

Raspberry Pi 3 alapú UGV drone robot tervezés & megvalósítása

Juhász Gergely Patrik
Informatika Kar
Debreceni Egyetem, Műszaki Kar
Debrecen, Magyarország
juhasz.gergely.94@gmail.com

Nagy István
Mechatronikai Tanszék
Debreceni Egyetem, Műszaki Kar
Debrecen, Magyarország
nistvan@eng.unideb.hu

Erdei Timotei István
Mechatronikai Tanszék
Debreceni Egyetem, Műszaki Kar
Debrecen, Magyarország
timoteierdei@eng.unideb.hu

Absztrakt—Az Ipar 4.0/IoT térhódításával megjelentek a vezeték nélküli (Wi-Fi) összeköttetésen alapuló technológiát kihasználó úgynevezett „okos” eszközök, amelyek távolról vezérelhetőek akár egy telefon segítségével. E projektben egy olyan UGV robot került megtervezésre és megépítésre, amelyet akár Android/iOS applikáción vagy PC-n keresztül lehet vezérelni, illetve a robot vision szenzor által látott kép streamelésre kerül.

A robot maga a Raspberry Pi 3 hardware-ére támaszkodik, ami rugalmassá teszi a feladatok elvégzésére, mivel kompakt, nagy teljesítményű és beépített Wi-fi adapter-el rendelkezik. Ezek mellett könnyen bővíthető olyan modulokkal, amelyek képesek szervo és DC motorokat vezérelni, ilyen is került alkalmazásra ebben a projektben. A robot az előbb említett vision szenzor segítségével képes adatot gyűjteni a külvilágból így felhasználó által való vezérlése a robotnak problémamentesen történik.

Kulcsszavak—*Raspberry Pi; IoT, Raspbian, Python, Andriod, Smart Car*

I. BEVEZETŐ

A Debreceni Egyetemen a fejlesztési és kutatási részleg, az Épületmechanikai Kutatóközpontban innovatív rendszerek fejlesztésével foglalkozik a jövő számára [11]. Ezen rendszereket az oktatásban is integrálásra kerülnek [12].

A projekt alapját a Raspberry Pi legújabb, azaz 3-as változata adja. A mára elismertté vált mini-számítógépet elsőnek 2012 Februárban adta ki a Raspberry Pi alapítvány az Egyesült Királyságban [1]. Eredeti célja a számítástechnika világának megismertetése és megkedveltetése volt a fiatal generációval. Évek alatt az eszköz fejlődése gyorsütemű maradt és, egyre nagyobb számítás kapacitású processzorok, nagyobb memória, és több multifunkcionális port került az eszközbe. A legnagyobb áttörést az előző modellekhez képest a beépített Wifi modul jelentette.

A jelenleg elérhető széria egyik hiányossága, hogy a 4 darab USB port a régebbi 2.0-ás verziót használja az újabb 3.0 vagy 3.1-es verzió helyett, talán a következő piacra kerülő „málna pc” már ezzel rendelkezni fog.

A hiányosságai ellenére az eszköz rendkívül sokoldalú, kiváló okos házakat felügyelni és vezérelni a benne lévő eszközöket.

II. TERVEZÉSI SZEMPONTOK

A robot megtervezésénél a legfontosabb szempontok a költséghatékonyság és a kis méret volt a könnyebb manőverezhetőség érdekében. Ezért is esett a választás a Raspberry Pi 3 mini számítógépre [2], ami nem csak költséghatékony megoldás, de könnyen adaptálható volt egy robot stabil-biztonságos működtetésére továbbá könnyen bővíthető további modulokkal, amelyekkel még sokoldalúbb felhasználhatóság érhető el.

További tervezési szempontok voltak, hogy a rendszert Linux alapú disztribúció képezze, illetve a forráskód Python programozási nyelven legyen megírva, mivel könnyen tanulható.

