



Képi adatszintézisen alapuló mélytanuló algoritmusok alkalmazása ipari robotikai rendszerek számára

Egyetemi doktori (PhD) értekezés

Kapusi Tibor Péter

Témavezető: Prof. Dr. habil Hajdu András

DEBRECENI EGYETEM

Természettudományi és Műszaki Tudományi Doktori Tanács

Informatikai Tudományok Doktori Iskola

Debrecen, 2026

Ezen értekezést a Debreceni Egyetem Természettudományi és Műszaki Tudományi Doktori Tanács, Informatikai Tudományok Doktori Iskola Diszkrét matematika, adatfeldolgozás és vizualizáció programja keretében készítettem a Debreceni Egyetem műszaki doktori (PhD) fokozatának elnyerése céljából.

Nyilatkozom arról, hogy a tézisekben leírt eredmények nem képezik más PhD disszertáció részét.

Debrecen, 2026

.....
Kapusi Tibor Péter

Tanúsítom, hogy Kapusi Tibor Péter doktorjelölt 2019- 2026. között a fent megnevezett Doktori Iskola Diszkrét matematika, adatfeldolgozás és vizualizáció programjának keretében irányításommal végezte munkáját. Az értekezésben foglalt eredményekhez a jelölt önálló alkotó tevékenységével meghatározóan hozzájárult. Nyilatkozom továbbá arról, hogy a tézisekben leírt eredmények nem képezik más PhD disszertáció részét.

Az értekezés elfogadását javasolom.

Debrecen, 2026

.....
Prof. Dr. habil Hajdu András

Képi adatszintézisen alapuló mélytanuló algoritmusok alkalmazása ipari robotikai rendszerek számára

Értekezés a doktori (Ph.D.) fokozat megszerzése érdekében
az informatikai tudományágban

Írta: Kapusi Tibor Péter okleveles mérnökinformatikus

Készült a Debreceni Egyetem Informatikai Tudományok doktori iskolája
(Diszkrét matematika, adatfeldolgozás és vizualizáció programja) keretében

Témavezető: Prof. Dr. habil Hajdu András

Az értekezés bírálói:

Dr.

Dr.

A bírálóbizottság:

elnök:

Dr.

tagok:

Dr.

Dr.

Dr.

Dr.

Az értekezés védésének időpontja: 20... ..

Tartalomjegyzék

1. Bevezetés	1
2. Módszertanok és eszközök	4
2.1. Az Ipar 4.0 és Ipar 5.0 technológiai bemutatása	5
2.1.1. Az Ipar 4.0 bemutatása	5
2.1.2. Az ipar 5.0 koncepcionális bemutatása	10
2.2. Ipari robotok és hardveres eszközök bemutatása	14
2.2.1. SONY SCARA – SRX-611 robotegység	14
2.2.2. KUKA KR5 robotegység és FLEXLINK XK	17
2.2.3. Sajátfejlesztésű FDM 3D nyomtató és teszt tárgyak nyomtatása	22
2.2.4. NVIDIA Jetson Nano development kit	24
2.3. A valós képek leírhatóságának háttere	25
2.4. Képből kép alapú fordítási technika háttere	27
2.4.1. Adatkészlet-transzformáció és tartományadaptáció	31
2.5. Az alkalmazott mélytanuláson alapú architektúrák és metó- dusok bemutatása	33
2.5.1. GAN hálózatok alapjai	33
2.5.2. pix2pixHD architektúra alapjai	34
2.5.3. YOLO alapú architektúrák	35
2.6. A kiértékelés során alkalmazott metrikák bemutatása	39
3. Képfordítás alapú megközelítés	42
3.1. 3D CAD modellezés és képi nyers adathalmaz generálása	43
3.2. A javasolt architektúrák és megközelítések bemutatása	45
3.2.1. Képből kép fordítás (pix2pixHD)	45

3.2.2.	Az alkalmazott valós idejű működésű, mélytanuló detektorok	47
3.3.	Tanítási és validálási folyamat	47
3.3.1.	pix2pixHD	48
3.3.2.	YOLO alapú detektorok	49
3.4.	Eredmények	51
3.4.1.	pix2pixHD	51
3.4.2.	A YOLO detektorok	52
3.5.	Konklúzió	55
4.	Adatkészlet-fordítási módszer	56
4.1.	A jeleneteket leíró teoretikus modell	58
4.2.	A javasolt mélytanuláson alapuló képtartomány konverziót megvalósító architektúra	59
4.3.	Adatkészletek létrehozása a valós és renderelt képekkel történő adatkészlet-fordítási módszer betanításához	62
4.4.	A mélytanuláson alapuló detektor algoritmusok architektúrájának leírása	65
4.5.	Tanítás és validálás	67
4.5.1.	pix2pixHD	68
4.5.2.	YOLO alapú detektorok	68
4.6.	Eredmények	69
4.6.1.	pix2pixHD	70
4.6.2.	YOLO alapú detektorok	72
4.7.	A valós alkalmazás bemutatása	72
4.8.	A fejezet konklúzió	75
5.	SCARA Assembly AI	82
5.1.	Releváns publikációk	83
5.2.	A robotkar térbeli munkaterülete	86
5.3.	A komponens-foglalat hozzárendelés leírhatósága	88
5.4.	A javasolt eljárás koncepciója	89
5.5.	A szintetikus adatgeneráló modulrendszer bemutatása	90
5.6.	A javasolt eljárás architektúrakomponenseinek bemutatása	93
5.6.1.	Detektor modul	93
5.6.2.	Szűrő modul	94

TARTALOMJEGYZÉK	iii
5.6.3. Hozzárendelő modul	96
5.7. Tanítási és validációs eredmények	99
5.7.1. A detektor modul	99
5.7.2. A szűrő modul	101
5.7.3. A hozzárendelő modul	101
5.8. Szimuláció Omniverse platformon	103
5.9. A valós roboton végzett kísérleti validáció	107
5.10. Konklúzió	108
Összefoglalás	110
Köszönetnyilvánítás	111
5.10.1. A PhD értekezés alapjául szolgáló közlemények	112
5.10.2. További közlemények	112
Irodalomjegyzék	117

Ábrák jegyzéke

2.1. Az Ipar 4.0 fő elemei és kapcsolatai.	6
2.2. Az Ipar 4.0 fő elemei és kapcsolatai.	10
2.3. Az Ipar 5.0 fő elemei és kapcsolatai.	14
2.4. A SCARA típusú beültető robotegység (SONY SRX-611).	15
2.5. A munkaterület és a SCARA robotegység robotkarjaihoz tartozó fizikai paraméterek mérettel.	15
2.6. A KUKA KR5 robot egység.	18
2.7. A KUKA KR5 robot egység munkakörnyezete.	18
2.8. A KUKA KR5 teherbírési diagramja.	19
2.9. A KUKA KR5 gyorsulási profiljának diagramja.	20
2.10. A KUKA KR5 csuklói és csatlakozásai.	21
2.11. A sajátfejlesztése FDH 3D nyomtató.	22
2.12. Az importált objektumok metszete.	23
2.13. A valós képek leírhatóságának modellje	26
3.1. A 3D CAD modellezés és renderelési folyamat.	44
3.2. A nyers renderelt képek tartományfordítása.	46
3.3. A pix2pixHD algoritmus kimenetei a szintetizált képsorozatok képzésének során: (a) mintavett kép a képzés kezdetén, (b) mintavett kép körülbelül 0,5-ös hasonlósági értéknél, (c) mintavett kép 0,6 feletti hasonlósági értéknél, (d) mintavett kép a képzés végén.	51
4.1. Az adatkészlet fordítási módszer architektúrája.	60
4.2. Képjelenetek tartományfordítása.	61
4.3. A módszertan keretdiagramja.	63

4.4. Munkadarabok a Flexlink XK és PARO QE szállítoszalagokon.	63
4.5. Szintetizált adatkészlet létrehozása.	64
4.6. 3D CAD-alapú képkalkotás.	65
4.7. A valós képek annotációi.	66
4.8. A pix2pixHD algoritmus tanítása során szintetizált képsorok kimenetei.	71
4.9. A megvalósított tesztkörnyezet.	73
4.10. A pix2pixHD algoritmus tanítása során szintetizált képhalmazok kimenetei.	74
5.1. A robot eszköz munkaterülete és a megfelelő alrégiók. . .	88
5.2. A javasolt módszerünk, az úgynevezett SCARA Assembly AI algoritmus architektúrája.	89
5.3. A megvalósított szintetikus összeszerelési adatgenerátor (SADG) modul.	90
5.4. A komponensszűrő neurális hálózat felépítése.	95
5.5. A hozzárendelő neurális hálózat felépítése.	97
5.6. A PCB panelek: (a) egyszerű felépítésű, (b) közepes bonyolultságú, (c) komplex felépítésű.	99
5.7. A járműgyártási laboratórium és kutatóközpont virtuális laboratóriumi helyszíne.	103
5.8. A SCARA Assembly AI algoritmus eredményei: (a,c,e) Az előre meghatározott slotokkal rendelkező üres PCB-k, (b,d,f) A PCB-k beültetett változata kijelölt alkatrészekkel.	106
5.9. A valós környezetben végzett validációhoz használt SCARA-alapú szerelőállomás kísérleti összeállítása. Az IP-kamera rögzíti a komponenselhelyezési képeket, és Wi-Fi-n keresztül továbbítja azokat a vezérlő- és adatfeldolgozó szkripteket futtató asztali számítógépre. A SONY SCARA SRX-611 a detektált komponens-foglalat párok alapján hajtja végre a pick-and-place műveleteket.	107

Táblázatok jegyzéke

2.1.	A SCARA robotkar kinematikai paramétereit.	16
2.2.	A SONY SRX-611 és modern SCARA robotok összehasonlítása	17
2.3.	A KUKA KR5 robotkar DH paramétereit.	22
2.4.	Az akrilnitril-butadién-sztirol (ABS) anyag tulajdonságait. .	24
2.5.	A Jetson Nano hardver specifikációit és feldolgozási teljesítménye.	25
3.1.	A kiszámított horgony-értékek minden egyes detektor esetében.	48
3.2.	A választott adatdúsító eljárások a hozzátartozó paraméter-értékekkel.	49
3.3.	A YOLO architektúrák detektorparamétereit és betanítási ideje.	50
3.4.	A képminőség eredményei a betanítás során szintetikus generált képek esetében.	53
3.5.	A detektor eredményei a szintetizált képek minőségétől függően.	54
4.1.	Újjonnan kiszámított horgony-értékek.	67
4.2.	Alkalmazott képjavító algoritmusok és azok paraméterbeállításai.	68
4.3.	A YOLOv3-alapú detektorok tanulási paramétereit.	69
4.4.	A YOLOv8-alapú detektorok tanulási paramétereit.	70
4.5.	A pix2pixHD algoritmus minőségi eredményei a szállítószalag-adatkészlet alapján.	76

4.6.	A Sony SCARA adatkészleten alapuló pix2pixHD algoritmus minőségi eredményei.	77
4.7.	A YOLOv3-alapú detektorok teljesítményének összehasonlítása a $CWSSIM_{obj}$ függvényében.	78
4.8.	Results of YOLOv8-Nano and YOLOv8-Small detectors (Sony SCARA dataset).	79
4.9.	Results of YOLOv8-Medium and YOLOv8-Large detectors (Sony SCARA dataset).	80
4.10.	A különböző tanítási adatkészletekkel kapcsolatos detektálási eredmények.	81
5.1.	A detektormodell képzési konfigurációja és paraméterei. . .	100
5.2.	A detektor modell képzési és validációs eredményei. . . .	101
5.3.	A szűrőháló konfigurációs paraméterei.	102
5.4.	A komponensszűrő modell tanítási és validálási eredményei.	102
5.5.	A feladatmodell képzési konfigurációja és paraméterei. . .	102
5.6.	A hozzárendelő modell tanítási és validálási eredményei. .	103
5.7.	A SCARA Assembly AI keretrendszer összesített kiértékelésének összefoglalója a detektálási, szűrési, hozzárendelési és valós környezeti validációs fázisokon keresztül.	108

1. fejezet

Bevezetés

Napjainkban az információtechnológiák megjelenésével és használatával az élet kulcsfontosságú területei jelentős fejlődésen mentek keresztül. Ennek egyik meghatározó kulstechnológiája a digitalizáció, amely számos előnyt nyújt azáltal, hogy átalakítja és modernizálja a gazdaság, az ipar és a vállalatok működésének alapelveit. Ezek közé sorolható a hatékonyság és az automatizálás növelése, mivel a technológia lehetővé teszi a folyamatok gyorsabb, pontosabb és költséghatékonyabb működését, és a mesterséges intelligencián alapuló megoldások integrációját is, amellyel új tanítható és intelligens üzleti modellek, gyártási és munkafolyamatok kialakítása válik elérhetővé. Megjelenik a vállalatok életében az adatvezérelt gondolkodás és az innováció képessége is, amelynek eredményeképpen növekedni fog a versenyképesség és a további fejlődés fenntartható lesz.

A digitalizáció megjelenésével a megfelelő mennyiségű és információtartalmú adatgyűjtés megléte és biztosítása szintén meghatározó tényezővé válik, és ez az ipari modernizáció folyamatában (Ipar 4.0 és Ipar 5.0) is lényeges szerephez fog jutni. Ahogy az élet más területein is megfigyelhető, a gyártási folyamatokban is megjelennek a digitális látáson és objektumazonosításon alapuló eljárások, prioritássá válik a robotizált termelés kidolgozása, az automatikus anyagmozgatás és logisztika kidolgozása és az intelligens szerelési rendszerek fejlesztése. Az adatmennyiség növekedése és változatos formában való előfordulása szintén magával hozza azokat a kulcsfontosságú technológiákat és fogalmakat mint például a nagymennyiségű adat (Big Data), szenzoradat-gyűjtés, a Tárgyak Internete (IoT)

alapú eszközök minél nagyobb fokú alkalmazása, és a felhőtechnológiákon alapuló feldolgozás (Edge. és fog-computing), amelyek lehetővé teszik a gyártás valós idejű felügyeletét és nyomon követését, a nagymennyiségű gyártási adatok elemzését és feldolgozását statisztikai alapokon és gépi tanuló modelleket alkalmazva.

A disszertáció a fentebb említett gyártási folyamatok digitalizációjához és modernizációjához kapcsolódó kulcsfontosságú kutatási irányokat tárgyalja, valamint újfajta módszertanokat, és alkalmazásokat fog bemutatni. A kutatás tématerülete pedig az ipari mélytanuláson és szintetikus adatgeneráláson alapuló megoldások és eljárásokhoz kapcsolódik.

A disszertáció felépítése a következő fő fejezetekből áll, amelyek az alábbi tématerületek fogják bemutatni.

A második fejezet foglalja magába azokat a fő módszertanokat és technológiákat, amelyekre épül az egész kutatási folyamat, és a kidolgozott mélytanuláson alapuló megoldások és alkalmazások. Áttekintésszerűen bemutatja az Ipar 4.0 és a koncepcionális Ipar 5.0 technológiai történetét és környezetét, a szintetikus adatgenerálást, részletesen ismerteti a képből kép alapú transzformáció alapjait (Image-to-Image translation), a mélytanulás jelenlegi alkalmazását ipari képfeldolgozásban, és végül az ezekhez kapcsolódó igen fontos mérési metrikákat is.

A harmadik fejezet ezt követően szintetikus adatgenerálással előállított képi adathalmazon tanított és mélytanuló neurális hálózatokon alapuló kidolgozott módszertant és architektúrát fog ismertetni. Az architektúra egy képből kép fordításon alapuló megoldás (pix2pixHD[1]) újfajta megközelítését fogja bemutatni ipari alkalmazási környezetben, amely lehetővé teszi a gépi látást és alakfelismerést régebbi roboteszközök számára, amelyek még a tervezésük idejében nem voltak ezen technológiákra felkészítve.

A negyedik fejezet szintén egy szintetikus adatgenerálási módszertanon alapuló eljárást fog ismertetni. Az ebben a fejezetben ismertett algoritmus és az alapját képező architektúra újfajta megközelítése a tervrajz-alapú tartománykonverzió lesz, amely a választott valós ipari környezetnek megfelelően képes lesz megtalálni és összepárosítani az összetartozó osztályrégió és anyagmintákat ezzel felépítve a valósághű képi adathalmazokat, amelyek a mélytanuló objektumdetektorok tanításához lesznek szükségesek. Az eljárás alkalmazásával kiküszöbölhetővé válik a hiányzó és időigényes

képi adatgyűjtés, továbbá a választott roboteszköz már előzetesen, virtuálisan felkészíthetővé válik a jövőbeli gyártósori feladatok ellátására. A fejezet szintén tartalmaz konkrét valós tesztelési eredményeket is beágyazott eszközökön implementálva.

Az ötödik fejezet végül egy gyártósori beültetési folyamatot támogató újfajta algoritmust (név szerint SCARA Assembly AI) mutat be, amelynek működése háromlépcsős architektúrán alapszik. A megközelítés újdonságát az úgynevezett önfigyelési mechanizmusokon alapuló mélytanuló eljárás adja, amely képes a felismert beültetendő alkatrészek célpozícióinak megtalálására sorrendtől függetlenül. Demonstrációként egy régebbi roboteszközön kerül bemutatásra a működése, amely egy virtuálisan összeállított környezetben lett elhelyezve. Virtuális platformként az NVIDIA által fejlesztett Omniverse [2] alapú szimulációs szoftveres környezet lett kiválasztva. A kidolgozott architektúra tesztelése valós ipari robotikai rendszeren szintén megtörtént.

2. fejezet

Az alkalmazott főbb módszertanok, technológiák és eszközök háttere

Ez a fejezet ismerteti azokat a főbb technológiákat, módszertani alapokat és eszközöket, amelyekre a disszertációban bemutatott eljárások épülnek, valamint összekapcsolja az ipari tématerületet a későbbi fejezetekben megjelenő saját fejlesztésekkel.

Először az Ipar 4.0 és az Ipar 5.0 kontextusát tárgyaljuk, mivel a vizsgált gyártási folyamatok digitalizációja adatvezérelt szemléletet, automatizálást és gépi látáson alapuló döntéstámogatást tesz szükségessé. Ezt követően bemutatjuk az alkalmazott robotikai és beágyazott hardveres környezetet, amely meghatározza az adatgyűjtés, a számítási erőforrások és a valós idejű feldolgozás gyakorlati korlátait. A valós képek leírhatóságának háttere azokat a környezeti és szenzorjelenségeket fogja rendszerezni, amelyek a szintetikus és valós adatok közötti eltérés egyik fő forrásai lesznek, és amelyek kezelésére a dolgozat célzott transzformációs eljárásokat fog alkalmazni. A képből-kép fordítás elméleti háttere és az adatkészlet-transzformáció és tartományadaptáció közvetlenül megalapozza a későbbi fejezetekben ismertetett, szintetikus adatokra épülő módszertanokat, amelyek a tartomány-különbség csökkentését és a címkézett adatok hatékony előállítását célozzák. Az alkalmazott mélytanuló architektúrák és módszerek áttekintése rögzíti azokat a modelleket és tervezési döntéseket (generatív

ellentétes hálózatok, pix2pixHD és YOLO-alapú megoldások), amelyekre a disszertáció kidolgozott eljárásai épülnek. Végül a kiértékelés során használt metrikák egységes értelmezési keretet adnak a bemutatott módszerek összehasonlításához és validálásához.

