

Kari Tudományos Diákköri Konferencia
Győr, Széchenyi István Egyetem
Gépészmérnöki, Informatikai és Villamosmérnöki Kar
Informatika Tanszék

2022. november 4.

Önvezető járművek fényjelzéseinek szimulálása

“A biztonság elsősegélyt nyújt a sértetleneknek.” – F.S. Hughes

Pályamunka lezárásának időpontja: 2022.11.04.

2022.

Konzulensek: Dr. Kovács Katalin és Dr. Szi Brigitta

Szerzők: Sándor Ágoston Pál és Dancs Ádám Dávid

Tartalom

Önvezető járművek fényjelzéseinek szimulálása	1
1. Bevezetés	4
2. A megoldandó probléma.....	6
3. A kutatási téma ismertetése	7
3.1 A területen eddig elért eredmények	7
3.2 A jelzésekre visszavezethető (pavlovi) reflexek.....	8
4. Ütemezések a virtuális valóságban	10
4.1 360° Tér létrehozása	11
5. 3D modellezés folyamata.....	14
5.1 Blender	14
5.2 VR addon használata.....	16
5.3 HTC Vive.....	16
6. Arduino Uno programozása a virtuális térben való lábmozgás adatgyűjtéséhez.....	19
7. Virtuális valóságban történő tesztelés folyamata.....	26
7.1 Helyszín	26
7.2 Lebonyolítás.....	26
7.3 Adatelemzés megközelítési módszere	28
8. Tesztelési kimenetek feldolgozása.....	29
8.1 Microsoft Office programcsomag: Excel 2019.....	29
8.2 Virtuális valóság szcenáriók szerint átlagolt kimutatás	29
9. Valós környezetben végzett felmérések tervezése, megvalósítása	45
10. A projekt jövője, fejlődési utak.....	48
10.1 Szem és fejmozgásból elemzett reakciók vizsgálata	48
10.2 Tesztesetek további megközelítési lehetőségei.....	48
10.3 Kormányról való vezérlés, optimalizálás.....	48

11.	Hipotézis alátámasztása vagy megcáfolása.....	50
12.	Témához kötődő idegen és szakszavak szótára.....	51
13.	Irodalomjegyzék	54

1. Bevezetés

Az önvezető járművek megjelenéséig klasszikus közlekedési szituáció a gyalogátkelőhelyeknél a gyalogos és a gépjárművet vezető verbális és nonverbális kommunikációja, vagyis a forgalmi szituációban a gyalogátkelőhelyen történő áthaladáskor két ember kommunikált egymással. Az autonóm, önvezető járművek közlekedési forgalomban történő megjelenése a korábbi szabályokat átírja. Az új forgalmi modellben már nem két ember verbális és nonverbális kommunikációját és viselkedését kell vizsgálni, hanem egy emberi tényező és egy nem emberi tényező kommunikációját kell elemzés alá venni. A közúti balesetek elkerülése érdekében a kommunikációnak minden résztvevő számára egyértelműnek kell lennie.

Az önvezető járműveknek a közúti közlekedésben való megjelenésével és elterjedésével számos kutatás foglalkozik. Vannak kutatások, amelyek az önvezető járműveknek a társadalmi hatásait vizsgálják, míg mások a jogszabályi háttér köré építik a vizsgálatot. Az ipari kutatások jelentős része az önvezető járművek szoftverfejlesztését, finomhangolását célozza meg. Számos kutatás a gyalogosok viselkedését, reakcióit jellemző megfigyelhető és mérhető paraméterek detektálására helyezi a hangsúlyt, majd azokat elemezve tesznek előrejelzéseket a gyalogos jövőbeni viselkedésére. Mások a közlekedési forgalmi modellben külön paraméterként kezelik a gyalogosnak az érzelmi reakcióit, kifejezetten a kontextuális információk detektálására (pl. fejmozgás, testhelyzet, tekintet stb.) fókuszálnak, azokat felmérve definiálnak a járműre vonatkozó viselkedési scénáriókat [1].

Napjainkban egyre inkább előtérbe kerülnek azok a kutatások, ahol egy külső ember-gép interfész segítségével próbálnak következtetni a gyalogosok gyalogátkelőhelyen történő átkelési döntéseire. Ez az úgynevezett külső ember-gép interfész (eHMI) a gépjárművön a gyalogos által jól látható módon kerül elhelyezésre [2]. Ez lehet egy fényjelzés vagy egy fényjelzéssorozat jól definiált jelentéstartalommal, amelyek egyértelmű üzenetet közvetítenek a gyalogos számára. Ezen fényjelzéseknek kifejezetten egyértelműnek és összetéveszthetetlennek kell lenniük, hiszen a közlekedésben az élet összes területéről, kortól függetlenül mindenki részt vehet. Az eHMI ötlete a jól ismert, a közúti forgalomban már jól bevált és alkalmazott fény alapú jelzések alapötletén nyugszik, amelyek a közlekedésben résztvevők számára figyelemfelhívást közölnek, avagy figyelmeztetnek, mint az irányjelzők vagy a hátsó féklámpák. A kutatásban a nagy vezető járműipari cégek is szerepet vállalnak, a

Mercedes-Benz AG is végez fényjelzés alapú kutatásokat. Az általuk alkalmazott modellben kifejlesztett zöldeskék szín fényjelzésével azt kommunikálják a gyalogos felé, hogy a gépjármű önvezető módra van kapcsolva, vagyis egyértelmű jelentéstartalmat közöl. A fényjelzés tehát csak azt jelenti, hogy a gépjármű önvezető módba van kapcsolva, de nem jelenti azt, hogy a gyalogos számára szabad az átkelés. Hochman és csapata a fényjelzések mellett további üzeneteket alkalmaz a gyalogos tájékoztatására, amelyet az alábbi ábra szemléltet [3].



FIGURE 3 | The e-HMI Messages; each column displays the message type (status or advice) in both written (in Hebrew) and symbolic messages. The background color is compatible with the message meaning according to the color convention.

1. ábra – Hochman és csapata által alkalmazott üzenetek a gyalogosok tájékoztatására
[Forrás: <https://www.frontiersin.org/articles/10.3389/fpsyg.2020.585280/full>]

A hagyományos, vezető által irányított gépjárművek esetén a gyalogos döntését segítő szemkontaktus hiányzik, ami biztonságot adott a gyalogosnak az átkelés megkezdésénél. Ez a bizonytalanság felvet olyan kérdéseket, mint az “önvezető járművekbe” vetett bizalom. A kutatás során elsősorban ezen bizalom kérdésére, vizsgálatára helyeztük a hangsúlyt.

A dolgozatban elsőként a megoldandó problémát, majd a megoldásra irányuló ötletet ismertetjük, valamint a témában végzett főbb kutatásokat. Ezt követően bemutatjuk az általunk alkalmazott adatgyűjtési módszert. Végül az adatok feldolgozását ismertetjük, majd az eredményekből megállapításokat teszünk az eredeti felvetéseinkre.

2. A megoldandó probléma

Napjainkban, az autonóm, önvezető járművek térnyerése folyamatos, kétségtelenül a jövő közlekedésének szerves résztvevői lesznek, ahogy erről már a dolgozat bevezető részében is írtunk. Mindezek ellenére, főleg Magyarországon az önvezetés közlekedésben való gyakorlata még egy rendkívül újkeletű dolog. Mint a közlekedés aktív résztvevői fontosnak tarjuk és egyben kötelességünknek is érezzük, hogy a témában a kutatásunk során olyan megoldást találjunk és lehetőség szerint valósítsunk meg a kutatás első szakaszában a virtuális térben, majd a következő szakaszban a valóságban is, amely képes a gyalogosok bizalmának felmérésére, hozzájárulva a közúti balesetek számának csökkentéséhez.

3. A kutatási téma ismertetése

Az általunk végzett kutatás a gyalogosnak az önvezető járművön elhelyezett fényjelzésbe vetett bizalmának feltérképezését és elemzését célozza meg. Ebben a modellben a gyalogosnak az önvezető járművön elhelyezett, a normál járművektől különböző fényjelzéseire tett reakciójára voltunk kíváncsiak, van-e bizalom a gyalogos részéről, hogy az önvezető jármű az adott közlekedési szituációban valós közlekedési esetekhez hasonlóan viselkedik. Az ötlet és az azzal kapcsolatos vizsgálatok személyes érdeklődésünkéből és a Közlekedésepítési és Vízmérnöki Tanszék javaslatára jött létre [4].

3.1 A területen eddig elért eredmények

Bizonyos elemzések alapján a gyalogosok nagyobb bizalmat vetnek az egyszerű közlekedési jelzőlámpák jelzéseibe, mint az önvezető autók autonóm döntéseibe. Felmerül az ötlet, mi lenne, ha a két területet, vagyis a jelzőlámpák jelzéseit és az önvezető járműbe vetett bizalom kérdéskört összefűznénk, vagyis, ha már a közlekedési jelzőlámpák jelzéseibe kialakult a bizalom, hogyan lehetne ezt felhasználni az önvezető járművekbe vetett bizalom mérésére. A felvetés nem újkeletű, azonban konkrét megoldás, eredmény még nem született a témában. Egy nemrégiben készült tanulmányban a Michigani Egyetem kutatói az autonóm járművek viselkedésén keresztüli kommunikációra összpontosítottak, hogy felmérjék hogyan reagálnak a járókelők az autókra különböző közlekedési helyzetekben. Létrehoztak egy virtuális valóság szimulátort, amelyben az embereknek el kellett dönteniük, hogy átmennek-e az úton a kereszteződésekben. Különböző szimulációkban az önvezető autók többé-kevésbé agresszívan közelítettek a gyalogátkelőhelyekhez, a gyalogosokhoz. A kereszteződésben esetenként közlekedési lámpák is működtek. Agresszívebb viselkedéssel az autó az utolsó pillanatban állt meg, hogy átengedjen egy gyalogost. Kevésbé agresszív üzemmódban korábban lassítani kezdett, azt „kommunikálva” a gyalogos számára, hogy átengedi. Az agresszív viselkedés csökkentette a gyalogosok önvezető járműbe vetett bizalmát és csökkentette annak valószínűségét, hogy átkeljenek az úton. Ez azonban jellemzően csak akkor volt megfigyelhető, amikor nem volt közlekedési jelzőlámpa. [4]

Abban az esetben, mikor közlekedési jelzőlámpa is volt az átkelőhelynél, a gyalogosok odafigyeltek annak jelzéseire, és döntéseik kvázi nem függték attól, hogy az autó agresszíven

vagy kevésbé agresszívan viselkedik. Ez azt jelzi, hogy a közlekedési lámpába vetett bizalmuk erősebb volt, mint az autó viselkedése miatti aggodalmuk.

A gyalogosok és az autósok gyakran használnak nonverbális kommunikációt kereszteződésekben és átkelőhelyeken, amire az önvezető autók nem képesek. Ez csökkentheti a gyalogosok biztonságát. Vannak kutatások, amelyek az önvezető autók és a gyalogosok közötti kommunikációra fókuszálnak és keresnek megoldásokat pl.: a kommunikációra a meglévő autóalkatrészeket, például a fényszórókat használták fel [5].

