

T.C.

SÜLEYMAN DEMİREL ÜNİVERSİTESİ

ELEKTRİK VE ELEKTRONİK MÜHENDİSLİĞİ BÖLÜMÜ

ELEKTRONİK VE HABERLEŞME MÜHENDİSLİĞİ PROGRAMI



EHB-492 TASARIM PROJESİ

**OPENCV İLE İNSAN HAREKETİ İZLEYEN MOBİL
ROBOT TASARIMI**

Bitirme Ödevi Danışmanı Dr. Öğr. Üyesi Umut Tilki

1611008001 Süleyman Samet Kaya

1711008031 Anıl Mirap

1711008043 Mehmet Emin Han Bıyık

1711008062 Ahmet Kazak

BİTİRME ÖDEVİ/TASARIM PROJESİ GRUP LİDERLİĞİ TABLOSU

Öğrencinin Adı Soyadı	Tarih
Mehmet Emin Han Bıyık	15.02.2021 - 21.03.2021
Ahmet Kazak	21.03.2021 - 25.04.2021
Süleyman Samet Kaya	25.04.2021 - 30.05.2021
Anıl Mirap	30.05.2021 - 04.07.2021

BİTİRME ÖDEVİ/TASARIM PROJESİ GÖREV DAĞILIMI TABLOSU

Öğrencinin Adı Soyadı	Görevi	Tarih
Mehmet Emin Han Bıyık	Literatür Taraması, Teorik Araştırma, Test Uygulamaları	15.02.2021 - 04.07.2021
Ahmet Kazak	Literatür Taraması, Teorik Araştırma, Test Uygulamaları	15.02.2021 - 04.07.2021
Süleyman Samet Kaya	Literatür Taraması, Teorik Araştırma, Test Uygulamaları	15.02.2021 - 04.07.2021
Anıl Mirap	Literatür Taraması, Teorik Araştırma, Test Uygulamaları	15.02.2021 - 04.07.2021

LİSANS BİTİRME ÖDEVİ SONUÇ FORMU

..... tarafından
danışmanlığında hazırlanan “.....” başlıklı lisans
bitirme ödevi tarafımızdan incelenmiş, kapsamı ve niteliği açısından, ... / ... / 20...
tarihinde bir Lisans Bitirme Ödevi olarak kabul edilmiş / edilmemiştir.

Unvan Ad SOYAD

Danışman

Unvan Ad SOYAD

Jüri Üyesi

Unvan Ad SOYAD

Jüri Üyesi

Unvan Ad SOYAD

Bölüm Başkanı

Bu çalışmada, başka kaynaklardan yapılan tüm alıntıların, ilgili kaynaklar referans gösterilerek açıkça belirtildiğini, alıntılar dışındaki bölümlerin, özellikle projenin ana konusunu oluşturan teorik çalışmaların ve yazılım/donanımın benim tarafımdan tüm akademik kurallara ve etik ilkelere bağlı kalınarak yapıldığını bildiririm.

Isparta,/.... / 20...

Ad Soyad :

İmza :

ÖZET

Günümüzde görüntü işlemeye dayalı mobil robotlar büyük ölçüde gelişmişlerdir ve gelişmeye devam etmektedirler. Yapılacak olan bu projenin amacının, OpenCV görüntü işleme kütüphanesini kullanarak istenilen hedefler doğrultusunda insan hareketlerini izleyen ve belli amaçlara hizmet eden bir mobil robot tasarlanması olduğu hedeflenmiştir.

ABSTRACT

Today, mobile robots based on image processing are greatly improved and continue to improve. The aim of this project is to design a mobile robot that tracks human movements in line with the desired targets and serves certain purposes using the OpenCV image processing library.

İÇİNDEKİLER

ÖZET	5
ABSTRACT	6
ŞEKİLLER DİZİNİ	8
TABLOLAR DİZİNİ	9
1. GİRİŞ	10
2. AMAÇ	11
3. MATERİYAL, METOT VE YÖNTEM	12
3.1. MediaPipe	12
3.2. OpenCV	14
3.3. Kullanılan Malzemeler	15
3.3.1. Raspberry Pi	16
3.3.2. Raspberry Pi Kamera Modülü	17
3.3.3. BTS7960B Motor Sürücü Kartı	17
3.3.4. Rpm Redüktörlü DC Motor 12V 35mm 111 Rpm	18
3.3.5. Güç Kaynağı 10050 mAh	19
3.3.6. Li-Po (Lithium Polymer) Pil	19
4. GELİŞTİRİLEN TASARIM	20
5. SONUÇLAR	25
6. TARTIŞMA	31
7. KAYNAKÇA	32

ŞEKİLLER DİZİNİ

- Şekil 3.1.1. Vitruvius adam - Leonardo da Vinci.
- Şekil 3.1.2. Postür landmark noktaları.
- Şekil 3.3.1.1. Raspberry pi 4.
- Şekil 3.3.2.1. Raspberry pi kamera modülü.
- Şekil 3.3.3.1. BTS 79060b motor sürücü kartı.
- Şekil 3.3.4.1. DC motor.
- Şekil 3.3.5.1. 10050 mAh güç kaynağı.
- Şekil 3.3.6.1. Lipo pil.
- Şekil 4.1. Akiş diyagramı.
- Şekil 4.2. Görüntü matrisinin izahı için temsili x-y düzlemi.
- Şekil 4.3. Şasının yandan görünümü.
- Şekil 4.4. Şasının üstten görünümü.
- Şekil 4.5. Tasarlanan 3 boyutlu mobil robot.
- Şekil 4.6. Mobil robotun üstten görüntüsü.
- Şekil 4.7. Mobil robotun sağdan görüntüsü.
- Şekil 4.8. Alt plakanın izgaralı tasarıma geçişti.
- Şekil 4.9. Üst plakanın hafif tasarıma geçişti.
- Şekil 5.1. 11 ve 12 numaralı noktaların arasındaki mesafe 140px'den büyük.
- Şekil 5.2. Motorların hareket etmemeye terminal çıktısı.
- Şekil 5.3. 1 ve 12 numaralı noktaların arasındaki mesafe 60 ile 140 px aralığında.
- Şekil 5.4. Motorlar ileri yönde hareket etmeye ekran çıktısı.
- Şekil 5.5. Landmark 12'nin x düzlemindeki konumu 30'dan küçük.
- Şekil 5.6. Motorların sola döndüğü terminal çıktısı.
- Şekil 5.7. Landmark 12'nin x düzlemindeki konumu 400'den büyük.
- Şekil 5.8. Motorların sağa döndüğü terminal çıktısı.

TABLOLAR DİZİNİ

Tablo 3.3.1. Malzeme listesi.

Tablo 3.3.4.1. DC motor özellikleri.

Tablo 5.1. Yazılımın test edildiği ve sorunsuz çalıştırıldığı bilgisayar ve donanım özellikleri.

1. GİRİŞ

Görüntü işleme kullanarak insanı izleyen ve taşıma görevi gören mobil robot tasarlanması amaçlanmıştır. Mobil robot; uzaktan kumanda ile, uydu üzerinden yönetilebilen veya yapay zeka ile kendisini yöneten hareket kabiliyetine sahip robotlardır. Görüntü işleme kısmında OpenCV kaynak kütüphanesi kullanılması hedeflenmiştir. İnsanların ihtiyaçlarına göre kullanım amacı esneyebilir ve her türlü ihtiyaca cevap verebilir nitelikte olacaktır.

OpenCV, gerçek zamanlı görüntü işleme uygulamaları için geliştirilmiş ve kullanılabilen açık kaynaklı bir kod kütüphanesidir. Genel olarak ve en sağlıklı çalıştığı programlama dili Python'dur. Python üzerinden geliştirilen projelerde sıkılıkla kullanılmaktadır. İnsan hareketi izleyen mobil robot için OpenCV kaynak kütüphanesini Python programlama dilinde kullanmak amaçlanmıştır.

Python programlama dili dışında da OpenCV kaynak kütüphanesinin kullanılabileceği diller mevcuttur fakat projede kullanılması öngörülen kontrol kartı Raspberry Pi ile en sağlıklı çalışan programlama dili Python'dur.

Günlük hayatta fiziksel güç gerektiren birçok durum mevcuttur. Proje ile bu durumların üstesinden rahat ve kolay bir şekilde gelmek temel hedefdir. Bu hedefleri sağlamak adına robotun insanı ve insan hareketlerini izlemesine ve ona göre hareket etmesine karar verilmiştir.

Amaç kısmında projenin amacının detaylı anlatımı, materyal ve metodlar kısmında projenin tamamlanması ve üretilmesi için gerekli veriler, kaynaklar ve araçların tanımları, yapılan çalışmalar kısmında proje ile ilgili başlanılan ve elde edilen verilerin sunumu, sonuçlar kısmında mevcut ilerleyiş, kaydedilen ilerleme, eksiklikler ve problemlerin anlatımı ve son olarak kaynakça kısmında, raporda kullanılan ve belirtilen kaynaklar mevcuttur.

