|  |
| --- |
| t.c.  PAMUKKALE ünİversİtesİ  TEKNOLOJİ FAKÜLTESİ  MEKATRONİK MÜHENDİSLİĞİ |
|  |
| gesture control |
|  |
| ahmet saraç |
| DENİZLİ, Aralık - 2020   |  | | --- | | t.c.  PAMUKKALE üniversitesi  TEKNOLOJİ FAKÜLTESİ  MEKATRONİK MÜHENDİSLİĞİ | |  | | gesture control | | ahmet saraç | | DENİZLİ, Aralık - 2020 | |

|  |
| --- |
| ÖZET |
| gesture control |
|  |
| ahmet saraç |
| PAMUKKALE ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ |
| MEKATRONİK MÜHENDİSLİĞİ |
|  |
| (PROJE DANIŞMANI:DOÇ. DR. METİN SAYER) |
| DENİZLİ, Aralık - 2020 |
| Birçok sektörde kullanılan joystick, klavye vb. araçların teknolojisinin eski kalması ve temas gerekmesinden dolayı ve şu an bulunduğumuz durumda (Kovid-19) Araç, Robotik, Uzay vb. sektörlerinde kontrol amacıyla kullanılan parçaların sadece el jestleri ve hareketleri ile kontrol edilmesi amaçlanmıştır. Bu hususta bazı kickstarter projeleri kullanılmış olup yazılım üzerine gidilmiştir. Kullanılan araçlar iki kamera ve iki kızılötesi sensörden oluşan Leap Motiondur. El görüntüleri alınıp koordinat sistemine entegre edilmiştir bu veriler ile kontrol edilme sağlanmıştır. |
| **ANAHTAR KELİMELER:** |
| Leap Motion, Jest Hareketi, Makeblock, Kumandasız Kontrol |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

İÇİNDEKİLER

Sayfa

[ÖZET i](#_Toc58191229)

[Leap Motion, Jest Hareketi, Makeblock, Kumandasız Kontrol i](#_Toc58191230)

[İÇİNDEKİLER ii](#_Toc58191231)

[ŞEKİL LİSTESİ iii](#_Toc58191232)

[ÖNSÖZ iv](#_Toc58191233)

[1. GİRİŞ 1](#_Toc58191234)

[1.1 Projenin Amacı 1](#_Toc58191235)

[1.2 Projenin Geliştirilmesi 1](#_Toc58191236)

[1.3 PROJEYİ ETKİLEYEN FAKTÖRLER 1](#_Toc58191237)

[1.3.1 Farklı Kişiler Tarafından Kontrol 1](#_Toc58191238)

[1.3.2 Bağlantı Kopmaları 1](#_Toc58191239)

[2. Malzeme Seçimi 2](#_Toc58191240)

[2.1 Proje Tasarım ve Malzemeler 2](#_Toc58191241)

[2.2 Kullanılan Malzemeler 2](#_Toc58191242)

[2.3 Leap Motion Nedir? 2](#_Toc58191243)

[2.4 Makeblock Nedir? 4](#_Toc58191244)

[3. Yapım Aşaması ve Kaynak Kodlar 6](#_Toc58191245)

[3.1 PROJENİN ÇALIŞMA TASARIMI VE TASLAĞI 6](#_Toc58191246)

[3.2 C# Kaynak Kodu 6](#_Toc58191247)

[3.3 C# Kaynak Kod İncelemesi 9](#_Toc58191248)

[3.4 Arduino kodu 10](#_Toc58191249)

[3.5 Arduino Kodu İncelemesi 12](#_Toc58191250)

[3.6 Projenin Son Halinden Görüntüler 12](#_Toc58191251)

[4. KAYNAKLAR 14](#_Toc58191252)

[5. ÖZGEÇMİŞ 15](#_Toc58191253)

ŞEKİL LİSTESİ

Sayfa

[Şekil I : Leap Motion 3](#_Toc58191215)

[Şekil II : Leap Motion Ayrıntı 3](#_Toc58191216)

[Şekil III : Makeblock Mbot Ranger 4](#_Toc58191217)

