz ve tarafsız) bulunan veri setinin eklendiğine dair ileti ekranda görünür (Görsel 3.13).

Duygu Durumu Analizi

Successfully added data with 3 labels!

Görsel 3.13: Veri ekleme uyarısı

Kodlama

Bu bölümde prototipe ait kod bloklarına yer verilmiştir.

Sahne ve Kukla Ayarları

Uygulama için uygun arka plan (backdrops) ve kukla eklenir. Kukla olarak Tobi kullanılır (Görsel 3.14).



Görsel 3.14: Sahne ve kukla ayarları

Duygu Durumu Analizi

Kukla kodları

Yeşil Bayrak tıkladığında çalışacak kodlar (Görsel 3.15)



Görsel 3.15: Tobi kuklası yeşil bayrak tıkladığında çalışacak kodlar

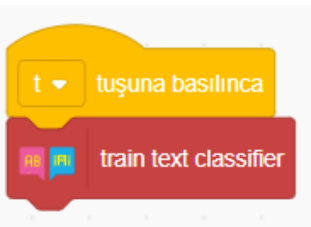
Kodların Açıklaması

Sanal asistanın kullanıcıyla iletişim kurmasını sağlayan kodlardır.

1. Yeşil bayrak tıkladığında.
2. Dil Türkçe olarak ayarlanır.
3. Alto sesi kullanılır.
4. Kullanıcıya uygulamanın kullanımıyla ilgili bilgi verilir.

Modeli Eğitme (t tuşuna basıldığında çalışacak kodlar)

Yüklenen veri setiyle bir metin sınıflandırıcı eğitilir. Artık model yeni örnekleri tahmin etmek için kullanıma hazırdır (Görsel 3.16).



Görsel 3.16: Metin sınıflandırıcının eğitimi

Boşluk tuşuna basıldığında çalışacak kodlar (Görsel 3.17)

Duygu Durumu Analizi



Görsel 3.17: Tobi kuklası boşluk tuşuna basıldığında çalışacak kodlar

Boşluk tuşuna basınca kullanıcıdan onun mağaza hakkında görüşleri alınır. Eğitilen model yorumu analiz ederek sınıf tahmini yapar. Kullanıcıya metin olarak ve sesli olarak ilettiği bölümdür.

1. Boşluk tuşuna basılınca
2. Kullanıcıya mağaza hakkındaki düşüncesi sorulur.
3. Kullanıcıdan gelen cevap analiz edilerek sınıflandırılır.
4. Gelen cevabın türüne göre ilgili değişken bir artırılır.

Çözümü Test Etme ve Değerlendirme

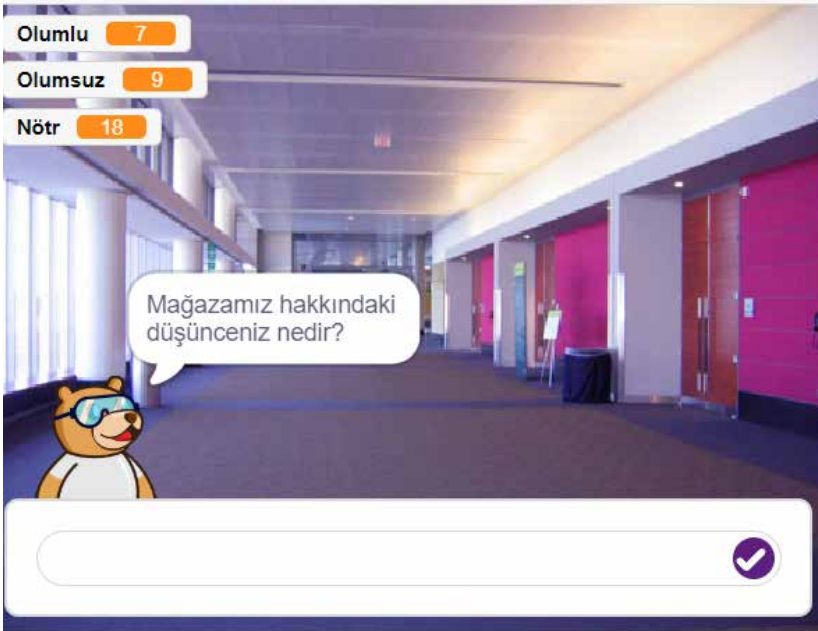
1. Daha önce yüklenmediyse veri seti yüklenir.
2. Yeşil bayrak tıklanarak uygulama çalıştırılır. Sanal asistan uygulama hakkında kullanıcıya bilgi verir.
3. t tuşuna basılarak model eğitilir.
4. Boşluk tuşuna basılarak sistem başlatılır. Kullanıcı yorumlarını girer (Görsel 3.18).

Duygu Durumu Analizi



Görsel 3.18: Programın çıktısı

5. Kullanıcın girdiği yorumlar analiz edilerek ilgili duygu durumu değişkenine eklenir. Örnekte olumlu bir ifade yazılarak test edilmiştir (Görsel 3.19: Programın test edilmesi).



Görsel 3.19: Programın test edilmesi

6. Olumlu bir ifade girildiği için olumlu değişkeni bir artar. Sürekli olarak görüş girilmesi için yanıt kutusu açılır. Olumlu, olumsuz ve tarafsız ifadelerle sistem test edilir.

Duygu Durumu Analizi

Sistem bir veri setindeki yorumları sınıflandırmak için geliştirilebilir.

NOT: Bu uygulamada natural language processing eklentisindeki sentiment analyse (duygu analizi) Türkçe duygu analizi yapamadığı için kullanılmamıştır.

Çözümün Sunulması

Bu aşamada öğrenciler takım arkadaşlarıyla birlikte oluşturdukları ürünleri diğer takımlara sunar. Öğretmen, sunum sırasında süreç değerlendirme formunu kullanarak öğrencilerin tasarımlarını ve uygulama sürecini değerlendirir. Öğretmen Değerlendirme Ölçeği Ek-1'de verilmiştir.

Yeniden Tasarlama / Revize Etme

Öğrencilerin aldıkları değerlendirmeler ve geri bildirimler doğrultusunda çözümlerinde gerekli iyileştirmeleri yaptıkları aşamadır. Öğrencilerin bu aşamada problemin kriter ve sınırlılıkları çerçevesinde tasarımlarını en iyi olabilecek şekilde iyileştirmeleri beklenir. Burada öğrencilerin oluşturmuş olduğu tasarım üründe iyileştirmenin nasıl yapılacağı, neden bu iyileştirmenin yapıldığını ve proje etkisini özetleyen takım hâlinde hazırlanmış bir rapor oluşturmaları istenebilir.

Kararın Tamamlanması

Kararların tamamlanmasında ise öğrencinin nihai tasarımının en iyi olup olmadığına kendisinin karar vermesi amacıyla Ek 2'de verilen öğrenci değerlendirme ölçeği kullanılacaktır.

Öğrenciler grup üyeleriyle birlikte aşağıda yer alan soruları cevaplandırarak proje değerlendirmesi yaparlar.

1. Bu etkinliği yaparken en çok nerede zorluk yaşadınız? Bu zorlukları nasıl aştınız?
2. Bu etkinliği yaparken en çok sevdiğiniz kısım hangisiydi?
3. Bu etkinlikten neler öğrendiniz?
4. Bu projeyi yeniden planladığında geliştirmek istediğin yönler nelerdir?

Kaynakça

5. [1] Hynes, M., Portsmore, M., Dare, E., Milto, E., Rogers, C., Hammer, D. & Carberry, A. (2011). Infusing engineering design into high school STEM courses. <http://ncete.org/flash/pdfs/Infusing%20Engineering%20Hynes>. Erişim tarihi: 10.02.2017.
6. [2] Brunsell, E. (2012). The engineering desing process. Brunsell, E. (Ed.) Integrating engineering + science in your classroom. Arlington, Virginia: National Science Teacher Association[NSTA] Press.

Duygu Durumu Analizi

EKLER

EK-1: Öğretmen Değerlendirme Ölçeği

Değerlendirme Kriterleri	GELİŞTİRİLMELİ	İYİ	ÇOK İYİ
Sorunu tanımlama ve analiz etme			
Olası çözümleri bulma ve en iyisini seçme			
Tasarım ürününün kabataslak çizimi veya simüle edilmiş formu			
Grup çalışması ve iş birliği			
Ürünü sunma			

EK-2: Öğrenci Öz Değerlendirme Ölçeği

Değerlendirme Kriterleri	Katılmıyorum	Kararsızım	Katılıyorum
Duygu durumu analizinin kullanım alanları hakkında bilgi sahibi oldum.			
Bilgi temelli hayat problemine yönelik bulduğum çözüm en uygun çözümdür.			
Senaryo sürecinde takım arkadaşlarımla iş birliği içinde bulunarak takım arkadaşlarıma katkı sağladım.			
Ürünü tasarlarken kriter ve sınırlılıkları dikkate aldım.			
Senaryo sürecini anlatan bir rapor oluşturdum.			

Plaka Tanıma Sistemi

Seviye	12 yaş ve üzeri
Konu	Basılı metinde OCR (Optik Karakter Tanıma)
Süre	6 ders saati



Planla ilgili dosyalara ulaşmak için karekod taratılabilir veya bağlantı kullanılabilir.

Özet

Öğrenme senaryosu gerçek yaşamdan alınarak hazırlanmış örnek bir olay içeren bilgi temelli hayat problemiyle başlar. Bilgi temelli hayat problemi oluşturulurken gerçek hayatta araç giriş çıkışının kontrol edilmesi ve bunun otomatikleştirilmesi için kullanılan plaka tanıma sistemlerinden esinlenilmiştir. Öğrenme senaryosunun geliştirilmesinde mühendislik tasarım süreci kullanılmıştır. Öğrenme senaryosu problemin tanımlaması amacıyla öğrenciler tarafından okunur ve öğrenme süreci başlar. Öğrenciler süreç boyunca takımlar hâlinde çalışır. Öğrencilerden senaryo sonucunda plakadan (basılı metin) OCR (Optik karakter tanıma) sistemi kullanarak yazıyı ve rakamları (karakter) tanıma modeli oluşturması beklenir. Burada amaç, öğrencilerin bilişim teknolojileri, fen bilimleri ve matematik derslerinde öğrendiklerini ve mühendislik becerilerini kullanarak bir tasarım ürünü oluşturmalarıdır. Prototip oluşturma sürecinde öğretmenler öğrencilere kodlama adımlarında rehberlik edecektir. Sürecin sonunda öğrenci takımları tasarımlarını sınıf arkadaşlarına sunacaklar. Öğrenci takımları ve prototip öğretmen tarafından hazırlanan ölçek ile değerlendirilecektir (Ek-1). Ayrıca öğrenciler öz değerlendirme ölçeğini kullanarak kendilerini değerlendireceklerdir (Ek-2).

STEM Konu Alanı 1

Bilişim Teknolojileri (Kazanımlar)

- Problem durumunu çözmek için uygun algoritmayı geliştirir.
- Arduino ile devre kurulumu yapar.
- Projelerine uygun yapay zekâ uygulamalarını seçer.
- Sensörlerden alınan verinin tabloya aktarılmasını sağlar.
- Blok tabanlı programlama aracını kullanarak program geliştirir.

STEM Konu Alanı 2

Mühendislik (Kazanımlar)

- Mühendislik ve tasarım ilişkisini ifade eder.
- Çevresindeki ürünleri mühendislik ve tasarım kavramları açısından ilişkilendirir.
- Mühendislik tasarım sürecindeki sınırlılıkları değerlendirir.
- Mühendislik tasarım sürecini kullanarak bir ürün tasarlar.
- Bir mühendislik projesinin içerdiği süreçleri tespit eder: Planlama, tasarım, prototip oluşturma, yürütme, kalite kontrol ve raporlama gibi aşamaları açıklar.
- Öğrenci proje çalışmasında kendisini farklı rollerdeki bir takım üyesi olarak görür, o rolün gerektirdiği çalışmaları başarıyla tamamlar.
- Çözüm önerisine yönelik düşüncelerini yazarak ve çizerek açıklar.
- Başkalarının çözüm önerisine yönelik düşüncelerini dikkate alır.

Plaka Tanıma Sistemi

1.3. Sosyal Ürün Kazanımları

- Öğrenci tasarlanan ürünü açık ve anlaşılır şekilde sınıfa sunar.

STEM Konu Alanı 3

Fen Bilimleri (Kazanımlar)

- Sesin yansımaları ve soğurulmasına örnek verir.

STEM Konu Alanı 4

Matematik (Kazanımlar)

- Veri toplamayı gerektiren araştırma soruları oluşturur.

Materyaller

Malzemeler

2 adet Servo motor

1 adet Arduino Uno R3

Çevrim İçi Araçlar

<https://about.draw.io/>

<https://www.tinkercad.com/>

<http://www.marijn.org/logos/online>

STEM Kariyer Bağlantıları

- Yapay zekâ mühendisliği
- Veri analizi
- Veri bilimi
- Bilgisayar mühendisliği
- Bilgisayar programcılığı

ETKİNLİK SÜRECİ

Bu etkinlik mühendislik tasarım süreci kullanılarak oluşturulmuştur. Öğretmen etkinlik sürecinin başlangıcında öğrencileri takımlara ayırarak çalışmalarına rehberlik eder.

Problemin Tanımlanması

Problem tanımlanırken mutlaka tasarımın kriterleri ve sınırlılıkları belirlenmelidir.

Problem senaryosunda yer alan kriterler ve sınırlılıklar: Öğrencilerin temel kodlama bilgisine sahip olması, yapay zekâ uygulamalarını tanıması ve Arduino ile devre tasarlaması gibi temalar olarak ortaya çıkmıştır. Öğretmen tarafından ihtiyaç ve problem durumu öğrencilere aşağıda yer alan bilgi temelli hayat problemi ile verilir.

Plaka Tanıma Sistemi

Bilgi Temelli Hayat Problemi

Antalya'da bulunan Akdeniz Üniversitesi yerleşkesinde birçok bölüm bulunan büyük bir üniversitedir. 2021-2022 eğitim öğretim yılında üniversiteye kayıtlı lisans öğrenci sayısı yaklaşık 67 bin kadardır. Üniversitede çalışan akademisyen sayısı ise yaklaşık 3 bindir [1]. Üniversite bünyesinde Tıp Fakültesi bulunması nedeniyle kampüste yoğun şekilde insan ve araç trafiği vardır. Üniversitede çok sayıda öğrenci ve personel bulunması güvenlik kontrolleri yapılırken zaman kaybı yaşanmasına neden olmaktadır. Güvenlik kontrollerinin hızlı ve hatasız şekilde yapılması için üniversite bünyesinde bütün öğrencilerin katılacağı bir yarışma düzenlenecektir. Yarışmaya katılan öğrenciler üniversiteye giriş çıkışların kontrollü olarak yapılmasını sağlayarak daha güvenli ve zamandan tasarruf sağlayabilecek bir uygulama geliştireceklerdir. Akdeniz Üniversitesinde öğrenim gören öğrencilerden biri olduğunuzu hayal ederek yarışmaya katılmak için ürününüzü geliştirmeye başlayabilirsiniz.

Tasarlanacak cihazın tasarım kriterleri şu şekilde belirlenmiştir:

- Araçların plakalarını görüntüden tanımalı.
- Plaka kayıtlı ise kapıyı otomatik olarak açmalı.
- Araçların giriş çıkış zamanlarını tabloya kaydetmeli.

NOT: Probleme örnek olarak verilen Akdeniz Üniversitesinde 2019 yılından itibaren plaka tanıma sistemi kullanılmaktadır. Kayıtlı araçlar giriş çıkış kontrol noktalarından otomatik olarak geçebilmektedir.

Probleme Yönelik İhtiyacın Belirlenmesi

Mühendisler akla gelen ilk çözümü sunmak yerine problem hakkında araştırma yaparak geçmişte kullanılan yöntemleri gözden geçirerek farklı çözüm yollarını bulmaya çalışırlar [2].

Bu aşamada öğrenciler çözüme yönelik ihtiyaçları belirlemek için OCR sistemlerini, giriş çıkış kontrol sistemlerini ve internet üzerinden tablolara veri kaydetmeyi araştırarak bilgi toplarlar.

Öğrenciler araştırmalar sonucunda OCR için yapay zekâ uygulamalarını, giriş çıkış sistemleri için Arduino, servo motorlarla kontrolleri ve internet üzerinden veri kaydetmek için Google e-Tablolar gibi uygulamalara erişeceklerdir.

Olası Çözümlerin Geliştirilmesi

Günlük yaşam problemlerinde tek bir doğru çözüm yoktur, genellikle birden fazla çözüm vardır [3]. Bu aşamada öğrenciler tasarım kriterlerini ve sınırlılıklarını dikkate alarak edindikleri bilgilere göre probleme yönelik çözüm önerileri geliştirirler. Öğrencilerin ürettikleri çözüm önerileri tabloya yazılır (Tablo 1).

Tablo 3.5: Çözüm Önerileri ve Sınırlılıklar

Çözüm Önerileri	Sınırlılıklar
1.	
2.	
3.	

Öğrenciler olası çözümlerini takım arkadaşlarıyla birlikte belirleyeceklerdir. Beyin fırtınası ile öğrencilerin oluşturdukları çözüm önerileri arasından problemin sınırlılıklarını dikkate alarak en uygun çözüme ulaşmaları beklenir. Öğrenciler oluşturmuş oldukları çözüm önerilerinin sınırlılıklarını beyin fırtınası ile kendileri belirleyeceklerdir. Öğretmen burada öğrencilerin beyin fırtınası yapmasını teşvik eder ve öğrenciler ile çözüm önerilerini değerlendirir. Öğrencilerin takım arkadaşlarıyla en uygun çözümü seçmelerine rehberlik eder.

Plaka Tanıma Sistemi

En İyi Çözümün Seçilmesi

Öğrencilerin birçok çözüm önerisini takım arkadaşlarıyla tanımlamasından sonra problem durumunun çözümüne yönelik bilgi temelli hayat probleminde yer alan kriterler ve sınırlılıklar çerçevesinde yapılan analizlerle en uygun çözüm için karar vermeleri beklenir. En uygun çözümün seçilmesinin ardından öğrencilerden mühendislik becerilerini geliştirmeye katkı sağlamak amacıyla çözümü iki boyutlu olarak bir kâğıda çizmeleri beklenir. Yapılan çizim üzerinde öğrenciler kullanacakları malzemeleri seçerek göstermelidir. Daha sonra öğrencilerden hazırlayacakları prototipin simülasyonunu gözlemlenmesi için Tinkercad üzerinden taslak devre modellerini oluşturmaları istenir. Tasarım fikri için öğrencilerin Arduino ve ultrasonik sensör ile devre tasarlaması, yapılan ölçümleri mBlock yardımıyla Google e-Tablolara kaydetmesi beklenmektedir.

Problem durumu analiz edildiğinde öğrencilerin hazır bulunuşlukları da dikkate alınarak problemi çözmeye yönelik geliştirilecek bir projenin üç ana bileşeni olduğu görülmektedir:

1. Devre tasarımı
2. Tablo oluşturma
3. Kodlama

Prototipin Yapılması

Bu aşamada seçilen çözüm önerisine yönelik uygulamayı sunmak ve ayrıntılarını göstermek için öğrencilerden bir tasarım geliştirmesi istenir.

Modelde araç giriş çıkışlarını kaydetmek ve otomatik girişe izin verilen araçların listesini tutmak için Google e-Tablolar kullanılmıştır.

Blok tabanlı kodlama için mBlock programı seçilmiştir. Devre tasarımı Tinkercad programında yapılmıştır. Öğrenci ve personele ait araç plakaları veri tabanında kayıtlıdır. Bir araç geldiğinde plakası okunur ve aracın plakasının veri tabanında bulunup bulunmadığı kontrol edilir. Araç veri tabanına kayıtlı ise Arduino'ya bağlı servo motor açılma çalışarak kapı açılır. Aracın giriş zamanı ve plaka bilgisi e-tabloya kaydedilir. Çıkış için B kapısına araç geldiğinde plakası tanınır, servo motor çalışarak kapı açılır ve aracın plakası ve çıkış zamanı e-tabloya kaydedilir.

Ön Hazırlık

Bu aşamada öğretmen öğrencilere devre elemanlarını tanıtır.

Servo motor 0-180 derece arasında bir açı ile dönebilmektedir. Servo motor çalışabilmesi için PWM pinine bağlanmalıdır. Servo motorun farklı açılarda dönmesi projede kapı kilidinin açılıp kapanması için kullanılır. Servo 0 derece açısı; kapının kapalı olması, 90 derece açısı ise kapının açık olması için kullanılır.

Aygıtın Bağlanması

Aygıtlar bölümünde ilgili Arduino kartı (bu projede UNO 3 kullanılmıştır) seçilerek projeye dâhil edilir. Arduino bilgisayara bağlanarak aygıtlar bölümünden mBlock ile bağlantısı gerçekleştirilmelidir. Yükleme işlemi yapılmalıdır. Bağlantı yapıldıktan sonra Arduino'ya firmware yüklenerek mBlock programı ile canlı modda bağlı duruma gelmesi sağlanır.

NOT: Konuyla ilgili ayrıntılı bilgi için 2. Bölüme bakılabilir.

Plaka Tanıma Sistemi

mBlock Uzantıları

Uygulamanın geliştirilebilmesi için gerekli eklentiler yüklenmelidir.

Tablo 3.6: Kullanılan Eklentiler

Eklentinin Adı	Yüklendiği Bileşen	Kullanım Amacı
 Cognitive Services By mBlock official	Kuklalar	Plaka tanıma işlemi için kullanılır.
 Time utilities Geliştiriciler: Kobi22-48 1s	Kuklalar	Kayıt zamanını almak için kullanılır.
 Google sayfa By mBlock official	Kuklalar	Araç giriş çıkışlarının Google E-Tablolara kaydedilmesini sağlamaktadır.

Devre Tasarımı ve Kurulumu

Bu aşamada seçilen çözüm önerisine yönelik tasarımı sunmak ve ayrıntılarını göstermek için prototip oluşturulur.

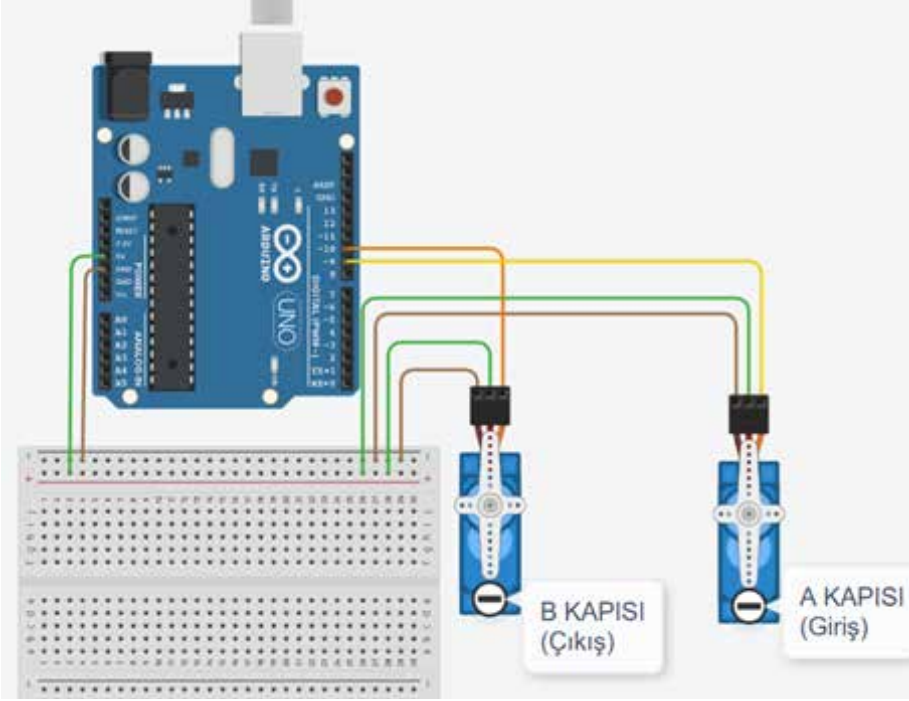
Prototip için gerekli malzemeler Tablo 3.7’de sunulmuştur.

Tablo 3.7: Kullanılan Malzemeler

Ad	Miktar	Bileşen	Kullanım Amacı
B1	1	Arduino Uno R3	-
B2	1	Servo motor	Kapı kontrolü
B3	1	Servo motor	Kapı kontrolü

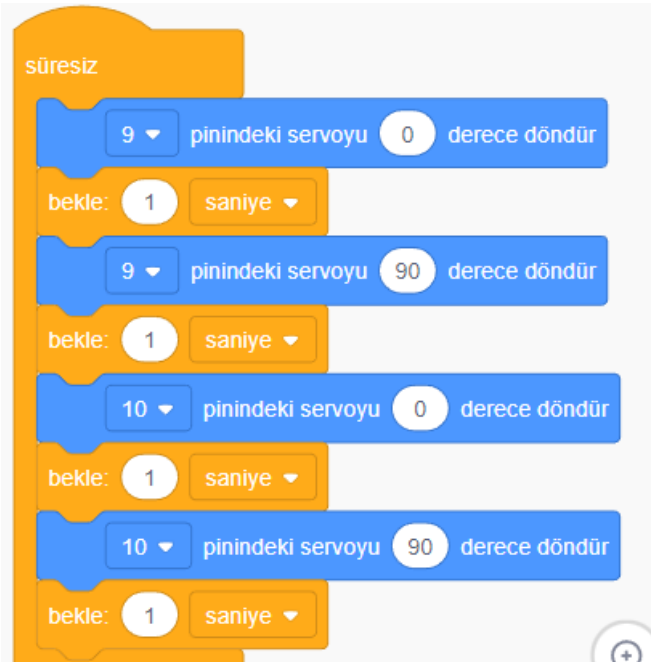
Simülasyon kullanılarak devrenin tasarlanması gerçek devre kurulumunda karşılaşılabilecek hataların ve arızaların önüne geçilmesini sağlar (Görsel 3.20).

Plaka Tanıma Sistemi



Görsel 3.20: Tinkercad devre şeması

Devrenin istendiği gibi çalışıp çalışmadığını test etmek için Tinkercad programında bir kodlama yapılır. Bu kodlamada devrede 9 ve 10 numaralı pinlere bağlı servo motorların sırayla önce 0 dereceye (açık konum) dönmesi, 1 saniye sonra da 90 dereceye (kapalı konum) dönmesi sağlanır (Görsel 3.21).



Görsel 3.21: Tinkercad blok kodlama

Plaka Tanıma Sistemi

Kodlama yapıldıktan sonra simülasyon başlatılır ve seri monitörde uzaklık ölçümü sonuçlarının görülüp görülmediği test edilir. Bir sorunla karşılaşılmazsa tasarlanan devre kurulumuna geçilir.

Yapılan devre tasarımına uygun bir biçimde devre kurulumu yapılır. HC-SR04 ultrasonik sensördeki trig pini Arduino D11 numaralı pine, echo pini ise Arduino D12 numaralı pine takılmıştır. SR04 ultrasonik sensördeki GND ve VCC pinleri de sırasıyla Arduino'nun GND ve 5V pinlerine takılmıştır.

Tablo Oluşturma

Araç giriş çıkışının ve izin verilen araçların listesinin kaydedileceği uygun bir tablo tasarlamak için (mBlock ile uyum sağlaması nedeniyle) Google E -Tablolar tercih edilmiştir. Bu amaçla Google Drive üzerinde iki adet Google E-Tablo oluşturulur. Birinci tabloda giriş izni verilen plakaların tutulduğu tabloda plaka bilgisi, sahibi ve sahibinin kimlik numarası gibi bilgiler tutulmaktadır (Görsel 3.22).

A	B	C
Plaka	Sahibi	KimlikNo
07FEYZA01	Murat ALTUN	*****
07FEYZA02	Sultan ALTUN	*****
07FEYZA03	Emine KUTLU	*****
06FEYZA01	Ömer ÇÖKLÜ	*****

Görsel 3.22: Google E-Tablo

Örnek tablo bağlantısı:

https://docs.google.com/spreadsheets/d/1rThiUZIEG2ZatIBw3goAkPBjocB8O5jEiL2BD_XoWi8/edit?usp=sharing

Bu tabloda yeni kayıtlar yapılacağı için bu dosyaya düzenleme yetkisi verilerek paylaşım yapılmalıdır.

İkinci tabloda araç giriş çıkış kayıt tablosunda plaka, tarih, saat ve kapı bilgisinin olduğu sütunlar bulunur. Bu tablo üniversiteye giriş çıkış yapan araçların kayıtlarını tutmak için tasarlanmıştır (Görsel 3.23).

A	B	C	D
Plaka	Tarih	Saat	Kapı
07FEYZA01			
07FEYZA01			
07FEYZA01			
07FEYZA01	9/8/2022	2:40	
07FEYZA01	9/8/2022	2:40	A kapısı

Görsel 3.23: Google E-tablo kayıt

Bu tabloda yeni kayıtlar yapılacağı için bu dosyaya düzenleme yetkisi verilerek paylaşım yapılmalıdır.

Kodlama

Bu bölümde prototipe ait kod bloklarına yer verilmiştir. Kod blokları aygıt kodları ve kukla kodları olmak üzere iki bölümde verilmiştir.

Plaka Tanıma Sistemi

Aygıt Kodları

Arduino kartı eklendikten sonra aşağıdaki kod blokları eklenir (Görsel 3.24).



Görsel 3.24: Arduino kod blokları

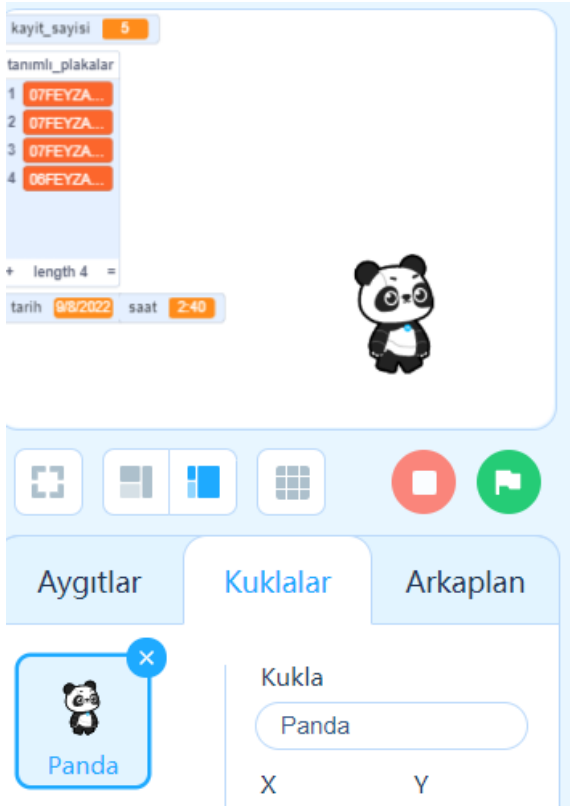
Kodlarının Açıklaması

İleti geldiğinde kapını açılması ve belirli bir süre sonra kapanması için gerekli kod blokları bulunmaktadır. İki servo motoru (iki kapı) kontrol için iki ayrı ileti bulunmaktadır.

Kukla Kodları

Bu bölümde prototipe ait kodlar yer almaktadır. Öncelikle mBlock programında sahne ve kuklalar ayarlanır.

Sahne ve Kukla Ayarları



Panda kuklası kullanılır. Sahnede başka bir şey yer almaz. İstenirse farklı bir arka plan yüklenebilir (Görsel 3.25).

Görsel 3.25: Sahne ve kukla ayarları

Plaka Tanıma Sistemi

Tanımlı plakaların tablodan alınması

d tuşuna basıldığında çalışacak kodlar (Görsel 3.26)



Görsel 3.26: Tobi kuklası d tuşuna basıldığında çalışacak kodlar

Kodların Açıklaması

1. d tuşuna basıldığında
2. Google e-tablolara bağlan (izin verilen araçların bilgisinin tutulduğu) https://docs.google.com/spreadsheets/d/1rThiUZIEG2ZatIBw3goAkPBjocB8O5jEiL2BD_XoWi8/edit?usp=sharing
3. E-tabloda kayıtlar 2. satırdan başladığı için (1. satırda sütun adları olduğu için) değişken değeri 2 olarak ayarlanır.
4. Boş satıra gelene kadar işlem tekrarlanır. İkinci satırdan itibaren tüm kayıtlar okunur.
5. Birinci sütundaki bu değerler tanımlı plakaların tutulduğu liste türündeki değişkene birer eleman olarak eklenir.
6. Bir sonraki satıra geçilmesi için (döngünün devam etmesi için) değişken bir artırılır.
7. Boş satıra gelindiğinde döngüden çıkılır ve yükleme tamamlandı mesajı gösterilir.

Plaka Tanıma Sistemi

A kapısından araç girişinin yapılması (Görsel 3.27)



Görsel 3.27: Tobi kuklası a tuşuna basıldığında çalışacak kodlar

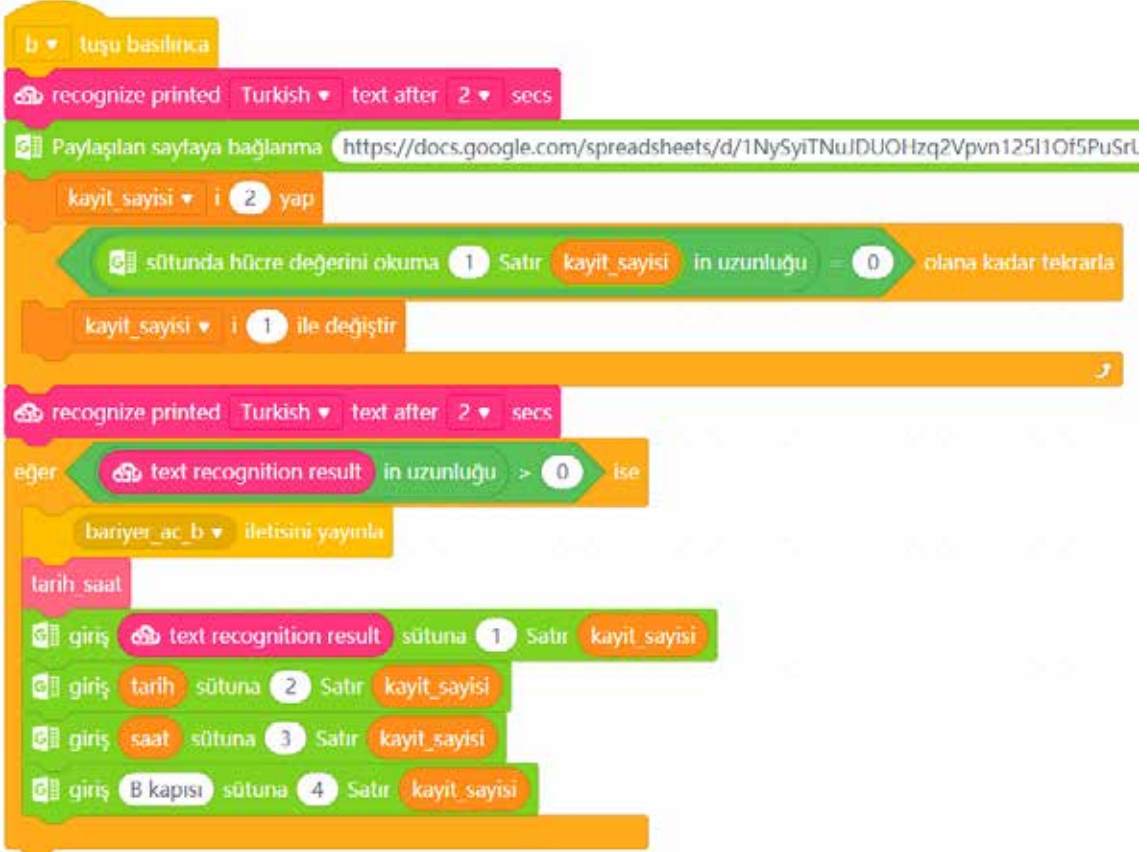
1. a tuşuna basıldığında
2. Bağlanacak E-tablonun paylaşılan bağlantısı yazılır ve tabloya bağlantı yapılır.
<https://docs.google.com/spreadsheets/d/1NySyiTNUJDUOHZq2Vpvn125l1Of5PuSrU6k8B C6Da-U/edit?usp=sharing>
3. E-tabloya kayıt işlemleri 2. satırdan başlayabileceği için (1. satırda sütun adları olduğu için) değişken değeri 2 olarak ayarlanır.
4. Önceki kayıtların üstüne kaydetmemek için kayıt numarası boş satırdaki numara olarak alınır.
5. Türkçe basılı metin tanıma işlemi yapılır (plaka tanıma).
6. Tanıma sonucu ekranda mesaj olarak gösterilir (konuşma balonu).
7. Plaka E-tabloya kaydedilir.
8. Eğer plaka izinli listede yer alıyorsa bariyerin açılması için aygıtı ileti gönderilir.
9. Bariyer açıldı mesajı verilir (konuşma balonu).
10. Tarih ve saat bilgisi alınır.

Plaka Tanıma Sistemi

11. Tarih, saat ve giriş kapısı bilgileri sisteme kaydedilir.
12. Plaka izinli listede değilse güvenlik için uyarı mesajı gösterilir (konuşma balonu).

B Kapısından araç çıkışının yapılması

b tuşuna basıldığında çalışacak kodlar (Görsel 3.28)



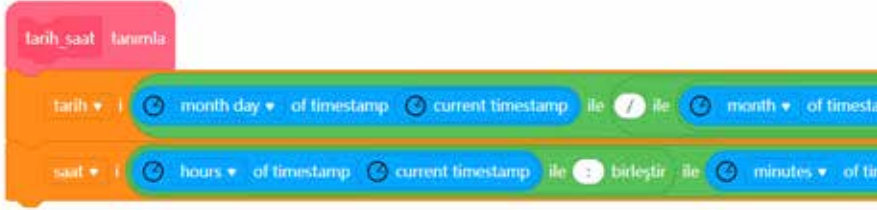
Görsel 3.28: Tobi kuklası b tuşuna basıldığında çalışacak kodlar

Kodların Açıklaması

1. b tuşuna basıldığında
2. Bağlanacak E-tablonun paylaşılan bağlantısı yazılır ve tabloya bağlantı yapılır. <https://docs.google.com/spreadsheets/d/1NySyiTNUJDUOHZq2Vpvn125I1Of5PuSrU6k8BC6Da-U/edit?usp=sharing>
3. E-tabloya kayıt işlemleri 2. satırdan başlayabileceği için (1. satırda sütun adları olduğu için) değişken değeri 2 olarak ayarlanır.
4. Önceki kayıtların üstüne kaydetmemek için kayıt numarası boş satırdaki numara olarak alınır.
5. Türkçe basılı metin tanıma işlemi yapılır (plaka tanıma).
6. Tanıma sonucu boş değilse
7. Bariyerin açılması için aygıtı ileti gönderilir.
8. Plaka E-tabloya kaydedilir.
9. Tarih ve saat bilgisi alınır.
10. Tarih, saat ve giriş kapısı bilgileri sisteme kaydedilir.

Plaka Tanıma Sistemi

Tarih saat bloku (Görsel 3.29)



Görsel 3.29: Tarih saat bloku

Çözümü Test Etme ve Değerlendirme

Model bir maket hâline getirilir.

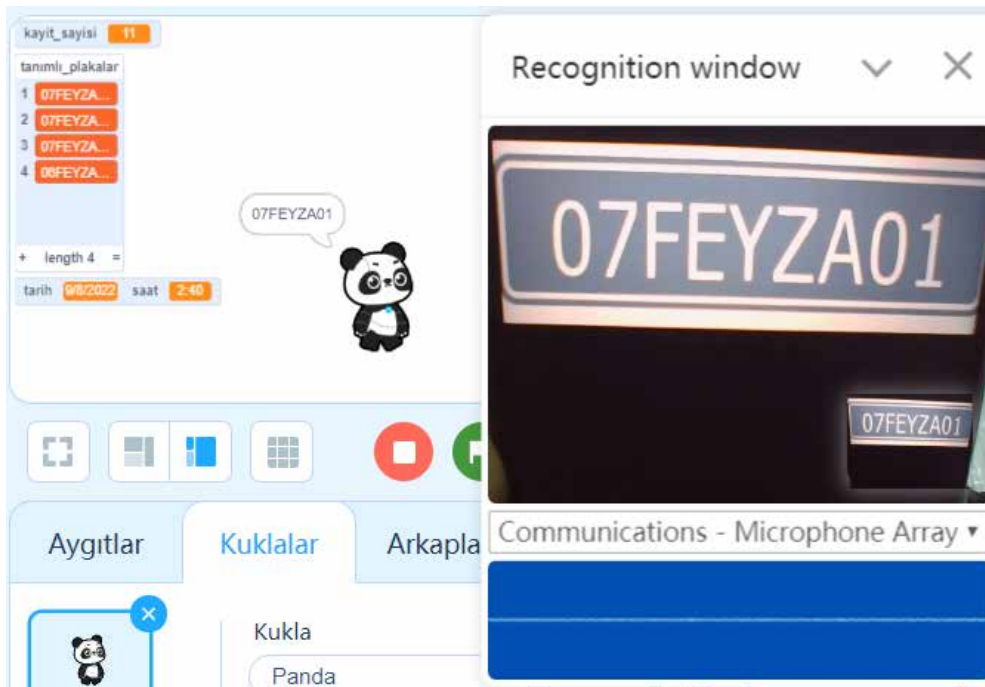
İzinli plaka listesinin bulunduğu E-tabloya plakalar eklenir. Kayıt yapılırken boşluk karakterine dikkat edilmezse eşleşmede sıkıntı çıkabilir.

D tuşuna basılarak tabloda izinli plakalar listeye alınır.

Sistemi test etmek için <http://www.marijn.org/logos/online> gibi sitelerden plakalar oluşturulabilir. Bu görseller doğrudan veya çıktı alınarak kullanılabilir.

Kamera, aracın plakasını görecek şekilde uygun bir yere konumlandırılır.

A tuşuna basılarak kameradan araç plakası tespit edilir. Eğer plaka listede yer alıyorsa otomatik olarak bariyer (A kapısı servo motor) açılır ve araca ilişkin bilgiler tabloya kaydedilir. Plaka listede yer almıyorsa güvenliğin dikkatine uyarısı görünecektir (Görsel 3.30).



Görsel 3.30: Programın çalıştırılması

B tuşuna basılarak çıkış yapılır. Çıkış için plaka tanıma işlemi yapıldıktan sonra kapı açılır ve araca ilişkin bilgiler E-tabloya kaydedilir.

Plaka Tanıma Sistemi

Prototip beklendiği gibi çalışmazsa plaka tanıma sonucunun listeye eşleşip eşleşmediği kontrol edilir.