2. AMAÇ

Projenin amacı, insan hareketlerini gerçek zamanlı olarak takip ederek işlediği veri ile otonom biçimde hareket eden bir mobil robotun tasarlanması ve üretilmesidir. Bu tasarım, itme, çekme, döndürme, kaldırma, taşıma işlerinin insan tarafından yapılması gereksinimini ortadan kaldırmak için yapılacaktır.

İnsanların zaman zaman çeşitli işlerde fiziksel olarak yardıma ihtiyacı olmaktadır. Projenin, bu ihtiyaçlara belli miktarda karşılık vermek amacıyla gerçekleştirilmesi temel amaçlardan birisidir. Robotun, insan hareketlerini otomatik olarak takip etmesi ve insan hareketlerine göre komut alması, insanın işlerini çok daha kolaylaştıracaktır.

Robotun kullanım alanı çok genişdir. Bir mağaza personeli tarafından, ürünlerin taşınması amacıyla kullanılabilir, müşteri yoğunluğu fazla olan bir restoranda garson, elindeki yemekleri robota yerleştirerek servisini hızlandırabilir, marketlerde gerek müşteriler tarafından gerekse personel tarafından kullanılabilir, bir noktadan bir noktaya taşınacak ve yaya olarak gidilecek bir yere, robot sayesinde daha az efor harcayarak varılabilir. Kullanım alanları daha da çeşitlendirilebilir ve artabilir.

Fiziksel olarak gerçekleştirilecek olan robotun, ergonomik olarak ve maddi olarak en uygun şekilde üretilmesi de projenin diğer amaçlarından birisidir. Kullanılan malzemelerin ağırlığı, maliyeti vb. durumların en uygun şekilde belirlenmesi projenin önemli amaçları arasındadır.

3. MATERİYAL, METOT VE YÖNTEM

Bu bölümde, projede kullanılan yazılım - donanım bileşenleri ve teknolojilerden bahsedilmiştir.

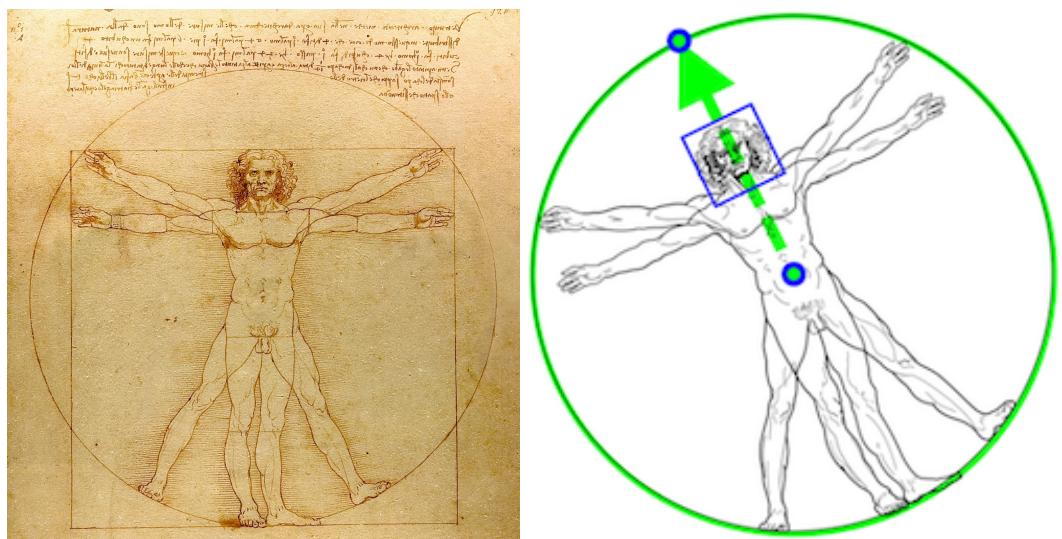
3.1. MediaPipe

MediaPipe, Alphabet Inc. iştiraki Google LLC tarafından geliştirilmiş, cross-platform olarak birden fazla programlama dili ile kullanılabilen, açık kaynaklı ve ücretsiz bir önceden modellenmiş, canlı makine öğrenmesi uygulamaları geliştirme arayüzüdür.

Farklı kullanım alanlarına göre tercih edilmek üzere yüz tespiti, yüz örnekleme, iris takibi, jest takibi, postür tahmini, tüm vücut özniteliklerinin bütünsel tahmini (yüz + iris + jest + postür), saç segmentasyonu, nesne takibi, çerçeve takibi, anlık hareket takibi, objectron, şablon bazlı öznitelik eşleme gibi çeşitli önceden eğitilmiş canlı makine öğrenmesi modelleri mevcuttur.

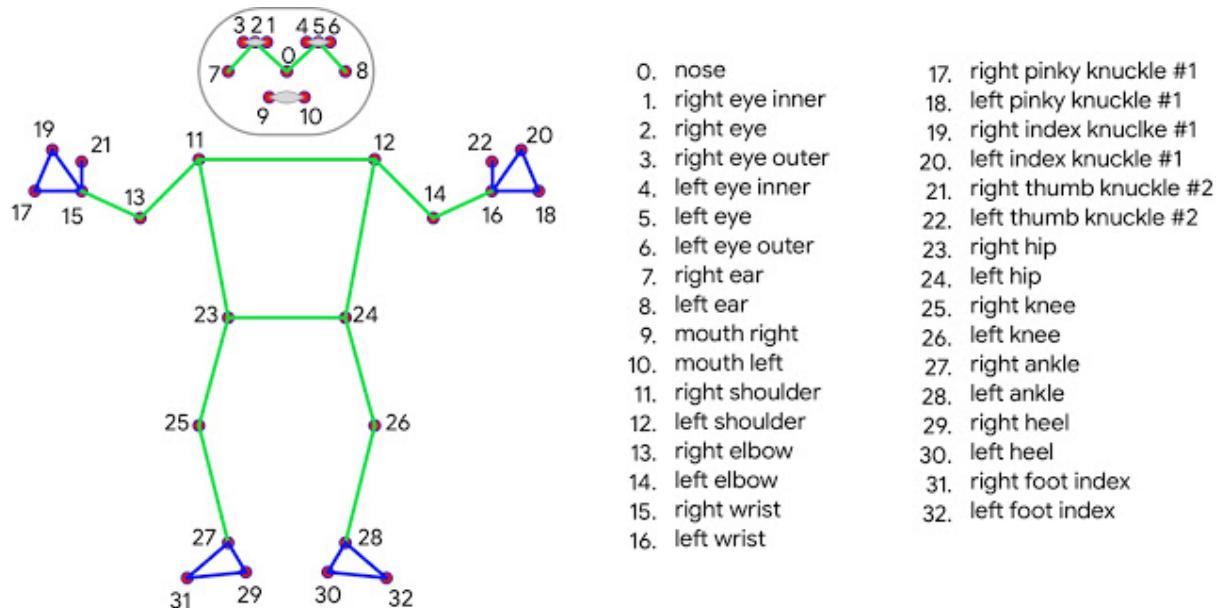
OpenCV ile İnsan Hareketleri Takip Eden Mobil Robot projesinde kullanılmak üzere insan vücudunun tespiti ve takibi için MediaPipe postür tahmini modeli OpenCV görüntü işleme kütüphanesi üzerinde çalıştırılmıştır. Bu model, MediaPipe'ın diğer çözüm modellerinde de etkili olduğu kanıtlanmış iki adımlı dedektör-izleyici makine öğrenmesi sinir ağını kullanır. Bir dedektör kullanarak, sinir ağı önce çerçeveye içinde kişi/postür ilgi bölgesini (ROI) bulur. İzleyici daha sonra ROI kırılmış çerçevesini girdi olarak kullanarak ROI içindeki postür yer işaretlerini tahmin eder. Video kullanım durumları için dedektörün yalnızca gerektiği gibi, yani ilk kare için ve izleyici önceki karede vücut postürü varlığını artık tanımlayamadığında çağrırlar. Diğer kareler için işlem hattı, ROI'yi önceki karenin postür işaretlerinden türetir [1].

Postür tahmin modeli, OpenCV'deki yerleşik Haar Cascade ve HOG öznitelik çıkarma modellerine göre düşük kapasiteli donanımlar üzerinde çok daha hızlı ve yüksek performanslı çalıştığı için nihai uygulamada tercih edilmiştir.