3.2 A jelzésekre visszavezethető (pavlovi) reflexek

Az autók fényjelzéseinek megválasztásához fontos megvizsgálunk az azokat kiváltó reflexek jellegét és működési mechanizmusát. Kiemelkedően fontos, hogy ezen színek ne csak a feltűnést és megkülönböztethetőséget szolgálják, hanem azonnali reakciót váltsanak ki, hiszen a gyalogos testi épsége függ ezen. Hans Eysenck egy korai, de átfogó vizsgálata alapján a színek preferenciája alá és fölérendeltségi viszonyba sorolható, mely szerint a piros a második legerősebb preferenciát élvező szín, a zöld pedig a harmadik [6]. Felmerülhet a kérdés, miszerint miért nem a két legerősebb preferencia színnel rendelkező színt használjuk, azaz a kék és a piros színeket. A válasz pedig igencsak egyszerű, ugyanis ezek már használatban vannak a rendfenntartási, katasztrófa elhárítási és egészségügyi szervek által, mint megkülönböztető jelzés. Eysenck kimutatása szerint ezek a preferenciák életkortól eltérően változnak, tehát egy kisgyermek eltérően reagál ugyanazon színekre, mint egy felnőtt. A megkülönböztető színjelzések megalkotásakor kerülnünk kell mindenképpen a neutrális csoportot, amelyek a homogén átmenetek árnyalataiban nyilvánulnak meg, ahol kettő vagy több egymáshoz hasonló színt keverünk, melyek neutrális hatást váltanak ki a nézőben. Ezen színek és árnyalatok sajnos nem alkalmasak figyelemfelkeltésre vagy legalábbis figyelmeztető jellegű megkülönböztetésre. A fényjelzés megalkotása során fontos tényező volt, hogy a fények alapvető "pavlovi reflexeket" indítsanak be, amelyek az emberekben már gyermekkoruk óta működnek [7].



2. ábra - Kezdeti tervek Blenderben (saját)

4. Ütemezések a virtuális valóságban

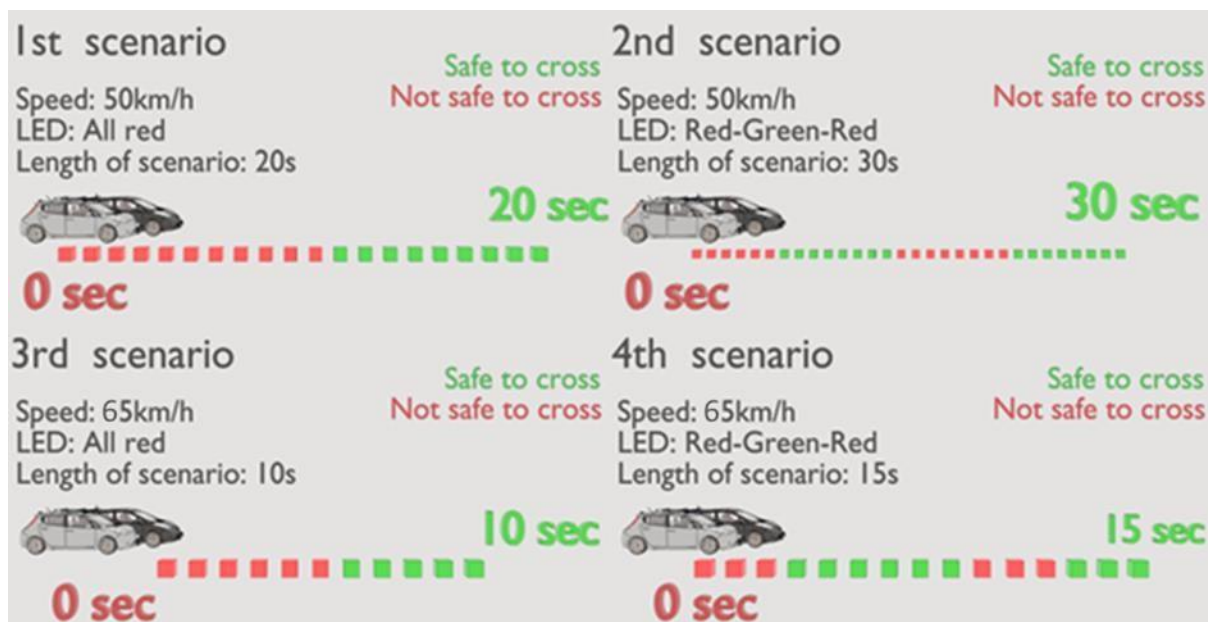
A kutatás első szakaszában nem valós környezetben, hanem szimulált környezetben végeztünk kísérleteket, ahol különböző közlekedési szituációkban két ütemezésben vizsgáltuk az önvezető járművek és a gyalogosok viselkedését.

Az első ütemezésben négy scenáriót határoztunk meg, Az első scenárióban egy jármű 20 km/h átlagsebességgel közelíti meg a gyalogátkelőhelyet elsőbbséget adva a gyalogosnak. A másodikban, ugyanezen sebességgel elmulasztja elsőbbségadási kötelezettségét. A harmadik és a negyedik scenárióban az érkező jármű sebességét 40 km/h-ra növeltük.



3. ábra - Első ütemezés rajzai Blenderben (saját)

A második ütemezésben szintén négy scenárió található, azonban további egy autóval bővült a vizsgálat. Elsőként a járművek 50km/h átlagsebességgel érkeznek elsőbbséget adva a gyalogosnak, majd ugyanezen sebességgel nem tesznek eleget elsőbbségadási kötelezettségüknek. A további kettő szituációban további sebességnövelés történt, ahol 65km/h sebességgel érkeznek meg a járművek a gyalogátkelőhelyhez.



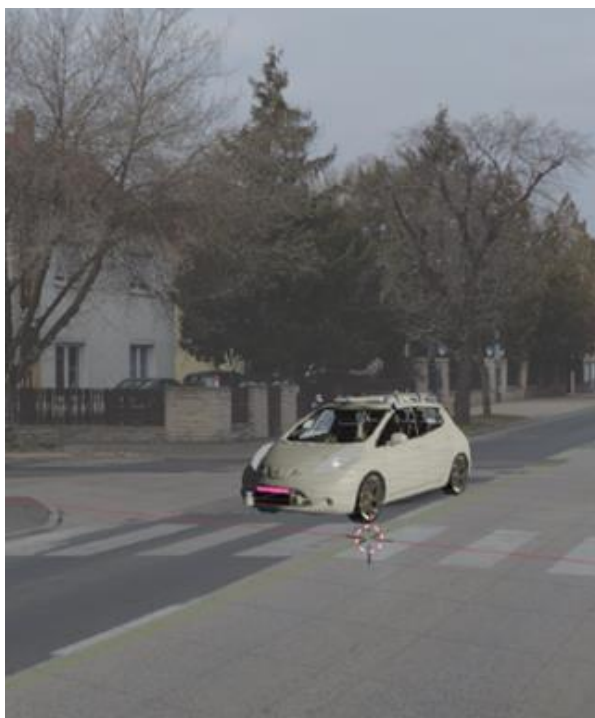
4. ábra - Második ütemezés rajzai Blenderben (saját)

A két ütemezésben érdekes különbség a gyalogosok megosztott figyelmének vizsgálata. Vajon képes-e a gyalogos egynél több objektumra összpontosítani a figyelmét és megfelelő döntést hozni, ugyanis a területorientált kognitív képességeket érintő tudományos álláspont szerint nem létezik megosztott figyelem.

4.1 360° Tér létrehozása

4.1.1 A virtuális szcenáriók helyszíne

Helyszín választásunk a győri Lomnitz és Nagy Imre utca kereszteződésében található gyalogátkelőhelyre esett Dr. Makó Emese, Közlekedésépítési és Vízmérnöki Tanszék Tanszékvezető Asszonyának javaslatára. A döntés azért erre a helyszínre esett, mert az itt található kereszteződés egy főútvonalon helyezkedik el, a járművek jellemzően nagy sebességgel haladnak, amely a gyalogosoknak potenciálisan nehezíti átkelésüket vagy beindítja veszélyérzetüket. A virtuális tér elkészítése elsősorban megkívánta, hogy közelről megismerjük a valós életbeli helyszínt, így a projekt megvalósítása több esetben is külső helyszíneken történő előzetes helyszíni bejárást igényelt [8].



5. ábra - Autó kezdetleges térbe helyezése Blenderben (saját)

4.1.2 Candao Go 360° kamera

A 360° tér létrehozásához több nagy felbontású fotót használunk, amelyeket később összeillesztve pontos mását tudtuk rekonstruálni a helyszínek. A fent leírt feladat megvalósítására a Közlekedésépítési és Vízmérnöki Tanszék biztosított számunkra egy *Candao Go* széles látószögű, speciális kamerát, amely alkalmas a kívánt fotók elvárt minőségű elkészítésére. A kamerával kimentünk a kísérlet helyszínéül szolgáló kereszteződésbe a kora délelőtti órákban, ezzel is elkerülve a nagy forgalmat, amely megkönnyítette a zajmentes képek készítését. Ezután készítettünk hat darab 180°-os fényképet a kamerával a gyalogos átkelőhely több pontjában is. Ezeket a képeket a Candao stúdióval utólagos feldolgozással összefűztük úgy, hogy egy 8K felbontású fotót kapjunk kimenetként. Ez a fotó biztosította számunkra a virtuális tér létrehozásához szükséges textúrákat a modellezési folyamat során.



6. ábra - Candao Go széleslátószögű kamera a helyszínen (saját)

5. 3D modellezés folyamata

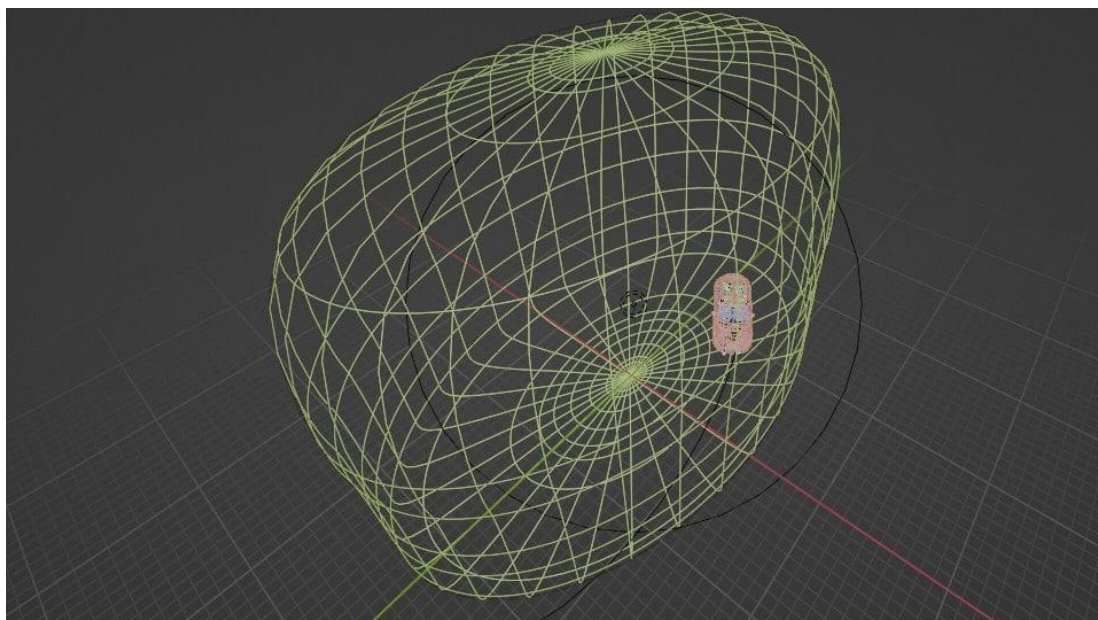
5.1 Blender

Az első ötlet a tér modellezéssel kapcsolatban egy nagy felbontású HDRI (High Dynamic Range Image - magas dinamikatartományú kép) volt. Ez azt jelentette elméletben, hogy a fotókat egy gömbre ráhúzzuk, majd fényvisszaverési és fényadási tulajdonságokkal látjuk el, így az később élethű környezetként szolgálhatott volna a kísérleti alanyoknak. Ez sajnos a valóságban nem működött, ugyanis a HDRI természetéből eredően környezeti textúra, mely nincsen mappolva, amely a bizonyos koordinátákat jelöli a textúrán, tehát nem reagál a kamera mozgására, nem lehet sem méretezni, sem jól formálni.



