



**MISKOLCI EGYETEM**  
**GÉPÉSZMÉRNÖKI ÉS INFORMATIKAI KAR**



## **TUDOMÁNYOS DIÁKKÖRI DOLGOZAT**

# **A kiterjesztett valóság folyamatipari alkalmazása**

**Bartók Roland**

Villamosmérnök hallgató

Konzulens:

**Trohák Attila**

egyetemi adjunktus

Automatizálási és Infokommunikációs Intézeti Tanszék

**Miskolc, 2014**

# Tartalomjegyzék

|          |                                                                |               |
|----------|----------------------------------------------------------------|---------------|
| <b>1</b> | <b>BEVEZETÉS .....</b>                                         | <b>- 1 -</b>  |
| <b>2</b> | <b>MI A KITERJESZTETT VALÓSÁG?.....</b>                        | <b>- 2 -</b>  |
| 2.1      | A MOBIL KITERJESZTETT VALÓSÁG RÖVID TÖRTÉNETE [1].....         | - 2 -         |
| 2.2      | INFORMÁCIÓSZERZÉS A KÖRNYEZETRŐL.....                          | - 4 -         |
| 2.2.1    | <i>Jelzések használata.....</i>                                | - 4 -         |
| 2.2.2    | <i>A környezet sajátosságainak felhasználása .....</i>         | - 4 -         |
| 2.2.3    | <i>A készülék érzékelőinek használata .....</i>                | - 5 -         |
| 2.3      | NÉHÁNY NÉPSZERŰ KITERJESZTETT VALÓSÁG KERETRENDSZER .....      | - 5 -         |
| 2.3.1    | <i>ARToolkit [5] [10] .....</i>                                | - 5 -         |
| 2.3.2    | <i>Vuforia [11] [12] .....</i>                                 | - 6 -         |
| 2.3.3    | <i>Metaio [13].....</i>                                        | - 7 -         |
| 2.4      | SIEMENS ÁLTAL TERVEZETT IPARI ALKALMAZÁS (2002) [14] .....     | - 8 -         |
| 2.5      | A BECKHOFF TANULMÁNYA GOOGLE GLASS-AL .....                    | - 9 -         |
| <b>3</b> | <b>FEJLESZTÉSI TERV, KÖVETELMÉNYEK .....</b>                   | <b>- 10 -</b> |
| 3.1      | A FEJLESZTÉS CÉLJA .....                                       | - 10 -        |
| 3.2      | MEGOLDANDÓ FELADATOK .....                                     | - 10 -        |
| 3.2.1    | <i>A platform kiválasztása .....</i>                           | - 10 -        |
| 3.2.2    | <i>Helymeghatározás, helyzet-meghatározás, azonosítás.....</i> | - 10 -        |
| 3.2.3    | <i>A mérőpontok adatainak valós idejű lekérdezése .....</i>    | - 11 -        |
| 3.2.4    | <i>Kommunikáció .....</i>                                      | - 11 -        |
| 3.2.5    | <i>Egyéb felmerült feladatok .....</i>                         | - 12 -        |
| <b>4</b> | <b>A RENDSZER VÁZLATA.....</b>                                 | <b>- 13 -</b> |
| 4.1      | A MOBIL KLIENS MŰKÖDÉSI VÁZLATA.....                           | - 14 -        |
| 4.2      | A KÖZPONTI KISZOLGÁLÓ MŰKÖDÉSI VÁZLATA, FUNKCIÓI.....          | - 15 -        |
| <b>5</b> | <b>A KLIENS PROGRAM .....</b>                                  | <b>- 16 -</b> |
| 5.1      | OPERÁCIÓS RENDSZER ÉS HARDVER.....                             | - 16 -        |
| 5.2      | A KÖRNYEZET AZONOSÍTÁSA .....                                  | - 17 -        |

|           |                                                              |               |
|-----------|--------------------------------------------------------------|---------------|
| 5.3       | A QR-KÓD [21] .....                                          | - 17 -        |
| 5.3.1     | A QR-kód rövid története .....                               | - 17 -        |
| 5.3.2     | A QR-kód felépítése.....                                     | - 18 -        |
| 5.3.3     | A QR-kód fajtái .....                                        | - 20 -        |
| 5.3.4     | A QR-kód felhasználása [22] .....                            | - 21 -        |
| 5.4       | QR-KÓD FELHASZNÁLÁSA A KITERJESZTETT VALÓSÁG RENDSZERHEZ ... | - 21 -        |
| 5.5       | A QR-KÓD AZONOSÍTÁSA, DEKÓDOLÁSA.....                        | - 22 -        |
| 5.6       | KOMMUNIKÁCIÓ A KISZOLGÁLÓVAL .....                           | - 22 -        |
| 5.6.1     | WebSocket .....                                              | - 23 -        |
| 5.6.2     | Üzenetek a kiszolgáló és a kliens között.....                | - 24 -        |
| 5.7       | FELHASZNÁLÓI FELÜLET ÉS MŰKÖDÉS .....                        | - 24 -        |
| 5.7.1     | Információ a jelzések fölött .....                           | - 25 -        |
| 5.7.2     | A bővebb információs buborék .....                           | - 27 -        |
| 5.7.3     | Beállítások panel .....                                      | - 27 -        |
| 5.7.4     | Az alkalmazás leállítása.....                                | - 28 -        |
| <b>6</b>  | <b>A KÖZPONTI KISZOLGÁLÓ .....</b>                           | <b>- 29 -</b> |
| 6.1       | KAPCSOLÓDÁS AZ IPARI RENDSZERHEZ .....                       | - 29 -        |
| 6.1.1     | Az OPC kapcsolat [27].....                                   | - 29 -        |
| 6.1.2     | OPC kapcsolat fajtái [27] .....                              | - 29 -        |
| 6.1.3     | UTGARD - OPC DA kliens [28].....                             | - 30 -        |
| 6.2       | QR-KÓDOK GENERÁLÁSA ÉS ÖSSZERENDELÉSE AZ ÉRZÉKELŐKKEL .....  | - 30 -        |
| 6.3       | KAPCSOLATTARTÁS A KLIENSEKKEL .....                          | - 31 -        |
| 6.4       | A KÖZPONTI KISZOLGÁLÓ ELŐNYE .....                           | - 32 -        |
| <b>7</b>  | <b>ELVÁRÁSOK A KIHELYEZETT JELZÉSEKKEL SZEMBEN .....</b>     | <b>- 33 -</b> |
| <b>8</b>  | <b>FEJLESZTÉSI TERVEK .....</b>                              | <b>- 34 -</b> |
| <b>9</b>  | <b>ÖSSZEGZÉS.....</b>                                        | <b>- 35 -</b> |
| <b>10</b> | <b>FORRÁSOK .....</b>                                        | <b>- 36 -</b> |

# 1 Bevezetés

---

Az embereknek általános igénye, hogy információt szerezzenek a környezetükről. A megszokott érzékelés azonban nem képes azonnal megmutatni mindent, amire kíváncsiak vagyunk. Például, ha csak ránézünk egy forgó motorra, nem tudjuk, hogy pontosan milyen fordulatszámmal forog, mekkora áramot vesz fel. A saját érzékelői mérik, de előfordulhat, hogy a kijelzés távolabb van, így nem láthatjuk.

Az okostelefonok, táblaszámítógépek, viselhető eszközök segítségével azonban egyre inkább elérhetővé válik egy régóta ismert és igényelt, friss technológia, amellyel a szemlélt környezetről olyan információkat is szerezhetünk, amely eredetileg nem látható, esetleg nem hallható. A technológia neve kiterjesztett valóság (angolul augmented reality). Körülbelül az 1990-es évek közepétől kezd lassan terjedni, ahogy egyre nagyobb számítási teljesítmény áll rendelkezésre. Napjainkban az egyre növekvő számítási teljesítménnyel bíró hordozható eszközöknek köszönhetően egyre szélesebb körben tud elterjedni. Kezdetben a mérnökök játéka volt, később a tervezők munkáját segítette, majd lecsapott rá a reklámpiar, használják navigációs rendszerekben, megjelentek a karbantartási, javítási munkálatokhoz segítséget nyújtó programok főleg bonyolult gépek esetén. Ipari alkalmazásában az utóbbi években tűnt fel, a technológia monitorozására használják.

Ez a dolgozat is egy monitorozó, megfigyelő rendszert mutat be, amely egy táblaszámítógép képernyőjén keresztül a kamera által látott képre vetíti a megfigyelt technológia működési paramétereit akár a tényleges mérési ponttól távolabbi helyeken is, ahol az értékek még helyesek lehetnek (például a nyomás vagy áramlási sebesség egy csővezeték szakasz teljes hosszában). Az iparban is egyre elterjedtebbek az érintőképernyős eszközök, felépítésük robosztus az ipari körülményeknek megfelelően. Ezeket futtatható a bemutatásra kerülő alkalmazás, hogy a karbantartók egy egyszerű pásztázással információt szerezhessenek a megfigyelt technológia működéséről. A rendszer egy központi kiszolgálóból, kliens alkalmazásból és a technológián elhelyezett helyőrzőkből épül fel, amelyeket a kliens felismer.

## 2 Mi a kiterjesztett valóság?

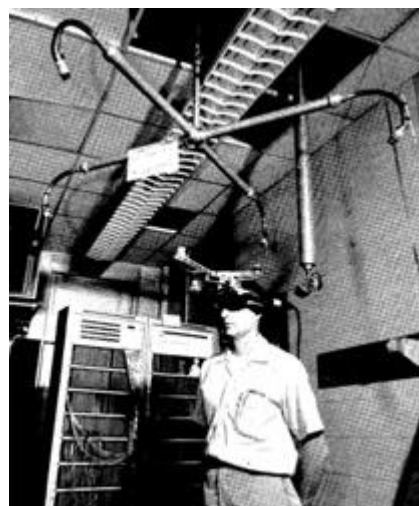
A kiterjesztett valóság és a virtuális valóság sok tekintetben hasonlít egymásra mégsem szabad összekeverni a két technológiát, bár több közös vonásuk van, mégsem ugyan arról van szó. Míg a virtuális valóság egy elképzelt, virtuális világot vetít a szemlélő elé, teljesen elfedve a valóságos képet, addig a kiterjesztett valóság virtuális képpel, információkkal kiegészíti a valóságot. Nem cseréli fel teljesen, hanem kibővíti azt.

### 2.1 A mobil kiterjesztett valóság rövid története [1]

Legelőször 1901-ben L. Frank Baum író a The Master Key című regényében ír egy olyan elektronikus kijelzőről, amelyen keresztül a megfigyelt személy homlokán megjelenik a fő tulajdonságának megfelelő betű (G – good, E – evil stb.). Az író szerint ez a technológia egy évszázaddal megelőzi a korát. [2]

1968-ban készítette el Ivan Sutherland az első fejre tehető kijelzőt.

1974-ben Myron Krueger kivetítőkkel és kamerákkal kötötte össze a valóságot a virtuális világgal. Gyors algoritmusokkal ismerte fel a homogén vászon előtt mozgó kesztyűt és más alakokat. [3]



1. ábra Ivan Sutherland fejre helyezhető kijelzője [1]

1994-ben Paul Milgram és Fumio Kishino definiálta a Valóság-Virtualitás kontinuumot, amely magába foglalja a valós környezetet és a virtuális környezetet. A kiterjesztett valóság szerintük közelebb áll a valósághoz, mint a virtualitáshoz.