A fejezet így előkészíti a későbbi módszertani fejezeteket: a képtartomány-transzformációra épülő ipari megoldásokat, a kontrollálható szintetikus adatkészlet-előállítást és a robotos alkalmazásokban szükséges tanítható komponenseket, amelyek a 3–5. fejezetekben kerülnek részletezésre.

2.1. Az Ipar 4.0 és Ipar 5.0 technológiai bemutatása

Ahogy a bevezető fejezetben is említve lett, a mindennapi élet az elmúlt években jelentős fejlődéseken ment keresztül, és ez különösen igaz az ipari folyamatokra is. Ebben az esetben a témához kapcsolódóan két meghatározó fejlődési korszakot fontos megemlíteni, amelyek név szerint az Ipar 4.0 és a koncepcionális Ipar 5.0 lesznek.

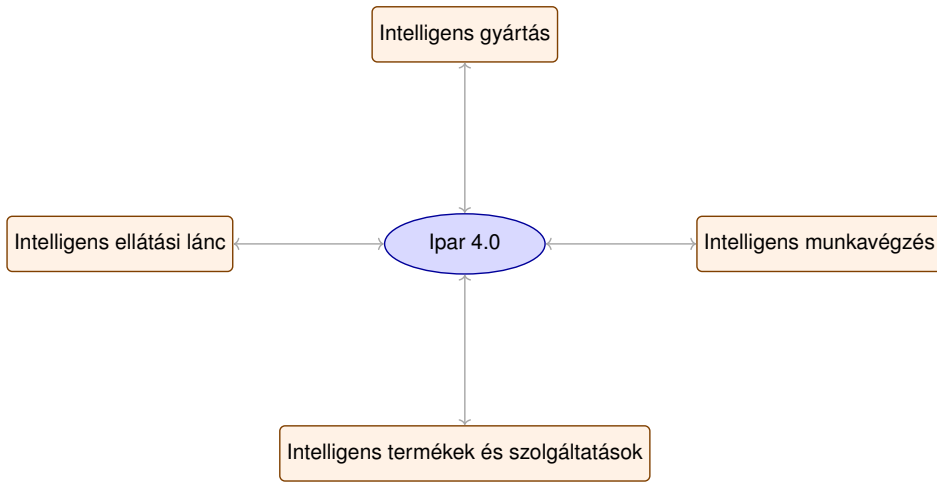
2.1.1. Az Ipar 4.0 bemutatása

Az elsőt szokás negyedik ipari forradalomként is említeni, és a fő célja az akkoriban egyre inkább fókuszba kerülő információtechnológiák és a digitalizáció integrációja volt az ipari és gyártási folyamatokban.

A témával kapcsolatban számos releváns publikáció megjelent az évek során. Caviggioli Federico és munkatársai egy elég részletesen átfogó publikációt készítettek, amely tartalmazza a legjelentősebb eseményeket és lépéseket, amelyek az ipari forradalom során megtörténtek [3].

A szerzők négy darab kulcsfontosságú dolgot azonosítottak, amelyek a következők voltak (lásd az 2.1 ábrán is):

- Intelligens gyártás
- Intelligens termékek és szolgáltatások
- Intelligens ellátási lánc
- Intelligens munkavégzés



2.1. ábra. Az Ipar 4.0 fő elemei és kapcsolatai.

Az Ipar 4.0-ról szóló szakirodalom főként a gyártási rendszerekben bekövetkezett változásokra összpontosított, amelyek ezeket a kiberfizikai rendszereket létrehozták [4]. Kipper és munkatársai átfogó áttekintést nyújtottak az Ipar 4.0-ról, kategorizálva a főbb támogató technológiákat, mint például a tárgyak internete, a nagy adathalmazok elemzése, a mesterséges intelligencia, a felhőalapú számítástechnika, az additív gyártás és a kiberfizikai rendszerek, ezzel megvitatta azok integrációját a különböző ipari területeken. A tanulmány továbbá azt is hangsúlyozza, hogy a sikeres digitális átalakuláshoz nemcsak technológiai átállás szükséges, hanem új üzleti modellek, adatkezelési keretrendszerek és a munkaerő továbbképzése is, hogy az intelligens gyártás teljes érettségét elérjék [5].

Az intelligens gyártás legelső koncepcióját Kagermann és munkatársai fogalmazták meg [4]. A formális definíciót pedig Kusiak alkotta meg kollegáival, amelynek a lényege hogy integrálja a mai és a jövő gyártási eszközeit érzékelőkkel, számítástechnikai platformokkal, kommunikációs technológiával, adatigényes modellezéssel, vezérléssel, szimulációval és prediktív mérnöki munkával [6].

Kusiak és Tao munkatársaikkal szintén megállapították, hogy az intelligens gyártás főként a kiberfizikai rendszerek, a Tárgyak Internete, a felhőalapú számítástechnika, a szolgáltatásorientált számítástechnika, a mesterséges intelligencia és az adattudomány koncepcióit alkalmazza [6, 7].

Bokrantz kollegáival megállapította azt is, hogy a fent említett tech-

nológiák az AI és a valós idejű adatok felhasználásával képesek támogatni a gyártástervezési és ellenőrzési folyamatot is annak érdekében, hogy a gyártási tevékenységet jobban megszervezzék [8].

Dalenogare és Wang munkatársaikkal leírták, hogy az Ipar 4.0 technológiái, mint például a gépek közötti kommunikáció, az információs rendszerek vertikális integrációja, a fejlett robotika – beleértve a kollaboratív robotikát is – olyan meghatározó részek, amelyek az intelligens gyártás dimenziójának részének tekinthetők [9, 10]. Továbbá megtalálható az is, hogy az intelligens gyártás az AI-alapú intelligens karbantartást is figyelembe veszi a potenciális meghibásodások és a berendezések leállításának előrejelzésére [8].

Az intelligens munkavégzés (SW) azt vizsgálja, hogy a technológiákat hogyan használják a munkavállalók támogatására a vállalat tevékenységében [3]. Elismeri, hogy a munkavállalók kritikus stratégiai szerepet játszanak a gyártási tevékenységekben, és hogy őket inkább támogatni kell, mint helyettesíteni [11].

Az utóbbi időben sok vita folyt az ember szerepéről az Ipar 4.0 kontextusában, és egyes tanulmányok egy új munkavállalói profilt javasoltak, az „Operátor 4.0” vagy „Intelligens Operátor” néven [12–14]. Bár több tanulmány is rámutat arra, hogy az autonóm gépek helyettesíthetik az operatív és alacsony hozzáadott értékű tevékenységeket, az Ipar 4.0 legjelentősebb potenciálja az, hogy támogatást nyújt a munkavállalóknak (az operátoroknak és más hierarchikus szinteknek) abban, hogy munkájukat okosabban végezzék. Ez a munka az emberi kognitív képességeken alapul, amelyek hozzáadott értéket jelentenek a termelési rendszer számára [15, 16]. Ezért az intelligens munkavégzés azt vizsgálja, hogyan lehet a legjobban kihasználni a munkavállalók potenciálját azáltal, hogy fejlett technológiákat alkalmaznak a döntéshozatali folyamatok támogatására [17, 18], a tudás kezelésére [11, 19], a kreativitás és a tervezés elősegítésére [16], valamint a munkavállalók biztonságának és elégedettségének növelésére [20]. Ebben az értelemben Frank és munkatársai [21] az „Intelligens munkavégzés” terminológiát alkalmazták, és figyelembe vették mind az intelligens operátorok által végzett operatív tevékenységeket, mind pedig a rugalmas és távoli tevékenységeket, amelyek szélesebb körű munkavállalókat érintenek, beleértve a vezetőket, mérnököket és felügyelőket, akik a gyártási folyamatok

kognitív tevékenységeit végzik.

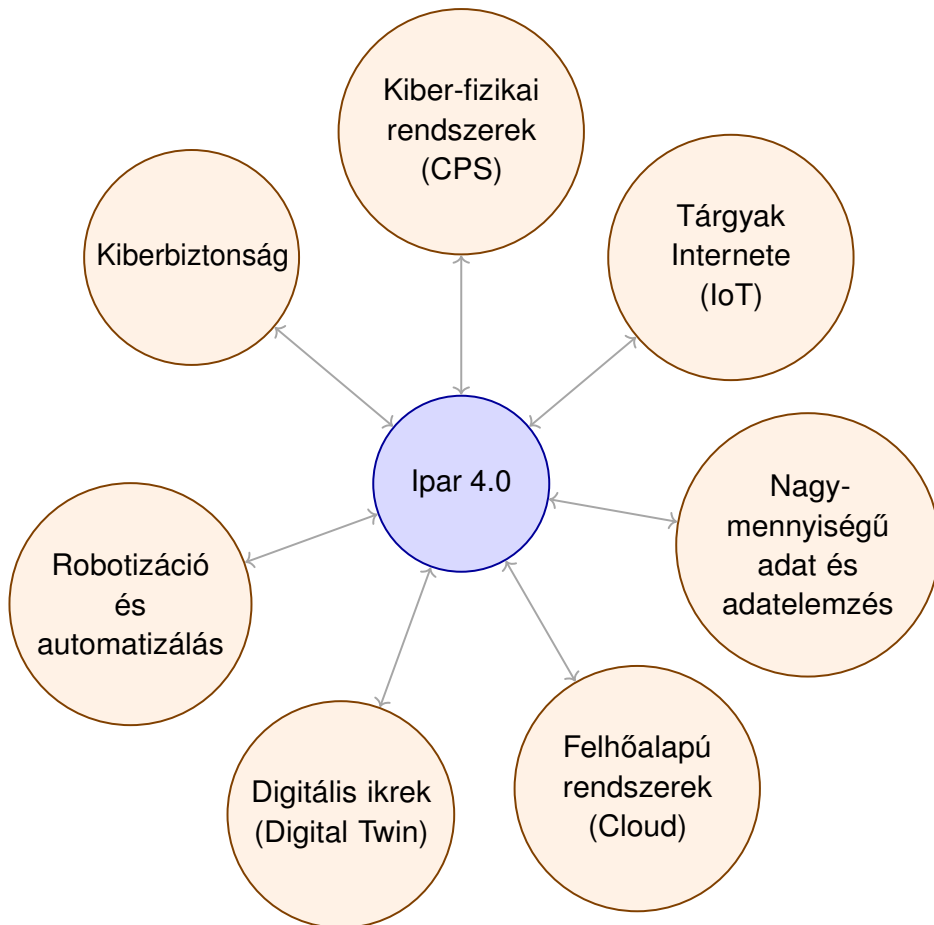
Az Ipar 4.0 harmadik meghatározó komponense az Intelligens ellátási lánc (SSC). Ez a koncepció egyesíti a korábbi definíciókat, mint például az ellátási lánc 4.0 [22], a digitális ellátási lánc [23] és a logisztika 4.0 [24]. Az intelligens ellátási lánc az Ipar 4.0 alaptermotechnológiáinak támogatását veszi figyelembe az ellátási lánc információáramlásának javítása érdekében [22]. Új lehetőségek nyílnak a kapcsolódás és a különböző ellátási láncban részt vevő érdekelt felek között valós időben megosztott adatok tömeges tárolása révén [25]. Az Ipar 4.0 olyan technológiai változásokat vezet be, amelyek hozzájárulnak az ellátási lánc átláthatóságának javításához, lehetővé téve az átfogó zavarok kockázatának kezelését az ellátási lánc végpontok közötti feltérképezésével [26]. Az integritás-ellenőrzésre alkalmazott technológiák (pl. érzékelők, nagy adathalmazok elemzése, decentralizált ügynökvezérelt ellenőrzés) biztosíthatják, hogy a megfelelő termékek a megfelelő időben, helyen, mennyiségben és állapotban, valamint a megfelelő áron kerüljenek forgalomba az ellátási lánc mentén [26, 27]. A fizikai logisztika szintjén az intelligens ellátási lánc dimenziója magában foglalja az autonóm robotok és járművek által végzett raktári kezelést, valamint a készletellenőrzéshez szükséges nyomonkövetési és döntéshozatali rendszereket is [24]. Ez magában foglalja a nyersanyagok (a gyártósor bemenete) és a gyártott termékek „intelligens” kezelését is a gyártócsarnokban. Az ilyen kezelést robotikus érzékelő technológiák, többek között automatikus vezérlésű járművek (AGV-k) és autonóm mobil robotok (AMR-ek) alkalmazásával lehet támogatni [28].

Az Intelligens termékek és szolgáltatások (SPS), az Ipar 4.0 keretrendszer negyedik meghatározó komponense. Két fő elkülöníthető részből áll, amelyek általában egy csomagban integrált megoldásban vannak beépítve. Az intelligens termékek olyan eszközök, amelyek fizikai alkatrészeiken kívül az Ipar 4.0 alaptermotechnológiáinak (IoT, felhő, nagy adatelemzés és mesterséges intelligencia) támogatásával gyűjtik, figyelik, ellenőrzik és optimalizálják a felhasználói adatokat [29, 30]. Az intelligens szolgáltatások pedig azokat a vállalatokat érintik, amelyek digitális technológiákat alkalmaznak annak érdekében, hogy szolgáltatásokat nyújtsanak felhasználóiknak, például felhőszolgáltatásokat, távoli segítségnyújtást és felügyeletet, valamint mesterséges intelligencián alapuló jelenléti ellenőrzést [31, 32]. Ezek a szol-

gáltatások önálló szolgáltatásként kínálhatók a termékek használatában való ügyfélszolgálat céljából, vagy maga a termék is kínálható szolgáltatásként egy használat alapú fizetési rendszerben [33].

A technológiai fejlődés szakaszhoz kapcsolódik számos főbb fogalom és meghatározás is, amelyek a gerincét alkották a mozgalomnak:

- Kiber-fizikai rendszerek (CPS)
- Tárgyak Internete (IoT)
- Nagymennyiségű adat és adatelemzés
- Felhőalapú rendszerek (Cloud Manufacturing)
- Digitális ikrek (Digital Twin)
- Robotizáció és automatizálás
- Kiberbiztonság



2.2. ábra. Az Ipar 4.0 fő elemei és kapcsolatai.

2.1.2. Az ipar 5.0 koncepcionális bemutatása

A második jelenlegi nagyobb mozgalmat nevezzük ötödik ipari forradalomnak, amely az ipar 4.0 alapjaira építkezve számos aspektusban szociotechnikai átalakulásnak is nevezhető, ugyanis míg az negyedik ipari forradalom elsősorban a digitális integrációra, az automatizálásra és a hatékonyságra összpontosított, a legújabb tanulmányok rávilágítottak egy új paradigma szükségességére, amelynek középpontjában az ember és a fenntarthatóság áll. Az új koncepció, az Ipar 5.0 ezeket a szempontokat veszi figyelembe, hangsúlyozva az ember és a gép közötti együttműködést, a társadalmi jólétet és az ökológiai felelősségvállalást [34]. A tanulmány szerint

az Ipar 5.0 az Ipar 4.0 technológiai alapjaira épül (kiberfizikai rendszerek, mesterséges intelligencia és a Tárgyak Internete), de céljaikat a rugalmasság, a fenntarthatóság és az emberközpontú értékteremtés felé irányítja. Ez a paradigmaváltás olyan munkahelyeket képzel el, ahol az intelligens automatizálás nem helyettesíti, hanem támogatja az emberi képességeket, elősegítve a kreativitást, a biztonságot és az alkalmazkodóképességet a gyártási rendszerekben. Következésképpen az Ipar 5.0 nem az Ipar 4.0 helyettesítését, hanem annak továbbfejlesztését jelenti, kiterjesztve hatókörét a technológiai optimalizálásraól a társadalmi és környezeti egyensúlyra.

Az ötödik ipari forradalommal kapcsolatban szintén számos releváns publikáció megjelent az utóbbi években. Ezek közé sorolható Orea-Giner és munkatársainak munkája is [35], amiben azt vizsgálták, hogy milyen kapcsolat van a vendégek érzelmei és a szállodai robotokkal való interakció által kiváltott érzései, valamint a szálloda értékelésére gyakorolt lehetséges hatása között. Véleményük szerint az Ipar 5.0 „a végfelhasználó élményének javítása a mesterséges intelligencia (AI) és a robotika területén rendelkezésre álló különböző eszközök alkalmazásával” [35]. Kaasinen és kollegái [36] a következő generációs gyártási rendszerek és az emberi operátorok tekintetében az intelligenciára és a rugalmasságra helyezik a hangsúlyt. Az 5.0-s ipar jövőbeli gyárai három alapvető elemét mutatják be: az emberközpontúságot, a fenntarthatóságot és a rugalmasságot [36], összhangban az európai vízióval [37]. A szerzők munkájukban egy olyan tervezési megközelítést mutatnak be, amelyben az emberi operátorok és az intelligens gépek együttműködő csapatokat alkotnak [36]. Egy másik kutatási irányzatban Popov és munkatársai [38] magas szintű kapcsolatot állapítanak meg az Ipar 5.0 és az egészségügy között, amelyet pályaként írnak le. Coronado és kollegái [39] az Ipar 5.0-t paradigmaváltásként mutatják be, amelynek célja a korábbi ipari forradalmakból eredő társadalmi és bolygónk előtt álló kihívások leküzdése az emberközpontú intelligens környezetek és az ember-robot interakció eszméjének népszerűsítésével. Ezenkívül, Kaasinen másokhoz hasonlóan, az Európai Unió Ipar 5.0 koncepciójára hivatkoznak, mint az emberközpontú, fenntartható és reziliens iparágak elérésének eszközére [36, 37, 39].