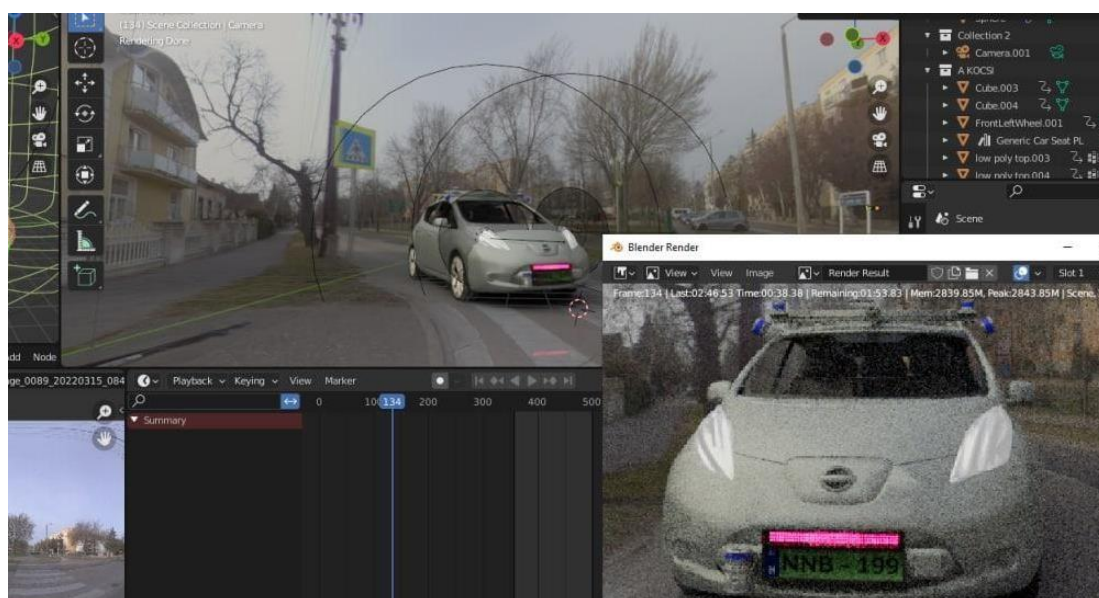
7. ábra - Kiterített 360°-os kép Blenderben (saját)

Hosszas gondolkodás után arra a megállapításra jutottunk, hogy érdemes volna egy basic mash-t addig deformálni, míg az arra felhúzott fotó textúra el nem éri a kívánt formát. Ehhez egy gömböt félbe vágtunk a Blender alkalmazás segítségével, subdividoltuk a fotó felbontásával egyenesen arányosan és a gömbre felhúztuk a fotónkat textúra formájában. Ezt a textúrát invertáltuk, hogy ne a gömbtől kifelé, hanem annak középpontja felé nézzen, tehát természetes képet adjon a nézőnek. A gömb felszínét shadow catcher tulajdonsággal láttuk el, hogy a benne lévő objektumok tudjanak árnyékot vetni annak felszínére, így a nézőnek még természetesebbnek tűnik majd a modelltér.



8. ábra - Virtuális kísérleti tér térhálója Blenderben (saját)

A gépjárműveket (Nissan Leafet) GitHub projektjében megtalálható alaprajzok és modellek alapján elkészítettük a Blender 3D nyíltforráskódú modellező szoftverben. A projekt bőséges dokumentációval rendelkezik, amely nagyban hozzájárult a modellezés folyamatához. A modellezés részeként az elkészült modellt és annak objektumait fotorealistikus textúrákkal és mozgási animációkkal láttuk el [9].



9. ábra - Animáció renderelés Blenderben. (saját)

Kezdetben jó ötletnek tűnt a fenti megoldás, hiszen a szimulációban résztvevő alanyokat nem fogja zavarni a virtuális autó alacsony felbontása és a nem teljes pontosságú megjelenése.

A szimulációk futtatására több lehetséges mód került megvitatásra. Az első lehetőség a kezdeti tervezet volt, miszerint a modellteret videó formájában exportáljuk, majd a videót egy 360 fokos videó lejátszására képes alkalmazásban játsszuk le a résztvevőknek, így a videók kellően magas felbontásúak maradnak és nem kell engedni a textúrák minőségéből sem. A másik variáció az volt, hogy a Blenderen belül játsszuk le a szimulációkat, viszont ehhez egy rendkívül erős számítógép kellett, amit a laborban sikerült beszerezni. Sajnos a módszert, amiben a videót exportáltuk volna, nem sikerült megvalósítani, mivel egy kritikus hibába ütköztünk. Nem találtunk nyílt forráskódú 360 fokos videólejátszót, amely jól működött volna. Megoldásként kipróbáltuk a YouTube által alkalmazott cupboard módszert is, ahol egy telefonon játsszuk le a videót, amelyet a néző fejére rögzítünk. Ezzel az volt a probléma, hogy rettentően leromlott a minőség, annyit veszítettünk a minőségből, mintha az alap modellen textúrákat váltunk [10]. Így a megoldás az lett, hogy csak Principled BDSF textúrákat és fotó textúrákat alkalmaztunk a modellekre, így egy szolid és állandó 60 FPS lejátszási sebességet kaptunk a Blenderben való szimuláció során is [11].

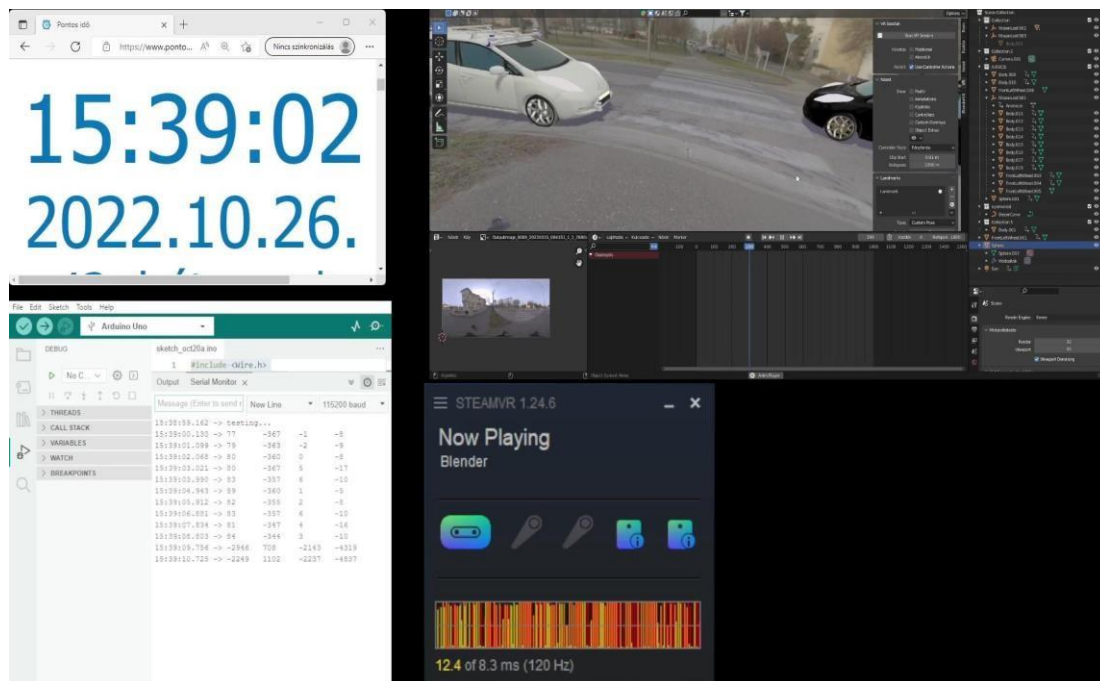
5.2 VR addon használata

A Blender kiegészítő listáján belül, 2022. eleje óta nyilvánosan elérhető mind VR és mind az XR kiegészítő is, melyek 3D terek vizsgálatához alkalmazhatóak. Kisebb utánajárás után rájöttünk, ez a funkció a Unity 2019-es szerkesztőjében már elérhető volt. Sajnos, ha Unity-re váltottunk volna, az egész projektet újra kellett volna gondolni, így maradtunk a Blendernél. A VR kiegészítőt egyszerűen engedélyeztük a preferenciákon belül és onnantól kezdve az alkalmazás kommunikált a steamVR alkalmazással, amin belül automatikusan indultak a szimulációk az animáció lejátszásának hatására. A kiegészítő segítségével a tesztalanyok magasságához képest be tudtuk állítani a kamera magasságát, elfordulási szögét, helyzetét és egyéb tulajdonságokat, ezáltal minden tesztalany fizikai adottságaihoz tudtunk igazodni.

5.3 HTC Vive

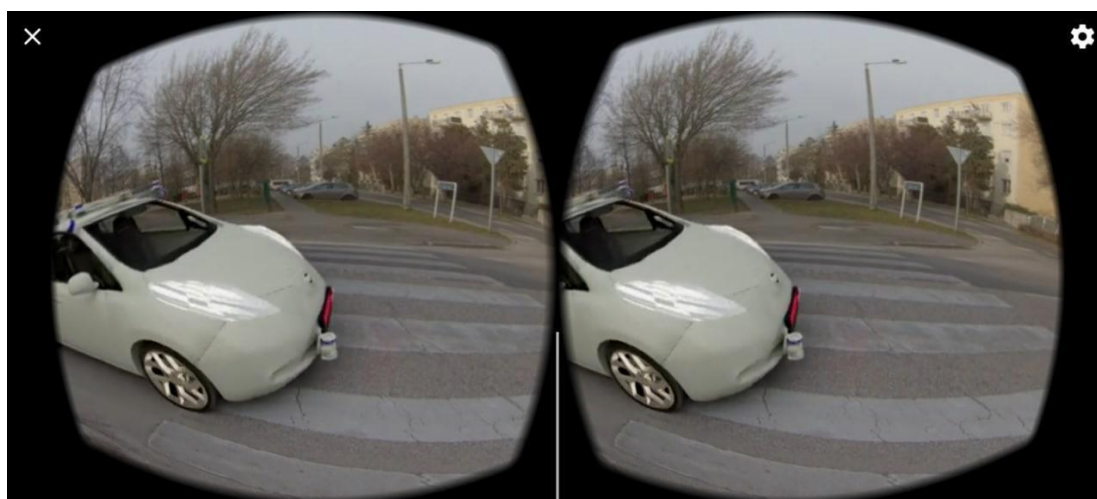
A HTC Vive Pro 2, melyet a Közlekedéscépesítési és Vízmérnöki Tanszék biztosított számunkra kiváló eszköznek bizonyult a kísérleti alanyok viselkedésének tesztelésére. Szerencsére az eszköz széleskörűen támogatott, így tudtunk integrálni a használt szoftverekbe. Lehetőségünk nyílt rögzíteni, amit a kísérleti alanyok láttak a szemüvegen belül a szimulációk során és azt valós

időben figyelni. Ha esetleg probléma adódott a hardver túlterheltsége miatt, azt jól nyomon tudtuk követni a monitorozási funkcióknak köszönhetően és egyből reagálhattunk. A HTC Vive pro soron következő verziója egy nagyon hasznos kiegészítővel jár, mellyel lehetséges a teszthalanyok pupilláját valós időben követni. A kísérlet jövőjében szeretnénk ezt a kiegészítőt alkalmazni és külön kimutatást végezni a pupilla mozgásról, hiszen ezzel a funkcióval lehetőségünk nyílna biztonsággal megmondani mire figyel pontosan a járókelő [12].



10. ábra - Szemüveg nézet tükrözés kísérlet alatt (saját)

11.