Şekil 3.1.1. Vitruvius Adam - Leonardo da Vinci

Dedektör, insan vücudunun merkezini, dönüşünü ve ölçüğünü bir daire olarak kesin olarak tanımlayan iki ek sanal anahtar noktası açıkça öngörür. Leonardo da Vinci'nin Vitruvius Adam'ından esinlenerek, bir kişinin kalçalarının orta noktasını, tüm kişiyi çevreleyen bir dairenin yarıçapını ve omuz ile kalça orta noktalarını birleştiren çizginin eğim açısını tahmin eder. Yapılan tahmine göre tahmin edilen insan postüründe anahat noktası canlı görüntü üzerinde işaretlenmektedir. Postür üzerine yerleştirilen ana hat noktaları şekil 3.1.2.'de gösterilmiştir.



Şekil 3.1.2. Postür landmark noktaları

3.2. OpenCV

OpenCV (Open Source Computer Vision Library), açık kaynaklı bir bilgisayarlı görüp makine öğrenimi yazılım kitaplığıdır. OpenCV, bilgisayarlı görüp uygulamaları için ortak bir altyapı sağlamak ve ticari ürünlerde makine algısının kullanımını hızlandırmak amacıyla oluşturulmuştur. BSD lisanslı bir ürün olan OpenCV, işletmelerin kodu kullanmasını ve değiştirmesini kolaylaştırır.

Kütüphane, hem klasik hem de son teknoloji bilgisayarlı görme ve makine öğrenimi algoritmalarından oluşan kapsamlı bir set içeren 2500'den fazla optimize edilmiş algoritmeye sahiptir. Bu algoritmalar, yüzleri algılamak ve tanımak, nesneleri tanımlamak, videolarda insan eylemlerini sınıflandırmak, kamera hareketlerini izlemek, hareketli nesneleri izlemek, nesnelerin 3B modellerini çıkarmak, stereo kameralardan 3B nokta bulutları üretmek, yüksek çözünürlük elde etmek için görüntüleri bir araya getirmek için kullanılabilir. Tüm bir sahnenin görüntüsü, bir görüntü veritabanından benzer görüntüler bulma, flaş kullanılarak çekilen görüntülerden kırmızı gözleri kaldırma göz hareketlerini takip etme, manzarayı tanıma ve artırılmış gerçeklikle kaplamak için işaretler oluşturma, vb. uygulamaları gerçeklemek için OpenCV'nin 47 binden fazla kullanıcısı vardır. Topluluk ve 18 milyonu aşan tahmini indirme sayısına sahip kütüphane, şirketlerde, araştırma gruplarında ve devlet kurumları tarafından yaygın olarak kullanılmaktadır.

Kütüphaneyi kullanan Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota gibi köklü şirketlerin yanı sıra, OpenCV'yi yoğun olarak kullanan Applied Minds, VideoSurf ve Zeitera gibi birçok startup bulunmaktadır. OpenCV'nin konuşlandırılan kullanıcıları, sokak görünümü görüntülerini bir araya getirmek, İsrail'deki güvenlik videolarındaki izinsiz girişleri tespit etmek, Çin'deki maden ekipmanlarını izlemek, robotların Willow Garage'da gezinmesine ve nesneleri almasına yardımcı olmak, Avrupa'da yüzme havuzunda boğulma kazalarını tespit etmek, interaktif sanat çalışıtmak gibi geniş bir aralığı kapsar. OpenCV kullanılarak İspanya, New York ve Türkiye'de doğal afet enkaz sahaları kontrol edilmekte, Japonya'da hızlı yüz algılamaya kadar dünyanın dört bir yanındaki fabrikalardaki ürünler üzerindeki etiketleri incelenmektedir.

OpenCV, C++, Python, Java ve MATLAB arayüzlerine sahiptir ve Windows, Linux, Android ve Mac OS'yi destekler. OpenCV, çoğunlukla gerçek zamanlı vizyon uygulamalarına yönelik ve mevcut olduğunda MMX ve SSE talimatlarından yararlanır. Tam özellikli bir CUDA ve OpenCL arayüzleri şu anda aktif olarak geliştirilmektedir. 500'den fazla algoritma ve bu algoritmaları oluşturan veya destekleyen işlev sayısı yaklaşık 10 kat daha fazladır. OpenCV yerel olarak C++ ile yazılmıştır ve STL kapsayıcıları ile sorunsuz çalışan şablonlu bir arayüze sahiptir [2].

3.3. Kullanılan Malzemeler

Proje için kullanılacak malzemeler seçilmiştir. Tablo 3.3.1.'de malzeme listesi sıralanmıştır.

Tablo 3.3.1. Malzeme Listesi

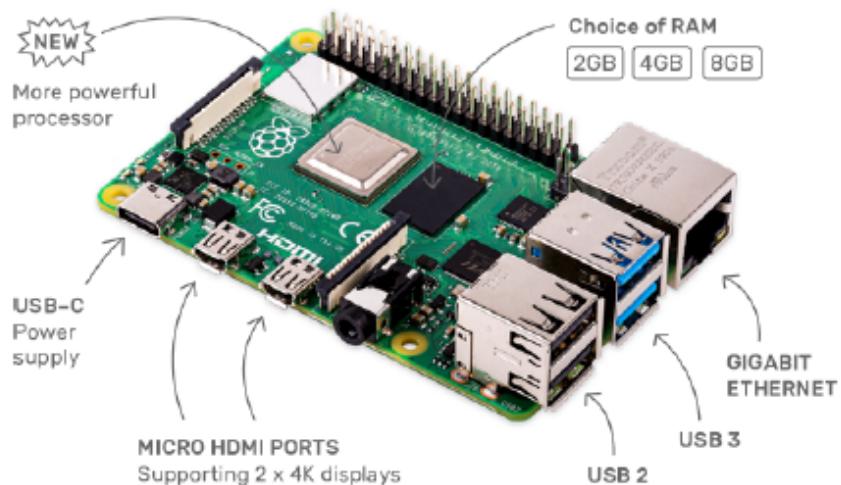
Adet	Malzemeler
1	Raspberry Pi 4b 8 GB
1	Raspberry Pi Kamera Modülü
2	BTS7960B Motor Sürücü Kartı
2	11.1 V 1300mAh 25C 3S Lipo pil
1	10050 mAh Güç Kaynağı
1	Breadboard
8	Krokodil Kablo
4	12V 35 mm 111 rpm DC Motor
10+	Erkek-Dışı Jumper Disi-Dışı Jumper
4	Ø88 Arazi Tekerleği
2	Alüminyum Şasi (2mm/43.5cm/50cm)
5	10mm M10 Civata

3.3.1. Raspberry Pi

Açık kaynak kodlu “Bilgisayar Görü” kütüphanesi olan , görüntü işleme ile ilgili temel ve ileri seviyedeki fonksiyonları optimize edilmiş, OpenCV kullanılarak yapılacak görüntü işleme

kodları ile mobil robotun motor ve aksamlarını harekete geçirecek geliştirme kartı araştırması yapılmıştır. Gömülü geliştirme kartları olan Jetson Nano, Raspberry Pi, Arduino ve Google Coral arasında seçim yapılmıştır. Arduino görüntü işleme konusunda yeterli olmadığından projemize uygun görülmemiştir. Görüntü işleme konusunda iyi performans sağlayan ve diğer geliştirme kartlarına göre ekonomik olan Raspberry Pi 4 tercih edilmiştir. Raspberry Pi 4 modelinin özellikleri şunlardır:

- 1.5 GHz dört çekirdekli ARM Cortex-A72 CPU
- 8GB LPDDR4 RAM SKU
- VideoCore VI Grafikleri
- 4kp60 HEVC video
- Gigabit Ethernet
- $2 \times$ USB 3.0 ve $2 \times$ USB 2.0 bağlantı noktaları
- $2 \times$ mikro HDMI bağlantı noktası ($1 \times 4K@60Hz$ veya $2 \times 4K@30Hz$)
- 5V/3A çalışmasını destekleyen USB-C güç girişi



Şekil 3.3.1.1. Raspberry Pi 4.

3.3.2. Raspberry Pi Kamera Modülü

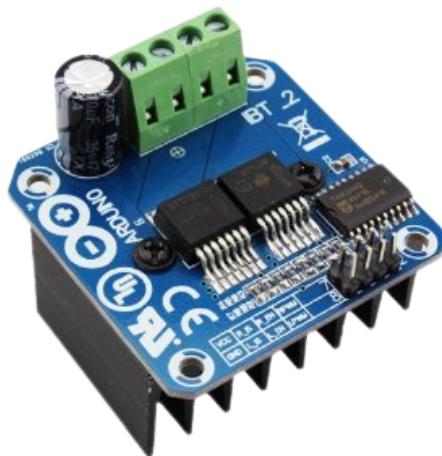
5 MP çözünürlüklü kamera sabit bir lense sahiptir. Modül Raspberry Pi ile 1080P Full Hd ve 30 Fps görüntü almamızı sağlamaktadır. Kamera modülünün boyutları 25x20x9 mm'dir. Raspberry Pi 3 ve 4 uyumlu bir modüldür. Şekil 3.3.2.1.'de kamera modülü gösterilmiştir. Raspberry Pi ile uyumlu olan tüm işletim sistemlerinde çalışmaktadır.



