

---

# K+F projekt dokumentáció ACSG Kft.

Időszak:  
2023.01.01. - 2023.02.28.

Készítette: Wenez Dominik

---



Advanced Cableharness Solution Group

# Tartalomjegyzék

<b>1. Összefoglaló</b>	<b>2</b>
<b>2. Objektum detektáló alrendszer</b>	<b>2</b>
2.1. Vizsgált hálóarchitektúrák . . . . .	2
<b>3. Tanítóadat generálás</b>	<b>2</b>
3.1. Probléma ismertetése . . . . .	2
3.2. Megoldási lehetőségek . . . . .	2
3.3. Vegyes tanító adat . . . . .	2
3.4. Valós adatra illesztés . . . . .	2
3.5. További fejlesztési irányok . . . . .	2
<b>4. Pozíciómeghatározás</b>	<b>2</b>
4.0.1. Megvizsgált módszerek . . . . .	3
<b>5. Robot manipulátor</b>	<b>3</b>
<b>6. Irodalomjegyzék</b>	<b>3</b>

# 1. Összefoglaló

A dokumentum által tekintett időintervallumon belül megtörtént a robotkar (SCARA) pontosságának tesztelése a munkatérben, melynek eredményeként elmondható, hogy a csatlakozóházak megfelelően precíz manipulálásához megfelelő az alkalmazott robotkar.

A pontosság mérésének tükrében a későbbi digitális iker tervezése is megkezdődött, de a tekintett időszakban mindösszesen a szimulációs keretrendszer beállítása indult el egy Linux operációs rendszerű PC-n.

A mélytanulós objektumdetektálás tanítóadat generáló szoftvere a jelenlegi teszt alkalmazásokhoz teljesen elkészült, a későbbiekben a rendszerszintű integráció kapcsán lesz csak szükséges további változtatás eszközölése rajta.

Folytattuk a különböző Deep Learningen alapuló objektum detektáló hálóarchitektúra vizsgálatát a csatlakozóházak detektálására.

Az objektum detektáláson túl a pozíciómeghatározás lehetséges megoldásait vizsgáltuk, melynek eredményei lentebb láthatók, illetve az objektum detektálást részletesen tárgyaló dokumentumba felvezetésre kerültek.

## 2. Objektum detektáló alrendszer

Az előző dokumentumban kifejtésre került, hogy a deep learning egy jó alternatívát jelent a hagyományos képfeldolgozási módszerekre többek közt az objektumdetektálás területén belül is.

### 2.1. Vizsgált hálóarchitektúrák

## 3. Tanítóadat generálás

### 3.1. Probléma ismertetése

### 3.2. Megoldási lehetőségek

### 3.3. Vegyes tanító adat

### 3.4. Valós adatra illesztés

### 3.5. További fejlesztési irányok

## 4. Pozíciómeghatározás

A fentebb említett hálóarchitektúrák önmagukban mindösszesen az objektumdetektálásra képesek, mint azt az előző részekben megállapítottuk, jelen alkalmazási formára a sebesség igényénél fogva elegendő a bounding-box típusú objektu felismerést alkalmaznunk, hiszen elegendő adatot tartalmaz a robotvezérléshez és szignifikánsan gyorsabb, mint a szegmentálással is kiegészített hálóarchitektúra.

Azonban a megfelelő fészekbe való pozícionáláshoz további input paraméterekre van szükségünk, az objektum elfordultsága is nélkülözhetetlen paraméter ha egy kamerával szeretnénk megoldani a feladatot. Azonban itt több lehetőségünk is van a megvalósításra, a dokumentum által tárgyalt időszakban ezen lehetőségeket térképeztük fel és alkalmazhatóságukat az adott problémán elkezdtük megvizsgálni.