A robot moduláris felépítése miatt, a komponensek egyszerűen cserélhetők ezáltal képes két üzemmódban is működni. Az egyik a kamera mód, amelyben a robot elejére szerelt HD kamera biztosítja a képet a kezelőnek, a másik mód a radar mód, amelyben egy ultrahanggal működő távolságérzékelő modul segítségével folyamatosan pásztázza a környezetét és feltérképezi a közelében lévő tárgyakat.

III. ROBOT FELÉPÍTÉSE

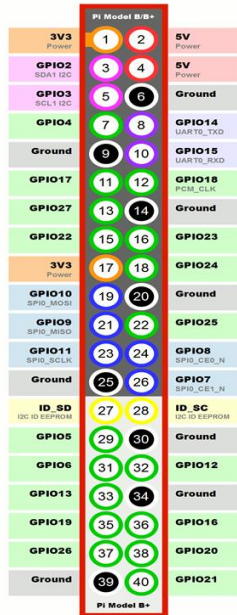
A robot komponensei polimetilmetakrilátból készült lapokra vannak felcsavarozva. Leghátul található a Raspberry Pi, ehhez csatlakozik az UGV (Unmanned Ground Vehicle) robot Shield-je.

Az Rpi-t, ami sokoldalúvá teszi az a General Purpose Input Output modul, azaz GPIO. Ez 40 darab általános felhasználású bementi és kimeneti pinto jelent amihez „anya” típusú jumper kábellel tudunk csatlakozni.

Az Rpi a Shieldhez a 2-es, a 3-as és 9-es számú GND az az földelés pineken keresztül van összekötve. Ez a csatlakozás úgy nevezett I2C buszon keresztüli csatlakozás [3]. Ezt a Raspberry Pi beállításai között külön engedélyezni kell.

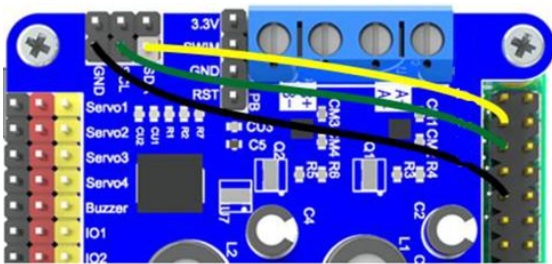
A csatlakozás után, hogy detektálhatni lehessen az I2C eszközöket szükséges egy i2c-tools csomag a Raspberry Pi-on, ami tartalmaz egy i2cdetect parancsot ez kilistázza a csatlakoztatott eszköz címét [4].

```
pi@raspberrypi:~$ i2cdetect -y 1
00:  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
10:  -- -- -- -- -- -- -- 18 -- -- -- -- --
20:  -- -- -- -- -- -- -- -- -- -- -- -- --
30:  -- -- -- -- -- -- -- -- -- -- -- -- --
40:  -- -- -- -- -- -- -- -- -- -- -- -- --
50:  -- -- -- -- -- -- -- -- -- -- -- -- --
60:  -- -- -- -- -- -- -- -- -- -- -- -- --
70:  -- -- -- -- -- -- -- -- -- -- -- -- --
```



1. ábra: Raspberry Pi 3 - I2cdetect & GPIO [5]

Az I2C buszon keresztül csatlakozáshoz szükséges 2 busz vonal, egy SDA (Serial Data Line) és egy SCL (Serial Clock Line). Amikor a kapcsolat elkezdődik mind az SDA mind az SCL „HIGH” állapotban van. A Serial Data vonalon elkezdődik az adatátvitel, a Serial Clock vonal egy referenciát biztosít az IC-nek, hogy minden egyes bit, ami a Data vonalon átmegy nem sérült és használható. Amikor a kapcsolat befejeződik egy speciális „STOP” bitet küld a az SDA, a Data vonalon, jelezve, hogy mostantól szabad a vonal és más is használhatja az adatkapcsolatot [3].