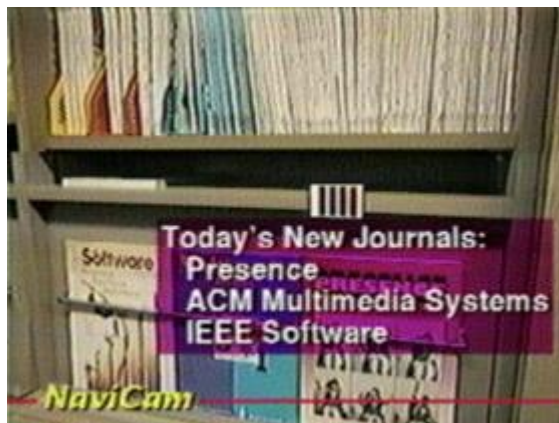


2. ábra Valóság-Virtualitás Kontinuum [1]

1995-ben Jun Rekimoto és Katashi Nagao elkészítette a NaviCam-ot, ami színes színkódolt jelzéseket azonosítva jelenítette meg a kapcsolódó információkat.

Hordozható kijelzőt és kamerát alkalmazott. A kamera képét egy közeli erős asztali munkaállomás dolgozta fel.

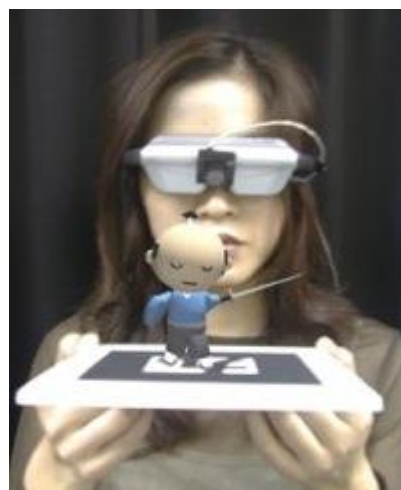
1996-ban Jun Rekimoto bemutatta a 2 dimenziós fekete-fehér mátrix kódokat, négyzet alakú vonalkódokat, amelyeket a kamera 6 szabadsági fokban tud követni. [4]



**3. ábra NaviCAM [1]**

1999-ben Hirokazu Kato és Mark Billinghurst bemutatta az ARToolkit-et, amely mára nagyon népszerű nyílt forrással is elérhető kiterjesztett valóság könyvtár. [5]

2000-ben Bruce Thomas és csapata bemutatta az AR-Quake-et, amely a népszerű PC játék, a Quake kiterjesztett változata. A kültéren és beltérben is játszható első személy nézetes játékhöz felhasználtak GPS-t, digitális iránytűt és vizuális jelzéseket. A játékosok egy-egy számítógépet cipeltek a hátukon, irányításhoz egyszerű kétgombos beviteli eszközt használtak. A kiterjesztett valóságot fejen hordott kijelzőn keresztül látták. A hagyományos billentyűzet és egér utasításokat a játékosok valóságos mozgása adta.



**4. ábra ARToolkit [1]**

2002-ben Michael Kalkusch és csapata bemutatott egy olyan rendszert, amely egy ismeretlen épületben segíti a tájékozódást ARToolkit alapon 2 dimenziós fekete-fehér jelzések segítségével.

2006-ban Reitmayr és csapata egy modell alapú hibrid objektumkövető rendszert mutatott be kültéri kiterjesztett valóság alkalmazásokhoz. Már képes valós időben a virtuális képet a valós képre vetíteni egy kézi számítógépen. [6]

2007-ben Kelin és Murray bemutatott egy rendszert, amely képes valós időben objektumkövetésre és térképezésre egy egyszerű kamerával kis munkaterületen.

2008-ban Wagner és csapata mutatta be az első valós idejű, mobil eszközön futó, 6 szabadsági fokú objektumkövető rendszert, amely természetes jellemzőket ismer fel a képen.

A METAIO pedig az első kereskedelmi mobil kiterjesztett valóság alapú múzeumi útmutatót mutatott be. Szintén a környezet természetes jellemzőit használták fel a virtuális kép helyének meghatározásához.

2009-től egyre több kiterjesztett valóság alapú alkalmazás jelent meg: Layar [7], amely a városban segíti a felhasználót például egy éttermet megtalálni. Google Sky Map [8], amely az égbolt csillagait rajzolja ki segítve az egyes csillagképek, bolygók azonosítását.

A reklámpia a vizuális jelzés alapján működő kiterjesztett valóság alkalmazásokat használja. Ilyen az Ikea mobil alkalmazása [9], amely a katalógus megfelelő ábráját felismerve a szobába rajzolja a kiválasztott bútort a kívánt színben, valós méretben.

## **2.2 Információszerzés a környezetről**

Ahhoz, hogy a megfelelő objektum kerüljön a kamerakép megfelelő helyére, meg kell határozni, hogy mit lát éppen a felhasználó a kijelzőn keresztül (ebben az esetben a kijelzőn keresztül vagy egy kamera valós képen keresztül látja a világot vagy egy átlátszó kijelzőn keresztül, ekkor nincs látható élő kamerakép).

### **2.2.1 Jelzések használata**

Az egyik legegyszerűbb mód, ha a fontos objektumokat, helyeket valamilyen kameraképen könnyen felismerhető és azonosítható címke jelöli. Ilyen címke lehet a már említett 2 dimenziós vonalkód vagy a napjainkban elterjedt változata, a QR-kód. A vonalkód kameraképen nehezen ismerhető fel, a sűrű függőleges vonalakat egy telefon kaméja általában összemosódva érzékeli, ezért előnyösebb a négyzet alapú jelzés. A négyzet torzulása segítséget nyújt ahhoz is, hogyan kell transzformálni a kirajzolt virtuális objektumot.

### **2.2.2 A környezet sajátosságainak felhasználása**

Az eljárás angol neve: natural feature detection/tracking. A környezetről készít egy absztrakciót, amely pontokból, vonalakból, esetleg foltokból vagy ezek

keverékéből áll. Attól függően választja ki, hogy a látott kép melyik absztrakciónak felel meg, hogy mennyire illeszkedik a tárolt absztrakt képre a látott absztrakt kép. Ezzel a módszerrel szinte bármi használható jelzésnek a kiterjesztett valóság alkalmazáshoz.

### **2.2.3 A készülék érzékelőinek használata**

Egy mai átlagos készüléket felszerelnek sok esetben GPS vevővel, elektronikus iránytűvel, gyorsulásmérővel, elfordulás mérővel. Ezek adataiból meghatározható, hogy a készülék éppen hol van, mit láthat a kamerája. Használják az egyes érzékelőket egyenként is, gyakori kombináció a GPS és az elektronikus iránytű párosa.

A felsorolt módszerek külön-külön is használhatóak, de általában valamilyen kombinációban használják őket. Az általunk megcélzott környezet tartalmaz fedett helyszíneket és olyan fémekkel körülhatárolt teret, ahol a GPS használhatatlan, az iránytű megbízhatósága is kérdéses. Ezért inkább a vizuális jelzéseken alapuló megvalósítást választottam.

A jelzés vagy helyőrző azonosítása, helyének meghatározása a kameraképen már megadja azt is, hová kell rajzolni az információt. Megfelelő 2 dimenziós kód akár plusz információt is tárolhat, így ha több helyen szükséges ugyan azt az információt megjeleníteni csak egy újabb jelzést kell elhelyezni a mobil alkalmazás számára.

## **2.3 Néhány népszerű kiterjesztett valóság keretrendszer**

### **2.3.1 ARToolkit [5] [10]**

1999-ben Hirokazu Kato és Mark Billinghurst által kifejlesztett mára az egyik legelterjedtebb kiterjesztett valóság könyvtár. Valós idejű kamerakép feldolgozást támogat, amelyre képes 3 dimenziós objektumokat rajzolni a felismert jelzésre és követi a nézőpont változását. Létezik jelzés alapú és a környezet sajátosságait felhasználó megvalósítása is. Jelzésnek fekete négyzet fehér belsejében elhelyezett képet, alakzatot használják. Elérhető Windows, iOS, Android platformokra. Támogatja a Flash-t, Unity-t és a webes alkalmazásokat. A felhasznált gyors

algoritmusoknak köszönhetően képes közel a kamera képfrissítési sebességével működni, viszonylag egyszerű egy új program implementációja a segítségével.

Mobil platformokból az iOS-t és az Android-ot támogatja. Létezik egy AndAR nevű Java alapú könyvtára, amely szintén használható Android platform is. Az Android rendszerekből a 2.2-től frissebbeken fut. A jelzések követésén túl lehetővé teszi a környezet jellemzőin alapuló nézőpontkövetést.

Elérhető GPLv2 licenc alatt, így ingyenesen használható, de az elkészült programnak is nyílt forrásúnak kell lenni és bárki számára ingyenesen elérhetőnek.

Létezik kereskedelmi licenc, ezzel többszolgáltatások is elérhetővé válnak pénzdíj ellenében. Az elkészült program zárt forrású, kereskedelmi forgalomba hozható lesz.



5. ábra ARToolkit működés közben [5]

### 2.3.2 Vuforia [11] [12]

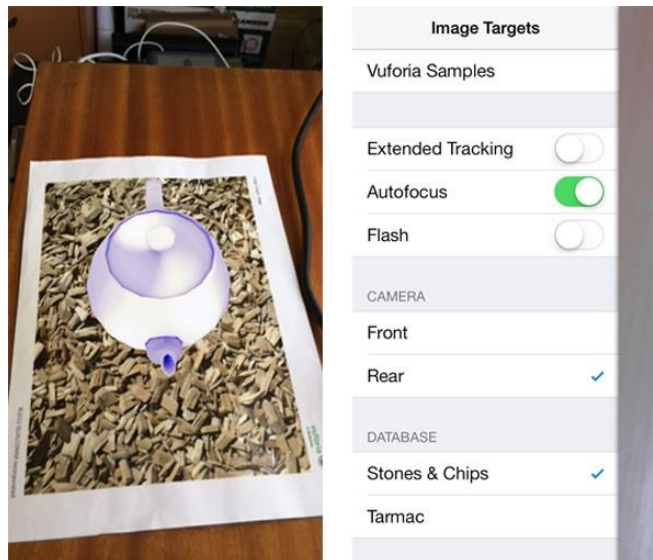
A Vuforia a Qualcomm Connected Experiences által kiterjesztett valóság alkalmazások létrehozására fejlesztett platform. Elérhető Android és iOS rendszerekre. Csak képfeldolgozáson alapul, hasonlóan az ARToolkit-hez.

A fejlesztőkörnyezet által nyújtott lehetőségek:

- Szó felismerés – körülbelül 100 ezer angol szót tartalmazó adatbázist használ vagy létrehozható egy saját szótár.
- Képek felismerése, követése a kameraképen. Saját kép használatára is lehetőséget nyújt.
- Forgástestek felismerése és követése, például üvegek, csészék, korsók.
- Egyszerű térbeli alakzatok, például dobozok felismerése.
- Keretes jelzések használata. A kereten belül bármilyen kép elhelyezhető, a keretek 512 variációt biztosítanak.
- A jelzéseken virtuális gombok kezelését támogatja, videó lejátszást.