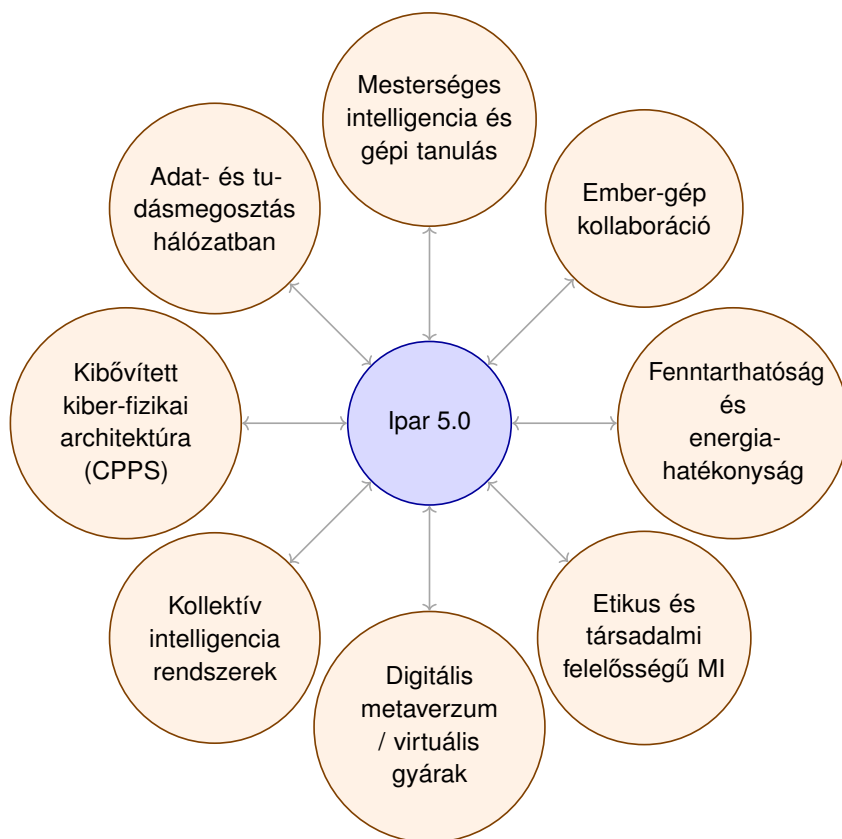
Ezenkívül az Ipar 5.0 koncepcióját értékvezérelt forgatókönyvként a gyártási szektorra összpontosítják, és az emberközpontúbb szempontokat

a „Társadalom 5.0” kifejezés alatt foglalják össze [39]. Többek között Carayannis és társai [40] kifejezetten megkülönböztetik az Ipar 5.0 és a Társadalom 5.0 fogalmát. Általánosan elfogadott véleményük szerint a Társadalom 5.0 egy emberközpontú „fejlett szolgáltatási platformokon alapuló, a jólétet szolgáló kölcsönös társadalmi infrastruktúra”, míg az Ipar 5.0 egy „megújult, emberközpontú ipari karaktertípus”, amely átalakítja az ipari termelési folyamatokat [40]. Ennek az átalakulásnak a lehetővé tévő technológiáiként a mesterséges intelligencia, a blokklánc és az IoT kerülnek bemutatásra [40]. Maier és kollegái [41] hasonlóan népszerűsítik a „Társadalom 5.0” kifejezést a termelés emberközpontúbb megközelítésére, ahelyett, hogy ezt az Industry 5.0 tágabb fogalmába ágyazzák. Duggal és munkatársai [42] két fő fázisból álló ütemtervet mutatnak be: szinergikus koprodukción és biofejlesztés. Az Ipar 5.0-t olyan koncepcióként határozzák meg, amelyben a robotok és az emberi munkaerő zökkenőmentesen együttműködik az emberi komponenssel, miközben hangsúlyozzák a személyre szabás és a testreszabás, különösen a személyre szabott biomérnöki munka fontosságát [42]. Shahbakhsh és munkatársai [43] az ember-robot együttműködést és a személyre szabást azonosítják az Ipar 5.0 változási tényezőiként. Maddikunta és kollegái [44] szerint az Ipar 5.0 olyan alkalmazásokhoz vezethet, mint az intelligens egészségügy, a felhőalapú gyártás vagy a tömeges testreszabás integrálása az ellátási lánc menedzsmentjébe [44], az emberek és a gépek közötti mélyebb együttműködés keretében. Fatima és munkatársai [45] áttekintést adnak az előnyök maximalizálásáról az emberi-gép interakció révén a gyártóüzemek és raktárak automatizálása tekintetében. Az Ipar 5.0-t az Ipar 4.0 továbbfejlesztésének tekintik, „amelyben az emberi kreativitás együttműködik az intelligens rendszerekkel, például robotokkal és gépekkel, különösen a gyártóüzemekben és raktárrendszerekben”, hogy „a gyártóüzemek és az iparágak gyorsabbá, hatékonyabbá és skálázhatóbbá váljanak” [45]. Paul és munkatársai [46] ugyanúgy az Ipar 5.0-t az Ipar 4.0 kiterjesztésének tekintik, amely nagyobb hangsúlyt fektet az emberközpontú számítógépes interakciókra. Pereira Guimarães és munkatársai [47] szintén az Ipar 5.0-t tekintik az Ipar 4.0 kiterjesztésének, és kiemeltebb figyelmet fordítanak a személyre szabottságra és az ember-számítógép interakcióra a gyártási folyamatok hőmérsékletmérésének csökkentésével kapcsolatos kritikai áttekintésükben. Különösen a technológiai biztonság

és védelem, az ember-robot interakcióval és a mesterséges intelligenciával kapcsolatos jogi kérdések [48, 49], valamint az ember alkalmazkodása az Ipar 5.0-hoz [43, 50] jelentik a legnagyobb kihívásokat. Az utóbbi témát két cikk tárgyalja, amelyek az oktatásra és a munkavállalóknak az 5.0-s iparág fejlődő környezetéhez való alkalmazkodásához szükséges készségekre és kompetenciákra [51], valamint az egyetemi környezet változásaira [40] összpontosítanak. A gyártás technológiai fejlődése folyamatos, és több szerző is tanulmányozta az Ipar 5.0 és az Ipar 4.0 párhuzamos fejlődését. Például részletesen bemutatták az Ipar 4.0 és 5.0 közös technológiai elemeit [52], vagy tanulmányaikban egy fejezetet szenteltek a személyre szabottabb gyártás kilátásainak, amelyben az embereket és a gépeket humanizáltabb környezetben kombinálják [53]. Egyéb példák közé tartozik az Ergonómia 4.0 és 5.0 [46], amely kiber-fizikai rendszereket alkalmaz, beleértve azok struktúráit, technológiáit, vagy az Operátor 4.0, amely mélyebb együttműködést valósít meg a robotokkal. Az Ipar 4.0 által lehetővé tett digitális ikrek, emberi modellezési módszerek és valós idejű monitorozás fontos eszközök az emberek védelme és a teljesítmény biztosítása érdekében [46] az emberközpontú Ipar 5.0 felé vezető úton. Más Ipar 5.0-ra vonatkozó áttekintések rámutatnak a személyre szabás fontosságára és a technológia szerepére a társadalmi szükségletek kielégítésében (pl. zöld lakások [54]), a Társadalom 5.0-val való szinergiákra [55], a robotok és a digitális ikrek megismerésére [56], valamint a társadalmi fejlesztési célokat és a fenntartható mérnöki munkát elősegítő technológiák használatára [57]. Mind a „4.0”, mind az „5.0” koncepció releváns, és szinergiáik feltárása hozzájárulhat a folyamatban lévő ipari átalakulásokkal kapcsolatos tudás gyarapodásához.

Összesítve a fenti publikációk alapján az ötödik ipari forradalom meghatározó alapfogalmai az alábbiak (lásd az 2.3 ábrán is):

- Mesterséges intelligencia és gépi tanulás
- Ember-gép kollaboráció
- Fenntarthatóság és energiahatékonyság
- Etikus és társadalmi felelősségű MI
- Digitális metaverzum / virtuális gyárak
- Kollektív intelligencia rendszerek
- Kibővített kiber-fizikai architektúra (CPPS)
- Adat- és tudásmegosztás hálózatban



2.3. ábra. Az Ipar 5.0 fő elemei és kapcsolatai.

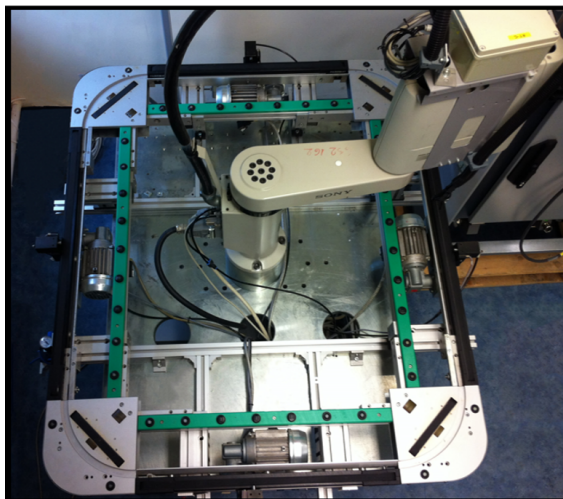
2.2. Ipari robotok és hardveres eszközök bemutatása

Ebben a részben kerülnek bemutatásra a kutatás során alkalmazott hardveres eszközök és régebbi ipari robotok, amelyek segítségével a kidolgozott módszertanok és eljárások tesztelve lettek.

2.2.1. SONY SCARA – SRX-611 robotegység

Az ebben részben ismertett gép maga egy régebbi SONY SCARA típusú roboteszköz, amelyet főként beültetési feladatokra használtak az iparban, és korábban széles körben elterjedtek számított [19]. Általánosságban

elmondható, hogy az SRX-611 típusú robotnak számos korlátozása van, amelynek fő oka, hogy 1978-as modellről van szó (lásd a 2.2 táblázatot)[20].



2.4. ábra. A SCARA típusú beültető robotegység (SONY SRX-611).

Ennek oka, hogy az akkori ipari szabványok teljesen mások voltak, ezért a robotvezérlő nem képes jelenleg hálózaton keresztül kommunikálni és nem rendelkezik USB (Universal Serial Bus) csatlakozási felülettel sem. A gép kialakítása RRT típusú, a kar súlya 35 kg. Felvétel és elhelyezési feladatokhoz ideális 2 kg-os terhelhetőséggel rendelkezik (lásd a 2.5 ábrán és az 2.1 táblázatban). Maga a robotegység kétujjas elektropneumatikus fogóval van felszerelve, amelynek működéséhez 4 bar üzemi nyomás szükséges.



2.5. ábra. A munkaterület és a SCARA robotegység robotkarjaihoz tartozó fizikai paraméterek mérettel.

A robot tartalmaz egy PARO QE 01 31-6000 szállítószalag-rendszert is, amely a robotkar körül helyezkedik el. Ez fogja biztosítja az alkatrészek megfelelő áramlását a válogatás vagy elhelyezés során [21].

A robotegység korlátai miatt, amint azt az előbb részleteztük, az SRX-611 alkalmazhatósága jelentősen lecsökkent, ami azt jelenti, hogy jelentős korlátozások mellett integrálható csak a gyártósorokba, mivel nem felel meg az Ipar 4.0 szabványoknak, amelyekre az Ipar 5.0 szintén nagymértékben támaszkodik.

2.1. táblázat. A SCARA robotkar kinematikai paraméterei.

Kar hossza	Mukaterület	Maximális sebesség	Ismétlési pontosság
1. tengely 350 mm	1. tengely 220°	-	-
2. tengely 250 mm	2. tengely $\pm 150^\circ$	-	-
-	Z tengely 150 mm	Z tengely 770 mm/s	Z tengely ± 0.02 mm
-	R tengely $\pm 360^\circ$	R tengely 1150°/s	R tengely ± 0.03 mm

2.2. táblázat. A SONY SRX-611 és modern SCARA robotok összehasonlítása

Tulajdonság	SONY SRX-611	Modern SCARA robot (pl. Epson GX, Omron TM, Dobot CR)
Gyártási év / generáció	1990-es évek vége	2020-as évek, Ipar 4.0/5.0 kompatibilis
Vezérlőegység	Zárt, dedikált MCU, RS-232 kommunikáció	Nyílt vezérlőarchitektúra (Ethernet/IP, OPC-UA, MQTT, ROS támogatás)
Számítási teljesítmény	Nincs GPU/AI gyorsítás, fix PID-vezérlés	Integrált CPU/GPU modul, neurális inferencia támogatás
Kamera- és szenzorintegráció	Külső eszközökkel, manuálisan illeszthető	Beépített vagy szabványos szenzor interfészek (látás, erő, gyorsulás, IMU)
Pontosság / ismétlési hiba	± 0.05 mm	± 0.01 mm vagy jobb, automatikus kalibrációval
Mozgásprofil / pályatervezés	Fix, lineáris interpoláció	Spline-, rángás- és energia-optimalizált mozgásprofilok
Kollaboratív üzemmód	Nem támogatott	ISO 10218 / TS 15066 kompatibilis biztonsági korlátokkal
Hálózati és felhőkapcsolat	Hiányzik	Felhő- és digitális ikerpár integráció, peremhálózati MI-alapú feldolgozás
Kiberbiztonság	Nincs titkosított kommunikáció	TLS, szerepkör-alapú hozzáférés, biztonságos firmware

2.2.2. KUKA KR5 robotegység és FLEXLINK XK

A másik bemutatandó roboteszköz egy soros kinematikájú KUKA KR5 típus lesz. A robot hat szabadságfokkal rendelkezik, ami viszonylag nagy, $8,4 \text{ m}^3$ munkaterületet tesz lehetővé. Az iparban a gépet főként hegesztési feladatokra használják, mivel nagy pontosságú, $\pm 0,04$ mm-es ismételhetséggel rendelkezik. Anyagmozgatási feladatokra csak korlátozottan alkalmas, mivel teherbírása 5 kg-ra korlátozott.



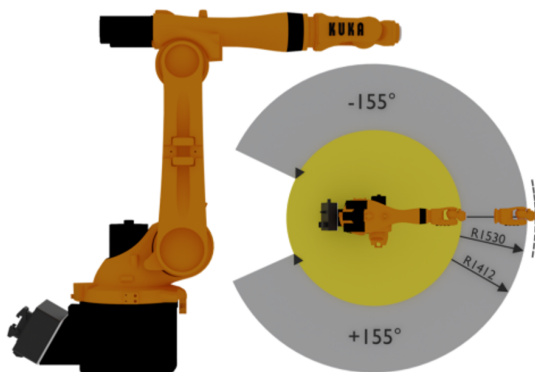
2.6. ábra. A KUKA KR5 robot egység.

A KUKA KR5 körüli robotcella (lásd 2.6 ábra) három platformmal van felszerelve anyagmozgatási feladatokhoz. A robotcella szabványos alumínium profilokból készül. A gép munkacellájának fő egysége a szállítószalag, amely biztosítja a különböző tesztelendő tárgyak pozicionálását.

Bonyolult műveleteket is képes végrehajtani, a mozgástartományát azonban jelentősen befolyásolják az egyes tengelyek korlátai:

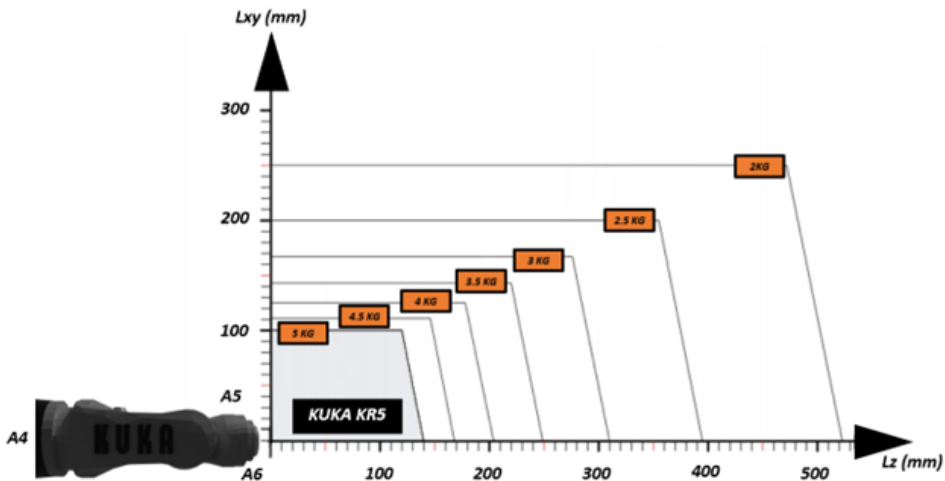
- J1 tengely: $\pm 155^\circ$
- J2 tengely: $+65^\circ / -180^\circ$
- J3 tengely: $+158^\circ / -15^\circ$
- J4 tengely: $\pm 350^\circ$
- J5 tengely: $\pm 130^\circ$
- J6 tengely: $\pm 350^\circ$ [58]

A KUKA KR5 robotkar munkaterületét szemlélteti a 2.7 ábra.



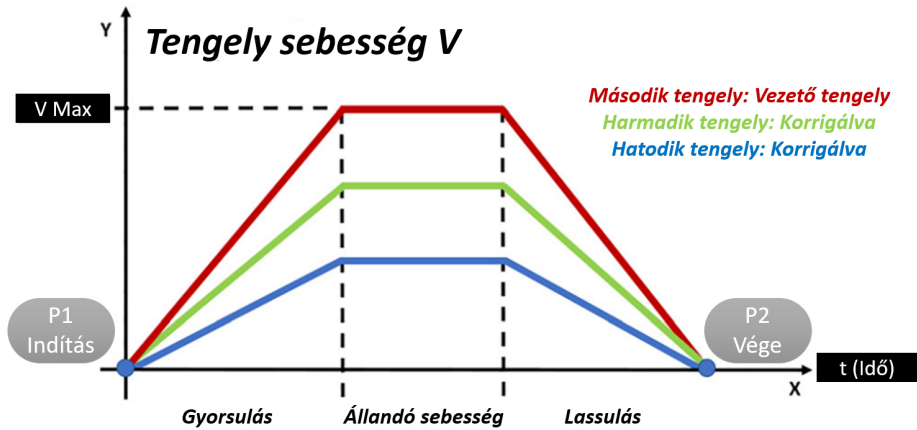
2.7. ábra. A KUKA KR5 robot egység munkakörnyezete.

A KUKA KR5 teherbírása befolyásolja annak dinamikáját, ami viszont hatással van a robot mozgási teljesítményére (lásd 2.8 ábra). A 4. ábra a maximálisan megengedett terhelést mutatja. Általában 0 és 100 mm között 5 kg terhelés megengedett, de 500 mm felett csak 2 kg.



2.8. ábra. A KUKA KR5 teherbírési diagramja.