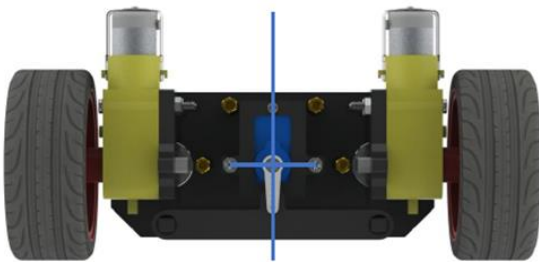
2. ábra: Smart Shield összekötése a Raspberry Pi 3-al

A Shield tartalmaz egy DC power jack csatlakozót, amibe a robot akkumulátorát jelentő 2 db 18650-es méretű elem

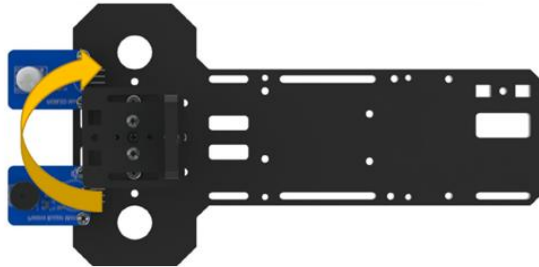
csatlakozik. Ez látja el megfelelő feszültséggel a Shield-t és a Raspberry Pi-t is. A Shield jobb alsó sarkában található 4 db töltöttséget jelző led, ami a két elem feszültség szintjét jelzi. A Shielden megtalálható Firmware verzió frissítéshez szükséges port is. Található rajta egy USB port, aminek segítségével köthető össze a Raspberry Pi micro USB csatlakozójával így tud a Raspberry áramellátást kapni. A bal alsó sarokban található a szervók vezérléséhez, a csengő működtetéséhez, a zöld, piros, kék színben világító LED felkapcsolásához szükséges portok illetve az ultrahanggal működő távolságérzékelő modul áramellátásához szükséges pinnek is. Balra fent található a két DC motor vezérléshez szükséges interfacek.

A robot működéséhez szükséges forgatásokat 3 darab szervó motor végzi [14]. Mindegy egyes szervó vezérléséhez 3 vonal szükséges egy végzi a vezérlést (Control Signal), a második az áramellátást (Power Supply), a harmadik a földelés (Ground) [6]. Az első számú szervó motor az elől lévő két darab vezető kerék forgatását vezérli, a függőleges Y tengelytől jobbra és balra is 90 fokkal. Így a robot kanyarodási szöge változtatható 10 és 60 fok között. A második szervó motor a robot elején található kamerát és az ultrahanggal működő távolságérzékelő modult, az X tengelyen képes forgatni 0 és 180 fok között. A harmadik szervó a kamerát az Y tengelyen képes forgatni 0 és 90 fok között. Ezáltal a kamera képes lefedni a robot előtt látható teljes területet. A robot meghajtását 2 darab DC motor végzi.