A felsoroltakon túl a fejlesztő vállalat felhő-szolgáltatást is nyújt a fejlesztett alkalmazások adatainak tárolására illetve készülék adatbázist.

Használható ingyenes és kereskedelmi licenc alatt.



**6. ábra Egy Vuforia SDK-val készült mintaprogram [12]**

### **2.3.3 Metaio [13]**

A Metaio GmbH. a világ legnagyobb kiterjesztett valóság szoftvertechnológiát fejlesztő vállalata. 2003-ban alapították és 2006 óta tagja az insideAR kiterjesztett valóság konferenciának.

Támogatják a Windows, iOS, Android operációs rendszereket, asztali és mobil hardverre, webes alkalmazásokat. A metaio nem csak képfeldolgozást használ, feldolgozza a futtató készülék érzékelőinek és GPS-ének az adatait is, ha elérhető. Támogatják a viselhető eszközöket is, mint például a Google Glass, Vuzix szemüvegek.

Létezik iparban használható kiterjesztett valóság alkalmazásuk is. Ipari megoldásaik többek között CAD objektumokat jelenítenek meg, új külsőt varázsolnak egy autóra, képesek vezérelni a KUKA robotot, interaktív javítási segédletet biztosítanak egyes autófajtákhoz a szerelőknek és a közembereknek.



**7. ábra Gépkocsi szerelési segédlet [13]**

Létezik olyan ipari alkalmazásuk is, amellyel a megfigyelt környezet monitorozható, a megfigyelt terület azonosítása a környezet jellemzői alapján történik.

Termékek:

- metaio SDK: kiterjesztett valóság alkalmazások készítésére használható fejlesztői csomag.
- metaio Creator: kiterjesztett valóság alkalmazás, amellyel a felhasználó készíthet teljes jeleneteket a kiterjesztett valóság alkalmazáshoz programozói tudás nélkül.
- junaio: Kiterjesztett valóság alapú böngésző.
- metaio CVS (Continuous Visual Search): képfelismerés.
- metaio Cloud: a metaio felhasználói számára online tárhely a tartalmak számára.
- metaio Engineer: ipari használatra szánt alkalmazás, CAD állományok valós méretű megjelenítésére.

A metaio SDK elérhető ingyenes és pénzdíjas változatban is. Az ingyenes változat megjelenít egy vízjelet az elkészített programban.

## **2.4 Siemens által tervezett ipari alkalmazás (2002) [14]**

Elsősorban meghibásodott eszközök felderítésére tervezte a Siemens a beszédet támogatott kiterjesztett valóság (Speech Enabled Augmented Reality – SEAR) nevű elképzelését.

A rendszer az üzemben elhelyezett infra-adók segítségével határozza meg a kezelő helyét és irányát, a pontosításhoz optikai jelzéseket is használnak. Segítségével a karbantartó mérnök hamarabb megtalálja a meghibásodott alkatrészt, mert a rendszer a pillanatnyi pozícióját felhasználva navigálja a megfelelő helyre. Szóbeli utasításokra az IP címmel rendelkező egységek válaszolnak



**8. ábra Siemens - Beszélő csövek [14]**

a PDA-n keresztül. A 2002-es cikkben az kereskedelmi forgalomba helyezését 5-7 évre becsülték.

## 2.5 A Beckhoff tanulmánya Google Glass-al

A Beckhoff a 2013. novemberi nürnbergi SPS IPC Drives automatizálási szakvásáron mutatta be a gépek kezelésének új, Google Glass szemüvegre alapozott koncepciójának műszaki tanulmányát. [15]

A leírásuk szerint a szemüveg a kameráján keresztül felismeri a gépeken elhelyezett QR-kódokat, amely alapján azonosítja az adott gépet. Ezután megjelennek a szemüveg kijelzőjén a



9. ábra Képek a Beckhoff alkalmazásáról [16]

megfigyelt gép működési paraméterei, elérhetők az adatlapok. Kép készíthető az adott gépről, amelyet megoszthat más felhasználókkal. Ha szükséges kapcsolatba léphet más felhasználókkal vagy hanginformációkat érhet el. A szemüveg az adatokat a rendszerhez kapcsolódó szerverről kérdezi le a QR-kódban tárolt információ alapján. A szerver a TwinCAT-el vezérelt automatizálási szerverrel kommunikál. [16] [17] [18]

## **3 Fejlesztési terv, követelmények**

---

### **3.1 A fejlesztés célja**

A cél egy olyan rendszer létrehozása, amely segítségével az iparban dolgozó felhasználó egyszerű módon juthat valós idejű információhoz a megfigyelt technológiával kapcsolatban. Mindezt úgy, hogy egy kamerával felszerelt készülék azonosítja a megfigyelt területet, kérdezi le a szükséges adatokat olyan helyen is, ahol már az adatokat mérő érzékelők kijelzője nem látható, hozzáférhető, de az értékek még jellemzik a rendszert. Ezen kívül fontos a mobilitás.

Az alkalmazással szemben elvárás, hogy olyan hardveren is fusson, amely veszélyes ipari környezetben is használható. Illetve tudnia kell kommunikálni, adatkapcsolatot létesíteni az elterjedt számítógépes folyamatirányító rendszerekkel (OPC kapcsolat), hogy valós idejű adatokat kérdezzen le az azonosított technológiarészlettel kapcsolatban.

### **3.2 Megoldandó feladatok**

#### **3.2.1 A platform kiválasztása**

Az alkalmazás számára a jelenleg legelterjedtebb mobil operációs rendszert, az Android-ot választom. Népszerűsége miatt nagyon sokfajta hardveren elérhető, használják az iparban is. Léteznek rá például Bluetooth alapú adatmegjelenítő, adatgyűjtő alkalmazások különféle mérőeszközökhöz. [19]

#### **3.2.2 Helymeghatározás, helyzet-meghatározás, azonosítás**

Fontos, hogy az alkalmazás tudja, mit lát éppen a kamera. Beltéri ipari környezetben és olyan külső téren, ahol a megfigyelőpont sok fémes építménnyel van körülvéve a GPS megbízhatatlan vagy akár használhatatlanná is válhat. Ezért a hely meghatározására nem alkalmazható. A pontos meghatározáshoz vizuális jelzések felismerése jelenthet megoldást. 2 dimenziós mátrix kódokkal a mérőpontok azonosítása is megoldható. Felismerésük viszonylag gyors, a kinyert információ alapján egyértelműen kiderül mely jelzés került a kamera látóterébe. A mérőpont által tárolt azonosító egyúttal a lekérdezendő adatok azonosítója is lehet. A vonalkódok

kamerával nehezen azonosíthatóak, a vékony vonalak könnyen összemosódhatnak a kamera képén.

### **3.2.3 A mérőpontok adatainak valós idejű lekérdezése**

Az azonosítást követően az ipari folyamatirányító szervertől kell az adatokat lekérdezni. Mivel várhatóan sok mérőpontot kell kezelni a rendszerben így szükséges lehet egy központi kiszolgáló felállítása. Így nem szükséges vastag kliensalkalmazást létrehozni a terepen használt mobil eszközre. A központi kiszolgáló kérdezi le a folyamatirányító rendszerről a kívánt információkat, bármilyen változás esetén elegendő a közös adatbázison változtatni nem kell a kliensalkalmazást minden mobil eszközön frissíteni. Ez főként akkor kényelmes megoldás, ha karbantartás vagy új mérőeszközök beépítése miatt akár időszakosan is gyakran kellene a műveletet elvégezni.

Előnyös azért is egy központi kiszolgáló használata, mert nem biztos, hogy minden folyamatirányítási rendszernek létezik támogatása mobil operációs rendszerre.

### **3.2.4 Kommunikáció**

A mobil eszközök körében a legelterjedtebb kommunikációs lehetőség a WiFi és a Bluetooth kapcsolat. Utóbbi kisebb hatótávolsága és az egyidejűleg kezelhető kliensek alacsony száma miatt nem használható. A WiFi segítségével akár a meglévő hálózat is igénybe vehető, ha a többlet adatmennyiség nem zavarja a megszokott kommunikációt. Ellenkező esetben szükséges lehet egy újabb hozzáférési pont hálózat kiépítése figyelembe véve, hogy az újabb rádiós eszközök között ne lépjenek fel zavarások.

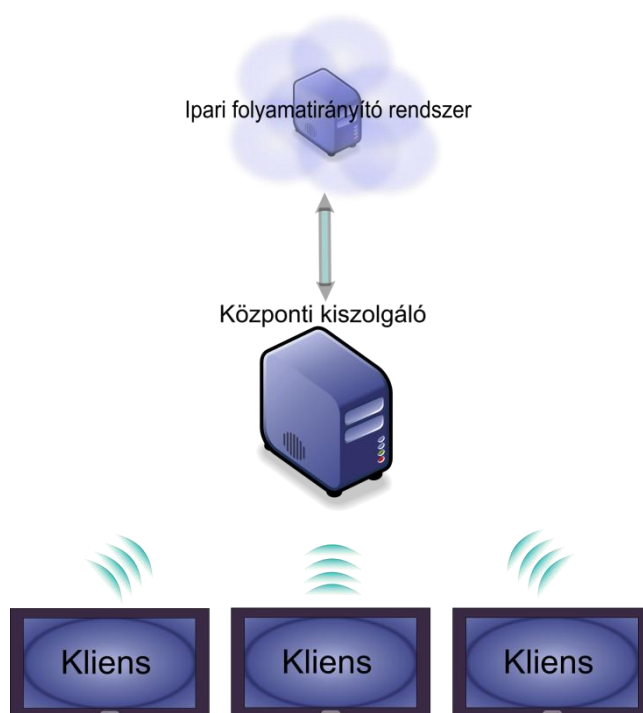
A központi kiszolgálónak egyszerre több kliens számára is biztosítani kell az adatokat, ezért olyan kapcsolati megoldást kell alkalmazni, amely támogatja, hogy egyszerre sok klienst kiszolgál. Tehát az egyszerű TCP/UDP csomagok küldésénél és fogadásánál érdemes valami intelligensebb megoldást választani.

### **3.2.5 Egyéb felmerült feladatok**

A használhatóság javítása érdekében a központi kiszolgálót ki kell egészíteni egy olyan lehetőséggel, hogy a kihelyezendő mátrixkódokat is képes legyen előállítani és legalább nyomtatható formátumba exportálni. Az ilyen módon generált kódokat automatikusan felveszi az adatbázisba így már csak össze kell rendelni a megfelelő mérőeszközök adataival.

## 4 A rendszer vázlata

---



**10. ábra A teljes rendszer vázlata**

A 10. ábra a teljes rendszer vázlatát mutatja. Az üzemben a mobil kliensek mozognak és kérdezik le a szükséges információkat a központi kiszolgálótól.

A központi kiszolgáló feladata a kapcsolat fenntartása az ipari rendszerrel, az adatok lekérdezése és a kért adatok kiküldése a kliensek számára.