Şekil 3.3.2.1. Raspberry Pi kamera modülü.

3.3.3. BTS7960B Motor Sürücü Kartı

BTS7960B motor sürücü kartı arazi robotları gibi yüksek akım gerektiren robotlarda kullanılır. 20 Ampere kadar akım çekebilen motorlar için kullanılabilen bir motor sürücü kartıdır. Raspberry Pi gibi geliştirme kartları ile uyumlu bir şekilde çalışır ve kullanılır. Yüksek akım çeken motorları kontrol ettiği için ısınma sorununa karşı çözüm olarak üzerinde pasif soğutucu bulunur. Pasif soğutucu soğutma işlemi için yeterlidir başka soğutucuya gerek yoktur. Şekil 3.3.3.1.’de motor sürücü kartı gösterilmiştir.



Şekil 3.3.3.1. BTS79060B motor sürücü kartı.

3.3.4. Rpm Redüktörlü DC Motor 12V 35mm 111 Rpm

Elektrik enerjisini mekanik enerjisine çeviren makinelere doğru akım motor denir. 12 volt 100 rpm 35mm ve 13 kgxcm zorlanma torkuna sahip redüktörlü DC motor proje için yeterli torku sağlamakla kalmayı istenilen 5 km/h hızlarına kolaylıkla ulaşabilmektedir. Tablo 3.3.4.1'de

projede kullandığımız motorun özellikleri gösterilmiştir. Şekil 3.3.4.1.'de DC motor gösterilmiştir.

Tablo 3.3.4.1. DC motor özellikleri.

Çalışma Voltajı	12 V _{DC}
Hız	111 rpm
Boşta Çektiği Akım	190mA + 40mA
Yükleme Torku	3.4kg x cm
Zorlama Torku	13 kg x m
Mil Çapı	6 mm D



Şekil 3.3.4.1. DC motor.

3.3.5. Güç Kaynağı 10050 mAh

Duracell 10050 mAh güç kaynağı yüksek kapasitesi ile 18 Wattlık iki çıkışlı bulunmaktadır. Öncelikli şarj özelliği ile duvar prizinden şarj olurken öncelikli olarak

enerjiyi çıkışlarına ulaştırma ve enerji kaybını minimuma indirir. Şekil 3.3.5.1.'de güç kaynağı gösterilmiştir.



Şekil 3.3.5.1. 10050 mAh güç kaynağı.

3.3.6. Li-Po (Lithium Polymer) Pil

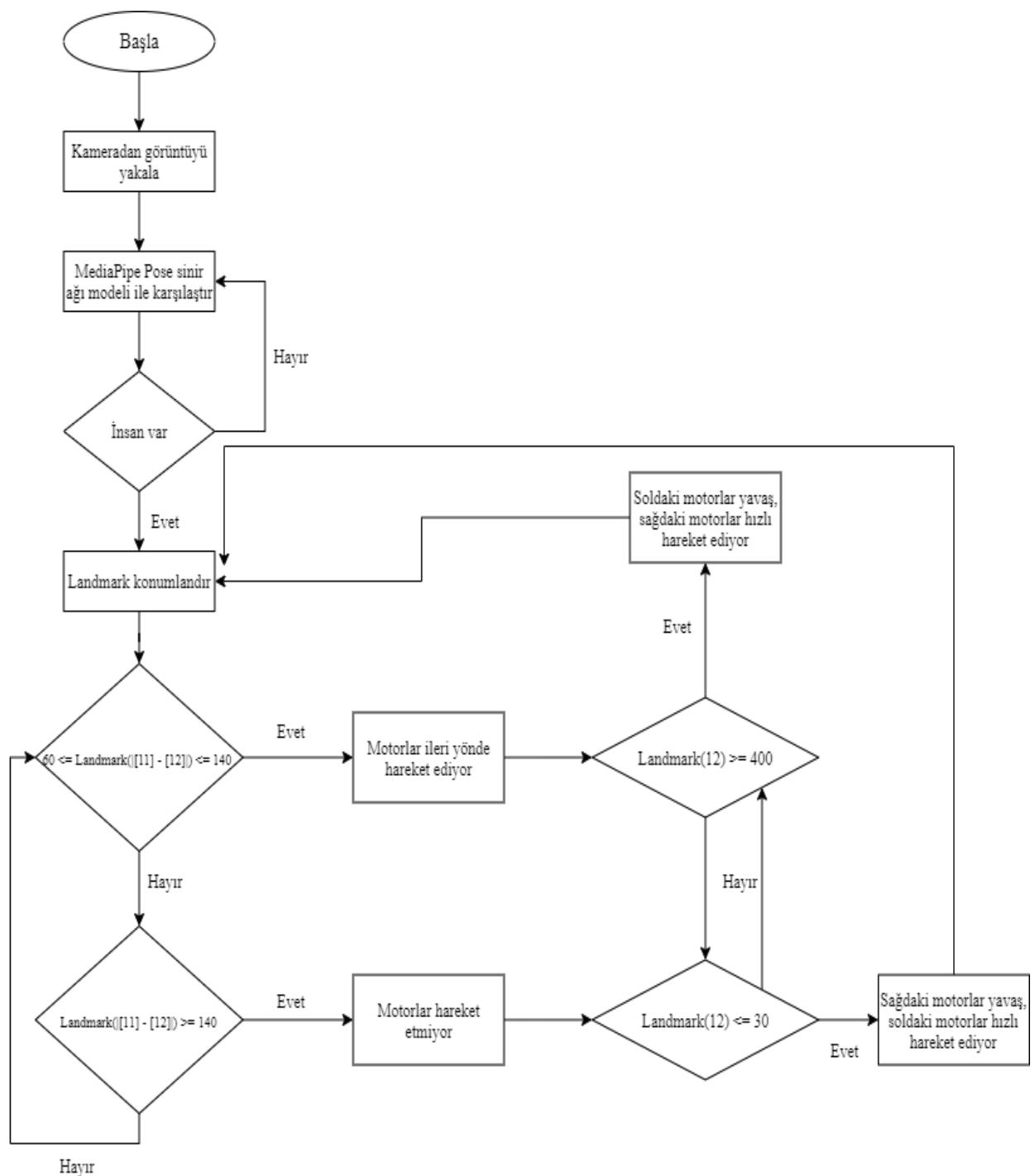
Lityum-polimer batarya sıvı elektrolit yerine polimer elektrolit kullanılan güç kaynaklarıdır. Yeniden şarj edilebilir. LiPo piller hücre sayılarına göre çeşitlendirilmektedir. Her hücre 3.7 V gerilim olarak hesaplanır. Projede kullanılan 3S LiPo pil 11.1 V gerilim değerine denk gelmektedir. Şekilde 3.3.5.1.'de kullanılan LiPo pil gösterilmiştir. Projede kullandığımızı yüksek akım çeken motorlar için düşük ağırlıkta uygun pillerdir.[3]



Şekil 3.3.6.1. LiPo pil.

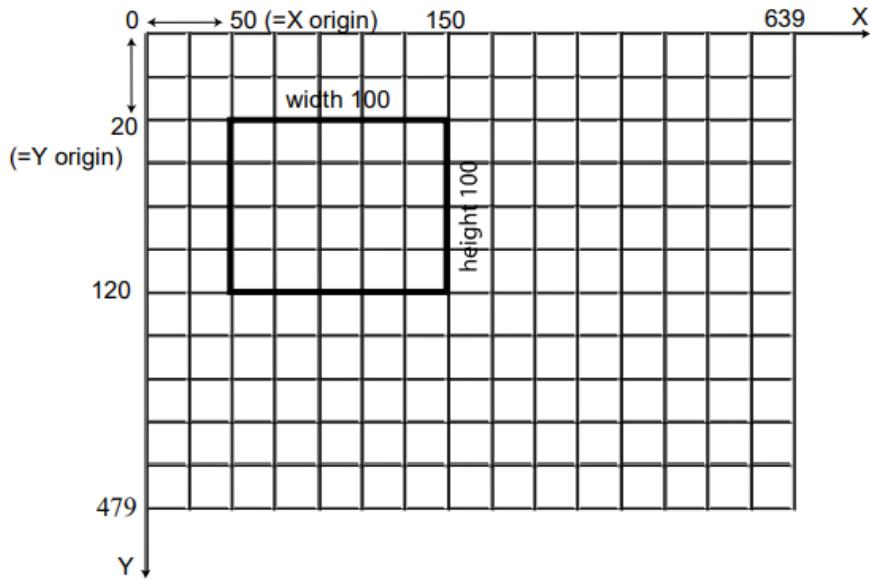
4. GELİŞTİRİLEN TASARIM

OpenCV ile İnsan Hareketleri İzleyen Mobil Robot Tasarımı projesi için geliştirilen mobil robotun program akış diyagramı şekil 4.1.'de gösterilmiştir.