12. ábra - VR szemüveg belő nézete. (saját)

6. Arduino Uno programozása a virtuális térben való lábmozgás adatgyűjtéséhez

A kutatás során a virtuális térben való lábmozgatásról is gyűjtöttünk adatokat, amelyhez saját megoldást fejlesztettünk ki. Az Arduino Uno mikrokontroller a láb gyorsulását és elmozdulását hivatott mérni, ezen fejezetben azt a kódot mutatjuk meg, mely ezt lehetővé teszi.

```
#include <Wire.h>
#include <TimerOne.h>
#include <time.h>

#define MPU9250_ADDRESS      0x68
#define MAG_ADDRESS          0x0C

#define GYRO_FULL_SCALE_250_DPS  0x00
#define GYRO_FULL_SCALE_500_DPS  0x08
#define GYRO_FULL_SCALE_1000_DPS 0x10
#define GYRO_FULL_SCALE_2000_DPS 0x18

#define ACC_FULL_SCALE_2_G      0x00
#define ACC_FULL_SCALE_4_G      0x08
#define ACC_FULL_SCALE_8_G      0x10
#define ACC_FULL_SCALE_16_G     0x18

void I2Cread(uint8_t Address, uint8_t Register, uint8_t Nbytes, uint8_t* Data) {
    Wire.beginTransmission(Address);
    Wire.write(Register);
    Wire.endTransmission();

    Wire.requestFrom(Address, Nbytes);  uint8_t
    index=0;
```

```

    while (Wire.available()) Data[index++]=Wire.read();
} void I2CwriteByte(uint8_t Address, uint8_t Register, uint8_t Data) {
    Wire.beginTransmission(Address);
    Wire.write(Register);
    Wire.write(Data);
    Wire.endTransmission();
} long int ti; volatile bool
intFlag=false; void setup()
{
    Wire.begin();
    Serial.begin(115200);
    Serial.println("testing...");
    I2CwriteByte(MPU9250_ADDRESS,29,0x06);
    I2CwriteByte(MPU9250_ADDRESS,26,0x06);
    I2CwriteByte(MPU9250_ADDRESS,27,GYRO_FULL_SCALE_1000_DPS);
    I2CwriteByte(MPU9250_ADDRESS,28,ACC_FULL_SCALE_4_G);
    I2CwriteByte(MPU9250_ADDRESS,0x37,0x02);
    I2CwriteByte(MAG_ADDRESS,0x0A,0x16);
    Timer1.initialize(10000);
    Timer1.attachInterrupt(callback);    ti=millis();
} long int
cpt=0;
void callback()
{
    intFlag=true;    digitalWrite(13,
digitalRead(13) ^ 1);
}

```

```

void loop()
{
  while
  (!intFlag);
  intFlag=false;

  Serial.print (millis()-ti,DEC);
  Serial.print ("\t");  delay(820);
  uint8_t Buf[14];
  I2Cread(MPU9250_ADDRESS,0x3B,14,Buf);
  // Accelerometer  int16_t ax=-
  (Buf[0]<<8 | Buf[1]);  int16_t ay=-
  (Buf[2]<<8 | Buf[3]);  int16_t
  az=Buf[4]<<8 | Buf[5];
  // Gyroscope  int16_t gx=-
  (Buf[8]<<8 | Buf[9]);  int16_t gy=-
  (Buf[10]<<8 | Buf[11]);  int16_t
  gz=Buf[12]<<8 | Buf[13];
  // Accelerometer
  Serial.print (ax,DEC);
  Serial.print ("\t");
  Serial.print (ay,DEC);
  Serial.print ("\t");
  Serial.print (az,DEC);
  Serial.print ("\t");
  // Gyroscope
  Serial.print (gx,DEC);
  Serial.print ("\t");
  Serial.print (gy,DEC);
  Serial.print ("\t");
  Serial.print (gz,DEC);
  Serial.print ("\t");
  uint8_t ST1;  do {

```

```

I2Cread(MAG_ADDRESS,0x02,1,&ST1);
}
while (!(ST1&0x01));
Serial.println("");
}

```

[13]

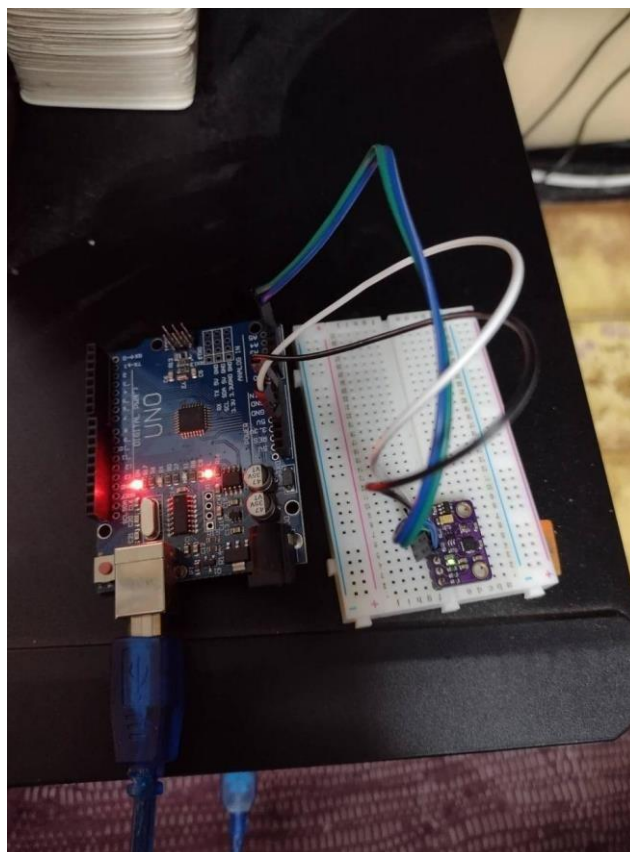
A fent található kód magyarázata a funkciók megértéséhez: az MPU9250 szenzor a gyorsulásmérőt és a giroszkópot használja. Az az alaplapon lévő chip címe MPU9250_ADDRESS, míg a MAG_ADDRESS annak az adatmemória-helynek a címe, amely információkat tartalmaz a giroszkóp forgási sebességéről (fok/másodperc). A kód három különböző teljes skálaértéket határoz meg mindegyik érzékelő számára: GYRO_FULL_SCALE_250_DPS, GYRO_FULL_SCALE_500_DPS és GYRO_FULL_SCALE_1000_DPS. Ezek az értékek megfelelnek annak, hogy az egyes érzékelők másodpercenként hány fokot tudnak mérni. Végül két változó szabályozza, hogy mennyi zaj adható hozzá az egyes érzékelőktől származó értékekhez: ACC_FULL_SCALE és ACC_. TELJES SZÁMÁRA.ACC_. A FULL SCALE beállítja, hogy milyen magas lehet egy adott érzékelő leolvasása mielőtt azt zajnak tekintenék. Ez segít nyomon követni, hogy mely értékek érvényesek, és melyeket kell elvetni, mert túl zajosak. A kód beállítja az MPU9250 gyorsulásmérő és giroszkóp használatához szükséges változókat. Az MPU9250_ADDRESS a kártyán lévő MPU9250 chip címe, míg a MAG_ADDRESS a GYRO érzékelőjének címe. Az ACC_FULL_SCALE változók minden tengelyhez beállítják a teljes skálatartományt. Például az ACC_FULL_SCALE_2_G azt jelenti, hogy a $2 \times 9,81 \text{ m/s}^2$ gyorsulási tartomány kerül felhasználásra. Az ACC_FULL_SCALE értéke 0 (gyorsulás nélkül) és 18 (teljes skálájú gyorsulás) között lehet. Az időzítő funkciók, ezalatt értjük az időpecsétek létrehozását, a TIMER One könyvtár meghívásával válnak lehetségessé.

A példában szereplő kód bájtokat olvas és ír egy I2C buszról. A kód a Wire objektum inicializálásával kezdődik, amely az I2C busszal kommunikál. Ezután a kód létrehoz egy átviteli hurkot, amely folyamatosan adatokat küld az I2C buszra, amíg az be nem záródik (azaz be nem fejeződik). Az első függvény, az I2Cread() bemeneti paraméterként egy címet és egy regiszterszámot vesz fel. Ezután ezeket az értékeket használja a megfelelő bájtérték

megkeresésére az Adatváltóban. Végül kiírja ezt a bájtértéket a konzol kimeneti adatfolyamába. A második függvény, az `I2CwriteByte()` bemeneti paraméterként egy címet és egy regiszterszámot vesz fel. Ezután szintén létrehoz egy átviteli hurkot, amely folyamatosan adatokat ír az I2C buszra, amíg az be nem záródik (azaz be nem fejeződik). Az egyetlen különbség a függvény és az `I2Cread()` között az, hogy ahelyett, hogy kinyomtatná az eredményeket, ez a függvény egyszerűen csak közvetlenül az adatváltóba írja őket köztes feldolgozás nélkül. A kód bájtokat olvas és ír az I2C buszról a megadott címen és regisztrál. A kód a Wire könyvtár inicializálásával kezdődik. Ezután a soros port inicializálása megtörténik, és a rendszer egy üzenetet nyomtat, amely megerősíti, hogy működik. Ezután az I2C könyvtár inicializálódik. A 0x06 cím első két bájtja az MPU9250_ADDRESS-be van írva, ami a chip működési módját 6 tengelyes giroszkóp/gyorsulásmérő módra állítja. A következő két bájt 1000 Hz-re állítja a mintavételezési frekvenciát, és lehetővé teszi a teljes léptékű kimenetet (ACC_FULL_SCALE_4_G). Ezután az 1. időzítő inicializálódik 10 másodperces időtartammal. Ez lehetővé teszi számunkra, hogy elvégezzünk néhány alapvető tesztelést, mielőtt ténylegesen elkezdjük a projektünket. Most pedig nézzük meg, hogyan működik ez a kód részletesebben: amikor adatokat írunk egy integrált áramkörü rendszerre, például az MPU9250-esre, akkor az órajelet (CLK) és az adatjelet (DAT) is tartalmaznia kell az eszköznek. Ebben az esetben CLK=5V(órajel) és DAT=0x37(adatjel). A kód a Wire könyvtár inicializálásával kezdődik. Ezután beállítja a soros portot és egy üzenetet nyomtat, amely megerősíti, hogy minden működik. Ezután a kód néhány értéket ír az I2C buszra az MPU9250 0x06 címével, valamint a különféle beállításokkal. Ezek az értékek közé tartozik a giroszkóp teljes skála (1000 dps), az ACC (teljes skála 4 g). A következő kódsor beállít egy időzítőt, amely 10 000 ezredmásodpercig fut.

A kód két változó inicializálásával kezdődik: `ti` (timer) és `cpt` (Kapacitív teljesítményátvitel inicializáló változó). A `timer1` objektumot ezután a `callback()` függvény inicializálja. Ez a függvény 1/2 másodpercenként kerül meghívásra. Ebben a függvényben az `intFlag` változó igazra van állítva, ha több, mint 1 másodperc telt el a `timer1` objektum utolsó hívása óta. Ha az `intFlag` értéke `false`, akkor a `digitalWrite()` függvény meghívásra kerül a 13. LED 1/2 másodpercre való bekapcsolásához. Végül a `millis()` kiírásra kerül a soros monitorra. `Serial.print("\t")`-t használunk, annak érdekében, hogy minden kimeneti időpecsét külön sorban jelenjen meg a soros monitor ablakán; úgy kerül végrehajtásra, hogy minden kimeneti sor külön sorban jelenjen meg a soros monitor ablakában. A kód tartalmaz egy ciklust

is, amely folyamatosan ellenőrzi, hogy az `intFlag` igaz vagy hamis. Ha hamis, akkor a `digitalWrite()` újabb hívása 1/2 másodpercre kikapcsolja a 13-as LED-et, és a kód visszatér a `main while loop()`-ba. Ellenkező esetben az `intFlag` igaz marad mindaddig, amíg ismét `false` nem lesz (ami 2 másodperc elteltével történik meg). A kód egy időzítőt hoz létre, amely 1/2 másodpercenként túlsordul. A kód beállít egy flag-et (zászlót), amely jelzi, ha az időzítő túlsordult. A flag beállítása után a kód aktiválja a `digitalWrite(13)`-as áramkört, amely hatására a 13. érintkezőn lévő LED kigyullad. A kód ezután visszaállítja a jelzőt `false` értékre, hogy ismét folytathassa a ciklus futtatását. A kód először az MPU9250 címét olvassa be az I2C-ből. Ezután 14 bájtot olvas ki a címből, amelyek a gyorsulásmérő és giroszkóp értékei. A kód először úgy számítja ki a gyorsulásmérő értékeit, hogy minden érték 8. bitjét kivonja 255-ből (0xFF). Az eredmény egy 0 és 3 közötti tartomány. Ezután minden érték 9. bitjét veszi, és kivonja 128-ból (0x80), így -128 és 127 közötti tartományt kap. Végül minden érték 10. bitjét veszi, és kivonja 64-ből (0x40), így -256 és 255 közötti tartományt kap. Ezután a kód kiszámítja a giroszkóp értékeit úgy, hogy az egyes értékek 8. bitjét kivonja 256-ból (0xFF). Az eredmény egy 0 és 3 közötti tartomány. Ezután minden érték 9. bitjét veszi, és kivonja 128-ból (0x80), így -128 és 127 közötti tartományt kap. Végül minden érték 10. bitjét veszi, és kivonja 64-ből (0x40), így -128 és 128 közötti tartományt kap. A kód először a gyorsulásmérő és giroszkóp érzékelők értékét olvassa ki. A gyorsulásmérő érzékelő 8 bites értéket ad vissza, amely a gyorsulás nagyságát Gs-ben (gravitációs erő) jelzi. A giroszkóp érzékelő 16 bites értéket ad vissza, amely a szögsebességet reprezentálja radián/másodpercben. A kód ezután kiszámítja az értékeket az egyes tengelyekhez úgy, hogy kivonja a kezdeti leolvasást a végső olvasásból. Ezután ezeket az értékeket nyomtatja a soros monitorra, ahonnan másolásra kerülnek az értékek későbbi elemzésre. A magnetométer érzékelő alkalmazását elvetettük, ugyanis értékei nem relevánsak a vizsgálat szempontjából. [14]