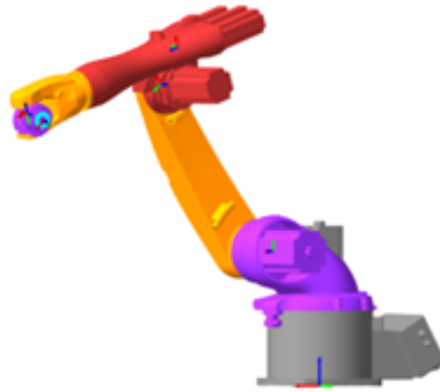
A 2.8 ábra a maximálisan megengedett terhelést mutatja. Általában 0 és 100 mm között 5 kg terhelés megengedett, de 500 mm felett csak 2 kg. A robot dinamikus paraméterei is változhatnak a terhelés függvényében, amit egy adott mozdulat végrehajtásakor figyelembe kell venni. Esetünkben a névleges teherbírás az a maximálisan megengedett tömeg, amelyet a gépezet normál körülmények között még képes kezelni. Az értékek azt mutatják, hogy az alsó tengelyek jelentősen lassabbak, ami a robot kialakításának köszönhető, mivel ezeken a tengelyeken a motoroknak nagyobb tehetetlenséggel kell megbirkózniuk. Ez fordítva is igaz a robot felső csatlakozásaira, amelyek a kisebb terhelésnek köszönhetően pontosabban és gyorsabban tudnak mozogni.



2.9. ábra. A KUKA KR5 gyorsulási profiljának diagramja.

A 2.9 ábra a KUKA KR5 robotkar tengelyeinek sebességprofilját mutatja az idő függvényében. A P_1 pont a kiindulási pozíciót, a P_2 pont pedig a robot munkaterületén belüli célpozíciót jelöli. A robot mozgásának dinamikája három fázisra bontható. Az első a gyorsulási fázis, amelyben a robot sebessége fokozatosan növekszik, amíg el nem éri a megengedett maximális sebességet. A grafikonon látható meredekség a gyorsulás mértékét jellemzi. Az állandó sebességű fázis azt jelenti, hogy a gépezet elérte a megengedett sebességet, azaz a gyorsulás mértéke nulla. A fékezési fázisban a gép sebessége fokozatosan csökken, amíg el nem éri a nullát. A grafikon azt mutatja, hogy az J_2 tengely a vezető tengely, amely a leggyorsabban mozog. Az J_3 és J_6 tengelyek a vezető tengelyhez viszonyított mozgásuknak megfelelően gyorsulnak vagy lassulnak.

A KUKA KR5 robotkar kinematikai leírásához az úgynevezett Denavit-Hartenberg-módszert (DH) alkalmaztuk. A módszer szabványosított módon írja le a csuklók és ízületek térbeli elhelyezkedését (lásd az 2.10 ábrán és a 2.3 táblázatban).



2.10. ábra. A KUKA KR5 csuklói és csatlakozásai.

A KUKA KR5 DH paraméterei a következők:

- θ : Forgatás a z-tengely körül, hogy párhuzamos legyen az x-tengellyel.
- d : A z-tengely mentén eltolva az x-tengelyek közös normáljának metszéspontjáig.
- ξ : Az x-tengely mentén eltolás a z-tengelyek közös normáljának metszéspontjáig.
- γ : Forgatás az x-tengely körül, hogy párhuzamos legyen a z-tengelyekkel [59].

A szükséges Denavit-Hartenberg paraméterek [59] meghatározásához a RoboAnalyzer nevű ingyenes robotszimulációs szoftvert használtuk, amely 3D grafikával segít meghatározni a KUKA KR5 robotkar csatlakozásainak paramétereit. Fontos megjegyezni, hogy a programot elsősorban régebbi Windows operációs rendszerekhez tervezték (.Net 2.0 Framework). Esetünkben egy virtuális operációs rendszeren futtattuk, így a rendszerkövetelmények teljesültek.

A projekthez a robotikai laboratórium ipari szerelősorát használtuk. A megadott tárgyak a robot munkaterületére telepített Flexlink szállítószalagon mozognak. A robotkar körül alumínium profilokból építettünk egy Flexlink szállítószalaggal ellátott robotcellát. A szállítószalag 10 kg súlyú terhet képes szállítani, és egy NEMA23 ST5918L4508-B léptetőmotor hajtja. A FlexLink hossza 5200 mm, szélessége 45 mm. A KUKA KR5 karegység Gmbh markolóval rendelkezik, hogy szükség esetén tárgyakat is szállíthasson.

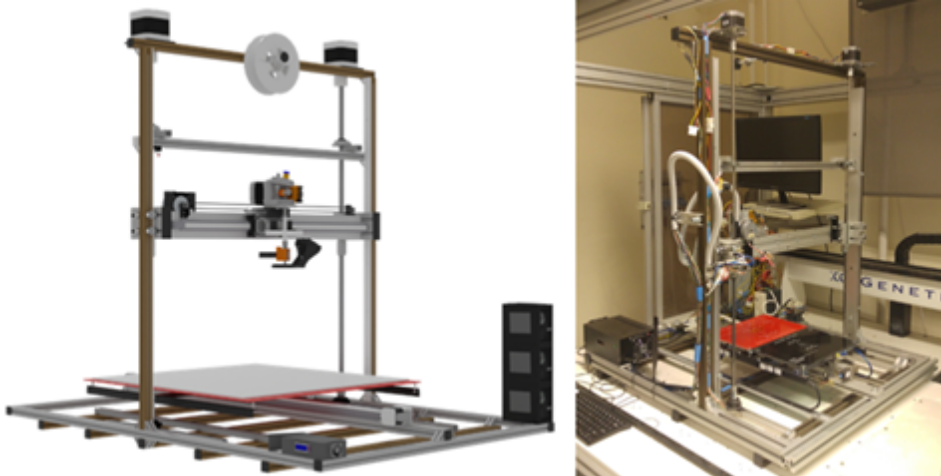
2.3. táblázat. A KUKA KR5 robotkar DH paraméterei.

Forgáspont	Tengely típus	Forgáspont szöge	Forgáspont eltolása	Kar hossza	Iker szög [rad]	Alapértelmezett (JV) [deg]
J1	Forgó	Változtatható	0,5	0,17	$\pi/2$	10
J2	Forgó	Változtatható	0,136	0,5	π	56
J3	Forgó	Változtatható	0,136	0,16	$-\pi/2$	-16
J4	Forgó	Változtatható	0,63	0,1	$\pi/2$	85
J5	Forgó	Változtatható	0,12	0,1	$-\pi/2$	75
J6	Forgó	Változtatható	0,115	0,1	0	42

2.2.3. Sajátfejlesztésű FDM 3D nyomtató és teszt tárgyak nyomtatása

A Járműgyártási laboratóriumban a közelmúltban számos kutatási és fejlesztési projekt zajlott. Ezért a testreszabott alkatrészek gyártása érdekében egy speciális FDM-alapú 3D-nyomtató került tervezésre és kialakításra (lásd az 2.11 ábrát).

Az FDP típusú 3D nyomtatókat gyakran használják olyan prototípusok készítésére, amelyek esetében a nyomtatandó elemek száma alacsony [60].



2.11. ábra. A sajátfejlesztése FDH 3D nyomtató.

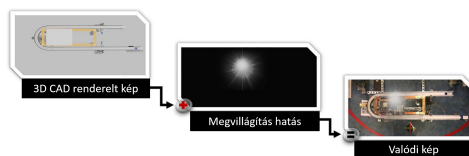
A 3D nyomtató TTT típusú, ami azt jelenti, hogy bármely adott pillanatban meghatározza a munka hatókörét is. Az x-tengelyen található az

adagolóegység, amelynek fő funkciója, hogy egy segédmotor segítségével behúzza a filamentet, hogy az megolvadjon a forró végben, amely elvégzi az olvasztást. A 3D nyomtatók esetében a fúvóka az egyik meghatározó tényező, amely nagyban befolyásolja a végtermék minőségét. Esetünkben a fúvóka átmérője 0,2 mm. Mivel az FDM nyomtatók esetében kulcsfontosságú tényező a hőfelbontás állandóságának fenntartása, erre a célra egy keret került kialakításra.

A házon belül kifejlesztett FDM 3D nyomtató nyomtatási tartománya a következő:

- Magasság: 32 cm
- Szélesség: 15 cm
- Hosszúság: 20 cm

3D CAD modellező program segítségével létrehoztunk egy 25×25 mm-es kocka alakú tesztmintát. A 3D modellt STL fájlként exportáltuk a későbbi utófeldolgozó szoftverhez. Esetünkben ez szoftver az Ultimaker Cura [61] volt. Az importált objektumokat az alábbi 2.12 ábra mutatja.



2.12. ábra. Az importált objektumok metszete.

A robotcellához használt elemek 3D-s mintamodellek voltak. Mivel a nyomtatandó alkatrészek ipari felhasználásra készültek, a hőre lágyuló műanyag szál típusát ennek megfelelően lettek kiválasztva. Esetünkben az anyag akrilnitril-butadién-sztirol (ABS+) volt, egy nagy keménységű és szilárdságú műanyag a kontraszt biztosítása érdekében fekete és fehér színekben. A nyomtatáshoz használt adagoló hőmérséklete $245\text{ }^{\circ}\text{C}$ volt, a fűtött ágyé $90\text{ }^{\circ}\text{C}$.

A filamentumra jellemző paramétereket a 2.4 táblázat tartalmazza [62].

2.4. táblázat. Az akrilnitril-butadién-sztirol (ABS) anyag tulajdonságai.

Anyag jellemző	Akrilnitril-butadién-sztirol (ABS)
Sűrűség ρ (Mg/m ³)	1.00–1.22
Rugalmassági modulus E (GPa)	1.12–2.87
Szakadási nyúlás (%)	3–75
Olvadáspont (°C)	88–128
Üvegesedési hőmérséklet (°C)	100
Szakítószilárdság (MPa)	33–110

2.2.4. NVIDIA Jetson Nano development kit

A harmadik bemutatandó eszköz egy beágyazott fejlesztő kártya lesz, nevezetesen egy NVIDIA Jetson Nano típus. A Jetson Nano 128 magos Maxwell GPU-val, négymagos ARM Cortex-A57 CPU-val és 4 GB LPDDR4 memóriával rendelkezik, ami alkalmassá teszi alacsony energiaigényű perifériás következtetéses feladatokra.

Az eszközzel kapcsolatos számos releváns publikáció jelent meg az elmúlt években, ezekre alapulva esett a választásunk erre a típusra. Choe Chungjae és munkatársai több 3D-objektum érzékelőt teszteltek Jetson-eszközökön, beleértve a Nano-t is, és elemezték a feldolgozás késleltetését, a memóriahasználatot és a pontosság kompromisszumait [63]. Másik munkában Lee JunKyu és munkatársai optimalizálták a YOLOv8 architektúrát a Jetson Nano-n a TensorRT és a vegyes pontosság használatával, valós idejű teljesítményt elérve a beágyazott végponti látás feladatokhoz [64]. A tanulmány a Jetson Nano-n történő könnyű YOLO-alapú detektorok telepítését vizsgálja, különös tekintettel az átviteli sebességre, a hőstabilitásra és a modellméret optimalizálására [65]. Ez a kutatás bemutatja a Jetson Nano hardverhez optimalizált dinamikus, precíziós adaptív objektumfelismerő keretrendszert, amely valós idejű perifériás alkalmazásokban egyensúlyt teremt a következtetés sebessége és a felismerés pontossága között [66].

A kísérleteket a PyTorch 2.2 keretrendszerrel, CUDA 11.8 és cuDNN 8.9 támogatással, Ubuntu 20.04 LTS rendszeren futtatva végeztük. Az optimális futási idő érdekében a betanított PyTorch modelleket ONNX formátumba

exportáltuk, és FP16 pontossággal TensorRT kompatibilissé konvertáltuk.

2.5. táblázat. A Jetson Nano hardver specifikációi és feldolgozási teljesítménye.

Paraméter	Specifikáció / Mérés
GPU	128-core NVIDIA Maxwell
CPU	Quad-core ARM Cortex-A57 @ 1.43 GHz
Memória	4 GB LPDDR4 @ 25.6 GB/s
Háttértár	microSD (Class 10 / UHS-I)
Operációs rendszer	Ubuntu 20.04 LTS (JetPack 5.1)
Keretrendszer	PyTorch 2.2, CUDA 11.8, cuDNN 8.9, TensorRT 8.6
Teljesítmény mód	10 W (MAXN)
Bemeneti felbontás	416×416 px – 640×640 px

2.3. A valós képek leírhatóságának háttere

Minden olyan esetben, amikor valós képfelvételeket vizsgálunk, számos egyedi jelenség, például tükröződések, színváltozások és különböző fényjelenségek figyelhetők meg a képeken, amelyeket specifikus környezeti hatások okoznak, ezek közé sorolhatóak az objektum és környezetének sajátos jellemzői, és a képfeltevő eszköz lencserendszer-felépítése is. Az utóbbi esetben, amikor a készülék gyártója nem hozza nyilvánosságra a fizikai tulajdonságokat és paramétereiket, rendkívül nehéz elméleti modellt készíteni minden egyes egyedi rendszerhez [67, 68].

A megvilágítás szempontjából számos gyártási folyamatban lehet optimális körülményeket beállítani [69], de ahogy Martinez és munkatársai vizuális érzékelőkkel támogatott automatizált acélgyártási folyamatról szóló munkájukban rámutattak, a gyártandó alkatrészekre korlátozások vonatkozhatnak [70]. A fenti okok miatt először is meg kell határozni a valós képen felfedezett jelenségeket, hogy a felmerülő problémákra robusztus megoldást lehessen találni és javasolni.

Egy valós kép a fényforrás által megvilágított komponensek, az adott környezetre jellemző fényjelenségek és véletlenszerű Gauss zaj összegeként írható le[71] (lásd a 2.13. ábrát):

$$I_{\text{real}} = I_{\text{normal}} + I_{\text{phenomena}} + N(0, \sigma_2). \quad (2.1)$$

A 2.1 egyenletben I_{normal} jelöli a jelenségmentes (alap) képi intenzitásösszetevőt (fényességi komponens), $I_{\text{phenomena}}$ a környezetre jellemző fényjelenségeket leíró intenzitásösszetevőt, míg $N(0, \sigma_2)$ egy additív, zérus várható értékű Gauss-zajtag, amely a két determinisztikus összetevő finomításaként (perturbációjaként) értelmezhető. Ennek megfelelően a zajtag nem önálló „képtartalom”, hanem a végső intenzitásérték kismértékű módosítása, ezért a mintavételezett értékei előjelet is válhatnak (negatívak is lehetnek). A zaj varianciája egy skálázott khí-négyzet eloszlásból kerül mintavételezésre ($\sigma_2 \sim 0.01\chi^2$).



2.13. ábra. A valós képek leírhatóságának modellje

A tükröződések és a fények a két alapvető kategória, amelyek az egyes jelenségeket tartalmazó komponenseket alkotják. A szélsőséges fényviszonyok tükröződések okoznak a kamerarendszer lencséinek felületén. Formájukat és gyakoriságukat elsősorban a lencserendszer kialakítása határozza meg. Típusuk alapján két csoportba sorolhatók: szétszórt és visszatükröződő. Ez a jelenség sokkal ritkábban fordul elő a gyártási folyamatokban, ahol a megvilágítás szigorúan szabályozott. Mindazonáltal fontos szem előtt tartani, hogy amikor előfordul, súlyosan kiegészített pixelterületeket eredményezhet, ami jelentősen befolyásolhatja a gépi tanulási algoritmusok teljesítményét és súlyosbíthatja az észlelési pontosság vesztését [72]. A második fő kategóriát a visszaverődések jelentik, amelyek

az objektum gyártási specifikációitól függően, például a felület alakjától és anyagától, simaságától, más fényvisszaverő fóliák, címkék vagy jelölések jelenlététől stb. függően jelentősen gyakrabban fordulhatnak elő.

A tárgyak felületének teljes visszaverődése a tárgyakról visszaverődő fényjelenségek és a megvilágítás spektrális energiaeloszlásainak összegeként definiálhatók, így a következőképpen adhatók meg [73]:

$$\kappa(x) = \kappa_s(x) + \kappa_d(x). \quad (2.2)$$

2.4. Képből kép alapú fordítási technika háttere

A képből kép alapú fordítás technika olyan számítógépes látásproblémák osztályát jelenti, amelyek célja két vizuális tartomány közötti leképezés megtanulása — például szintetikus renderelések valódi fényképekké, szegmentációs térképek RGB — képekké vagy nappali jelenetek éjszakai jelenetekké történő fordítása.

A képből-kép fordítás (képtartomány-transzformáció) feladata két képhalmaz (képtartomány) közötti leképezés tanulása. Legyen A a forrásképek halmaza és B a célképek halmaza, ahol egy-egy kép például fotorealistikus RGB képként $\mathbb{R}^{H \times W \times 3}$ alakban reprezentálható. Ekkor a tanulandó transzformációt

$$G : A \rightarrow B \quad (2.3)$$

alakban adjuk meg, ahol $a \in A$ egy tetszőleges forrásképp, és $G(a) \in B$ a hozzá tartozó transzformált célkép (azaz $b = G(a)$, ahol $b \in B$).

Ennek a megközelítésnek a motivációja az adatok átalakításában és a tartomány adaptációban rejlik. Sok gyakorlati helyzetben — például az ipari robotikában vagy az autonóm járművekben — nagy mennyiségű annotált valós kép beszerzése drága vagy kivitelezhetetlen. Ennek a leképezésnek a megtanulásával valóság-hű, címkézett mintákat lehet generálni a kép céltartományában, ezzel áthidalva a szintetikus és a valós adatok közötti nehezen leírható és jelentős eltérést.

A korai képfordítási technikák determinisztikus leképezéseken alapultak, ezek közé sorolható a színátvitelen, a hisztogram-illesztésen vagy az automatikus kódoló modelleken alapú megoldások [74–78]. Ezek a módszerek azonban nem tudták kinyerni a valós képekben jelen lévő komp-

lex, magas dimenziós kapcsolatokat (textúrák, megvilágítás, geometria és stílusváltozatok).

A következő nagyobb mérföldkő a generatív ellentétes hálózatok megjelenése volt, amellyel a fent említett problémák megoldhatóvá váltak. Gui és munkatársai munkájukban a generatív ellentétes hálózatok (GAN) egyik legátfogóbb áttekintését nyújtják, amely mind az elméleti alapokat, mind a gyakorlati megvalósításokat lefedi [79]. A publikáció a GAN-kutatásokat négy fő perspektívába sorolta be:

1. Algoritmus generációk megjelenése, ezek közé sorolhatóak a feltételes GAN-ok, Wasserstein GAN-ok és energiaalapú változatok.
2. Elméleti újdonságok, ide tartozik a konvergencia viselkedésének, az optimalizálás dinamikájának elemzése.
3. Alkalmazási területek, ide tartozik a képszintézis, stílusátvitel, szuperfelbontás generálása, videógenerálás és adatterjesztés.
4. Alkalmazható metrikák mint például a mennyiségi (FID, IS), és a minőségi értékelési megközelítések megvitatása.