1. számú szervó és két darab DC motor



2. számú szervó forgási iránya



3. számú szervó forgási iránya

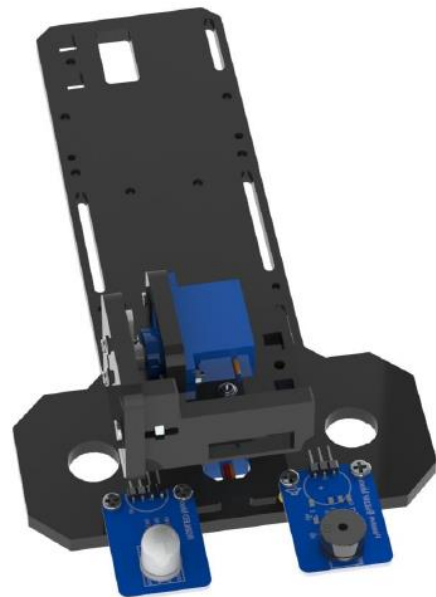


3. ábra: 1-2-3 szervó morok

IV. UGV ROBOT ÖSSZESZERELÉS

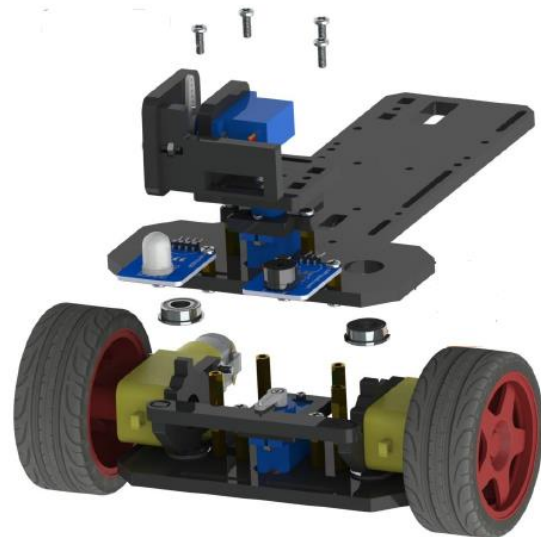
A főbb lépések SolidWorksben lettek megtervezve [7]. A robot összeszerelése a 2-es szervó felhelyezésével a fővázra kezdődött. Beszerelés előtt a szervók tesztelésre kerültek, hogy a megfelelő pozícióban legyen rá a felhelyezve a lengőkar [13].

Ezután került rá a kamerát illetve az ultrahangos távolságérzékelő eszközt tartó modul és az ezt mozgató 3. számú szervó.



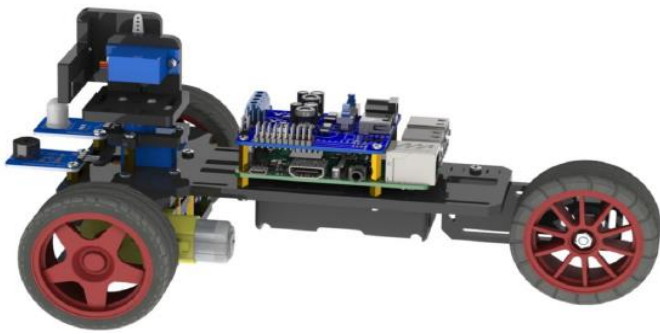
4. ábra: A kamera és távolságérzékelőt tartó modul és a 3. számú szervó

Ezt követően került összeszerelésre a panel két vezető kerékkel a DC motorokkal és azt vezérlő 1. számú szervóval, majd a panel rögzítése a fővázra.



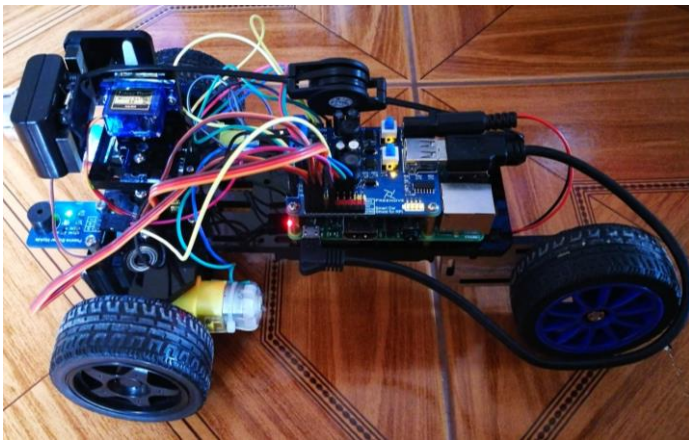
5. ábra: Panel rögzítése a fővázra

Az utolsó lépések közé tartozott a robot tápegységét jelentő két darab akkumulátort tartó modul felhelyezése a fővázra, illetve a Smart Car Shield és a Raspberry Pi felhelyezése



6. ábra: Az összeszerelt robot

Legutolsó lépésként az eszközök összekötése a Shielddel és a Raspberry Pi 3-al.