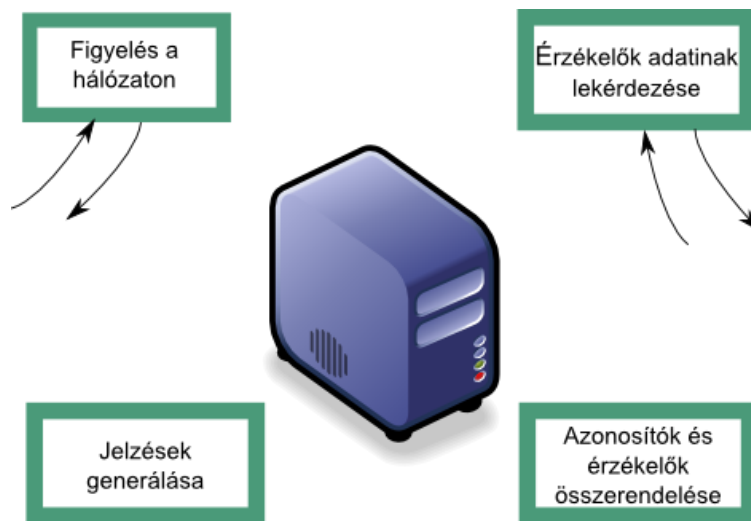
## 4.1 A mobil kliens működési vázlata



11. ábra A mobil kliens működési vázlata

A mobil kliens a kameraképen felismeri és azonosítja a telepített jelzéseket. Hálózati kapcsolaton keresztül lekérdezi a jelzéshez tartozó információkat és a kameraképen a jelzésnek megfelelő helyre rajzolja. Az adatok nem a 2. fejezet képein látható 3 dimenziós megjelenítési módon tűnnek fel a képernyőn, hanem 2 dimenziós síkban kerülnek kirajzolásra. Mivel csak adatok megjelenítéséről van szó, nem feltétlenül szükséges a térbeli transzformációjuk.

## 4.2 A központi kiszolgáló működési vázlata, funkciói



**12. ábra A központi kiszolgáló vázlata**

A központi kiszolgáló egy adatbázisban tárolja mely jelzésekhez mely adatokat kell lekérdezni az ipari szervertől. Tartja a kapcsolatot az aktív kliensekkel és az ipari szerverrel. Fogadja a kéréseket a kliensektől és a megfelelő adatok lekérdezése után válaszként küldi a kívánt információt.

Lehetőséget nyújt új jelzések generálására, hozzájuk adatok rendelésére és a meglévő jelzésekhez új érzékelők rendelésére.

## 5 A kliens program

A kliens program felhasználói főként az üzem területén, a technológia, gépek közelében karbantartási, felügyeleti munkát végző személyek. A kliens segítségével a kameraképen láthatják, hogy a megfigyelt mérőpontokon a technológia helyesen működik-e és a fontos paraméterek értékét valós időben követhetik.

### 5.1 Operációs rendszer és hardver

A kliens alkalmazás számára a jelenleg legelterjedtebb és általam is legjobban ismert mobil operációs rendszert, az Android-ot választottam.

Egyik követelmény, hogy a kifejlesztett rendszer veszélyes ipari környezetben is használható legyen. Ennek feltétele, hogy a kliensalkalmazást futtató hardver rendelkezzen a megfelelő ATEX minősítéssel. A tábla számítógépek széles



13. ábra Z720-EX [33]

választékában elérhető szinte minden zónához megfelelő készülék. Természetesen olyan területen, ahol nem fenyeget robbanás vagy egyéb veszély a megszokott, de tartós kivitelű készülékek is alkalmazhatóak. Célszerű az alkalmazást tesztelni a kiválasztott készülékkel, mert előfordulhat, hogy a kamera gyenge minőségű, ami miatt a jelzések felismerése lehetetlenné válik vagy olyan módosított Android változatot

telepített a gyártó a készülékére, ami miatt módosítani kell az alkalmazáson vagy más készüléket kell keresni.

A jelzések megfelelő elhelyezésével lehetséges, hogy nem kell például a legszigorúbb előírásoknak megfelelő készüléket használni, mert biztonságos távolságból is monitorozható a veszélyes üzemrész működése. Itt látható az egyik előnye a kiterjesztett valóság alapú monitorozó rendszernek, mert nem szükséges külön SCADA/HMI panelt a biztonságos területre telepíteni, hanem elegendő a jelzést felfesteni vagy kihelyezni.

Másik fontos követelmény, hogy a készüléken legyen egy megfelelő minőségű hátoldali kamera. A használhatóság érdekében fontos, a kamera jó felbontóképessége, tehát videó üzemmódban is részletes képet adjon, amelyen jól látszik a jelzés akár nagyobb távolságról is. Fontos, hogy gyenge fényviszonyok mellett se legyen a kép túl zajos, mert az lehetetlenné teheti a felismerést. Ennek a

követelménynek a közép-, és felsőkategóriás illetve ipari kivitelű készülékek többsége eleget tesz.

Szintén fontos a készülék kijelzője. Felhasználástól függően célszerű lehet olyan rezisztív érintőképernyővel szerelt készüléket választani, amely kesztyűben is kezelhető. A képernyő mérete lényeges, egy kisebb méretű képernyőn jóval kevesebb információ fér el és bizonyos esetekben a képernyő pixeleiben számított mérete meghatározza a kamera élőképeinek is a legnagyobb méretét pixeleiben. Tapasztalat szerint 800\*480pixel méretű élőképpel és képernyővel már kényelmesen használható az alkalmazás. A táblagépek általában legalább 7"-12" képátlóval készítik és egyre gyakoribb a HD-nek megfelelő felbontás (1920\*1050).

A napjainkban használt hardverek számítási teljesítménye elegendő az alkalmazás megfelelő használatához a közép-, és felsőkategóriában az ipari és ellenálló kivitelű készülékek teljesítménye szintén elegendő.

## **5.2 A környezet azonosítása**

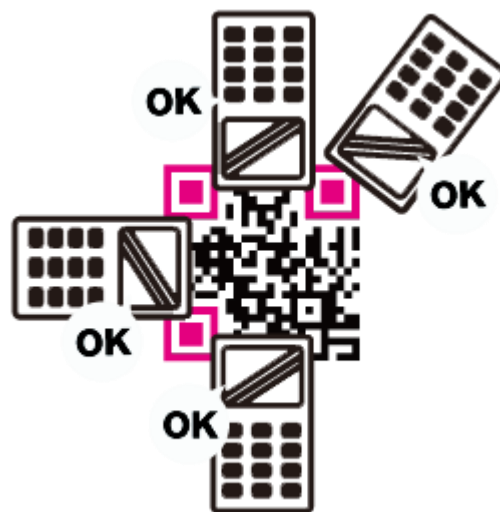
A 2. fejezetben bemutatott módszerek alapján a mérőpontok azonosítása 2 dimenziós fekete-fehér mátrix kódokkal történik. Ezek közül is a közismert QR-kóddal. Ennek előnye a vonalkóddal szemben, hogy kamerával is könnyen felismerhető és értelmezhető. Elterjedtebb, mint a többi hasonló megoldás, például az Aztec-kód [20]. Az egyszerű fekete keretes mátrixkódokkal szemben jóval több egyedi azonosítót lehet létrehozni. Jelen esetben nem elég azt felismerni, hogy hol van a jelzés a képen, hanem azonosítani is kell, hogy melyik jelzésről van szó. Előnye, hogy bármilyen szögben elforgatva dekódolható.

## **5.3 A QR-kód [21]**

### **5.3.1 A QR-kód rövid története**

Japánban az 1960-as években fejlesztették ki a vonalkódot termékazonosításra. Az ok, hogy a pénztárosok ínhüvelygyulladászt kaptak a sok gépeléstől. A vonalkódnak köszönhetően, ami 20 karaktert, betűt és számokat tudott tárolni egy leolvasó segítségével a termék azonosítója a számítógépbe került, nem kellett tehát a pénztárosnak begépelni az árat sem. A vonalkód olvasó készülékeket a DENSO WAVE INCORPORATED (az akkori DENSO CORPORATION divíziója)

fejlesztette. Felmerült az igény arra, hogy ne csak latin karaktereket tároljanak a kódokban, hanem Kanji és Kana karaktereket is. Ehhez nagyobb adatmennyiség tárolása vált szükségessé. A nagyobb adatmennyiség nem csak más karakterek kódolására, hanem jóval több egyszerű információk tárolására is lehetőséget nyújtott, ami nagyon jól jött a logisztikában. A vonalkódokat csak egy irányban lehet leolvasni, olyan kódra volt igény, amely több irányból is leolvasható, dekódolható.



14. ábra QR-kód olvasási irányok, pozíciójelzések [21]

Erre a problémára jelentett megoldást a 2 dimenziós adatmátrix, de még mindig nem lehetett bármilyen irányból leolvasni ezért raktak rá pozíciójelzéseket, egy teljesen egyedi kódot létrehozva.

1994-ben a DENSO WAVE bejelentette a QR-kódot. A QR-kód neve egy rövidítés, **Quick Response Code**, azaz gyors válasz, mivel tízszer gyorsabban lehet dekódolni, mint a hagyományos kódokat. Masahiro Hara ötlete (a QR-kód) nagyban hozzájárult a logisztikai folyamatok gyorsításához, pontosításához. A DENSO WAVE bárki számára szabadon hozzáférhetővé tette a QR-kód leírását, ezzel is nagyban hozzájárult a kód gyors elterjedéséhez.

2000 óta nemzetközi szabvány (ISO/IEC18004).

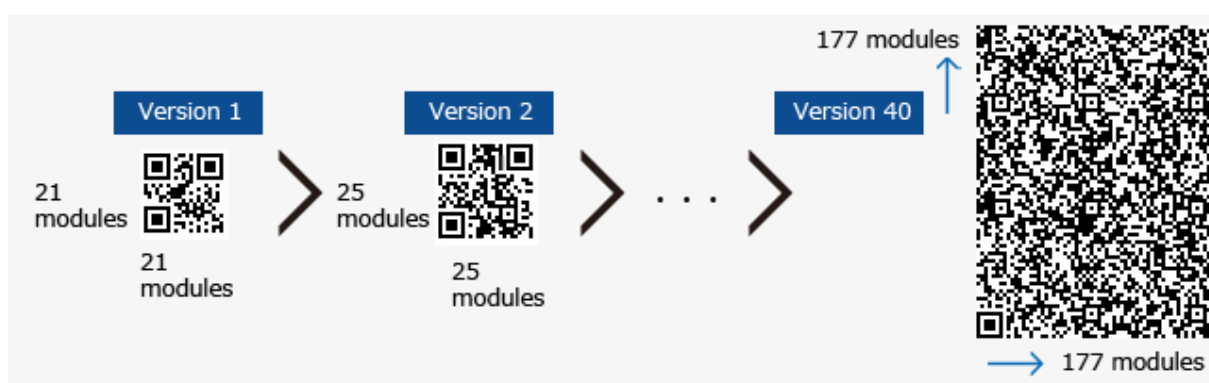
A legnagyobb lökést a 2002-től megjelenő telefonok jelentették, mert így ténylegesen bárki használhatta, leolvashatta a kódokat, amelyek segítségével weboldalat érhetett el és érhet el napjainkban is.