A szerzők emellett olyan tartós kihívásokat is azonosítanak, mint a tanítás instabilitása és az egységes értékelési szabványok hiánya, miközben továbbfejlesztési irányokat javasolnak a kontrollálható és értelmezhető generatív modellek kutatásához. Ez a tanulmány széles körben alapvető referenciaként tekinthető, amely összefoglalja a GAN-kutatások 2020-ig elért eredményeit, és elméleti alapként szolgál a későbbi fejlesztésekhez.

Aggarwal és munkatársai munkájukban tömör és alkalmazásorientált áttekintést adnak a GAN keretrendszeréről, hangsúlyozva annak koncepcionális felépítését és ipari relevanciáját [80]. A tanulmány a generátor és a diszkriminátor közötti ellentétes képzési folyamat intuitív magyarázatával kezdődik, majd összefoglalja a legfontosabb architektúrális kiterjesztéseket, mint például a CycleGAN [81], a StyleGAN [82] és a pix2pix [83]. Kiemeli a GAN-ok sokoldalúságát több területen is mint például az orvosi képalkotástól és az egészségügyi diagnosztikától az adatszintézisig, a képhelyreállításig és a kreatív tervezésig.

A mélyebb elméleti áttekintésekkel ellentétben ez a cikk a gyakorlati alkalmazásokra összpontosít, bemutatva, hogy a GAN-ok hogyan integrálhatók a valós világ rendszereibe. A szerzők etikai és szabályozási szempontokat is megvitatnak, rámutatva a generatív modellek kettős felhasználási

lehetőségére (pl. deepfake vagy adatvédelem terén). Összességében a cikk a GAN fejlődésének és annak a mesterséges intelligencia kutatás és az alkalmazott ipari területek közötti hidat képező szerepének magas szintű összefoglalását adja.

A fenti publikációkon alapulva kétfajta esetet különböztetünk meg a képből kép transzformációk során. Az elsőt párosított esetnek, míg a másikat párosítatlan esetnek nevezzük.

Jelöljön A és B egy-egy képtartományt (képhalmazt), és legyen a tetszőleges bemeneti forráskép, illetve b tetszőleges kimeneti célkép. Abban az esetben, ha minden $a_i \in A$ bemeneti képhez egyértelműen hozzárendelhető egy $b_i \in B$ célkép, akkor párosított képből-kép fordításról beszélünk, és a hozzátartozó adathalmaz:

$$\mathcal{D}_{\text{paired}} = \{(a_i, b_i)\}_{i=1}^N, \quad a_i \in A, b_i \in B. \quad (2.4)$$

A (2.4) egyenlet alapján párosított esetben a fő cél a $G : A \rightarrow B$ leképezés megtanulása, amelyre a pix2pix alapú architektúrák [83] adnak megoldást.

Abban az esetben, ha nem létezik egyértelmű kapcsolat $a_i \in A$ és $b_j \in B$ között, akkor párosítatlan képből-kép fordításról beszélünk, és az egymástól független képhalmazokat a következőképpen adjuk meg:

$$\mathcal{D}_A = \{a_i\}_{i=1}^{N_A}, \quad a_i \in A, \quad \mathcal{D}_B = \{b_j\}_{j=1}^{N_B}, \quad b_j \in B. \quad (2.5)$$

A (2.5) egyenlet alapján a leképezés kétirányúvá válik, mivel mind a $G : A \rightarrow B$, mind pedig a $F : B \rightarrow A$ fordítást biztosítani kell. Ennek megvalósítására alkalmasak a ciklikus konzisztencia-veszteségen alapuló architektúrák, amelyek közé a CycleGAN is tartozik [81].

A képből-kép fordítás irodalmában a generatív ellentétes hálózatokra épülő megközelítések mellett több alternatív módszer családj is megjelent, ezért a téma keretbe rendezéséhez rövid kitekintést adunk. Az egyik kézenfekvő irány a determinisztikus kódoló–dekódoló típusú modellek alkalmazása, ahol a hálózat a bemeneti képből közvetlen regresszióval állítja elő a célképet (például U-alakú architektúrákkal), tipikusan pixel-alapú (ℓ_1 , ℓ_2) és/vagy észlelési veszteségekkel. Előnyük a stabil tanítás és az egyszerű kiértékelhetőség, ugyanakkor többértelmű feladatokban (ahol több lehetséges

helyes kimenet létezik) gyakran elmosódott eredményekhez vezethetnek. A korábbi, automatikus kódoló alapú megoldások is ehhez a determinisztikus, rekonstrukció-orientált szemlélethez kapcsolódnak [77].

A valószínűségi keretek közül a variációs automatikus kódoló alapú módszerek explicit valószínűségi modellt adnak, és képesek többféle kimenet mintavételezésére is, ugyanakkor a nagy felbontású, éles részletek megbízható előállításuk sok esetben nehezebb, mint ellentétes tanításon alapuló eljárásokkal. További alternatívát jelentenek az invertálható transzformációkon (normalizáló áramlásokon) alapuló modellek, amelyekben a mintavételezés és a sűrűségszámítás elméletileg jól kezelhető, valamint a tanítás log-likelihood alapú optimalizációval is megfogalmazható, ugyanis képi feladatokban azonban az architektúra-korlátok és a memóriaigény gyakran gyakorlati kompromisszumokat tesznek szükségessé [84].

Az utóbbi években kiemelt figyelmet kaptak a diffúziós generatív modellek, amelyek zajhozzáadásos előre-folyamatot és ennek tanult inverzét (zajeltávolításos vissza-folyamatot) alkalmazzák [85, 86]. Az előre-folyamat tipikusan:

$$q(x_t | x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1 - \beta_t} x_{t-1}, \beta_t \mathbf{I}\right), \quad (2.6)$$

ahol x_0 a valós kép, x_t pedig a t -edik zajosított állapot. A generálás a vissza-folyamat lépéseinek iteratív alkalmazásával történik, amelyet különböző gyorsítási eljárások is támogatnak [87]. A diffúziós megközelítések egyik fő előnye, hogy ellentétes tanítás nélkül is kimagasló mintaminőséget képesek elérni, ugyanakkor a többlépcsős mintavételezés számításigénye ipari, erőforrás-korlátos vagy közel valós idejű felhasználásban gyakran korlátozó tényező. A képből-kép jellegű feladatok diffúziós keretben feltételes generálásként is megfogalmazhatók, ahol a modell a bemeneti képre feltételesen állít elő kimenetet. Ilyen irányú egységes keretet ad például a Palette [88].

A dolgozat a saját módszertani hozzájárulásokat a GAN-alapú képből-kép transzformációkra építi, mivel ipari munkafolyamatok támogatásában gyakran fontos a számítási igény, a taníthatóság és a gyakorlati integrálhatóság kedvező kompromisszuma, különösen erőforrás-korlátos környezetben vagy közel valós idejű alkalmazások esetén. Emiatt a dolgozat fókuszja a GAN-alapú megoldásokon marad, miközben az alternatív irányok rövid

bemutatása a téma általánosabb keretezését szolgálja.

2.4.1. Adatkészlet-transzformáció és tartományadaptáció

A képi adatkészletek transzformációja és a szimuláció–valóság átmenet kezelése (szimulációból a valós környezetbe történő átvitel) a modern ipari gépi látási rendszerek egyik meghatározó kihívása. A valós adatok gyűjtése és annotálása sok esetben költséges, időigényes, vagy a gyártási környezet korlátai miatt csak korlátozottan kivitelezhető, miközben a tanítandó modellek teljesítménye erősen függ a tanító és a célkörnyezet közötti eloszláskülönbségtől. Ennek csökkentésére az irodalomban több, egymással kombinálható megközelítés alakult ki, amelyek közös célja a forrás- és céltartomány közötti eltérés mérséklése az adatok megjelenésének, a reprezentációknak vagy a tanítási célfüggvénynek a módosításával [89].

A legegyszerűbb módszerek a címkemegtartó augmentációk, amelyek geometriai (például forgatás, skálázás, eltolás, perspektivikus torzítás) és fotometriai (például fényerő, kontraszt, színtónus, zaj, elmosás) transzformációkkal növelik a tanító adathalmaz variabilitását. Ezekkel a technikákkal a modell robusztusabbá tehető a szenzorok, a kamera-beállítások és a környezeti tényezők természetes változásaira. Szintetikus adatok esetén ennek kiterjesztett és célzott formája a tartomány véletlenszerűsítés, amely a szimulált környezet paramétereinek széles tartományú változtatásával (például megvilágítás, anyag- és textúratulajdonságok, háttér, objektumpóz, kameraállás, részleges takarások) igyekszik olyan tanító eloszlást létrehozni, amely „lefedí” a várható valós variációkat [90]. A módszer előnye, hogy címkék szintetikus környezetben automatikusan és nagy mennyiségben előállíthatók, hátránya viszont, hogy a paraméterterek megválasztása és a realiztikusság és változatosság közötti kompromisszuma nem triviális, mivel az előállított képek vizuálisan gyakran eltérnek a valós tartománytól.

A tartománykülönbség csökkentésének következő szintje a tartományadaptáció, ahol a cél már nem pusztán a változatosság növelése, hanem a forrás- és céltartomány közötti reprezentációs eltérés explicitebb kezelése. Az irodalomban elterjedt felosztás szerint a tartományadaptáció megvalósítható képszinten (a képek megjelenésének transzformálásával), jellemzőszinten (belső reprezentációk illesztésével), illetve hibrid módon, ahol a két megközelítés kombinálódik [89]. Képszintű adaptáció esetén a

cél, hogy a forrásadatok megjelenése a céltartományhoz hasonlóvá váljon, miközben az annotációk (például osztálycímkék, dobozok vagy szegmentációs maszkok) megőrizhetők legyenek. Ezt gyakran generatív modellekkel valósítják meg, mivel a transzformációt úgy lehet megtanítani, hogy a kimeneti képek statisztikai és vizuális jellemzői közelebb kerüljenek a valós tartományhoz, miközben a geometriai struktúrák és a címkék konzisztenciája fenntartható. A képszintű módszerek előnye, hogy a „megjelenési rés” közvetlenül csökkenthető, ugyanakkor gyakorlati kockázatot jelenthet a címkekonzisztencia sérülése (például finom kontúrok torzulása), amely különösen detektálási és szegmentálási feladatokban kritikus.

Jellemzősintű adaptáció esetén a tanítás célja olyan reprezentációk kialakítása, amelyekben a forrás- és céltartomány mintái nehezen megkülönböztethetők, így a forrásból tanult döntési határ jobban általánosíthat a célkörnyezetre. Klasszikus példa erre a tartományok közti eltérést csökkentő ellentétes tanítás, ahol egy tartománydiszkriminátor próbálja megkülönböztetni a forrás- és céltartományból származó jellemzőket, míg a fő hálózat célja ennek „megtévesztése”, azaz tartományinvariáns jellemzők tanulása [91]. A jellemzősintű adaptáció előnye, hogy nem feltételez explicit képszintű transzformációt, és sok esetben stabilabban integrálható meglévő tanítási folyamatokba. Ugyanakkor kihívást jelenthet, hogy a céloldali adatok gyakran címkézetlenek, és a tartományinvariancia erősítése a feladat-specifikus diszkriminatív információk csökkenéséhez is vezethet, ezért a veszteségkomponensek és a tanítási dinamika gondos hangolást igényel.

A fenti, tanuló alapú megközelítések mellett ipari környezetben gyakran előnyös a kontrollálható, reprodukálható, szabályalapú vagy hibrid adat-előállítás. Ilyenkor nem pusztán „stílus-szintű” közelítés történik, hanem a gyártási környezet fizikai és geometriai korlátai, valamint a szenzorparaméterek explicit figyelembevételével állítható elő olyan adatkészlet, amely célzottan lefedi a releváns eseteket. Ebbe a körbe sorolhatók például a tervrajz- és anyagmintázat-alapú szintetikus adatelőállítási folyamatok, illetve a digitális iker szemlélethez kapcsolódó, paraméterezett szimulációk, ahol a variációk kontrolláltan és dokumentáltan generálhatók. A dolgozatban ismertetett adatkészlet-transzformációs eljárások elsősorban ezen utóbbi irányokra épülnek, mivel ezek ipari környezetben jól kontrollálhatók, reprodukálhatók, és közvetlenül illeszthetők a tervezett gépi látási és

robotikai munkafolyamatokhoz.

2.5. Az alkalmazott mélytanuláson alapú architektúrák és metódusok bemutatása

2.5.1. GAN hálózatok alapjai

A generatív ellentétes hálózatok (GAN) két, egymással versengő neurális hálóból állnak: egy generátorból (G) és egy diszkriminátorból (D). A generátor feladata olyan szintetikus minták előállítása, amelyek a valós adateloszláshoz hasonlóak, míg a diszkriminátor célja annak eldöntése, hogy egy bemeneti minta valós adathalmazból származik-e, vagy a generátor által előállított szintetikus minta. A tanítás során a két háló egy minimax jellegű optimalizációs játékot játszik, ahol a generátor a diszkriminátor megtévesztésére, a diszkriminátor pedig a helyes megkülönböztetésre törekszik.

Legyen $p_{\text{data}}(x)$ a valós minták eloszlása, és $p_z(z)$ a generátor bemeneti zajvektorának eloszlása. A klasszikus GAN célfüggvénye:

$$\min_G \max_D \mathcal{L}_{\text{GAN}}(G, D) = \mathbb{E}_{x \sim p_{\text{data}}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log (1 - D(G(z)))]. \quad (2.7)$$

A fenti formulában a diszkriminátor $D(\cdot)$ egy 0 és 1 közé eső értéket ad vissza, amely a „valós” minta valószínűségi becsléseként értelmezhető. A generátor $G(\cdot)$ úgy tanul, hogy a $G(z)$ kimenetei a valós mintákhoz hasonlóvá váljanak, így a diszkriminátor ne tudjon megbízhatóan különbséget tenni a valós és a generált adatok között.

Gyakorlati alkalmazásokban a generátor tanítása sok esetben a következő, numerikusan stabilabb (nem szaturáló) alakban történik:

$$\min_G \mathcal{L}_G = -\mathbb{E}_{z \sim p_z(z)} [\log D(G(z))], \quad (2.8)$$

míg a diszkriminátor célja a valós és a generált minták szétválasztása az előző $\mathcal{L}_{\text{GAN}}(G, D)$ kifejezés maximalizálásával.

A generatív ellentétes hálózatok különösen alkalmasak nagy realiz-

musú képek előállítására és tartományok közötti leképezések tanulására. A disszertáció későbbi fejezeteiben alkalmazott képből-kép fordítási eljárások (például feltételes GAN-alapú megoldások) erre a kétkomponensű tanulási sémára épülnek, ahol a generátor célja egy bemeneti kép másik tartományba történő transzformálása, a diszkriminátor pedig a transzformált képek minőségének és realizmusának ellenőrzése.

2.5.2. pix2pixHD architektúra alapjai

A pix2pixHD eljárás egy feltételes generatív ellentétes hálózat (cGAN) architektúra, amelyet nagy felbontású képből-kép fordítási feladatokra terveztek [1]. Kiterjeszti az eredeti pix2pix modellt [83] egy többszintű generátor-diszkriminátor struktúrával és jellemző-illesztési veszteségek bevezetésével, lehetővé téve fotorealisztikus, részletes és nagy felbontású képek (jellemzően akár 2048×1024 pixel) szintézisét.

Az előző alfejezetben említett képtartományt leképező függvény valósítja meg a mélytanuló algoritmus, melynek architektúrája a következő főbb komponensekből áll:

- többskálás G generátor komponens
- többskálás D_1, D_2, D_3 diszkriminátorok

A generátor egy kódoló-dekódoló felépítésen alapszik, amelyet blokkokkal és többszintű jellemzőfúzióval bővítettek. Feladata, hogy megtanulja szintetizálni a célként előállítandó B kép tartalmát a bemeneti szemantikai vagy struktúrális A mátrix alapján.

A generátor kimenete több felbontáson keresztül fokozatosan javul. mivel az alacsonyabb felbontású felvételek eredményezik majd a globális struktúrális információkat, és a nagyfelbontású képek fogják megadni az apró részleteket, mint például textúrákat, megvilágítást és tükröződések.

Az algoritmus működéséhez három darab D_1, D_2 és D_3 diszkriminátor betanítására van szükség különböző térbeli méretekben (pl. teljes, fél és negyed felbontás). Ennek célja, hogy minden egyes diszkriminátor biztosítani tudja a valóság-hűséget a saját feldolgozási szintjén. D_1 a globális struktúrális szerkezetekért, D_2 a közepes szintű konzisztenciáért, míg D_3 a finom részletekért lesz felelős a működés során.

A másik nagy előnye ennek a felépítésnek, hogy a többszintű ellentétes visszacsatolás lehetővé teszi a generátor számára, hogy minden

felbontásban vizuálisan meggyőző eredményeket produkáljon.

A pix2pixHD teljes veszteségfüggvénye több kiegészítő komponenszt ötvözve a következőképpen írható le [1]:

$$L_{pix2pixhd} = \min_G \max_{D_1, D_2, D_3} \left(\sum_{k=1}^3 \mathcal{L}_{GAN}(G, D_k) + \lambda \sum_{k=1}^3 \mathcal{L}_{FM}(G, D_k) + \eta \mathcal{L}_{VGG}(G) \right), \quad (2.9)$$

ahol $\mathcal{L}_{GAN}(G, D_k)$ jelöli a generátor költségfüggvényét, $\mathcal{L}_{FM}(G, D_k)$ jelenti az úgynevezett jellemző egyezőségi veszteség komponenszt, és $\mathcal{L}_{VGG}(G)$ pedig a perceptuális veszteségfüggvény lesz.

A fenti (2.9) egyenlet alapján a G generátor fő célja, hogy a költségfüggvény teljes minimalizálja amellet, hogy a diszkriminátor komponensek pont ellentétesen a maximalizálásra törekszenek.