Şekil 4.1. Akış diyagramı.

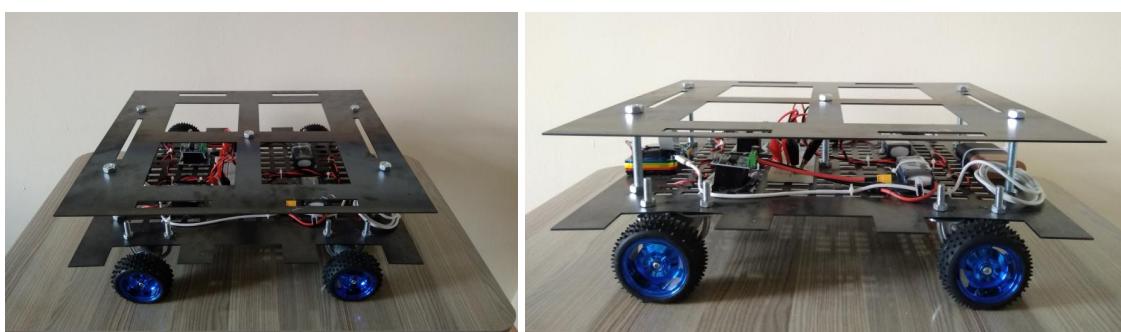
Şekil 4.1'de belirtilen Landmark noktalarının görüntü matrisi üzerindeki piksel konumları şekil 4.2.'deki örnek X-Y düzleminde görüleceği üzere sol üst köşe orijin noktası olacak şekilde belirlenmektedir.



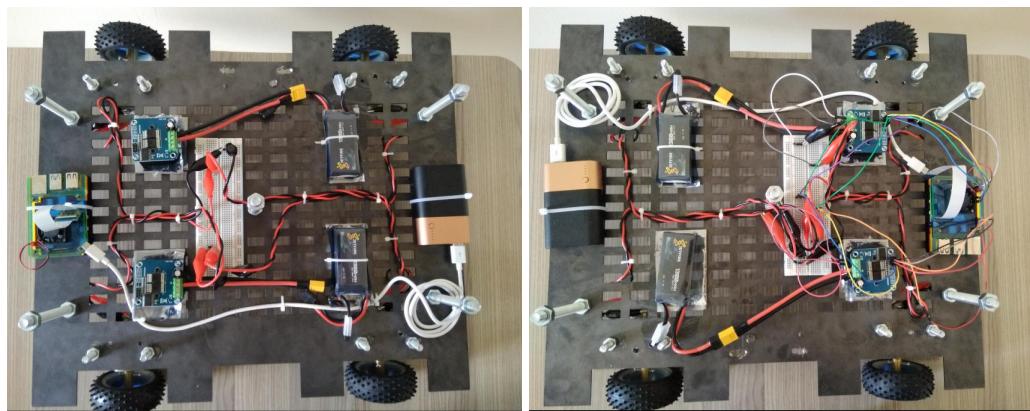
Şekil 4.2. Görüntü matrisinin izahî için temsili X-Y düzlemi.

Program algoritmasını akış diyagramı üzerinden kısaca özetlemek gerekirse; şekil 3.1.2'de gösterilen landmark noktalarından 11. ve 12. landmark noktaları arasındaki dinamik mesafe farkı ve 12. landmark noktasının genel görüntü matrisi üzerindeki pozisyonuna göre mobil robot görüntündeki insana yaklaşmakta, sağ ve sol yöndeki harekete göre insanı takip etmektedir.

Mobil robotun fiziksel dizaynında, elektronik aksamlar ve motorlar, şekil 4.3. ve şekil 4.4.'te gösterildiği gibi, robotun alt kısmında kalan iki alüminyum plaka arasına yerleştirilecek şekilde tasarlanmıştır. Bu iki plakanın üzerinde kalan kısım, taşınacak yük için ayrılan haznenin yerleştirileceği tabladır.

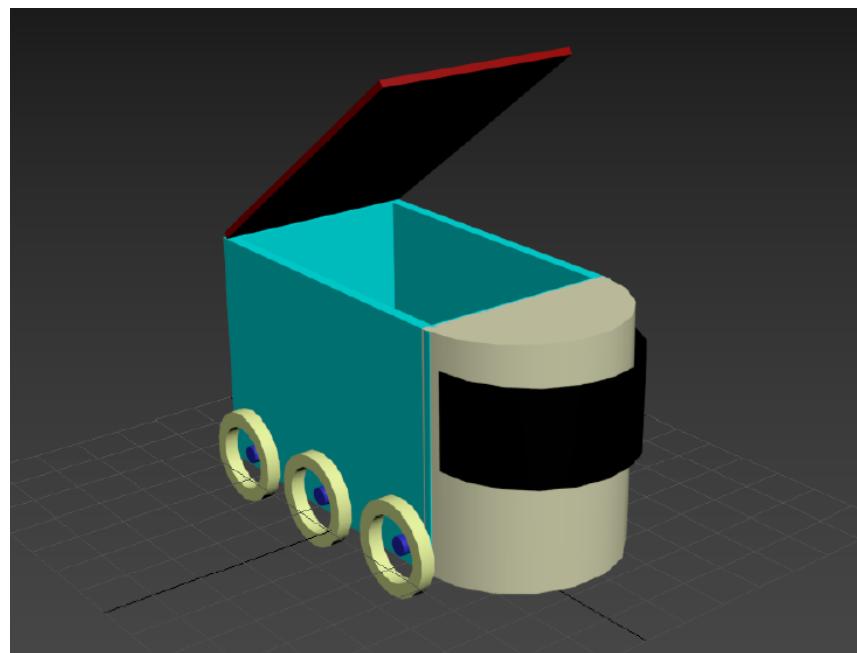


Şekil 4.3. Şasının yandan görünümü.



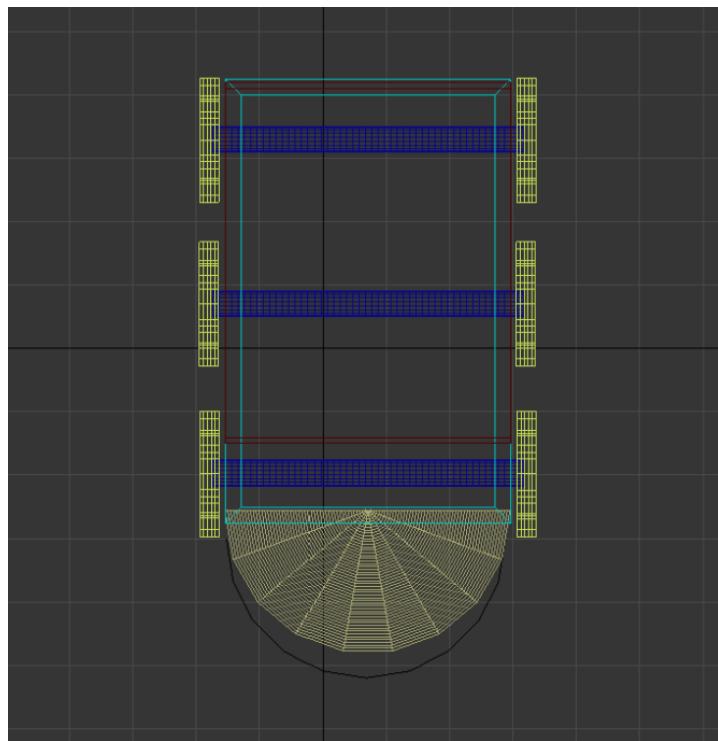
Şekil 4.4. Şasının üstten görünümü.

Projenin fiziksel tasarımlı için iki farklı 3 boyutlu tasarım programı düşünülmüş, sonucunda 3Ds Max görselleştirme, animasyon ve modelleme programı seçilmiştir. 3Ds Max programında, mobil robotun planlanan son hali 3 boyutlu olarak tasarlanmıştır. Şekil 4.5.'de tasarlanan robot gösterilmiştir.