7. ábra: Az összeszerelt robot kábelezéssel együtt

V. TESZTELÉS SORÁN MEGHATÁROZOTT MŰSZAKI JELLEMZŐK

Mivel a robot megépítésekor nem volt fő szempont a nagy sebesség, inkább a jobb manőverezhetőség volt a cél, a következő sebesség adatok lettek meghatározva három méteres távon.

TABLE I. SEBESSÉG

Maximális sebesség százalékos kihasználása	Maximális sebesség [m/s]	Megtett út [m]	Idő [sec]
100 %	1,02	3	2,94
80 %	0,812	3	3,69
50 %	0,677	3	4,43

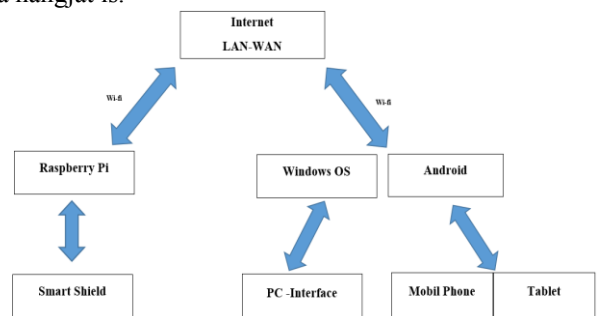
Összehasonlításképpen a sebesség mérés el lett végezve 10 méteres távon is. Az adatokból látszik, hogy a második teszten a robot nagyobb sebességet produkált, amiből következik, hogy a robotnak minimum három méter megtétele szükséges, hogy elérje a maximális sebességét.

TABLE II. SEBESSÉG

Maximális sebesség százalékos kihasználása	Maximális sebesség [m/s]	Megtett út [m]	Idő [sec]
100 %	1,17	10	8,55
80 %	1,098	10	9,1
50 %	0,92	10	10,8

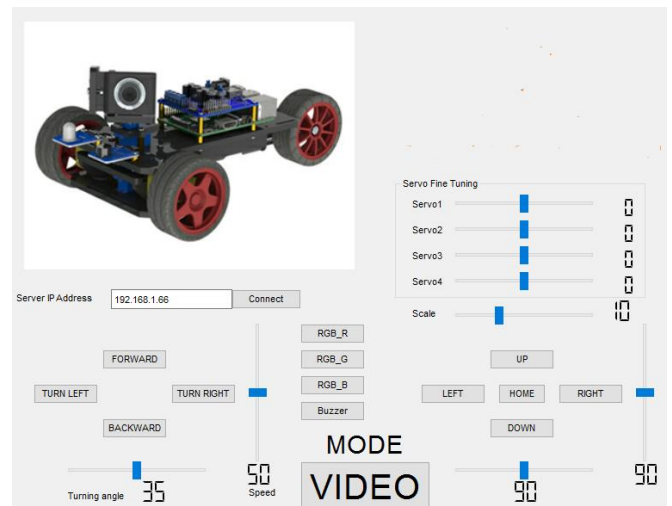
VI. PC & MOBIL INTERFACE CONTROL

Az irányítás történhet PC-ről egy interface-n keresztül vagy pedig Android operációs rendszert futtató okos telefonról vagy tablet-ről. PC-ről történő vezérlés esetén az interface több lehetőséget biztosít a robot irányítására, mint a mobil applikáción keresztüli vezérlés esetén. PC-ről módosítható a sebesség százalékos kihasználása, a fordulás szöge, a kamera helyzete (fokban megadva), állítható, hogy a skála milyen mértékben változzon, a szervók finomítását is be lehet állítani illetve a háromszínű led fényét lehet módosítani és a dudu hangját is.



8. ábra: A robot blokk diagramja

Az irányítás történhet PC-ről egy interface-n keresztül vagy pedig Android operációs rendszert futtató okos telefonról vagy tabletről.