[Şekil IV : Auriga kartı 5](#_Toc58191218)

[Şekil V : Leap Motion Yazılımı 12](file:///D:\Mekatronik%20Mühendisliği%20Proje%20Yazım%20Şablonu%20(1).docx#_Toc58191219)

[Şekil VI : Araç ve yazılım 13](file:///D:\Mekatronik%20Mühendisliği%20Proje%20Yazım%20Şablonu%20(1).docx#_Toc58191220)

ÖNSÖZ

Bu tez çalışmasının planlanmasında, araştırılmasında, yürütülmesinde, oluşumunda ilgi ve desteğini esirgemeyen, engin bilgi ve tecrübelerinden yararlandığım, yönlendirme ve bilgilendirmeleriyle çalışmamı bilimsel temeller ışığında şekillendiren sayın hocam Doç. Dr. Metin SAYER’ e sonsuz teşekkürlerimi sunarım.

1. GİRİŞ
   1. Projenin Amacı

Sıradan kontrol yöntemleri eski ve öğrenim gerektiği için bu sistemlerin teknolojinin gelişmesiyle eski kalmış olduğu için yeni teknolojinin VR, AR ve Sanal Gerçekliğin gelişmesiyle gerçek hayatın Sanal aleme entegrasyonunda kontrol sistemlerinin de entegre edilmesi düşünülmüştür. Bu kontrol yazılımı sayesinde yer sistemleri hava sistemleri ve birçok alanda kolayca entegre edilerek sistemi kolaylaştırma ve temassız kontrol sağlanmaktadır.

* 1. Projenin Geliştirilmesi

Proje geliştirilirken Leap Motion kiti ve SDK’sı kullanılmıştır. Bu ürün kamera ve kızılötesi sensörler sayesinde elin koordinat sisteminde her elin parmaklarının verisi, elin uçuş sistemlerinde de kullanılan x ekseninde dönme, y ekseninde dönme ve z ekseninde dönme hareketlerinin datasını çıkarmaktadır. Bu veriler sayesinde, Visual Studio ile alıp Bluetooth ile araçlara entegrasyonu yapılmaya çalışılmıştır.

* 1. PROJEYİ ETKİLEYEN FAKTÖRLER
     1. Farklı Kişiler Tarafından Kontrol

Cihazın karşısına çıkan her kişinin elini (en fazla iki el) algılamaktadır bu da güvenlik sorunlarını devamında getirir.

* + 1. Bağlantı Kopmaları

Cihaz bluetooth özelliği ile birleştiği için ve dünyada 2.4 Ghz bandı bütün sistemlerde kullanıldığı için ara sıra sinyal karışmaları olmaktadır.

1. Malzeme Seçimi
   1. Proje Tasarım ve Malzemeler

Projenin yapıp aşamalarına başlamadan önce kafamızda bir taslak oluşturmak gerekiyor. Nasıl bir sistem olacak hangi kontrolcü kullanılacak kasası nasıl olacak hangi programlama dili kullanılacak gibi bu tarz projeler hobi severlerin hep ilgisini çeken projelerden olmuşlardır

* 1. Kullanılan Malzemeler
* 1 Adet Leap Motion kiti
* 1 Adet Mbot Ranger kiti
* 2 Adet 18650 Pil
  1. Leap Motion Nedir?

Gelecek, sonsuz dünyaların parmaklarınızın ucunda olduğu bir yerdir. Doğal ve 3 boyutlu etkileşim kurduğunuz yer. Denetleyicilere ihtiyacınız olmayan bir yer. Ultraleap, Leap Motion ve Ultrahaptics 2019'da bir araya geldiğinde oluşmuştur. Dünyanın en gelişmiş el takibini havada dokunma hissi yaratan tek dokunsal teknoloji ile birleştirdik. ABD'nin Silikon Vadisi ve İngiltere'nin Bristol kentinde bulunan, dünya çapında 150'den fazla kişiden oluşan bir ekibimiz var. Ekibimiz ara yüz tasarımı, akustik, makine öğrenimi ve bilgisayarla görü alanlarında dünya lideri uzmanlardan oluşmaktadır.