Araçların giriş çıkış kayıtları (e tablodaki) kontrol edilir (Görsel 3.31).

	A	B	C	D
1	Plaka	Tarih	Saat	Kapı
2	07FEYZA01	9/8/2022	2:40	A kapısı
3	07FEYZA01	9/8/2022	2:40	A kapısı
4	07FEYZA01	9/8/2022	2:40	A kapısı
5	07FEYZA01	9/8/2022	2:40	A kapısı
6	07FEYZA01	9/8/2022	2:40	B kapısı
7	07FEYZA01	10/8/2022	0:48	B kapısı

Görsel 3.31: Google E tabloya kayıt

Çözümün Sunulması

Bu aşamada öğrenciler takım arkadaşlarıyla birlikte oluşturdukları ürünleri diğer takımlara sunar. Öğretmen, sunum sırasında süreç değerlendirme formunu kullanarak öğrencilerin tasarımlarını ve uygulama sürecini değerlendirir. Öğretmen Değerlendirme Ölçeği Ek-1'de verilmiştir.

Yeniden Tasarlama / Revize Etme

Öğrencilerin aldıkları değerlendirmeler ve geri bildirimler doğrultusunda çözümlerinde gerekli iyileştirmeleri yaptıkları aşamadır. Öğrencilerin bu aşamada problemin kriter ve sınırlılıkları çerçevesinde tasarımlarını en iyi olabilecek şekilde iyileştirmeleri beklenir. Burada öğrencilerin oluşturmuş olduğu tasarım; üründe iyileştirmenin nasıl yapılacağı, neden bu iyileştirmenin yapıldığını ve proje etkisini özetleyen takım hâlinde hazırlanmış bir rapor oluşturmaları istenebilir.

Kararın Tamamlanması

Kararların tamamlanmasında ise öğrencinin nihai tasarımının en iyi olup olmadığına kendisinin karar vermesi amacıyla Ek-2'de verilen öğrenci değerlendirme ölçeği kullanılacaktır.

Öğrenciler grup üyeleriyle birlikte aşağıda yer alan soruları cevaplandırarak proje değerlendirmesi yaparlar.

1. Bu etkinliği yaparken en çok nerede zorluk yaşadınız? Bu zorlukları nasıl aştınız?
2. Bu etkinliği yaparken en çok sevdiğiniz kısım hangisiydi?
3. Bu etkinlikten neler öğrendiniz?
4. Bu projeyi yeniden planladığında geliştirmek istediğin yönler nelerdir?

Kaynakça

[1] Akdeniz Üniversitesi. 2022. Tanıtım. <https://www.akdeniz.edu.tr/misyonumuz-ve-vizyonumuz/>. Erişim tarihi: 06.06.2022.

[2] Hynes, M., Portsmore, M., Dare, E., Milto, E., Rogers, C., Hammer, D. & Carberry, A. (2011). Infusing engineering design into high school STEM courses. <http://ncete.org/flash/pdfs/Infusing%20Engineering%20Hynes>. Erişim tarihi: 10.02.2017.

Plaka Tanıma Sistemi

[3] Brunsell, E. (2012). The engineering desing process. Brunsell, E. (Ed.) Integrating engineering + science in your classroom. Arlington, Virginia: National Science Teacher Association[NSTA] Press.

EKLER

EK-1: Öğretmen Değerlendirme Ölçeği

Değerlendirme Kriterleri	GELİŞTİRİLMELİ	İYİ	ÇOK İYİ
Sorunu tanımlama ve analiz etme			
Olası çözümleri bulma ve en iyisini seçme			
Tasarım ürününün kabataslak çizimi veya simüle edilmiş formu			
Grup çalışması ve iş birliği			
Ürünü sunma			

EK-2: Öğrenci Öz Değerlendirme Ölçeği

Değerlendirme Kriterleri	Katılmıyorum	Kararsızım	Katılıyorum
OCR (Optik Karakter Tanıma) teknolojilerini nerelerde kullanacağım hakkında bilgi sahibi oldum.			
Bilgi temelli hayat problemine yönelik bulduğum çözüm en uygun çözümdür.			
Senaryo sürecinde takım arkadaşlarımla iş birliği içinde bulunarak takım arkadaşlarıma katkı sağladım.			
Ürünü tasarlarken kriter ve sınırlılıkları dikkate aldım.			
Senaryo sürecini anlatan bir rapor oluşturdum.			

Kayıp Aranıyor

Seviye

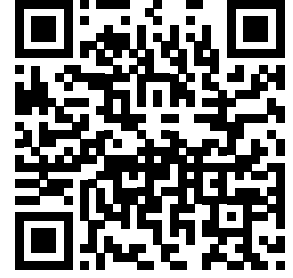
12 yaş ve üzeri

Konu

Yüz Tanıma Sistemleri

Süre

6 ders saati



Planla ilgili dosyalara ulaşmak için karekod taratılabilir veya bağlantı kullanılabilir.

Özet

Öğrenme senaryosu, yapay zekâ ile yüz tanıma teknolojileri kullanılarak kayıp olan bireyleri bulan bir programın geliştirilmesi üzerine odaklanır. Gerçek hayatta gazete, televizyon ve sosyal medyada sıkça kayıp bireylerle ilgili haberlerle karşılaşılmaktadır. Bu öğrenme senaryosunda öğrenciler gerçek yaşama ilişkin bir sorun olan kayıp bireylerin bulunması için çözüm üretmeye çalışacaktır. Öğrenme senaryosunun geliştirilmesinde mühendislik tasarım süreci kullanılmıştır. Öğrenme senaryosu problemin tanımlaması amacıyla öğrenciler tarafından okunur ve öğrenme süreci başlar. Öğrenciler süreç boyunca takımlar hâlinde çalışır. Öğrencilerden senaryo sonucunda yüz tanıma sistemlerini kullanarak bir program geliştirmeleri beklenir. Program kamera görüntüsünden aranan kişiyi belirlediğinde kullanıcıya bildirir. Burada amaç, öğrencilerin bilişim teknolojileri ve matematik derslerinde öğrendiklerini ve mühendislik becerilerini kullanarak bir tasarım ürünü oluşturmalarıdır. Ayrıca öğrencilerin farklı disiplinlerini günlük yaşamla ilişkilendirerek sorunları çözmek için kullanılabileceklerini fark etmelerini sağlamaktır. Prototip oluşturma sürecinde öğretmenler öğrencilere kodlama adımlarında rehberlik edecektir. Sürecin sonunda öğrenci takımları tasarımlarını sınıf arkadaşlarına sunacaklar. Öğrenci takımları ve prototip öğretmen tarafından hazırlanan ölçek ile değerlendirileceklerdir (Ek-1). Ayrıca öğrenciler öz değerlendirme ölçeğini kullanarak kendilerini değerlendireceklerdir (Ek-2).

STEM Konu Alanı 1

Bilişim Teknolojileri (Kazanımlar)

- Problem durumunu çözmek için uygun algoritmayı geliştirir.
- Projelerine uygun yapay zekâ uygulamalarını seçer.
- Blok tabanlı programlama aracını kullanarak program geliştirir.

STEM Konu Alanı 2

Mühendislik (Kazanımlar)

- Mühendislik ve tasarım ilişkisini ifade eder.
- Çevresindeki ürünleri mühendislik ve tasarım kavramları açısından ilişkilendirir.
- Mühendislik tasarım sürecindeki sınırlılıkları değerlendirir.
- Mühendislik tasarım sürecini kullanarak bir ürün tasarlar.
- Bir mühendislik projesinin içerdiği süreçleri tespit eder: Planlama, tasarım, prototip oluşturma, yürütme, kalite kontrol ve raporlama gibi aşamaları açıklar.
- Öğrenci proje çalışmasında kendisini farklı rollerdeki bir takım üyesi olarak görür, o rolün gerektirdiği çalışmalarını başarıyla tamamlar.

Kayıp Aranıyor

STEM Konu Alanı 3

Matematik (Kazanımlar)

- Veri toplamayı gerektiren araştırma soruları oluşturur.
- Oran, orantı kavramlarını fark eder.
- Oran, orantı kavramlarına günlük hayattan örnekler verir.

Materyaller

Çevrim içi araçlar

<https://about.draw.io/>

<https://pictoblox.ai/>

STEM Kariyer Bağlantıları

Yapay zekâ mühendisliği

Veri analizi

Veri bilimi

Bilgisayar mühendisliği

Bilgisayar programcılığı

ETKİNLİK SÜRECİ

Bu etkinlik mühendislik tasarım süreci kullanılarak oluşturulmuştur. Öğretmen etkinlik sürecinin başlangıcında öğrencileri takımlara ayırarak çalışmalarına rehberlik eder.

Problem Tanımlanması

Problem tanımlanırken mutlaka tasarımın kriterleri ve sınırlılıkları belirlenmelidir.

Problem senaryosunda yer alan kriterler ve sınırlılıklar: Öğrencilerin temel kodlama bilgisine sahip olması ve yapay zekâ uygulamalarını tanıması gibi temalar ortaya çıkmıştır. Öğretmen tarafından ihtiyaç ve problem durumu öğrencilere aşağıda yer alan bilgi temelli hayat problemi ile verilir.

Bilgi Temelli Hayat Problemi

Yapay zekâ ile oluşturulmuş yüz tanıma sistemleri günümüzde yaygın biçimde kullanılmaktadır. Yüz tanımanın en yaygın kullanıldığı alanlardan biri artık hemen hepimizin kullandığı akıllı cihazlardaki ekran kilidini açma yöntemidir. Ulusal güvenlikte, suçla mücadelede, kayıp veya kaçak insanların aranması ve bulunması gibi olaylarda da yüz tanıma sistemleri yaygın olarak kullanılmaktadır. Televizyon programlarında ve sosyal medyada görüldüğü üzere dünyada ve ülkemizde pek çok kişi kaybolmaktadır. Bu kişileri aramak, bulmak uzun süren, zor ve maliyetli bir süreçtir. Ayrıca bu süreçte kaybolan kişinin yakınları da maddi ve manevi zorluklar yaşamaktadır. Bu yüzden böyle durumlarda kayıp kişilerin oldukça hızlı bir şekilde bulunması önemlidir. Bu sorunun çözümünün yapay zekâda olduğunu düşünen FEYZA projesi siz mühendislerimize yüz tanıma sistemlerini kullanarak kayıp kişileri bulmayı sağlayacak bir program geliştirme görevini veriyor.

Geliştirilecek programın kriterleri şu şekilde belirlenmiştir:

- Aranılan kişinin yüzü sisteme bir fotoğraf ile tanıtılabilmeli.
- Kameradan arama yapabilmeli.
- Aranılan kişi bulunduğu anda zamanı tabloya kaydetmeli.

Kayıp Aranıyor

Probleme Yönelik İhtiyacın Belirlenmesi

Mühendisler akla gelen ilk çözümü sunmak yerine problem hakkında araştırma yaparak geçmişte kullanılan yöntemleri gözden geçirerek farklı çözüm yollarını bulmaya çalışırlar [1].

Burada öğrenciler çözüme yönelik ihtiyaçları belirlemek için yüz tanıma sistemlerini araştırarak bilgi toplar. Araştırmaları sonucunda kayıp insanları bulmak için yüz tanıma uygulamalarını kullanabileceklerini ve bunlarla program geliştirebileceklerini keşfedeceklerdir.

Olası Çözümlerin Geliştirilmesi

Günlük yaşam problemlerinde tek bir doğru çözüm yoktur, genellikle birden fazla çözüm vardır [2]. Bu aşamada öğrenciler tasarım kriterlerini ve sınırlılıklarını dikkate alarak edindikleri bilgilere göre probleme yönelik çözüm önerileri geliştirirler. Öğrencilerin ürettikleri çözüm önerileri tabloya yazılır (Tablo 3.7).

Tablo 3.7: Çözüm Önerileri ve Sınırlılıklar

Çözüm Önerileri	Sınırlılıklar
1.	
2.	
3.	

Öğrenciler olası çözümlerini takım arkadaşlarıyla birlikte belirleyeceklerdir. Beyin fırtınası ile öğrencilerin oluşturdukları çözüm önerileri arasından problemin sınırlılıklarını dikkate alarak en uygun çözüme ulaşmaları beklenir. Öğrenciler oluşturmuş oldukları çözüm önerilerinin sınırlılıklarını beyin fırtınası ile kendileri belirleyeceklerdir. Öğretmen burada öğrencilerin beyin fırtınası yapmasını teşvik eder ve öğrenciler ile çözüm önerilerini değerlendirir. Öğrencilerin takım arkadaşlarıyla en uygun çözümü seçmelerine rehberlik eder.

En İyi Çözümün Seçilmesi

Öğrencilerin birçok çözüm önerisini takım arkadaşlarıyla tanımlamasından sonra problem durumunun çözümüne yönelik bilgi temelli hayat probleminde yer alan kriterler ve sınırlılıklar çerçevesinde yapılan analizlerle en uygun çözüm için karar vermeleri beklenir. En uygun çözümün seçilmesinin ardından öğrencilerden mühendislik becerilerini geliştirmeye katkı sağlamak amacıyla problemin çözümüne ilişkin bileşenleri ve aralarındaki ilişkiyi belirlemeleri beklenir. Öğrencilerin modelin algoritmasını oluşturmaları ve algoritmayı anlatmaları beklenir. Ayrıca öğrencilerden kâğıt, kalem veya <https://about.draw.io/> sitesini kullanarak bir akış diyagramı oluşturmaları ve yapacakları programın algoritmasını görselleştirmeleri beklenir.

Öğrencilerin geliştirecekleri model, aranan kişiler kamerada tespit edildiğinde kamerada görüldüğü zamanının tutulmasına (CSV formatında) olanak vermelidir.

Prototipin Yapılması

Bu aşamada seçilen çözüm önerisine yönelik uygulamayı sunmak ve ayrıntılarını göstermek için öğrencilerden modeli geliştirmesi istenir. Modeli geliştirmek için yüz tanıma uzantılarını içeren blok tabanlı kodlama aracı PictoBlox kullanılacaktır.

Ön Hazırlık

Öğretmen öğrencilere yüz tanıma eklentisinin nasıl kullanıldığını örneklerle (2. Bölümde bulunan) gösterebilir.

Kayıp Aranıyor


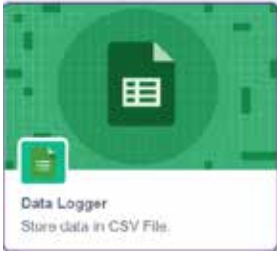
Yüz tanıma ile arama sisteminin çalışabilmesi için öncelikle aranan kişinin fotoğrafı yüz tanıma modeline tanıtılmalıdır. Yüz tanıma modelleri bir kişinin yüzünü yalnızca bir fotoğraf kullanarak öğrenebilir ve diğerlerinden ayırabilecek şekilde ayırabilir.

Veri dosyasını kaydetmek için PictoBlox, fare sağ tuşuna tıklanarak yönetici olarak çalıştır modunda açılmalıdır. Uygulamada veri dosyası varsayılan olarak Belgeler\Pictoblox\DataLogger konumuna kaydedilir. PictoBlox yönetici olarak çalıştırılmazsa veri dosyası oluşturmada sıkıntı yaşanabilir.

PictoBlox Uzantıları

Uygulamanın geliştirilebilmesi için gerekli eklentiler yüklenmelidir.

Tablo 3.8: Kullanılan Eklentiler

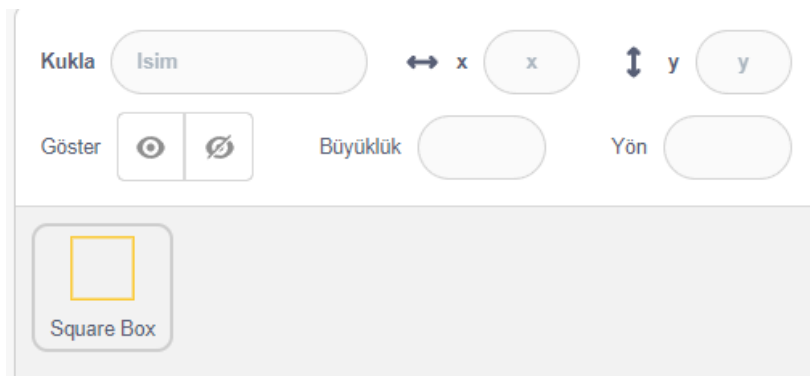
Eklentinin Adı	Yüklendiği Bileşen	Kullanım Amacı
 Face Detection Detect & recognize human face	Kuklalar	PictoBlox programında yüz tanıma ve eşleştirme işlemleri için face detection (yüz algılama) eklentisi kullanılmaktadır.
 Data Logger Store data in CSV File.	Kuklalar	Yüz eşleştirme sonuçlarının kaydedilmesi için (aranan kişi ve tespit edilme zamanı) Data Logger eklentisi kullanılmaktadır.

Kodlama

Bu bölümde prototipe ait kod bloklarına yer verilmiştir.

Sahne ve Kukla Ayarları

Dekor olarak aranan kişinin fotoğrafı yüklenir. Kukla olarak aranan kişi tespit edildiğinde yüzünü çerçevelemek ve adını yazmak amacıyla kukla kütüphanesinde bulunan Square Box kullanılmıştır (Görsel 3.32).



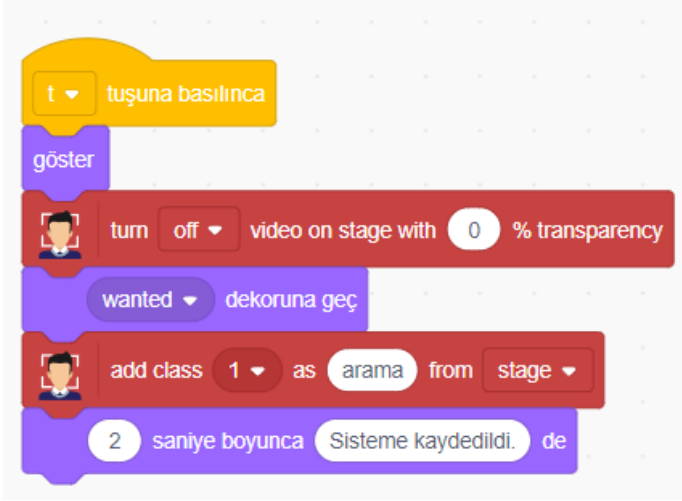
Görsel 3.32: Sahne ve kukla ayarları

Kayıp Aranıyor

Kukla Kodları

Aranan kişinin yüzünü sisteme tanıtmaya (t tuşuna tıklandığında çalışacak kodlar)

Aranan kişiye ait fotoğraftan oluşan dekor kullanılarak aranan kişinin yüzü model tarafından öğrenilir (Görsel 3.33).



Görsel 3.33: Square box kuklası t tuşuna basıldığında çalışacak kodlar

Kodların Açıklaması

1. **t tuşuna** basıldığında.
2. Kamera kapatılır.
3. Aranılan kişinin fotoğrafının bulunduğu dekora geçilir.
4. Aranılan kişinin **yüzünü tanı** ve **arama** olarak kaydedilir.
5. Kaydedildi mesajı verilir.

Arama işleminin yapılması (boşluk tuşuna basınca çalışacak kodlar)

Boşluk tuşuna basıldığında bir veri dosyası oluşturulur. Sistem aranan kişinin yüzünü sürekli olarak kameradaki görüntülerle karşılaştırır. Eşleştirme olduğunda bilgi verilir ve veri dosyasına kaydedilir (Görsel 3.34).

Kayıp Aranıyor

```
boşluk tuşuna basıldığında çalışacak kodlar
create file: arama
sesi alto olarak ayarla
dil: Türkçe olarak ayarla
turn on video on stage with 0 % transparency
2 saniye bekle
sürekli olarak
do face matching on camera
1 saniye bekle
eğer (get # faces > 0) ise
yüz = 0 yap
yüz = get # faces olarak kaçar tekrarla
yüz = 1 kadar değişir
eğer (get class of face yüz detected = arama) ise
boyutu % (get width of face yüz yap)
x: (get x position of face yüz) y: (get y position of face yüz) konumuna git
de (get class of face yüz detected de)
göster
konuş Aranan kişi bulundu.
Aranan kişi de
set data 1 to (get class of face yüz detected)
set data 2 to (get timestamp)
save data
```

Görsel 3.34: Square box kuklası boşluk tuşuna basıldığında çalışacak kodlar

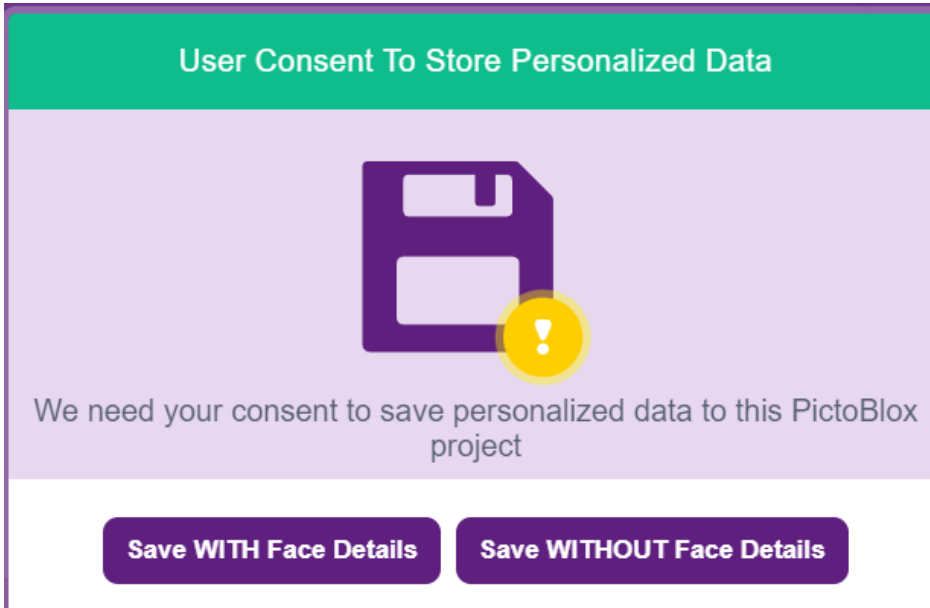
Kodların Açıklaması

1. Boşluk tuşuna basınca.
2. Ses alto olarak ayarlanır.
3. Dil Türkçe olarak ayarlanır.

Kayıp Aranıyor

4. Kamera açılır.
5. Kamera görüntüsünün aktarılabilmesi için iki saniye bekleme.
6. Sürekli olarak kayıtlı yüzlerle kameradaki görüntüden eşleştirme yapılır.
7. Kameradan yüz eşleştirme yapılır.
8. Eğer en az bir yüz tespit edilirse.
9. yüz değişkenine değer olarak 0 atanır.
10. Tespit edilen yüz sayısı kadar işlemler tekrarlanır.
11. Sonraki yüze geçmek için indeks sayısı bir artırılır.
12. Kameradaki her yüz için aranan kişi olup olmadığı kontrol edilir. Aranan kişi bulunduysa blok içindeki kodlar çalışır.
13. Kuklanın boyutu tespit edilen yüzün genişliği eşlenir.
14. Yüzü çerçeve içine almak için kukla tespit edilen yüzün konumuna gider.
15. Tespit edilen yüzün ait olduğu kişi (sınıf) adı yazılır.
16. Çerçevenin görünmesi sağlanır.
17. Aranan kişinin bulunduğu mesaj (sesli).
18. Aranan kişinin bulunduğu mesaj (konuşma balonu).
19. Birinci sütuna eşleştirilen yüzün sınıfı (kime ait olduğu) kaydedilir.
20. İkinci sütunda kayıt zamanı kaydedilir.

Proje kaydedilirken Save WITH Face Details butonuna tıklanırsa t tuşuna basılarak modele tanıtılan sınıflar kaydedilir. Böylece uygulama daha sonra açıldığında aranan kişi için yüz tanıtmaya gerek kalmadan boşluk tuşuna basılarak eşleştirme (arama) işlemi yapılabilir. Bulunan kişiler veri tabanına kaydedilir. Eğer Save WITHOUT Face Details butonuna tıklanırsa uygulama sonradan açıldığında modele aranan kişinin tanıtılması gerekir (Görsel 3.35).



Görsel 3.35: Kişisel bilgilerin kullanım izni

Kayıp Aranıyor

Çözümü Test Etme ve Değerlendirme

1. t tuşuna basılarak aranan kişinin fotoğrafı modele tanıtılır.
2. Boşluk tuşuna basıldığında sürekli olarak kameradan eşleştirme yapılır ve aranan kişi bulunursa mesaj verilir ve veri tabanına kaydedilir.

Aranan bir kişi bulandıktan sonra veri tabanına kaydedilir. Veri tabanı dosyası şu şekilde görünecektir (Görsel 3.36).

	A	B	C	D
1	arama,202288	13:37:46.964,,,,,,		
2	arama,202288	13:37:50.134,,,,,,		
3	arama,202288	13:37:53.133,,,,,,		
4	arama,202288	13:37:56.434,,,,,,		
5	arama,202288	13:37:59.536,,,,,,		

Görsel 3.36: Veri tabanı

Çözümün Sunulması

Bu aşamada öğrenciler takım arkadaşlarıyla birlikte oluşturdukları ürünleri diğer takımlara sunar. Öğretmen, sunum sırasında süreç değerlendirme formunu kullanarak öğrencilerin tasarımlarını ve uygulama sürecini değerlendirir. Öğretmen Değerlendirme Ölçeği Ek-1’de verilmiştir.

Yeniden Tasarlama / Revize Etme

Öğrencilerin aldıkları değerlendirmeler ve geri bildirimler doğrultusunda çözümlerinde gerekli iyileştirmeleri yaptıkları aşamadır. Öğrencilerin bu aşamada problemin kriter ve sınırlılıkları çerçevesinde tasarımlarını en iyi olabilecek şekilde iyileştirmeleri beklenir. Burada öğrencilerin oluşturmuş olduğu tasarım üründe iyileştirmenin nasıl yapılacağı, neden bu iyileştirmenin yapıldığını ve proje etkisini özetleyen takım hâlinde hazırlanmış bir rapor oluşturmaları istenebilir.

Kararın Tamamlanması

Kararların tamamlanmasında ise öğrencinin nihai tasarımının en iyi olup olmadığına kendisinin karar vermesi amacıyla Ek-2’de verilen öğrenci değerlendirme ölçeği kullanılacaktır.

Öğrenciler grup üyeleriyle birlikte aşağıda yer alan soruları cevaplandırarak proje değerlendirmesi yaparlar.

1. Bu etkinliği yaparken en çok nerede zorluk yaşadınız? Bu zorlukları nasıl aştınız?
2. Bu etkinliği yaparken en çok sevdiğiniz kısım hangisiydi?
3. Bu etkinlikten neler öğrendiniz?
4. Bu projeyi yeniden planladığında geliştirmek istediğin yönler nelerdir?

Kaynakça

[1] Hynes, M., Portsmouth, M., Dare, E., Milto, E., Rogers, C., Hammer, D. & Carberry, A. (2011). Infusing engineering design into high school STEM courses. <http://ncete.org/flash/pdfs/Infusing%20Engineering%20Hynes>. Erişim tarihi: 10.02.2017.

[2] Brunzell, E. (2012). The engineering desing process. Brunzell, E. (Ed.) Integrating engineering + science in your classroom. Arlington, Virginia: National Science Teacher Association[NSTA] Press.

Kayıp Aranıyor

EKLER

EK-1: Öğretmen Değerlendirme Ölçeği

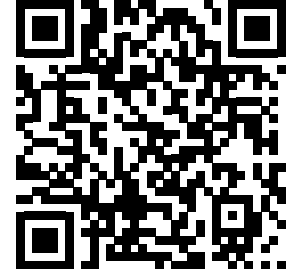
Değerlendirme Kriterleri	GELİŞTİRİLMELİ	İYİ	ÇOK İYİ
Sorunu tanımlama ve analiz etme			
Olası çözümleri bulma ve en iyisini seçme			
Tasarım ürününün kabataslak çizimi veya simüle edilmiş formu			
Grup çalışması ve iş birliği			
Ürünü sunma			

EK-2: Öğrenci Öz Değerlendirme Ölçeği

Değerlendirme Kriterleri	Katılmıyorum	Kararsızım	Katılıyorum
Yüz Tanıma Sistemleri teknolojilerini nerelerde kullanacağım hakkında bilgi sahibi oldum.			
Bilgi temelli hayat problemine yönelik bulduğum çözüm en uygun çözümdür.			
Senaryo sürecinde takım arkadaşlarımla iş birliği içinde bulunarak takım arkadaşlarıma katkı sağladım.			
Ürünü tasarlarken kriter ve sınırlılıkları dikkate aldım.			
Senaryo sürecini anlatan bir rapor oluşturdum.			

Sanal Doktor

Seviye	12 yaş ve üzeri
Konu	Doğal Dil İşleme (Metin sınıflandırma)
Süre	6 ders saati



Planla ilgili dosyalara ulaşmak için karekod taratılabilir veya bağlantı kullanılabilir.

Özet

Öğrenme senaryosunda doğal dil işleme teknikleri kullanılarak hastalık semptomlarına göre yönlendirme yapan bir program geliştirilmesi üzerinde odaklanılmıştır. COVID-19 salgını sürecinde grip, sıtma ve diyabet gibi hastalıklar COVID-19 ile karıştırılmıştır. Bu süreçte insanlar COVID-19 şüphesiyle sağlık kuruluşlarına başvurarak hastanelerde yoğunluğu arttırmışlardır. Öğrencilerden senaryo sonucunda doğal dil işleme ile bir model geliştirmeleri beklenir. Modelin hastalık semptomlarına göre kişiye hastalığını belirtmesi beklenir. Burada amaç öğrencilerin bilişim teknolojileri ve matematik derslerinde öğrendikleriyle mühendislik becerilerini de kullanarak bir tasarım ürünü oluşturmalarıdır. Ayrıca öğrencilerin farklı disiplinleri günlük yaşamla ilişkilendirerek sorunları çözmek için kullanabileceklerini fark etmelerini sağlamaktır. Prototip oluşturma sürecinde öğretmenler öğrencilere kodlama adımlarında rehberlik edecektir. Sürecin sonunda öğrenci takımları tasarımlarını sınıf arkadaşlarına sunacaklar. Öğrenci takımları ve prototip öğretmen tarafından hazırlanan ölçek ile değerlendirilecektir (Ek-1). Ayrıca öğrenciler öz değerlendirme ölçeğini kullanarak kendilerini değerlendireceklerdir (Ek-2).

STEM Konu Alanı 1

Bilişim Teknolojileri (Kazanımlar)

- Problem durumunu çözmek için uygun algoritmayı geliştirir.
- Projelerine uygun yapay zekâ uygulamalarını seçer.
- Blok tabanlı programlama aracını kullanarak program geliştirir.

STEM Konu Alanı 2

Mühendislik (Kazanımlar)

- Mühendislik ve tasarım ilişkisini ifade eder.
- Çevresindeki ürünleri mühendislik ve tasarım kavramları açısından ilişkilendirir.
- Mühendislik tasarım sürecindeki sınırlılıkları değerlendirir.
- Mühendislik tasarım sürecini kullanarak bir ürün tasarlar.
- Bir mühendislik projesinin içerdiği süreçleri tespit eder: Planlama, tasarım, prototip oluşturma, yürütme, kalite kontrol ve raporlama gibi aşamaları açıklar.
- Öğrenci proje çalışmasında kendisini farklı rollerdeki bir takım üyesi olarak görür, o rolün gerektirdiği çalışmaları başarıyla tamamlar.
- Çözüm önerisine yönelik düşüncelerini yazarak ve çizerek açıklar.
- Başkalarının çözüm önerisine yönelik düşüncelerini dikkate alır.

1.3. Sosyal Ürün Kazanımları

- Öğrenci tasarlanan ürünü açık ve anlaşılır şekilde sınıfa sunar.

Sanal Doktor

STEM Konu Alanı 3

Fen Bilimleri (Kazanımlar)

- Sistemlerin sağlığı için yapılması gerekenleri araştırma verilerine dayalı olarak tartışır.
- a) Sistem hastalıklarından Türkiye’de en sık rastlanan hastalıklara değinilir.

STEM Konu Alanı 4

Matematik (Kazanımlar)

- Veri toplamayı gerektiren araştırma soruları oluşturur.
- Kümeler ile ilgili temel kavramları anlar.

Materyaller

Çevrim içi araçlar

<https://about.draw.io/>

<https://pictoblox.ai/>

STEM Kariyer Bağlantıları

Yapay zekâ mühendisliği

Veri analizi

Veri bilimi

Bilgisayar mühendisliği

Bilgisayar programcılığı

ETKİNLİK SÜRECİ

Bu etkinlik mühendislik tasarım süreci kullanılarak oluşturulmuştur. Öğretmen etkinlik sürecinin başlangıcında öğrencileri takımlara ayırarak çalışmalarına rehberlik eder.

Problem Tanımlanması

Problem tanımlanırken mutlaka tasarımın kriterleri ve sınırlılıkları belirlenmelidir.

Problem senaryosunda yer alan kriterler ve sınırlılıklar: Öğrencilerin, temel kodlama bilgisine sahip olması ve yapay zekâ uygulamalarını tanıması gibi temalar olarak ortaya çıkmıştır. Öğretmen tarafından ihtiyaç ve problem durumu öğrencilere aşağıda yer alan bilgi temelli hayat problemi ile verilir.

Bilgi Temelli Hayat Problemi

Problem tanımlanırken mutlaka tasarımın kriterleri ve sınırlılıkları belirlenmelidir.

Problem senaryosunda yer alan kriterler ve sınırlılıklar: Öğrencilerin, temel kodlama bilgisine sahip olması, yapay zekâ uygulamalarını tanıması ve Arduino ile devre tasarlaması gibi temalar olarak ortaya çıkmıştır. Öğretmen tarafından ihtiyaç ve problem durumu öğrencilere aşağıda yer alan bilgi temelli hayat problemi ile verilir.

Tüm dünyayı kasıp kavuran COVID-19 salgını Türkiye’de resmi olarak 11 Mart 2020’de tespit edilen ilk vakadan sonra kısa sürede (1 Nisan 2020) ülke geneline yayıldı [1]. Salgın sürecinde insanlar yoğun olarak hastalığa ilişkin bilgi almak ve semptomları öğrenmek için farklı kaynaklardan bilgi alma yoluna gitmişlerdir. Bu bilgi kaynaklarının başında da internet gelmektedir. Salgın sürecinde doğal olarak internette COVID-19’a ilişkin aramalar artmıştır [2]. Yapılan araştırmalar bazı hastalıkların (benzer semptomlarından dolayı) COVID-19 ile karıştırıldığını ortaya koymuştur. Genellikle soğuk algınlığı, sinüzit, alerjik astım gibi hastalıklar COVID-19 ile karıştırılmaktadır [3]. İnternet üzerinden ulaşılan

Sanal Doktor

bilgiler her zaman güvenilir kaynaklardan olmamaktadır. İnsanlar hastalık şüphesiyle internet üzerinden yaptığı aramalar veya edindiği bilgiler doğrultusunda COVID-19 semptomları ile sağlık kuruluşlarına başvurarak buradaki yoğunluğu artırmışlardır. Sağlık kuruluşlarında bu insanların doğru şekilde yönlendirilmesi zaman ve maliyet açısından hem sağlık kuruluşları için hem de hastalar için sorun olmaktadır. Sağlık kuruluşları tarafından sunulan yönlendirmeler ve karmaşık yönergeleri takip etmek insanlar için zor olabilmektedir. Bu durum acil olarak sağlık hizmetine ihtiyaç duyan bireylere geç müdahale edilmesine ve durumlarının kötüleşmesine yol açabilmektedir. Gereksiz yoğunluk salgının artmasına da neden olabilmektedir. Hasta olduğumuzu düşündüğümüzde evden çıkmadan bize bilgi veren, semptomlar üzerinden bizi doğru şekilde yönlendiren bir uzmana veya asistana ihtiyaç duyuyoruz.

İnsanların bilgiye ulaşma yollarını düşünerek dijital bir ortamda semptomları ve gerekli bilgiler verildiğinde insanları doğru şekilde yönlendirecek bir yapay zekâ uygulaması yapılabilir miyiz?

Geliştirilecek modele ilişkin kriterler şunlar olmalıdır;

- Üç hastalık sıtma, diyabet ve COVID-19 belirtilerini tanımalı,
- Kullanıcının girdiği belirtileri analiz ederek hastalık tahmini yapmalıdır.

Probleme Yönelik İhtiyacın Belirlenmesi

Mühendisler akla gelen ilk çözümü sunmak yerine problem hakkında araştırma yaparak geçmişte kullanılan yöntemleri gözden geçirerek farklı çözüm yollarını bulmaya çalışırlar [4].

Bu aşamada öğrenciler çözüme yönelik ihtiyaçları belirlemek için doğal dil işleme konusunu araştırarak bilgi toplar. Bunun için öğrencilere sorular yöneltilir ve araştırma konuları verilir.

- Bilgisayarlar insanların dilini nasıl anlar?
- Bu problem durumundaki hastalıklara ve onların semptomlarına ilişkin bir liste oluşturunuz.

Olası Çözümlerin Geliştirilmesi

Günlük yaşam problemlerinde tek bir doğru çözüm yoktur, genellikle birden fazla çözüm vardır [5]. Bu aşamada öğrenciler tasarım kriterlerini ve sınırlılıklarını dikkate alarak edindikleri bilgilere göre probleme yönelik çözüm önerileri geliştirirler. Öğrencilerin ürettikleri çözüm önerileri tabloya yazılır (Tablo 3.9).

Tablo 3.9: Çözüm Önerileri ve Sınırlılıklar

Çözüm Önerileri	Sınırlılıklar
1.	
2.	
3.	

Öğrenciler olası çözümlerini takım arkadaşlarıyla birlikte belirleyeceklerdir. Beyin fırtınası ile öğrencilerin oluşturdukları çözüm önerileri arasından problemin sınırlılıklarını dikkate alarak en uygun çözüme ulaşmaları beklenir. Öğrenciler, oluşturmuş oldukları çözüm önerilerinin sınırlılıklarını beyin fırtınası ile kendileri belirleyeceklerdir. Öğretmen burada öğrencilerin beyin fırtınası yapmasını teşvik eder ve öğrenciler ile çözüm önerilerini değerlendirir. Öğrencilerin takım arkadaşlarıyla en uygun çözümü seçmelerine rehberlik eder.

Sanal Doktor

En İyi Çözümün Seçilmesi

Öğrencilerin birçok çözüm önerisini takım arkadaşlarıyla tanımlamasından sonra problem durumunun çözümüne yönelik bilgi temelli hayat probleminde yer alan kriterler ve sınırlılıklar çerçevesinde yapılan analizlerle en uygun çözüm için karar vermeleri beklenir. En uygun çözümün seçilmesinin ardından öğrencilerden mühendislik becerilerini geliştirmeye katkı sağlamak amacıyla problemin çözümüne ilişkin bileşenleri ve aralarındaki ilişkiyi belirlemeleri beklenir. Öğrencilerin modelin algoritmasını oluşturmaları ve algoritmayı anlatmaları beklenir. Ayrıca öğrencilerden kâğıt, kalem veya <https://about.draw.io/> sitesini kullanarak bir akış diyagramı oluşturmaları ve yapacakları programın algoritmasını görselleştirmeleri beklenir. Öğrencilerin bir doğal dil işleme modeli oluşturmaları ve bu modeli kullanarak verilen metinlerin duygu durumlarının (olumlu, olumsuz ve tarafsız) doğru olarak sınıflandırılması beklenmektedir.

Problem durumu analiz edildiğinde öğrencilerin hazır bulunuşlukları da dikkate alınarak problemi çözmeye yönelik geliştirilecek bir projenin üç ana bileşeni olduğu görülmektedir:

1. Veri toplama (CSV formatında)
2. Modelin eğitilmesi (Kodlama)
3. Modelin test edilmesi (Kodlama)

Prototipin Yapılması

Bu aşamada seçilen çözüm önerisine yönelik uygulamayı sunmak ve ayrıntılarını göstermek için öğrencilerden bir model geliştirmesi istenir. Modeli geliştirmek için blok tabanlı kodlama aracı (PictoBlox) kullanılacaktır.

Ön Hazırlık

Model kullanıcıdan semptomlara ilişkin bilgi alarak onların hastalığını tahmin eder. Bu aşamada öğretmen öğrencilere TM'de yapılan örnekleri hatırlatarak model oluşturma hakkında bilgi verebilir. Bir metin sınıflandırıcının nasıl kullanıldığını örneklerle (2. Bölümde bulunan) gösterebilir.

mBlock Uzantıları

Uygulamanın geliştirebilmesi için gerekli eklentiler yüklenmelidir.

Tablo 3.10: Kullanılan Eklentiler

Eklentinin Adı	Yüklendiği Bileşen	Kullanım Amacı
 Text to Speech Make your projects talk	Kuklalar	Yazıları seslendirir. Programda sesli mesaj ve yönlendirme için kullanılmaktadır.
 Natural Language Processing Create Text Classifiers with NLP	Kuklalar	Doğal dil işleme teknikleri için kullanılan yapay zekâ eklentisidir.

Veri Toplama ve Yükleme

Modelde 3 hastalık türüne (sıtma, diyabet ve COVID-19 ait semptomların bulunduğu bir veri seti kullanılmıştır. Bu veri seti hastalıklar ve semptomları ile ilgili araştırmalar yapılarak oluşturulur (Görsel 3.37).

Text	Class
nöbetler şeklinde ateş	sıtma
üşüme	sıtma
titreme	sıtma
baş ağrısı	sıtma
kas ağrısı	sıtma
yorgunluk	sıtma
Bulantı	sıtma
Kusma	sıtma
İshal	sıtma
değişen aralıklarla nöbet	sıtma
şiddetli titreme	sıtma
yükselen ateş	sıtma
Sık idrara çıkma	diabet
Aşırı susama	diabet
Yorgunluk	diabet
Kilo kaybı	diabet
Bulanık görme	diabet
Tekrarlayan enfeksiyonlar	diabet
Yaralarda geç iyileşme	diabet
ateş	COVID-19
öksürük	COVID-19
yorgunluk	COVID-19
tat alma duyusunun kaybı	COVID-19

Görsel 3.37: Veri seti

Bu aşamada öğrenciler araştırma yaparak kendi veri setlerini oluşturabilir. Örnek veri seti bağlantısı: <http://meb.ai/UWAWOAg>

Takımların veri setlerini oluşturmasının ardından matematik entegrasyonunu sağlamak amacıyla öğrencilere aşağıdaki soru yöneltilir.

Modelde verilen 3 hastalık türüne ait küme oluşturularak hastalık semptomlarının kesişimlerini Venn şeması hâlinde gösteriniz.