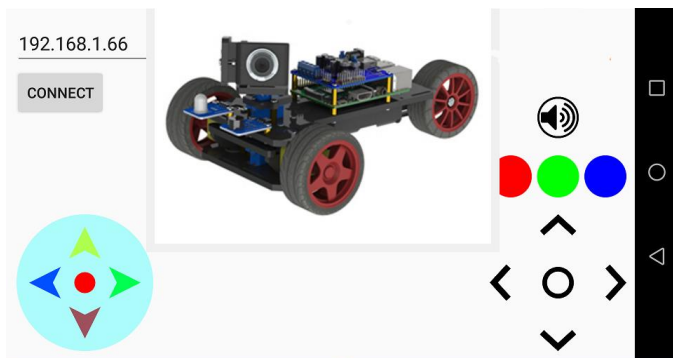


9. ábra: A PC-ről indítható interface

PC-ről történő vezérlés esetén az interface több lehetőséget biztosít a robot irányítására, mint a mobil applikáción keresztüli vezérlés esetén. PC-ről módosítható a sebesség százalékos kihasználása, a fordulás szöge, a kamera helyzete (fokban megadva), állítható, hogy a skála milyen mértékben

változzon, a szervók finomítását is be lehet állítani illetve a három színű led fényét lehet módosítani és a dudu hangját is.

Applikáción keresztül is állítható a kamera pozíciója, de nincsen megadva a fókusz, illetve a sebesség sem állítható csak irányítható egy adott sebességen. Van lehetőségünk beállítani a háromszínű led fényét illetve a dudu hangját.



10. ábra: Mobil applikáció interface

VII. FORRÁSKÓD

Az egyik fontos szolgáltatás, ami szükséges a robot működéséhez, az a PyQt4. Ez egy toolkit, aminek segítségével GUI (Graphical User Interface) alkalmazásokat lehet készíteni. A PyQt4 a Python programozási nyelv és Qt könyvtár egyvelege, ami az egyik leghasznosabb GUI könyvtár is egyben [8][9]. A szerver a Raspberry Pi-on fut és ezt biztosítja a kamera képének megjelenítéséhez szükséges mjpg-streamert. Ami egy nyílt forráskódú szolgáltatás, aminek segítségével lehet megjeleníteni a külön erre dedikált Raspberry Pi kamera modul vagy egy ezzel kompatibilis USB kamera által szolgáltatott képet [10]. Ha sikeresen feltelepült és elindult a szolgáltatás, akkor a Raspberry Pi IP címén a 8090-es porton, elérhető mjpg-stream demo oldala, ahol meg lehet nézni a kamera képét és lehet rajta finomítani, amennyiben szükséges. A kamera szerver egy Camera_Server.py Python forráskódban van elhelyezve. Ennek a programkódnak a lényege, hogy, amikor a megfelelő gomb megnyomásakor elindul a kamera szerver és megnyitja az mjpg-streamer indítását végző shellt, ha a kamera szerver leáll, akkor meghívódik mjpg-streamer leállítását végző shell [10]. A hálózati kapcsolódást TCPserver.py python szkript tartalmazza. Ez kreál egy portot a hálózati eszköz és az okos robot között, illetve továbbítja a Shield felé az eszköz felől érkező irányító parancsokat. Magukat a parancsokat a Command.py szkript definiálja. A kliens csatlakozását a szerverhez a ClientTCP.py szkript tartalmazza, amelyben szerepel a host IP címe illetve az a TCP port, amin keresztül az adattovábbítás történik.