Şekil I : Leap Motion

Leap Motion Controller, ellerinizin hareketlerini benzersiz bir doğrulukla yakalayan optik bir el izleme modülüdür. XR'den temassız kiosklara, Leap Motion Controller, dijital içerikle etkileşimi doğal ve zahmetsiz hale getirir.

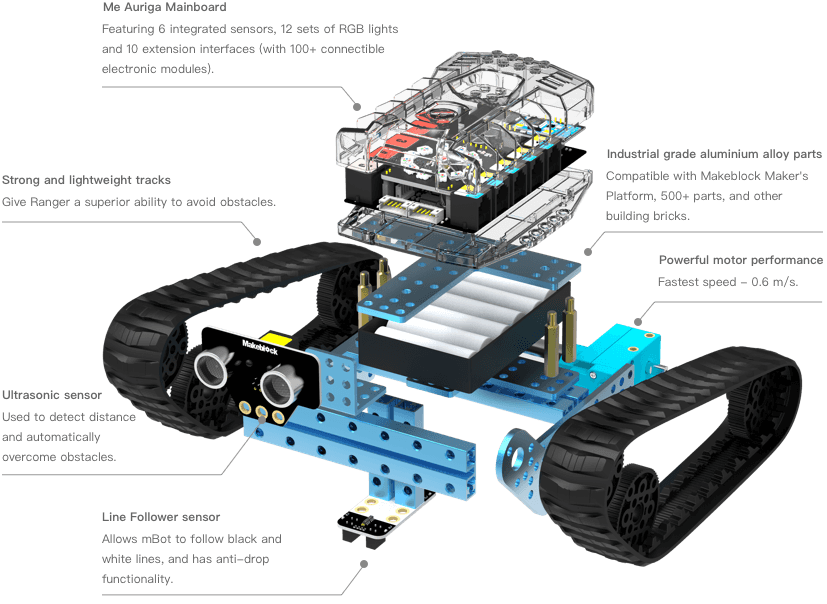


Şekil II : Leap Motion Ayrıntı

İçinde bulunan çift kamera ve kızılötesi sensörleri sayesinde elin geniş bir haritalandırması ve konum takibini sağlamaktadır. Kameralar görüntü işleme teknolojisini kullanmaktadır.

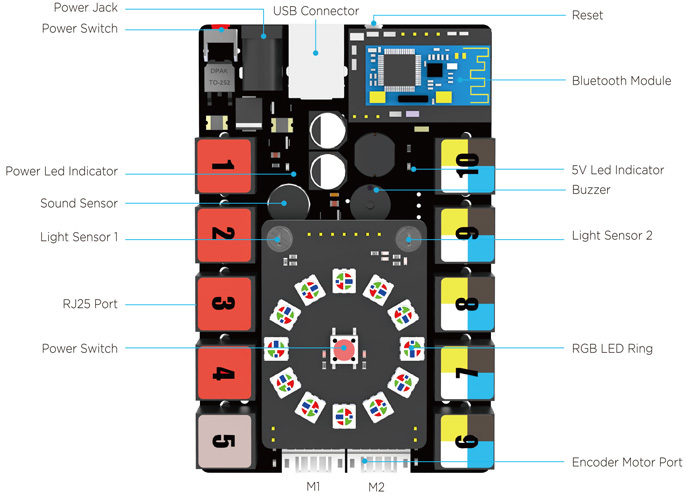
* 1. Makeblock Nedir?

MBot Ranger, eğlencenin üç katı için önceden ayarlanmış 3 formla birlikte gelen MBot'un gelişmiş bir sürümüdür! Yapımdan çalıştırmaya ve programlamaya kadar, Ranger, çocukların STEAM eğitimini basit ve eğlenceli bir şekilde almalarına olanak tanır. Ranger ayrıca çocukların robot programlamaya başlamasını sağlar, büyürken ve problem çözme yeteneklerini geliştirirken onlarla birlikte ilerlemelerini sağlar.