12. ábra - Arduino tesztelésre készen (saját)

7. Virtuális valóságban történő tesztelés folyamata

7.1 Helyszín

Tesztelési helyszínnek a Széchenyi István Egyetem GeoInformatikai laborján belül található VR Labor lett megjelölve, ahol az első scénáriókat már augusztusban elkezdjük végrehajtani a tesztalanyokkal. Egy tesztalany által elvégzett teszt eleinte 15-20 percet vett igénybe, ezt az időt az eltelt hónapok alatt sikerült 3-4 percre redukálnunk, természetesen minőségromlás nélkül. Az idő redukálása köszönhető annak, hogy időközben számítógépet váltottunk és a folyamatok nagy részét leegyszerűsítettük, finomítottuk.

7.2 Lebonyolítás

A tesztelési környezetet és rendszert úgy alakítottuk ki, hogy a tesztelés a tesztalanyoknak kényelmes legyen, a mérés pontos legyen, és minden használt eszközt biztonságosan tudjunk használni. A labor galériáját tartó gerendákra a HTC Vive szenzorokat felerősítettük fix pozícióban, hogy azok mindig egy helyen legyenek a tesztelés folyamata alatt (180-190 cm magasságban).



13. ábra - Tesztelési folyamat laborkörnyezetben (saját)

A tesztalanyt a szenzorok metszésében a tér közepén leültették, fejére helyezték a virtuális valóság szemüveget, majd a lábára az Arduino Uno mozgásérzékelő szenzort. Ezek után elmagyaráztuk mit kell majd tегyenek, ugyanakkor igyekeztünk kerülni a szituációval kapcsolatos konkrétumokat, hogy ne befolyásoljuk a döntéseiket a virtuális térben. Minden tesztet videóra rögzítettünk, hogy később az elemzés folyamán, ha bizonytalanságok lépnének fel, azokat egyszerűen tisztázni tudjuk. Minden teszt ugyanazon forgatókönyv alapján szabályosan folyt le, ahol a folyamat vége az adatok mentésével zárult.



14. ábra - Arduino Uno lábra erősítése (saját)

7.3 Adatelemzés megközelítési módszere

Az első ütemezésben a rendszert 51 ember bevonásával sikerült tesztelni. Ezen tesztalanyok Magyarország demográfiai eloszlását követik és reprezentálják. A második ütemezésben csak fiatalokat vontunk be a folyamatba, hogy megnézzük mekkora különbség lehet a fiatalok önvezető járművekbe vetett bizalmában, a demográfiailag befolyásolt eredményekhez képest.



15. ábra - Tesztelési folyamat laborkörnyezetben 2. (saját)

8. Tesztelési kimenetek feldolgozása

8.1 Microsoft Office programcsomag: Excel 2019

Minden egyes tesztalany adatai egy különálló Excel táblába kerültek mentésre. Az Excel file elnevezése a tesztelt személy vezeték és keresztnévéről tevődött össze minden esetben. A .xls fájlokban belül négy egység található, ezen egységek reprezentálják az egyes szcenáriókat. A szcenárió hosszától függően ezen egységek hosszabbak vagy rövidebbek.

A mozgási adatok tartalmazzák az alanyok lábának gyorsulását és elmozdulását X és Y tengelyeken. Ahhoz, hogy meg tudjuk mutatni az eredmények szélső és medián értékeit a mozgási adatok abszolút értékét véve mátrix addíciót végeztünk. Azt szeretnénk volna elérni, hogy ahol a mozgási adatok intenzívek, ott az addíció során legyenek még intenzívebbek, és ahol a mozgási adatok gyengék, legyenek még gyengébbek. Ebből következik, hogy nem kellett volna abszolút értékeket vennünk, viszont nem szeretnénk volna nulla érték alá menni, így szeretnénk volna elkerülni a vizuális reprezentáció folyamán a zéruspont alatti értékeket. Ebben az esetben, ahol erős mozgás látható, ott a reprezentáció során egyértelműen látszik a leggyorsabb és legelső reakcióidő akár csak a leglassabb és leggyengébb a zérust megközelítve. A kapott eredményeket egy összegző táblázatba vontuk, majd azokat egyszerű pont és lineáris gráfokkal reprezentáltuk.

8.2 Virtuális valóság szcenáriók szerint átlagolt kimutatás

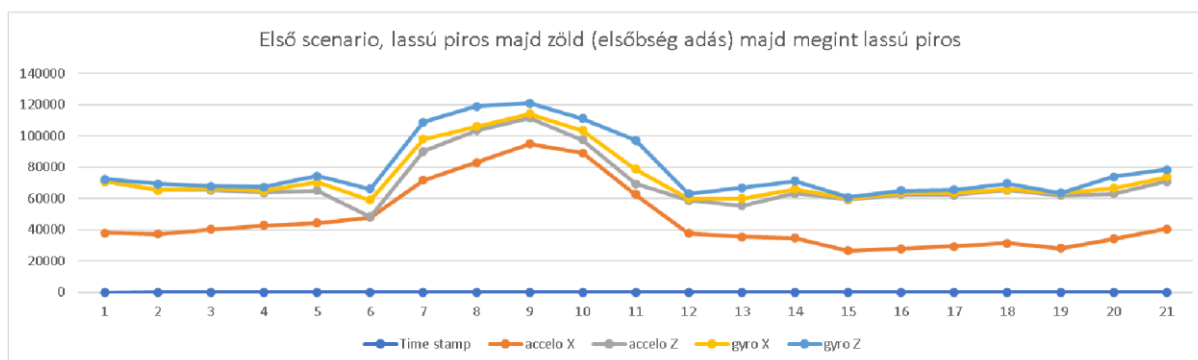
8.2.1 Első ütemezés kimenetelei

A táblázatban az összes átkelő első szituációra való reakciójának kumulált értékei láthatók, melyeket grafikusán is ábrázoltunk.

Time stamp	acelo X	acelo Z	gyro X	gyro Z	status
1	37957	32819	170	1470	Not safe to pass
2	37322	28090	21	3929	
3	40246	25066	1024	1459	
4	42801	20899	1177	2427	
5	44438	20558	5284	4021	

6	47963	359	10560	7057	
7	71701	18278	8021	10911	Safe to pass
8	83127	20446	2364	12924	
9	94776	16859	2399	6835	
10	89154	8330	6008	7648	
11	62582	6771	9276	18458	
12	37798	21009	1212	2959	Not safe to pass
13	35514	19924	4561	6744	
14	34783	28440	2814	5051	
15	26498	32718	601	977	
16	27808	34632	965	1701	
17	29284	32959	1285	1911	
18	31403	34012	275	3943	Safe to pass
19	28192	33817	1196	330	
20	34065	29017	3431	7337	
21	40570	30215	2964	4586	

Első scenárió, lassú piros majd zöld (elsőbség adás) majd megint lassú piros
sebesség: ~ 20km/h

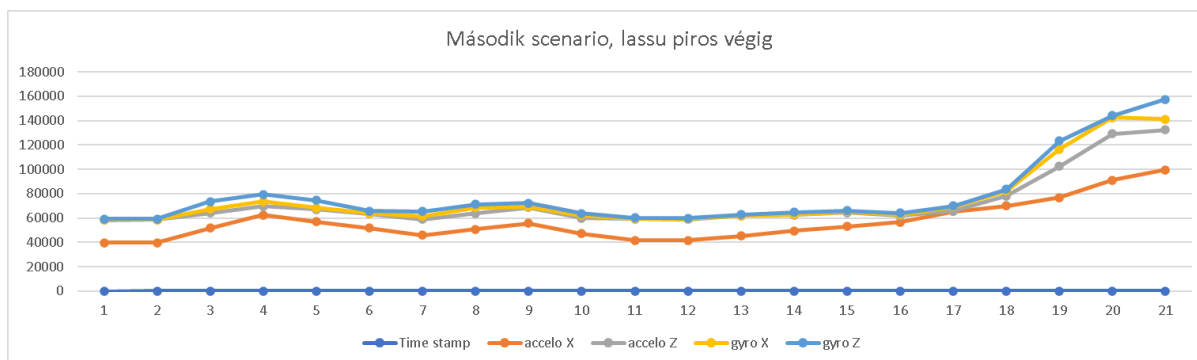


16. ábra – Első ütemezés első scenáriójának diagramja (saját)

A grafikonról leolvashatjuk a szélsőértékeket és a mediánt. A tesztalanyok nagy része a 6. és 12. másodpercek között kezdte meg az átkelést. Ezen időszak alatt az autó megadja az elsőbbséget, így a gyalogosok biztonságosan kelhetnek át az útszakaszon. A tesztalanyok egy kisebbik része csak az autók elhaladása után kezdi meg az átkelést a 19. másodperctől.

A táblázatban az összes átkelő második szituációra való reakciójának kumulált értékei láthatók, amelyeket grafikusán is megjelenítettünk.