VIII. ÖSSZEZÉS

A Raspberry Pi 3 alapú UGV robot megvalósításra került, amely Android applikáción vagy PC-n keresztül vezérelhető helyi hálózaton belül. Az akkumulátorok egyetlen töltéssel, 50% százalékos sebességkihasználtság mellett 1 óráig tudják ellátni a Raspberry Pi 3 és a Shieldet árammal. A projektben

felhasznált technológia mögött rengeteg lehetőség van felxibilitása és stabilitása révén, teljesen egészében önműködő egységként lehetne alkalmazni és terepen is alkalmazható lenne felderítési célból. Továbbá kezdetleges mesterséges intelligencia implementálásával is lehetne növelni a hatékonyságát.

KÖSZÖNETNYILVÁNÍTÁS

Szeretnék köszönetet mondani Erdei Timotei István tanár úrnak, aki nélkül ez a projekt nem jöhetett volna létre.

A publikáció elkészítését az EFOP-3.6.1-16-2016-00022 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

HIVATKOZÁSOK

- [1] Conor Lyons, (2017.05.09). A History of the Raspberry Pi [Online]. Available: <http://novadigitalmedia.com/history-raspberry-pi/>
- [2] Raspberry Pi 3 - Model B - ARMv8 with 1G RAM, (2017.05.15.). [Online]. Available: <https://www.adafruit.com/product/3055>
- [3] Jean-Marc Irazabal, Steve Blozis, (2017.05.10). I2C Manual [Online]. Available: http://www.nxp.com/documents/application_note/AN10216.pdf
- [4] Kevin Townsend, (2017.05.11.). Configuring Your Pi for I2C [Online]. Available: <https://learn.adafruit.com/adafruit-16-channel-servo-driver-with-raspberry-pi/configuring-your-pi-for-i2c>
- [5] Simple Guide to the RPi GPIO Header and Pins, (2017.05.12.). [Online]. Available: <http://www.raspberrypi-spy.co.uk/2012/06/simple-guide-to-the-rpi-gpio-header-and-pins/>
- [6] Byron J., (2017.05.13). Servo Trigger Hookup Guide [Online]. Available: <https://learn.sparkfun.com/tutorials/servo-trigger-hookup-guide>
- [7] SolidWorks, (2017.05.13.). [Online]. Available: <http://www.solidworks.com/>
- [8] David Boddie, (2017.05.11.). About PyQt [Online]. Available: <https://wiki.python.org/moin/PyQt4>
- [9] Introduction to PyQt4, (2017.05.11.). [Online]. Available: <http://zetcode.com/gui/pyqt4/introduction/>
- [10] Raspberry Pi Webcam Over the Internet Using MJPG-Streamer Walkthrough, (2017.05.11.). [Online]. Available: <https://www.zigsphere.com/howto/mjpgstreamer.html>
- [11] G. Husi, T. I. Erdei, Zs. Molnár, „A Novel Design of an Augmented Reality Based Navigation System & its Industrial Applications,” 15th IMEKO TC10 – Technical Diagnostics in Cyber-Physical Era Budapest, Hungary, 6 – 7 June, 2017 - Organised by: MTA SZTAKI – Hungarian Academy of Sciences - Institute for Computer Science and Control
- [12] Fedak V, Bauer P, Hájek V, Weiss H, Davat B, Manias S, Nagy I, Korondi P, Miksiewicz R, Duijsen P, Smékal P, " Interactive E-Learning in Electrical Engineering," Proceedings of the 15th International Conference on Electrical Drives and Power Electronics EDPE-2003. Konferencia helye, ideje: Podbanské, Szlovákia, 2003.09.24-2003.09.26. Košice: Technical University of Kosice, 2003. pp. 368-373.
- [13] T. I. Erdei, Zs. Molnár, N. C. Obinna, G. Husi, „Cyber physical systems in mechatronic research centre,” MATEC Web Conf. Volume 126, 2017.
- [14] T. I. Erdei, Zs. Molnár, G. Husi, „Robot visual and virtual control technology In industrial environment,” WoS (Web of Science) publication, International Symposium on Small-Scale Intelligent Manufacturing Systems (SIMS), Narvik, NORWAY- IEEE, Jun 21-24, 2016.