Veri Setini Yükleme

PictoBlox programında Natural language processing eklentisine tıklanarak Text Classifier başlığı altında Ad/Update data from csv file butonu kullanılarak önceden oluşturulan CSV dosyası yüklenir (Görsel 3.38).

Sanal Doktor

Natural Language Processing

Text Classifier

Download Template CSV

Add/Update data from .csv file

Görsel 3.38: NLP eklentisi

Yükleme işleminden sonra 3 etiketi (olumlu, olumsuz ve tarafsız) bulunan veri setinin eklendiğine dair ileti ekranda görünür.

Successfully added data with 3 labels!

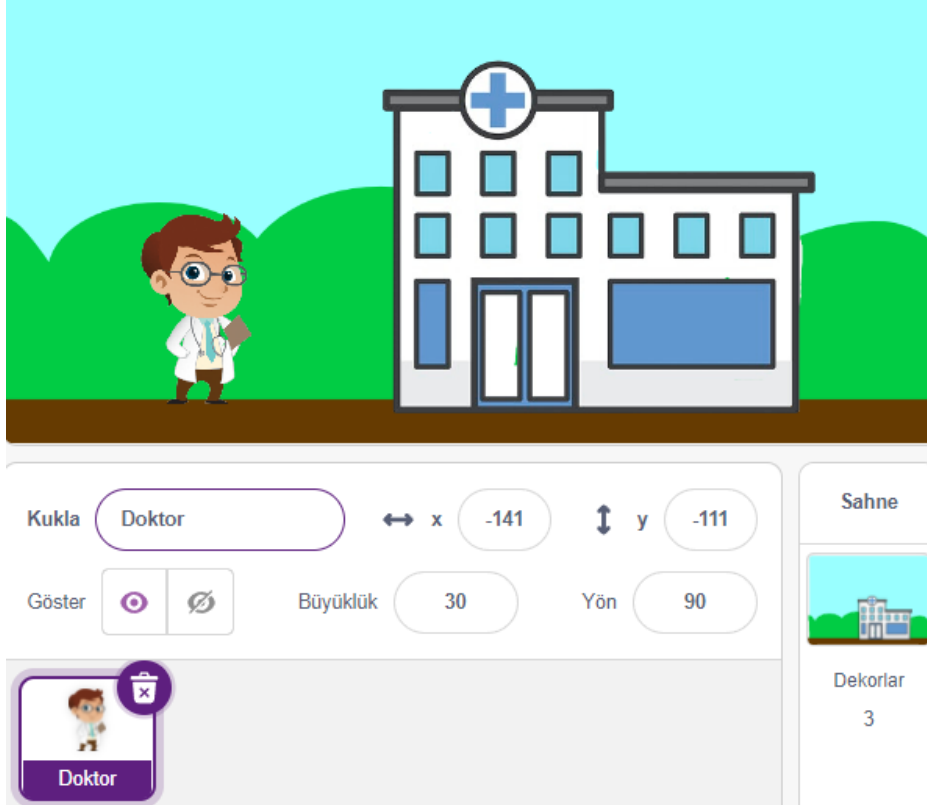
Görsel 3.39: Veri ekleme uyarısı

Kodlama

Bu bölümde prototipe ait kod bloklarına yer verilmiştir.

Sahne ve Kukla Ayarları

Uygulama için uygun arka plan ve kukla eklenir. Arka plan ve kukla dosyalarına karekod okutularak erişilebilir (Görsel 3.40).



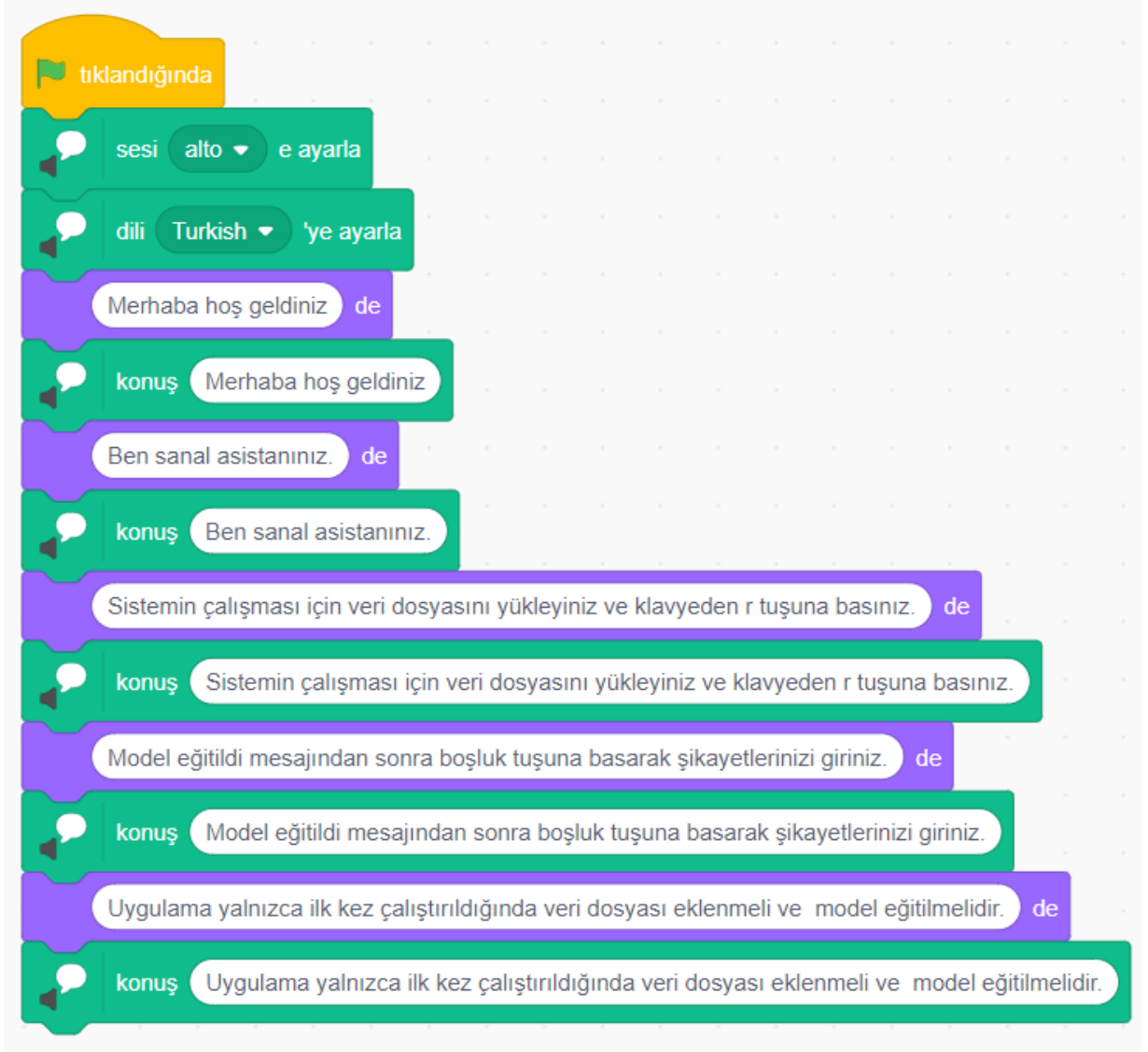
Görsel 3.40: Sahne ve kukla ayarları

Sanal Doktor

Kukla Kodları

Yeşil Bayrağa Tıklandığında Çalışacak Kodlar

Sanal asistanın kullanıcıyla iletişim kurmasını sağlayan kodlardır (Görsel 3.41).



Görsel 3.41: Doktor kuklası yeşil bayrak tıklandığında çalışacak kodlar

Kodların Açıklaması

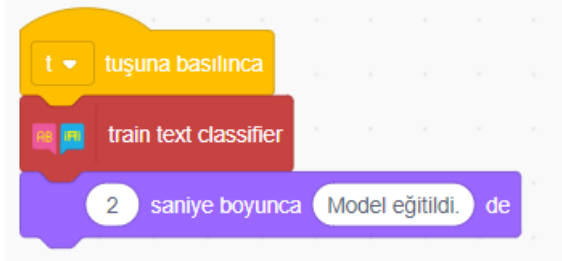
- 1.Yeşil bayrak tıklandığında.
2. Alto sesi kullanılır.
3. Dil Türkçe olarak ayarlanır.
4. Hoş geldiniz mesajı yazılır (konuşma balonu).
5. Hoş geldiniz mesajı (seslendirme).
6. Asistan kendini tanıtır (konuşma balonu) .
7. Asistan kendini tanıtır (seslendirme).

Sanal Doktor

Kodların devamında uygulamanın nasıl kullanılacağına dair yazılı ve sesli mesajlar verilmektedir.

Modeli Eğitme (t tuşuna basıldığında çalışacak kodlar)

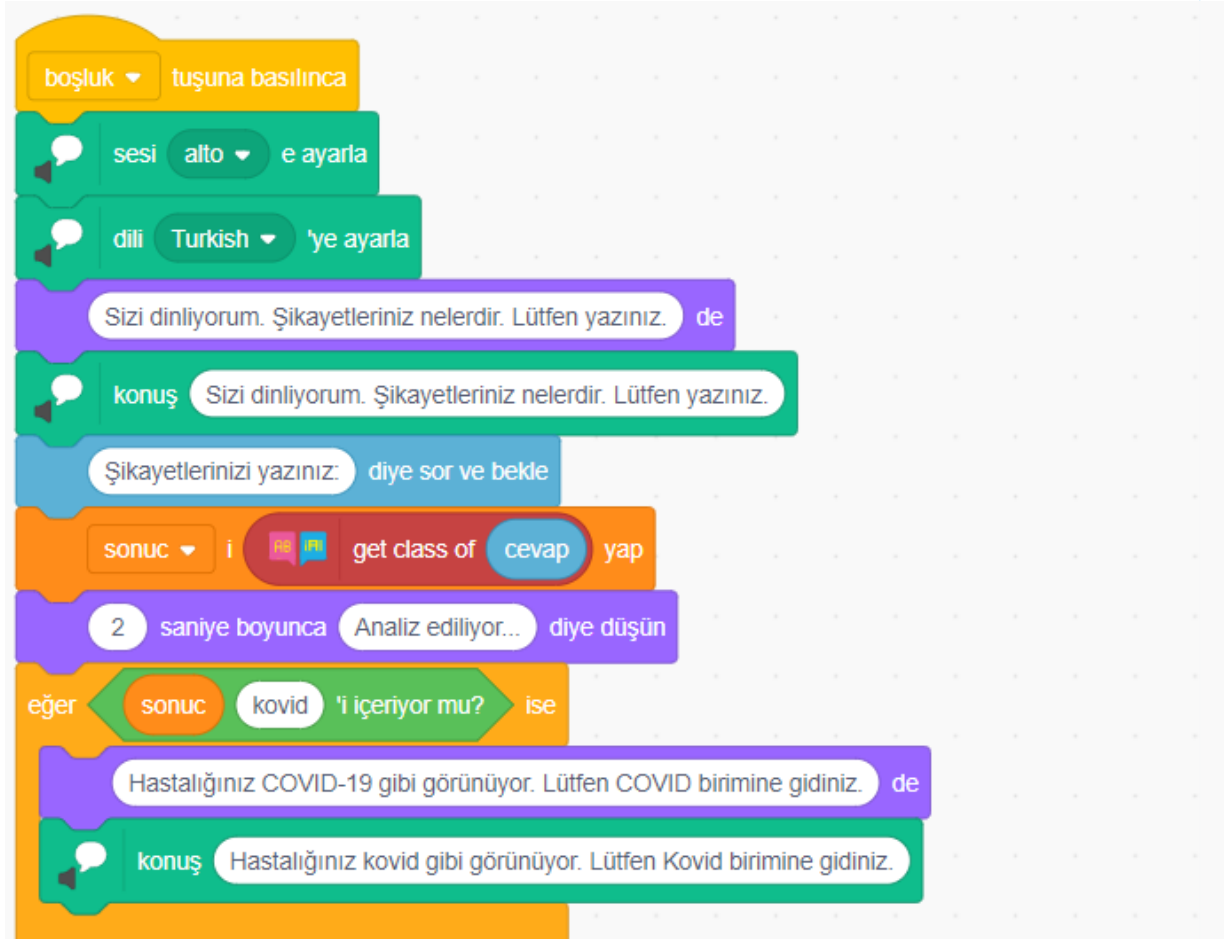
Yüklenen veri setiyle bir metin sınıflandırıcı eğitilir. Artık model yeni örnekleri tahmin etmek için kullanıma hazırdır (Görsel 3.42).



Görsel 3.42: Doktor kuklası metin sınıflandırıcısının eğitimi

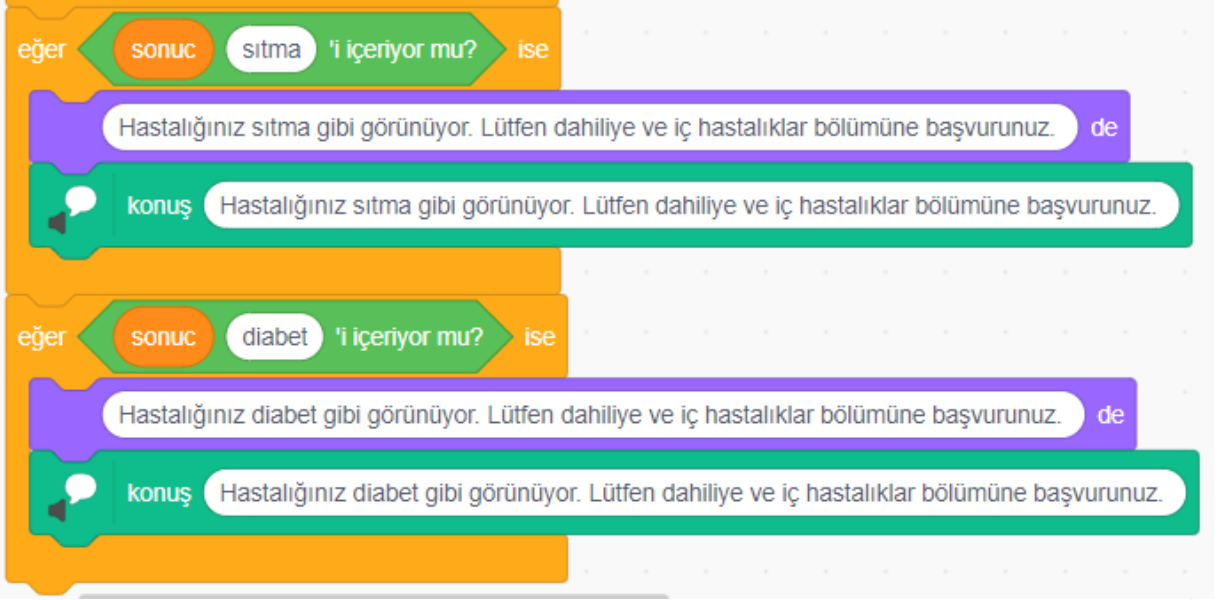
Boşluk Tuşuna Basınca Çalışacak Kodlar

Boşluk tuşuna basınca kullanıcıdan onun hastalığıyla ilgili bilgiler alarak modelin bu bilgilere göre bir sınıflandırma yaparak kullanıcıya metin olarak ve sesli olarak ilettiği bölümdür (Görsel 3.43-a Görsel 3.43-b).



Görsel 3.43-a: Doktor kuklası boşluk tuşuna basıldığında çalışacak kodlar

Sanal Doktor



Görsel 3.43-b: Doktor kuklası boşluk tuşuna basıldığında çalışacak kodlar

Kodların Açıklaması

1. Boşluk tuşuna basılınca.
2. Alto sesi kullanılır.
3. Dil Türkçe olarak ayarlanır.
4. Kullanıcıya şikâyetleri sorulur (konuşma balonu).
5. Yanıt alınır.
6. Verilen yanıt (cevap) metin sınıflandırıcıyla analiz edilir ve sonuc değişkenine atanır.
7. Kullanıcıya analizin yapıldığına dair mesaj verilir (konuşma balonu).
8. Analiz sonucu hastalık sınıfına göre kullanıcının hastalığı hakkında bilgi verilir (konuşma balonu ve seslendirme).

Sınıfları Sıfırlama

Hastalık sınıfları silinmek istenirse r tuşuna basılarak yapılabilir. Eğer sınıflar silinirse boşluk tuşuna basılarak hastalık teşhisi çalıştırıldığında tahmin yapamaz. Bunun için model tekrar eğitilmelidir (Görsel 3.44).



Görsel 3.44: Sınıfları resetlemek

Sanal Doktor

Çözümü Test Etme ve Değerlendirme

1. Hastalık semptomlarını içeren veri seti dosyası yüklenir.
2. t tuşuna basılarak model eğitilir.
3. Yeşil bayrak tıklanarak uygulama çalıştırılır. Sanal asistan, uygulama hakkında kullanıcıya bilgi verir.
4. Boşluk tuşuna basılarak teşhis koyma işlemi başlatılır. Sanal asistan kullanıcıdan şikâyetleri alır (Görsel 3.45).



Görsel 3.45: Programın çalıştırılması

Model şikâyetleri arka plandaki metin sınıflandırıcıyla (kullanıcı tarafından eğitilen) analiz eder ve belirtilerin ait olduğu hastalığı belirtir (Görsel 3.46).



Görsel 3.46: Modelin test edilmesi

Kullanıcı hastalıkların ortak semptomlarına ait belirtiler girerse sınıflandırma yapılamayabilir.

DİKKAT!

Bu uygulamadaki semptomlar ve bilgiler örnek olup hiçbir şekilde tıbbi bir tedavi önerisi veya yönlendirme değildir. Bu amaçla kesinlikle kullanılmamalıdır.

Sanal Doktor

Çözümün Sunulması

Bu aşamada öğrenciler takım arkadaşlarıyla birlikte oluşturdukları ürünleri diğer takımlara sunar. Öğretmen, sunum sırasında süreç değerlendirme formunu kullanarak öğrencilerin tasarımlarını ve uygulama sürecini değerlendirir. Öğretmen Değerlendirme Ölçeği Ek-1'de verilmiştir.

Yeniden Tasarlama / Revize Etme

Öğrencilerin aldıkları değerlendirmeler ve geri bildirimler doğrultusunda çözümlerinde gerekli iyileştirmeleri yaptıkları aşamadır. Öğrencilerin bu aşamada problemin kriter ve sınırlılıkları çerçevesinde tasarımlarını en iyi olabilecek şekilde iyileştirmeleri beklenir. Burada öğrencilerin oluşturmuş olduğu tasarım üründe iyileştirmenin nasıl yapılacağı, neden bu iyileştirmenin yapıldığını ve proje etkisini özetleyen takım hâlinde hazırlanmış bir rapor oluşturmaları istenebilir.

Kararın Tamamlanması

Kararların tamamlanmasında ise öğrencinin nihai tasarımının en iyi olup olmadığına kendisinin karar vermesi amacıyla Ek-2'de verilen öğrenci öz değerlendirme ölçeği kullanılacaktır.

Öğrenciler grup üyeleriyle birlikte aşağıda yer alan soruları cevaplandırarak proje değerlendirmesi yaparlar.

1. Bu etkinliği yaparken en çok nerede zorluk yaşadınız? Bu zorlukları nasıl aştınız?
2. Bu etkinliği yaparken en çok sevdiğiniz kısım hangisiydi?
3. Bu etkinlikten neler öğrendiniz?
4. Bu projeyi yeniden planladığında geliştirmek istediğin yönler nelerdir?

Kaynakça

[1] Sağlık Bakanlığı. 2020. <https://covid19.saglik.gov.tr/TR-66494/pandemi.html> . Erişim Tarihi: 07.06.2022

[2] Google Trends. 2022.

https://trends.google.com.tr/trends/story/DE_cu_vbfWEnEBAAAaJM_en Erişim tarihi: 22.07.2022.

[3] Acıbadem. 2022. <https://www.acibadem.com.tr/koronavirus/koronavirus-coronavirusu-nedir-belirtileri/> . Erişim Tarihi: 07.06.2022

[4] Hynes, M., Portsmore, M., Dare, E., Milto, E., Rogers, C., Hammer, D. & Carberry, A. (2011). Infusing engineering design into high school STEM courses. <http://ncete.org/flash/pdfs/Infusing%20Engineering%20Hynes>. Erişim tarihi: 10.02.2017.

[5] Brunsell, E. (2012). The engineering desing process. Brunsell, E. (Ed.) Integrating engineering + science in your classroom. Arlington, Virginia: National Science Teacher Association[NSTA] Press.

Sanal Doktor

EKLER

EK-1: Öğretmen Değerlendirme Ölçeği

Değerlendirme Kriterleri	GELİŞTİRİLMELİ	İYİ	ÇOK İYİ
Sorunu tanımlama ve analiz etme			
Olası çözümleri bulma ve en iyisini seçme			
Tasarım ürününün kabataslak çizimi veya simüle edilmiş formu			
Grup çalışması ve iş birliği			
Ürünü sunma			

EK-2: Öğrenci Öz Değerlendirme Ölçeği

Değerlendirme Kriterleri	Katılmıyorum	Kararsızım	Katılıyorum
Doğal dil işleme (metin sınıflandırıcılar) teknolojilerini nerelerde kullanacağım hakkında bilgi sahibi oldum.			
Bilgi temelli hayat problemine yönelik bulduğum çözüm en uygun çözümdür.			
Senaryo sürecinde takım arkadaşlarımla iş birliği içinde bulunarak takım arkadaşlarıma katkı sağladım.			
Ürünü tasarlarken kriter ve sınırlılıkları dikkate aldım.			
Senaryo sürecini anlatan bir rapor oluşturdum.			

Öğrenci Devamı Takip Sistemi

Seviye	12 yaş ve üzeri
Konu	Yüz Tanıma Sistemleri
Süre	6 ders saati

Planla ilgili dosyalara ulaşmak için karekod taratılabilir veya bağlantı kullanılabilir.



Özet

Bu öğrenme senaryosu yüz tanıma teknolojilerinin kullanılarak okullarda öğrenci devamsızlığının tespit edilmesini kolaylaştıracak bir program geliştirilmesi üzerine tasarlanmıştır. Dersler sırasında öğretmenlerin yoklama alma işlemi hem zaman kaybına neden olmakta hem de öğrencilerin motivasyonunu etkilemektedir. Ayrıca el yazısıyla yazılan devamsız öğrenci bilgilerinin e-okul sistemine işlenmesi için okunması idareciler için de ek bir iş yükü oluşturmaktadır. Bu durumu ortaya koyan bilgi temelli hayat senaryosunun öğrenciler tarafından okunmasıyla öğrenme senaryosu süreci başlar. Öğrenme senaryosunun geliştirilmesinde mühendislik tasarım süreci kullanılmıştır. Öğrenciler süreç boyunca takımlar hâlinde çalışır. Öğrencilerden senaryo sonucunda yüz tanıma teknolojilerini kullanarak bir program geliştirmeleri beklenir. Bu program kameradan veya görüntüden sınıftaki öğrencileri tanıyarak bir tabloya kaydedecektir. Burada amaç öğrencilerin bilişim teknolojileri ve matematik derslerinde öğrendiklerini ve mühendislik becerilerini kullanarak bir tasarım ürünü oluşturmalarıdır. Ayrıca öğrencilerin farklı disiplinleri günlük yaşamla ilişkilendirerek sorunları çözmek için kullanılabileceklerini fark etmelerini sağlamaktır. Prototip oluşturma sürecinde öğretmenler öğrencilere kodlama adımlarında rehberlik edecektir. Sürecin sonunda öğrenci takımları tasarımlarını sınıf arkadaşlarına sunacaklar. Öğrenci takımları ve prototip öğretmen tarafından hazırlanan ölçek ile değerlendirilecektir (Ek-1). Ayrıca öğrenciler öz değerlendirme ölçeğini kullanarak kendilerini değerlendireceklerdir (Ek-2).

STEM Konu Alanı 1

Bilişim Teknolojileri (Kazanımlar)

- Problem durumunu çözmek için uygun algoritmayı geliştirir.
- Projesine uygun yapay zekâ uygulamalarını seçer.
- Verinin tabloya aktarılmasını sağlar.
- Blok tabanlı programlama aracını kullanarak program geliştirir.

STEM Konu Alanı 2

Mühendislik (Kazanımlar)

- Mühendislik ve tasarım ilişkisini ifade eder.
- Çevresindeki ürünleri mühendislik ve tasarım kavramları açısından ilişkilendirir.
- Mühendislik tasarım sürecindeki sınırlılıkları değerlendirir.
- Mühendislik tasarım sürecini kullanarak bir ürün tasarlar.
- Bir mühendislik projesinin içerdiği süreçleri tespit eder: Planlama, tasarım, prototip oluşturma, yürütme, kalite kontrol ve raporlama gibi aşamaları açıklar.

Öğrenci Devamı Takip Sistemi

- Öğrenci proje çalışmasında kendisini farklı rollerdeki bir takım üyesi olarak görür, o rolün gerektirdiği çalışmaları başarıyla tamamlar.
- Çözüm önerisine yönelik düşüncelerini yazarak ve çizerek açıklar.
- Başkalarının çözüm önerisine yönelik düşüncelerini dikkate alır.

1.3. Sosyal Ürün Kazanımları

- Öğrenci tasarlanan ürünü açık ve anlaşılır şekilde sınıfa sunar.

STEM Konu Alanı 3

Matematik (Kazanımlar)

- Veri toplamayı gerektiren araştırma soruları oluşturur.
- Araştırma sorularına ilişkin verileri toplar, sıklık tablosu ve sütun grafiğiyle gösterir.
- Bir veri grubuna ait aritmetik ortalamayı hesaplar ve yorumlar.

Materyaller

Çevrim İçi araçlar

<https://about.draw.io/>

<https://pictoblox.ai/>

STEM Kariyer Bağlantıları

Yapay zekâ mühendisliği

Veri analizi

Veri bilimi

Bilgisayar mühendisliği

Bilgisayar programcılığı

ETKİNLİK SÜRECİ

Bu etkinlik mühendislik tasarım süreci kullanılarak oluşturulmuştur. Öğretmen etkinlik sürecinin başlangıcında öğrencileri takımlara ayırarak çalışmalarına rehberlik eder.

Problemin Tanımlanması

Problem tanımlanırken mutlaka tasarımın kriterleri ve sınırlılıkları belirlenmelidir.

Problem senaryosunda yer alan kriterler ve sınırlılıklar: Öğrencilerin, temel kodlama bilgisine sahip olması ve yapay zekâ uygulamalarını tanıması gibi temalar olarak ortaya çıkmıştır. Öğretmen tarafından ihtiyaç ve problem durumu öğrencilere aşağıda yer alan bilgi temelli hayat problemi ile verilir.

Bilgi Temelli Hayat Problemi

Emine Öğretmen, Antalya'da büyük bir okulda Fen Bilimleri Öğretmeni olarak görev yapmaktadır. Derse girdiğinde sınıfta diğer tüm öğretmenler gibi rutin olarak yoklama almaktadır. Derslerde yoklama işlemi hem zaman kaybına yol açmakta hem de öğrencilerin ders motivasyonlarını azaltmaktadır. Aynı zamanda devamsızlık yapan öğrencilerin kayıtlarının müdür yardımcıları tarafından e-okul sistemine aktarılması gerekmektedir. Bunun için de sınıf defterine el yazısıyla yazılan bilgilerin okunarak işlenmesi ek bir iş yükü oluşturmaktadır. Emine Öğretmen okuldaki yoklama sorununu çözmek için neler yapabileceğini araştırmaya başlamıştır. Siz Emine Öğretmene nasıl yardım edebilirsiniz? Emine Öğretmen teknolojiyi kullanarak yoklama alma sorununu nasıl çözebilir?

Öğrenci Devamı Takip Sistemi

Geliştirilecek olan programın kriterleri şu şekilde belirlenmiştir:

- Program fotoğrafları kullanarak veya kameradan yüz tanıma sistemiyle öğrencileri kaydetmeli.
- Kameradan veya görüntüden kayıtlı öğrencileri belirleyebilmeli.
- Sınıfta bulunan öğrencileri yüz tanımayla tespit ederek tabloya kaydetmeli.

Probleme Yönelik İhtiyacın Belirlenmesi

Mühendisler akla gelen ilk çözümü sunmak yerine problem hakkında araştırma yaparak geçmişte kullanılan yöntemleri gözden geçirerek farklı çözüm yollarını bulmaya çalışırlar [1].

Bu aşamada öğrenciler çözüme yönelik ihtiyaçları belirlemek için yüz tanıma sistemleri ve internet üzerinden tablolara veri kaydetmeyi araştırarak bilgi toplarlar. Öğrenciler araştırmalar sonucunda öğrenci devamsızlık takip sistemi için kodlama araçlarını, veri kaydetmek için CSV gibi dosyaları ve yüz tanıma sistemleri için de yapay zekâ uygulamalarını kullanabileceklerini keşfedeceklerdir.

Olası Çözümlerin Geliştirilmesi

Günlük yaşam problemlerinde tek bir doğru çözüm yoktur, genellikle birden fazla çözüm vardır [2].

Bu aşamada öğrenciler tasarım kriterlerini ve sınırlılıklarını dikkate alarak edindikleri bilgilere göre probleme yönelik çözüm önerileri geliştirirler. Öğrencilerin ürettikleri çözüm önerileri aşağıdaki Tablo 3.11'e yazılır.

Tablo 3.11: Çözüm Önerileri ve Sınırlılıklar

Çözüm Önerileri	Sınırlılıklar
1.	
2.	
3.	

Öğrenciler olası çözümlerini takım arkadaşlarıyla birlikte belirleyeceklerdir. Beyin fırtınası ile öğrencilerin oluşturdukları çözüm önerileri arasından problemin sınırlılıklarını dikkate alarak en uygun çözüme ulaşmaları beklenir. Öğrenciler oluşturmuş oldukları çözüm önerilerinin sınırlılıklarını beyin fırtınası ile kendileri belirleyeceklerdir. Öğretmen burada öğrencilerin beyin fırtınası yapmasını teşvik eder ve öğrenciler ile çözüm önerilerini değerlendirir. Öğrencilerin takım arkadaşlarıyla en uygun çözümü seçmelerine rehberlik eder.

En İyi Çözümün Seçilmesi

Öğrencilerin birçok çözüm önerisini takım arkadaşlarıyla geliştirildikten sonra problem durumunun çözümüne yönelik bilgi temelli hayat probleminde yer alan kriterler ve sınırlılıklar çerçevesinde yapılan analizlerle en uygun çözüm için karar vermeleri beklenir. En uygun çözümün seçilmesinin ardından öğrencilerden mühendislik becerilerini geliştirmeye katkı sağlamak amacıyla problemin çözümüne ilişkin bileşenleri ve aralarındaki ilişkiyi belirlemeleri beklenir. Öğrencilerin modelin algoritmasını oluşturmaları ve algoritmayı anlatmaları beklenir. Ayrıca öğrencilerden kâğıt, kalem veya <https://about.draw.io/> sitesini kullanarak bir akış diyagramı oluşturmaları ve yapacakları programın algoritmasını görselleştirmeleri beklenir.

Öğrenci Devamı Takip Sistemi

Prototipin Yapılması

Bu aşamada seçilen çözüm önerisine yönelik uygulamayı sunmak ve ayrıntılarını göstermek için öğrencilerden modeli geliştirmesi istenir. Modeli geliştirmek için yüz tanıma uzantılarını içeren blok tabanlı kodlama aracı PictoBlox kullanılacaktır.

Ön Hazırlık

Öğretmen öğrencilere yüz tanıma eklentisinin nasıl kullanıldığını örneklerle (2. Bölümde bulunan) gösterebilir.


Yüz tanıma ile arama sisteminin çalışabilmesi için öncelikle aranan kişinin fotoğrafı yüz tanıma modeline tanıtılmalıdır. Yüz tanıma modelleri bir kişinin yüzünü yalnızca bir fotoğraf kullanarak öğrenebilir ve diğerlerinden ayırabilecek şekilde ayırabilir.

Veri dosyasını kaydetmek için PictoBlox, fare sağ tuşuna tıklanarak yönetici olarak çalıştır modunda açılmalıdır. Uygulamada veri dosyası varsayılan olarak Belgeler\Pictoblox\DataLogger konumuna kaydedilir. Pictoblox yönetici olarak çalıştırılmazsa veri dosyası oluşturmada sıkıntı yaşanabilir.

PictoBlox Uzantıları

Uygulamanın geliştirilebilmesi için gerekli eklentiler yüklenmelidir.

Tablo 3.12: Kullanılan Eklentiler





Eklentinin Adı	Yüklendiği Bileşen	Kullanım Amacı
 Face Detection Detect & recognize human face	Kuklalar	PictoBlox programında yüz tanıma ve eşleştirme işlemleri için face detection (yüz algılama) eklentisi kullanılmaktadır.
 Data Logger Store data in CSV File.	Kuklalar	Yüz eşleştirme sonuçlarının kaydedilmesi için (aranan kişi ve tespit edilme zamanı) Data Logger eklentisi kullanılmaktadır.

Kodlama

Bu bölümde prototipe ait kod bloklarına yer verilmiştir.

Öğrenci Devamı Takip Sistemi

Sahne ve Kukla Ayarları

- 
damat_ferit
249 x 360
- 
guduk_necmi
191 x 250
- 
inek_saban
308 x 161
- 
hababm_sinifi
420 x 236

Prototip yazılım için 3 öğrenci sisteme tanıtılarak toplu bir fotoğraftan veya kameradan tespit edilmeleri için bilgileri veri tabanına kaydedilecektir. Bu nedenle Hababam Sınıfı adlı filmin 3 öğrencisi seçilmiştir. Bu üç öğrencinin her biri için bir yüz fotoğrafı ve bir de toplu fotoğrafları dekor olarak eklenir (Görsel 3.47). Öğrencilerin tekli fotoğrafları onları sisteme tanıtmak için kullanılırken toplu fotoğrafları ise eşleştirme yaparak kimlerin sınıfta olduğunu tespit etmek için kullanılacaktır. Amaç öğrencilerin devam takibini yapacak bir prototip test etmek olduğu için sistem çalışırsa daha fazla öğrenci ve kameradan eşleştirme gibi özellikler de eklenebilir. Kukla olarak öğrencilerin yüzünü çerçevelemek ve adlarını yazmak amacıyla kukla kütüphanesinde bulunan Square Box kullanılmıştır.

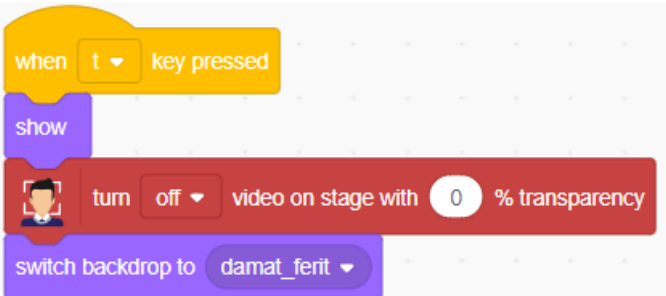


Görsel 3.47: Sahne ayarları

Kukla Kodları

Öğrencilerin yüzlerini sisteme tanıtmak (t tuşuna tıklandığında çalışacak kodlar)

Öğrencilere ait fotoğraflardan oluşan dekor kullanılarak öğrencilerin yüzü model tarafından öğrenilir. Her öğrenciye ait fotoğraflardan oluşmuş dekorlar kullanılarak öğrencilerin yüzleri model tarafından öğrenilir. Bu işlem için üç öğrencinin sahnedeki vesikalık fotoğrafları kullanılarak adları sınıf adı şeklinde kaydedilir (Görsel 3.48-a-b).



Görsel 3.48-a: Square box kuklası yüzlerin tanımlanması

Öğrenci Devamı Takip Sistemi



Görsel 3.48-b: Square box kuklası yüzlerin tanımlanması

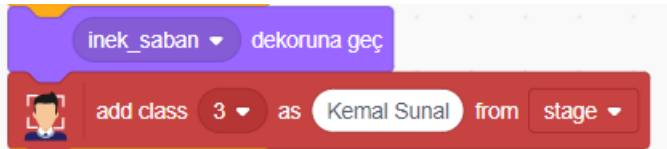
Kodların Açıklaması

1. t tuşuna basıldığında
2. Kamera kapatılır.
3. damat_ferit dekoruna geçilir.
4. Dekordaki yüz algılanarak Tarık Akan adıyla kaydedilir.
5. Aynı işlem 2 öğrenci için tekrarlanır. Onların da yüzü sisteme adlarıyla kaydedilir (Görsel 3.49).



Görsel 3.49: Square box kuklası öğrencileri kaydetme

inek_saban dekorunda algılanan yüz "Kemal Sunal" sınıf adı olarak kaydedilir

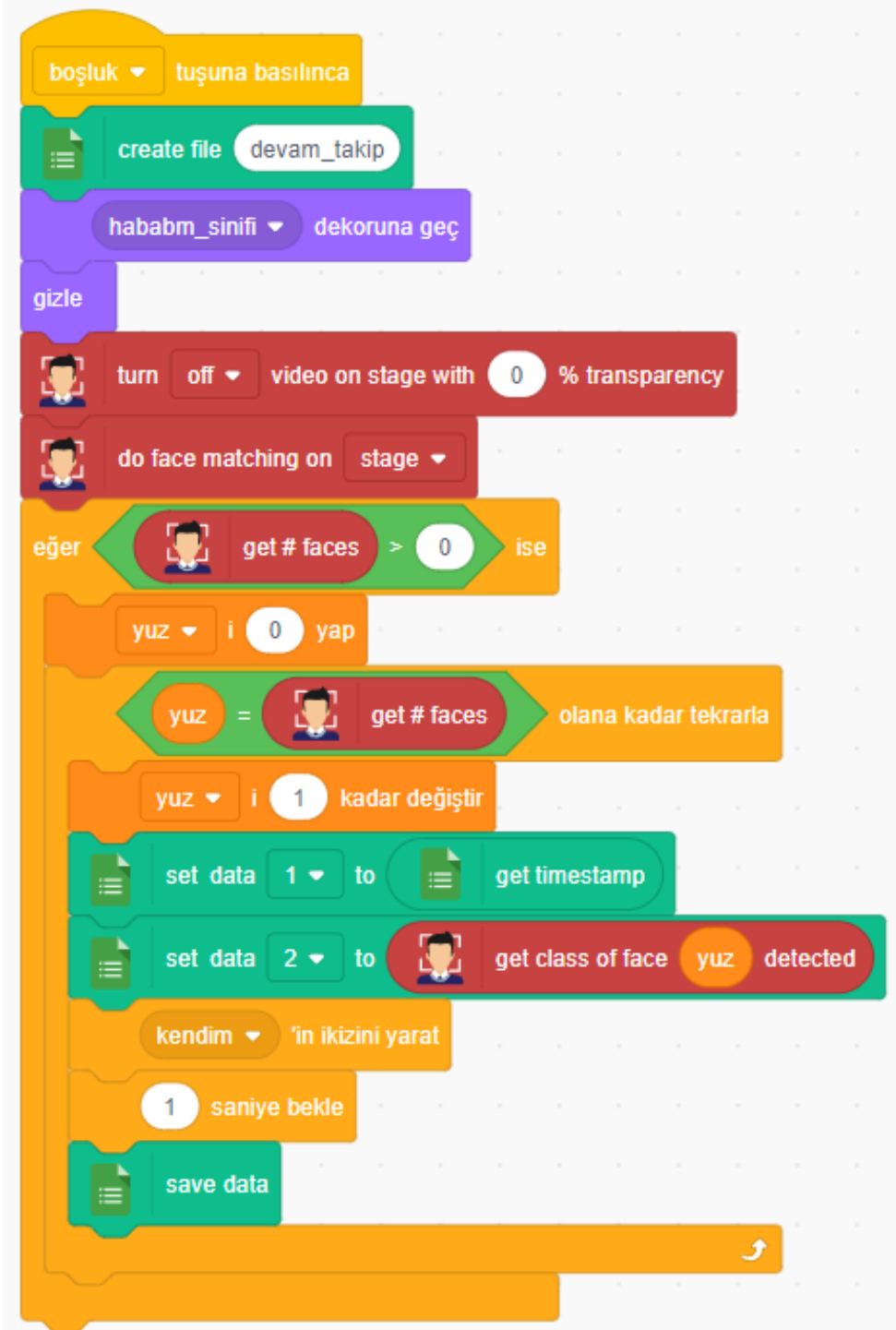


Görsel 3.50: Square box kuklası yüzlerin tanıtılması

Öğrenci Devamı Takip Sistemi

Yüzlerin eşleştirilmesi (boşluk tuşuna basınca çalışacak kodlar)

Boşluk tuşuna basıldığında bir veri dosyası oluşturulur. Kayıtlı öğrenci yüzleri ekrandaki görüntü ile eşleştirilir ve bilgileri veri tabanına kaydedilir (Görsel 3.51).



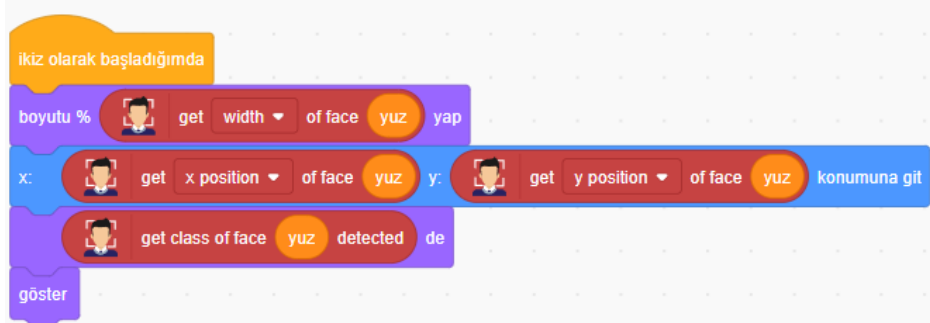
Görsel 3.51: Square box kuklası boşluk tuşuna basınca çalışacak kodlar

Öğrenci Devamı Takip Sistemi

Kodların Açıklaması

1. Boşluk tuşuna basınca
2. devam_takip adında bir CSV dosyası oluşturulur.
3. Öğrencilerin toplu hâlde bulunduğu sınıf fotoğrafı dekoruna geçilir.
4. Kukla gizlenir
5. Kamera görüntüsü kapatılır.
6. Sahneden yüz eşleştirme yapılır.
7. Eğer en az bir yüz tespit edilirse.
8. yuz değişkenine değer olarak 0 atanır.
9. Tespit edilen yüz sayısı kadar tekrarlanır.
10. Sonraki yüze geçmek için indeks sayısı bir artırılır.
11. Birinci sütuna kayıt zamanı kaydedilir.
12. İkinci sütunda eşleştirilen yüzün sınıfı (kime ait olduğu) kaydedilir.
13. Kuklanın (yüz) çerçevesinin ikizi oluşturulur.
14. Bir saniye bekleme.
15. Veri kaydedilir (Yüz sayısı kadar 10-15 arası kodlar tekrarlanır.).