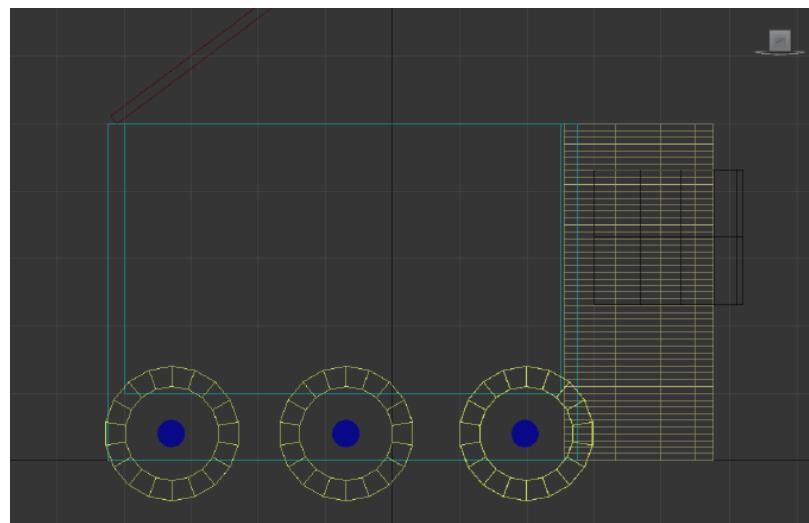


Şekil 4.5. Tasarlanan 3 boyutlu mobil robot.

Tasarımın hem üstten hem de sağdan olan görüntüsü şekil 4.6. ve şekil 4.7.'de gösterilmiştir.



Şekil 4.6. Mobil robotun üstten görüntüsü.

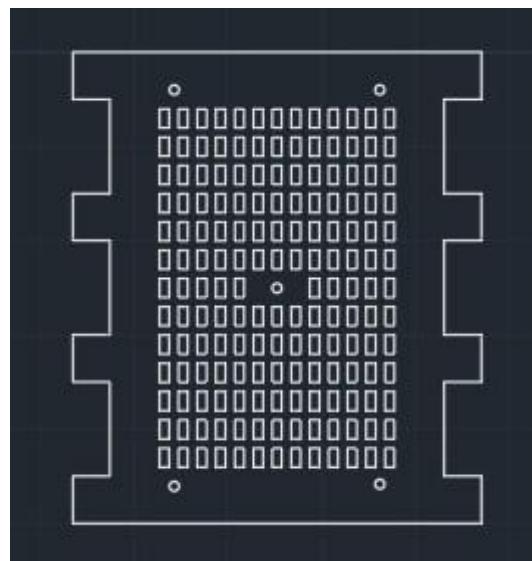


Şekil 4.7. Mobil robotun sağdan görüntüsü.

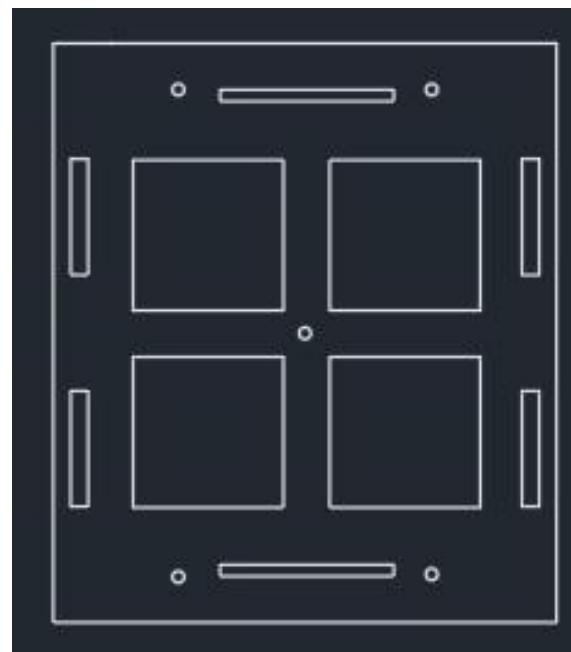
Mobil robot, 3Ds Max programında tasarlandığı ölçülere uygun olarak gerçeklemek üzere, robotun yapısında kullanılacak malzemeler belirlenmiştir. Robotun alt haznesi iki adet alüminyum plakadan, üst haznesi de pleksiglastan yapılacak şekilde planlanmıştır. 2 mm kalınlığında, 43.5 cm genişliğinde, 50 cm uzunluğunda olacak şekilde 2 adet alüminyumdan, 40 cm yüksekliğinde, 43.5 cm genişliğinde ve 50 cm uzunluğunda olacak şekilde 1 adet

pleksiglasdan oluşacak şekilde tasarlanmıştır. Alüminyum plakaların bir tanesi yaklaşık 3 kg olduğundan dolayı, üzerinde bir takım değişiklikler yapılmak zorunda kalınmıştır. Alt plakanın izgara yapılı olması, üst plakadan da büyük parçaların kesilip çıkarılması kararlaştırılmış, böylelikle toplamda yaklaşık 2.5 kg hafifletme işlemi yapılması kararlaştırılmıştır.

Alt iki plaka, revizyon yapılmak üzere AutoCAD programında yeniden çizilmiştir. Alt plaka, şekil 4.8.’de gösterildiği üzere, izgaralı tasarıma geçiş yapmıştır. Üst plaka, şekil 4.9.’da gösterildiği üzere, üzerinden büyük parçalar çıkarılarak hafifletme işlemi görmüştür.



Şekil 4.8. Alt plakanın izgaralı tasarıma geçisi.



Şekil 4.9. Üst plakanın hafif tasarıma geçisi.

5. SONUÇLAR

OpenCV ile İnsan Hareketleri İzleyen Mobil Robot Tasarımı projesi, Raspberry Pi 4 üzerinden, kamera ile görüntü alarak, insan davranışına göre hareket edecek şekilde tasarlanmış, çalışma mantığı da bunun üzerine kurulmuş bir projedir. Projenin çalışma prensibi, hem motorları, hem de görüntü işleme işlemini Raspberry Pi 4 geliştirme kartı üzerinden yapmaktadır.

Proje geliştirilirken, ilk olarak bilgisayar üzerinden Python programlama dili ile OpenCV kütüphanesi test edilmiş, MediaPipe geliştirme arayüzü ile görüntü işleme uygulaması geliştirilmiştir. Bilgisayar üzerindeki testlerde, geliştirilen algoritma sorunsuz bir şekilde çalışmaktadır. Algoritma, iki omuz noktasından birisi referans alınarak, perspektife göre aralarındaki mesafenin değişimine göre çalışacak şekilde tasarlanmıştır. Takip edilecek insan, kameradan uzaklaşıkça, omuzlar arasındaki mesafe perspektife göre daralacağından, koşullu karar yapıları (if - else) ile kontrol edilmektedir. Kontrol sonucuna göre motorlara gerekli uyarının verilmesi kararlaştırılmıştır.

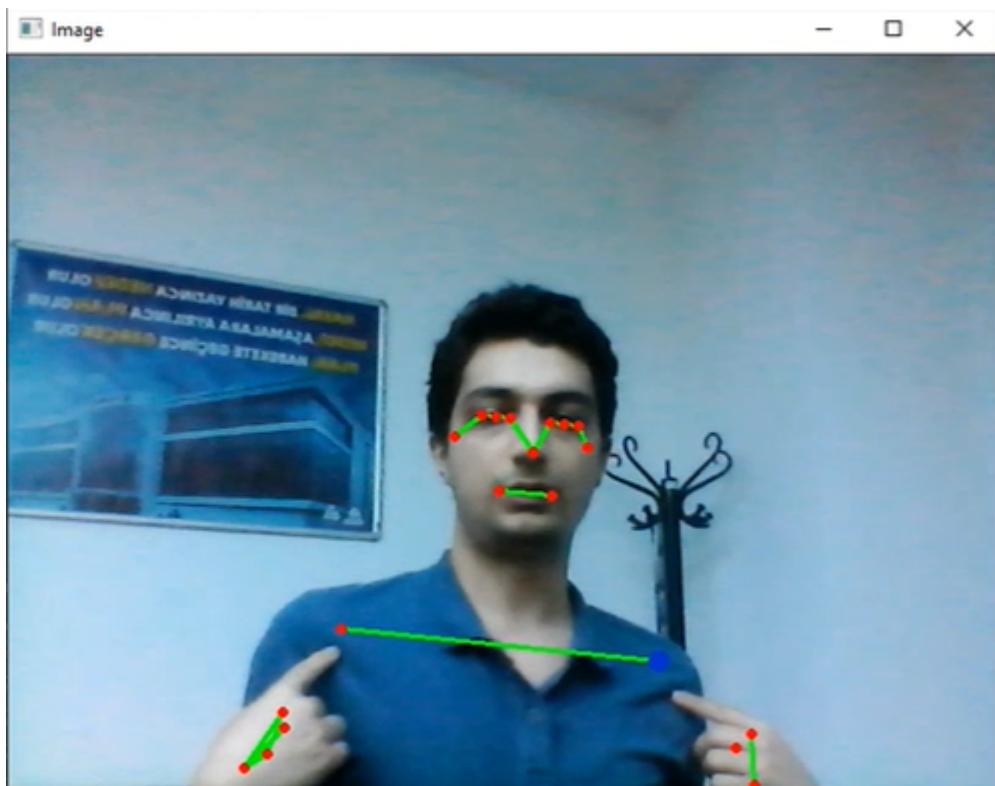
Tablo 5.1.'de donanım ve yazılım özellikleri verilen bilgisayarlar üzerinde sorunsuz bir şekilde çalışmakta olan görüntü işleme programı, Raspberry Pi 4 üzerinde sorunsuz bir şekilde çalışmakta olan motor kontrol programı ile birleştirilememektedir. Birleştirilememesinin sebebi, Raspberry Pi 4 geliştirme kartı içerisinde MediaPipe kurulumunun yapılamamasıdır. Raspberry Pi 4 içerisinde, gündelen eskiye doğru olmak üzere Python 3 versiyonları kurulmuş fakat MediaPipe çalıştırılamamıştır.