A pix2pixHD $\mathcal{L}_{GAN}(G, D_k)$ költségfüggvénye az alábbi összefüggéssel adható meg:

$$\mathcal{L}_{GAN}(G, D_k) = \mathbb{E}_{(s,x)}[\log D_k(s, x)] + \mathbb{E}_s[\log(1 - D_k(s, G(s)))]. \quad (2.10)$$

A pix2pixHD $\mathcal{L}_{FM}(G, D_k)$ veszteségfüggvénye a következő egyenlettel írható le:

$$\mathcal{L}_{FM}(G, D_k) = \mathbb{E}_{(s,x)} \sum_{i=1}^T \frac{1}{N_i} \left\| D_k^{(i)}(s, x) - D_k^{(i)}(s, G(s)) \right\|_1. \quad (2.11)$$

A pix2pixHD $\mathcal{L}_{VGG}(G)$ perceptuális veszteségfüggvénye a következő egyenlettel írható le:

$$\mathcal{L}_{VGG}(G) = \mathbb{E}_{(s,x)} \sum_{j=1}^L \frac{1}{M_j} \left\| \phi_j(x) - \phi_j(G(s)) \right\|_1. \quad (2.12)$$

2.5.3. YOLO alapú architektúrák

A YOLO (You Only Look Once) egy valós idejű objektumfelismerő algoritmuscsalád, amely a felismerést több lépéses osztályozási és lokalizációs folyamat helyett egylépéses regressziós problémaként fogalmazza meg. A korábbi modellektől eltérően, mint például az R-CNN, a Fast R-CNN vagy a Faster R-CNN — amelyek először régiójavaslatokat generálnak,

majd ezt követően osztályozzák ezeket — a YOLO a teljes képet egy lépésében dolgozza fel.

A YOLO (You Only Look Once) leképezési függvénye a következőképpen adható meg:

$$f_{\theta} : I \rightarrow \{(b_i, c_i, p_i)\}_{i=1}^N, \quad (2.13)$$

ahol I jelöli a bemeneti képet, $b_i = (x_i, y_i, w_i, h_i)$ jelenti a prediktált körülhatároló keretet, c_i lesz a hozzátartozó osztálycímke és p_i pedig az ehhez tartozó valószínűség értéke.

A bemeneti kép I legyen felosztva egy $\zeta \times \zeta$ rácsra. Minden rácscella felelős azoknak az objektumoknak az észleléséért, amelyek középpontja a cella határain belül esik. Minden (i, j) cellára a modell prediktálni fogja a $\{b_k = (x_k, y_k, w_k, h_k)\}_{k=1}^B$ körülhatároló kereteket egy adott C_k megbízhatósági pontszámmal és a $P(c|\text{object})$, $c \in \{1, 2, \dots, C\}$ egy osztály valószínűsége halmazával. Itt C_k a k -adik prediktált körülhatároló keret megbízhatósági pontszáma (objektumosság), amely azt fejezi ki, hogy a cellához tartozó predikció mennyire tartalmaz objektumot, illetve mennyire illeszkedik a valós keretre. A klasszikus felírás szerint $C_k = P(\text{objektum}) \cdot IoU(b_k, b^{gt})$, ahol $P(\text{objektum})$ annak valószínűsége, hogy a cellában van objektum, $IoU(b_k, b^{gt})$ pedig a prediktált és a valós körülhatároló keret átfedését méri.

A hálózat kimenete legyen \mathbf{Y} , ekkor

$$\mathbf{Y} \in \mathbb{R}^{\zeta \times \zeta \times (B \times 5 + C)}, \quad (2.14)$$

ahol a $(B \times 5)$ kifejezés a körülhatároló keret koordinátáit és a megbízhatósági értékeket kódolja, és C az objektumosztályok számát jelöli. A harmadik dimenzió pedig a cellánkénti predikciós vektor hossza lesz.

A YOLO egyetlen konvolúciós neurális hálózatot (CNN) használ a határolókeretek és az osztályok valószínűségének egyidejű előrejelzésére. Ez a végpontok közötti tanítás lehetővé teszi a teljes kép globális értelmezését, így a modell képes a kontextus információk rögzítésére és rendkívül gyors felismerésre.

Maga az architektúra két fő részből áll, amelyek az ún gerincrészt és fejrészek lesznek. A gerincrész felelős a jellemzők kinyeréséért, a fej-

rész pedig több konvolúciós rétegből álló detektálónak tekinthető, amely regressziós módszerrel számítja ki a doboz koordinátáit és az osztály pontszámait.

Az architektúra előnyei közé sorolható a rendkívüli gyors detektálási sebesség, amely lehetővé teszi a valós idejű működést is, továbbá a globális kontextus értelmezése, mellyel a téves pozitívok száma csökkenthető, és a kompakt méret is, mellyel beágyazott eszközökön is alkalmazhatóvá válik. Az architektúra számos limitációval is rendelkezik, ezek közé sorolható a kisebb objektumok esetében a pontosság csökkenése, a sebesség és a pontosság közötti kompromisszum megtalálása, illetve a határolókeretkorlátozások is, mivel az ún. horgony-alapú tervezés korlátozhatja a rugalmasságot szabálytalan alakú objektumok esetében. A következő részekben a kutatás során alkalmazott főtípusok (YOLO-v3[92] és YOLO-v8 [93]) is bemutatásra kerülnek.

YOLO-v3

A YOLOv3 detektorok [92] szintén a következő két fő architektúra-komponensre oszthatók: a gerinchálózat és az detektálási részek. A gerinchálózat blokk az elsődleges rétegekből áll, mint például a konvolúciós, a köteg-normalizációs, a MaxPool és a LeakyRELU aktivációs rétegek, amelyek az elsődleges képjellemzők kivonásáért felelősek. Másrészt az észlelési rész a gerinchálózat blokk által összegyűjtött jellemzők alapján megjósolja az objektum határolókeretét.

Esetünkben a következő négy különböző architektúrát választottuk ki az elsődleges beépített detektorokhoz: YOLOV3-tiny, YOLOV3-tiny-3l, YOLOV3-SPP és YOLOv3-5l [94].

A következő egyenletek a detektorokhoz kapcsolódó veszteségfüggvényt, illetve méretezési összefüggéseket ismertetnek.

A V3-as verziójú YOLO alapú detektorok veszteségfüggvénye a következőképpen fejezhető ki:

$$\begin{aligned}
\mathcal{L}_{\text{YOLOv3}} = & \lambda_{\text{coord}} \sum_{i=0}^{S^2} \mathbb{K}_i^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (\sqrt{w_i} - \sqrt{\hat{w}_i})^2 \right. \\
& \left. + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2 \right] + \sum_{i=0}^{S^2} \mathbb{K}_i^{\text{obj}} (\hat{C}_i - 1)^2 \\
& + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \mathbb{K}_i^{\text{noobj}} (\hat{C}_i)^2 + \sum_{i=0}^{S^2} \mathbb{K}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2.
\end{aligned} \tag{2.15}$$

Az egyes YOLO rétegek előtt álló konvolúciós szűrők mennyiségét a következőképpen számíthatjuk ki:

$$F = 3 \cdot (C + 5), \tag{2.16}$$

ahol C jelöli a detektálandó osztályok számát.

A V3-as verziójú YOLO detektorok optimális és szükséges tanítási iterációnak száma a következőképpen számítható ki:

$$E = 2000 \cdot C, \tag{2.17}$$

ahol E jelöli a maximális tanítási iterációk számát, és C lesz a detektálandó objektumok száma.

YOLO-v8

A YOLO-v8 jelentős architektúraváltást jelent a YOLO-v3-ban használt korábbi horgony-alapú tervezéshez képest. Míg a YOLO-v3 előre meghatározott horgony-keretekre és az azokhoz viszonyított, előre jelzett kereteltolásokra támaszkodott, a YOLO-v8 horgony-mentes felismerési stratégiát alkalmaz. Ez egyszerűsíti a kimeneti ábrázolást és szükségtelenné teszi az horgony-ok kézi beállítását. Ezenkívül a YOLO-v8 egy szétválasztott felismerő fejet alkalmaz, amely elválasztja a besorolási és regressziós ágakat, javítva a gradiens áramlást és a konvergenciát. A gerinchálózat is fejlődött: a YOLO-v3 Darknet-53 struktúráját egy hatékonyabb, CSP-alapú tervezés váltotta fel, amelyet Cross-Stage-Partial kapcsolatok és egy mélyebb, könnyebb PAN-neck egészít ki a többszintű jellemzők fúziójához.

A YOLO-v3 veszteség ötvözi a határolókeret koordinátáinak és osztályelőrejelzéseknek a középnégyzetes hibáját (MSE) egy olyan kifejezéssel, amelyet koordinációs és objektummentes súlyozási tényezők szabályoznak. Ezzel szemben a YOLO-v8 ezt a formulát egy többfeladatos hibrid veszteséggel helyettesíti, amely integrálja a DFL (Distribution Focal Loss)-t a határolókeret-regresszióhoz, a CIoU (Complete-IoU)-t a geometriai konzisztenciához, valamint a BCE (Binary Cross-Entropy)-t az osztályozáshoz és az objektumosság pontszámokhoz. Ezek a változások a YOLO-v8-at robusztusabbá teszik a méretváltozások, az osztályok egyensúlytalansága és az átfedő objektumok tekintetében, így nagyobb pontosságot és stabilitást biztosítanak az horgony egyezőség komplexitása nélkül.

A YOLOv8 alapú architektúrák veszteségfüggvénye a következőképpen fejezhető ki:

$$\mathcal{L}_{\text{YOLOv8}} = \lambda_{\text{box}} \mathcal{L}_{\text{CIoU}} + \lambda_{\text{dfl}} \mathcal{L}_{\text{DFL}} + \lambda_{\text{cls}} \mathcal{L}_{\text{BCE}}^{\text{cls}} + \lambda_{\text{obj}} \mathcal{L}_{\text{BCE}}^{\text{obj}}. \quad (2.18)$$

2.6. A kiértékelés során alkalmazott metrikák bemutatása

Az objektumdetektorok és hozzárendelő modellek teljesítményének értékelésére a leggyakrabban használt metrikák a pontosság (precision), a visszahívás (recall), az F1-mutató és az átlagos precízió (mean Average Precision, mAP). Ezek a mutatók a detekciós eredmények és a valós címkék (ground truth) közötti egyezést mérik [95].

Az *Intersection over Union* (IoU) metrika a valós és a detektált objektumok közti átfedést méri:

$$IoU = \frac{|\mathcal{A} \cap \mathcal{B}|}{|\mathcal{A} \cup \mathcal{B}|}, \quad (2.19)$$

ahol \mathcal{A} jelöli a valós régiót, \mathcal{B} pedig a prediktált régiót.

A **pontosság** a helyes pozitív detekciók arányát mutatja az összes pozitív előrejelzéshez viszonyítva [96]:

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (2.20)$$

ahol TP a helyes pozitív, FP pedig a téves pozitív detekciók száma.

A **visszahívás** azt mutatja meg, hogy a modell a valós pozitív minták hány százalékát ismerte fel [96]:

$$\text{Recall} = \frac{TP}{TP + FN}, \quad (2.21)$$

ahol FN a fel nem ismert (tévesen negatív) esetek száma.

A két mutató (pontosság és visszahívás) harmonikus átlaga az **F1-mutató**, amely kiegyensúlyozottan méri a pontosság és a visszahívás közötti kompromisszumot [96]:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (2.22)$$

Megjegyezzük, hogy a TP , FP és FN mennyiségekre épülő mutatók (pontosság, visszahívás és F_1) eredetileg bináris osztályozási esetben kerültek bevezetésre. Objektumdetektálásban a TP , FP , FN fogalmakat a dobozpárosítás, az osztályegyezés, valamint egy előre rögzített IoU küszöbérték alapján értelmezzük: egy predikció akkor TP , ha egy valós objektumhoz párosítható, az osztálycímke egyezik, és az IoU meghaladja a küszöböt; a nem párosítható predikciók FP -nek, a párosítatlan valós objektumok pedig FN -nek számítanak.

Többosztályos esetben a pontosság, visszahívás és F_1 értékek osztályonként is kiszámíthatók egy „egy az összes többi ellen” szemlélet szerint. Jelölje Precision_k , Recall_k és $F_{1,k}$ a k -edik osztályhoz tartozó értéket. Az osztályonkénti mutatók összesítésére gyakori a makróátlag (az osztályonkénti értékek átlaga) és a mikroátlag (az összesített $TP/FP/FN$ alapján képzett érték), amelyek eltérően érzékenyek az osztályok gyakoriságára.

Az **átlagos precízió** (AP) az úgynevezett precízió–visszahívás görbe alatti területet méri [97]:

$$AP = \int_0^1 \text{Precision}(r) dr, \quad (2.23)$$

A **mAP** (mean Average Precision) az összes osztályra vett átlagos

értéket jelenti :[97]

$$mAP = \frac{1}{N_c} \sum_{k=1}^{N_c} AP_k. \quad (2.24)$$

A YOLO családú modellek esetében a leggyakrabban használt értéke-
lési szintek az $mAP@0.5$ és az $mAP@0.5 : 0.95$, amelyek az Intersection
over Union (IoU) küszöbértékét 0.5-re, illetve 0.5 és 0.95 közé állítják.

Ezek a metrikák együttesen teljes képet adnak a detektorok és hozzá-
rendelő modellek teljesítményéről, mind a pontosság, mind a robusztusság
szempontjából.

3. fejezet

Képfordításon alapuló megközelítés alkalmazása ipari rendszerben

Az Ipar 5.0 új lehetőségeket nyit meg az ipari automatizálás számára. Ez új kihívást jelent a mérnökök számára is, amely rendszerszintű gondolkodást igényel. A gyártás során a gyártósor olyan adatokat generál, amelyeket elemezni kell. A nagy mennyiségű adatot mesterséges intelligencia segítségével lehet értelmezni és értékelni. A probléma az, hogy sokféle mesterséges intelligencia modell létezik. Ezért a megfelelő adatállománymal rendelkező megfelelő modell kiválasztása és betanítása kihívást jelentő feladat.

Az iparban számos régebbi gépet használnak, amelyek funkciói modernizálással bővíthetők. Ezek a berendezések azonban gyakran nem felelnek meg a korszerű szabványoknak, így nem képesek hálózaton keresztül kommunikálni [98]. A valós idejű pályagenerálás és az ütközésselkerülés rendkívül összetett feladat, amely számos tényezőtől függ [99–101]. A mélytanuló neurális hálózatok alkalmazása hatékonynak bizonyult a kulcsfontosságú pontok osztályozásában [99, 100]. Az elavult rendszerek frissítése komoly kihívást jelent, mivel a fizikai hardverek többsége nem frissíthető [101]. A neurális hálózatokat korábban már sikeresen alkalmazták 2D robotok útvonaltervezésében is [99, 100], ahol képfeldolgozással azonosították a munkadarabokat és a kritikus pontokat [101]. A robotrendszerek áttervezése

ugyanakkor további nehézségekkel jár, mivel a beágyazott vezérlők konfigurációs lehetőségei korlátozottak, ezért a képelemzés önállóan, kizárólag a robotvezérlőn nem valósítható meg.

Az innovatív megoldásokkal ez a probléma megoldható, és a gépek élettartama meghosszabbítható. A fejezet bemutatja, hogyan lehet a mélytanulást beépíteni a régebbi robotokba, és hogyan lehet modelleket betanítani. A projektben DL-alapú (mélytanulás) technikát alkalmazunk, amely nem igényel jelentős átalakításokat, és mesterséges intelligenciát használunk a szállítószalagon lévő kocka felismeréséhez. Korábban már megvitattuk a régi robotok funkcionalitásának kiterjesztésének lehetőségeit is [102]. Ebben az esetben a kép szintézise során a teljes környezetet, beleértve a megvilágítás és a színek tükröződéseit is, megvizsgáljuk.

Ez a fejezet az alábbiak szerint épül fel. Az anyagmozgató gép alapjait és paramétereit a 2.2.2 szakasz mutatta már be korábban. A valós képekből vett tipikus jelenségeket kapcsolatos részletek a 2.4 szakaszban találhatóak. A 3.1 szakasz a modellezési eljárás részleteit ismerteti. Az 3.2 szakasz az képből kép alapú fordítással létrehozott adatkészlet mögött álló elképzelést ismerteti, és leírja a kiválasztott architektúrát. A 3.2.2 szakasz a kiválasztott detektor algoritmusok részleteit magyarázza, a 3.3 szakasz pedig összefoglalja a környezetet, valamint a tanítási folyamatban változó paramétereket. A 3.4 szakasz a képminőségtől függetlenül értékeli a mélytanuláson alapuló algoritmusok hatékonyságát, a 3.5 szakasz pedig a fejezet következtetéseit tartalmazza.

3.1. 3D CAD modellezés és képi nyers adathalmaz generálása

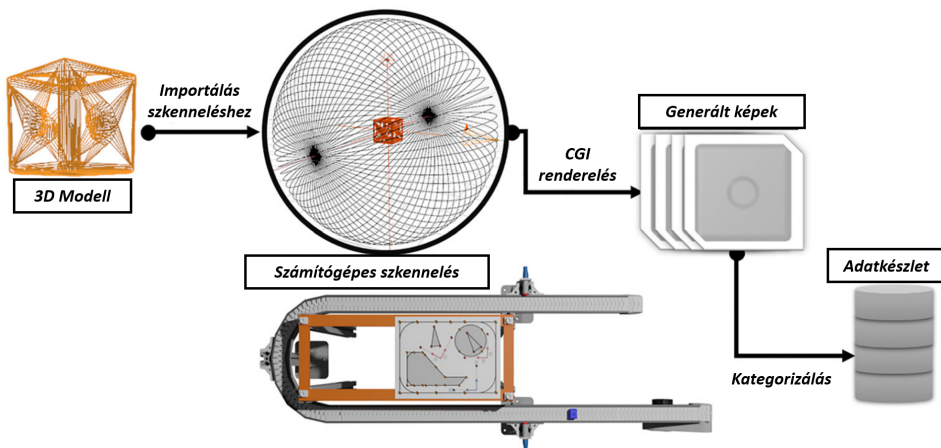
Meg kell jegyezni, hogy az ipari robotrendszerek alapértelmezés szerint nem képesek a nyersanyagok térbeli helyzetének meghatározására. Az IP-kamerák felvételeinek felhasználásával azonban a szállítószalagon mozgó tárgyak egyértelműen azonosíthatók.

Az eljárás ugyanakkor időigényes, mivel a gépi tanulási algoritmusok nagy mennyiségű, változatos körülmények között előállított bemeneti adatot igényelnek. Amennyiben azonban a Flexlink XK rendszer teljes 3D CAD-

modellje rendelkezésre áll, a virtuális renderelés segítségével tetszőleges számú szintetikus képfájl generálható.

A 3D CAD-modell a SketchUp szoftverben készült, és vázlattevként szolgált a kiválasztott, képfordításon alapuló módszer bemeneti adatainak előállításához. A módszerben a Flexlink XK modell várhatóan mintegy kétezer, egymástól független alkatrészt tartalmazó, részletes összeállításként kerül felhasználásra.

Számos módszer létezik a megfelelő méretű és változatosságú kép-adatbázis létrehozására. A jelen kutatásban a jelenet egy virtuális szkennelőrrel került megvalósításra, amely kulcsképeket rögzített különböző nézőpontokból. Az objektumot a virtuális 3D tér origójába helyeztük, míg a szkennelő kamera szabályos körpályán mozgott körülötte (lásd a 3.1. ábrát). A virtuális kamera egy teljes kör mentén 360 képet renderelt, amely folyamat szükség szerint többször is megismételhető. A kocka alakú objektumot ezen felül animáltuk, hogy áthaladjon a Flexlink XK szállítószalagon, így biztosítva a mozgás- és pozícióváltozások reprezentációját a kép adatokban.