İkiz olarak çalıştığında çalışacak kodlar (Görsel 3.52)



Görsel 3.52: Square box kuklası ikiz olarak çalıştığında çalışacak kodlar

Kodların Açıklaması

1. Kukla (yüz çerçevesi) ikiz olarak çalıştığında kuklanın boyutu ve tespit edilen yüzün genişliği eşlenir.
2. Yüzü çerçeve içine almak için kukla tespit edilen yüzün konumuna gider.
3. Tespit edilen yüzün ait olduğu kişi (sınıf) adı yazılır.
4. Çerçevenin görünmesi sağlanır.

Kameradan Yüz Eşleştirme

Öğrenci devam takip işlemini kameradan yapmak için kamera görüntüsünü açmak ve analizi kameradan yapmak yeterlidir (Görsel 3.53).

Öğrenci Devamı Takip Sistemi



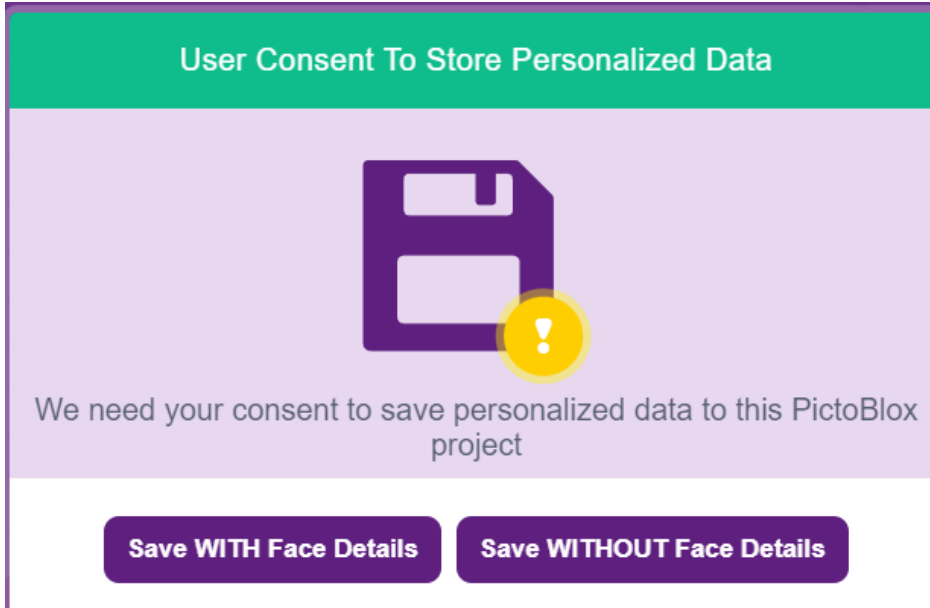
Görsel 3.53: Square box kuklası e tuşuna basıldığında çalışacak kodlar

Öğrenci Devamı Takip Sistemi

Kodların Açıklaması

1. e tuşuna basınca
2. devam_takip adında bir CSV dosyası oluşturulur.
3. Kukla gizlenir
4. Kamera görüntüsü açılır.
5. Kameradan yüz eşleştirme yapılır.
6. Gecikmeler için 2 saniye beklenir.
7. Eğer en az bir yüz tespit edilirse.
8. yuz değişkenine değer olarak 0 atanır.
9. Tespit edilen yüz sayısı kadar tekrarlanır.
10. Sonraki yüze geçmek için indeks sayısı bir artırılır.
11. Birinci sütuna kayıt zamanı kaydedilir.
12. İkinci sütunda eşleştirilen yüzün sınıfı (kime ait olduğu) kaydedilir.
13. Kuklanın (yüz) çerçevesinin ikizi oluşturulur.
14. Bir saniye bekleme.
15. Veri kaydedilir (Yüz sayısı kadar 10-15 arası kodlar tekrarlanır.).

Proje kaydedilirken Save WITH Face Details butonuna tıklanırsa t tuşuna basılarak modele tanıtılan sınıflar kaydedilir. Böylece uygulama daha sonra açıldığında aranan kişi için yüz tanıtmaya gerek kalmadan boşluk tuşuna basılarak eşleştirme (arama) işlemi yapılabilir. Bulunan kişiler veri tabanına kaydedilir. Eğer Save WITHOUT Face Details butonuna tıklanırsa uygulama sonradan açıldığında modele aranan kişinin tanıtılması gerekir (Görsel 3.54).



Görsel 3.54: Kişisel bilgilerin kullanım izni

Öğrenci Devamı Takip Sistemi

Çözümü Test Etme ve Değerlendirme

1. t tuşuna basılarak öğrenci fotoğrafları modele tanıtılır.
2. Boşluk tuşuna basıldığında öğrencilerin toplu hâldeki görüntülerinden eşleştirme yapılır ve veri tabanına kaydedilir (Görsel 3.55).



Görsel 3.55: Programın test edilmesi

Tespit edilen öğrenciler veri tabanına kaydedildikten sonra dosya şu şekilde görünecektir (Görsel 3.56).

	A	B	C	D
1	202287	18:8:30.662,	Kemal Sunal,	,,,,,,
2	202287	18:8:31.719,	Tarik Akan,	,,,,,,
3	202287	18:8:32.778,	Halit Akcatepe,	,,,,,,
4	202287	18:8:58.170,	Kemal Sunal,	,,,,,,
5	202287	18:8:59.203,	Tarik Akan,	,,,,,,
6	202287	18:9:0.254,	Halit Akcatepe,	,,,,,,
7				

Görsel 3.56: Devam kayıt tablosu

Verilerin toplanmasının ardından matematik entegrasyonunu sağlamak amacıyla takımlara aşağıdaki soru yöneltilir.

Öğrenci devamsızlıkları için topladığımız aylık verilerin aritmetik ortalamasını alarak sütun grafiğini aşağıdaki boşluğa çiziniz. Hangi aylarda devamsızlığın artıp ya da azaldığını tespit ederek problem durumunuzun çözümünü test ediniz.

Çözümün Sunulması

Bu aşamada öğrenciler takım arkadaşlarıyla birlikte oluşturdukları ürünleri diğer takımlara sunar. Öğretmen, sunum sırasında süreç değerlendirme formunu kullanarak öğrencilerin tasarımlarını ve uygulama sürecini değerlendirir. Öğretmen Değerlendirme Ölçeği Ek-1'de verilmiştir.

Yeniden Tasarlama / Revize Etme

Öğrencilerin aldıkları değerlendirmeler ve geri bildirimler doğrultusunda çözümlerinde gerekli iyileştirmeleri yaptıkları aşamadır. Öğrencilerin bu aşamada problemin kriter ve sınırlılıkları çerçevesinde tasarımlarını en iyi olabilecek şekilde iyileştirmeleri beklenir. Burada öğrencilerin oluşturmuş olduğu tasarım üründe iyileştirmenin nasıl yapılacağını, neden bu iyileştirmenin yapıldığını ve proje etkisini özetleyen takım hâlinde hazırlanmış bir rapor oluşturmaları istenebilir.

Öğrenci Devamı Takip Sistemi

Kararın Tamamlanması

Kararların tamamlanmasında ise öğrencinin nihai tasarımının en iyi olup olmadığına kendisinin karar vermesi amaçlanmaktadır. Öğrenciler grup üyeleriyle birlikte aşağıda yer alan soruları cevaplandırarak proje değerlendirmesi yaparlar.

1. Bu etkinliği yaparken en çok nerede zorluk yaşadınız? Bu zorlukları nasıl aştınız?
2. Bu etkinliği yaparken en çok sevdiğiniz kısım hangisiydi?
3. Bu etkinlikten neler öğrendiniz?
4. Bu projeyi yeniden planladığında geliştirmek istediğin yönler nelerdir?

Kaynakça

[1] Hynes, M., Portsmore, M., Dare, E., Milto, E., Rogers, C., Hammer, D. & Carberry, A. (2011). Infusing engineering design into high school STEM courses. <http://ncete.org/flash/pdfs/Infusing%20Engineering%20Hynes>. Erişim tarihi: 10.02.2017.

[2] Brunsell, E. (2012). The engineering desing process. Brunsell, E. (Ed.) Integrating engineering + science in your classroom. Arlington, Virginia: National Science Teacher Association[NSTA] Press.

EKLER

EK-1: Öğretmen Değerlendirme Ölçeği

Değerlendirme Kriterleri	GELİŞTİRİLMELİ	İYİ	ÇOK İYİ
Sorunu tanımlama ve analiz etme			
Olası çözümleri bulma ve en iyisini seçme			
Tasarım ürününün kabataslak çizimi veya simüle edilmiş formu			
Grup çalışması ve iş birliği			
Ürünü sunma			

EK-2: Öğrenci Öz Değerlendirme Ölçeği

Değerlendirme Kriterleri	Katılmıyorum	Kararsızım	Katılıyorum
Yüztanıma sistemleri teknolojilerini nerelerde kullanacağım hakkında bilgi sahibi oldum.			
Bilgi temelli hayat problemine yönelik bulduğum çözüm en uygun çözümdür.			
Senaryo sürecinde takım arkadaşlarımla iş birliği içinde bulunarak takım arkadaşlarıma katkı sağladım.			
Ürünü tasarlarken kriter ve sınırlılıkları dikkate aldım.			
Senaryo sürecini anlatan bir rapor oluşturdum.			

Sürat Kontrol Sistemi

Seviye

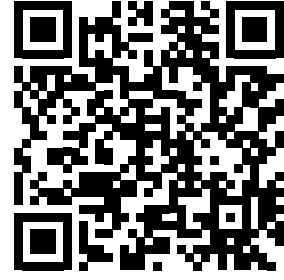
12 yaş ve üzeri

Konu

Otonom Araçlar

Süre

6 ders saati



Planla ilgili dosyalara ulaşmak için karekod taratılabilir veya bağlantı kullanılabilir.

Özet

Otonom araçların kullanıldığı öğrenme senaryosunda aracın hızını yavaşlatacak bir yazılım modeli geliştirilmesi istenmiştir. Bu öğrenme senaryosunda gerçek hayatta çok sık karşılaşılan sürücü kusurlarından kaynaklı trafik kazalarından yola çıkılarak hazırlanan bilgi temelli hayat problemine öğrencilerin çözüm bulmaları istenmiştir. Bu öğrenme senaryosunda mühendislik tasarım süreci kullanılmıştır. Öğrenme senaryosu problemin tanımlaması amacıyla öğrenciler tarafından okunur ve öğrenme süreci başlar. Öğrenciler süreç boyunca takımlar hâlinde çalışır. Öğrenciler, senaryo sonucunda hız limiti gösteren trafik levhalarını tanıyan ve buna göre gerekirse aracı yavaşlatan bir program geliştirecektir. Burada amaç öğrencilerin bilişim teknolojileri ve matematik derslerinde öğrendikleriyle mühendislik becerilerini kullanarak bir tasarım ürünü oluşturmalarıdır. Ayrıca öğrencilerin farklı disiplinleri günlük yaşamla ilişkilendirerek sorunları çözmek için kullanılabileceklerini fark etmelerini sağlamaktır. Prototip oluşturma sürecinde öğretmenler öğrencilere kodlama adımlarında rehberlik edecektir. Sürecin sonunda öğrenci takımları tasarımlarını sınıf arkadaşlarına sunacaklar. Öğrenci takımları ve prototip öğretmen tarafından hazırlanan ölçek ile değerlendirilecektir (Ek-1). Ayrıca öğrenciler öz değerlendirme ölçeğini kullanarak kendilerini değerlendireceklerdir (Ek-2).

STEM Konu Alanı 1

Bilişim Teknolojileri (Kazanımlar)

- Problem durumunu çözmek için uygun algoritmayı geliştirir.
- Projelerine uygun yapay zekâ uygulamalarını seçer.
- Blok tabanlı programlama aracını kullanarak program geliştirir.

STEM Konu Alanı 2

Mühendislik (Kazanımlar)

- Mühendislik ve tasarım ilişkisini ifade eder.
- Çevresindeki ürünleri mühendislik ve tasarım kavramları açısından ilişkilendirir.
- Mühendislik tasarım sürecindeki sınırlılıkları değerlendirir.
- Mühendislik tasarım sürecini kullanarak bir ürün tasarlar.
- Bir mühendislik projesinin içerdiği süreçleri tespit eder: Planlama, tasarım, prototip oluşturma, yürütme, kalite kontrol ve raporlama gibi aşamaları açıklar.
- Öğrenci proje çalışmasında kendisini farklı rollerdeki bir takım üyesi olarak görür, o rolün gerektirdiği çalışmaları başarıyla tamamlar.
- Çözüm önerisine yönelik düşüncelerini yazarak ve çizerek açıklar.

Sürat Kontrol Sistemi

- Başkalarının çözüm önerisine yönelik düşüncelerini dikkate alır.

1.3. Sosyal Ürün Kazanımları

- Öğrenci tasarlanan ürünü açık ve anlaşılır şekilde sınıfa sunar.

STEM Konu Alanı 3

Fen Bilimleri (Kazanımlar)

- Sürati tanımlar ve birimini ifade eder. a. Sürat birimleri olarak metre/saniye (m/sn.) ve kilometre/saat (km/sa.) dikkate alınır.
- Yol, zaman ve sürat arasındaki ilişkiyi grafik üzerinde gösterir.

STEM Konu Alanı 4

Matematik (Kazanımlar)

- En çok dört basamaklı bir doğal sayıyı, en çok iki basamaklı bir doğal sayıya böler.
- Uzunluk ölçme birimlerini tanır; metre-kilometre, metre-desimetre-santimetre-milimetre birimlerini birbirine dönüştürür ve ilgili problemleri çözer.
- Araştırma sorularına ilişkin verileri toplar, sıklık tablosu ve sütun grafiğiyle gösterir.
- Zaman ölçme birimlerini tanır, birbirine dönüştürür ve ilgili problemleri çözer.
- Veri toplamayı gerektiren araştırma soruları oluşturur.

Materyaller

Malzemeler

mBot

Online Araçlar

<https://about.draw.io/>

STEM Kariyer Bağlantıları

Yapay zekâ mühendisliği

Veri analizi

Veri bilimi

Bilgisayar mühendisliği

Bilgisayar programcılığı

ETKİNLİK SÜRECİ

Bu etkinlik mühendislik tasarım süreci kullanılarak oluşturulmuştur. Öğretmen etkinlik sürecinin başlangıcında öğrencileri takımlara ayırarak çalışmalarına rehberlik eder.

Problemin Tanımlanması

Problem tanımlanırken mutlaka tasarımın kriterleri ve sınırlılıkları belirlenmelidir.

Problem senaryosunda yer alan kriterler ve sınırlılıklar: Öğrencilerin, temel kodlama bilgisine sahip olması, yapay zekâ uygulamalarını tanıması ve Arduino ile devre tasarlaması gibi temalar olarak ortaya çıkmıştır. Öğretmen tarafından ihtiyaç ve problem durumu öğrencilere aşağıda yer alan bilgi temelli hayat problemi ile verilir.

Sürat Kontrol Sistemi

Bilgi Temelli Hayat Problemi

Karayolu trafik kazaları nedeniyle her yıl yaklaşık 1,3 milyon kişi hayatını kaybetmektedir. Karayolu trafik kazaları çoğu ülkeye gayri safi yurt içi hasıllarının %3'üne mal olmaktadır. Karayolu trafik yaralanmaları, 5-29 yaş arası çocuklar ve genç yetişkinler için önde gelen ölüm nedenidir. Trafik kazalarında 20 ila 50 milyon arasında daha fazla insan ölümcül olmayan yaralanmalara maruz kalmaktadır ve birçoğu yaralanmalarının bir sonucu olarak engelli olmaktadır [1].

Ülkemizde de trafik kazaları ciddi biçimde mal ve can kaybına yol açmaktadır. İstatistiklere göre her yıl ortalama 1 milyonun üzerinde trafik kazası meydana gelmektedir. Trafik kazalarına bağlı olarak yılda ortalama 5-6 bin kişinin öldüğü görülmektedir [2].

Tablo 3.13: Trafik Kazası İstatistikleri

Yıl	Toplam kaza sayısı	Ölümlü yaralanmalı kaza sayısı	Maddi hasarlı kaza sayısı	Ölü sayısı			Yaralı sayısı
				Toplam	Kaza yerinde	Kaza sonrası ⁽¹⁾	
2010	1 106 201	116 804	989 397	4 045	4 045	-	211 496
2011	1 228 928	131 845	1 097 083	3 835	3 835	-	238 074
2012	1 296 634	153 552	1 143 082	3 750	3 750	-	268 079
2013	1 207 354	161 306	1 046 048	3 685	3 685	-	274 829
2014	1 199 010	168 512	1 030 498	3 524	3 524	-	285 059
2015	1 313 359	183 011	1 130 348	7 530	3 831	3 699	304 421
2016	1 182 491	185 128	997 363	7 300	3 493	3 807	303 812
2017	1 202 716	182 669	1 020 047	7 427	3 534	3 893	300 383
2018	1 229 364	186 532	1 042 832	6 675	3 368	3 307	307 071
2019	1 168 144	174 896	993 248	5 473	2 524	2 949	283 234
2020	983 808	150 275	833 533	4 866	2 197	2 669	226 266
2021	1 186 353	187 963	998 390	5 362	2 421	2 941	274 615

(1) Trafik kazasında yaralanıp sağlık kuruluşuna sevk edilenlerden kazanın sebep ve tesiriyle 30 gün içinde ölenleri kapsamaktadır.

- Bilgi yoktur.

Kaynak: TÜİK [3] (<http://meb.ai/UomZ8Uq>)

Trafik kazalarının nedenleri incelendiğinde kazaya neden olan kusurlar içinde sürücü kusurları %87,1 ile ilk sıradadır (TÜİK). Sürücü kusurlarının başında ise "araç süratini yol, hava ve trafiğin gerektirdiği şartlara uydurmamak" gelmektedir.

Ortalama süratte bir artış, hem bir kazanın meydana gelme olasılığını hem de kazanın ciddiyetini doğrudan artırmaktadır. Örneğin, ortalama süratteki her %1'lik artış, ölümcül kaza riskinde %4'lük bir artışa ve ciddi çarpışma riskinde %3'lük bir artışa neden olmaktadır. Sürate bağlı olarak arabaların önünden çarptığı yayaların ölüm riski hızla artar. Örneğin bir araç 50 km/sa. yerine 65 km/sa. süratle gittiğinde kaza anında yayanın ölüm riski 4,5 kat artmaktadır. Arabadan arabaya yandan çarpmalarda, araba yolcuları için ölüm riski 65 km/sa.'te %85'tir.

Birleşmiş Milletler Genel Kurulu, karayolu trafik kazalarından kaynaklanan küresel ölüm ve yaralanma sayısını 2030 yılına kadar yarıya indirmeyi hedeflemektedir (DSÖ, 2022). Yapay zekâ sistemine sahip otonom araçların kullanılmasının ve yaygınlaştırılmasının trafik kazalarını ciddi bir miktarda azaltacağı düşünülmektedir.

- Sizce sürat nedir? Hangi faktörler bir cismin süratini etkileyebilir?

Sürat Kontrol Sistemi

- Süratin ne olduğunu ve hangi faktörlere bağlı olduğunu öğrendik. Sizce süratin birimini nasıl ifade edebiliriz?

Alınan Yol (km)	0	50	100	150	200
Geçen Zaman (sa.)	0	1	2	3	4

- Bir bisikletlinin yol zaman tablosu yukarıdaki gibidir. Bisikletlinin süratini hesaplayarak, yol-zaman ve sürat-zaman grafiklerini çiziniz (Yukarıdaki tablo örnek olarak verilmiştir.).
- Bu tablonun yol-zaman ve sürat-zaman grafiklerini Sagemodeler de simüle ediniz ve anlık grafik değişimlerinin nedenlerini araştırarak raporlaştırınız (<https://sagemodeler.concord.org/>).
- Birleşmiş Milletlerin hedefine ulaşması ve aşırı süratten kaynaklanan sürücü kusurlarının neden olduğu kazaları en aza indirmek için nasıl bir sistem geliştirilebilir?

Geliştirilecek olan modelin sahip olması gereken kriterler şunlardır.

- Hız uyarı levhasını tanımalıdır.
- Hız uyarı levhasını tanıyarak mBot aygıtının hızını ayarlamalıdır.

Probleme Yönelik İhtiyacın Belirlenmesi

Mühendisler akla gelen ilk çözümü sunmak yerine problem hakkında araştırma yaparak geçmişte kullanılan yöntemleri gözden geçirerek farklı çözüm yollarını bulmaya çalışırlar [4].

Burada öğrenciler çözüme yönelik ihtiyaçları belirlemek için yapay zekâ ve otonom araçlar ile ilgili araştırma yaparak bilgi toplar. Ayrıca öğrencileri yönlendirmek amacıyla aşağıdaki sorular sorularak araştırmaları istenebilir.

- Sürücü hataları nasıl engellenebilir?
- Sürücüsüz araçları daha önce duydunuz mu? Sürücüsüz araçlar nasıl çalışır?

Olası Çözümlerin Geliştirilmesi

Günlük yaşam problemlerinde tek bir doğru çözüm yoktur, genellikle birden fazla çözüm vardır [5]. Bu aşamada öğrenciler tasarım kriterlerini ve sınırlılıklarını dikkate alarak edindikleri bilgilere göre probleme yönelik çözüm önerileri geliştirirler. Öğrencilerin ürettikleri çözüm önerileri tabloya yazılır (Tablo 3.14).

Tablo 3.14: Çözüm Önerileri ve Sınırlılıklar

Çözüm Önerileri	Sınırlılıklar
1.	
2.	
3.	

Öğrenciler olası çözümlerini takım arkadaşlarıyla birlikte belirleyeceklerdir. Beyin fırtınası ile öğrencilerin oluşturdukları çözüm önerileri arasından problemin sınırlılıklarını dikkate alarak en uygun çözüme ulaşmaları beklenir. Öğrenciler oluşturmuş oldukları çözüm önerilerinin sınırlılıklarını beyin fırtınası ile kendileri belirleyeceklerdir. Öğretmen burada öğrencilerin beyin fırtınası yapmasını teşvik eder ve öğrenciler ile çözüm önerilerini değerlendirir. Öğrencilerin takım arkadaşlarıyla en uygun çözümü seçmelerine rehberlik eder.

Sürat Kontrol Sistemi

En İyi Çözümün Seçilmesi

Öğrencilerin birçok çözüm önerisini takım arkadaşlarıyla tanımlamasından sonra problem durumunun çözümüne yönelik bilgi temelli hayat probleminde yer alan kriterler ve sınırlılıklar çerçevesinde yapılan analizlerle en uygun çözüm için karar vermeleri beklenir. En uygun çözümün seçilmesinin ardından öğrencilerden mühendislik becerilerini geliştirmeye katkı sağlamak amacıyla problemin çözümüne ilişkin bileşenleri ve aralarındaki ilişkiyi belirlemeleri beklenir. Öğrencilerin modelin algoritmasını oluşturmaları ve algoritmayı anlatmaları beklenir. Ayrıca öğrencilerden kâğıt, kalem veya <https://about.draw.io/> sitesini kullanarak bir akış diyagramı oluşturmaları ve yapacakları programın algoritmasını görselleştirmeleri beklenir.

Prototipin Yapılması

Bu aşamada seçilen çözüm önerisine yönelik tasarımı sunmak ve ayrıntılarını göstermek için prototip oluşturulur.

Trafik hız uyarı işaretlerini tanıyan (prototip için 80 ve 120 hız levhaları) eğer hızı limitin üstündeyse otomatik olarak yavaşlayan (araba kuklasının yavaşlaması) ve bağlı aygıt (mBot) mesaj göndererek yavaşlamasını sağlayan bir uygulama geliştirilecektir.

Ön Hazırlık

Öğretmen öğrencilere TM kullanarak model oluşturma konusuyla ilgili bilgi verir.

Bu aşamada aygıtın mBlock programı ile bağlantısının yapılması ve uzantıların yüklenmesi gibi işlemler yapılır.



Aygıtın Bağlanması

Aygıtlar bölümünde mBot seçilerek projeye dâhil edilir. mBot bilgisayara bağlanarak aygıtlar bölümünden mBlock ile bağlantısı gerçekleştirilmelidir. Bu aşamada kablosuz bağlantı seçilirse uygulamanın test edilmesi daha kolay olur.


mBlock Uzantıları

Uygulamanın geliştirebilmesi için gerekli eklentiler yüklenmelidir.

Tablo 3.15: Kullanılan Eklentiler

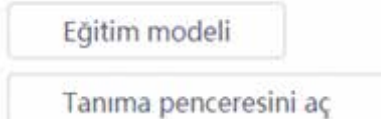
Eklentinin Adı	Yüklendiği Bileşen	Kullanım Amacı
 Text to Speech Geliştiriciler: MIT Media Lab	Kuklalar	Metinleri seslendirmek için kullanılır. Bu eklenti ile programda sesli mesaj verilebilir.
 Yükleme modu yayını By mBlock official	Kuklalar Aygıtlar	Kukla ile aygıt arasındaki iletişimi sağlayarak mBot ile kuklanın haberleşmesini sağlamaktadır. mBlock programında aygıt uzantısı mBot ve yükleme modu yayını eklenir.

Sürat Kontrol Sistemi

Eklentinin Adı	Yüklendiği Bileşen	Kullanım Amacı
 <p>Makine Öğrenimi By mBlock official</p> <p>Makine Öğrenimi ile, program yapmak zorunda değilsiniz, ancak yine de bilgisayarın bir şeyler öğrenmek için eğitebilir ve insanıman</p>	Kuklalar	TM (Teachable Machine) bir model eğitmek için kullanılır.

TM ile Model Oluşturma

TM uzantısı kullanılarak 3 sınıflı bir model oluşturulur. Eğitim modeli butonuna tıklanır (Görsel 3.57).



Görsel 3.57: TM Eklentisi

Açılan pencereden TM modeli için sınıflara örnekler verilerek (2. Ünitede ayrıntılı biçimde anlatıldığı şekilde) 80 km ve 120 km hız levhaları ve sürücünün görüntüsü (diğer sınıfları iyi ayırmak için) üç sınıf oluşturulur. Hız levhaları için önceden oluşturulmuş renkli kıktılar veya telefonda görüntüler kullanılabilir (Görsel 3.58).



Görsel 3.58: Model eğitimi

Model kameranın önüne uyarı levhaları veya fotoğraflar gösterilerek sonuçları kontrol edilir. Modelin performansı yeterli görülürse modeli kullan butonuna basılır. Model uyarı levhalarını tanımakta başarısızsa daha çok örnek kullanılarak eğitilebilir. Model eğitildikten sonra modele ait sınıf bilgileri görünecektir (Görsel 3.59).

Sürat Kontrol Sistemi



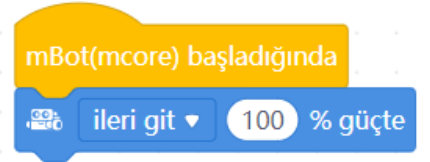
Görsel 3.59: Modele ait bloklar

Kodlama

Bu bölümde prototipe ait kod bloklarına yer verilmiştir. Kod blokları aygıt kodları ve kukla kodları olmak üzere iki bölümde verilmiştir.

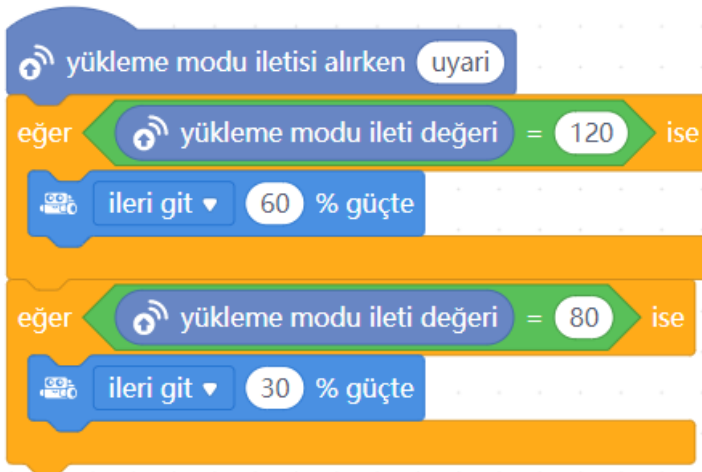
Aygıt Kodları

mBot ilk çalıştığında tam hızla hareket ettirilir (Görsel 3.60).



Görsel 3.60: mBot kodları

Yükleme kodu iletisi geldiğinde iletinin değeri kontrol edilerek mBot'un hızı gerekli şekilde azaltılır (Görsel 3.61).



Görsel 3.61: Yükleme modu iletisi

Sahne ve Kukla Ayarları

Model simülasyonu için uygun bir dekor (city night2) ve kuklalar kütüphaneden eklenir. Aracın hareketini sağlamak için arka plandaki kuklalar geriye doğru hareket ettirilir (Görsel 3.62).

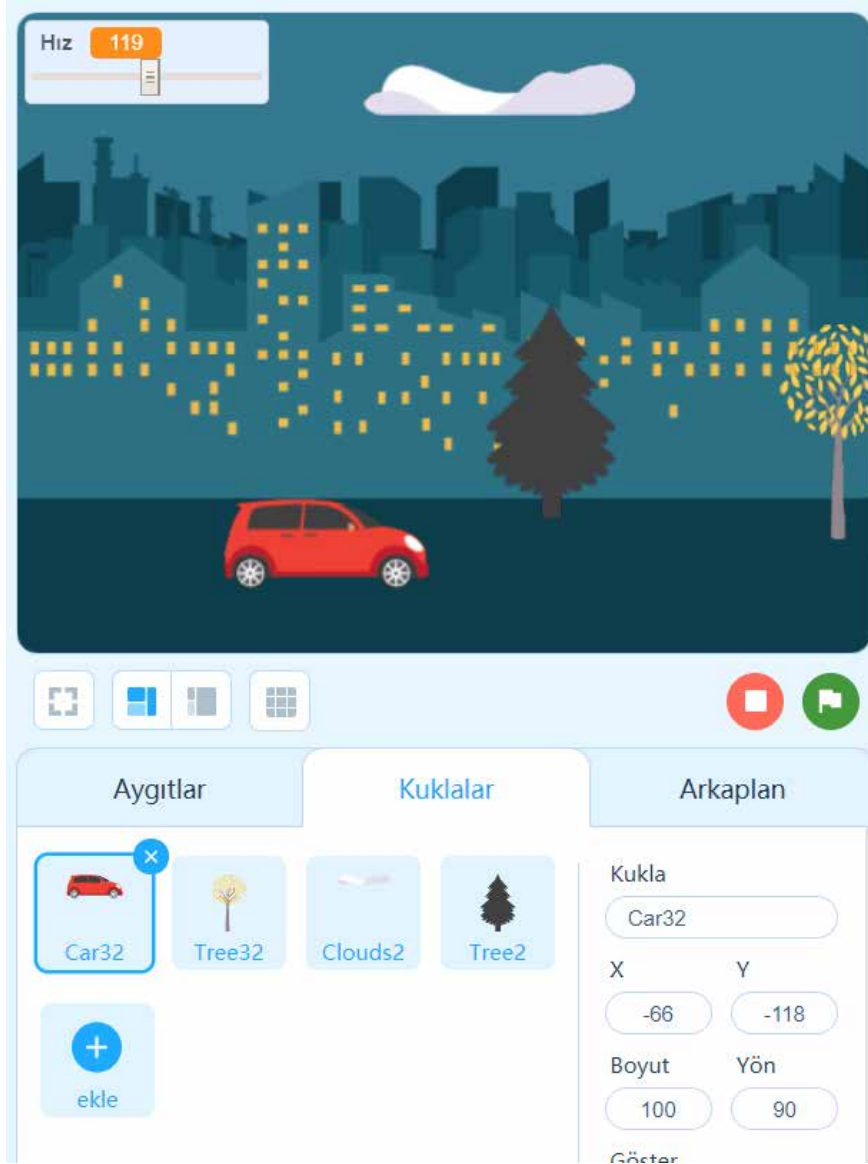
Kuklalar

Car32

Tree32

Clouds2

Tree2

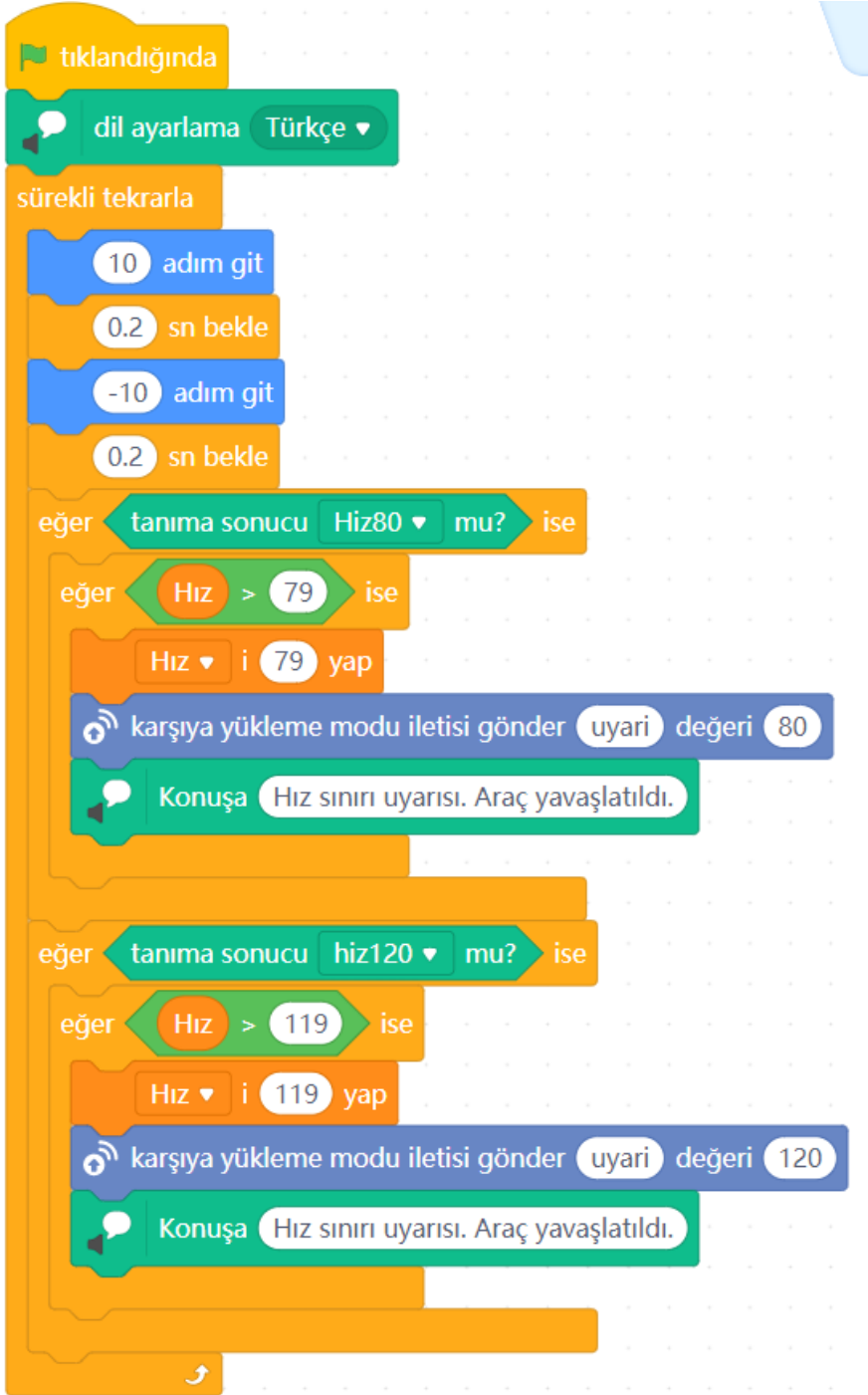


Görsel 3.62: Sahne ve kukla ayarları

Sürat Kontrol Sistemi

Araba (Car32) kuklasının kodları

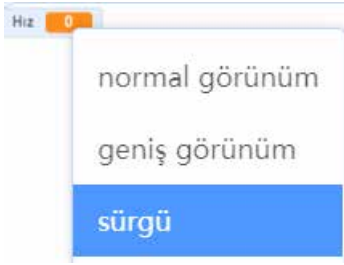
Yeşil Bayrak tıkladığında çalışacak kodlar (Görsel 3.63)



The image shows a Scratch script for a speed control system. The script starts with a 'clicked' event block. The first block is 'set language to Turkish'. This is followed by a 'repeat' loop with a 'forever' option. Inside the loop, there are four blocks: 'move 10 steps', 'wait 0.2 seconds', 'move -10 steps', and 'wait 0.2 seconds'. After the loop, there are two conditional blocks. The first is 'if (tanima sonucu Hiz80 mu?) ise'. Inside this block, there is another 'if (Hız > 79) ise' block. This inner block contains three sub-blocks: 'set speed to 79', 'send message to karşıya yükleme modu iletilisi gönder uyarı değeri 80', and 'say Hız sınırını uyarısı. Araç yavaşlatıldı.'. The second main conditional block is 'if (tanima sonucu hiz120 mu?) ise'. It contains an 'if (Hız > 119) ise' block with three sub-blocks: 'set speed to 119', 'send message to karşıya yükleme modu iletilisi gönder uyarı değeri 120', and 'say Hız sınırını uyarısı. Araç yavaşlatıldı.'. The script ends with a 'loop' block.

Görsel 3.63: Car 32 kuklası Yeşil bayrak tıkladığında çalışacak kodlar

Sürat Kontrol Sistemi



Değişkenler bölümünden hız değişkeni oluşturulur ve sahnede değişken üstüne fare ile sağ tıklanarak sürgü olarak ayarlanır.

Kodların Açıklaması

1. Yeşil bayrak tıklandığında.
2. Dil Türkçe olarak ayarlanır.
3. Sürekli olarak tekrarlanır.
4. Kukla 10 adım gider 0.2 saniye bekletilir ve -10 adım daha gider 0.2 saniye bekletilir. Bu kod araç kuklasına hareket efekti vermek için kullanılır.
5. TM modeline kameradaki görüntü verilir.
6. Eğer 80 hız uyarı levhası algılanmışsa ve aracın hızı 80 veya daha fazla ise hız 79'a düşürülür. Yükleme modu iletisi gönderilir (mBot için) ve sesli uyarı mesajı verilir.
7. Eğer 120 hız uyarı levhası algılanmışsa ve aracın hızı 120 veya daha fazla ise hız 119'a düşürülür. Yükleme modu iletisi gönderilir (mBot için) ve sesli uyarı mesajı verilir.

Ağaç 1 (Tree 32) kuklasının kodları

Yeşil Bayrak tıklandığında çalışacak kodlar (Görsel 3.64)



Görsel 3.64: Tree 32 kuklası yeşil bayrak tıklandığında çalışacak kodlar

Sürat Kontrol Sistemi

Kodların Açıklaması

1. Yeşil bayrak tıklandığında.
2. Sürekli olarak tekrarla.
3. Arka planın sola hareket etmesi ve hız arttıkça hızlanması için gerekli hesaplama.
4. Sol kenara geldiğinde.
5. Sağ kenardan tekrar sahneye girsin.
6. Sahneyi monotonluktan kurtarmak için kostüm değiştirilerek farklı ağaç veya sokak lambası görünmesi sağlanır.

Bulutlar (Clouds2) kuklasının kodları

Yeşil Bayrak tıklandığında çalışacak kodlar (Görsel 3.65)



Görsel 3.66: Clouds 2 yeşil bayrak tıklandığında çalışacak kodlar

Kodların Açıklaması

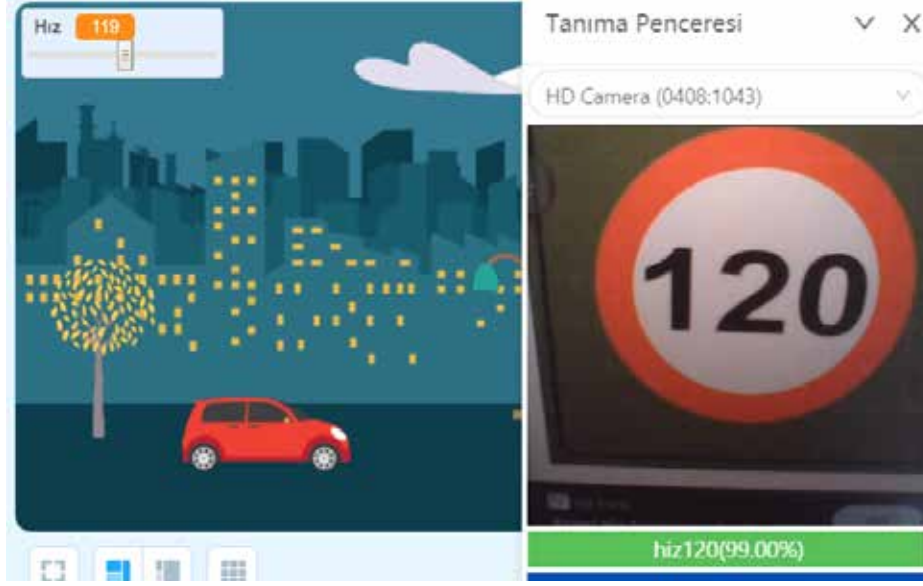
1. Yeşil bayrak tıklandığında.
2. Sürekli olarak tekrarla.
3. Arka planın sola hareket etmesi ve hız arttıkça hızlanması için gerekli hesaplama.
4. Sol kenara geldiğinde.
5. Sağ kenardan tekrar sahneye girsin.
6. Sahneyi monotonluktan kurtarmak için kostüm değiştirilerek farklı bulutların görünmesi sağlanır.

Çözümü Test Etme ve Değerlendirme

1. mBot bilgisayara bağlanır.
2. Yeşil bayrağa tıklanarak program çalıştırılır.
3. mBot tam hızla hareket edecektir.

Sürat Kontrol Sistemi

4. Hız değişkeni süngüsü hareket ettirilerek araç hızlandırılır. Simülasyonda aracın hızlandığı görülür.
5. Tanıma penceresi sürekli açıktır. Kameraya uyarı levhalarından biri gösterilir.
6. Uyarı levhasına göre eğer hız sınırı aşılmışsa sesli uyarı verir ve hız azaltılır. Eş zamanlı olarak mBot aygıtının da hızı değişir (Görsel 3.67).