Time stamp	acelo X	acelo Z	gyro X	gyro Z	status	
1	39843	18430	240	727	Not safe to pass	
2	39899	18606	140	636		
3	51944	12117	3309	6133		
4	62588	7170	3798	5763		
5	56853	10296	1089	6296		
6	51846	11788	58	2066		
7	45947	12979	2500	3883		
8	50867	13007	4865	2425		
9	55681	13072	557	2859		
10	47150	12760	2155	1829		
11	41726	17618	152	832	Safe to pass	
12	41796	17292	189	805		
13	45288	16639	37	833		
14	49653	12930	426	1655		
15	53055	11335	1053	776		
16	56578	5230	888	1582		
17	65310	486	3637	422		
18	69918	8165	4145	1410		
19	76969	25697	13702	6976		
20	91072	38275	13176	1589		
21	99556	32794	8919	16301		
Második szcenárió, lassú piros végig sebesség: ~ 20km/h						

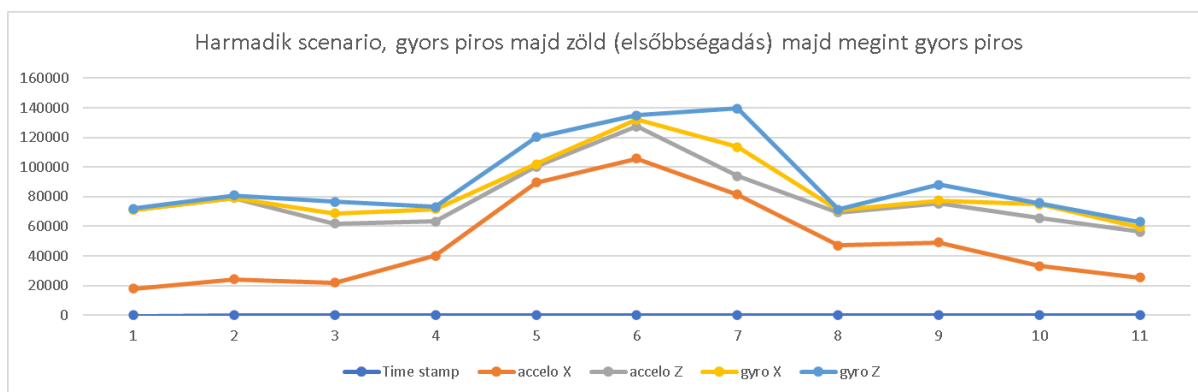


17. ábra - Első ütemezés második scenáriójának diagramja (saját)

A grafikonról leolvashatjuk a szélsőértékeket és a mediánt. Látható, hogy a gyalogosok egy kis része a 2. és 6. másodpercek között indult el, itt az autó nem állt meg és nem is adott elsőbbséget, ezek a gyalogosok a lassú sebességre hivatkozva indultak útnak. A túlnyomó többség a 17. és 21. másodpercek között kelt át a szimulált gyalogos átkelőhelyen. Őket a biztonságos kategóriába sorolhatjuk.

A következő táblázatban az összes átkelő harmadik szituációra való reakciójának kumulált értékei láthatók, melyeket grafikusan is ábrázoltunk.

Time stamp	acelo X	acelo Z	gyro X	gyro Z	status
1	17866	53083	75	994	Not safe to pass
2	24192	55013	75	1468	
3	21840	39964	7066	7566	
4	40211	23059	8410	1428	
5	89566	10791	1620	18190	Safe to pass
6	105835	21687	4665	2864	
7	81550	12246	19642	26023	
8	46983	22375	1393	659	Not safe to pass
9	49096	26223	2217	10606	
10	33055	32618	9016	875	Safe to pass
11	25432	30916	3276	3405	
Harmadik szcenárió, gyors piros majd zöld (elsőbbségadás) majd megint gyors piros sebesség: ~ 40km/h					

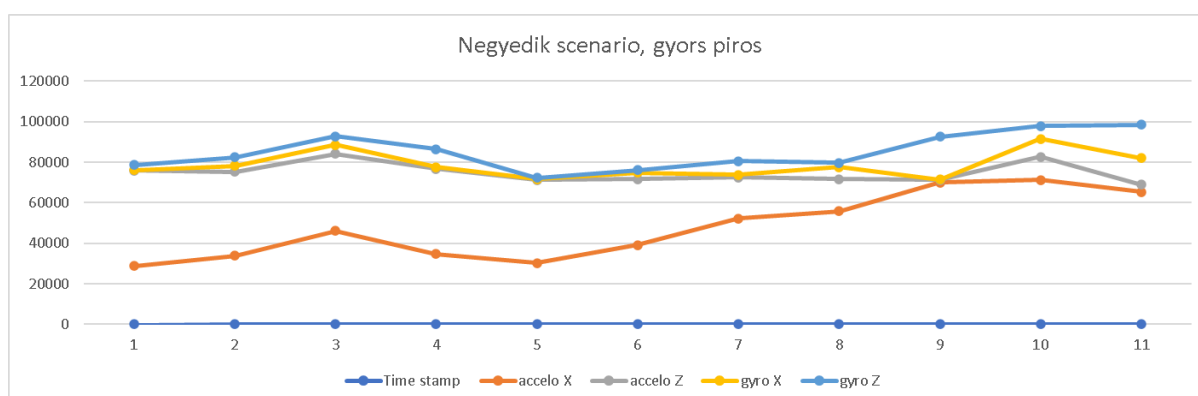


18. ábra - Első ütemezés harmadik szcenáriójának diagramja (saját)

A grafikonból leolvashatjuk a szélsőértékeket és a mediánt. A gyalogosok egy része a lassú közeledésre való tekintettel a 3. és 4. másodpercek között elindult, a túlnyomó többség viszont csak az 5. és 8. másodpercek között kelt át, valamint elenyésző mennyiség a 8. és 10. másodpercek között.

Time stamp	acelo X	acelo Z	gyro X	gyro Z	status
1	28846	47110	60	2697	Not safe to pass
2	33768	41571	2853	4173	
3	46091	37983	4519	4246	
4	34643	42162	743	9011	
5	30203	41037	534	478	Safe to pass
6	39027	32585	3221	1270	
7	52214	20339	1215	6696	
8	55767	15883	5870	2120	
9	70026	1115	229	21232	
10	71329	11323	8900	6284	
11	65327	3634	12975	16599	
Negyedik szcenárió, gyors piros sebesség: ~ 40km/h					

A táblázatban az összes átkelő negyedik szituációra való reakciójának kumulált értékei láthatók, melyeket grafikusán ábrázoltunk.



19. ábra - Első ütemezés negyedik scenáriójának diagramja (saját)

A grafikonon látható, hogy ennél a scenáriónál kifejezetten vegyesek lettek az eredmények. Ez a kimenetel annak tudható be, hogy rövid ideig volt veszélyes az átkelés és utána relatíve hosszán biztonságos. A gyalogosok egy kisebb része a 2. és 4. másodpercek között kezdte meg az átkelést, ez még nem mondható biztonságosnak, hiszen az autó nagy sebességgel közeledik és nem ad elsőbbséget. A biztonságos átkelők az 5. másodperctől kezdték meg az áthaladást, mikor a jármű már elhagyta az átkelőt. A maradék, szélsőértéket reprezentáló csoport a 9. másodperctől egészen a kísérleti időintervallum végéig haladt át.

8.2.2 Második ütemezés kimenetelei

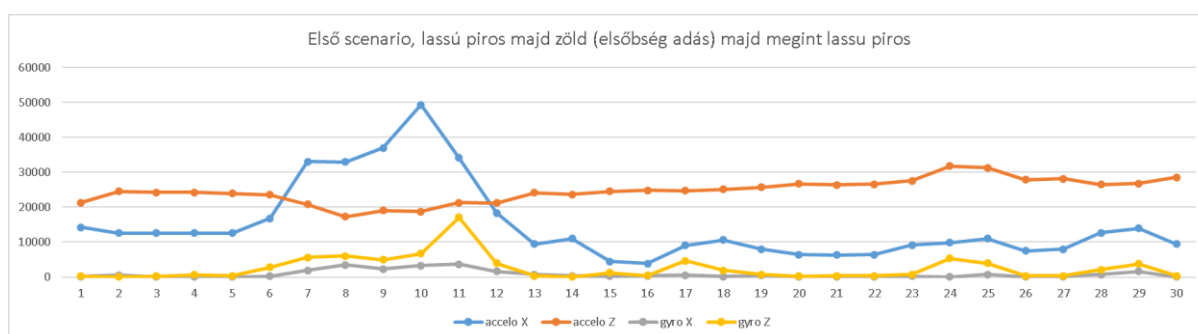
A táblázatban az összes átkelő első szituációra való reakciójának kumulált értékei láthatók, melyeket grafikusán is ábrázoltunk.

Time stamp	acelo X	acelo Z	gyro X	gyro Z	status
1	14227	21282	101	266	Not safe to pass
2	12568	24568	496	62	
3	12587	24278	24	218	
4	12603	24269	86	595	
5	12564	23942	19	318	
6	16820	23610	201	2758	
7	33057	20855	1890	5616	
8	33008	17341	3502	6012	Safe to pass
9	37025	19025	2343	4932	

10	49390	18779	3284	6735	
11	34131	21275	3674	17063	
12	18332	21194	1637	4018	
13	9503	24162	767	359	
14	11009	23699	354	23	
15	4450	24564	226	1230	Not safe to pass
16	3878	24837	306	480	
17	9086	24763	497	4693	
18	10658	25135	111	1875	
19	7946	25729	475	691	
20	6422	26705	179	260	
21	6366	26410	48	376	
22	6389	26659	96	322	
23	9124	27585	229	707	
24	9890	31765	79	5356	Safe to pass
25	11042	31317	762	3923	
26	7513	27839	98	368	
27	7946	28163	109	358	

28	12690	26555	710	2147	
29	14001	26802	1602	3790	
30	9472	28597	65	347	

Első szcenárió, lassú piros majd zöld (elsőbbsegadás) majd megint lassú piros
sebesség: ~50km/h



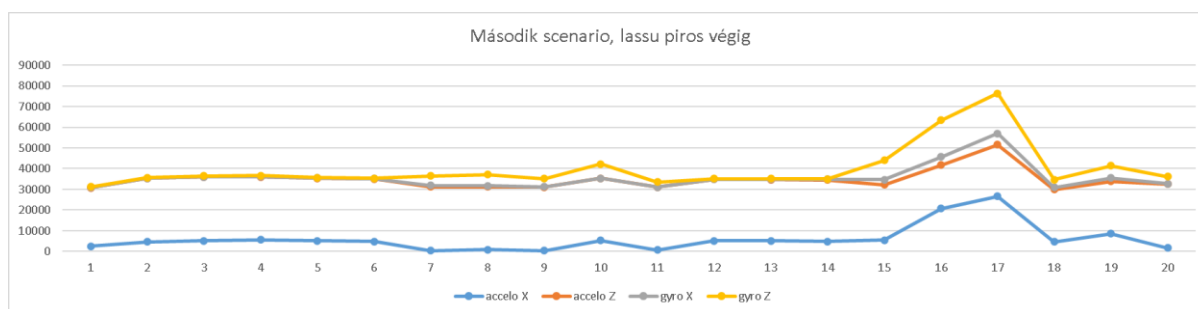
20. ábra - Második ütemezés első szcenáriójának diagramja (saját)

A grafikonból leolvashatjuk a szélsőértékeket és a mediánt. A tesztalanyok kis része már az 5. és 6. másodpercben biztosnak találta az átkelést, mikor a jelzésünk még piros volt, de az autó már lassított. Az átkelők nagyrésze (a medián) a 10. másodpercnél kezdte meg az átkelést, mikor a jelzés már zöld volt és a kocsí teljesen megállt. Látható egy apróbb törés a 27. és 30. másodperc között, ezek a gyalogosok csak akkor keltek át, mikor már fényjelzéstől függetlenül minden jármű elhagyta az átkelőhelyet.

A táblázatban az összes átkelő második szituációra való reakciójának kumulált értékei láthatók, melyeket grafikusán is ábrázoltunk.