### 5.3.2 A QR-kód felépítése

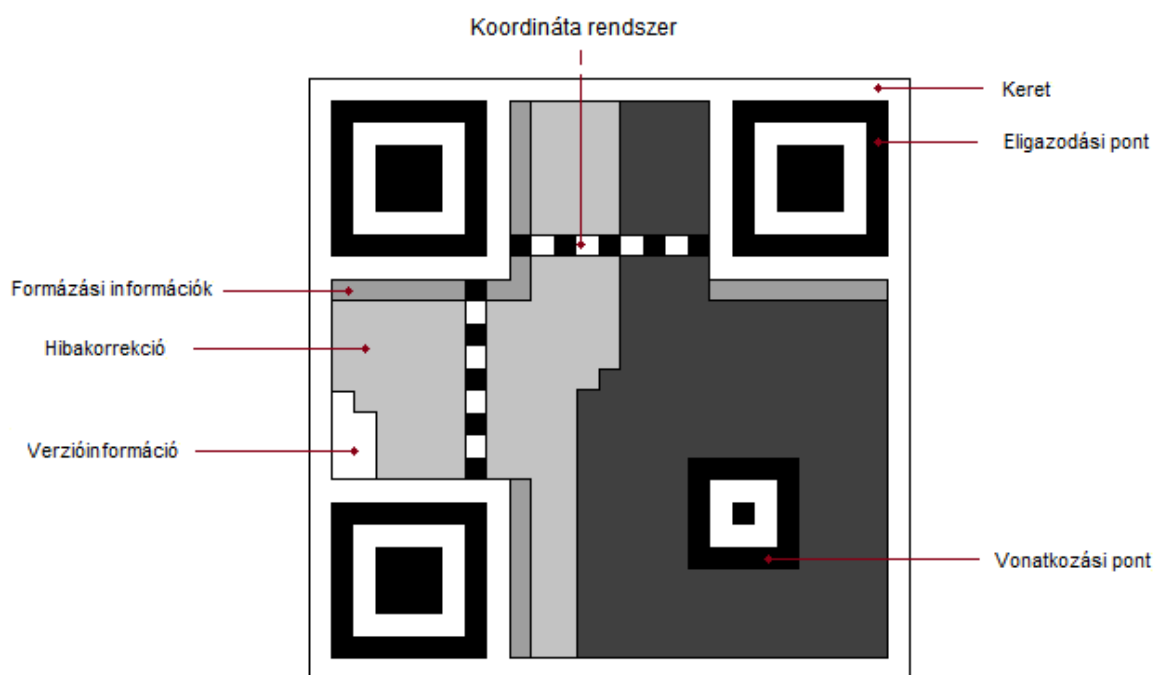
A QR-kód egy 2 dimenziós pont mátrix, pozíciójelzésekkel. 21x21-től 177x177 darab modulból épülhet fel oldalanként 4-4 modullal növekedve. Egy-egy modul egy-egy 8 bites mintának felel meg. A különböző QR-kód verziók eltérő adatmennyiséget képesek tárolni. A tárolt adatmennyiség függ a QR-kód méretétől és a tárolandó

karakterek fajtájától is. Számokból képes a legtöbbet tárolni. Európában a legelterjedtebb a 3-as verziójú QR-kód.

A QR-kód kinyomtatva kisméretű, kosz és sérülésálló a hibajavításnak köszönhetően. Négy hibajavítási szintet tartalmaz a szabvány: L, M, Q, H szinteket, amelyek sorban a kódszavak 7%, 15%, 25% és 30%-át képesek visszaállítani. A hibajavításra a zenei CD-k nyomán ismert Reed-Solomon eljárást használják. A fekete négyzetek egy-egy 8bitet tároló mintát rejtenek, amely az információt tárolja.



16. ábra A QR-kód verziói [21]



15. ábra A QR-kód felépítése [34]

A QR-kód felépítését a 15. ábra mutatja. A 2-es modell tartalmaz egy vonatkozási pontot is, amely segítségével nagyobb szögből és enyhén meggyűrődve

is olvasható marad a kód. A QR-kód jó olvashatóságát segíti, ha a keret legalább a belső négyzeteket alapul véve 4-5 négyzet szélességű és lehetőleg fehér színű.

### 5.3.3 A QR-kód fajtái

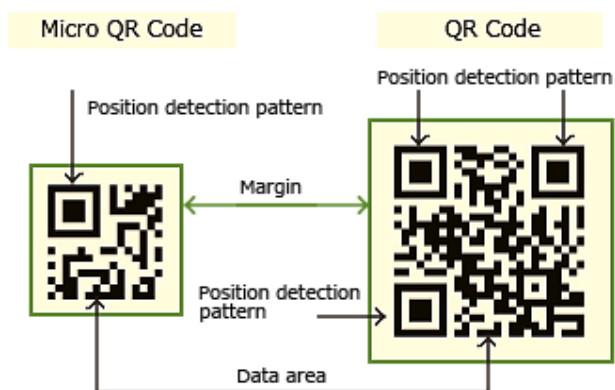
A QR-kód a megszokott fekete-fehér megjelenésén kívül az alábbi fajtái léteznek:

**Mikro QR-kód:** Egyetlen eligazodási pontja van. Előnye, hogy sokkal kisebb, mint az általános változat. A legnagyobb verziója M4-es, amely 17x17 modulból áll 35 számot képes tárolni.

**iQR-kód:** olyan QR-kód, amely nem csak négyzet alakú lehet, hanem az adalmátrixhoz hasonlóan téglalap alakú is. Előnye, hogy kisebb méretben képes a QR-kód által tárolt információt tárolni, jobb a hibajavítási képessége is.

**SQRC:** Formáját tekintve nem különbözik a hagyományos QR-kódtól, hogy a tényleges információ csak a megfelelő készülékkel nyerhető ki belőle. Ebben a kódban tárolható személyes információ, vállalatok magán információi, tehát minden, ami nem tartozik mindenkire. Normál QR-kód olvasóval is le lehet olvasni, de értelmetlen számsort lehet csak kinyerni belőle.

**LogoQ:** Olyan QR-kód, amely egy teljes színes képet ágyaz magába.



17. ábra Általános és mikroméretű QR-kód [21]



18. ábra iQR-kód [21]



19. ábra LogoQ [21]

#### **5.3.4 A QR-kód felhasználása [22]**

Gyártás területen gyártmánykövetésre, folyamatirányításra, rendeléskövetésre, gépek és berendezések kezelésére használják.

Raktározás és logisztika területen csomagok, szállítmányok követésére, a kiskereskedelemben termékazonosításra és leltárkezelésre használják.

Az egészségügyben a leletek kezelésére, betegazonosításra és követésre alkalmazzák.

Ezen kívül QR-kód alapon vásárolhatunk például vonatjegyet is (MÁV-START) [23]. Ekkor elektronikus levél formájában küldik a QR-kódot is, amelyet a jegyellenőrzésnél a telefon kijelzőjéről olvasnak le.

Népszerű a marketing és reklámparban, ahol címetek, URL-eket rejtenek vagy hűségpont illetve kuponként is működhetnek a QR-kódok.

#### **5.4 QR-kód felhasználása a kiterjesztett valóság rendszerhez**

A környezet azonosítására QR-kódok kerülnek felhasználásra, mégpedig a 2-es verzió 2-es modellje, amely tartalmaz a három pozicionálójelzésen kívül egy negyedik jelzést is, hogy nagyobb szögből, akár torzítva is nagy távolságról olvasható legyen.

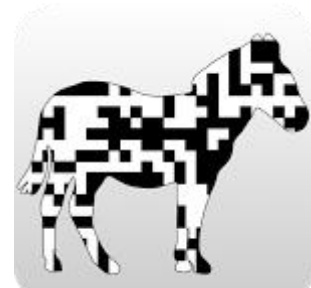
A teszteléshez használt telefon kamerájának gyújtótávolsága 35mm-es rendszerben 24mm-nek felel meg, ez 85°-os látószöget jelent. Az élőkép legnagyobb mérete 800x480pixel. Egy 50x50mm-es 1-es verziójú QR-kódot még 1 méterről felismer. 4 méterről már 150\*150mm-es méret szükséges egy 1-es verziójú QR-kódból.

Előfordulhat, hogy vibráló fény vetül a QR-kódra vagy eleve sötét helyre kell valamiért elhelyezni. Ez gyakran olvashatatlaná teszi a kódot. A probléma orvosolható, ha saját LED-es fényforrás világítja meg a mérőpont jelzését.

## 5.5 A QR-kód azonosítása, dekódolása

A QR-kódok feldolgozása ZXing (Zebra Crossing) projekt csomagjával történik. Ez egy gyors és népszerű nyílt forráskódú csomag, elérhető az Android mellett más mobil és asztali operációs rendszerekre is különböző programozási nyelveken.

Képes többfajta „zebra-kódot” értelmezni, lehetőség van egyszerre több QR-kód olvasására is. Minden felismert és dekódolt kódnak megadja a jellegzetes pontjait. Az alkalmazás ezt használja fel a kirajzolandó információk középpontjának meghatározásához.



20. ábra ZXing logo [25]

A fekete-fehér kódokat az élő képfolyam világossági értékeinek binarizálása után a kép síkfrekvenciájának megnövekedése alapján találja meg, majd értelmezi, hogy azon a helyen valóban QR-kódot talált-e.

Amennyiben sikeres az azonosítás elérhető a QR-kód sarkainak pozíciója és a benne található információ.

A ZXing csomag nem csak a QR-kódok olvasásra, hanem azok generálására is használható. Erről bővebben a kiszolgáló működésénél lesz szó. [24] [25]

## 5.6 Kommunikáció a kiszolgálóval

A mobil eszközön futó alkalmazás egy vékony kliens, amely csak a kihelyezett jelzések azonosításával és az azokhoz tartozó adatok megjelenítésével foglalkozik. A szükséges információkat egy központi kiszolgáló biztosítja, amely a klienseken kívül az ipari folyamatirányító rendszerrel is tartja a kapcsolatot.

Mivel egy gyár területén nem csak egyetlen dolgozónak szándéka információt szerezni a technológiáról, amelynél munkát végez, ezért egyidejűleg több kliens is kapcsolódhat a kiszolgálóhoz. Ehhez érdemes olyan kommunikációs csatornát választani, amely kezeli a több kliens kapcsolódását és kiszolgálását duplex módon.

A kliensnek és a kiszolgálónak is tudniuk kell egymásról, jelezni, ha megszakadt a kapcsolat. A kapcsolat megszakadása főleg a kliens szempontjából fontos. A kiszolgáló számára a kliensek számontartása fontosabb, hogy tudja kik csatlakoztak hozzá és kinek mely információkat kell lekérdeznie és továbbítania.