Görsel 3.67:

Tanıma penceresi

7. Diğer uyarı levhası da gösterilerek uygulama test edilir. Bu işlem birkaç kez tekrarlanabilir.

Çözümün Sunulması

Bu aşamada öğrenciler takım arkadaşlarıyla birlikte oluşturdukları ürünleri diğer takımlara sunar. Öğretmen, sunum sırasında süreç değerlendirme formunu kullanarak öğrencilerin tasarımlarını ve uygulama sürecini değerlendirir. Öğretmen Değerlendirme Ölçeği Ek-1'de verilmiştir.

Yeniden Tasarlama / Revize Etme

Öğrencilerin aldıkları değerlendirmeler ve geri bildirimler doğrultusunda çözümlerinde gerekli iyileştirmeleri yaptıkları aşamadır. Öğrencilerin bu aşamada problemin kriter ve sınırlılıkları çerçevesinde tasarımlarını en iyi olabilecek şekilde iyileştirmeleri beklenir. Burada öğrencilerin oluşturmuş olduğu tasarım üründe iyileştirmenin nasıl yapılacağı, neden bu iyileştirmenin yapıldığını ve proje etkisini özetleyen takım hâlinde hazırlanmış bir rapor oluşturmaları istenebilir.

Kararın Tamamlanması

Kararların tamamlanmasında ise öğrencinin nihai tasarımının en iyi olup olmadığına kendisinin karar vermesi amacıyla Ek-2'de verilen öğrenci öz değerlendirme ölçeği kullanılacaktır.

Öğrenciler grup üyeleriyle birlikte aşağıda yer alan soruları cevaplandırarak proje değerlendirmesi yaparlar.

1. Bu etkinliği yaparken en çok nerede zorluk yaşadınız? Bu zorlukları nasıl aştınız?

Sürat Kontrol Sistemi

2. Bu etkinliği yaparken en çok sevdiğiniz kısım hangisiydi?
3. Bu etkinlikten neler öğrendiniz?
4. Bu projeyi yeniden planladığında geliştirmek istediğin yönler nelerdir?

Kaynakça

EKLER

EK-1: Öğretmen Değerlendirme Ölçeği

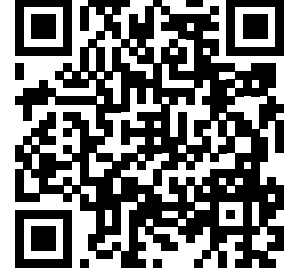
Değerlendirme Kriterleri	GELİŞTİRİLMELİ	İYİ	ÇOK İYİ
Trafik işaretlerini tanıma sistemlerinde kullanılan teknolojiler hakkında bilgi sahibi olma			
Olası çözümleri bulma ve en iyisini seçme			
Tasarım ürününün kabataslak çizimi veya simüle edilmiş formu			
Grup çalışması ve iş birliği			
Ürünü sunma			

Ek 2 Öğrenci Öz Değerlendirme Ölçeği

Değerlendirme Kriterleri	Katılmıyorum	Kararsızım	Katılıyorum
Otonom araçlarda kullanılan yapay zekâ teknolojileri hakkında bilgi sahibi oldum.			
Bilgi temelli hayat problemine yönelik bulduğum çözüm en uygun çözümdür.			
Senaryo sürecinde takım arkadaşlarımla iş birliği içinde bulunarak takım arkadaşlarıma katkı sağladım.			
Ürünü tasarlarken kriter ve sınırlılıkları dikkate aldım.			
Senaryo sürecini anlatan bir rapor oluşturdum.			

Akıllı Ev

Seviye	12 yaş ve üzeri
Konu	Ev Otomasyon Sistemleri
Süre	6 ders saati



Planla ilgili dosyalara ulaşmak için karekod taratılabilir veya bağlantı kullanılabilir.

Özet

Bu öğrenme senaryosu, ev otomasyon sistemlerini kullanan sesle kontrol edilen ve elektrik enerjisinden tasarruf sağlayan akıllı bir ev tasarlanması üzerine odaklanır. Engelli bir bireyin günlük yaşamında yaşadığı zorluklardan esinlenerek yazılan bilgi temelli hayat probleminin, öğrenciler tarafından okunmasıyla öğrenme senaryosu süreci başlar. Öğrenme süreci öğrencilerin elektrik tasarrufunun aile ve ülke ekonomisine katkısını fark etmelerini sağlayacak sorular ile devam eder. Bu öğrenme senaryosunda mühendislik tasarım süreci kullanılmıştır. Öğrenciler süreç boyunca takımlar hâlinde çalışır. Öğrencilerden senaryo sonucunda sesle kapıyı, temizlik robotunu kontrol edebilecekleri, müzik dinleyebilecekleri, ışıkları ve fanı açabilecekleri bir akıllı ev prototipi oluşturmaları beklenmektedir. Burada amaç öğrencilerin bilişim teknolojileri, fen bilimleri ve matematik derslerinde öğrendiklerini, mühendislik becerilerini kullanarak bir tasarım ürünü oluşturmalarıdır. Ayrıca öğrencilerin farklı disiplinlerini günlük yaşamla ilişkilendirerek sorunları çözmek için kullanılabileceklerini fark etmelerini sağlamaktır. Prototip oluşturma sürecinde öğretmenler öğrencilere kodlama adımlarında rehberlik edecektir. Sürecin sonunda öğrenci takımları tasarımlarını sınıf arkadaşlarına sunacaklar. Öğrenci takımları ve prototip öğretmen tarafından hazırlanan ölçek ile değerlendirilecektir (Ek-1). Ayrıca öğrenciler öz değerlendirme ölçeğini kullanarak kendilerini değerlendireceklerdir (Ek-2).

STEM Konu Alanı 1

Bilişim Teknolojileri (Kazanımlar)

- Problem durumunu çözmek için uygun algoritmayı geliştirir.
- Projelerine uygun yapay zekâ uygulamalarını seçer.
- Arduino ile devre kurulumu yapar.
- Blok tabanlı programlama aracını kullanarak program geliştirir.

STEM Konu Alanı 2

Mühendislik (Kazanımlar)

- Mühendislik ve tasarım ilişkisini ifade eder.
- Çevresindeki ürünleri mühendislik ve tasarım kavramları açısından ilişkilendirir.
- Mühendislik tasarım sürecindeki sınırlılıkları değerlendirir.
- Mühendislik tasarım sürecini kullanarak bir ürün tasarlar.
- Bir mühendislik projesinin içerdiği süreçleri tespit eder: Planlama, tasarım, prototip oluşturma, yürütme, kalite kontrol ve raporlama gibi aşamaları açıklar.
- Öğrenci proje çalışmasında kendisini farklı rollerdeki bir takım üyesi olarak görür, o rolün gerektirdiği çalışmaları başarıyla tamamlar.

Akıllı Ev

- Çözüm önerisine yönelik düşüncelerini yazarak ve çizerek açıklar.
- Başkalarının çözüm önerisine yönelik düşüncelerini dikkate alır.

1.3. Sosyal Ürün Kazanımları

- Öğrenci tasarlanan ürünü açık ve anlaşılır şekilde sınıfa sunar.

STEM Konu Alanı 3

Fen Bilimleri (Kazanımlar)

- Elektrik enerjisinin bilinçli ve tasarruflu kullanılmasının aile ve ülke ekonomisi bakımından önemini tartışır.
- Evlerde elektriği tasarruflu kullanmaya özen gösterir.

STEM Konu Alanı 4

Matematik (Kazanımlar)

- Veri toplamayı gerektiren araştırma soruları oluşturur.

Materyaller

Malzemeler

1 Adet Arduino UNO R3

1 Adet Servo motor

1 Adet Kırmızı LED

1 Adet Yeşil LED

1 Adet Sarı LED

1 Adet Orta boy breadboard

Jumper kablo

Karton, (mukavva veya strafor)

Çevrim İçi Araçlar

<https://ide.mblock.cc/>

<https://www.tinkercad.com/>

STEM Kariyer Bağlantıları

Yapay zekâ mühendisliği

Veri analizi

Veri bilimi

Bilgisayar mühendisliği

Bilgisayar programcılığı

ETKİNLİK SÜRECİ

Bu etkinlik mühendislik tasarım süreci kullanılarak oluşturulmuştur. Öğretmen etkinlik sürecinin başlangıcında öğrencileri takımlara ayırarak çalışmalarına rehberlik eder.

Problemin Tanımlanması

Problem tanımlanırken mutlaka tasarımın kriterleri ve sınırlılıkları belirlenmelidir.

Problem senaryosunda yer alan kriterler ve sınırlılıklar: Öğrencilerin, temel

Akıllı Ev

kodlama bilgisine sahip olması, yapay zekâ uygulamalarını tanıması ve Arduino ile devre tasarlaması gibi temalar olarak ortaya çıkmıştır. Öğretmen tarafından ihtiyaç ve problem durumu öğrencilere aşağıda yer alan bilgi temelli hayat problemi ile verilir.

Bilgi Temelli Hayat Problemi

Kemal amca 69 yaşındadır. Bedensel olarak engeli bulunmakta ve zor yürümektedir. Yalnız yaşayan Kemal amca evinde birçok işini yürütle zar zor yapabilmektedir. Kemal amca, sabah haberlerini izlediği sırada zil sesiyle irkildi. Gelen kişinin kim olduğunu kapı deliğinden göremediği için tedirgin olarak "Kim o?" diye seslendi. Kemal amca torununun sesini duyunca rahatlayarak kapıyı açtı. Kemal amcanın bilgisayar mühendisi olan torunu Mustafa, birkaç günlüğüne dedesinin yanına ziyarete gelmişti. Mustafa uyumak için odasına geçerken dedesinin odasının ve koridorun ışığının hâlâ açık olduğunu fark etti. Dedesinin yanına giden Mustafa dedesine bunun nedenini sordu. Dedesi de yürüme engeli olduğu için ışığı kapatınca yatağına geçmekte zorlandığını, bu yüzden ışıkları açık bırakmak zorunda kaldığını, elektrik israfı olduğu için üzüldüğünü Mustafa'ya anlattı. Dedesinin durumuna üzülen Mustafa, dedesinin sorunlarına çözüm bulmak üzere düşünmeye başladı. Siz Mustafa'nın yerinde olsanız dedesinin kapıyı ve ışıkları kolay kapatmasını ayrıca temizlik robotunun, müzik sisteminin ve fanın çalışmasını sağlayacak bir uygulamayı nasıl oluşturursunuz?

- Sizler evinizde elektrik tasarrufu sağlamak için hangi önlemleri alıyorsunuz?
- Kemal amcanın elektrik enerjisini tasarruflu kullanmaması aile ve ülke ekonomisini nasıl etkiler?

Probleme Yönelik İhtiyacın Belirlenmesi

Mühendisler akla gelen ilk çözümü sunmak yerine problem hakkında araştırma yaparak geçmişte kullanılan yöntemleri gözden geçirerek farklı çözüm yollarını bulmaya çalışırlar [1].

Burada öğrenciler çözüme yönelik ihtiyaçları belirlemek için akıllı ev sistemlerini araştırarak bilgi toplar. Bunun için öğrencilerden şu soruları araştırmaları istenir.

- Ev otomasyon sistemleri hakkında bilginiz var mı?
- Ev otomasyon sistemlerinde hangi yapay zekâ teknolojileri kullanılmaktadır?

Olası Çözümlerin Geliştirilmesi

Günlük yaşam problemlerinde tek bir doğru çözüm yoktur, genellikle birden fazla çözüm vardır [2]. Bu aşamada öğrenciler tasarım kriterlerini ve sınırlılıklarını dikkate alarak edindikleri bilgilere göre probleme yönelik çözüm önerileri geliştirirler. Öğrencilerin ürettikleri çözüm önerileri tabloya yazılır (Tablo 1).

Tablo 3.16: Çözüm Önerileri ve Sınırlılıklar

Çözüm Önerileri	Sınırlılıklar
1.	
2.	
3.	

En İyi Çözümün Seçilmesi

Öğrencilerin birçok çözüm önerisini takım arkadaşlarıyla tanımlamasından sonra problem durumunun çözümüne yönelik bilgi temelli hayat probleminde yer alan kriterler ve sınırlılıklar çerçevesinde yapılan analizlerle en uygun çözüm için karar vermeleri beklenir.

Akıllı Ev

En uygun çözümün seçilmesinin ardından öğrencilerden mühendislik becerilerini geliştirmeye katkı sağlamak amacıyla çözümü iki boyutlu olarak bir kâğıda çizmeleri beklenir. Yapılan çizim üzerinde öğrenciler kullanacakları malzemeleri seçerek göstermelidir.

Daha sonra öğrencilerden hazırlayacakları prototipin simülasyonunu gözlemlenmeleri için Tinkercad üzerinden taslak devre modellerini oluşturmaları istenir. Tasarım fikri için öğrencilerin tasarladıkları devreyi Arduino ile kurmaları ve mBlock ile yapay zekâ eklentilerini kullanarak kodlama yapmaları gerekmektedir.

Problem durumu analiz edildiğinde öğrencilerin hazır bulunuşlukları da dikkate alınarak problemi çözmeye yönelik geliştirilecek bir projenin iki ana bileşeni olduğu görülmektedir:

1. Devre Tasarımı
2. Kodlama

Prototipin Yapılması

Bu aşamada seçilen çözüm önerisine yönelik tasarımı sunmak ve ayrıntılarını göstermek için akıllı ev prototipi oluşturulur.

Ön Hazırlık

Bu aşamada öğretmen öğrencilere devre elemanlarını tanıtır.

Servo motor 0-180 derece arasında bir açı ile dönebilmektedir. Servo motorun çalışabilmesi için PWM pinine bağlanması gerekir. Servo motorun farklı açılarda dönmesi projede kapı kilidinin açılıp kapanması için kullanılır. Servo 0 derece açısı kapının kapalı olmasını, 180 derece açısı ise kapının açık olması için kullanılır.

Arduino ve gerekli devre elemanları kullanılarak akıllı evdeki kapı ve ışıklar için devre tasarlanır. Devreyi yapay zekâ eklentileriyle kontrol etmek için ise mBlock kullanılır.

Prototipin özellikleri

Sesli komut ile şu işlemleri yapabilmektedir.

Işıkları açma ve kapama (devre üstündeki 3 LED)

Kapıyı açma ve kapama (devre üstündeki servo motor)

Müzik oynatma (kukla)

Fan (kukla)

Temizlik (kukla)

Aygıtın Bağlanması

Aygıtlar bölümünde ilgili Arduino kartı (Bu projede UNO 3 kullanılmıştır.) seçilerek projeye dâhil edilir. Arduino bilgisayara bağlanarak aygıtlar bölümünden mBlock ile bağlantısı gerçekleştirilmelidir. Yükleme işlemi yapılmalıdır. Bağlantı yapıldıktan sonra Arduino'ya firmware yüklenerek mBlock programı ile bağlantılı duruma gelmesi sağlanır. Arduino canlı moda kullanıma hazırlanmalıdır.

mBlock Uzantıları

Uygulamanın geliştirebilmesi için gerekli eklentiler yüklenmelidir.

Akıllı Ev

Tablo 3.17: Kullanılan Eklentiler

Eklentinin Adı	Yüklendiği Bileşen	Kullanım Amacı
 Cognitive Services By reBlock official	Kuklalar	Ses tanıma işlemi için kullanılır.
 Text to Speech Geliştirildi: MIT Media Lab Make your projects talk. (not available in China yet)	Kuklalar	Yazıları seslendiriyor. Programda sesli mesaj ve yönlendirme için kullanılır.

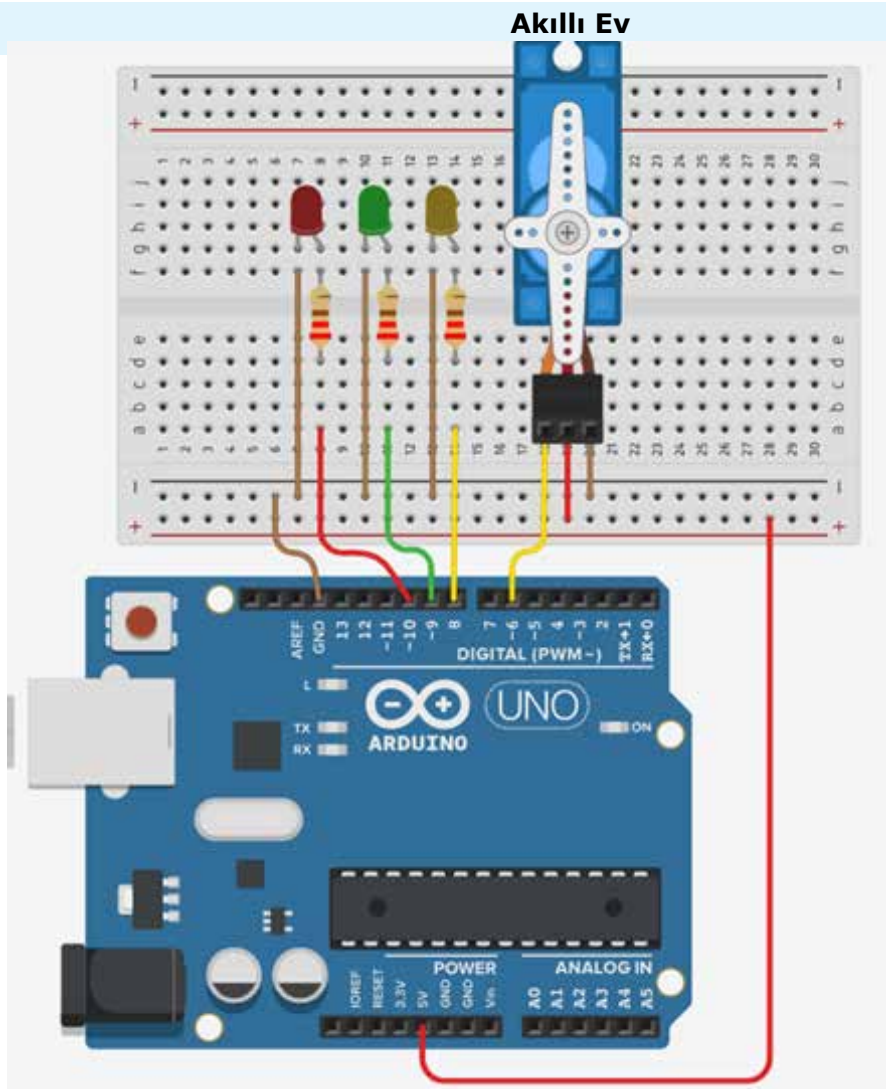
Devre Tasarımı ve Kurulumu

Öğrenciler akıllı ev prototipi için karton, strafor gibi istediği herhangi bir malzemeyi seçebilir. Prototipin kodlaması için gerekli malzemeler Tablo 3.18’de sunulmuştur.

Tablo 3.18: Kullanılan Malzemeler

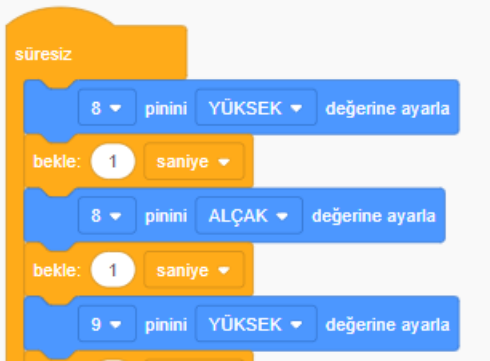
Ad	Miktar	Bileşen	Kullanım Amacı
B1	1	Arduino Uno R3	-
B2	1	Servo motor	Kapı kontrolü
B3	1	Kırmızı LED	Oda 1 ışık
B4	1	Yeşil LED	Oda 2 ışık
B5	1	Sarı LED	Oda 3 ışık
B6	1	Orta boy breadboard	-
B7	-	Jumper kablo	-

Simülatör kullanarak devrenin tasarlanması gerçek devre kurulumunda karşılaşılabilecek hataların ve arızaların önüne geçilmesini sağlamaktadır (Görsel 3.68).

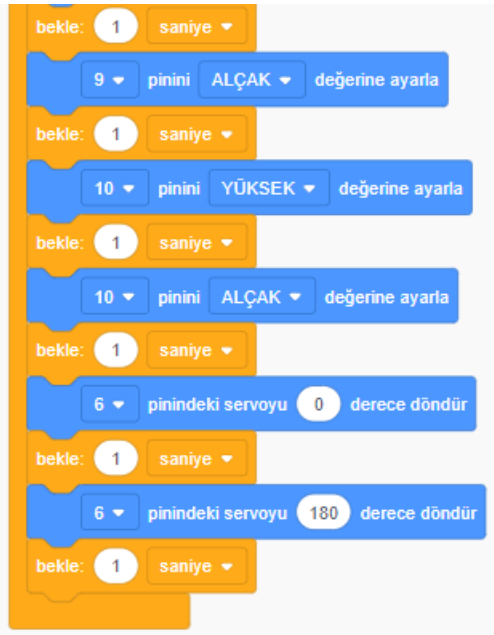


Görsel 3.68: Tinkercad devre şeması

Devrenin istendiği gibi çalışıp çalışmadığını test etmek için bir kodlama yapılır (Görsel 3.14). Bu kodlamada amaç, devrede 8, 9 ve 10 numaralı pinlere bağlı LED'ler birer saniye aralıklarla yanıp sönmeleri ve sonrasında servo motorun önce 0 dereceye dönmesi, 1 saniye sonra da 180 dereceye dönmesidir (Görsel 3.69-a-b).

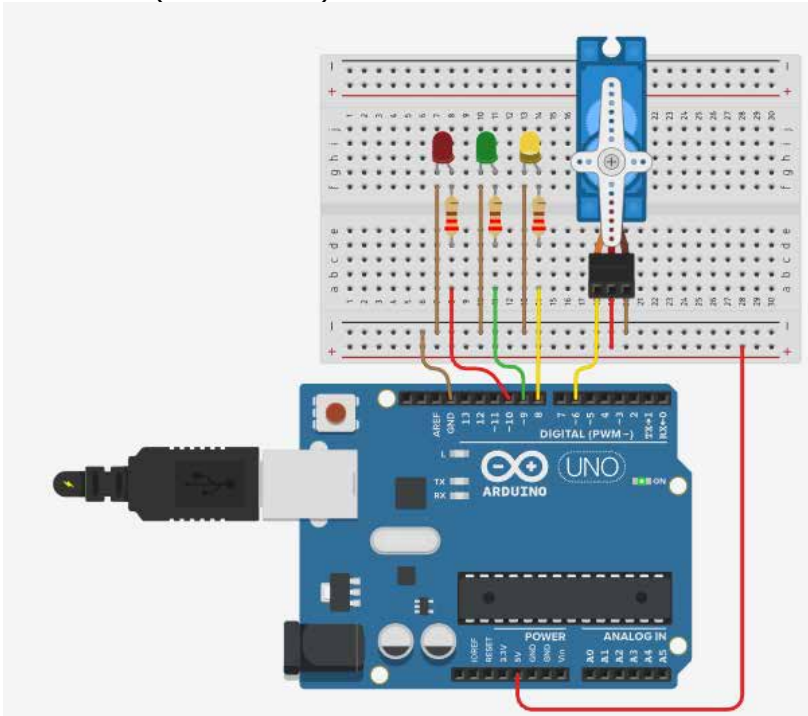


Görsel 3.69-a: Tinkercad blok kodlama



Görsel 3.69-b: Tinkercad blok kodlama

Kodlama yapıldıktan sonra simülasyon başlatılır. LED'lerin sırasıyla yanıp sönmesi ve servo motorun önce 0 dereceye sonra 180 dereceye dönmesi gerekir. Bu işlem sürekli tekrarlanır (Görsel 3.70).



Görsel 3.70: Tasarımın çalıştırılması

Simülasyon çalışırken bir sorunla karşılaşılmazsa tasarlanan devre kurulur.

Akıllı Ev

Devre Kurulumu

Yapılan devre tasarımına uygun bir biçimde devre kurulumu yapılır.

Sarı LED D8 pinine, yeşil LED D9 pinine ve kırmızı LED D10 pinine bağlanır. Servo motor ise D6 pinine bağlanmıştır.

Kodlama

Bu bölümde prototipe ait kod bloklarına yer verilmiştir. Kod blokları aygıt kodları ve kukla kodları olmak üzere 2 bölümde verilmiştir.

Kukla Kodları

Bu bölümde prototipe ait kodlar yer almaktadır. Öncelikle mBlock programında sahne ve kuklalar ayarlanır.

Sahne ve Kukla Ayarları


Sahne arka planı olarak bir oda (bedroom) ayarlanır. Kostümler bölümünde arka planın bir kopyası oluşturularak odanın ışığı kapalı olduğunda arka plan olarak kullanılmak üzere kostüm düzenleme ekranında siyah renge boyanır (Görsel 3.71).



Görsel 3.71: Sahne ve kukla ayarları

Sahneye eklenecek kuklalar ve işlevleri şöyledir (Tablo 3.19).

Tablo 3.19: Kuklalar ve Görevleri

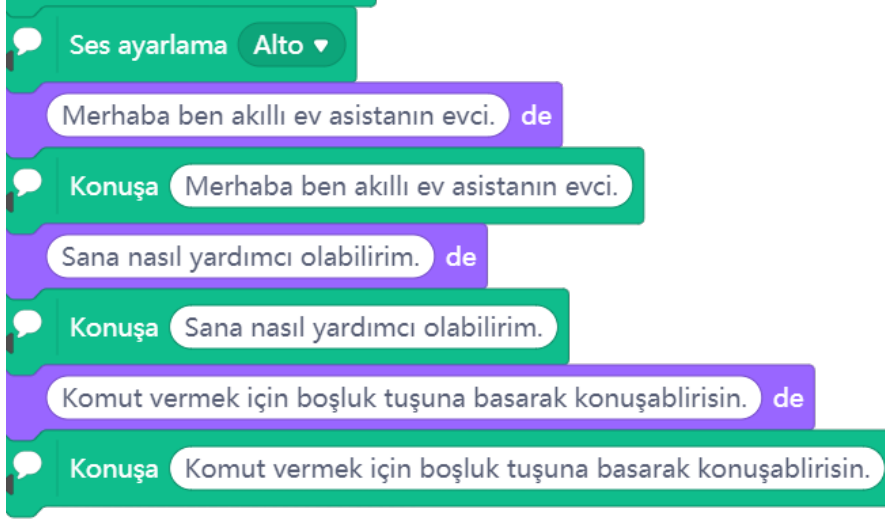
Kukla Adı	2. Kostüm	İşlevi
 Radio1	-	Şarkı oynatmak için kullanılır.
 Bulb	Işık yanma efekti	Işık açıldığında veya kapandığında kostüm değişir.
 C-mbot	-	Evcil-Ev asistanı kullanıcıyla iletişim kurar ve kullanıcının komutlarını yerine getirir.
 Fan blade	Dönme efekti var.	Fan çalıştırıldığında kostüm değişir.

C-mBot Kuklasının Kodları

Kullanıcıyla iletişim kurulması (asistanın kodlanması) için C-mbot kuklasının kodlanması. Yeşil bayrak tıkladığında çalışacak kodlar (Görsel 3.72-a-b)

**Görsel 3.72-a:** C-mBot kuklası yeşil bayrak tıkladığında çalışacak kodlar

Akıllı Ev



Görsel 3.72: C-mBot kuklası yeşil bayrak tıkladığında çalışacak kodlar
Programda kullanılan değişkenler ve görevleri Tablo 3.20’de gösterilmiştir.

Tablo 3.20: Değişkenler

Değişken	Tanım
komut	Sesli komut sisteminde kullanıcının konuşmasının tanınarak metne çevrilmiş hâli.
oda1	Yatak odasındaki ışığın durumunu göstermektedir. Değişken 1 ise ışık yanıyor, 0 ise yanmıyor demektir.
oda2	Salon odasındaki ışığın durumunu göstermektedir. Değişken 1 ise ışık yanıyor, 0 ise yanmıyor demektir.
oda3	Mutfak odasındaki ışığın durumunu göstermektedir. Değişken 1 ise ışık yanıyor, 0 ise yanmıyor demektir.
radio	Şarkı oynatma durumunu gösteren değişkendir.

Kodların Açıklaması

1. Değişken tanımları.
2. Dil ayarları.
3. Kullanıcıya prototip hakkında bilgi verme.
4. Kullanıcının verdiği sesli komutları tanıma ve buna göre iletiler göndermek için C-mbot kuklası kodlanır.

Boşluk tuşuna basınca çalışacak kodlar (Görsel 3.73)

Akıllı Ev



Görsel 3.73: C-mBot kuklası boşluk tuşuna basınca çalışacak kodlar

Boşluk tuşuna basıldığında Türkçe için ses tanıma penceresi açılarak iki saniye boyunca kullanıcının verdiği sesli komut alınır ve ses tanıma ile metne çevrilir. Kullanıcının konuşması metin olarak komut değişkeninde tutulur. Kullanıcının verdiği sesli komut içinde belirli anahtar sözcükler geçerse ileti yayımlanarak ilgili kuklaların veya aygıtın çalışması sağlanır. Anahtar sözcükler, iletiler, kukla veya aygıt bilgisi tabloda verilmiştir (Tablo 3.21).

Tablo 3.21: İletiler

Türü	Anahtar Sözcükler	İleti	Açıklama
Kukla (bulb) ve aygıt	Yatak	oda1	Sarı LED'i yakmak ve kapatmak için ileti.
Aygıt	Salon	oda2	Yeşil LED'i yakmak ve kapatmak için ileti.

Akıllı Ev

Türü	Anahtar Sözcükler	İleti	Açıklama
Aygıt	Mutfak	oda3	Kırmızı LED'i yakmak ve kapatmak için ileti.
Kukla (c-mbot)	Temizlik	temizlik	Mbot kuklasını sahnede temizlik yapması için ileti.
Kukla (fan blade)	Fan	fan	Fan kuklasının kostümünü değiştirerek çalışıyor görüntüsü vermek için ileti.
Radio	Şarkı	radyo	Bir şarkı çalar.
Aygıt (servo motor)	Kapıyı aç	kapi_ac	Servo motoru 180 dereceye döndürmek için ileti.
Aygıt (servo motor)	Kapıyı kapat	kapi_kapat	Servo motoru 0 dereceye döndürmek için ileti.
Aygıt (tüm ledler)	Işıkları kapat	isiklar_kapat	Tüm LED'leri kapatmak için ileti.

DİKKAT!

İleti adları değişken adlandırma kurallarına uygun olmalıdır (Türkçe karakter, boşluk, özel karakter vb. içermemelidir.).

temizlik iletisi alındığında

C-mbot temizlik iletisi alındığında sahnede hareket ederek temizlik yapmaya başlar (Görsel 3.74).

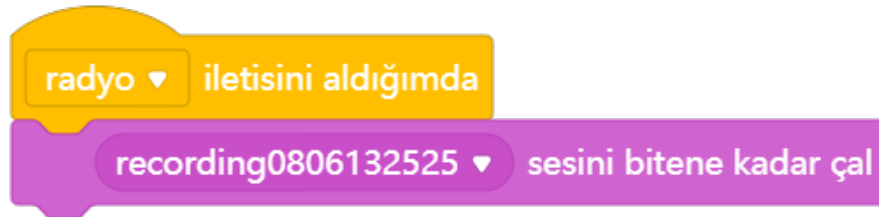


Görsel 3.74: Temizlik iletisi

Diğer Kuklaların Kodları

Radio kuklasının kodları

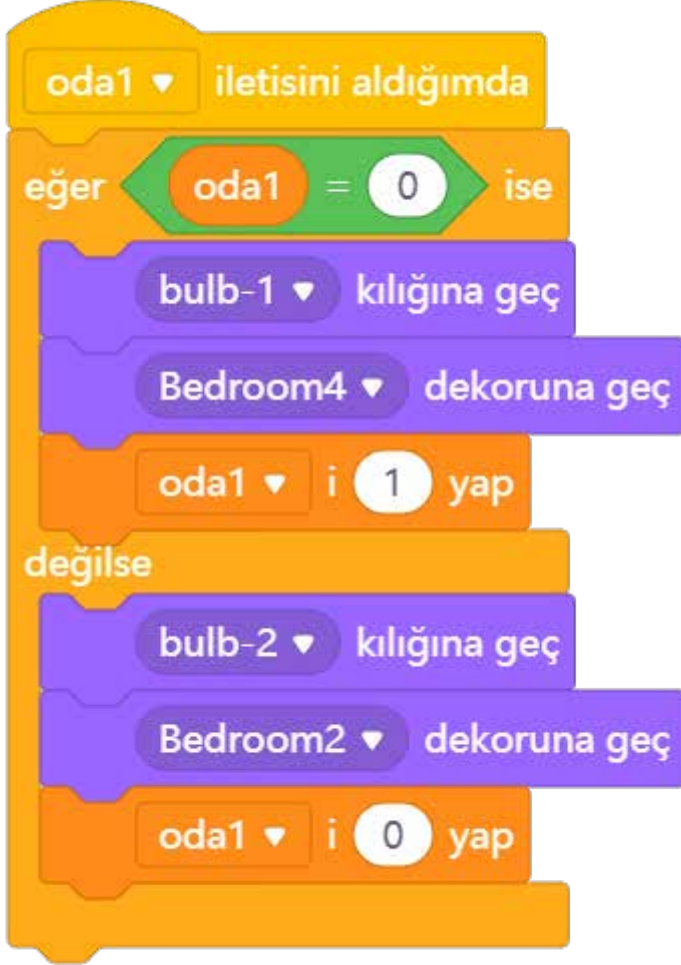
Radyo iletisi alındığında kayıttaki bir şarkı çalınır (Görsel 3.75).



Görsel 3.75: Radyo iletisi alındığında çalışacak kodlar

Bulb kuklasının kodları

Oda1 iletisi alındığında eğer odanın ışığı yanıyor ise kapatılır, kapalıysa da açılır. oda1 değişkenine yeni duruma uygun değer atanır (Görsel 3.76).



Görsel 3.77: fan iletisinde çalışacak kodlar

Aygıt Kodları

Arduino üzerinden yapılan kontrollere ilişkin kodlar bölümler hâlinde sunulmuştur.

Oda1 (yatak odası) ışık kontrolü

Arduino 8 numaralı pine bağlı olan sarı LED'in (oda 1 aydınlatması) kontrolü yapılır. İleti alındığında LED yanıyor ise kapatılır, kapalıysa da yanar. oda1 değişkenine yeni duruma uygun değer atanır (Görsel 3.78).



Görsel 3.78: Oda1 iletisi sarı LED kontrolü

Oda2 (salon) ışık kontrolü

Arduino 9 numaralı pine bağlı olan yeşil LED'in (oda 2 aydınlatması) kontrolü yapılır. İleti alındığında LED yanıyor ise kapatılır, kapalıysa da yanar. oda2 değişkenine yeni duruma uygun değer atanır (Görsel 3.79).



Görsel 3.79: Oda2 iletisi yeşil LED kontrolü

Oda3 (mutfak) ışık kontrolü

Arduino 10 numaralı pine bağlı olan kırmızı LED'in (oda 3 aydınlatması) kontrolü yapılır. İleti alındığında LED yanıyor ise kapatılır kapalıysa da yanar. oda3 değişkenine yeni duruma uygun değer atanır (Görsel 3.80).



Görsel 3.80: Oda3 iletisi kırmızı LED kontrolü

Tüm ışıkları kapatma

Bu kod, ileti alındığında tüm LED'leri kapatır ve LED'lerin durumlarını tutan değişkenlerin değerini de 0 yapar (Görsel 3.81).

```

ileti alınıyor isiklar_kapat
sayısal giriş ayarla 8 çıkış düşük
i 0 yap
sayısal giriş ayarla 9 çıkış düşük
i 0 yap
sayısal giriş ayarla 10 çıkış düşük
i 0 yap

```

Görsel 3.81: isiklar_kapat iletisi alındığında çalışacak kodlar

Kapıyı açma

Bu kod, ileti alındığında tüm servoyu 180 dereceye döndürerek kapının açılmasını sağlar (Görsel 3.82).

```

ileti alınıyor kapi_ac
1 defa tekrarla
servo pin 6 açı 180

```

Görsel 3.82: kapi_ac iletisi alındığında çalışacak kodlar

Kapıyı kapatma

Bu kod, ileti alındığında tüm servoyu 0 dereceye döndürerek kapının kapanmasını sağlar (Görsel 3.83).



Görsel 3.83: kapi_kapat iletisi alındığında çalışacak kodlar

DİKKAT!

Servo motoru 1 kere döndürmek için döngü kullanılmasının nedeni motorun titremesini önlemektir.

Çözümü Test Etme ve Değerlendirme

Yeşil bayrak tıklanarak kodlar çalıştırılır. Asistan kullanıcıyla iletişime geçer. Kullanıcı boşluk tuşuna basılarak sesli komutlar verilir. Sesli komutlar içerdiği anahtar kelimelere göre iletiler gönderir ve akıllı evdeki cihazları kontrol eder. Uygulamayı test etmek için uygulama test kontrol listesi kullanılabilir (Tablo 3.7).

Akıllı Ev

Tablo 3.22: Uygulama Test Kontrol Listesi

SN	Örnek cümle	Beklenen Sonuç	Kontrol (+/-)
1	"Yatak odasını aç / kapat." "yatak"	Sarı LED yanıyorsa kapanır. Kapalıysa yanar. Ayrıca sahne lamba kostümü ve dekor değişir.	
2	"Salonu aç / kapat." "salon"	Yeşil LED yanıyorsa kapanır. Kapalıysa yanar.	
3	"Mutfağı aç / kapat" "Mutfak"	Kırmızı LED yanıyorsa kapanır. Kapalıysa yanar.	
4	"Kapıyı aç"	Servo motor 180 dereceye döner.	
5	Kapıyı kapat"	Servo motor 0 dereceye döner.	
6	"Işıkları kapat"	Tüm LED'ler söner.	
7	-	Değişken değerleri doğru atanmış.	
8	"Şarkı çal"	Bir şarkı çalar.	
9	"Temizlik yap"	Robot sahnede dolaşmaya (temizlik yapmaya) başlar. Bir süre dolaşır.	
10	"Fan aç / kapa"	Fan açıksa kapanır, kapalıysa açılır.	

Çözümün Sunulması

Bu aşamada öğrenciler takım arkadaşlarıyla birlikte oluşturdukları ürünleri diğer takımlara sunar. Öğretmen, sunum sırasında süreç değerlendirme formunu kullanarak öğrencilerin tasarımlarını ve uygulama sürecini değerlendirir. Öğretmen Değerlendirme Ölçeği Ek-1'de verilmiştir.

Yeniden Tasarlama / Revize Etme

Öğrencilerin aldıkları değerlendirmeler ve geri bildirimler doğrultusunda çözümlerinde gerekli iyileştirmeleri yaptıkları aşamadır. Öğrencilerin bu aşamada problemin kriter ve sınırlılıkları çerçevesinde tasarımlarını en iyi olabilecek şekilde iyileştirmeleri beklenir. Burada öğrencilerin oluşturmuş olduğu tasarım üründe iyileştirmenin nasıl yapılacağını, neden bu iyileştirmenin yapıldığını ve proje etkisini özetleyen takım hâlinde hazırlanmış bir rapor oluşturmaları istenebilir.

Kararın Tamamlanması

Kararların tamamlanmasında ise öğrencinin nihai tasarımının en iyi olup olmadığına kendisinin karar vermesi amacıyla Ek-2'de verilen öğrenci değerlendirme ölçeği kullanılacaktır.

Öğrenciler grup üyeleriyle birlikte aşağıda yer alan soruları cevaplandırarak proje değerlendirmesi yaparlar.

1. Bu etkinliği yaparken en çok nerede zorluk yaşadınız? Bu zorlukları nasıl aştınız?
2. Bu etkinliği yaparken en çok sevdiğiniz kısım hangisiydi?
3. Bu etkinlikten neler öğrendiniz?
4. Bu projeyi yeniden planladığında geliştirmek istediğin yönler nelerdir?
5. Elektrik enerjisinin tasarruflu kullanımı için çevremizi nasıl bilinçlendirebilirsiniz?

Akıllı Ev

Kaynakça

- [1] Hynes, M., Portsmore, M., Dare, E., Milto, E., Rogers, C., Hammer, D. & Carberry, A. (2011). Infusing engineering design into high school STEM courses. <http://ncete.org/flash/pdfs/Infusing%20Engineering%20Hynes>. Erişim tarihi: 10.02.2017.
- [2] Brunsell, E. (2012). The engineering desing process. Brunsell, E. (Ed.) Integrating engineering + science in your classroom. Arlington, Virginia: National Science Teacher Association[NSTA] Press.

EKLER

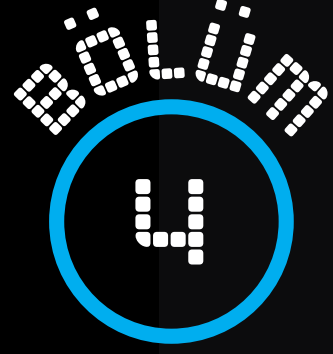
EK-1: Öğretmen Değerlendirme Ölçeği

Değerlendirme Kriterleri	GELİŞTİRİLMELİ	İYİ	ÇOK İYİ
Sorunu tanımlama ve analiz etme			
Olası çözümleri bulma ve en iyisini seçme			
Tasarım ürününün kabataslak çizimi veya simüle edilmiş formu			
Grup çalışması ve iş birliği			
Ürünü sunma			

EK-2: Öğrenci Öz Değerlendirme Ölçeği

Değerlendirme Kriterleri	Katılmıyorum	Kararsızım	Katılıyorum
Ev otomasyon sistemlerinde kullanılan yapay zekâ teknolojileri hakkında bilgi sahibi oldum.			
Bilgi temelli hayat problemine yönelik bulduğum çözüm en uygun çözümdür.			
Senaryo sürecinde takım arkadaşlarımla iş birliği içinde bulunarak takım arkadaşlarıma katkı sağladım.			
Ürünü tasarlarken kriter ve sınırlılıkları dikkate aldım.			
Senaryo sürecini anlatan bir rapor oluşturdum.			





YAPAY ÖĞRENME, ALGORİTMALARI ve UYGULAMALARI

Yazar
Dr. Sultan UÇAR ALTUN



• Yapay zekâ, yapay öğrenme ve derin öğrenme

Yapay öğrenme yapay zekânın veriden öğrenen bir alt kümesidir. Yapay öğrenme terimi yerine Türkçede yaygın olarak "makine öğrenmesi" kullanılmaktadır. Yapay zekânın yapay öğrenmeden başka alt dalları olduğu gibi yapay öğrenmenin de alt kümeleri bulunmaktadır (Görsel-4.1). Bunlardan en popülerleri derin öğrenmedir. Veriden öğrenme özelliğinden dolayı günümüzde yapay öğrenme ve onun bir alt kümesi olan derin öğrenme (deep learning) yapay zekânın en güçlü kollarıdır. Günümüzde yapay zekâ adına ortaya konan sanal asistanlardan otonom araçlara tüm uygulamalar yapay öğrenme ve derin öğrenme tabanlıdır. Bu nedenle kitabın bu bölümünde de yapay öğrenme konusu ele alınmıştır.