3.1. ábra. A 3D CAD modellezés és renderelési folyamat.

3.2. A javasolt architektúrák és megközelítések bemutatása

Ez a szakasz részletesebb magyarázatot ad a kiválasztott képfordítási technikáról, amely fotórealisztikus képek létrehozására szolgál, továbbá az alkalmazott valós idejű detektorok is bemutatásra kerülnek.

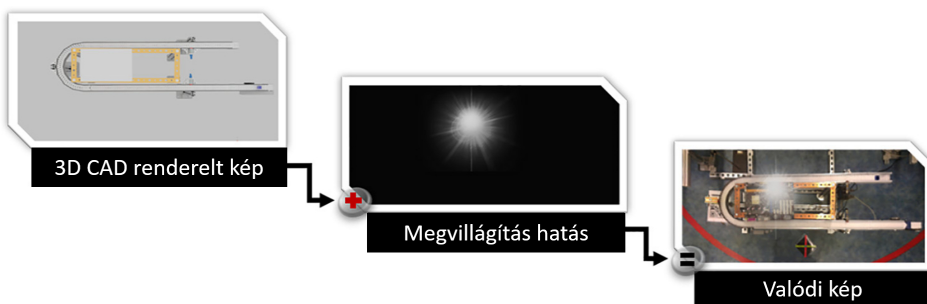
3.2.1. Képből kép fordítás (pix2pixHD)

A közelmúltban számos alkalmazás került kifejlesztésre a mélytanulás területén. Brnatskyi bemutatta a GAN (generatív ellentétes hálózatok) újszerű felhasználását, amely egy jelentősen átalakított neurális hálózat volt [103]. Ez az architektúra tartalmazott egy digitális vizuális feldolgozó réteget [104]. Eversberg és munkatársai azt vizsgálták, hogy a fizikaalapú renderelés (PBR) és a tartomány véletlenszerűsítő (DR) technikák hogyan befolyásolják az ipari objektumok vizuális felismerésének pontosságát szintetikus generált adatok felhasználásával [105]. Jason Li és munkatársai bebizonyították, hogy a tartomány randomizációval tanított modellek hatékonyan működnek különböző ipari környezetekben [106]. Különösen akkor, ha a „valós világból” származó adatok gyűjtése korlátozott vagy költséges. Xu Yang és társai bemutattak egy módszert, amely ötvözi a tartomány randomizációt és a stílusátvitelt. A javasolt megközelítés javítja az objektumok felismerésének pontosságát valós környezetben és csökkenti a kiterjedt valós adatgyűjtés szükségességét, így a szintetikus adatok hatékonyabbá válnak a mélytanulási modellek képzésében [107]. Jiwei Hu és munkatársai egy érdekes megközelítést mutattak be, amely egy továbbfejlesztett CycleGAN architektúrán alapul, integrálva a reziduális sűrű blokkok (RDB) és a strukturális hasonlósági index mérőszámmal [108]. Ez a megközelítés költséghatékony megoldást kínál az ipari objektumfelismerési feladatokban használt tanítási adatkészletek bővítésére. Összehasonlítva az ipari robotikában eddig alkalmazott megközelítéseket, a mi általunk javasolt módszertan egyedülálló módon integrálja a nagy felbontású, fotórealisztikus szintetikus képalkotást, és kifejezetten a régebbi, hagyományos robotrendszereket célozza meg. Az ultrarealisztikus képek létrehozása érdekében azonban a képtartomány-átalakításra is összpontosítottunk, amelynek segít-

ségével megtanulhatjuk a fotórealisztikus környezetek jellemzőit és alakjait, beleértve a jelenségeket és az anyagokat is.

Ebben a munkában többrétegű Gauss-képpirámis alapú konvolúciós zajszűrési és automatikus kódolási neurális hálózati modellt használtunk. Ez a modell képes szintetizálni az észlelési eredményeket azáltal, hogy adott felbontásban újrateremti a képfoltokat. Szintén releváns Kaji [109] munkája, amely a transláció alapú módszerek alkalmaságát vizsgálja a rekonstrukcióval, zajszűréssel és a nagy felbontás elérésével kapcsolatos problémák megoldására az orvosi képelemzésben. Esetünkben tartomány átalakítást kell alkalmazni, amely a $G : A \mapsto B$ képtartomány leképezés megtanulásán alapul.

Fontos megjegyezni, hogy mivel az $\{a_i, b_i\}$ adathalmaz értelmezett, és az a_i képeknek léteznek b_i képpárjai, ezért a párosított eseteket fogjuk alkalmazni. A rendelkezésre álló megoldások közül pedig a pix2pixHD alapú architektúrát [1] választottuk, mivel kivételes pontossággal, nagy felbontású és fotórealisztikus képeket képes előállítani. A detektorok betanításához szükséges képgyűjtemény létrehozásához a renderelt képeket át kellett alakítani (lásd a 3.2 ábrán).



3.2. ábra. A nyers renderelt képek tartományfordítása.

Számos tanítást és tesztet végeztünk a CycleGAN [110] architektúrára alapozva. A pix2pixHD megközelítéshez képest lényegesen kevésbé pontos eredményeket kaptunk.

3.2.2. Az alkalmazott valós idejű működésű, mélytanuló detektorok

Az egyik korábbi munkában a mélytanuláshoz egy YOLO (You Only Look Once) nevű architektúrát használtak vizuális ellenőrzésként, amelyet Zamora-Hernandez [111] írt le. FPGA-gyorsítás és a YOLO V3-on alapuló implementáció segítségével Yu egy gépi látáson alapuló hibajelző eljárást javasolt [112]. Zhou egy másik tanulmányában a YOLO 4. verzióján és a MobileNetv2-n alapuló vegyes/hibrid módszert mutatott be, amelynek célja a fényképeken látható, észrevehetően kicsi tárgyak/objektumok felismerésének pontosságának növelése [113]. Módszerünkhöz a YOLO architektúrát [114] választottuk a valós idejű észleléshez, mert ez különböző feladatokban gyorsan megvalósítható, és az észlelés pontossága magas.

Esetünkben elsősorban beágyazott detektorokat alkalmaztunk, amelyekhez négy különböző architektúrát választottunk. A vizsgált detektorok a következők voltak: YOLOv3-5l, YOLOv3-SPP (Spatial Pyramid Pooling), valamint két kisebb változat a YOLOv3-ból (YOLOv3-tiny és YOLOv3-tiny-3l).

Ahelyett, hogy jelentős architektúraváltoztatásokat hajtottunk volna végre, a detektorblokkokat az adatszintézis során előállított adathalmazhoz optimalizáltuk. Elsődleges célunk a detektorok funkcionalitásának és pontosságának megerősítése volt a generált képek elemzése alapján.

Az egyes architektúrák méretezéséhez a 2.16 egyenletet alkalmaztuk. Jelen esetben a szűrők száma 18 volt, mivel a detektálás egyetlen objektumtípusra korlátozódott.

Végül újraszámoltuk a megfelelő horgony értékeket, hogy a beépített k-means++ klaszterezési algoritmus segítségével [115] a lehető legjobb felismerési teljesítményt érjük el. Az új értékeket a 3.1. táblázat foglalja össze.

3.3. Tanítási és validálási folyamat

Ez a szakasz a tanítási és validációs eljárás további részleteit és paramétereit tartalmazza. Minden neurális hálózat egy Ryzen 5900x CPU-val és NVIDIA RTX 2070 GPU-val felszerelt számítógépen lett betanítva.

3.1. táblázat. A kiszámított horgony-értékek minden egyes detektor esetében.

Horgony index	YOLO v3 Spatial Pyramid Pooling	YOLO v3-5l	YOLO v3-tiny-3l	YOLO v3-tiny
0	12×7	10×6	12×7	10×9
1	9×10	10×10	9×10	13×8
2	10×10	9×10	10×10	11×9
3	11×10	10×10	11×10	11×10
4	11×10	11×10	11×10	12×10
5	11×11	11×11	11×11	–
6	13×10	11×13	13×10	–
7	14×9	14×9	14×9	–
8	15×11	13×10	15×11	–
9	–	11×10	–	–
10	–	11×10	–	–
11	–	12×11	–	–
12	–	13×11	–	–
13	–	14×11	–	–
14	–	15×12	–	–

3.3.1. pix2pixHD

A szintetizált fotók létrehozásához a következő mélytanulási hálózati konfigurációt és hiperparamétereket választottuk. Az Adam algoritmust [116] alkalmaztuk optimalizálóként, 0,0005 tanulási sebességgel, amely az első 40 tanítási ciklusban állandó maradt, majd lineárisan csökkent.

A legnagyobb kihívást a hálózat hiperparamétereinek legjobb kombinációjának megtalálása jelentette. Ezért folyamatosan tanítottuk az architektúrát az adatkészletek alapján, és több architektúraparamétert is módosítottunk. A diszkriminátorok számát 2-ről 5-re növeltük, a lefelé mintavételi szinteket 3-ról 9-re emeltük, valamint a jellemzőklasztereket 10-ről 16-ra szintén módosítottuk. Ezeknek a kiválasztott értékeknek köszönhetően a tanítási folyamat jobban konvergált, és a legjobb eredményeket kaptuk. Az algoritmust 1024×576 képpontos, pixelcímkézés nélküli képpárok (400)

alapján tanítottuk. A választott tanítási ciklus értéke 200 volt. A tanítóköteg méretét 20-ra állítottuk be, hogy minimalizáljuk a feldolgozási terhelést. A valós fotókhoz képpárokat készítettünk a tényleges helyszínen, normál gyári környezeti feltételek mellett. A képzés során kizárólag forgatás és tükrözés adatdúsítást alkalmaztunk. A teljes hálózat betanítása körülbelül 26 órát vett igénybe.

3.3.2. YOLO alapú detektorok

Az első lépésben elkészítettük a kezdeti adatkészletet, amelyet tanító és validációs komponensekre osztottunk. A validációs részben csak 100 ténylegesen rögzített fénykép szerepelt, míg a tanító adatkészletben szereplő 400 kép szintén a mi fordítási algoritmusunk segítségével lett szintetizálva. Mindkét esetben a kép felbontása 512×256 volt. A tanulási folyamat legnehezebb része az volt, hogy elkerüljük a túltanítással járó problémákat. Ezért számos adatdúsító algoritmust is alkalmaztunk, mint például színárnyalat-eltolást, képek méretezését, valamint a telítettség és az expozíció beállítását [117]. Az ezekhez kapcsolódó értékeket a 3.2 táblázat foglalja magába.

3.2. táblázat. A választott adatdúsító eljárások a hozzátartozó paraméterértékekkel.

Algoritmus neve	Választott paraméter értéke
Szaturáció	1.5
Expozíció	1.5
Átméretezés	1.5
Színezet-eltolás	0.3

A detektorok tanítása során többfajta optimalizálót is megvizsgáltunk mint például a sztochasztikus gradiens alapú módszert [118], és az Adam algoritmust is [116]. Eredményképpen utóbbit választottuk, mivel az elsőnél számos esetben instabil tanítást kaptunk.

Ezenkívül szükség volt a tanulási sebesség több paraméterének konfigurálására, például az ütemező és a kezdeti tanulási érték beállítására. Ezek segítségével minden tanítási iterációban megfelelő tanulási sebesség görbét kaptunk. Esetünkben a megfelelő típus egy exponenciális ütemezőre esett, amely az alábbi egyenlet segítségével fejezhető ki:

Az exponenciális tanulási ráta ütemező a következő egyenlettel írható le [119]:

$$l_r = l_{\text{initial}} \cdot \exp\left(-\frac{k}{\text{epochs}}\right), \quad (3.1)$$

ahol l_{initial} jelöli a kezdeti tanulási sebességet, $-\frac{k}{\text{epochs}}$ fejezi ki a csökkenés mértékét minden egyes tanítási iteráció alatt, a k számszerűsíti a csökkenés sebességét, és az *epochs* lesz a tanítási iterációk száma.

Számos kísérletet végeztünk annak érdekében, hogy meghatározzuk a kezdeti értékek és a k paraméter optimális beállításait. A $k = 0, 1$ számot így kapjuk meg, ha logaritmikus keresést alkalmazunk a paraméter intervallumra. A 3.3 táblázat tartalmazza a legjobb tanulási sebesség értékeket, amelyeket felfedeztünk. A tanulási paramétereket, beleértve a beégett és a maximális tanítási ciklusok értéket, a következő szakaszban tovább finomítottuk. Mivel a beégett paraméter felelős a képzési folyamat konvergenciájának és stabilitásának kialakításáért, ennek az értéknek a gondos kiválasztása elengedhetetlen. Minden detektor esetében az optimális érték a mi helyzetünkben 100 volt.

3.3. táblázat. A YOLO architektúrák detektorparaméterei és betanítási ideje.

Detektor neve	Tanulási sebesség	Momentum értéke	Csökkentés értéke	Tanítási idő (h)
YOLO Version3-tiny	0.007	0.92	0.0003	7.82
YOLO Version3-tiny-3l	0.007	0.92	0.0003	8.45
YOLOV3-Spatial Pyramid Pooling	0.0005	0.92	0.0003	30.32
YOLO Version3-5l	0.0002	0.92	0.0003	32.25

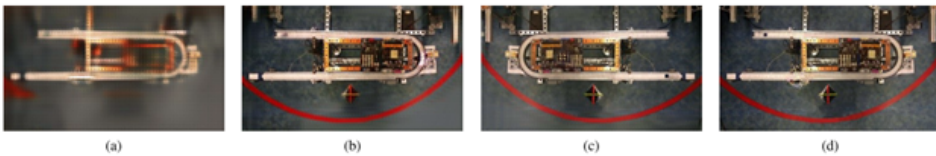
Az utolsó lépés részeként meg kellett határoznunk az alkalmazható tanítási iterációk számát, amelyet a 2.17. egyenlet alapján határoztunk meg. Esetünkben a szükséges érték természetesen 2000 lett.

A 3.3 táblázat összefoglalja az egyes YOLO detektorok egyedi időértékeit. Kisebbs detektorok, például a YOLO Version3-tiny és a YOLO Version3-tiny-3l esetében a minibatch értéket nyolcra állítottuk be, azonban a nagyobb típusoknál (YOLOV3- Spatial Pyramid Pooling és YOLO Version3-5l) esetében csak négy kép fért el a videokártya memóriájában.

3.4. Eredmények

3.4.1. pix2pixHD

Az 3.4 táblázat a kiválasztott pix2pixHD módszer eredményeit mutatja be. A 3.3 ábra a tanítási eljárás során kapott mintaképeket mutatja. További mérőszámokat is kiválasztottunk, amelyeket az egész képre és az objektumok régióira alkalmaztunk. Annak megállapításához, hogy a jelenlegi tanítás sikeres-e, egy komplex wavelet strukturális hasonlóság (CW-SSIM) [120] néven ismert robusztus megközelítést alkalmaztunk, mivel ez a mi esetünkben nagyon hasznos, és különböző transzformációk, például forgatás és átméretezés esetén is invariáns marad. A klasszikus strukturális hasonlóságot is teszteltük, viszont megállapítottuk, hogy az esetünkben nem feltétlenül alkalmazható, mivel az objektumok és a képek méretének eltérése téves értékeket eredményezhet.



3.3. ábra. A pix2pixHD algoritmus kimenetei a szintetizált képsorozatok képzésének során: (a) mintavett kép a képzés kezdetén, (b) mintavett kép körülbelül 0,5-ös hasonlósági értéknél, (c) mintavett kép 0,6 feletti hasonlósági értéknél, (d) mintavett kép a képzés végén.

Alkalmazásunk esetében több tanítási és esettanulmányi tesztet végeztünk, például különböző fényviszonyok és átrendezett környezeti háttér mellett. Mivel az információtartalom döntő fontosságú a tanítás szempontjából, a kiválasztott mérőszámot nem csak a teljes kép, hanem az objektumrégiók standard eltéréseivel is kiszámítottuk. Kezdetben azt tapasztaltuk, hogy a teljes kép régiójának minősége viszonylag gyorsan, néhány iteráció alatt elérte a magas szintet. Az objektumrégiók és azok tartalma azonban csak a tizedik iteráció után váltak megkülönböztethetővé. Fontos megjegyezni, hogy a tanítás sikeressége elsősorban a probléma jellegétől függ. A CW-SSIM segítségével azonban egyértelműen megállapítottuk azt a kritikus szintet, amely alatt az objektum nem különböztethető meg a

háttértől. Minden tesztesetben ezt a kritikus értéket 0,5-nél tapasztaltuk.

Az adatkészlet standard értékeinek (legjobb és legrosszabb eredmények) kiszámításával szintén megállapítottuk, hogy bár az objektumok szakaszai jelentősen nagyobb értékekkel rendelkeztek, a teljes képek nagyon kevés eltérést mutattak. A második esetben a hasonlóság értéke nagy hatással lehet a detektorok teljesítményére, a régiók minőségének arányától függően.

3.4.2. A YOLO detektorok

A 3.5 táblázat a betanított YOLO detektorok eredményeit tartalmazza. Referenciaként a detektorokat valós képeken is betanítottuk, és a táblázat első sora tartalmazza az ehhez tartozó értékeket. A generált adatkészlet viszonylag nagy volt, 256 000 feldúsított képből és 400 valós képből állt a validáláshoz. Az előző fejezetben említettek szerint a kapott CW-SSIM érték döntő fontosságú a detektorok számára, amely technikai értelemben szintetikus küszöbértéknek is tekinthető. Az értékekből arra a következtetésre jutottunk, hogy amikor a komplex wavelet szerkezeti hasonlóság 0,5 alá esett, az detektorok pontossága drámaian csökkent, ugyanis az detektálandó objektum beleolvadt a háttérbe. Az értékelés során megállapítottuk, hogy a detektorok teljesítménye 0,6-os vagy annál magasabb hasonlósági érték esetén már megközelíti a valós fotókon betanított referenciaeredményeket. Ennek oka, hogy 0,5-ös értéknél az objektum már észlelhető módon elválik a háttértől, de az információvesztés még mindig elég jelentős ahhoz, hogy csökkentsék a detektorok teljesítményét. Fontos megjegyezni, hogy annak meghatározása, hogy a YOLO hálózat milyen szinten éri el az eredeti valós képeken mért pontossági értékeket, nagyban függ az észlelendő objektumok részleteitől és a választott detektor robusztusságától. Esetünkben ez az érték 0,6 volt. A lényeg, hogy a fent említett hasonlósági értékek az objektumot meghatározó jellemzők legkisebb elemi részeire teljesüljenek.