Tablo 5.1. Yazılımin test edildiği ve sorunsuz çalıştırıldığı bilgisayar ve donanım özellikleri

Üretici	Model	CPU	GPU	İşletim Sistemi
ASUSTek Computer Inc.	VivoBook 14 X405UQ	Intel® Core™ i5-7200U	NVIDIA GeForce 940MX	Microsoft Windows 10 Home SL Version 21H1
Monster Bilgisayar Teknoloji Ürünleri A.Ş.	Tulpar T7 V20.2	Intel® Core™ i7-10750H	NVIDIA GeForce RTX 2060	Microsoft Windows 10 Home SL Version 2004
Monster Bilgisayar Teknoloji Ürünleri A.Ş.	Abra A7 V12.2.1	Intel® Core™ i5-10300H	NVIDIA GeForce GTX 1650 Ti	Microsoft Windows 10 Pro Version 21H1

Dell Inc.	Inspiron 5558	Intel® Core™ i5-5200U	NVIDIA GeForce 920M	Microsoft Windows 10 Pro Version 1903
-----------	---------------	-----------------------	---------------------	---------------------------------------

Yazılımın test edildiği ve sorunsuz çalıştığı bilgisayarlarda geliştirilen programda, Raspberry Pi kamerası ile yakalanan insan postüründen 32 ana hat noktası işaretlenir. 32 ana hat noktasından 12 numaralı nokta landmark olarak seçilir. 12 numaralı landmark insan postüründe sol omuza denk gelmektedir. Geliştirilen programda mesafe sensörü kullanılmadan insanı takip edebilmesi için 11 ve 12 numaralı landmark arasındaki mesafe ile motorun hareket edip etmemesi sağlanmaktadır. 12 numaralı landmarkın sağ tarafa geçmesi ile sol motorlar hızlanır, sağ motorlar yavaşlar. 12 numaralı landmarkın sol tarafa geçmesi ile sol motorlar yavaşlar, sağ motorlar hızlanır. 11 numaralı landmark ve 12 numaralı noktanın arasındaki mesafenin mutlak değeri 140 piksel'den büyük ise motorlar hareket etmez. Şekil 5.1.'de motorların hareket etmediği iki nokta arasındaki mesafenin 140 piksel'den büyük olduğu durum gösterilmiştir.



Şekil 5.1. 11 ve 12 numaralı noktaların arasındaki mesafe 140px'den büyük.

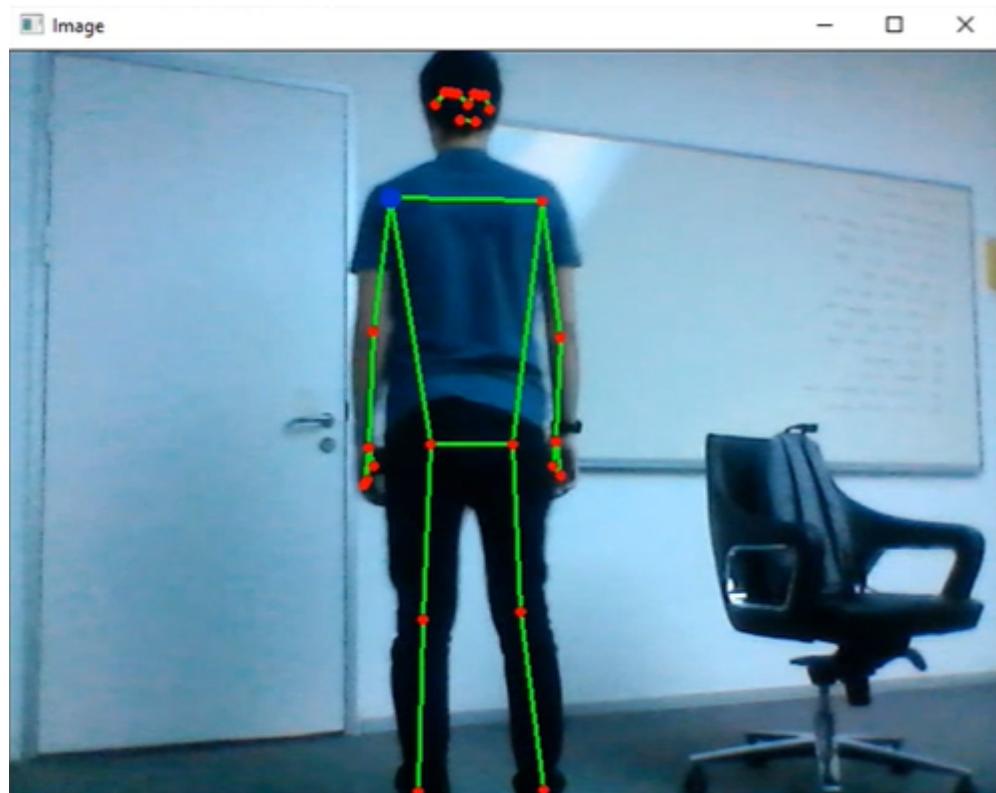
Motorların hareket etmemesi durumunun terminal çıktısı şekil 5.2.'de gösterilmiştir.

```
Run: PoseEstimation 0.1 ×
G ↑ Motorlar Hareket Etmiyor
D ↓ [13, 661, 669]
E ⌂ Motorlar Hareket Etmiyor
F ⌂ [13, 660, 673]
H ⌂ Motorlar Hareket Etmiyor
I ⌂ [13, 659, 673]
J ⌂ Motorlar Hareket Etmiyor
K ⌂ [13, 658, 677]
L ⌂ Motorlar Hareket Etmiyor
M ⌂ [13, 657, 671]

Run TODO Problems Terminal R Jobs Python Packages R Console Python Console
```

Şekil 5.2. Motorların hareket etmemesi terminal çıktısı.

11 ve 12 numaralı landmarklar arasındaki mesafenin mutlak değeri 60 ile 140 piksel aralığında olduğu zaman motorlar ileri yönde hareket etmektedir. Motorları ileri hareket ettiği görüntüsü Şekil 5.3.'de gösterilmiştir.



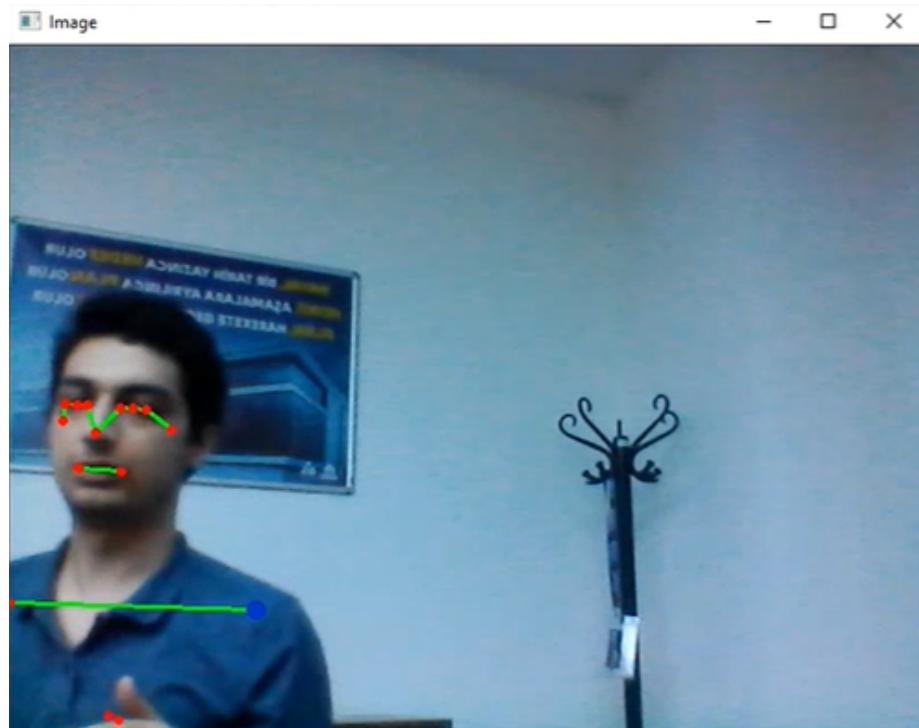
Şekil 5.3. 1 ve 12 numaralı noktaların arasındaki mesafe 60 ile 140 px aralığında.