Görsel-4.1: Yapay zekâ, yapay öğrenme ve derin öğrenme

• Programlama x Yapay Öğrenme

Klasik programlamada program girdileri yazılan kodlara göre işleyerek çıktılar üretir. Farklı girdiler için farklı çıktılar üretilmekle birlikte sabit bir kod vardır. Kod içinde belirli şartlara ve kurallara göre program akışı değişir ve farklı çıktılar üretilebilir. Klasik programlamada programcının tüm olası durumlar için gerekli kodları yazması beklenir. Aksi takdirde bu durum için uygun bir çıktı üretilemez. Yapay öğrenmede ise basitçe anlatmak gerekirse girdiler ve çıktılar verilir yapay öğrenme algoritmalarının girdiler ve çıktılar arasındaki bağlantıları, kuralları, örüntüleri veya formülleri bulması, ortaya çıkarması "öğrenmesi" hedeflenir.

• Problem durumları

Yapay öğrenme uygulandığı problem durumuna göre tahmin edici ve tanımlayıcı modeller olmak üzere ikiye ayrılır. Tahmin edici modeller bir etiketi tahmin etmek için kullanılır. Bu etiket sayısal bir değer ya da sınıf adı olabilir. Tanımlayıcı modellerde amaç bir sonucu tahmin etmek değil veriler arasındaki ilişkileri ortaya koyarak onları gruplara ayırmak, birlikteliklerini veya zamana göre değişimlerini analiz etmektir. Farklı problem durumları farklı öğrenme modelleri ve farklı veri türleri gerektirir.

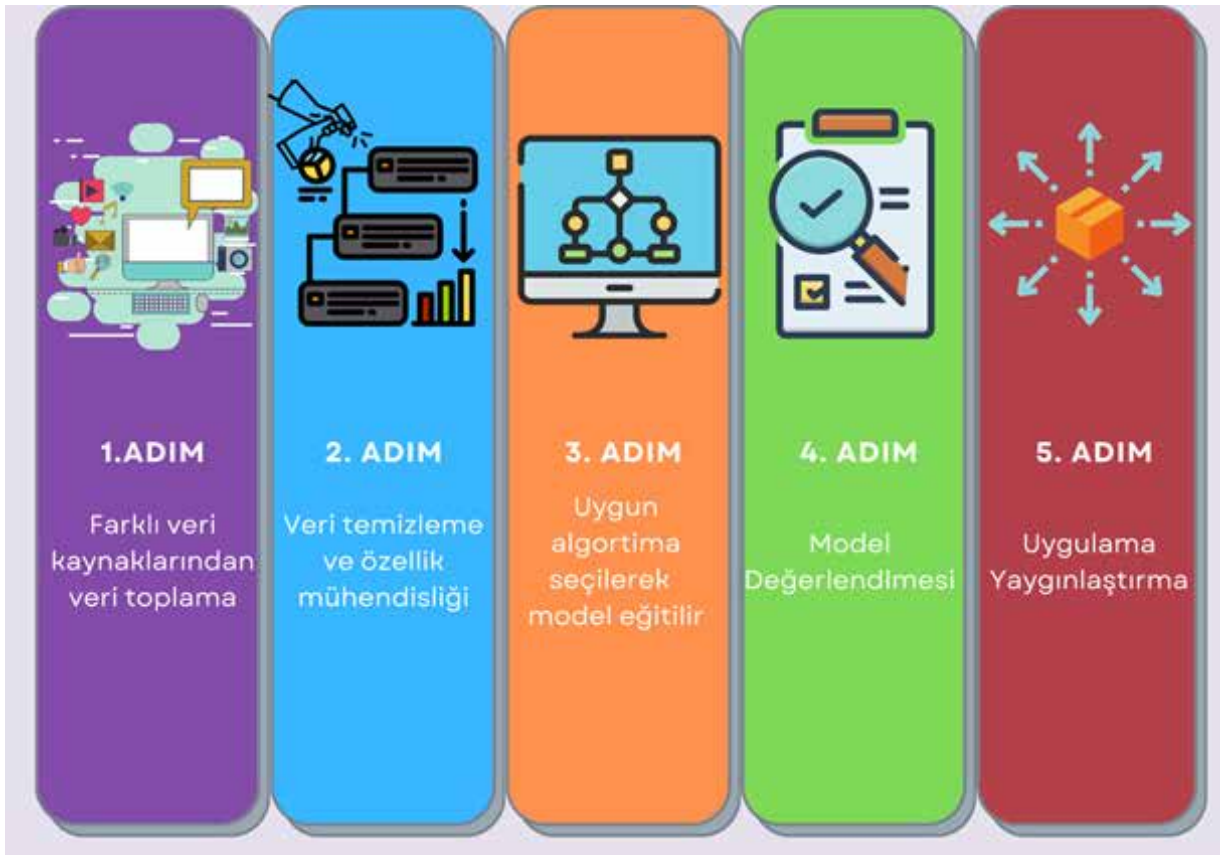


Okuma Parçası

Cahit Öğretmen sınıfta kare köklü ifadeler konusunu anlatmaktadır. Öncelikle konuya ilişkin temel bilgileri veren Cahit Öğretmen kare köklü ifadelerle ilgili örnekleri öğrencilere anlatarak, örnek soruları onlara göstererek çözer. Konuyla ilgili yeteri kadar örnek gösterdiğini ve konunun öğrenciler tarafından anlaşıldığını (öğrenildiğini) düşünen Cahit Öğretmen öğrencilerin konuyu anlayıp anlamadıkları test etmek için öğrencilere önceki sorularla benzer (ama aynı olmayan) sorulardan oluşan çoktan seçmeli ve doğru yanlış sorularının olduğu bir sınav (test) uyguluyor. Öğrencilerin yanıtlarını cevap anahtarı ile kontrol ederek başarı durumlarını bir puan olarak onlara söylüyor. Öğrencilerin başarı durumlarının belirli bir ölçüde birbirinden farklı olması beklenir. Belirli bir başarı puanının altında kalan öğrencilerin konuyu iyi anlamadıkları "öğrenemedikleri" söylenebilir. Belki bu öğrenciler için daha fazla örnek soru çözmek gerekecektir.

• Öğrenme ve Yapay Öğrenme

Doğal öğrenmede olduğu gibi yapay öğrenmede de çeşitli öğrenme biçimleri bulunmaktadır. Cahit Öğretmen örneği öğrenmenin tüm yönlerini açıklamaktan uzak olmakla birlikte yapay öğrenme ile ilgili kavramları açıklamak için kullanışlı bir metafordur. Bu metafor üzerinden yapay öğrenmenin adımları (Şekil 4.2.) açıklanmıştır.



Görsel 4.2: Yapay Öğrenme sürecinin adımları

1. Adım: Farklı veri kaynaklarından veri toplama

Yapay öğrenmenin temelini anlayabilmek için en çok kullanılan yapay öğrenme modellerinden biri olan denetimli/gözetimli öğrenme üzerinden konu ele alınmıştır. Örnekte Cahit Öğretmen'in rehberliğinde (denetim/gözetim) bir öğrenme gerçekleşmektedir. Burada kullanılan denetim/gözetim kavramı problem durumundaki girdilerle birlikte problemi çözmek için gerekli işlemleri yaptıktan sonra yanıtın öğrencilere gösterildiği örneklerin verilmesi anlamına gelmektedir.

Öğrencilerin Cahit Öğretmen'in gösterdiği örneklerden öğrendiği gibi yapay öğrenme sürecinde de algoritmalar veri setleri kullanılarak eğitilir. Yapay öğrenme süreci için veriyi bir aracın yakıtına benzetebiliriz. Bir aracın iyi çalışabilmesi için yakıtın yeterli miktarda ve kaliteli olması ne kadar önemli ise yapay öğrenme için de verinin miktarı ve niteliği o kadar önemlidir. Hemen her durumda olduğu gibi yapay öğrenme süreci de bir problem durumuyla başlar. Problem durumunun gerektirdiği verinin farklı veri kaynaklarından alınması yapay öğrenme sürecinin ilk adımındır. Bu aşama Cahit Öğretmen'in derste göstereceği örnekleri ve sınavda soracağı sorular için çeşitli kaynakları tarayarak veri toplamasına benzetilebilir.

2. Adım: Veri temizleme ve özellik mühendisliği

Cahit Öğretmen topladığı örneklerden seçme, eleme, düzenleme yaparak bir örnek/soru havuzu oluşturmuştur. Bu soru havuzundaki bazı soruları derste çözmeye bazılarını ise sınavda sormaya karar vermiştir. Cahit Öğretmen'in öğrencilere çözümlenerek göstereceği örnekler yapay öğrenme sürecinde eğitim verisi (training data) olarak adlandırılmaktadır. Eğitim verisi bir tablo olarak (sütunlar ve satırlar) düşünüldüğünde sütunlardan bazıları girdileri (bağımsız değişken) biri de çıktıyı (bağımlı değişken) temsil etmektedir. Veri setindeki satırlar ise örnekleri temsil etmektedir. Veri setleri her zaman olduğu gibi kullanılamaz. İnsan hatalarından veya teknolojik arızalardan dolayı veri setleri içinde tutarsız ve imkânsız veri bulunabilir. Bazı özelliklere ait eksik/kayıp değer/veri (missing value) bulunabilir. Veri setindeki tüm sütunların/özelliklerin kullanılması gerekmeyebilir. Bu durumda bazı özelliklerin seçilmesi gerekebilir. Bazı özellikler üzerinde veri dönüşümü yapmak gerekebilir. Veri setinin eğitim ve test seti olarak bölünmesi de bu aşamada gerçekleştirilir. Tüm bu işlemler yapay öğrenmenin ikinci adımında ele alınır.

3. Adım: Uygun algoritma seçilerek model eğitilir

Fen Bilimleri dersindeki deneylerden hatırlanabileceği üzere bağımsız değişkenlerin bağımlı değişken üzerinde etkisi olduğu düşünülmektedir. Yapay öğrenmede de girdilerin çıktıları etkilediği varsayılır ve girdi-çıktı arasındaki ilişkiyi bulmak hedeflenir. Cahit Öğretmen'in sınıfta verdiği örneklerdeki sayılar girdileri/bağımsız değişkenleri ifade ederken işlemin sonucu da çıktıyı/bağımlı değişkeni ifade etmektedir. Yapay öğrenmede Cahit Öğretmen'in sınıfta yaptığı gibi veri setinde girdi-çıktı örnekleri verilerek uygun bir algoritma seçilir ve eğitilerek bir model oluşturulur. Model her yeni örnekte tıpkı bir öğrenci gibi kendini geliştirir ve girdiler ve çıktılar arasındaki ilişkiyi daha iyi açıklayabilecek şekilde parametrelerin ağırlıklarını günceller. Her öğrencinin farklı öğrenme yöntemi, farklı soru çözme stilleri olduğu gibi yapay öğrenmede de çeşitli algoritmalar kullanılmakta ve farklı modeller geliştirilmektedir. Algoritmalar örnekteki öğrencilerin düşünme biçimi, öğrenme yöntemi veya işlem becerileri olarak düşünülebilir. Nasıl her öğrencinin her derste aynı derecede başarılı olması beklenmezse her algoritmada her problem durumunda, her veri setinde aynı derecede başarılı olmaz. Hatta problem durumu ve veri türü bazı algoritmaların kullanılmasına izin verirken bazılarının kullanılmasına izin vermez. Böyle bir durumda ikinci adıma dönülerek veri setinde bazı dönüştürme işlemleri yapılabilir.

4. Adım: Modelin değerlendirilmesi

Yapay öğrenmede oluşturulan modelin başarısını test etmek için Cahit Öğretmen'in öğrencilere yaptığı gibi bir sınav yapılır. Modelin başarısını test etmek için eğitim sırasında

kullanılmayan verilerin (örneklerin) olduğu bir veri seti kullanılmalıdır. Bu durum Cahit Öğretmen'in derste çözdüğü örneklerin birebir aynısını sınavda sormasına benzer. Bu durumda öğrencilerin öğrenmesinden daha çok ezberlemesinden söz edilebilir. Bu tür durumlarda öğrencilerin veya yapay öğrenme modelinin başarısı sağlıklı bir şekilde değerlendirilmez. Modeli test etmek için kullanılan veri setine test verisi (test data) denir. Test datasında sonuç/çıktı sütunu verilmez. Bu durum Cahit Öğretmen'in doğru yanıtların işaretlenmiş olduğu sınav kağıtlarıyla sınav yapmasına benzer. Cahit Öğretmen sınavda derste çözdüğü örneklerle hiç benzemeyen sorular sorarsa öğrencilerin başarıları düşebilir. Tüm bu durumlar yapay öğrenme sürecinde de geçerlidir. Öğrencilerin sınavda soruları çözerken yanıtları verdiği gibi yapay öğrenme modeli de test verisindeki girdileri kullanarak sonuçları bulmaya çalışır. Öğrencilerin değerlendirilmesiyle benzer şekilde modelin bulunduğu sonuçlarla doğru yanıtlar karşılaştırılarak modelin performansını gösteren başarı puanı hesaplanır. Yapay öğrenmede çıktı her zaman sayısal bir değişken olmaz. Her ne kadar sayısal hesaplamalar yapılsa da sonuç kısmında bu sayılar bir kategoriyle eşleştirilebilir. Verilen kan değerlerine göre bir hastalığa ilişkin tahmin üreten (pozitif/negatif) bir modelin çıktısı ikili değerden birini alır. Yapay öğrenmede sonuç, bağımlı değişken sütununa etiket adı verilir. Etiket bir kategorinin/sınıfın adı olabileceği gibi sayısal bir değer de olabilir.

5. Adım: Uygulama ve yaygınlaştırma

Değerlendirme aşamasından sonra geliştirilen model kullanım amacına göre bir bilgisayar programı olarak veya gömülü bir yazılım olarak paketlenir. Akıllı cihazlardaki uygulamalar uygulama ve yaygınlaştırma adımıyla örnek olarak verilebilir. Bu aşamada geliştirilen model bir ara yüz tasarlanarak hitap ettiği kişi ve kurumların kullanımına hazır hale getirilir.

• Yapay öğrenme modelleri

Yapay öğrenme için çeşitli sınıflandırma yöntemleri bulunmaktadır. Bu başlıkta öğrenme durumlarına göre yapay öğrenme modelleri ele alınmıştır. Öğrenme türlerine göre dört gruba ayrılır (Görsel-4.3).



Denetimli öğrenme modelleri yapay öğrenme modelleri dolayısıyla yapay zekâ içinde en çok kullanılan modellerdir. Denetimli öğrenme modelleri için etiketli veri setleri gerekmektedir. Yapay zekâ uygulamalarında etiketli veriler etiketsiz verilerden çok daha yararlıdır. Google gibi dev şirketler CAPTCHA gibi sistemlerle verinin internet kullanıcıları tarafından etiketlenmesini sağlamaktadır. Özellikle trafik işaretleri, yollar, köprüler, araçlar vb. gibi görselleri kullanıcılara sorarak otonom araçlar ve diğer yapay zekâ teknolojileri için etiketli veri setleri oluşturuyorlar. Ülkemizde Savunma Sanayi Bakanlığı tarafından başlatılan Veri Kovanı Projesi (<https://verikovani.ssb.gov.tr/>) etiketli veri seti oluşturmak ve bunları yerli yapay zekâ teknolojilerinde kullanmak için hayata geçirilmiştir. Sistem veri etiketleyerek gelir elde etme olanağı da sunmaktadır.

MAKİNE ÖĞRENMESİ

Denetimli Öğrenme (Etiket/Sonuç)

1. Sınıflandırma (Çıktı: Kategorik değişken)

Destek Vektör Makineleri
Diskriminant Analizi
Lojistik Regresyon
Karar Ağaçları (Decision Tree)
Sinir Ağları

Örnek 1: Bir çiçeğin türünü belirlemek (iris veri seti)
Örnek 2: İstenmeyen eposta sınıflandırılması
Örnek 3: Bir görseldeki nesneyi tanımlamak

2. Regresyon (Çıktı: Sürekli, sayısal değişken)

Lineer Regresyon
Destek Vektör Regresyon
Topluluk (Ensemble) Öğrenmesi
Karar ağaçları (Decision Tree)
Sinir Ağları

Örnek 1: İkinci el araç fiyat tahmini
Örnek 2: Hava sıcaklığı tahmini

Denetimsiz Öğrenme (Etiket/Sonuç yok)

1. Birliklilik (Association) (Girdiler: Kategorik değişkenler)

Apriori
FP-Growth

Örnek 1: Market sepeti analizi

2. Kümeleme (Clustering) (Girdiler: Sürekli, sayısal değişken)

K-Means, K-Medoids, Fuzzy
Hiyerarşik
Sinir Ağları

Örnek 1: Müşteri segmentasyonu

3. Boyut indirgeme (Dimension Reduction) (Girdiler: Sürekli, sayısal değişken)

PCA (Temel bileşenler analizi)
Bağımsız bileşen analizi
Faktör analizi

Örnek 1: Metin işleme
Örnek 2: Büyük veri görselleştirme
Örnek 3: Yüz tanıma

Yarı Denetimli Öğrenme (Az sayıda Etiket veri/çok sayıda etiketsiz)

Kendi kendine eğitim (Self Training)
Üretken (Generative) modeller
Grafik tabanlı (graph based)
Çoklu görünüm algoritmaları

Örnek 1: Web sayfalarını sıralama
Örnek 2: Genetik çalışmalarını, genleri bulma

Pekiştirmeli Öğrenme (Veri Seti Yok)

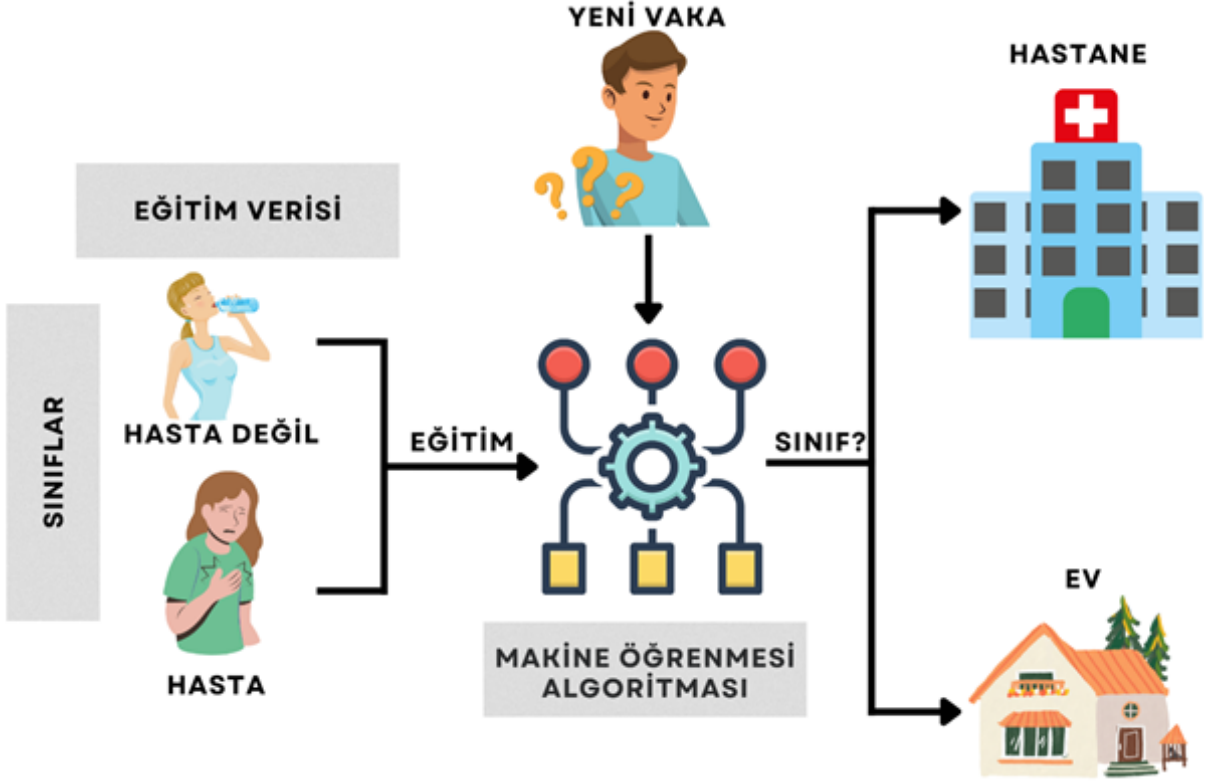
Q- Öğrenme
R- Öğrenme
TD (Zamansal Fark) Öğrenme

Örnek 1: Robotlar ve otonom araçlar için yön bulma

Görsel 4.3: Yapay öğrenme türleri

1. Denetimli/gözetimli öğrenme (supervised learning)

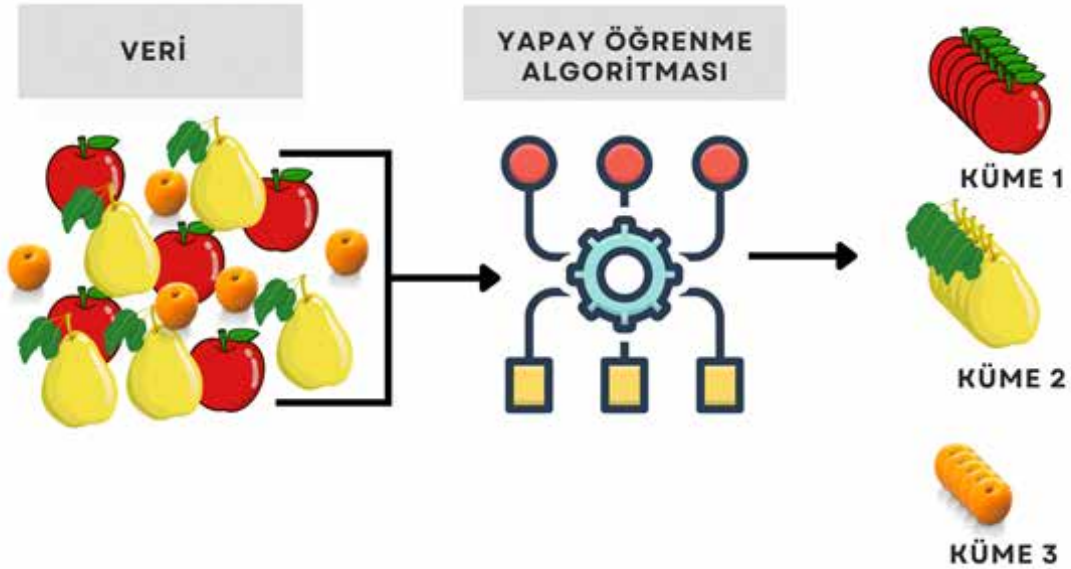
Yapay öğrenme modelinde etiketli veri seti kullanılırsa ve modelin eğitimi sırasında etiketler modele öğretilirse bu öğrenme türü denetimli öğrenme olarak adlandırılır. Örnekte (Görsel 4.4) COVID-19 hastalığına ilişkin bir yapay öğrenme modeli oluşturulmak istenmektedir. Hasta ve hasta olmayan kişilere ait örneklerin bulunduğu bir veri seti kullanılarak bir model eğitilmekte ve bu modele yeni örnekler verildiğinde o örneklerin hangi sınıfa (hasta/hasta değil) ait olduğunu tahmin etmektedir.



Görsel 4.4: Denetimli öğrenme süreci

2. Denetimsiz/gözetimsiz öğrenme (unsupervised learning)

Denetimli öğrenme modelleri çok güçlü olmakla birlikte gücünü etiketli veriden almaktadır. Çoğu durumda etiketli veriye ulaşmak ya da veriyi etiketlemek oldukça zahmetli ve maliyetli olabilmektedir ve genelde etiketsiz veri miktarı etiketli veriden çok daha büyüktür. Yapay öğrenme modelinde **etiketsiz** veri seti kullanılırsa bu öğrenme türü denetimsiz öğrenme olarak adlandırılır. Denetimsiz öğrenmede örnekler bir etikete sahip değildir. Örnekte (Şekil 4.5) meyvelerin özelliklere göre kümelere ayrılması gösterilmiştir. Meyvelerin adı (etiketi) belirtilmemiştir. Denetimsiz yapay öğrenme modeli meyveleri özelliklerine göre ayırarak onları üç kümeye ayırmıştır. Örnekte kullanılan kümeleme algoritması birbirine benzeyen meyveleri aynı diğerlerini farklı kümelere ayırmaktadır. Kümeleme işlemi sonunda ortaya çıkan kümelere bir isim verilirse bu kümeler artık birer sınıf küme isimleri de onların etiketleri olur.



Görsel 4.5: Denetimsiz öğrenme süreci



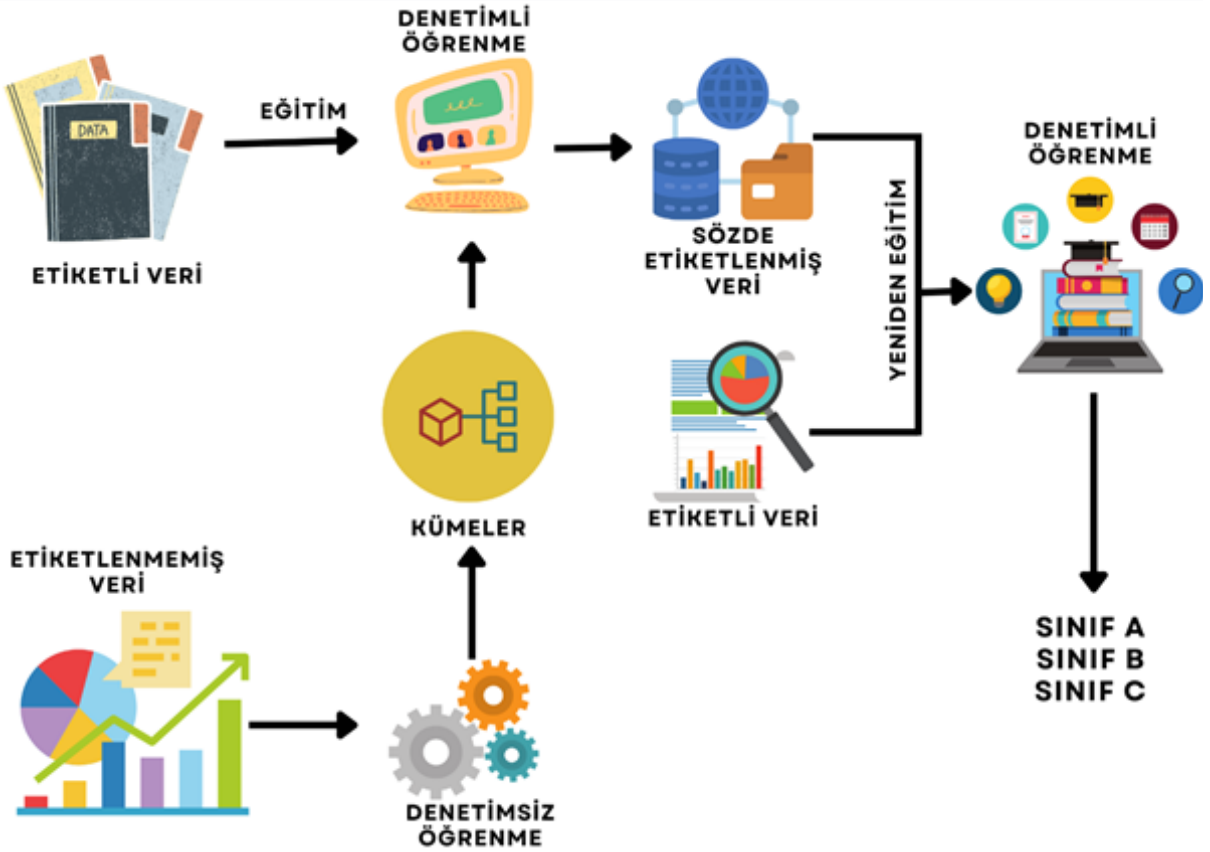
Netflix 190'dan fazla ülkede 160 milyondan fazla üyesi bulunan bir internet televizyonu platformudur. Şirketin gelirleri üyelerinin aboneliklerini sürdürmesine bağlı olduğu için üyelerinin memnuniyetini hep ön planda tutmaktadır. Şirket gelirlerini artırmak için sisteminde çok sayıda yapay



öğrenme modeli kullanmaktadır. Platformun öneri sisteminde kullanılan modellerden biri de benzer beğenilere sahip üyeleri kümelere ayırmak ve bu kümeler içinde yer alan üyelerin izledikleri videoları diğerlerine önermeye dayanan bir denetimsiz öğrenme modelidir.

3. Yarı denetimli öğrenme (semi supervised learning)

Yarı denetimli öğrenme modeli denetimli ve denetimsiz öğrenme modelinin birlikte kullanıldığı modellerdir. Çok miktarda etiketsiz veri ve az miktarda etiketli verinin olduğu problem durumlarında yarı denetimli yapay öğrenme modelleri kullanılır. Bu yöntemle hem etiketli verilerin etiket bilgisinden hem de etiketsiz ama büyük miktardaki veriden yararlanılmış olabilir (Şekil 5.6). Yarı denetimli öğrenme hem denetimli hem de denetimsiz öğrenme modellerinin kullanıldığı problem durumlarında kullanılabilir.



Görsel 4.6: Yarı denetimli öğrenme modeli



Bilgisayarlı görü yarı denetimli öğrenmenin kullanıldığı alanlardan biridir. Bilgisayarlı görü uygulamalarında yeterli miktarda etiketli veri bulmak sorun olmaktadır. Buna karşın çok miktarda etiketsiz görselle erişilebilmektedir. Yarı denetimli öğrenme ile otonom araçlar için trafik işaretlerinin tanınması gibi uygulamalar geliştirmek için etiketsiz veriler de kullanılabilir.

4. Pekiştirmeli öğrenme (reinforcement learning)

Bir veri setinin olmadığı ve ortam hakkında fazla bilgi sahibi olunmadığı durumlarda kullanılan yapay öğrenme modelidir. Deneme yanılma ve ödüllendirme ilkelerine göre çalışmaktadır. Labirentteki bir farenin deneme yanılma yoluyla çıkışı bulması veya peynire

ulaşması bu öğrenme türüne benzetilebilir. Pekiştirmeli öğrenme deneyimlerden öğrenmeyi esas alır. Robotik alanı ve oyunlar pekiştirmeli öğrenmenin en çok kullanıldığı alanların başında gelmektedir.

Yapay öğrenme modelleri oluşturmak için kullanılan araçlar

Orange Data Mining

Orange, açık kaynak makine öğrenmesi ve veri görselleştirme aracıdır. Veri analizi iş



Bir robot süpürgeci deneme yanılma yoluyla temizlik yapacağı ortamın haritasını çıkarması veya karşılaştığı bir engeli aşmaya çalışması pekiştirmeli öğrenmeye örnek verilebilir.

akışlarını görsel olarak oluşturmak için gerekli birçok araç içermektedir. Orange hem yeni başlayanlar hem de uzman veri bilimcilerin kullanabileceği bir veri madenciliği aracıdır. Kullanıcı ara yüzü kodlama yapmadan veri analizi yapmaya olanak verir. Karmaşık veri analitiği iş hatlarının oluşturulmasını basitleştirir.

Piyasada bulunan farklı açık kaynak veri madenciliği araçlarının özellikleri tabloda sunulmuştur.

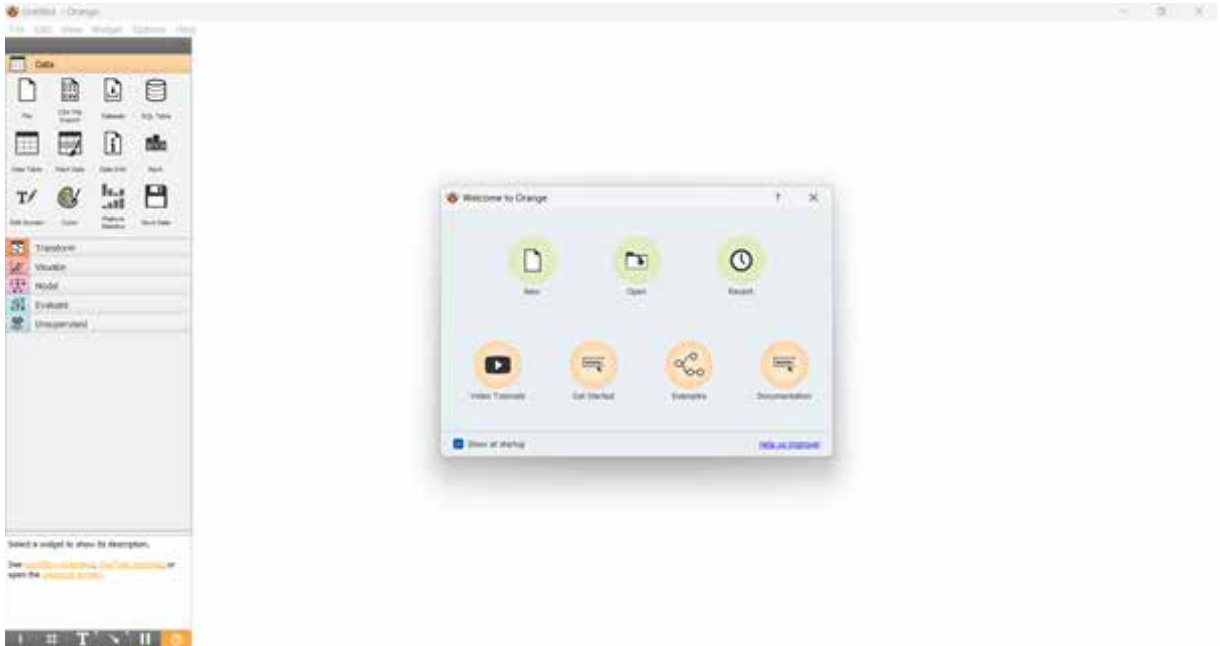
Tablo 4-1: Popüler Veri Madenciliği Araçları

Name	Weka	RapidMiner	KEEL	Orange	Tanagra
Developer	Univ. of Waikato, New Zealand	RapidMiner, Germany	Univ. of Granada, Spain	KNIME.com AG, Switzerland	Lumiere Univ. Lyon 2, France
Programming Lang.	Java	Java	Java	Java	C++
License	Open-source, GNU GPL 3	Open-source (v.5 or lower); Closed source (free started ed. v.6)	Open-source, GNU GPL 3	Open-source, GNU GPL 3	Open-source, GNU GPL 3
Current Version	3.6.10	6	2	2.7	1.4.50
GUI/Command line	Both	GUI	Command line	Both	GUI
Main purpose	General data mining	General data mining	General data mining	General data mining	General data mining

Orange Programı Karşılama Ekranı

Program çalıştırıldığında bir karşılama ekranı gelir. Bu ekrandaki semboller kullanılarak veri analizi çalışmasına başlanabilir.

- New** : Yeni bir iş akışı (workflow) oluşturmak için kullanılmaktadır.
- Open** : Var olan bir iş akışını açmak için kullanılmaktadır.
- Recent** : Son çalışılan iş akışlarını açmak için kullanılır.



Görsel 4.7: Orange programı karşılama ekranı

Video Tutorials: Eğitim videolarına ulaşmak için kullanılır.

Get Started: Orange resmî web sayfası

Examples: Örnek iş akışları

Documentation: Orange kullanımı ve araçları için açıklamalar, belgeler.

"Show at startup" ifadesinin yanındaki onay kaldırılırsa program başlatıldığında karşılama ekranı gelmez.

Orange ana ekranı menü çubuğu, araç seti ve çalışma alanından oluşmaktadır. Menü çubuğunda standart bileşenler (dosya, düzen, görünüm, ayarlar ve yardım) yer almaktadır.

Araç setinde araçlar görevlerine göre gruplandırılmış şekilde yer almaktadır.

Data (Veri) : Veri seti ve veri dosyalarını yükleme, tanımlayıcı istatistikler, veri seti hakkında bilgi almak için kullanılan araçlar bulunmaktadır.

Transform (Dönüştürme): Veri setini bölme, örnekleme, satırları seçme, tablo pivot, benzersiz değerleri bulma, gruplama, veri setlerini birleştirme, yeni sütun ekleme

Visualize: Grafik gibi veri görselleştirme araçları yer almaktadır.

Model: Makine öğrenmesi modellerini oluşturmak, modelleri yüklemek ve kaydetmek için gerekli araçlar yer almaktadır.

Evaluate: Oluşturulan modellerin değerlendirilmesi için kullanılan araçlar.

Unsupervised: Gözetimsiz makine öğrenmesi araçlarının yer aldığı araç setidir.

Araçlar, tuvale bırakarak aralarında bağlantılar oluşturulur. Araçların giriş çıkış kanalları vardır. Veri araç setindeki araçlar çalışma alanına sürüklenerek bir veri seti eklenir.

Orange

Seviye

12 yaş ve üzeri

Konu

1. Orange ile ilk iş akışının (workflow) oluşturulması

Süre

6 ders saati

Amaç

Bu uygulamada Orange programı kullanılarak basit bir iş akışı oluşturulmaktadır. Uygulamada iş akışına bir veri dosyası eklenerek nesnelere (widget) aracılığıyla görüntülenmekte ve görselleştirilmektedir.

Motivasyon

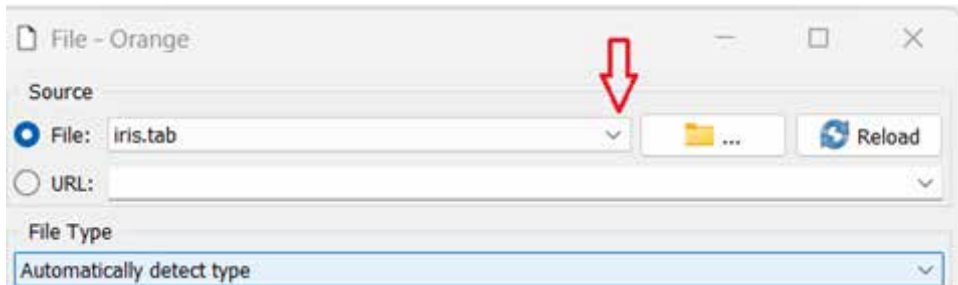
Makine öğrenmesinin ilk adımı veri toplamadır. Toplanan veri farklı veri dosyası türlerinde olabilir. Orange ile oluşturulan iş akışında ilk olarak veri dosyası çalışma alanına eklenir. Data Table nesnesi ile veri dosyası satır ve sütun şeklinde görüntülenebilir. Saçılım grafiği (scatter plot) gibi veri görselleştirme nesnelere kullanılarak veri dosyası incelenebilir ve veri hakkında bir görüş elde edilebilir.

İş akışının oluşturulması

- Data araç setinden File nesnesi (tıklanarak veya sürükleyip bırak yöntemiyle) çalışma alanına eklenir.

NOT: Bir araç çalışma alanına eklemek için başka bir yöntem de çalışma alanına fare ile sağ tıklayarak ve çıkan menüden arama bölümüne araç adını yazarak istenilen listeden aracın seçilmesidir.

- File nesnesi çift tıklanarak listeden program içinde gömülü olarak bulunan veri tabanlarından biri olan iris seçilir.



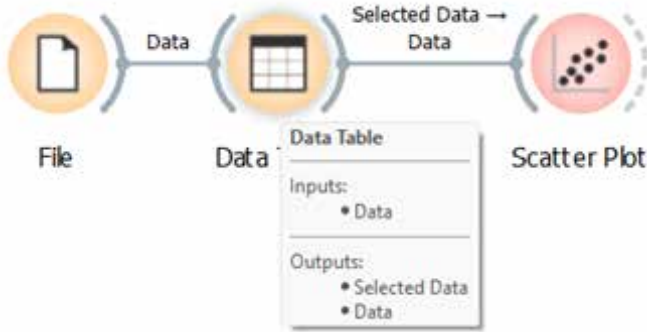
Görsel 4.8: Veri dosyasının seçilmesi

NOT: Iris veri dosyası File nesnesi ile varsayılan olarak seçilmiş olarak gelebilir. Daha önce farklı bir veri dosyası belirtilmişse o seçili olarak gelir.

- Visualize araç setinden Scatter Plot aracı eklenir. File nesnesinden farenin sağ tuşu ile Data Table ve Scatter Plot nesnelere bir bağlantı oluşturulur.

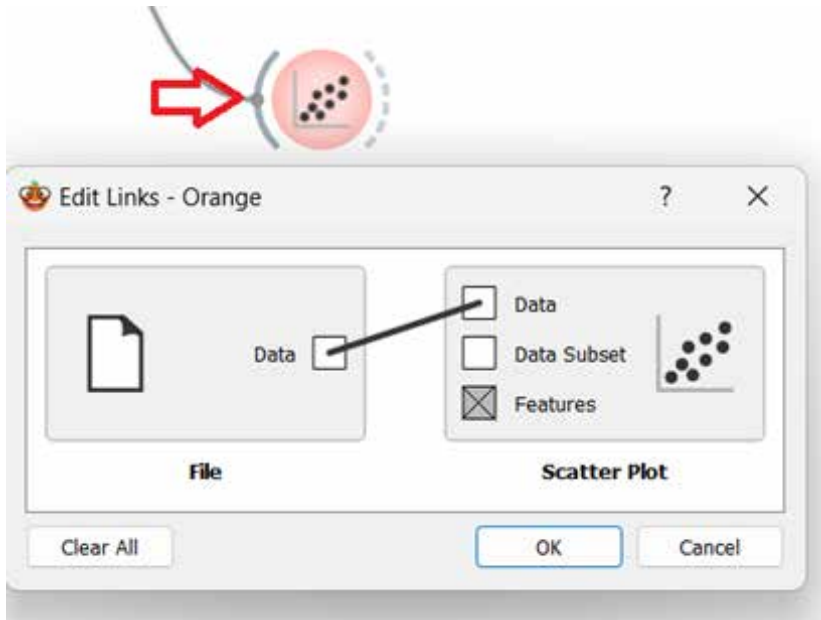
Orange

Bir nesne girdiler (inputs) ve çıktılar (outputs) bileşenlerinden oluşur. Nesnelere işlevlerine göre girdiler üzerinde işlemler yaparak çıktılar üretir. Fare ile nesnenin üzerine gelindiğinde nesnenin girdi türleri ve çıktı türleri görülebilir. Böylece nesnelere arası bağlantılar yapılırken nesnelere çıktılar uygun nesnelere girdi olarak seçilir.