Şekil III : Makeblock Mbot Ranger

MBot Ranger yaklaşık 100 mekanik parça ve elektronik modül içerir. Önceden belirlenmiş 3 form vardır: tank benzeri Off-road Land Raider, üç tekerlekli yarış arabası. Birçok sensör eklenmesine olanak sağlayan Arduino Mega Temeli bulunan Auriga kartı bulunmaktadır. Auriga kartı üzerinde bulunan motor sürücü, Buzzer, Ses sensörü, Gyro bulmaktadır. Ayrıca portlarından başka sensörlerin takılmasına olanak sağlar.



Şekil IV : Auriga kartı

Sistemde bulunan RJ25 portları sayesinde arduino megada bulunan çıkışları daha kolay bağlantı yapmaya imkân veren Auriga kartı birçok sensörün bağlantısı için olanak sağlıyor.

1. Yapım Aşaması ve Kaynak Kodlar
   1. PROJENİN ÇALIŞMA TASARIMI VE TASLAĞI

Proje bilgisayar tabanlı çalışıp Leap Motion bilgisayara USB kablosu ile bağlanır. C# programı ile Leap Motion’dan aldığımız veriler işlenip gene bilgisayardaki bluetooth sayesinde alınan verilere göre hareketi sağlaması için kablosuz olarak araca aktarıp aracın aldığı verileri işleyip harekete dönüştürmesi sistemi ile çalışmaktadır.

* 1. C# Kaynak Kodu

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using Leap;

using System.IO.Ports;

namespace TEZ

{

public partial class Form1 : Form,ILeapEventDelegate

{

private Controller controller;

private LeapEventListener listener;

public Form1()

{

InitializeComponent();

this.controller = new Controller();

this.listener = new LeapEventListener(this);

controller.AddListener(listener);

foreach (string portlar in SerialPort.GetPortNames())

{

comboBox1.Items.Add(portlar);

}

groupBox2.Enabled = false;

}

private void button1\_Click(object sender, EventArgs e)

{

try

{

serialPort1.PortName = comboBox1.Text;

serialPort1.BaudRate = 115200;

serialPort1.Open();

}

catch (Exception)

{

}

if (serialPort1.IsOpen)

{

groupBox2.Enabled = true;

}

}

delegate void LeapEventDelegate(string EventName);

public void LeapEventNotification(string EventName)

{

if (!this.InvokeRequired)

{

switch (EventName)

{

case "onInit":

break;

case "onConnect":

break;

case "onFrame":

detectGesture(this.controller.Frame());

detectHandPosition(this.controller.Frame());

break;

}

}

else

{

BeginInvoke(new LeapEventDelegate(LeapEventNotification), new object[] { EventName });

}

}

public void detectGesture(Leap.Frame frame)

{

GestureList gestures = frame.Gestures();

}

public void detectHandPosition(Leap.Frame frame)

{

HandList allHands = frame.Hands;

foreach(Hand hand in allHands)

{

float pitch = hand.Direction.Pitch;

float yaw = hand.Direction.Yaw;

float roll = hand.Direction.Roll;

double degPitch = pitch \* (180 / Math.PI);

double degYaw = yaw \* (180 / Math.PI);

double degRoll = roll \* (180 / Math.PI);

int intPitch = (int)degPitch;

int intYaw = (int)degYaw;

int intRoll = (int)degRoll;

float grab = hand.GrabStrength;

textBox1.Text = intPitch.ToString();

textBox2.Text = intYaw.ToString();

textBox3.Text = intRoll.ToString();

textBox4.Text = grab.ToString();

if (groupBox2.Enabled == true)

{

if (intPitch < -20)

serialPort1.Write("F");

else if (intPitch > 30)

serialPort1.Write("B");

else if (intYaw < -20)

serialPort1.Write("M");

else if (intYaw > 20)

serialPort1.Write("N");

else

serialPort1.Write("S");

}

}

}

}

public interface ILeapEventDelegate

{

void LeapEventNotification(string EventName);

}

public class LeapEventListener : Listener

{

ILeapEventDelegate eventDelegate;

public LeapEventListener(ILeapEventDelegate delegateObject)

{

this.eventDelegate = delegateObject;

}

public override void OnInit(Controller controller)