Time stamp	accelo X	accelo Z	gyro X	gyro Z	status
1	2522	28206	96	391	Not safe to pass
2	4621	30666	17	330	
3	5195	30728	95	458	
4	5709	30332	101	482	
5	5155	30245	113	123	
6	4779	30271	121	226	
7	355	30929	543	4613	
8	904	30269	620	5257	
9	387	30686	228	3880	
10	5327	29990	0	6967	
11	762	30318	8	2398	
12	5097	29658	75	316	Safe to pass
13	5168	29493	165	258	

14	4887	29675	314	82	
15	5431	26714	2573	9226	
16	20746	20973	3986	17656	
17	26607	24926	5447	19265	
18	4656	25250	923	3824	
19	8649	25217	1673	5913	
20	1642	30876	156	3391	
Második scenárió, lassú piros végig sebesség: ~50km/h					



21. ábra - Második ütemezés második scenáriójának diagramja (saját)

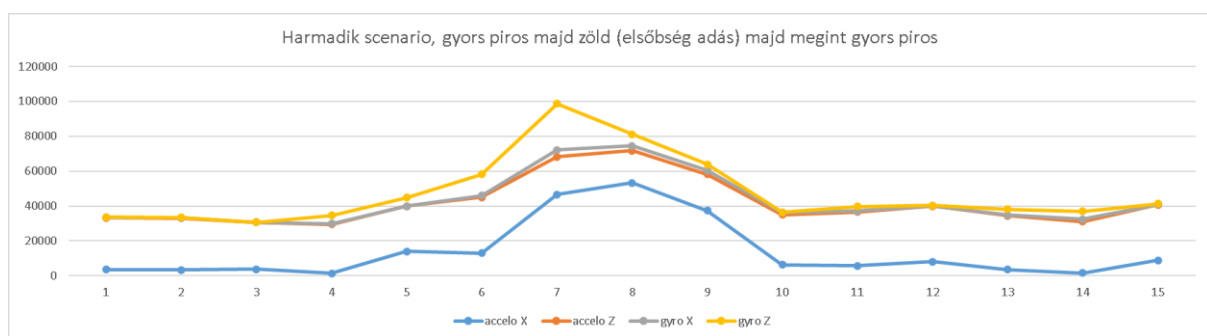
A grafikonból leolvashatjuk a szélsőértékeket és a mediánt. A tesztalanyok kis része már a 10. másodpercben biztosnak találta az átkelést, mikor a jelzés még piros volt és az autó nem lassított, hiszen nem adta meg az elsőbbséget. Az átkelők nagyrésze (a medián) a 15. és 18. másodperc között kezdte meg az átkelést, mikor az autók teljesen elhaladtak.

A táblázatban az összes átkelő harmadik szituációra való reakciójának kumulált értékei láthatók, melyeket grafikusán is ábrázoltunk.

Time stamp	acelo X	acelo Z	gyro X	gyro Z	status
1	3510	29779	94	291	Not safe to pass
2	3406	29739	26	317	
3	3790	26852	48	4	
4	1395	28096	465	4701	Safe to pass
5	13993	25864	164	4797	
6	13091	31929	1059	12090	
7	46680	21451	4096	26426	
8	53234	18542	2727	6770	
9	37461	20687	2299	3385	Not safe to pass
10	6330	28700	1167	145	
11	5689	30916	491	2627	
12	8080	31891	77	352	

13	3607	30786	495	3356	Safe to pass
14	1705	29310	1542	4327	
15	8804	31873,5	264,5	418,5	

Harmadik szcenárió, gyors piros majd zöld (elsőbbségadás)
majd megint gyors piros sebesség: ~65km/h



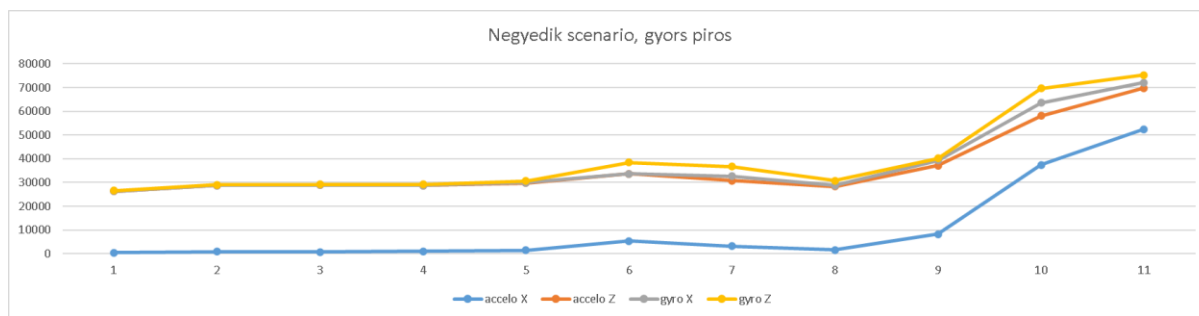
22. ábra - Második ütemezés harmadik szcenáriójának diagramja (saját)

A grafikonból leolvashatjuk a szélsőértékeket és a mediánt. A tesztalanyok kis része már a 4. és 6. másodpercek között biztosnak találta az átkelést, mikor a jelzés már zöld volt és az autó már lassított. Az átkelők nagyrésze (a medián) a 6. és 10. másodpercnél kezdte meg az átkelést, mikor a jelzés már zöld volt és a kocsí teljesen megállt. Látható egy apróbb törés a 16. és 15. másodperc között, ezek a gyalogosok csak akkor keltek át, mikor már fényjelzéstől függetlenül minden jármű elhagyta az átkelőhelyet.

A táblázatban az összes átkelő negyedik szituációra való reakciójának kumulált értékei láthatók, melyeket grafikusán is ábrázoltunk.

Time stamp	acelo X	acelo Z	gyro X	gyro Z	status
1	468	25880	6	228	Not safe to pass
2	864	27889	52	238	
3	778	28052	65	323	
4	1054	27693	66	378	
5	1419	28364	84	691	
6	5400	28221	15	4740	
7	3209	27534	1898	4014	Safe to pass
8	1605	26767	365	1982	
9	8253	28896	2128	838	
10	37472	20572	5536	6013	
11	52445	17416	2254	3205	

Negyedik scenário, gyors piros
sebesség: ~65km/h



23. ábra - Második ütemezés negyedik scenáriójának diagramja (saját)

A grafikonból leolvashatjuk a szélsőértékeket és a mediánt. A tesztalanyok kis része már az 5. és 7. másodpercek között biztosnak találta az átkelést, mikor a jelzés piros volt és az autó nem is lassított, hiszen nem adta meg az elsőbbséget. Azok a gyalogosok, akik még nem keltek át, azok egyértelműen csak az autók elhaladásakor a 8. és 11. másodpercek között keltek át.

9. Valós környezetben végzett felmérések tervezése, megvalósítása

A virtuális valóságban elvégzett tesztekől és minta esetekből konzekvenciákat levonva, majd közös ötletelés után egyértelművé vált számunkra, hogy a valóságban is meg szeretnénk építeni a kísérlet prototípusát, azaz a fényjelzéseket felszerelni és magát a valós szituációt rekonstruálni, hogy a projekt fejlődése biztosított lehessen.

Első lépésként a kivitelezési lépésekkel kapcsolatos kérdéseket jártuk körbe. Mindjárt az elején jogi problémába ütköztünk, ugyanis közúton különleges rendőrségi engedély nélkül tilos bármely, a megszokottól eltérő fényjelzést felszerelni a járműre, ugyanis ez megtévesztő lehet és a KRESZ ezt kifejezetten tiltja.

A problémát rendőrségi engedély kéréssel háríthattuk volna el, azonban idő szűkében azt láttuk a legjobbnak, ha magánterületen tesztelünk majd.

Dr. Élő Gábor és Dr. Makó Emese közreműködésével hozzájutottunk egy LED-sorhoz, amely képes többek között zöld és piros színben világítani, továbbá jobbról és balról induló terelőnyilakat is képes kirajzolni, amely szintén egy további értékes fejlesztési lehetőséget vetett fel.



24. ábra - Valós kivitelezés saját járművön (saját)

Mindenképpen szeretnénk volna fedélzeti kamera segítségével rögzíteni a GPS koordinátákat, a sebességet, valamint a gépjármű mozgását képpel és hanggal együtt a későbbi elemzések végett. Ezen törekvésünk okozta a következő problémát, ugyanis a LED-panel és a fedélzeti

kamera is szivargyújtós csatlakozóval volt ellátva. A megoldást egy új adatkábel és egy külső akkumulátor jelentette. Ezután kezdődhetett a LED-panel rögzítése, melyet kábelkötegelővel oldottunk meg, ugyanis ez stabil és diszkrét vázat nyújtott. Miután minden készen állt, beültünk az autóba, majd rácsatlakoztunk a LED-panel hálózatára és egy belső hálózati cím segítségével irányítottuk azt. A színek váltása közötti késés szinte elhanyagolható, 0,4 másodpercesnek bizonyult.

Forrasztottunk egy saját csatlakozót a LED sorhoz, amely elméletileg képes lett volna meghajtani a lámpát számítógépről USB 3.0-án keresztül. Sajnos ez a megoldás nem volt eredményes, mert az USB maximum 0.5A áramerősséggel rendelkezik. A megoldás végül egy olyan „krokodil” csatlakozóval ellátott kábel lett, melyet egy magasteljesítményű telefonos töltővel dugtunk a szivargyújtóba. Sajnos ez a megoldás jelenleg kényelmetlen, de a működés ezáltal biztosított. A végső, működőképes megoldás egy külső tápegység lett, amely elegendő áramerősséggel és feszültséggel rendelkezik.



25. ábra - Belső munkatér kísérlet alatt (saját)

Az egyetem területén a legtöbb hallgatónak és járókelőnek egyaránt érdekesnek bizonyult a projekt első fázisa, érdeklődve figyelték a járművet és az arra felszerelt fényjelzést, fényképeztek és érdeklődtek. Rengeteg biztatást kaptunk és a többség mindenképpen segíteni szeretne volna projektünk előrehaladását.

Ezen motivációtól további erőre kapva teszteltük és finomítottuk a kivitelezési módot. Megfigyeléseink szerint a gyalogosok több mint háromnegyede felfigyelt a kijelzőnkre. Megállapítottuk, hogy sebességüinktől függetlenül a zöld szín bizonyosságukra hozta, hogy megkapják a nekik járó elsőbbséget, valamint a piros szín egyértelműen jelezte feléjük, hogy az átkelés nem biztonságos.

10. A projekt jövője, fejlődési utak

10.1 Szem és fejmozgásból elemzett reakciók vizsgálata

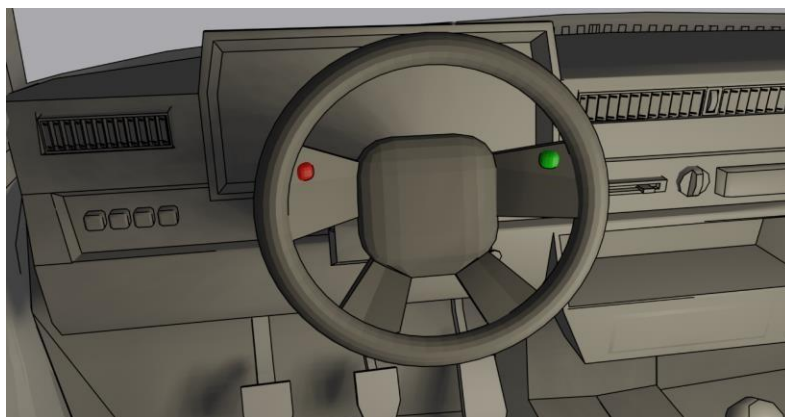
A HTC Vive általunk használt verziója nem rendelkezik pupilla követő, eye tracking funkcióval. Szeretnénk a jövőben egy olyan bővítménnyel dolgozni, amely erre képes. Szeretnénk megvizsgálni, hogy a tesztalanyok az előttük lejátszódó térbeli események mely részeire koncentrálnak a legjobban. Lehetőségünk nyílna kritikus helyzetek szimulálására és annak tanulmányozására, min érdemes javítani a való életben a balesetek elkerüléséhez. Ugyanakkor minden kísérlethez rögzítettünk videót, hogy láthassuk merre néz a tesztalany, ezen videók elemzéséből további konklúziókat tudunk levonni. Jelenleg a kísérlet ezen folyamata csak elméleti síkon létezik, sajnos idő hiányában nem tudtunk részletes és átfogó elemzést készíteni. Érdemes lehet majd azt is megfigyelni, hogy a tesztalanyok melyik irányba, melyik járműre figyelnek, és ez a folyamat részükről mennyi ideig tart.

10.2 Tesztesetek további megközelítési lehetőségei

Szeretnénk feldolgozni a korábbi tesztesetekből rendelkezésünkre álló videókat, amiből pontos elemzést kaphatunk arról, hogy a virtuális valóságban végbement forgalmi szituációkban miként és milyen mértékben befolyásolta a gyalogosokat az érkező autó sebességének és a LED színének együttese.