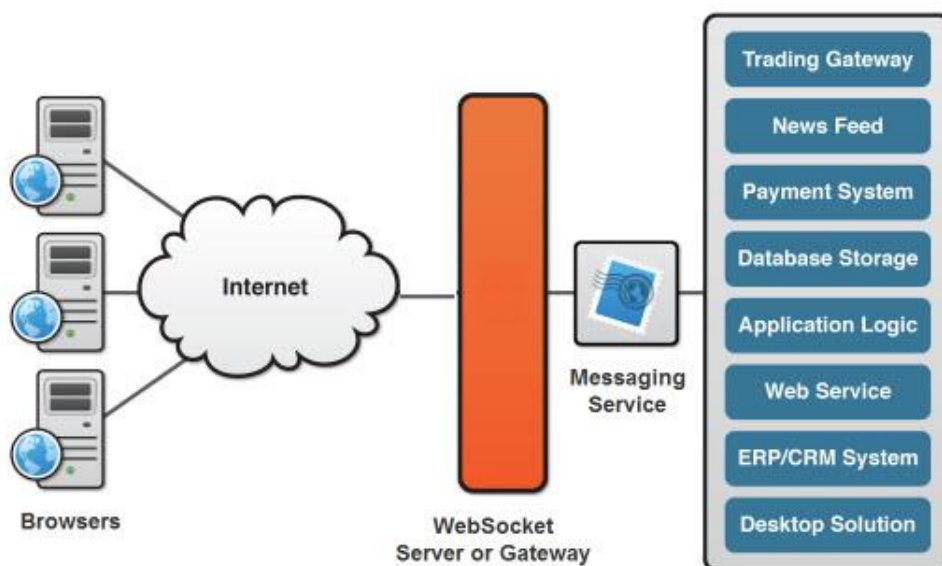
Nem elhanyagolható az üzenetváltás sebessége sem. Lényeges, hogy ne kelljen sokat várni az adatok lekérdezésétől azok megérkezéséig. Néhány, például 2-5 másodperces késés már kevésbé használhatóvá teheti a monitorozó rendszert.

A fenti elvárásoknak eleget tesz a WebSocket technológia, amely kétirányú kapcsolatot biztosít egyetlen TCP socket-en keresztül alacsony fejlécforgalommal, ezzel is segítve a gyors üzenetváltást.

### 5.6.1 WebSocket

A WebSocket protokoll a HTML5 részeként jött létre, kétirányú üzenetkapcsolatot biztosít egyetlen socket kapcsolaton keresztül a kliens és a szerver között. A WebSocket 2011 óta szabvány (RFC6455), ami leegyszerűsíti a kétirányú webes kommunikációt és kapcsolatkezelést. Segítségével sokkal gyorsabb üzenetváltás valósítható meg, mint az eddigi hasonló megoldásokkal http alapon, 2byte adat küldése 150ms-ről 50ms-ra csökken a WebSocket használatával. Lehetővé válik a valós idejű, eseményvezérelt webes alkalmazások készítése a felesleges hálózati forgalom csökkenése mellett.

A WebSocket használható tűzfalakon és proxy-kon keresztül is. Egyszerre több klienssel tartja a kapcsolatot a szerver. Lehetőséget biztosít, az egyéni és a csoportos válaszadásra a kliensek felé. [26]



21. ábra WebSocket vázlat [26]

### 5.6.2 Üzenetek a kiszolgáló és a kliens között

Az alkalmazás indulásakor automatikusan létrejön a kapcsolat a kiszolgálóval és fut a jelzések keresése. Egyszerű szöveges üzeneteket küldenek egymásnak, amikor egy jelzés felismerése, értelmezése megtörtént létrejön egy objektum, amely az alkalmazás kliensén keresztül kérést küld a kiszolgálónak a saját azonosítójával és az alkalmazást használó személy azonosítójával:

```
GET@[azonosító]@[mérőhely_azonosító]
```

A kiszolgáló erre válaszként elküldi a megfelelő adatokat a kliensnek:

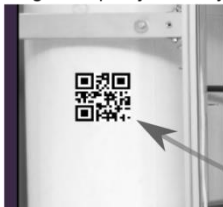
```
DATA@[mérőhely_azonosító]@[mérőhely_neve]##[mérőhely_állapota]@[adat]##[mértékegység]
```

Az adat-mértékegység páros többször ismétlődhet, az első három helyen a legfontosabb adatokat kell szerepeltetni, ezeket rajzolja az alkalmazás közvetlenül a felismert kódra.

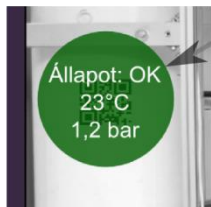
## 5.7 Felhasználói felület és működés

Ahogy a 22. ábra mutatja, az alkalmazás a jelzések fölé egy-egy kört rajzol, amelyen megjelennek a technológiára jellemző folyamatosan frissített valós idejű információk.

A jelzés a technológia megfelelő pontján elhelyezve



A jelzésen elhelyezett információs buborék



22. ábra Illusztráció az felhasználói felületről

### 5.7.1 Információ a jelzések fölött

Egy-egy információs buborékon megjelenik a mérőpont neve, az állapota és legfeljebb 3 mért adat. Az adatokat a kliens kérdezi le másodpercenként, de ez az idő változtatható. A képernyőn bármilyen QR-kód helye megjelenik alapértelmezetten „N/A” felirattal. Ekkor bekerül a hozzátartozó információs buborék objektuma egy listába. Az objektum tartalmazza a QR-kódból kiolvasott információt, a képernyőn elfoglalt helyének középpontját és egy adatstruktúrát, amely feltöltésre vár. Az objektum létrejöttkor regisztrálja magát a WebSocket kliensnél, aki másodpercenként a listát bejárva elküldi a kéréseket a kiszolgálónak és folyamatosan várja a választ. A beérkezett válaszokban található azonosító alapján interface-en keresztül beírja az adatokat az objektumba. Amíg az adatok nem íródnak be addig „N/A” jelzés látható az információs buborékban, beírás után a mérőpontra jellemző információk jelennek meg. Tehát a WebSocket kliens a listájában lévő információs buborékokat időközönként lekérdezi.

Az információs buborékok életciklusa a felismeréssel kezdődik, amikor létrejön a hozzájuk tartozó objektum és bekerül az alkalmazás listájába. A listát folyamatosan kirajzolja az alkalmazás. Mivel nem minden előnézeti képen sikerül minden jelzést felismerni, ezért az információs buborék objektumoknak van egy minimális láthatósági ideje, ez 2 másodperc. Minden felismert objektum legalább 2 másodpercig megjelenik a képernyőn. Ez alatt a felismerés sikerességétől függően frissül a pozíciója. A lágyabb mozgás érdekében nem azonnal az új pozícióra ugrik át, hanem átlagolással átúszik az új helyére. A kirajzolt információs buborék 2 másodperc után nem törlődik azonnal, hanem csak a láthatósága szűnik meg. Ezt követően még 6 másodpercig kapja az információt.

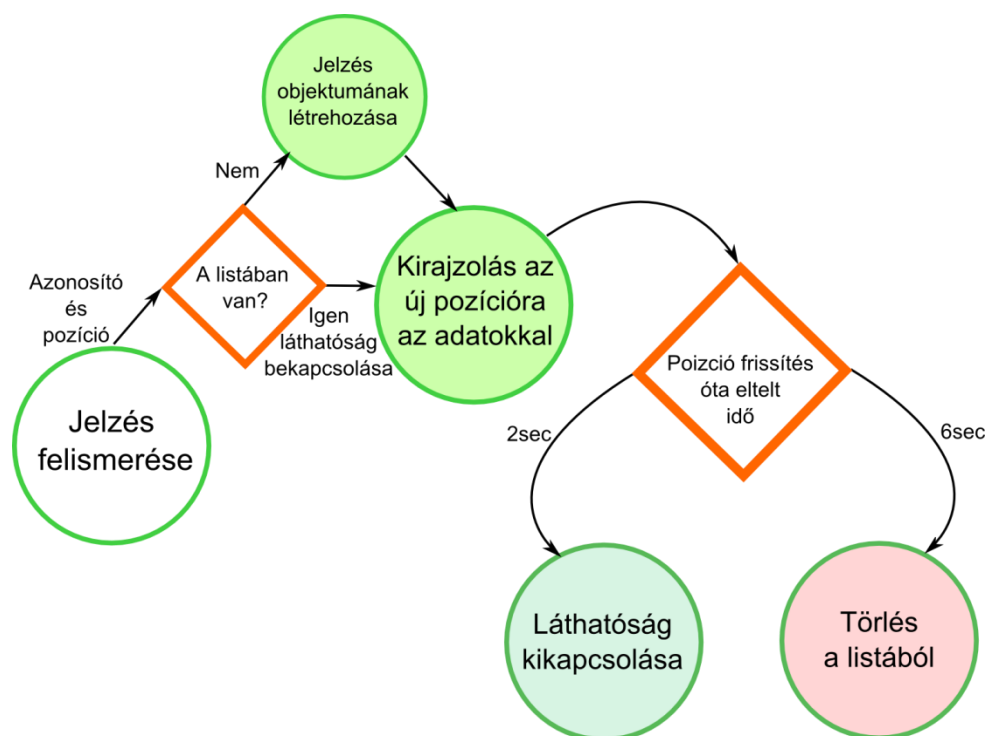
A jelzés a technológia megfelelő pontján elhelyezve



A jelzésen elhelyezett információs buborék



**23. ábra Jelzés és rajta az információs buborék**



**24. ábra Az információs buborék életciklusa**

Egy információs buborék összesen legalább 8 másodpercig tartózkodik a listában a felismerést követően, ez az az eset, amikor csak egyetlen egyszer sikerül azt a QR-kódot felismerni. A további láthatatlan állapotra azért van szükség, hogy ha valaki pásztázik, akkor ne kelljen újra új objektumot létrehozni, hanem csak a láthatóságát megváltoztatni és azonnal a friss információkkal jelenik meg újra az információs buborék. A 2 és 6 másodperc tapasztalati úton került meghatározásra, a további tesztek és használat folyamán valószínűleg még változhat.

Ha 8 másodpercig nem frissül egy információs buborék pozíciója, akkor az objektuma törlődik a listából, ezzel a WebSocket kliens listájából is törlődik, nem kerül lekérdezésre tovább a hozzátartozó információ.

Az információs buborékok mérete igazodik hozzájuk tartozó QR-kódok méretéhez, de van egy minimális méretük, mert ha túl kicsi a képen a QR-kód, akkor olvashatatlanul válhat a megjelenített információ. Maximális mérete az előkép, vagyis a képernyő pixelben mért mérete.

### **5.7.2 A bővebb információs buborék**

Mivel az alap információs buborékokon csak minimális és fontos információk jelennek meg, ezért szükség van arra, hogy egy-egy mérőpontról további információ jelenjen meg. Ezt a mérőpont információs buborékának megérintésével lehet elérni. Az érintést követően a kiválasztott információs buborék egy teljes képernyős mezőt nyit magának, amelyen minden, a mérőponthoz tartozó információ megjelenik folyamatosan frissítve.

Mi történik ilyenkor az információs buborékok életciklusában? A kiválasztott buborék ekkor kitüntetett szerepet kap, nem kerül ki a listából akkor sem, ha a QR-kódja nem látható a kameraképen, a többiek pedig végigmennek a cikluson és törlődnek a listából. Ezzel az is lehetővé válik, hogy ha egy nehezebb gépet kell maga előtt tartania a dolgozónak, nyugodtan kényelmesebb pozícióba eresztetheti az értékek elemzésének idejére.