Motorların ileri yönde hareket ettiği zaman terminal çıktısı Şekil 5.4.'de gösterilmiştir.

```
Motorlar İleri Yönde Hareket Ediyor  
[13, 247, 311]  
Motorlar İleri Yönde Hareket Ediyor  
[13, 248, 310]  
Motorlar İleri Yönde Hareket Ediyor  
[13, 248, 310]  
Motorlar İleri Yönde Hareket Ediyor  
[13, 248, 310]  
Motorlar İleri Yönde Hareket Ediyor  
[13, 248, 312]  
Motorlar İleri Yönde Hareket Ediyor  
[13, 247, 313]  
Motorlar İleri Yönde Hareket Ediyor
```

Şekil 5.4. Motorlar ileri yönde hareket etme ekran çıktısı.

11 ve 12 numaralı landmarklar arasındaki mesafenin mutlak değeri 60 ve 140 piksel arasında olduğu ve landmark 12'in şekil 5.5.'de gösterilen temsili X-Y düzleminin konumunun 30'dan az olduğu durumlarda soldaki motorlar yavaşlar, sağdaki motorlar hızlanır. Mobil robotumuz sola doğru hareket eder. Landmarkın 12'in X düzlemindeki konumuna göre sola hareket ettiği kamera görüntüsü şekil 5.6.'da gösterilmiştir.



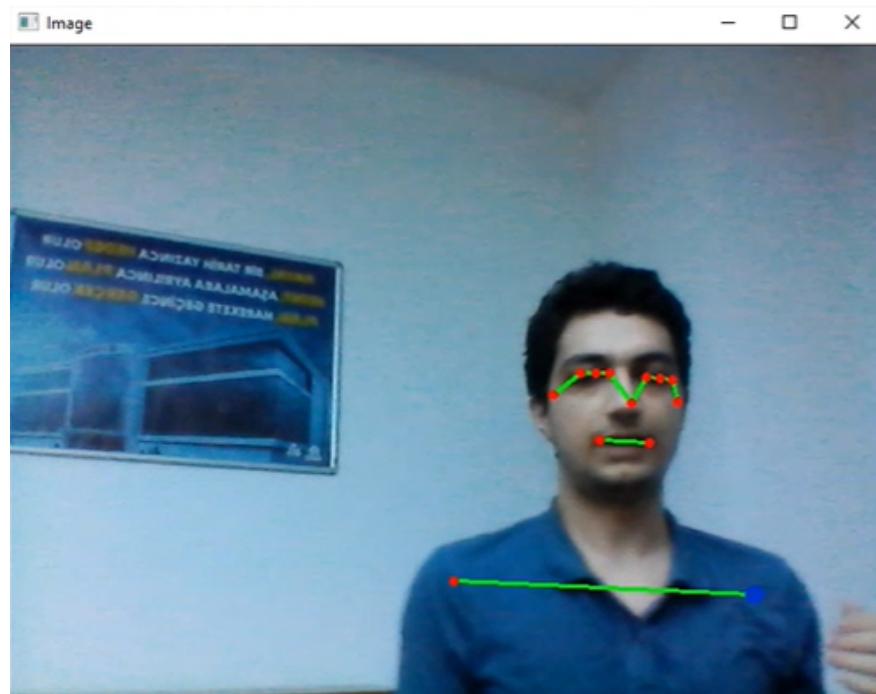
Şekil 5.5. Landmark 12'nin X düzlemindeki konumu 30'dan küçük.

Motorların sola doğru hareket ettiği zaman terminal üzerindeki çıktısı şekil xx'te gösterilmiştir.

```
Sola dönüyor  
[13, 737, 424]  
Motorlar İleri Yöne Hareket Ediyor  
Sola dönüyor  
[13, 724, 428]  
Motorlar İleri Yöne Hareket Ediyor  
Sola dönüyor  
[13, 730, 419]  
Motorlar İleri Yöne Hareket Ediyor  
Sola dönüyor
```

Şekil 5.6. Motorların sola döndüğü terminal çıktısı.

11 ve 12 numaralı landmarklar arasındaki mesafenin mutlak değeri 60 ve 140 piksel arasında olduğu ve landmark 12'in şekil 5.7.'de gösterilen temsili X-Y düzleminin konumunun 400'den fazla olduğu durumlarda soldaki motorlar hızlanır, sağdaki motorlar yavaşlar ve mobil robot sağa doğru hareket eder. Landmark 12'nin X düzlemindeki konumuna göre sola hareket ettiği kamera görüntüsü şekil 5.8.'de gösterilmiştir.



Şekil 5.7. Landmark 12'nin X düzlemindeki konumu 400'den büyük.

Motorların sağa doğru hareket ettiği zaman terminal üzerindeki çıktısı şekil 5.8'de gösterilmiştir.

```
Saga donuyor
[13, 73, 311]
Motorlar İleri Yände Hareket Ediyor
Saga donuyor
[13, 73, 311]
Motorlar İleri Yände Hareket Ediyor
Saga donuyor
[13, 74, 311]
Motorlar İleri Yände Hareket Ediyor
Saga donuyor
[13, 74, 311]
Motorlar İleri Yände Hareket Ediyor
Saga donuyor
```

Şekil 5.8. Motorların sağa döndüğü terminal çıktısı.

6. TARTIŞMA

OpenCV ile İnsan Hareketleri İzleyen Mobil Robot Tasarımı projesinin gerçekleştirilen süresince yapılan çalışmalar sonucunda, MediaPipe kütüphanesi, Raspberry Pi 4 geliştirme kartına entegre edilememiştir. Raspberry Pi 4 varsayılan işletim sistemi olan Raspbian işletim sistemine, hazır geliştirme ortamları aracılığı ile (PyCharm, Thonny IDE) ve el ile entegre edilmeye çalışılmış fakat başarılı olunamamıştır. IDE'den ve Raspbian Terminal üzerinden gelen hata bildirimleri, Python sürümü ile ilgili bir problem olduğunu belirtmiş, fakat tüm Python 3 sürümleri yüklenip denendiği zaman da sorun çözülememiştir. Sorunun çözümü için beta sürümünde olan ve 64-bit mimarisi ile geliştirilen Raspbian x64 işletim sistemi Raspberry Pi 4 geliştirme kartı içerisinde yüklenmiştir. Raspbian x64 işletim sistemi içerisinde MediaPipe kurulumu başarılı bir şekilde gerçekleştirılmıştır fakat Raspbian x64 işletim sistemi beta aşamasında olduğundan dolayı, kamera çalışma komutları pasifdir ve kullanılamamaktadır.

Raspberry Pi 4 geliştirme kartı içerisinde, varsayılan işletim sisteminden farklı olarak Ubuntu işletim sistemi kurulmuş fakat mevcut sorunlar bu işletim sisteminde de çözülememiştir.

Programın test edildiği ve çalıştığını doğrulandığı Windows 10 işletim sistemi ise, Raspberry Pi 4 geliştirici kartındaki ARM-64 tabanlı işlemci frekansının üzerinde bir frekans gereksinimine sahip olduğu için overclock işlemi ile gerekli işlemci frekansı elde edilebilir olmaktadır, fakat bu durumda da çok iyi bir soğutma yapılamadığı sürece ciddi performans sorunları yaşanma ihtimali ve yanım riski ile karşılaşılmış, Windows 10'un yüklenmesi denenmemiştir.

OpenCV ile İnsan Hareketleri İzleyen Mobil Robot Tasarımı projesinin diğer takip özellikli mobil robotlardan farkı, MediaPipe yapay zekâ arayüzü ile oluşturulan insanın postürü üzerinde işlem ve analiz yapması, insan hareketleri sonucu bu postüre göre kararlar almasıdır. Omuzlar arasındaki mesafeye göre hızını ayarlamaktadır ve sol omuza göre geleceği yeri, geleceği zamanı belirlemektedir. MediaPipe kurulum sorunu, Raspbian x64 işletim sistemi beta sürümünden çıkışip tam sürümle geçildiği zaman çözüleceği ve projenin tam anlamıyla çalışır duruma geleceği tahmin edilmektedir.

7. KAYNAKÇA

- [1] Google LLC, MediaPipe Pose, <https://google.github.io/mediapipe/solutions/pose.html> (03.07.2021)
- [2] OpenCV, About, <https://opencv.org/about/> (03.07.2021)
- [3] Brian Schneider, Lityum Polimer Pil, <https://rogershobbycenter.com/lipoguide/> (02.07.2021)