Az általunk megfontolandónak talált megközelítések a következők:

- Egy a munkaterület felett elhelyezett kamerától különálló kamera, mely a fészek felett van elhelyezve és hagyományos képfeldolgozási technikával (pl.: szögműködés meghatározás) megfelelő rotációs értéket adja a robotvezérlőnek. Természetesen léteznek nem explicit szög meghatározó algoritmusok is, melyek egy adott hibaminimalizációra, vagy mintakövetésre törekzenek, azaz kvázi regresszív módszerrel közelítik a megfelelő orientációt, mellyel a fészekbe kell kerülnie a csatlakozóháznak.  
Ez a megoldás viszonylag egyszerű, azonban mivel két kamera szükséges hozzá (munkatér nagysága miatt is), illetve nem minden csatlakozó esetén feltétlenül hibamentesen generalizálható, ezért alkalmazhatóságát meg kell vizsgálni részletesebben.
- Az előző esetet egy kamerával is megoldhatjuk ha azt a robotkar és a munkatér, illetve a fészek geometriája lehetővé teszi.

- A hagyományos képfeldolgozási módszereken túl ebben az esetben is használhatunk neurális hálózatokat, azaz deep learningen alapuló módszereket.  
Egyik módszer, hogy már az objektumdetektáló neurális hálózatot bővítjük ki egy orientációt meghatározó résszel. Ekkor annyi szabadságunk van, hogy tetszőleges kondíciókat adhatunk meg, azaz lehetséges a fészek-be való behelyezés több variációval való kiegészítése, illetőleg regressziós módon megadhatunk 1-3 elfordulási komponenst (tengelyek körül), de tekintve, hogy SCARA robottal dolgozunk, így elegendő egy elfordulással operálni ha már megállapítottuk az objektumról, hogy felvehető.  
A rendszer ilyen szintű bővítésével eljutunk oda, hogy implicit meghatározzuk az objektum 3D térben vett 6 szabadságfokú pozícióját, mely a mélytanulás ilyen típusú alkalmazásában egy teljesen külön álló kutatási terület, melyre léteznek megoldási módok, melyeket a későbbiekben tesztelni is fogunk, azonban jelen alkalmazásnál SCARA robotnál egyszerűbb modellel is képesek vagyunk elérni a megfelelő pozicionálást elméleti szinten.
- A pozíció meghatározása nem feltétlen kell, hogy explicit módon történjen, ha egy adott objektum osztályon belül úgynevezett keypoint-okat keresünk, akkor azok elhelyezkedéséből determinálható az objektum pozíciója. Ennek előnye, hogy rengeteg információt tudunk meghatározni egy lépésben, de jellemzően egy ilyen háló betanítása több tanítóadatot igényel és nagyobb osztályszám esetén nem feltétlenül konvergál a valósághoz, továbbá természetesen a végrehajtási idő is jeentősen megnő a háló komplexitásával együtt.

#### 4.0.1. Megvizsgált módszerek

Az adott időintervallumban egyelőre főként csak a lehetséges megoldások elméleti kidolgozása zajlott, illetve a hagyományos képfeldolgozáson alapuló módszerek közül elkezdtek az algoritmusok implementációját is, persze az OpenCV-ben található hasonló algoritmusok, de a feladat specifikussága miatt mindenképp akár ezeket felhasználva is, de saját algoritmust fejlesztenünk a robosztusabb megoldáshoz.

## 5. Robot manipulátor

Miután már kiválasztásra került a technológiai szempontból optimális végberendezés, annak valódi környezetben való tesztelését végeztük, továbbá magának a robotvezérlőnek a beállítása, kalibrálása, a robot felprogramozása és egyszerűbb illetve összetettebb mozgások megvalósítása LUA programnyelven.

Többek között a maximális csuklógyorsulást és csuklósebességet teszteltük. Ezek a robot adatlapjában is megvannak adva számszerűen, azonban mindig érdemes az ilyen jellegű validálás, főként mivel az adatlapok az ideális, adott modellezési módhoz tartozó értékeket tartalmazzák.

(...)

## 6. Irodalomjegyzék