{

this.eventDelegate.LeapEventNotification("onInit");

}

public override void OnConnect(Controller controller)

{

this.eventDelegate.LeapEventNotification("onConnect");

}

public override void OnFrame(Controller controller)

{

this.eventDelegate.LeapEventNotification("onFrame");

}

public override void OnExit(Controller controller)

{

this.eventDelegate.LeapEventNotification("onExit");

}

public override void OnDisconnect(Controller controller)

{

this.eventDelegate.LeapEventNotification("onDisconnect");

}

}

}

* 1. C# Kaynak Kod İncelemesi

İlk satırda bulunan using ifadesi donanımın çalışması için gerekli kütüphanelerin eklenmesi için kullanılmaktadır. Public Form1 ifadesi de masaüstü uygulamamız üzerinde bulunan araçların işlevlerinin yazıldığı bölümdür. Burada Bluetooth serial haberleşme hızımızı ve bağlantı sonrası verileri kullanması için kutuların aktifleşmesi için yazdığımız bölüm bulunmaktadır. Public void detectHandPosition ise el verilerinin textbox içine yazılması için yazdığımız bölümdür. If (groupBox2.Enabled == true) bu bölüm ise bluetooth bağlantısı sağlandıktan sonra textbox yazdırdığımız verileri okumak için açıp daha sonra if döngüsüne sokup belirli bir veriden sonra karakter olarak verileri bluetooth ile aktarmak amaçlanmıştı.

* 1. Arduino kodu

#include "MeAuriga.h"

#include <SoftwareSerial.h>

int d=10;

char btdata;

MeEncoderOnBoard Encoder\_1(SLOT1);

MeEncoderOnBoard Encoder\_2(SLOT2);

void isr\_process\_encoder1(void)

{

if(digitalRead(Encoder\_1.getPortB()) == 0)

{

Encoder\_1.pulsePosMinus();

}

else

{

Encoder\_1.pulsePosPlus();;

}

}

void isr\_process\_encoder2(void)

{

if(digitalRead(Encoder\_2.getPortB()) == 0)

{

Encoder\_2.pulsePosMinus();

}

else

{

Encoder\_2.pulsePosPlus();

}

}

void setup()

{

attachInterrupt(Encoder\_1.getIntNum(), isr\_process\_encoder1, RISING);

attachInterrupt(Encoder\_2.getIntNum(), isr\_process\_encoder2, RISING);

Serial.begin(115200);

Serial.print("bluetooth başlatılıyor");

TCCR1A = \_BV(WGM10);

TCCR1B = \_BV(CS11) | \_BV(WGM12);

TCCR2A = \_BV(WGM21) | \_BV(WGM20);

TCCR2B = \_BV(CS21);

}

void loop()

{

if(Serial.available())

{

btdata = Serial.read();

if (btdata == 'F') {

Encoder\_1.setTarPWM(200);

Encoder\_2.setTarPWM(-200);

}

else if (btdata == 'B') {

Encoder\_1.setTarPWM(-200);

Encoder\_2.setTarPWM(200);

}

else if (btdata == 'M') {

Encoder\_1.setTarPWM(200);

Encoder\_2.setTarPWM(200);

}

else if (btdata == 'N') {

Encoder\_1.setTarPWM(-200);

Encoder\_2.setTarPWM(-200);

}

else if (btdata=='S'){

Encoder\_1.setTarPWM(0);

Encoder\_2.setTarPWM(0);