Fejlesztés alatt áll, hogy ne csak számok, hanem indokolt esetben grafikonok és egyéb grafikus elemek is megjelenjenek a teljes képernyős nézetben ezzel is segítve a gyors információszerzést a technológiáról.

### **5.7.3 Beállítások panel**

A beállítások panelen adható meg a központi kiszolgáló címe, portja, amin figyel. Továbbá a felhasználó neve és azonosítója és a lekérdezések időköze.

A beállításokat az Android beépített Shared Preferences módszerével tárolja. Ezen a helyen az alkalmazás teljes leállítása után is megmaradnak a szükséges információk, onnan csak az alkalmazás törlésével tűnnek el. Más alkalmazások nem férnek hozzá a tárolt információkhoz csak a létrehozója, hogy ha privát módban jöttek létre a beállítások.

Az alkalmazás induláskor lekérdezi a beállításokat és azoknak megfelelően indul. Az első indításkor ezek az információk üresek, ezért az alkalmazás kéri őket.

A felhasználó nevére és azonosítójára a kiszolgálóban történő regisztrálás miatt van szükség, ez információt szolgáltat arról, hogy ki éppen melyik mérőpontnál tartózkodik, éppen használatban van-e a készülék és ki által.

A lekérdezések időköze a WebSocket kliensre vonatkozik, hogy milyen gyakorisággal járja be az információs buborékok listáját, vagyis milyen sűrűn frissítse a kirajzolt információkat.

#### **5.7.4 Az alkalmazás leállítása**

A „Vissza” gomb hatására az alkalmazás megkérdezi a felhasználót, hogy valóban ki szeretne lépni vagy sem. Ha igen, akkor elengedi a kamerát, mert ez egy osztott erőforrás az Android-ban és egyszerre csak egy program használhatja. Megszakad a kapcsolat a központi kiszolgálóval, leáll a WebSocket kliens és végül az alkalmazás is kilép.

## 6 A központi kiszolgáló

---

A központi kiszolgáló lényegében átjáróként és adatbázisként működik a rendszerben. Kapcsolatot tart a kliensekkel és az ipari szerverrel, a kliensek által küldött azonosító alapján lekérdezi az érzékelők adatait és továbbítja a kérésnek megfelelő kliensnek. Számon tartja az aktív klienseket. Ezen kívül elvégezhető rajta a QR-kódok generálása és bennük kódolt azonosító összerendelése a technológia érzékelőivel. A központi kiszolgáló Java nyelven készült.

### 6.1 Kapcsolódás az ipari rendszerhez

Az alkalmazás a tanszéki DCS rendszeren kerül tesztelésre, amely OPC szerver OPC DA klienseket képes kiszolgálni. A kapcsolódáshoz Java alapú nyílt forrású OPC klienst kerestem, amely integrálható a központi kiszolgálóba.

#### 6.1.1 Az OPC kapcsolat [27]

Az OPC nyitott szabvány (OLE for **P**rocess **C**ontrol, OLE – **O**bject **L**inking and **E**mbdding) 1996-ban jelent meg először. Célja, hogy a PLC protokollok egy egységes felületen keresztül folyamatosan és torzítatlanul legyenek elérhetőek. Az OPC-nek két fontos eleme van, az OPC szerver és az OPC kliens.

Az OPC szerver kapcsolatban áll a terepi műszerekkel és egy jól definiált felületen keresztül biztosítja az összegyűjtött adatok elérését az OPC kliensek számára.

Az OPC kliensek az OPC szerver által biztosított adatok felhasználói.

#### 6.1.2 OPC kapcsolat fajtái [27]

OPC DA: Data Access, amely adatelérést tesz lehetővé, értékek, idő és minőség információk lekérdezését.

OPC AE: Alarm and Events, amely az alarm és esemény típusú üzenetek cseréjét írja le.

OPC HDA: Historical Data Access, lekérdezési eljárásokat és elemzéseket határoz meg, amelyek időbélyegzett adatokhoz használhatóak.

OPC UA: Unified Architecture, egy platform független szolgáltatásorientált architektúra, amely a fentebb felsorolt leírásokat egyesíti.

### 6.1.3 UTGARD - OPC DA kliens [28]

Az openSCADA részeként készült nyílt forrású, ingyenes, tisztán Java nyelven írt OPC kliens, amely jelenleg csak OPC DA 2.0 hozzáférést támogat. Használható az openSCADA-tól függetlenül is. A tisztán Java nyelv azt jelenti, hogy nem tartalmaz JNI (Java Native Interface) állományokat. Ennek köszönhetően bármilyen Java-t futtatni képes rendszeren használható.

Tehát a központi kiszolgáló az UTGARD OPC DA kliens segítségével kapcsolódik a tanszéki OPC serverhez.

## 6.2 QR-kódok generálása és összerendelése az érzékelőkkel

A QR-kódokat a kiszolgáló a ZXing csomag segítségével generálja és menti el PNG (Portable Network Graphics) formátumba. A PNG egy veszteségmentes tömörített formátum, amely kezeli az átlátszóságot és gamma korrekciót. Ezek a nyomtatásnál játszhatnak fontos szerepet. Az elkészült QR-kód megjelenik a kijelzőn, ha megtörtént az összerendelés helyben ki is próbálható, hogy a kívánt adatok jelennek-e meg a mobil készülék kijelzőjén. A kész kódokat a program a saját könyvtárában tárolja, az „Exportálás...” gombbal a kívánt helyre menthető. (25. ábra)



25. ábra Központi kiszolgáló, jelzések összerendelése

A QR-kódban kódolt azonosítóhoz tartozó OPC címeket egy XML állományban tárolja, amely formátuma a következő:

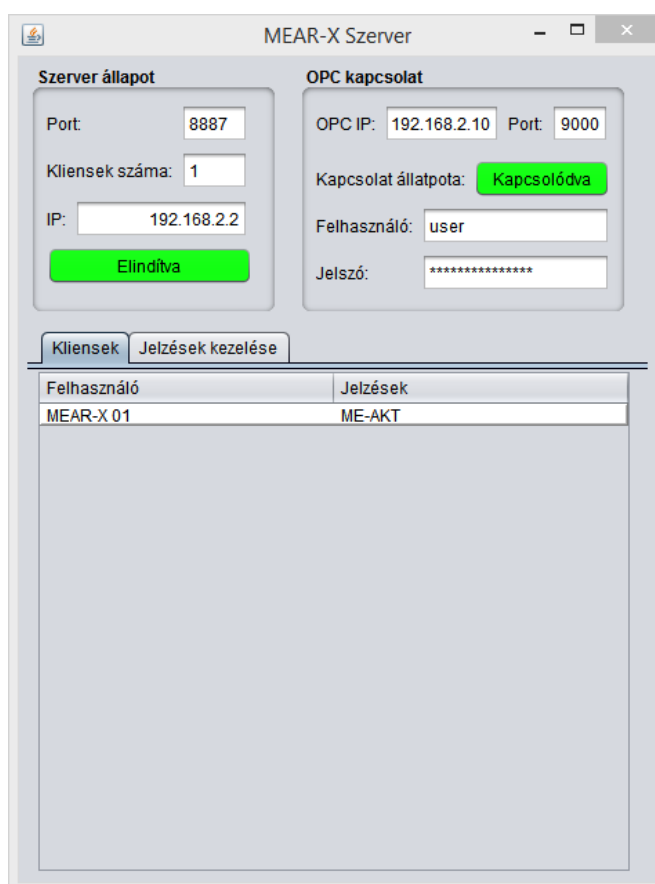
```
<?xml version="1.0" encoding="UTF-8"?>
<markers>
  <marker>
    <id>mérőpnt_azonosító</id>
    <name>mérpont neve</name>
    <sensor>
      <sid>opc_cím</sid>
      <sunit>mértékegység<sunit>
    </sensor>
  </marker>
</markers>
```

A marker és a sensor tag-ek ismétlődnek a jelzések és a megjeleníteni kívánt érzékelők számának megfelelően.

### 6.3 Kapcsolattartás a kliensekkel

A kliensekkel a kapcsolatot a WebSocket szervere tartja. Amikor egy kliens elküldi a kérését, a kiszolgáló feljegyzi a mérőpont azonosítóját és az összerendelés alapján lekérdezi az ipari szerverről az értékeket. Az adatok megérkezésekor összeállítja a csomagot és továbbítja a kérést kezdeményező kliensnek.

Emellett egy listában megjeleníti, hogy mely kliensek elérhetőek a rendszerben, illetve éppen mely mérőpontokat kérdezik le. Ez információt ad arról, hogy a felhasználók merre mozognak a gyár területén.



26. ábra Központi kiszolgáló, kliensek listája

## **6.4 A központi kiszolgáló előnye**

A központi kiszolgáló segítségével megoldható az is, hogy egy mérőponthoz más értékeket jelenítsen meg például érzékelő csere esetén vagy abban az esetben, ha mégsem az az érték szükséges, amelyet eredetileg az információs buborék megjelenít.

A központi kiszolgáló váza (grafikus felület, WebSocket szerver, QR-kód generátor) állandó maradhat a különféle rendszerekhez könnyen alakítható, ha az ipari rendszerrel való kapcsolatért felelős csomagot kicseréljük így szinte bármilyen rendszerhez illeszthető a kiterjesztett valóság rendszer.

## 7 Elvárások a kihelyezett jelzésekkel szemben

---

A kihelyezett QR-kódok esetében nagyon fontos a jó olvashatóság a kamera szempontjából. Az olvashatóságot meghatározza a kliens kamerájának minősége, az élőkép mérete, a kihelyezett jelzések környezetének fényviszonyai, a jelzést hordozó felület tulajdonságai és a jelzés mérete, mely befolyásolja a leolvasási távolságot.

A jelzést hordozó felülettel szembeni elvárás, hogy matt legyen, ne csillogjon, ragyogjon rajta a környező fényforrások fénye, mert a kamera nem képes a túl nagy fényerőkülönbséget kezelni és megghiúsul a felismerés.

A jelzés mérete azt határozza meg, hogy milyen távolságból lehet leolvasni a kódot. 150x150 mm-es jelzést jó látási viszonyok mellett 3-4 méter távolságból felismer az alkalmazás egy átlagos hátoldali kamerával. Ez az esetek többségében elegendőnek bizonyul. Tovább a jelzést nem feltétlenül muszáj közvetlenül az érzékelő közelébe helyezni, ha annak mért adatai kissé távolabb való megjelenítése sem ad okot a félreértelmezésre.

Tapasztalatunk szerint, ha a jelzés bár matt, jól látható felületen van, akkor is előfordulhat, hogy nem képes a készülék értelmezni. Ennek oka a vibráló fény. Bár ipari környezetben erre egyébként is külön figyelmet fordítanak, de előfordulhat, hogy olyan közel van egy fénycsöves világítótesthez, hogy vibráló fény éri a jelzést. Erre és a sötét helyen elhelyezett jelzésre is megoldás a saját LED-es fényforrás, amely szemből megvilágítja a jelzést.