10.3 Kormányról való vezérlés, optimalizálás

Távlati célunk, hogy a panel meghajtása külön akkumulátorról működjön, ne függjön az autó akkumulátorától és a szivargyújtótól, ehhez azonban 12V és 2,5-3A szükséges. Továbbá szeretnénk megoldani a fényjelzés irányítását két gombbal, amely a kormányról lenne vezérelhető.



26. ábra - Konceptiós rajz Blenderben (saját)

11. Hipotézis alátámasztása vagy megcáfolása

Az elvégzett felmérések alapján elmondható, hogy a gyalogosok túlnyomó többsége megértette a feladatot. A vizsgálatok során az is egyértelműen megállapítható, hogy a tesztalanyok viselkedésük, reakciójuk alapján különböző osztályokba sorolhatóak, mint a bátor/vakmerő, biztonságos, avagy extra biztonságos típusok. A vakmerő gyalogosok, akik a szélsőérték korai átkelési részét képviselik. Ezen gyalogosok abban a tudatban kelnek át, hogy az elsőbbség mindenképpen őket illeti meg és a közlekedési helyzet majd hozzájuk alkalmazkodik. A medián csoportot képviselik a biztonságos gyalogosok, akik megvárták a pozitív fényjelzés megérkezését és az autók megállását. A második szélsőértékhez társítható csoport az extra biztonságos, akik nem bíztak meg a fényjelzésekben vagy azokat észre sem vették, így csak akkor keltek át, ha már nem volt autó az átkelő közelében.

A közlekedési szituációkban a jelzőlámpák fényjelzései egyértelműek a gyalogosok számára, tudatában vannak a színek jelentésének. Az önvezető járművön általunk elhelyezett, a közlekedési jelzőlámpáknál használt alkalmazott fényjelzések, a piros és zöld színek mögötti jelentéstartalom és az önvezető járműbe vetett bizalom között van relációs kapcsolat, a minta átlaga alátámasztotta az eredeti feltevést. Természetesen a modellt érdemes tovább finomítani, akár új, a gyalogosok viselkedésének objektív mérésére alkalmas paraméterekkel, akár a bizalmat befolyásoló egyéb megoldásokkal, eszközökkel.

12. Témához kötődő idegen és szakszavak szótára

Interfész - Az interfész (az angol interface szóból) két számítógépes (hardver vagy szoftver) eszköz, vagy a számítógép és az azt használó ember érintkezési felülete. Az interfész olyan megoldásokat tételez fel, amelyeket mindkét fél ért. **eHMI** - External human-machine interfaces/Külső ember-gép interfészek.

Mercedes-Benz AG - A Mercedes-Benz Group AG egy németországi székhelyű multinacionális járműgyártó vállalat. 1926-tól 1998-ig Daimler-Benz, 1998-tól 2007-ig DaimlerChrysler, 2007-től 2022-ig Daimler néven működött, 2022 óta Mercedes-Benz Group néven fut. Székhelye Stuttgartban található.

LED - A fénykibocsátó dióda (LED) egy félvezető eszköz, amely fényt bocsát ki, amikor áram folyik rajta. A félvezetőben lévő elektronok rekombinálódnak az elektronlyukakkal, és energiát bocsátanak ki fotonok formájában. Light Emitting Diod.

Pavlovi reflex - Pavlov az etetés előtt a kutyáknak bemutatott csengő hangját használta, és egy sor ismételten után az állatok nyáladzani kezdtek, amikor meghallották a csengőt, még akkor is, ha nem szolgáltak fel ételt. Feltételes reflexek.

8k felbontás - A 8K-ban a „K” a Kilo (1000) rövidítése, ami olyan felbontású képet jelent, amely körülbelül 8000 pixeles horizontális felbontást ért el.

HDRI - A fotózásban és a videózásban a több expozíciós HDR rögzítés egy olyan technika, amely lehetővé teszi nagy dinamikatartományú képek készítését ugyanazon tárgy több különböző expozíciójának elkészítésével, majd kombinálásával. A kamerákkal rögzített képek csak a fényerő bizonyos tartományán belül teszik lehetővé a megkülönböztetést.

Basic mesh - A Basic mesh egy 3D objektum alakját leíró csúcsok, élek és lapok gyűjteménye: A csúcs egyetlen pont. (A vertex többes száma a "csúcsok") Az él két csúcsot összekötő egyenes szakasz. Az arc lapos felület, amelyet élek zárnak be.

Textúra - A textúrákat az anyagok tetejére helyezik, hogy bonyolult színeket és egyéb hatásokat kapjanak. Egy tárgyat olyan anyag borít be, amely többféle textúrát is tartalmazhat:

egy kőből készült képtextúrát, egy olyan textúrát, amely a követ göröngyösnek tűnik, és egy textúrát, amely a kő különböző módon deformálódását eredményezi.

Principled BSDF textúrák - A Principled BSDF, amely több réteget egyesít egyetlen könnyen használható csomópontban. A Disney-elvi modellen, más néven „PBR” shader-en alapul, így kompatibilis más szoftverekkel.

FPS - A képkockasebesség az egymást követő képek rögzítésének vagy megjelenítésének gyakorisága. A kifejezés egyaránt vonatkozik film- és videokamerákra, számítógépes grafikára és mozgásrögzítő rendszerre. A képkockasebesség nevezhető képkockafrekvenciának is, és hertzben fejezhető ki. Frame Per Second.

Addon - Szoftverbővítmény vagy -bővítmény bármely harmadik féltől származó szoftverprogram vagy szkript, amelyet a programhoz adnak hozzá, hogy további funkciókat és képességeket biztosítson. Például az Adobe Flash, amely lehetővé tette a felhasználók számára, hogy videókat nézzenek vagy játszhassanak egy internetböngészőn belül, kiváló példa a szoftverbővítményekre. A bővítményeket gyakran olyan személyek vagy szervezetek hozzák létre, akik nem állnak kapcsolatban az eredeti fejlesztővel.

VR - A virtuális valóság egy szimulált élmény, amely pózkövetést és 3D-s szemközeli kijelzőket alkalmaz, hogy a felhasználó magával ragadó érzést nyújtson a virtuális világban. A virtuális valóság alkalmazásai közé tartozik a szórakoztatás, az oktatás és az üzleti élet. (Virtual Reality).

XR - A kiterjesztett valóság a kibővített valóságra és a virtuális valóságra utal. Néha az „XR” mozaikszót használják a helyén. A virtuális valóság és a kibővített valóság területe gyorsan növekszik, és sokféle módon alkalmazzák, például a szórakoztatás, a marketing, az ingatlanügy, a képzés és a távmunka területén.

steamVR - A SteamVR egy fejlett eszköz a VR-tartalom megéléséhez a kívánt hardveren. A SteamVR támogatja a Valve Indexet, a HTC Vive-ot, az Oculus Riftet, a Windows Mixed Reality fejhallgatókat és másokat.

Arduino Uno - Az Arduino Uno a Microchip ATmega328P mikrokontrolleren alapuló nyílt forráskódú mikrovezérlő kártya, amelyet az Arduino.cc fejlesztett ki, és eredetileg 2010-ben

adták ki. A kártya digitális és analóg bemeneti/kimeneti érintkezők készletével van felszerelve, amelyek különféle bővítőkártyákhoz csatlakoztathatóak.

Renderelés - Fotórealisztikus, háromdimenziós megjelenítés, árnyékkal és tükrözéssel.

13. Irodalomjegyzék

- [1] Lukovics Miklós–Udvari Beáta–Zuti Bence–Kézy Béla Az önvezető autók és a felelősségteljes innováció Közgazdasági szemle, LXV. évf., 2018. szeptember (949–974. o.) Available: http://real.mtak.hu/83397/1/04_LukovicsUdvariZutiKezyA_u.pdf
- [2] Transportation Research Part F: Traffic Psychology and Behaviour Volume 67, November 2019, (175.o.-194.o.) Available:
https://www.sciencedirect.com/science/article/pii/S1369847819302293?casa_token=8BQ-DV57RGwAAAAA:OVj3iv70IEF-JWpJlOhbWvwblVcTHJRI2RXDa5xaBdMZjZLKxXzKcrTLdCYmmMqBsZpA2IwSF0w
- [3] Shatanawi, M.; Mészáros, F. Implications of the Emergence of Autonomous Vehicles and Shared Autonomous Vehicles Doi: su141710952 (1.o-10.o) Available:
<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwizrcTZmrj8AhU0RuUKHZAUBFEQFnoECBEQAQ&url=https%3A%2F%2Fwww.mdpi.com%2F2071-1050%2F14%2F17%2F10952%2Fpdf&usg=AOvVaw3cmu9KkKejhgPOtysUJbEl>
- [4] Linking self-driving cars to traffic signals might help pedestrians give them the green light Published: April 21, 2020 2.48pm CEST Available:
<https://theconversation.com/linking-self-driving-cars-to-traffic-signals-might-help-pedestrians-give-them-the-green-light-132952>
- [5] Schuette, S, 2019. Pedestrian Safety, Pebble (10.o-20.o). Available:
https://books.google.hu/books/about/Pedestrian_Safety.html?id=dGqLDwAAQBAJ&redir_esc=y
- [6] Aston University, 2015 How Autonomous Vehicles Will Profoundly Change the World (9.o39.o)Doi:RG.2.2.36272.10249
Available:https://www.researchgate.net/publication/338412717_How_Autonomous_Vehicles_Will_Profoundly_Change_The_World

- [7] Vass Zoltán Flaccus Kiadó, 2021 A rajzvizsgálat pszichodiagnosztikai alapjai - Projekció, kifejezés, mintázatok (417.o-450.o).
Available:https://books.google.hu/books?id=E_qwX3AU1NsC&printsec=copyright&hl=hu#v=onepage&q&f=false
- [8] Pavlov, I. (2012). Conditioned Reflexes. Courier Corporation ISBN:9780486430935 (219.o-320.o)
Available:https://books.google.hu/books/about/Conditioned_Reflexes.html?id=cknrYDqAClkC&redir_esc=y
- [9] Lionel P. Robert Jr. *Trust in AV: An Uncertainty Reduction Model of AV-Pedestrian Interactions* (1.o-2.o). Available:<https://deep-blue.lib.umich.edu/bitstream/handle/2027.42/140747/Trust-AV-Uncertainty.pdf?sequence=1>
- [10] Beginning Blender: Open Source 3D Modeling, Animation, and Game Design Lance Flavell ISBN: 978-1-4302-3127-1 (160.o.-370.o.) Available: Beginning Blender: Open Source 3D Modeling, Animation, and Game Design - Lance Flavell - Google Könyvek
- [11] Tony Mullen Mastering Blender Author Tony Mullen Edition 2, illustrated Publisher Wiley, 2012 ISBN: 9781118333396 (3.o.-37.o.) Available: Mastering Blender - Tony Mullen - Google Könyvek
- [12] Modeling and Animation Using Blender: Blender 2.80: The Rise of Eevee, Ezra Thess Mendoza Guevarra Publisher Apress, 2019 ISBN: 9781484253403 (167.o. – 270.o.) Available: Modeling and Animation Using Blender: Blender 2.80: The Rise of Eevee - Ezra Thess Mendoza Guevarra - Google Könyvek
- [13] HTC Corporation, 2021, Getting Started; System Requirements - HTC VIVE Pro 2 User Manual available:<https://www.manualslib.com/manual/2292718/Htc-Vive-Pro-2.html?page=5#manual>

- [14] Mark Geddes, 2016, Arduino Project Handbook 25 Practical Projects to Get You Started ISBN-13: 978-1-59327-690-4 (o.117-o.143) Available: <https://ss-valpovo.hr/wp-content/uploads/2020/01/arduino-project-handbook.pdf>
- [15] Arduino Programming in 24 Hours, Sams Teach Yourself (Sams Teach Yourself: In 24 Hours) 1st Edition by Richard Blum ISBN: 978-0672337123 (310.o.-400.o.) Available: Arduino Programming in 24 Hours, Sams Teach Yourself - Richard Blum - Google Books