Görsel 4.9: Nesne girdiler ve çıktılar

Bir nesne için uygun birden fazla girdi veya çıktı türünden birini seçmek için bağlantı noktasına fare ile tıklanarak açılan pencereden kutucuk üzerinden (Data) sürükleyip bırakarak bağlantı yapılmak istenen diğer kutucuğa bağlantı sağlanır. Bağlantıyı iptal etmek için fare ile üzerine tıklamak yeterlidir.



Görsel 4.10: Bağlantı oluşturma

NOT: Bir bağlantı üzerine fare ile sağ tıkladığında çıkan menüde remove ve reset signals seçenekleri bulunmaktadır. Remove seçeneğine tıkladığında bağlantı silinir. Reset signals seçeneğine tıkladığında görseldeki bağlantı ayarlarının yapıldığı menü çıkmaktadır.

Orange

Herhangi bir nesneyi çalışma alanından kaldırmak için üzerine sağ tıklanarak çıkan menüden **Remove** seçeneğine tıklanır. Bu menüden nesneyle ilgili diğer işlemler de yapılabilir.



Görsel 4.11: Nesneyle ilgili işlemler

Open : Nesnenin işlevini görüntülemek için kullanılır.

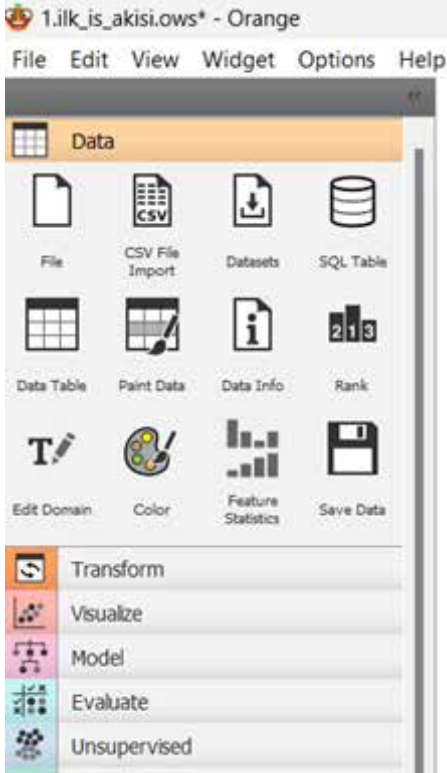
Rename : Nesnenin adını değiştirmek için kullanılır.

Duplicate : Nesneyi çoğaltmak için kullanılır.

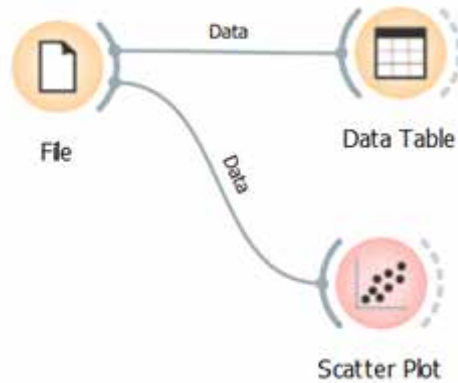
Copy : Nesneyi kopyalamak için kullanılır.

Help : Nesne hakkında bilgi almak için kullanılır.

- Nesneler eklenerek ve gerekli bağlantılar yapılarak görseldeki iş akışı oluşturulur.



İş Akışı Oluşturulması

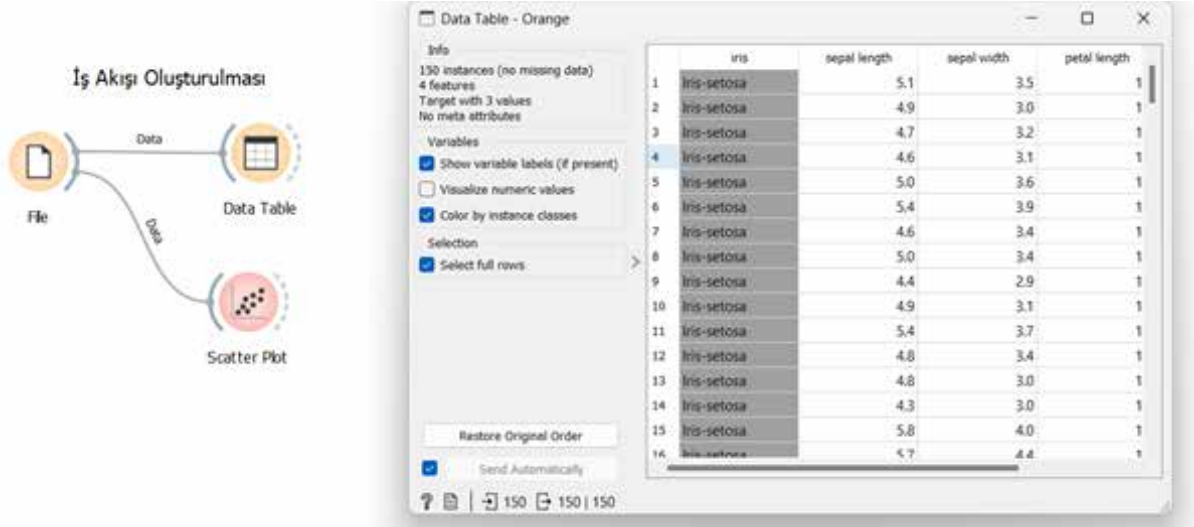


Görsel 4.12: Orange iş akışı

Orange

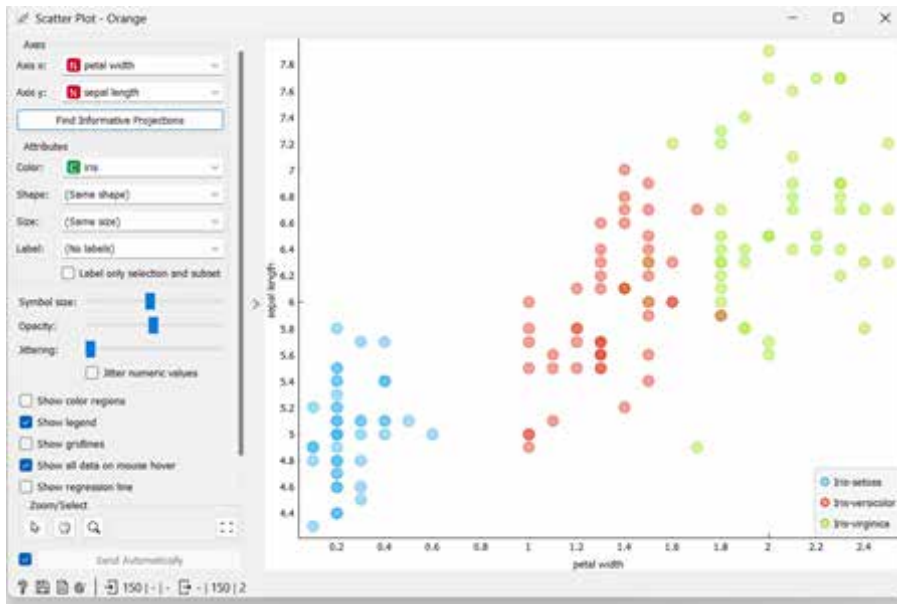
NOT: Orange ile farklı veri türündeki dosyalarla çalışılabilir. Program .tab uzantılı programa özel bir CSV formatında veri türü kullanmaktadır. Program içinde nesnelere kullanılarak veri türleri dönüştürülebilir.

- Bağlantılar oluşturulduktan sonra Data Table tıklanarak veri seti tablo şeklinde görüntülenir. Data Table nesnesi veri setini satırlar ve sütunlar olarak tablo şeklinde gösterir.



Görsel 4.13: Data Table nesnesi

- Veri setinin görselleştirilmesi amacıyla Scatter Plot nesnesi kullanılabilir. Scatter Plot nesnesine tıkladığında ise bir saçılım grafiği görülmektedir.



Görsel 4.14: Saçılım grafiği

Orange ile Keşifsel Veri Analizi

Seviye	12 yaş ve üzeri
Konu	2. Orange ile Keşifsel Veri Analizi
Süre	6 ders saati

Amaç

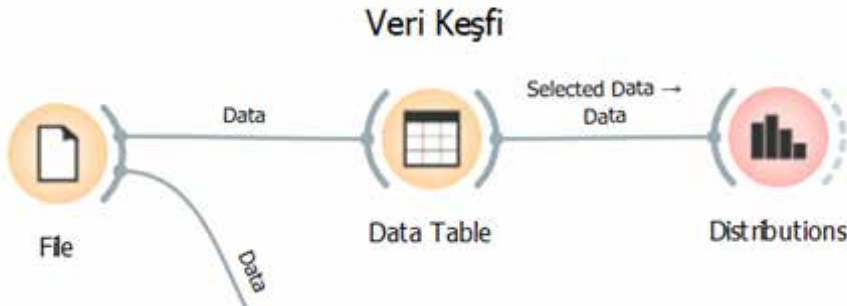
Bu uygulamada Orange programı kullanılarak bir iş akışı oluşturularak Iris veri dosyası yüklenerek incelenmektedir.

Motivasyon

Makine öğrenmesi uygulamalarında üzerinde çalışılacak veri setini incelemek, tanımak çok önemli bir adımdır. Bu aşama veri keşfi olarak adlandırılabilir. Veri keşfi aşamasında veri setindeki özellikler (sütunlar) , varsa etiket sütunu (hedef), eksik, hatalı veri tespit edilir ve veri ön işleme aşamasında kullanılacak stratejiler belirlenir.

İş Akışının Oluşturulması

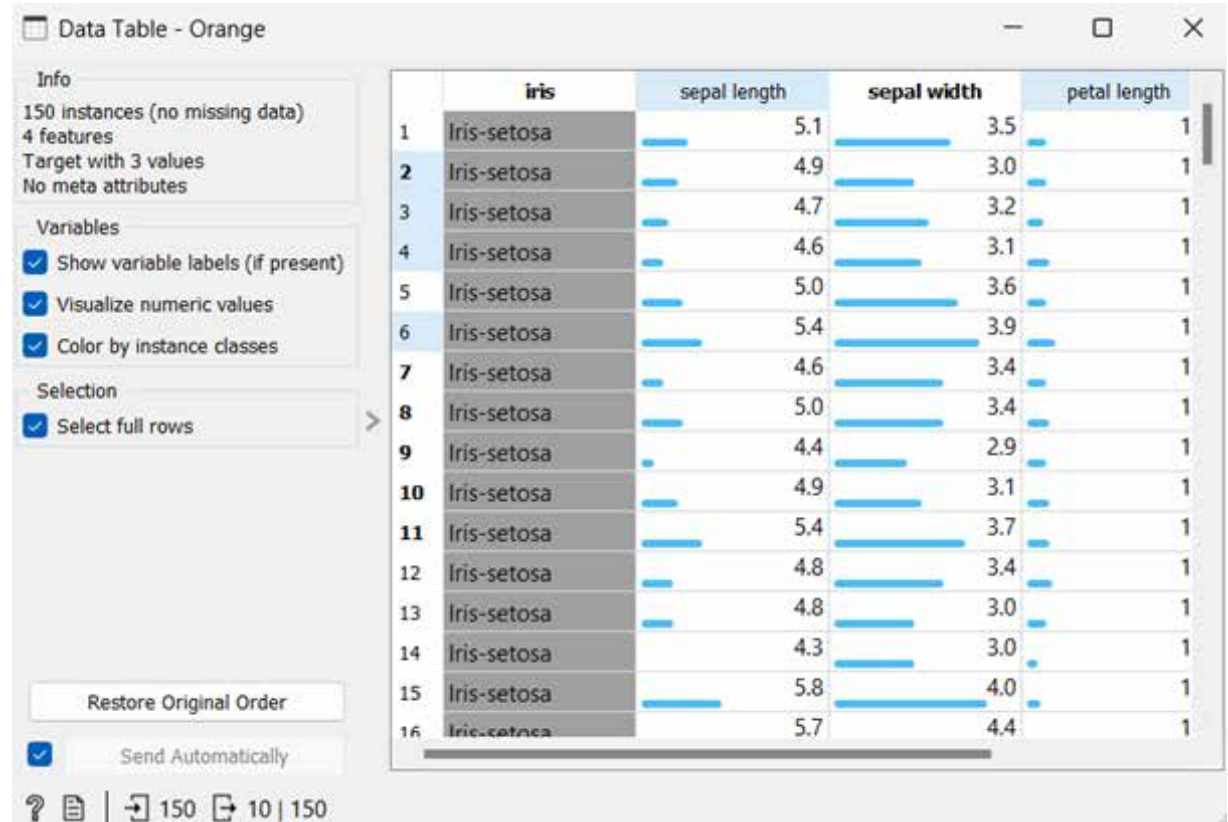
- Data araç setinden File nesnesi çalışma alanına eklenir. File nesnesine çift tıklanarak listeden iris.tab veri dosyası seçilir.
- Data araç setinden Data Table nesnesi eklenir. File nesnesinden Data Table nesnesine bir bağlantı oluşturulur.
- Visualize araç setinden Distributions nesnelere eklenir.
- Nesnelere eklenerek ve gerekli bağlantılar yapılarak görseldeki iş akışı oluşturulur.



Görsel 4.15: Keşifsel veri analizi

- Data Table çift tıklanarak tablo incelenir. Data Table nesnesi veri seti hakkında bilgi verir. Tablo incelenerek veri seti hakkında bilgi alınır.

Orange ile Keşifsel Veri Analizi



The screenshot shows the Orange Data Table widget with 16 rows of data. The columns are 'iris', 'sepal length', 'sepal width', and 'petal length'. The 'iris' column contains the value 'Iris-setosa' for all rows. The 'sepal length' column ranges from 4.3 to 5.8. The 'sepal width' column ranges from 3.0 to 4.0. The 'petal length' column ranges from 2.9 to 4.4. The left sidebar shows the following settings:

- Info:** 150 instances (no missing data), 4 features, Target with 3 values, No meta attributes.
- Variables:** Show variable labels (if present), Visualize numeric values, Color by instance classes.
- Selection:** Select full rows.

Buttons: Restore Original Order, Send Automatically (checked). Status bar: 150 rows, 10 columns, 150 instances.

Görsel 4.15: Data Table

Veri hakkında bilgi edinme

Bilgi (**Info**) başlığı altında 150 örnek (sıra-**instances**) olduğu görülmektedir. **No missing data** eksik veri olmadığını göstermektedir.

Dört sütun (özellik, **features**) bulunmaktadır. Dört özellikten biri hedef (etiket) özelliğidir.

No meta attributes ifadesi meta özellik olmadığını belirlemektedir.

Değişkenler (**Variables**) başlığı altında aşağıdaki ayarlar yapılabilir.

Show variable labels (If present) : Varsa değişken etiketlerini göster

Visualize numeric values : Sayısal değerleri görselleştir.

Color by instance classes : Örnekleri sınıflara göre renklendir.

Select full rows : Tüm satırları seç.

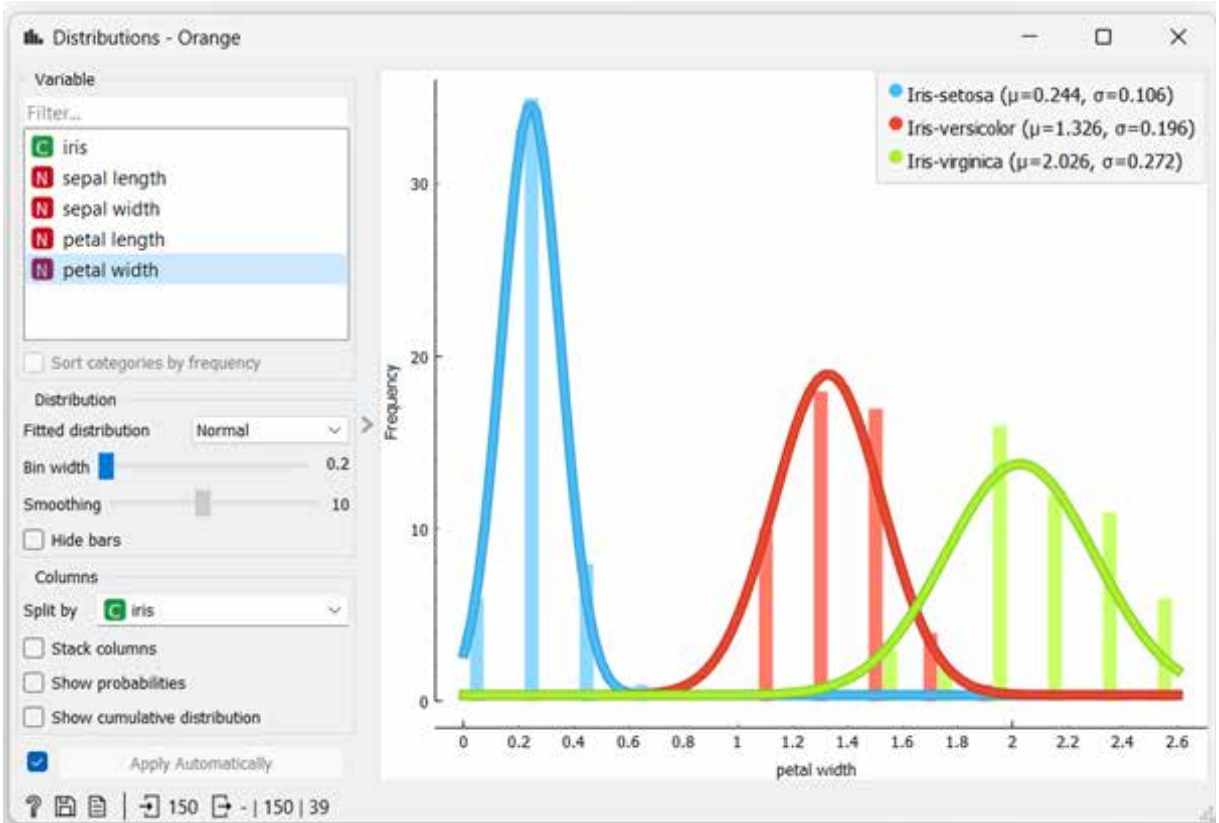


İris veri seti Fisher tarafından oluşturulan ve 1936'da yayınlanan makalesinde kullanılan bir veri setidir. Iris veri setinde beş sütun (özellik) ve 150 satır bulunmaktadır. Veri setinde her biri 50 örnek içeren üç iris (setosa, versicolor ve virginica) çiçek türü içerir. Veri setinde tür, çanak yaprağı uzunluğu, çanak yaprağı genişliği, taç yaprağı uzunluğu ve taç yaprağı genişliği özellikleri bulunmaktadır.

Iris veri seti veri biliminde kullanılan popüler veri setlerinden biridir.

Orange ile Keşifsel Veri Analizi

- Dağılımlara (Distributions) çift tıklanarak veri görselleştirme ile veri setindeki ögelerin durumu hakkında bilgi edinilir. Değişken başlığı altında filtreleme (Filter) bölümünde veri setinin özellikleri seçilerek verinin dağılımı görülebilir.



Görsel 4.16: Veri dağılımı

NOT: Herhangi bir nesnedeki değişiklik iş akışındaki tüm nesnelere etkiler.

Veri görselleştirme

Seviye

12 yaş ve üzeri

Konu

3. Veri görselleştirme

Süre

6 ders saati

Amaç

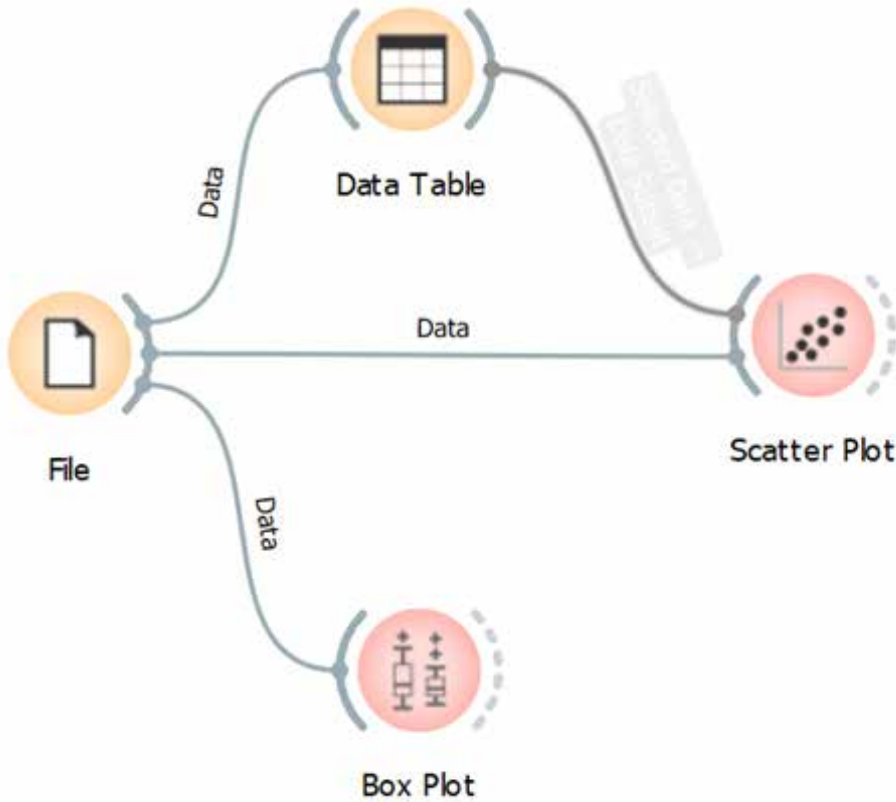
Bu uygulamada Orange programı kullanılarak bir iş akışı oluşturularak Iris veri dosyası görselleştirilmektedir. Görselleştirme araçları olarak **boxplot** ve **scatterplot** kullanılmıştır.

Motivasyon

Keşifsel veri analizi aşamasında veri görselleştirme araçları veri hakkında bilgi edinmek için oldukça kullanışlıdır. Farklı grafik türleri kullanılarak veri seti hakkında detaylı bir keşif yapılabilir.

İş Akışının Oluşturulması

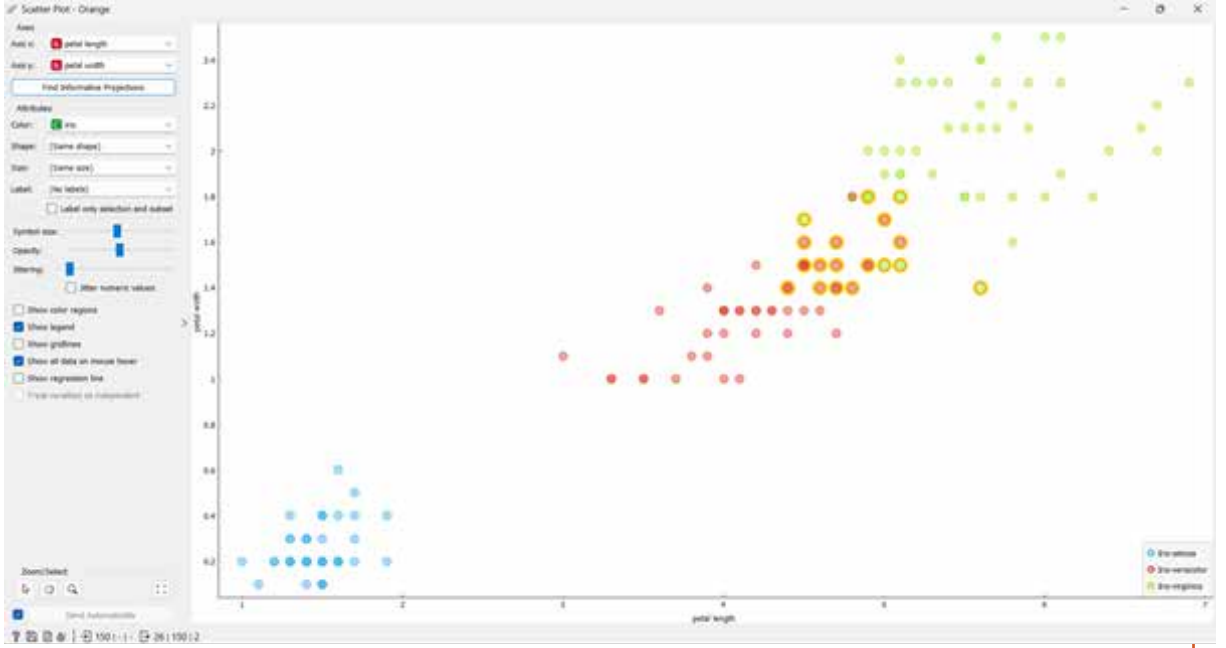
- **Data** araç setinden **File** nesnesi çalışma alanına eklenir. **File** nesnesine çift tıklanarak listeden iris.tab veri dosyası seçilir.
- **Data** araç setinden **Data Table** nesnesi eklenir. File nesnesinden **Data Table** nesnesine bir bağlantı oluşturulur.
- **Visualize** araç setinden **Scatter Plot** ve **Boxplot** nesneleri eklenir.
- Nesnelere eklenerek ve gerekli bağlantılar yapılarak görseldeki iş akışı oluşturulur.



Görsel 4.17: Akış diyagramı

Veri görselleştirme

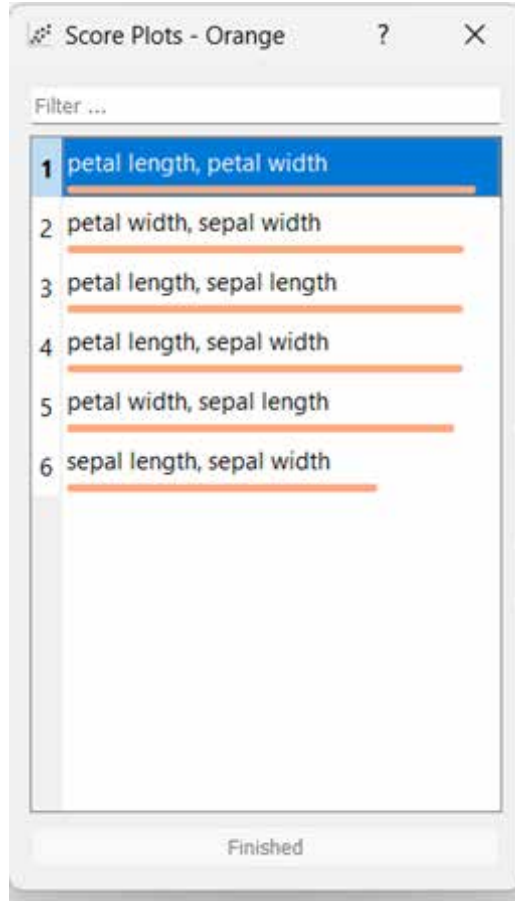
- Veri setini daha iyi tanımak için veri görselleştirme yolları kullanılabilir. İş akışında yer alan saçılım grafiğine tıklanarak sınıf etiketlerine göre renklendirilmiş olarak veri dağılımı görülebilir.



Görsel 4.18: Saçılım grafiği

- Axis x ve Axis y grafikteki x ve y eksenlerini temsil etmektedir. X ve y eksenleri için açılır listeden özellik seçilerek veri seti görselleştirilir. Bilgilendirici projeksiyonları bul (Find Informative Projectons) ile gösterim optimize edilebilir. Bu buton, veri setindeki özellikleri çiftler halinde eşleştirilerek bu özellik çiftlerini ortalama sınıflandırma doğruluğuna göre puanlar ve eşzamanlı bir görselleştirmeye en yüksek puan alan nitelik çiftini döndürür. Renk bölümünde kategorik bir değişken seçilirse puan hesaplanır. Her veri örneği için yöntem, yansıtılan 2B uzayda, yani nitelik çiftlerinin birleşiminde en yakın 10 komşuyu bulur. Bulunan bu 10 komşunun kaç tanesinin aynı renge sahip olduğunu kontrol eder. Projeksiyonun toplam puanı, aynı renkli komşuların ortalama sayısıdır. Iris veri setinde sınıflandırma için petal length -petal width çiftinin en bilgilendirici gösterim olduğu görülmektedir (Görsel 4.19).

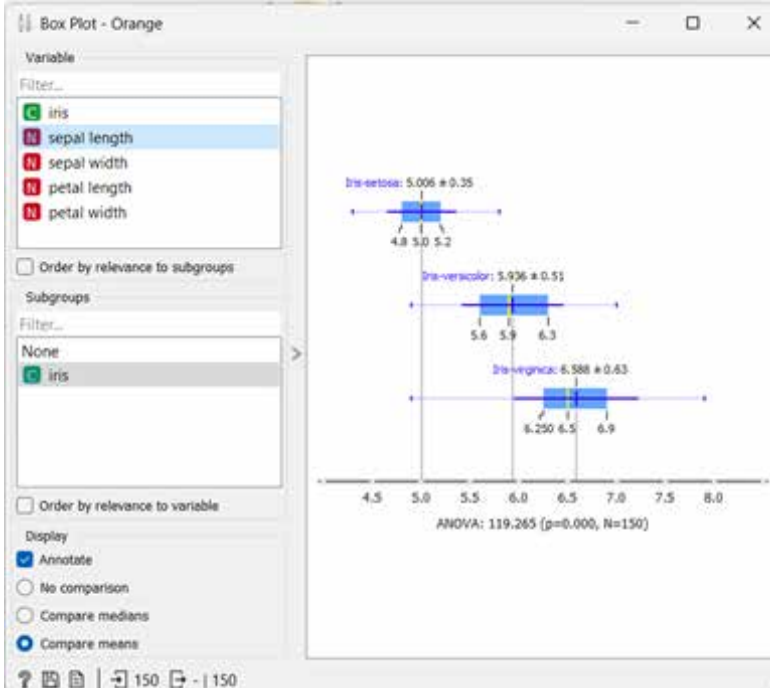
Veri görselleştirme



Görsel 4.19: Bilgilendirici gösterim keşfi

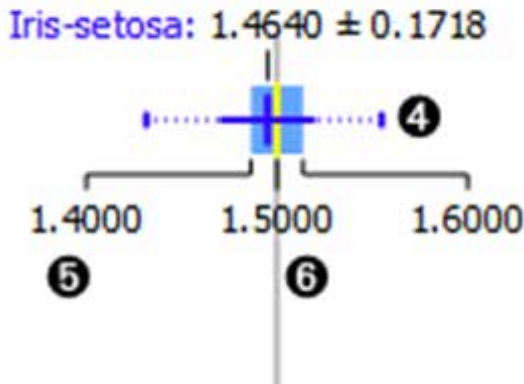
- Kutu Grafiği (Boxplot) kullanımı veri setinde yinelenen değerler, aykırı değerler ve ve anormallikleri hızla keşfetmek ve veriyi kontrol etmek için bir araçtır. Kutu grafiği kullanılarak kategorik veriler için değerler veya sayısal veriler için nicel aralıklar (sayısal değerler) kullanılabilir. Kutu Grafiği pencere aracı, bir veri kümesinin istatistiksel özelliklerini gözlemlemek için yaygın olarak kullanılan bir araçtır.

Veri görselleştirme



Görsel 4.19: Bilgilendirici gösterim keşfi

Görüntüle (Display) bölümünde Iris veri setindeki kategorilere göre ortancaları (median) veya ortalamaları (means) karşılaştırarak aralarında istatistiki olarak anlamlı bir fark olup olmadığını test eder.



Görsel 4.20: Bilgilendirici gösterim keşfi detayı

Ortalama (koyu mavi dikey çizgi).

İnce mavi çizgi standart sapmayı temsil eder.

Birinci (%25) ve üçüncü (%75) çeyreklik değerleri. Mavi ile vurgulanan alan, birinci ve üçüncü çeyrek arasındaki değerleri temsil eder.

Ortanca Medyan (sarı dikey çizgi).

Ayrı öz nitelikler için çubuklar, her bir belirli öz nitelik değerine sahip örneklerin sayısını temsil eder.

Veri Kümesi Oluşturma ve Yükleme

Seviye	12 yaş ve üzeri
Konu	4. Veri Kümesi Oluşturma ve Yükleme
Süre	6 ders saati

Amaç

Bu uygulamada e-tablolar kullanılarak bir veri seti oluşturulmaktadır. Google e-tablolar kullanılarak oluşturulan veri seti içe aktarılır.

Motivasyon

Makine öğrenmesi uygulamalarında farklı veri türleriyle çalışmak gerekebilir. Bu veri türleri (CSV, txt, xls) veri dosyaları olabileceği gibi çevrim içi e-tablolarda olabilir. Çevrim içi tablolarla çalışmak verinin sürekli olarak güncellenmesini böylelikle Orange iş akışında güncel veri ile çalışmasına olanak verir.

- Google e-tablolar kullanılarak şekildeki veri seti oluşturulmuştur. Oluşturulan tablo paylaşımına açılarak bağlantı linki kopyalanır.

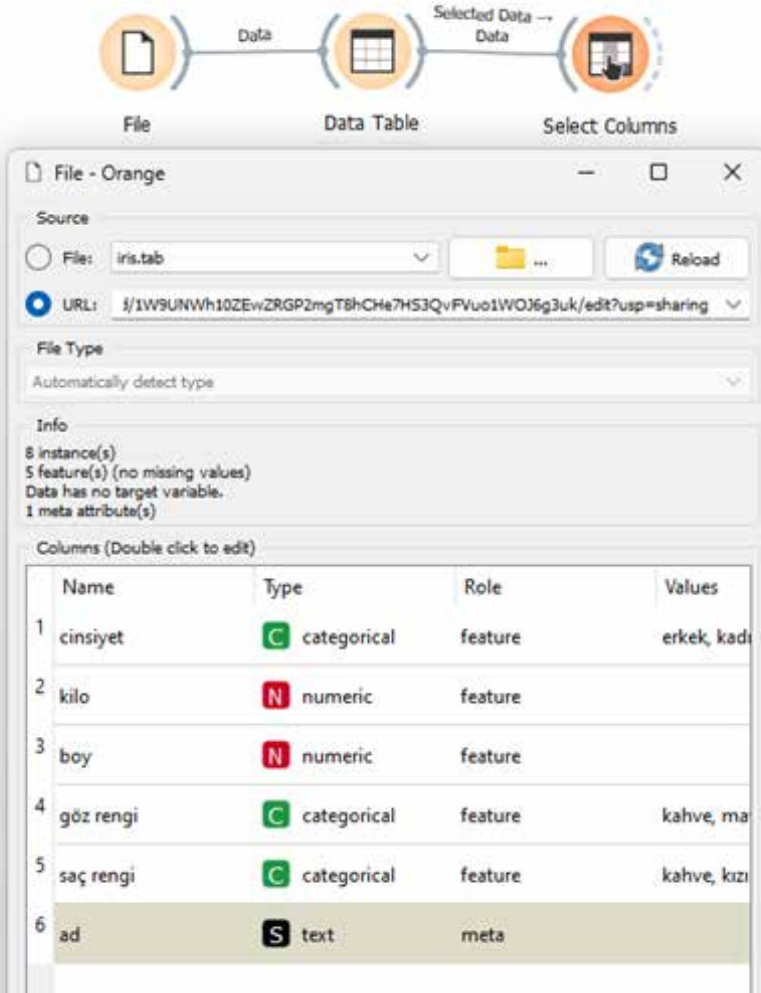
	A	B	C	D	E	F
1	ad	cinsiyet	kilo	boy	göz rengi	saç rengi
2	jill	kadın		52	1,6 mavi	kahve
3	Jack	erkek		78	1,75 kahve	kahve
4	Mark	erkek		67	1,65 kahve	siyah
5	Ann	kadın		62	1,7 yeşil	sarı
6	Bob	erkek		89	1,85 mavi	sarı
7	Tom	erkek		93	1,9 mavi	kızıl
8	Kate	kadın		56	1,67 kahve	kahve
9	Mary	kadın		70	1,64 mavi	kızıl
10						
11						

Görsel 4.21: Google e-tablo

İş Akışının Oluşturulması

- Çalışma alanına **File** nesnesi eklenerek çift tıklanır, **URL** bölümüne Google e-tablo bağlantı adresi yapıştırılır. Sütunlar (**columns**) bölümünde e-tablodaki özellikler sütun adı, veri türü, rol ve değerler görünecektir.

Veri Kümesi Oluşturma ve Yükleme



Görsel 4.22: Veri kümesi dosyası

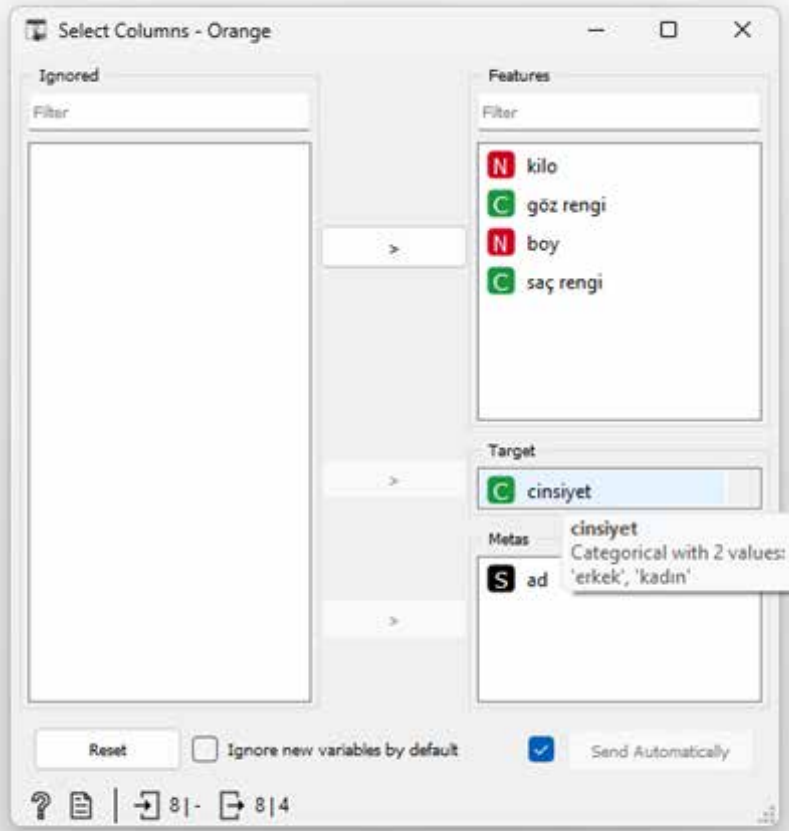
- Nesnelere eklenerek ve gerekli bağlantılar yapılarak görseldeki iş akışı oluşturulur.



Görsel 4.23: İş akışı

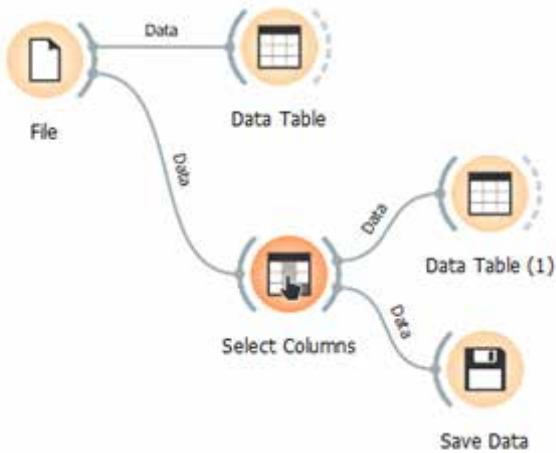
- Sütun seçme (**Select Columns**) nesnesi kullanılarak özellikler uygun bir şekilde belirlenir. Özellikler (**Features**) ve **Target** şeklindeki gibi belirlenir.

Veri Kümesi Oluşturma ve Yükleme



Görsel 4.24: Sütunların seçimi

- Oluşturulan tabloyu Orange yerel veri dosyası formatında kaydetmek için Save Data nesnesi kullanılır. Save Data nesnesine çift tıklanarak tablo yerel formatta (.tab) istenilen konuma kaydedilir.



Görsel 4.25: İş akış şeması

Kümeleme

Seviye	12 yaş ve üzeri
Konu	5. Kümeleme
Süre	6 ders saati

Amaç

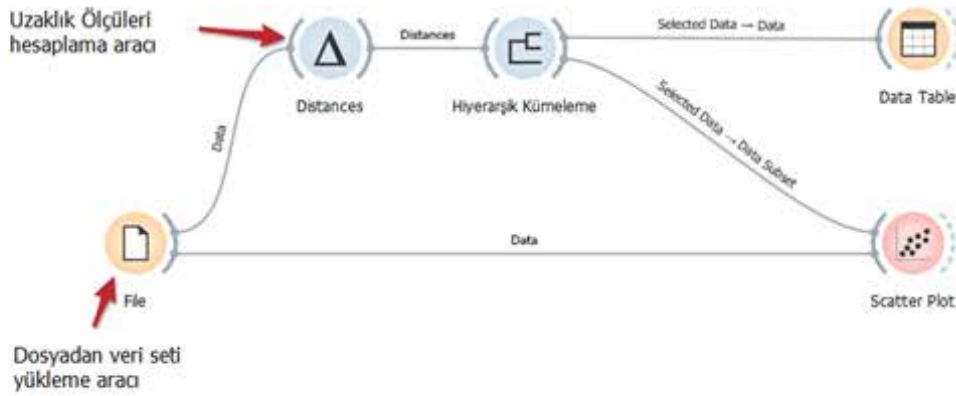
Bu uygulamada Orange programı kullanılarak bir kümeleme uygulaması yapılmıştır.

Motivasyon

Kümeleme yöntemi bir veri setindeki özelliklere göre veri setindeki grupların ve alt grupların keşfedilmesinde kullanılır. Kümeleme yönteminde aynı kümedeki öğelerin birbirleriyle arasındaki uzaklığın en az olduğu, küme merkezlerinin birbirleriyle uzaklıklarının en fazla olduğu kümeler oluşturulmak amaçlanmaktadır.

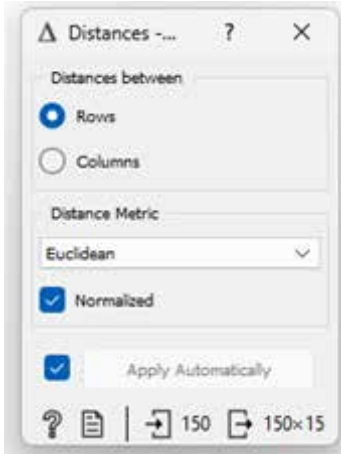
İş Akışının Oluşturulması

- **Data** araç setinden File ve Data Table nesneleri eklenir. File nesnesinde iris veri dosyası yolu gösterilir.
- **Unsupervised** araç setinden **Distances** ve **Hierarchical Clustering** nesneleri,
- **Visualize** araç setinden **Scatter Plot** nesnesi eklenir.
- **File** nesnesi çift tıklanarak iris veri dosyası yolu gösterilir.
- Nesneler arası bağlantılar yapılır.
- Nesneler eklenerek ve gerekli bağlantılar yapılarak görseldeki iş akışı oluşturulur.