}

}

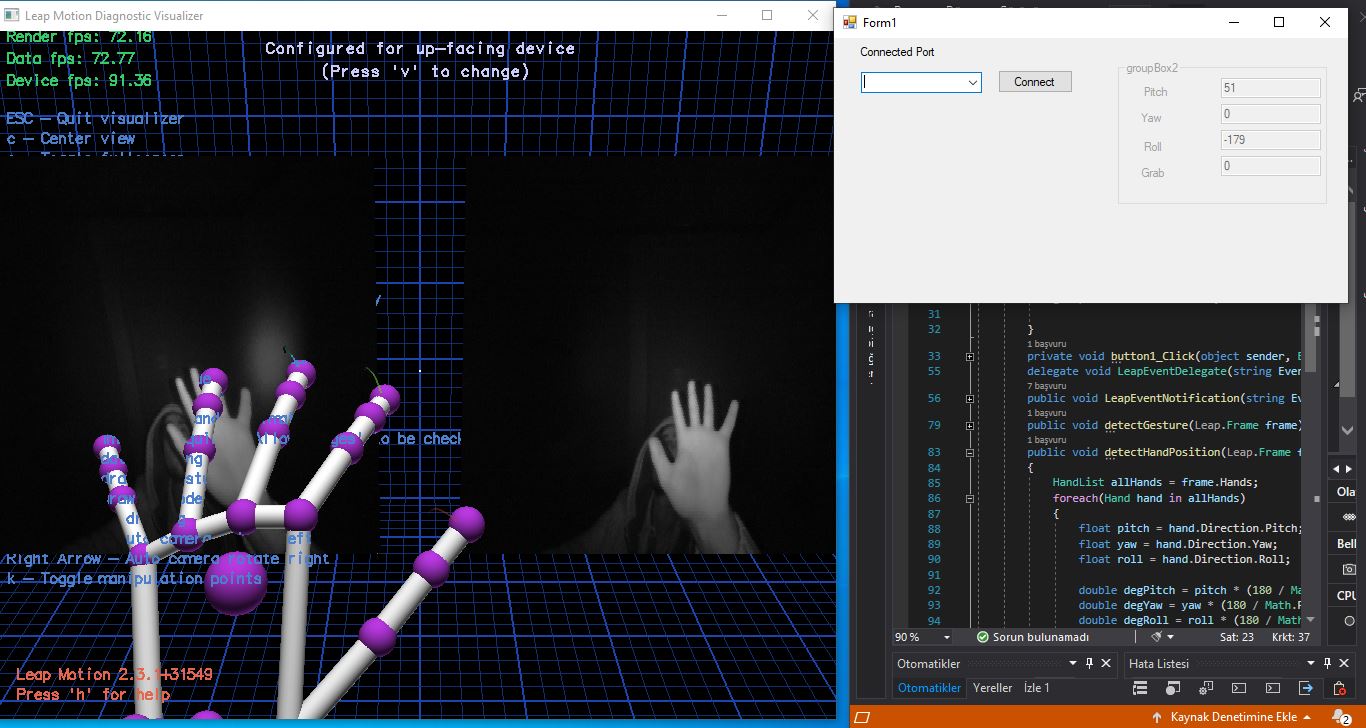
Encoder\_1.loop();

Encoder\_2.loop();