Fontos elvárás, hogy a jelzés síkfelületen legyen elhelyezve ezzel is növelve az olvashatóságát a kamera számára. Mivel a merőleges megfigyelés nem elvárható ezért az oldalról történő megfigyelés miatt a síkfelületre elhelyezett jelzés eleve torzítva fog megjelenni a kamera képén, de ha egy viszonylag egyenes jelzés megy át a torzuláson, azt még az algoritmus fel fogja ismerni.

A QR-kód akkor ismerhető fel biztosan, ha a kód körül van egy homogén és lehetőleg világos keret, amelynek a szélessége legalább 4 négyzetnyi az adatterületen található négyzetekből. [21]

## 8 Fejlesztési tervek

---

A legközelebbi fejlesztési terv, hogy a kliensek egymás között képesek legyenek kommunikálni. Lehetőséget biztosítva arra, hogy az éppen megjelenített adatokat a felhasználó egy másik felhasználó képernyőjére továbbítsa, és szöveges úton megbeszélhessék a problémát, gyors megoldási javaslatot kaphat az, aki a segítséget kérte.

Jelenleg a kiterjesztett valóság rendszer csak passzív megfigyelést tesz lehetővé, azonban sokszor szükség lehet beavatkozásra is, egy szelep kinyitására vagy más műveletre. Ehhez meg kell oldani a felhasználó azonosítását, az esetleges jogosultsági szintek kezelését (ki avatkozhat be és melyik rendszerbe?) és ha a felhasználó valahol leteszi a táblagépét, más ne vezérelhesse a rendszert. Kérdéses, hogy ekkor az alkalmazás a felhasználóra bízsa a képernyő lezárását vagy valamilyen módszerrel, például a készülék mozgása, közelségérzékelő használata alapján érzékelje a tétlenséget és automatikusan lezárja a képernyőt.

Egy ilyen alkalmazás nem csak megfigyelésre vagy vezérlésre használható. Nagy segítséget nyújthat a betanulás ideje alatt. Ez alatt nem csak az új munkaerőnek jelenthet segítséget, hanem a tapasztalt kollégáknak abban az esetben, ha egy új üzemszert kell kezelniük. Az állapotinformációk mellett megjelenhet például az, hogy milyen határértékeken belül működik optimálisan a rendszer. Természetesen közben is működik a szöveges üzenetváltás a többi felhasználóval akár egy operátorral is.

Karbantartási munkálatoknál külön segítséget nyújthat a rendszer, ha elnavigálja a karbantartót a hiba helyéhez. Ez több módszerrel is megoldható. Egyik, hogy a felismert jelzések alapján határozza meg a pillanatnyi helyét és ez alapján jelzi, merre van a hiba. Vagy folyamatosan a kamerakép alapján az elmozdulásból és a jelzésekől pontosan számítható merre jár a felhasználó, a térkép alapján pedig az, hogy merre kell mennie.

Mivel egy 500-1000g-os készüléket is nehéz sokáig tartani ezért az alkalmazást érdemes kompatibilissé tenni olyan viselhető eszközökkel, mint a Google Glass [29] vagy az Epson Moverio [30] esetleg a Vuzix [31] kijelzővel és kamerával ellátott szemüvege. Ezáltal a felhasználó kezei szabadon maradnak, miközben ugyan úgy megjelennek előtte a kívánt információk.

## 9 Összegzés

---

Az intelligens mobil eszközök fejlődésével egyre szélesebb körben elérhetővé válik a régóta ismert, de mégis újszerű technológia, a kiterjesztett valóság a mindennapi életben. Már jelenleg is számos kiterjesztett valóság alkalmazást lehet találni a polgári, orvosi, mérnöki, oktatási alkalmazásban. A technológia lényege, hogy egy kamera által a képernyőn vagy kijelzővel ellátott szemüvegen keresztül látott valóság kibővül virtuális objektumokkal, információkkal.

Ipari alkalmazására is lehet példákat találni, főleg a modellek, CAD rendszerek térbeli objektumainak megjelenítésével. Néhány tanulmány és alaprendszer is megjelent az elmúlt hónapokban, amely egy folyamatirányító rendszer monitorozását segíti egy táblaszámítógép vagy intelligens szemüveg segítségével.

A dolgozatban is egy folyamatirányításban is használható monitorozó rendszer került bemutatásra. Alapja, hogy a rendszerben bizonyos kitüntetett pontokat QR-kódok jelölnek, amikor ezek valamelyike a táblaszámítógép kamerájának látóterébe kerül, a képernyőn a QR-kód képernyőn elfoglalt helyének megfelelő helyen megjelennek olyan valós idejű információk, amelyek a technológia adott pontján érvényesek, jellemzőek. Az információkat a táblaszámítógép alkalmazása egy központi kiszolgálón keresztül kérdezi le a folyamatirányító rendszerről például OPC kapcsolaton keresztül. Az így megjelenített információk egyfajta előnézeti adatok. Az információs buborékokra bökve további adatok olvashatók ki a rendszerből, amelyek az adott mérőponthoz kapcsolódnak.

Az eddig elkészült programpáros - az Android operációs rendszert futtató mobil eszközön futó kliens program és a folyamatirányító rendszerhez kapcsolódó központi kiszolgáló – egy minta arra, hogyan használható a kiterjesztett valóság a folyamatirányításban, egy ipari rendszer monitorozására.

*"A kutató munka a Miskolci Egyetem stratégiai kutatási területén működő Mechatronikai és Logisztikai Kiválósági Központ keretében valósult meg."*

## 10 Források

---

- [1] „A mobil kiterjesztett valóság története,” [Online]. Elérhető: <https://www.icg.tugraz.at/~daniel/HistoryOfMobileAR/>.
- [2] „For real! The incredible story of augmented reality,” Nokia, 13. 01. 2013. [Online]. Elérhető: <http://conversations.nokia.com/2013/01/13/for-real-the-amazing-story-of-augmented-reality/>.
- [3] „Myron Krueger - Videoplace,” 1974. [Online]. Elérhető: <http://www.youtube.com/watch?v=dmmxVA5xhuo>.
- [4] J. Rekimoto, „Matrix: a realtime object identification and registration method for augmented reality,” 1996. [Online]. Elérhető: <http://www.sonyosl.co.jp/person/rekimoto/papers/apchi98.pdf>.
- [5] [Online]. Elérhető: <http://www.hitl.washington.edu/artoolkit/>. [Hozzáférés dátuma: 2014].
- [6] T. W. D. Gerhard Reitmayr, „Going out: robust model-based tracking for outdoor augmented reality,” *Mixed and Augmented Reality, 2006. ISMAR 2006. IEEE/ACM International Symposium on*, 10.1109/ISMAR.2006.297801, pp. 109-118, 2006.
- [7] [Online]. Elérhető: <https://www.layar.com/>.
- [8] [Online]. Elérhető: <http://www.google.com/mobile/skymap/>. [Hozzáférés dátuma: 2014].
- [9] 2013. [Online]. Elérhető: <http://www.gizmag.com/ikea-augmented-reality-catalog-app/28703/>. [Hozzáférés dátuma: 2014].
- [10] [Online]. Elérhető: <http://www.artoolworks.com/products/desktop/artoolkit-for-desktop/>. [Hozzáférés dátuma: 2014].
- [11] Qualcomm, [Online]. Elérhető: <http://www.qualcomm.com/solutions/augmented-reality>. [Hozzáférés dátuma: 2014].
- [12] [Online]. Elérhető: <https://www.vuforia.com/>. [Hozzáférés dátuma: 2014].
- [13] [Online]. Elérhető: <http://www.metaio.com/>. [Hozzáférés dátuma: 2014].
- [14] Siemens, „Industry – Augmented Reality,” 2002. [Online]. Elérhető: [http://www.siemens.com/innovation/pool/en/publikationen/publications\\_pof/pof\\_fall\\_2002/industry\\_articles/augmented\\_reality/PoF104art15\\_1202602.pdf](http://www.siemens.com/innovation/pool/en/publikationen/publications_pof/pof_fall_2002/industry_articles/augmented_reality/PoF104art15_1202602.pdf).
- [15] „Google Glass az ipari automatizálásban,” 2014. [Online]. Elérhető: [http://www.gyartastrend.hu/gyartastechnologia/cikk/google\\_glass\\_az\\_ipari\\_automatizalasban](http://www.gyartastrend.hu/gyartastechnologia/cikk/google_glass_az_ipari_automatizalasban).

- [16] Beckhoff, [Online]. Elérhető: <https://www.beckhoff.com/english.asp?press/news1613.htm>. [Hozzáférés dátuma: 2014].
- [17] [Online]. Elérhető: <http://www.youtube.com/watch?v=8o2TkYGCgUQ>. [Hozzáférés dátuma: 2014].
- [18] [Online]. Elérhető: <http://trends.directindustry.com/news-trends/industrial-automation-soon-to-be-seen-through-google-glass/>. [Hozzáférés dátuma: 2014].
- [19] „Agilent Mobile Logger,” Agilent Technologies, 2012. [Online]. Elérhető: [https://play.google.com/store/apps/details?id=com.agilent.mobilemeterlogger&feature=also\\_installed](https://play.google.com/store/apps/details?id=com.agilent.mobilemeterlogger&feature=also_installed).
- [20] [Online]. Elérhető: [http://en.wikipedia.org/wiki/Aztec\\_Code](http://en.wikipedia.org/wiki/Aztec_Code). [Hozzáférés dátuma: 2014].
- [21] Denso Wave, [Online]. Elérhető: [www.qrcode.com](http://www.qrcode.com).
- [22] [Online]. Elérhető: <http://www.nacs.org/LinkClick.aspx?fileticket=D1FpVAwvJuo%3D&tabid=1426&mid=4802>.
- [23] [Online]. Elérhető: <http://www.mav-start.hu/hirek/hir.php?mid=150c5b800cd127>.
- [24] [Online]. Elérhető: <http://zxingnet.codeplex.com/>. [Hozzáférés dátuma: 2014].
- [25] [Online]. Elérhető: <https://github.com/zxing/zxing>. [Hozzáférés dátuma: 2014].
- [26] [Online]. Elérhető: [www.websocket.org](http://www.websocket.org). [Hozzáférés dátuma: 2014].
- [27] OPC Foundation, [Online]. Elérhető: <https://opcfoundation.org/>. [Hozzáférés dátuma: 2014].
- [28] OpenSCADA, [Online]. Elérhető: <http://openscada.org/projects/utgard/>.
- [29] Google, [Online]. Elérhető: <http://www.google.com/glass/start/>.
- [30] Epson, [Online]. Elérhető: <https://www.epson.com/cgi-bin/Store/jsp/Moverio/Home.do>.
- [31] Vuzix, [Online]. Elérhető: <http://www.vuzix.com/>.
- [32] „Augmented reality: The past, present and future,” 03. 07. 2011. [Online]. Elérhető: <http://thenextweb.com/insider/2011/07/03/augmented-reality-the-past-present-and-future/>.
- [33] „Mobile Computing in Zone2,” [Online]. Elérhető: <http://www.mobile-computing-hazardous-area.com/en/learn-about/z710-ex/>. [Hozzáférés dátuma: 2014].
- [34] [Online]. Elérhető: <http://www.qr-kod.hu/>.

Linkek ellenőrizve: 2014-04-14