Görsel 4.26: Hiyerarşik kümeleme

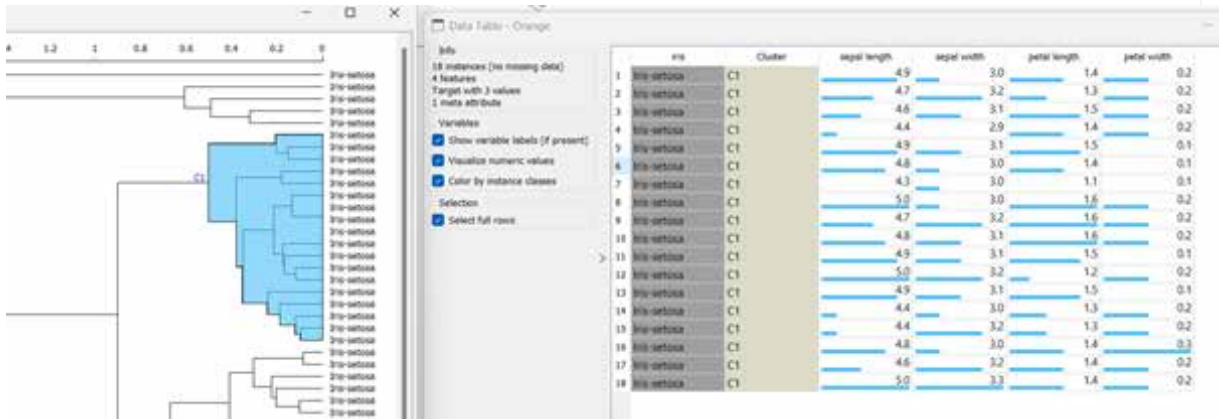
Kümeleme



Görsel 4.27: Distance

NOT: Hiyerarşik kümeleme kümeler oluştururken ögeler arası uzaklıkları hesaplamaktadır. Bu uzaklıklar Distance nesnesi kullanılarak hesaplanır. Distance nesnesine çift tıklanarak uzaklık metriği değiştirilebilir. Her uzaklık metriği ögeler arası uzaklığı farklı bir şekilde hesaplamaktadır.

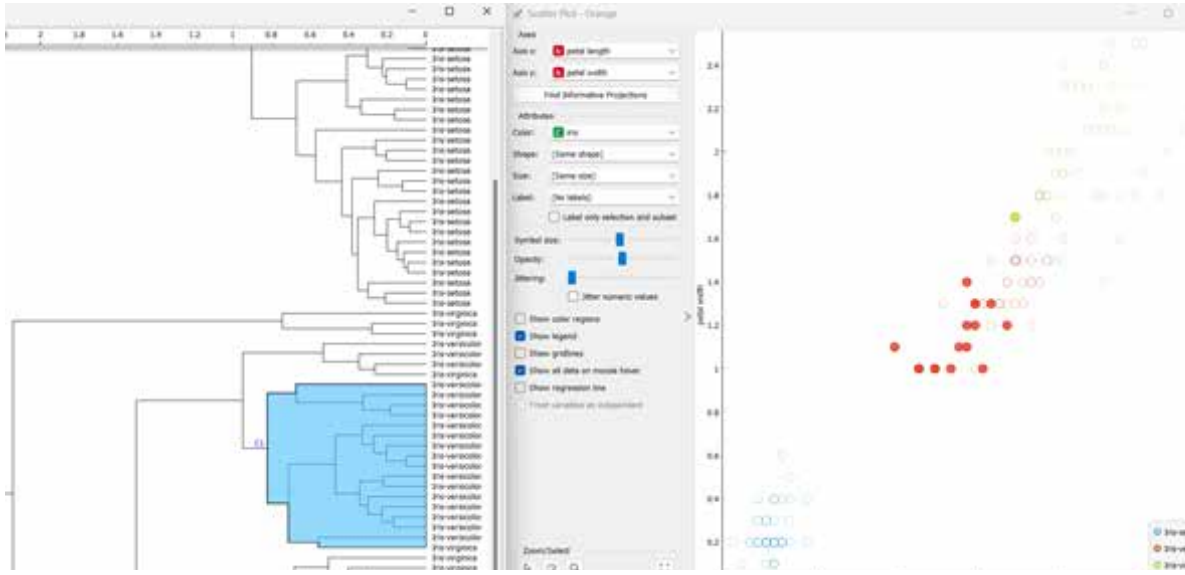
- Hiyerarşik kümeleme nesnesine çift tıkladığında ögeler görseldeki gibi görünecektir. Bu bölümdeki hiyerarşik dallanmadan seçilen veri veya veri grupları Data Table nesnesinde listelenir.



Görsel 4.28: Hiyerarşik dallanma

- Hiyerarşik kümeleme nesnesinde seçilen ögeler **Scatter Plot** nesnesinde görselleştirilebilir. Bu sayede öge grupları saçılım grafiğinde (scatter plot) görselleştirilebilir.

Kümeleme



Görsel 4.29: Hiyerarşik kümeleme (scatter plot)

Tahmin (Prediction) Modelleri

Seviye	12 yaş ve üzeri
Konu	6. Tahmin (Prediction) Modelleri
Süre	6 ders saati

Amaç

Motivasyon

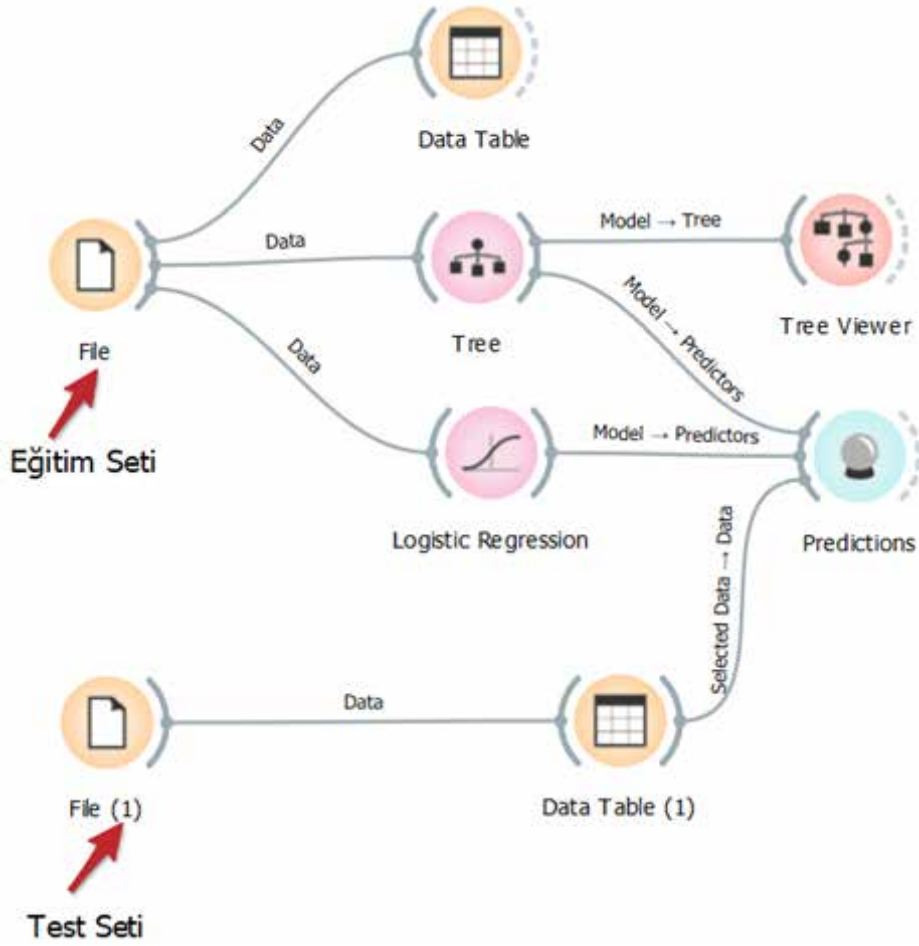
Bu uygulamada bir veri seti (eğitim seti) kullanılarak bir model eğitilmekte ve bu model kullanılarak sınıf etiketi verilmeyen bir veri setindeki (test seti) örneklerin ait oldukları sınıflar tahmin edilmektedir.

İş Akışının Oluşturulması

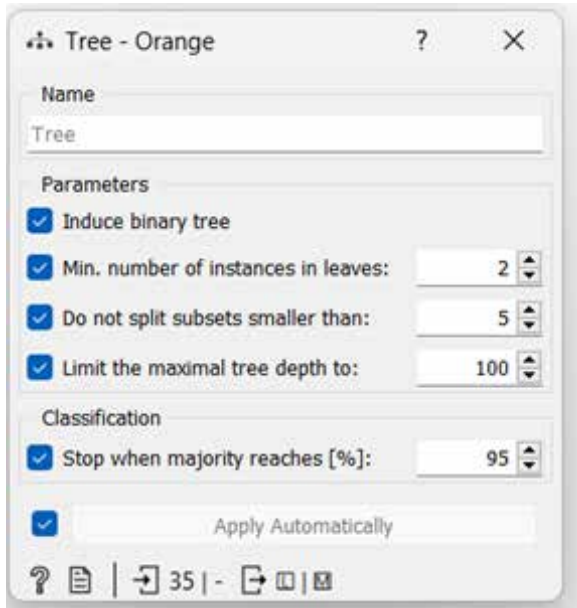
Çalışma alanına

- **Data** araç setinden iki adet **File** ve iki adet **Data Table**, eğitim seti için fruits an vegetables train, test seti için fruits an vegetables test dosyaları gösterilir.
- **Model** araç setinden **Tree, Logistic Regression**
- **Visualize** araç setinden **Tree Viewer**
- **Evaluate** araç setinden **Predictions** nesneleri eklenir.
- Nesneler arası bağlantılar yapılır.

Tahmin (Prediction) Modelleri



Tree nesnesine çift tıklanarak ağaç parametre ayarları yapılır.

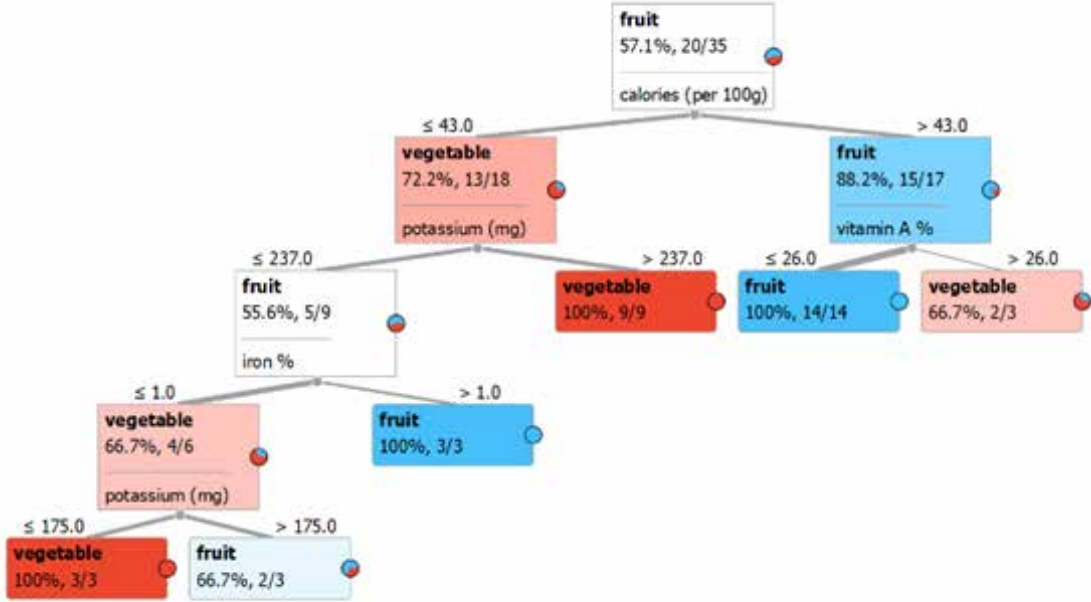


Görsel 4.31: Ağaç parametrelerinin ayarlanması

Tahmin (Prediction) Modelleri

Bu iş akışında iki sınıflandırma modeli (**Tree** ve **Logistic Regression**) kullanılmıştır. Birden fazla model kullanılarak içlerinden en iyi sonucu veren model seçilebilir.

Tree Viewer nesnesine tıklandığında karar ağacı ekrana gelir. Öğenin özellik değerlerine göre ait olduğu sınıf etiketi dallar takip edilerek yapraklara ulaşılarak bulunur.



Görsel 4.32: Karar ağacı

Predictions nesnesi girdi (input) olarak model tahmin edici (model predictor) ve veri seti (data) alır. Predictions eğitim veri seti kullanılarak eğitilen model, test veri seti üzerinden etiket tahmini yapmaktadır. Predictions nesnesine çift tıklanarak model içinde yapılan tahminler karşılaştırılabilir.

	Tree	Logistic Regression	name	vitamin A %	vitamin C %	calcium %	iron %	
1	fruit	fruit	?	1.0	154.0	3.0	1.0	4.
2	fruit	vegetable	?	15.0	9.0	2.0	11.0	3.
3	fruit	fruit	?	0.0	43.0	2.0	3.0	5.

Görsel 4.33: Tahmin oranları

Modelleri Deęerlendirme

Seviye	12 yař ve üzeri
Konu	7. Modelleri Deęerlendirme
Süre	6 ders saati

Amaç

Motivasyon

Bir sınıf tahmin modelinin oluřturularak modelin başarısının nasıl deęerlendirileceęi gösterilmektedir.

İř Akıřının Oluřturulması

Çalıřma alanına

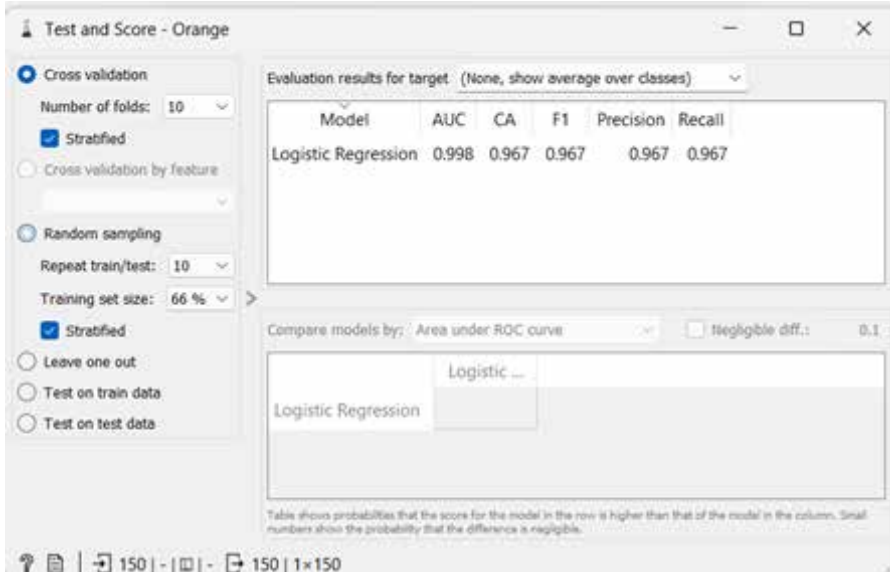
- Data araç setinden File ve Data Table nesnesi eklenir, File nesnesine çift tıklanır ve iris veri dosyası yolu gösterilir.
- Model araç setinden Logistic Regression nesnesi eklenir.
- Evaluate araç setinden Test and Score nesnesi eklenir.
- Nesnelar arasında gerekli baęlantılar yapılır.



Görsel 4.34: Modellerin deęerlendirilmesi

Test and Score nesnesine çift tıklanarak veri seti eğitim ve test olarak bölünebilir. Bu bölümde başka bir model test yöntemi olarak Cross Validation seçilebilir.

Modelleri Değerlendirme

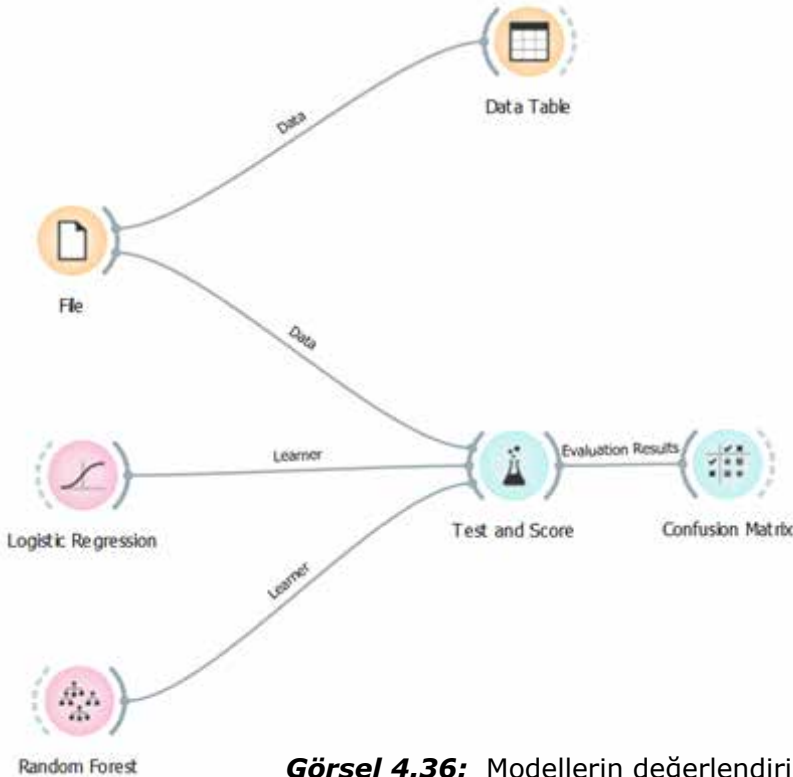


Görsel 4.35: Test yönteminin seçimi

Bu bölümde modele ilişkin performans metrikleri yer almaktadır: AUC, CA, F1, Precision ve Recall.

Eğer iş akışında birden fazla model oluşturulursa bu bölümde modellere ilişkin performans metrikleri alt alta listelenir.

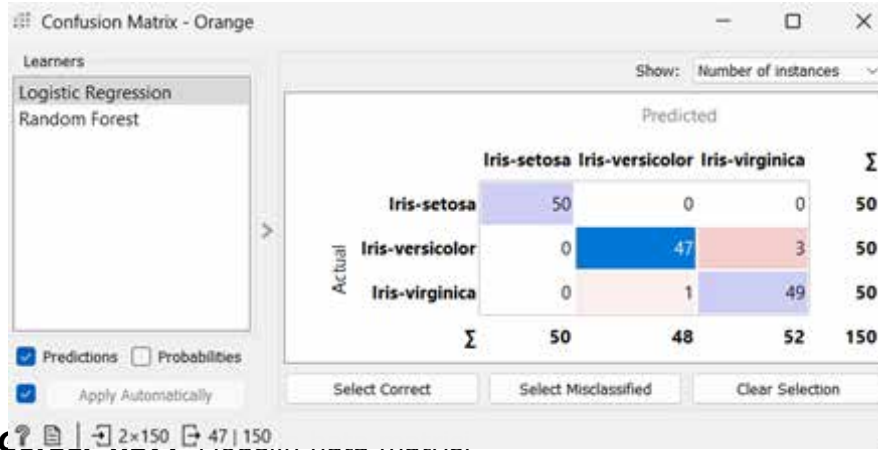
Bir iş akışında birden fazla model oluşturularak performansları değerlendirilebilir. İş akışında Lojistik regresyon ve Rastgele Orman modelleri oluşturularak Test and Score nesnesiyle modelin performans metrikleri hesaplatılır.



Görsel 4.36: Modellerin değerlendirilmesi

Modelleri Deęerlendirme

Confusion Matrix (Karışıklık-Hata Matrisi) nesnesi modelin tahminleriyle geręek sınıfların bir matrisini sunar. Bu sayede örneklerin ait oldukları geręek sınıflar ile modelin tahminleri matris üzerinden karşılaştırılabilir. İş akışında birden fazla model varsa **Learners** bölümünde yer alan model ismine tıklanarak o modele ait hata matrisine ulaşılır.



Orange eklentileri

Seviye

12 yaş ve üzeri

Konu

7. Modelleri Değerlendirme

Süre

6 ders saati

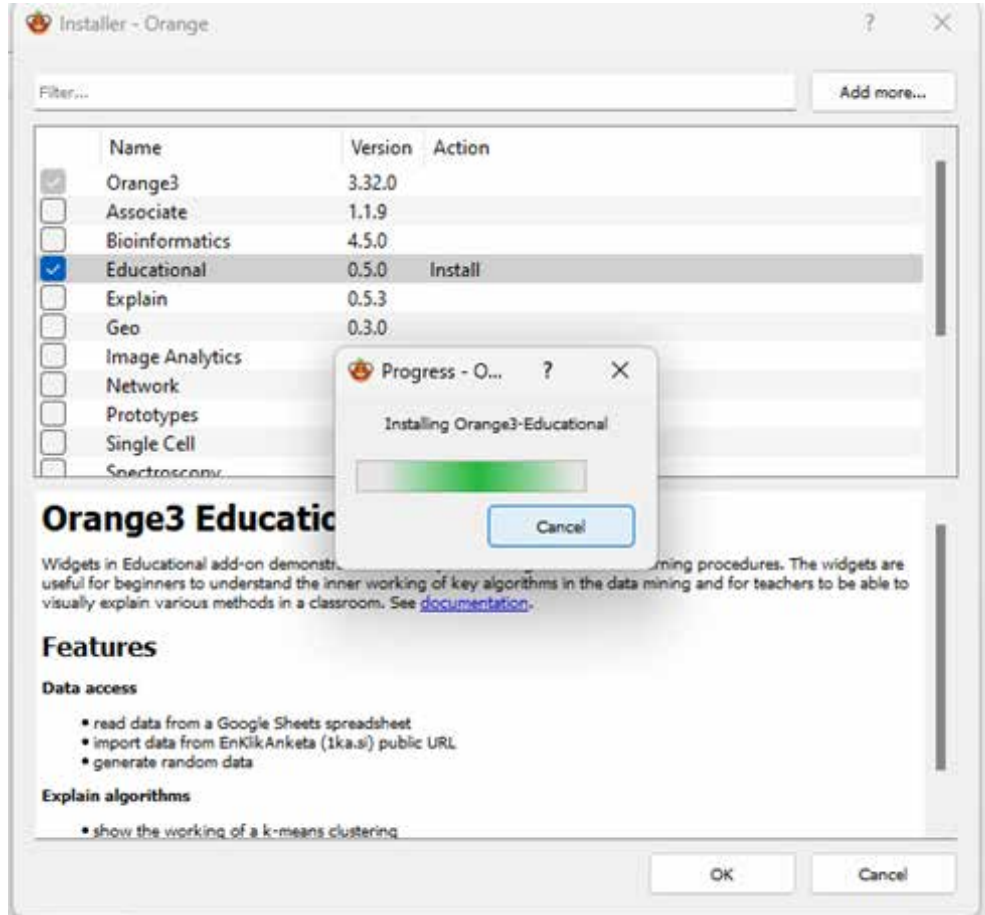
Amaç

Motivasyon

Orange programında çeşitli çalışmalar için eklentiler bulunmaktadır. Kullanıcılar yaptıkları çalışmaya göre bu eklentilerden yararlanabilir. En popüler eklentiler biyoenformatik, eğitsel, imaj analitik, metin ve zaman serileridir.

İş Akışının Oluşturulması

Eklentileri yüklemek için Options menüsünden Add-ons seçilir. Kullanılmak istenen paket seçilir ve kurulur.



Görsel 4.38: Option menüsünden eklenti yüklemek

Paket kurulduğunda programın yeniden başlatılması gerekebilir. Program yeniden başladığında kurulan paket araç seti olarak panelde yerini alır.

Orange eklentileri



Görsel 4.38: Eklentinin görünümü

Temel Bileşenler Analizi (PCA)

Seviye

12 yaş ve üzeri

Konu

7. Modelleri Değerlendirme

Süre

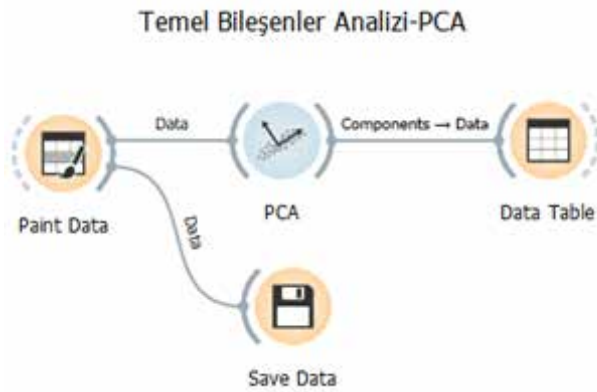
6 ders saati

Amaç

Motivasyon

Bazı durumlarda veri setinde onlarca özellik olabilir. Bu özelliklerden anlamlı olanların seçilmesi daha az özellikle modeller oluşturulmasına olanak verir. Veri sıkıştırma gibi işlemlerde de kullanılmaktadır. Veriyi en iyi şekilde temsil edecek öznelikler bulunur. Böylece onlarca özelliği temsil edecek daha az sayıda özellikli etkili modeller oluşturulabilir. Boyut azaltma olarak adlandırılan bu yöntemlerin en popülerlerinden biri de Temel Bileşenler Analizi (Principal Component Analyse-PCA)'dir.

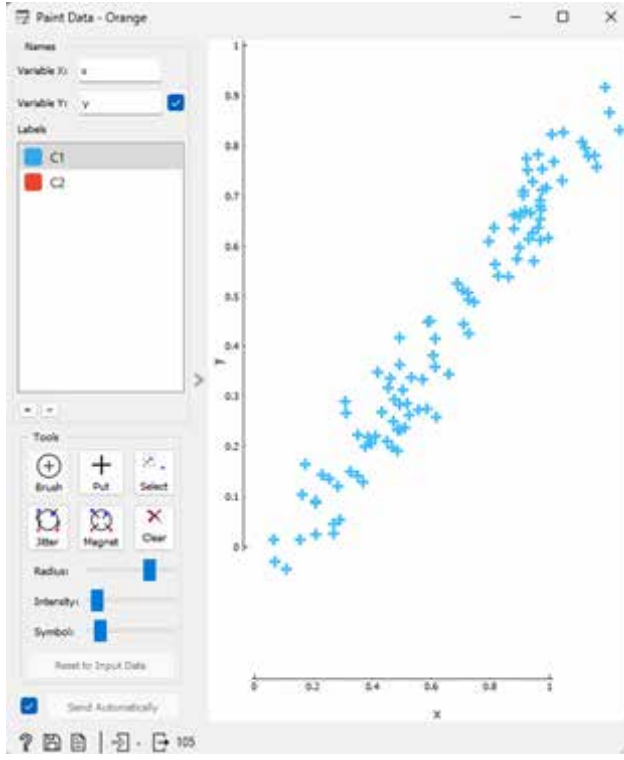
İş Akışının Oluşturulması



Görsel 4.39: Temel bileşenler analizi için iş akışı uygulaması

Paint Data nesnesi kullanılarak veri noktaları fare ile işaretlenerek (boyanarak) veri seti oluşturulmasını sağlamaktadır.

Temel Bileşenler Analizi (PCA)



Görsele 4.40: Paint Data nesnesi ile veri seti oluşturulması

KAYNAKÇA

Bölüm-1

- [1] Wikipedia. https://en.wikipedia.org/wiki/Alan_Turing adresinden 20.06.2022 tarihinde alındı.
- [2] BBC. <https://www.bbc.com/news/technology-27762088> adresinden 21.06.2022 tarihinde alındı.
- [3] IMDB. <https://www.imdb.com/title/tt2084970/> adresinden 21.06.2022 tarihinde alındı.
- [4] Britannica. <http://meb.ai/hRNibI> adresinden 20.06.2022 tarihinde alındı.
- [5] Ulakbim. <http://meb.ai/LQLv2p> adresinden 20.06.2022 tarihinde alındı.
- [6] DeepLearning. 2022. <http://meb.ai/Oq56XL> adresinden 19.06.2022 tarihinde alındı.
- [7] TDK. 2022. <https://sozluk.gov.tr/> adresinden 19.06.2022 tarihinde alındı
- [8] Oracle. 2022. <https://www.oracle.com/tr/artificial-intelligence/what-is-ai/> adresinden 19.06.2022 tarihinde alındı.
- [9] BBC. 2022. <http://meb.ai/hMxF2W> adresinden 20.06.2022 tarihinde alındı
- [10] Dartmouth Üniversitesi.2022. <http://meb.ai/Uk2xLM6> adresinden 23.06.2022 tarihinde alındı.
- [11] Britannica.2022 <http://meb.ai/Uudnry9> adresinden 22.06.2022 tarihinde alındı.
- [12] Wikipedia. 2022. <http://meb.ai/7KPYjK> adresinden 20.06.2022 tarihinde alındı.
- [13] Queensland Üniversitesi.2022. <http://meb.ai/uYpHNO> adresinden 25.06.2022 tarihinde alındı.
- [14] Youtube.2018. <http://meb.ai/4XtfBJ> adresinden 17.08.2022 tarihinde alındı.
- [15] BBC. 2022. <http://meb.ai/UY99V6w> adresinden 19.06.2022 tarihinde alındı.
- [16] IBM.202. <http://meb.ai/UtQg7ML> adresinden 17.06.2022 tarihinde alındı.
- [17] Azure. 2022. <http://meb.ai/PJDsLU> adresinden 17.06.2022 tarihinde alındı.
- [18] CBDO. 2022. <https://cbddo.gov.tr/yapay-zeka/> adresinden 15.06.2022 tarihinde alındı.
- [19] Bloomberght. 2022. <http://meb.ai/qTEb7N> adresinden 17.06.2022 tarihinde alındı.
- [20] BBC. 2017. <http://meb.ai/UW2xomH> adresinden 17.06.2022 tarihinde alındı.

- [21] TRT. 2019. <http://meb.ai/UVV5E1d> adresinden 13.06.2022 tarihinde alındı.
- [22] IBM. 2022. <https://www.ibm.com/topics/computer-vision> adresinden 22.06.2022 tarihinde alındı.
- [23] BM. 2022. <http://meb.ai/IhMUhX> adresinden 22.06.2022 tarihinde alındı.
- [24] Differencebetween. 2022. <http://meb.ai/0ryqXc> adresinden 22.06.2022 tarihinde alındı.
- [25] Youtube. 2022. <http://meb.ai/UVu4pHA> adresinden 22.06.2022 tarihinde alındı.
- [26] Youtube. 2022. <http://meb.ai/UVu4pHA> adresinden 22.06.2022 tarihinde alındı.
- [27] Generated. 2022. <https://generated.photos/face-generator/new> adresinden 25.06.2022 tarihinde alındı.
- [28] Youtube. 2022. <https://youtu.be/r2dQgdktUJg> adresinden 25.06.2022 tarihinde alındı.
- [29] OpenAI. 2022. <https://beta.openai.com/examples> adresinden 26.06.2022 tarihinde alındı.
- [30] Hyperight. 2022. <http://meb.ai/jLEss6> adresinden 25.06.2022 tarihinde alındı.
- [31] Beyazperde. 2022. <https://www.beyazperde.com/filmler/film-140005/> adresinden 26.06.2022 tarihinde alındı.
- [32] IBM. 2022. <http://meb.ai/UVEeraR> adresinden 26.06.2022 tarihinde alındı.
- [33] Wikipedia. 2022. <http://meb.ai/jqeqvj> adresinden 26.06.2022 tarihinde alındı.
- [34] Yapay Zekâ TR. 2018. <http://meb.ai/ZV6lPl> adresinden 26.06.2022 tarihinde alındı.
- [35] Toyota. 2022. <http://meb.ai/UKDYjzq> adresinden 27.06.2022 tarihinde alındı.
- [36] Youtube. 2022. <http://meb.ai/oOqFGb> adresinden 22.06.2022 tarihinde alındı.
- [37] Dünya. 2022. <http://meb.ai/gYi2jW> adresinden 20.06.2022 tarihinde alındı.
- [38] The Great Hack. Bağlantı: <http://meb.ai/ndUtfA>
- [39] TRT. 2022. <http://meb.ai/UEedAEa> adresinden 23.06.2022 tarihinde alındı.
- [40] V7Labs. 2022. <https://www.v7labs.com/blog/ai-in-agriculture> adresinden 24.07.2022 tarihinde alındı.
- [41] Google Drive. <http://meb.ai/rOr0vi>
- [42] Sputnik. 2022. <http://meb.ai/Ut8uzHy> adresinden 24.06.2022 tarihinde alındı.
- [43] Altun. M., 2019, Doktora tezi.

[44] Youtube. Bağlantı: <http://meb.ai/avWEKi>

[45] Google AI. 2022. <https://ai.google/static/documents/exploring-6-myths.pdf> adresinden 17.08.2022 tarihinde alındı.

[46] ITIF. 2022. <http://meb.ai/Uhsnyrn> adresinden 17.08.2022 tarihinde alındı.

Bölüm-2

[1] Code.org bağlantısı. <https://studio.code.org/s/oceans/lessons/1/levels/2>

[2] Google TM. 2022. <https://teachablemachine.withgoogle.com/> adresinden 10.03.2020 tarihinde alındı.

[3] Model bağlantısı. https://teachablemachine.withgoogle.com/models/SMwI3_ID/

[4] Model bağlantısı. <https://teachablemachine.withgoogle.com/train/audio>

[5] Model bağlantısı. <https://teachablemachine.withgoogle.com/models/AYyTe8oF/>

[6] <https://www.youtube.com/watch?v=59bMh59JQDo> adresinden transkrip edilerek Türkçe 'ye çevrilmiştir.

[7] Kaspersky. 2022. <http://meb.ai/8EI9Pc> adresinden 15.06.2022 tarihinde alındı.

[8] Visage bağlantısı. <http://meb.ai/B7NSVG>

[9] Tensorflow, https://www.tensorflow.org/lite/examples/object_detection/overview adresinden 23/06/2022 tarihinde alındı.

[10] Tensorflow, https://www.tensorflow.org/lite/examples/pose_estimation/overview adresinden 23/06/2022 tarihinde alındı.

[11] stemPedia bağlantısı <https://thestempedia.com/project/>

[12] İşNEt. <https://www.isnet.net.tr/BlogIcerik/api-nedir-isnet-blog> adresinden 25/06/2022 tarihinde alındı

Bölüm-3

[1] Partnershipfor 21st Century Learning (21. Yüzyıl Öğrenme Ortaklığı) (2015). *Framework for 21st Century Learning*. <http://www.p21.org/ourwork/> p21- framework, Erişim tarihi: 19.07.2022.

[2] Gonzalez, H.B. ve Kuenzi, J.J. (2012). *Science, technology, engineering, and mathematics (STEM) education: A primer*. Retrieved from [http:// fas.org/sgp/crs/misc/R42642.pdf](http://fas.org/sgp/crs/misc/R42642.pdf) on March 1, Erişim tarihi: 11.07.2022.

[3] Üçüncüoğlu, İ. (2018). *Fen bilgisi öğretmen adaylarına yönelik STEM odaklı laboratuvar uygulamalarının tasarlanması ve etkililiğinin araştırılması*. Sinop Üniversitesi: Yayınlanmamış yüksek lisans tezi.

[4] Baran, E., Canbazoğlu S. ve Mesutoğlu, C. (2015). Fen, teknoloji, mühendislik ve matematik (FeTeMM) spotu geliştirme. *Araştırma Temelli Etkinlik Dergisi (ATED)*, 5(2):

- [5] Ostler, E. 2012. 21st century STEM education: a tactical model for long-range success. *International Journal of Applied Science and Technology*, 2(1): 28-33.
- [6] Akgündüz, D., Aydeniz, M., Çakmakçı, G., Çavaş, B., Çorlu, M., Öner, T. ve Özdemir, S.(2015). STEM eğitimi Türkiye raporu: "Günün modası mı? Yoksa gereksinim mi? İstanbul: STEM Merkezi ve Eğitim Fakültesi, İstanbul Aydın Üniversitesi.
- [7] Kutlu, E. (2019). FeTeMM destekli fen öğretiminin 8. Sınıf öğrencilerinin sorgulayıcı öğrenme becerileri algısı ve mühendislik bilgi düzeyi üzerindeki etkisi: basit makineler örneği. Van Yüzüncü Yıl Üniversitesi:Yayınlanmamış yüksek lisans tezi.
- [8] Morrison, J., (2006). TIES STEM education monograph series, attributes of STEM education. Teaching Institute for Essential Science. Retrieved.
- [9]Kavacık, İ. (2019). Fen, teknoloji, mühendislik ve matematik (stem) uygulamalarının; öğrencilerin öğrenme yaklaşımlarına, sorgulayıcı öğrenme becerisi algılarına ve stem'e yönelik tutumlarına etkisi. Mersin Üniversitesi: Yayınlanmamış yüksek lisans tezi.
- [10] Kutlu, E.(2021).Yaşam Becerilerinin Gerçekleştirilmesinde FeTeMM Eğitiminin Önemi, Bakırcı H. (Ed), 21. Yüzyıl Becerilerine Uzanan Yolculuk: Fen Öğretiminde Güncel Yaklaşımlar ve Yaşam Becerileri içinde (s.227-246), İzmir: Duvar yayınları.
- [11] Yıldırım, A. (1996). Disiplinlerarası Öğretim Kavramı ve Programlar Açısından Doğurduğu Sonuçlar. Hacettepe Üniversitesi Eğitim Fakültesi Dergisi, 12: 89-94.
- [12] Cavanagh, S. & Trotter. A. (2008). Where's the "T" in STEM?. <http://www.edweek.org/ew/articles/2008/03/27/30stemtech.h27.html>. Erişim tarihi: 01.07.2022.
- [13] Bozkurt Altan, E. 2017a. Fen, Teknoloji, Mühendislik Ve Matematik (FeTeMM-STEM) Eğitimi, Hastürk, H. G. (Ed.), Teoriden Pratiğe Fen Bilimleri Öğretimi içinde, (s.354-388). Ankara:Pegem Yayıncılık 354-388.
- [14] Stohlmann, M. Moore, T, Roehrig, G. 2012. Considerations for Teaching Integrated STEM Education. *Journal of Pre-College Engineering Education Research (J- PEER)*, 2(1): 27-34.
- [15] Karahan, E. ve Canbazoğlu Bilici, S. (2018). STEM eğitiminde teknoloji entegrasyonu. A. Tekbıyık. ve G. Çakmakçı (Ed.). Fen bilimleri öğretimi ve STEM etkinlikleri içinde (s.265-282). Ankara: Nobel Yayıncılık.
- [16] Özdemir, Ş. (2019). Geleceğin Meslekleri Listesi.
- [17] Smith, J., Karr-Kidwell, PJ. 2000. The interdisciplinary curriculum: a literary review and a manual for administrators and teachers. <http://files.eric.ed.gov/fulltext/ED443172.pdf>. Erişim tarihi: 12.06.2022.
- [18]Sanders, M.E., Wells, J.G. 2010. Virginia tech, integrative STEM education graduate program <http://web.archive.org/web/20100924150636/> <http://www.soe.vt.edu/istemed> Erişim tarihi: 12.06.2022.
- [19] Bozkurt, E. 2014. Mühendislik tasarım temelli fen eğitiminin fen bilgisi öğretmen adaylarının karar verme becerisi, bilimsel süreç becerileri ve sürece yönelik algılarına etkisi. Gazi Üniversitesi: Yayınlanmamış Doktora Tezi.
- [20] Ercan S. 2014. Fen eğitiminde mühendislik uygulamalarının kullanımı: tasarım temelli fen eğitimi. Marmara Üniversitesi: Yayınlanmamış Doktora Tezi.
- [21] Gencer, A. S., Doğan, H., Bilen, K. ve Can, B.(2019). Bütünleşik STEM eğitimi modelleri. Pamukkale Üniversitesi Eğitim Fakültesi Dergisi, 45, 38-55.
- [22] Buyruk, B. ve Korkmaz, Ö. (2016). FeTeMM farkındalık ölçeği (FFÖ): Geçerlik ve

güvenirlık çalıřması. Türk Fen Eđitimi Dergisi, 13(2), 61-76.

[23] Özkan, c. (2020). STEM eđitimi bađlamında öđretmenlerin el becerilerinin ölçümü ve deđerlendirilmesi. Bursa Uludađ Üniversitesi:Yayınlanmamıř doktora tezi.

[24] Topçu, M., S. ve Gökçe, A., (2018). STEM ve Mühendislik. Kırkıç, K. A. ve Aydın E. (Ed).Merhaba STEM Yenilikçi Bir Öđretim Yaklařımı içinde, (s.79-94). Konya: Eđitim Yayınevi.

[25] Hynes, M., Portsmore, M., Dare, E., Milto, E., Rogers, C., Hammer, D., Carberry, A. 2011. Infusing engineering design into high school STEM courses

[26] WHO. 2022. <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries> Eriřim tarihi: 21.07.2022.

[27] EGM. 2022. <http://trafik.gov.tr/istatistikler37> Eriři tarihi: 21.07.2022

[28] TÜİK. 2022. <https://data.tuik.gov.tr/Bulten/Index?p=Karayolu-Trafik-Kaza-Istatistikleri-2021-45658> Eriřim tarihi: 10.06.2022

[129] Salman U. ve Sarı İ. (2020). Beden Eđitimi Öđretmenleri, Öđrenciler Ve Ailelerin Bakıř Açıřı İle Sađlık İlgili Fiziksel Uygunluk Karnesi: Yeni Bir Uygulama. Spormetre The Journal of Physical Education and Sport Sciences, 18(2), 77-103. DOI: 10.33689/spormetre.589153.

[30] Hynes, M., Portsmore, M., Dare, E., Milto, E., Rogers, C., Hammer, D. & Carberry, A. (2011). Infusing engineering design into high school STEM courses. <http://ncete.org/flash/pdfs/Infusing%20Engineering%20Hynes>.Eriřim tarihi: 10.02.2017.

[31] Brunsell, E. (2012). The engineering desing process. Brunsell, E. (Ed.) Integrating engineering + science in your classroom. Arlington, Virginia: National Science Teacher Association[NSTA] Press.

[32] Hynes, M., Portsmore, M., Dare, E., Milto, E., Rogers, C., Hammer, D. & Carberry, A. (2011). Infusing engineering design into high school STEM courses. <http://ncete.org/flash/pdfs/Infusing%20Engineering%20Hynes>.Eriřim tarihi: 10.02.2017.

[30] Brunsell, E. (2012). The engineering desing process. Brunsell, E. (Ed.) Integrating engineering + science in your classroom. Arlington, Virginia: National Science Teacher Association[NSTA] Press.