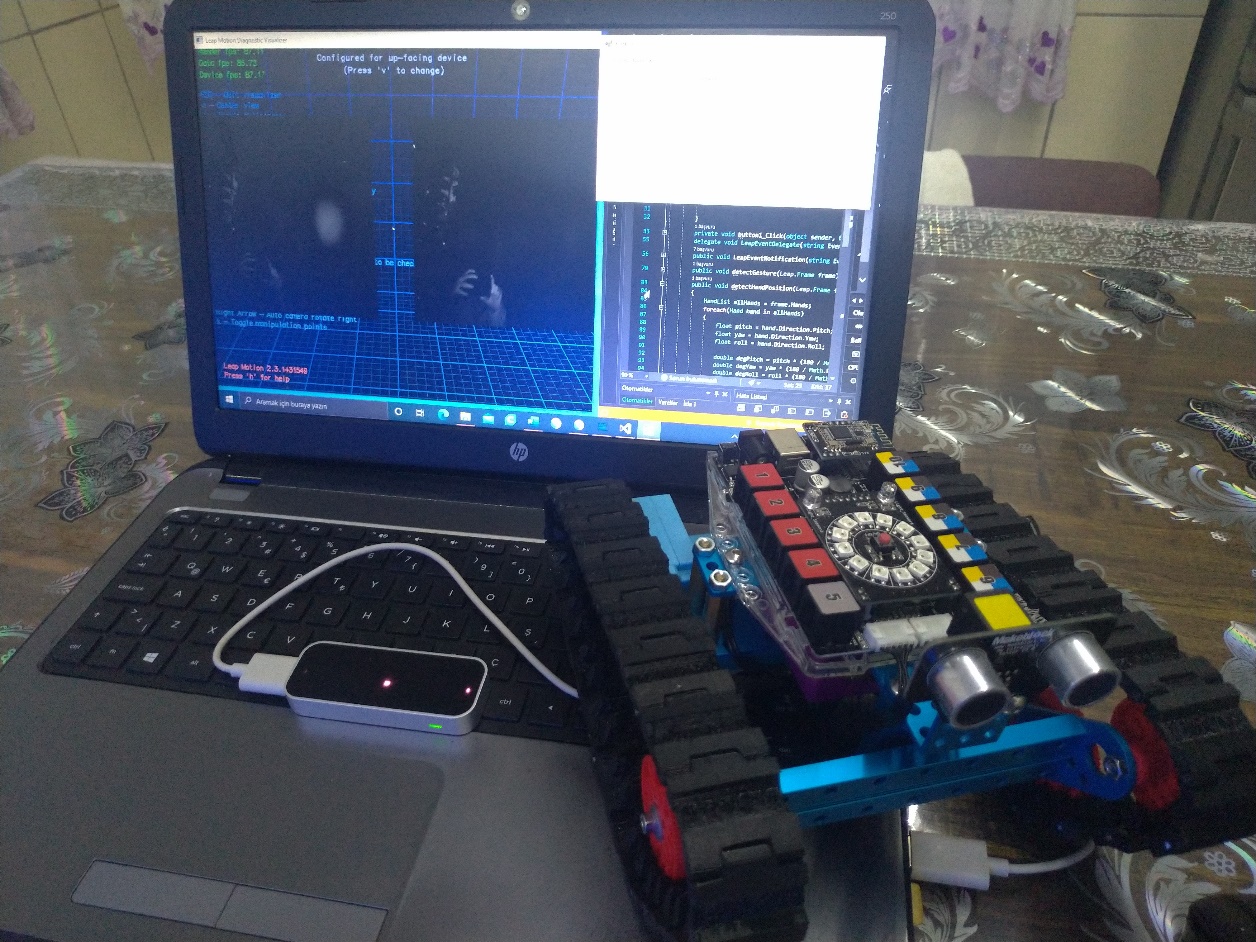
}

* 1. Arduino Kodu İncelemesi

İlk satırda kart için yazılmış üreticinin kütüphanesi ve bluetooth gibi serial veri kullanılması için kütüphane eklenmiştir. Sonraki satırda gecikme için değer ve gelen dataları tutan char değişkeni verilmiştir. Diğer satırda motorların çıkışları ve slotlar yazılmıştır. Daha sonra motorların Min ve Max değerleri verilmiş olup if döngüsüne atanmıştır. Serial haberleşmeyi başlatmak için hız yazılıp loop döngüsünde gelen karakter değerlerine göre araç yönü ayarlanmıştır ve döngü sonunda motorlara döndürülmüştür.

* 1. Projenin Son Halinden Görüntüler

Şekil V : Leap Motion Yazılımı



Şekil VI : Araç ve yazılım

1. KAYNAKLAR

* <https://forums.leapmotion.com/t/unity-c-api-documentation-for-sdk-v4/10545>
* <https://forum.makeblock.com/c/makeblock-products/ranger>
* <https://www.intorobotics.com/how-i-hacked-the-mbot-ranger-kit-for-autonomous-driving-capabilities/>
* <https://elektrikelektronikprojeleri.blogspot.com/2015/03/arduino-c-bluetooth-baglants-ile-dc.html>
* <https://developer-archive.leapmotion.com/documentation/csharp/devguide/Sample_Tutorial.html>

1. ÖZGEÇMİŞ

Adı Soyadı : Ahmet Saraç

Doğum Yeri ve Tarihi : ANTALYA 15.09.1999

Lisans Üniversite : Pamukkale Üniversitesi

Elektronik posta : ahmetsaraca07@gmail.com

İletişim Adresi :05077294267