Összegzésként elmondható, hogy a bemutatott módszer akkor tekinthető sikeresnek, ha olyan képeket tudunk szintetizálni, amelyeken az objektum régiója jól megkülönböztethető a háttértől.

3.4. táblázat. A képmínőség eredményei a betanítás során szintetikus generált képek esetében.

# Ciklus	CWSSIM				CWSSIM				CWSSIM			
	i	σ	min	max	i	σ	min	max	i	σ	min	max
1	0.6477	7.0878e-4	0.6457	0.6493	0.4085	0.0761	0.2502	0.7217				
2	0.7281	7.6168e-4	0.7259	0.7296	0.4401	0.0868	0.2258	0.7459				
3	0.7576	6.7552e-4	0.7560	0.7591	0.4574	0.0896	0.2282	0.7562				
4	0.7816	6.8852e-4	0.7796	0.7827	0.4653	0.0897	0.2295	0.7578				
5	0.8006	6.6298e-4	0.7990	0.8019	0.4879	0.0933	0.2083	0.7674				
6	0.8113	6.6731e-4	0.8094	0.8128	0.5056	0.0910	0.2252	0.7810				
7	0.8207	5.8425e-4	0.8193	0.8218	0.4967	0.0924	0.2262	0.7662				
8	0.8256	6.2135e-4	0.8242	0.8272	0.5176	0.0872	0.2485	0.7962				
9	0.8282	6.5839e-4	0.8257	0.8297	0.5334	0.0817	0.3025	0.7884				
10	0.8409	5.9780e-4	0.8423	0.8392	0.5779	0.0680	0.3366	0.7789				
20	0.8583	5.9699e-4	0.8594	0.8569	0.6039	0.0651	0.3469	0.7716				
30	0.8687	4.9685e-4	0.8696	0.8679	0.6017	0.0646	0.3631	0.7688				
40	0.8738	5.1662e-4	0.8749	0.8732	0.5986	0.0626	0.3573	0.7862				
50	0.8772	5.1611e-4	0.8782	0.8748	0.6089	0.0659	0.3390	0.8095				
60	0.8784	4.5833e-4	0.8793	0.8792	0.6113	0.0602	0.3962	0.7962				
70	0.8804	4.7160e-4	0.8814	0.8793	0.6099	0.0625	0.3465	0.7810				
80	0.8815	4.5871e-4	0.8823	0.8815	0.6113	0.0601	0.3817	0.7972				
90	0.8809	4.8511e-4	0.8814	0.8797	0.6214	0.0612	0.3857	0.7896				
100	0.8823	3.9526e-4	0.8833	0.8818	0.6214	0.0631	0.3676	0.7809				
140	0.8855	3.5346e-4	0.8833	0.8843	0.6183	0.0582	0.3991	0.7858				
200	0.8827	3.9410e-4	0.8835	0.8817	0.6225	0.0572	0.4041	0.7579				

3.5. táblázat. A detektor eredményei a szintetizált képek minőségétől függően.

CWSSIM obj	YOLOW3-Spatial Pyramid Pooling				YOLO Version3-5l				YOLO Version3-tiny-3l				YOLO Version3-tiny			
	mAP	pontosság	visziahívás	f1-mutató	mAP	pontosság	visziahívás	f1-mutató	mAP	pontosság	visziahívás	f1-mutató	mAP	pontosság	visziahívás	f1-mutató
1,0000e+0	8,4234e-1	9,0031e-1	8,0122e-1	8,5995e-1	9,5985e-1	1,0000e+0	8,8881e-1	9,3158e-1	8,1158e-1	8,3458e-1	8,5401e-1	8,4045e-1	9,1123e-1	9,2423e-1	9,0024e-1	9,1486e-1
4,0851e-1	4,4491e-1	5,0251e-1	5,2021e-1	5,1214e-1	4,8391e-1	6,2475e-1	1,2245e-1	2,1269e-1	8,6136e-2	0,0000e+0	0,0000e+0	0,0000e+0	2,4102e-1	4,6364e-1	3,0051e-1	3,6325e-1
4,4014e-1	4,9163e-1	5,7156e-1	5,2694e-1	5,1556e-1	5,9236e-1	8,2236e-1	3,5698e-1	4,9548e-1	1,9999e-1	5,6458e-1	1,2213e-1	2,0091e-1	3,4772e-1	6,2584e-1	2,0622e-1	3,0192e-1
4,5746e-1	5,6062e-1	7,3112e-1	4,0156e-1	5,2116e-1	6,7912e-1	8,6775e-1	4,7446e-1	6,1106e-1	3,2888e-1	4,2112e-1	4,0025e-1	4,1996e-1	4,4741e-1	5,9156e-1	4,3364e-1	4,9094e-1
4,9677e-1	5,5381e-1	7,7478e-1	2,5569e-1	3,8478e-1	7,5814e-1	8,3446e-1	7,5325e-1	7,9478e-1	4,1861e-1	8,8487e-1	1,7144e-1	2,9487e-1	4,8810e-1	6,7168e-1	3,0658e-1	4,1495e-1
5,1762e-1	5,5664e-1	6,6458e-1	5,7332e-1	6,1569e-1	7,8736e-1	9,6458e-1	5,7021e-1	7,2745e-1	6,2956e-1	9,4032e-1	3,8136e-1	5,4354e-1	6,4872e-1	8,7658e-1	6,5354e-1	7,4569e-1
5,7796e-1	7,5028e-1	7,2323e-1	7,7125e-1	7,5158e-1	9,4694e-1	1,0000e+0	9,0158e-1	9,5458e-1	6,6854e-1	7,2489e-1	6,5158e-1	6,8358e-1	6,4481e-1	8,2125e-1	6,8368e-1	7,4698e-1
6,0397e-1	8,1891e-1	7,8435e-1	8,0618e-1	7,9784e-1	9,1564e-1	9,5264e-1	8,8154e-1	9,1102e-1	7,3612e-1	8,4265e-1	6,5695e-1	7,3348e-1	8,7501e-1	9,2365e-1	8,8354e-1	9,2222e-1
6,2916e-1	7,9335e-1	7,2912e-1	7,8654e-1	7,4145e-1	9,3946e-1	9,2145e-1	8,8245e-1	9,0254e-1	7,8928e-1	8,3489e-1	7,3154e-1	7,7487e-1	8,9501e-1	8,6125e-1	9,0105e-1	8,8182e-1

3.5. Konklúzió

Ebben a munkában az neurális hálózatok tanítása adatszintézis segítségével történt. Az adatszintézis megvalósításához 3D modellezést és kép-kép fordítást alkalmaztunk. A komplex wavelet szerkezeti hasonlósági mérőszámot használtuk a szintetikus generált képek minőségének és hasonlóságának becsléséhez, és ezeket az eredményeket felhasználva tanítottuk be a YOLO detektorokat. Továbbá megtaláltuk azt a küszöbértéket, amelynél a detektorok tanítási pontossága összehasonlíthatóvá válik a referencia képzési pontossággal.

A megoldás lehetővé tette, hogy a mélytanuló neurális hálózatot egy meglévő ipari robotegységre telepítsük anélkül, hogy a vezérlőrendszert teljesen ki kellett volna cserélni, illetve további költségeket kellett volna vállalni.

Első tézispont

Tématerületnek megfelelően kidolgoztam egy saját optimalizált gépi tanulási architektúrát, amely képből-kép fordítási eljárásnak egy újfajta alkalmazását ipari robotika rendszerek számára. A megvalósításhoz a pix2pixHD alapú mélytanuló architektúrát vettem kiindulási alapként, és azon végeztem el architektúrális optimalizálást. A kiértékelés során megállapítottam, hogy az eljárás abban az esetben alkalmazható, ha a renderelt és valós képpárok rendelkezésre állnak. Megállapítottam továbbá azt is, hogy az elfogadható minőségű képszintetizáláshoz legalább 0,5-ös értékű komplex wavelet struktúrális hasonlóság szükséges, melynek értékét elvégzett kísérletek alapján határoztam meg.

Kapcsolódó publikáció:

[J3]. *Kapusi, T.P.; Erdei, T.I.; Husi, G.; Hajdu, A. Application of image translation-based approach in industrial manufacturing system for object identification. Int. Rev. Appl. Sci. Eng., 2025, 1–11. Q2*

4. fejezet

Adatkészlet fordításon alapuló eljárás ipari robotrendszerekben

Ez a fejezet részletesen bemutat egy kidolgozott képi adatkészlet fordítási eljárást, mely szintén alkalmazható ipari gyártási folyamatokban.

Az iparban jelenleg is számos régebbi gépeszköz üzemel, amelyek működési területe modernizációval tovább bővíthető lenne. A gyártósorokon használt robotok mechanikai szempontból alkalmasak különféle megmunkálási és logisztikai feladatok ellátására, korlátaik azonban elsősorban a szoftveres oldalon jelentkeznek. Mivel az Ipar 4.0 egyik alapelve a hálózati alapú kommunikáció és felügyelet, ezen funkciók megvalósítása a régi rendszerekben komoly kihívást jelent [121]. Emellett a mélytanuláson alapuló képfeldolgozási feladatok magas számítási igényeit a hagyományos vezérlők nem képesek kielégíteni. Számos tanulmány foglalkozik a mélytanulás régi robotegységekre történő alkalmazásával.

Az egyik kutatásban a mélytanulás javította egy régebbi ABB IRB2400L robot alapvető funkcionalitását, lehetővé téve a hatékonyabb adatfeldolgozást [122]. Hasonló ipari környezetben neurális hálózatokat integráltak egy Baxter nevű kétkarú robotegységbe, amelyet eredetileg korlátozott érzékelési képességek jellemeztek. A fejlesztéshez létrehozták az F-SIOL-310 adatkészletet, amellyel az objektumfelismerés pontossága tovább javítható volt [123]. A neurális hálózatok alkalmazása a raktározási és logisztikai rendszerek fejlesztésének is egyik központi iránya. Egy tanulmány például bemutatta, hogy egy régi Toshiba Machine VL500 7 DOF

robot textúra nélküli dobozokat válogatott mélytanuló modell segítségével [124].

Bizonyos esetekben, például a Franka Emika Panda orvosi robot esetében [125], a fejlesztés során először szimulációs keretrendszerben tanították be a modellt, mielőtt azt valós környezetben alkalmazták volna. Az ilyen rendszerekben általában látásérzékelők és neurális hálózatok biztosítják a robotok intelligens működését [126]. Többtengelyes robotkarak esetében az objektumok elmozdulásának észlelése hasonló jelentőségű, mint a mobil robotoknál a kitérő manőverek végrehajtása [127]. Fontos különbség azonban, hogy a robotegységek munkaterületén a nyersanyagokat szállító szalagok meghatározott, korlátozott pályán mozognak.

Ebben a fejezetben két különböző alkalmazási területen vizsgálunk anyagmozgatási feladatokat végző futószalagos robotegységeket, egy olyan módszer kidolgozása céljából, amely lehetővé teszi a régebbi gépegységek számára a mélytanuláson alapuló képelemzési feladatok végrehajtását.

Az első vizsgált rendszer egy többtengelyes KUKA KR5 robot [128], amelynek munkaterületén egy Flexlink XK szállítószalag található. A második esetben egy Sony SCARA SRX-611 típusú robotot alkalmaztunk, amely PARO QE 01 31-6000 szállítószalaggal rendelkezik (lásd az 2.6. ábrát).

A megvalósítás során a mintadarabokat megterveztük és legyártottuk 3D-nyomtatás és additív gyártási technológia alkalmazásával. A szállítószalagon áthaladó alkatrészek detektálását a saját, betanított neurális hálózatunk végezte.

Bár a gépi tanulási technikák robusztus és megbízható megoldásokat kínálnak olyan problémákra, amelyek klasszikus algoritmusokkal nehezen vagy kevésbé hatékonyan kezelhetők, a megfelelő adatkészlet továbbra is a tanítási folyamat egyik legkritikusabb eleme. Korábbi munkánkban [102] egy mélytanuláson alapuló neurális hálózat segítségével vizsgáltuk a régebbi robotegységek funkcionalitásának bővítési lehetőségeit, és renderelési módszerekkel olyan tanulási adatkészletet hoztunk létre, amely ésszerű pontosságot eredményezett. A valós képeken végzett vizsgálatok azonban kimutatták, hogy az algoritmusok teljesítménye jelentősen romlik, amikor a valós képek megjelenése nagymértékben eltér a szintetikus mintáktól, illetve amikor a képeken tükröződések, torzulások, változó megvilágítás

vagy színbeli eltérések nagy számban fordulnak elő.

Ennek oka, hogy a képadatok minősége és az adatkészlet információtartalma alapvetően meghatározza a tanítás pontosságát, valamint azt, hogy a modell mennyire képes általánosítható megoldást biztosítani. Esetünkben a mélytanulási algoritmusok tanításához használt képadatkészletet adatszintézissel állítottuk elő, pontosabban képfordításon (image-to-image translation) alapuló szintetizálási technikákkal, a lehető leghízelibb eredmények elérése érdekében. A szintetizált képek információtartalmának és minőségének elemzésével meghatározható az a küszöbérték, amelyet el kell érni ahhoz, hogy a mélytanulási algoritmusok megfelelő pontossággal taníthatók legyenek.

Ezen felül munkánk egy új adatkészlet-átalakítási eljárást is javasol, amely különösen olyan esetekben alkalmazható, amikor az adatkészletek újragenerálása a jövőben nehézségekbe ütközik vagy nem lehetséges. A betanított neurális hálózat futtatásához továbbá meghatároztuk a szükséges számítástechnikai hardverkövetelményeket is.

A munka az alábbiak szerint épült fel. A többszemes robotegységek és szállítószalagok műszaki háttérét a 2.2 szakasz ismerteti. A 4.3 szakasz az adatkészlet-átalakítási módszert ismerteti, a 4.1 pedig az elméleti háttérét és a modellt. A 2.2 szakasz szintén tartalmazza a a házon belül kifejlesztett FDM 3D nyomtatót és a teszt tárgyak nyomtatását. Ezenkívül tárgyalja az adatállomány létrehozását a neurális hálózat valós és renderelt képekkel történő betanításához. A kiválasztott detektorok architektúrájának leírása a 4.4. szakaszban található. A 4.5 szakasz leírja a betanítás során végzett beállításokat és paraméter-módosításokat, a 4.6 összefoglalja a mélytanuláson alapuló algoritmusok eredményeit, az eredményektől függően. A 4.7 szakasz egy valós alkalmazást is bemutat, míg a következtetések a 4.8 szakaszban találhatóak meg.

4.1. A jeleneteket leíró teoretikus modell

Az előző alfejezet alapján (részletekért lásd a 2.3 szakaszt) több jelölést és definíciót vezettünk be. A mi esetünkben egy képjelenet több osztályból és objektumból áll, amelyek tovább vannak osztva megfelelő alrégiókra.

Jelöljön $\Omega = \bigcup_{i=1}^n C_i$ egy olyan geometriai jelenetet, amely n darab objektumosztályt tartalmaz. A C_i képjelenet osztályrégiója felbontható további m_i darab $r_{i,j}$ ($j = 1, \dots, m_i$) alrégiókra:

$$C_i = \bigcup_{j=1}^{m_i} r_{i,j}. \quad (4.1)$$

Ezenkívül minden régióknak van egy saját hozzátartozó anyaga az összes lehetséges $\mathcal{M} = \{m_k : k = 1, \dots, l\}$ anyag közül.

Egy Ω' teoretikus jelenet leírható és egyértelműen definiálható a $(r_{i,j}, m_k)$, $\mathcal{S} = \{(r_{i,j}, m_k) : i = 1, \dots, n; j = 1, \dots, m_i; k = 1, \dots, l\}$ régiók és anyagok összes lehetséges kombinációjának figyelembe vételével.

A fenti jelöléseken és definíciókon alapulva bevezetjük a képek teoretikus modelljét. Jelöljön I egy tetszőleges képet. Ha I reális fényviszonyokat és jelenségeket tartalmaz, akkor a következőképpen fejezhető ki:

$$I = S + L + P + N(0, \sigma_2) \quad (4.2)$$

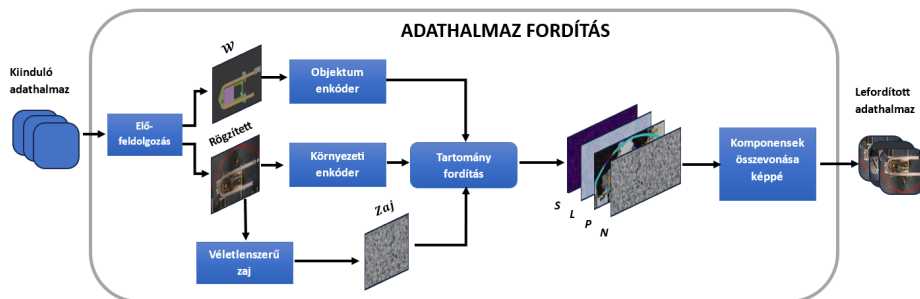
A fenti 4.2 összefüggésben S jelöli a \mathcal{S} raszterizált, perspektívában vetített és transzformált (megjelenített) változatát, L jelenti a a fényerősség és a színhőmérséklet tartalmazó komponens, és P lesz a jelenségre korlátozó komponens, beleértve a visszaverődéseket is. Ezenkívül a modell tartalmazza az N véletlenszerű Gauss-zaj komponens is.

4.2. A javasolt mélytanuláson alapuló képtartomány konverziót megvalósító architektúra

Az elméleti modell szerint (részletekért lásd a 2.3 és a 4.1 szakaszokat) egy képjelenet további főkomponensekre és alrégiókra bontható. Különböző esetekben ezeknek a részeknek a szerkezete túl bonyolult lehet ahhoz, hogy klasszikus megközelítéssel leírható legyen, vagy nem lehet pontosan meghatározni, mert több paraméter nem ismert vagy nem áll rendelkezésre. Ezért robusztus megközelítésre van szükség, hogy automatikusan felfedezzük ezeket a lényeges jellemzőket, és megtanuljuk a megfelelő

régiók és anyagok párosait $((r_{i,j}, m_k))$ az alkalmazás aktuális kontextusában. A mi esetünkben szükséges a képtartomány átalakítása, hogy a valós világ jelenségeinek és anyagainak hatását a renderelt háttérképekre vetítsük.

A javasolt adatkészlet-fordítási módszerünk felépítését és főbb összetevőit a 4.1 ábra mutatja be.



4.1. ábra. Az adatkészlet fordítási módszer architektúrája.

Működése során az algoritmus több lépésben generálja a modellben meghatározott komponenseket, amelyek az adatkészlet átalakításához is elengedhetetlenek. Első lépésként, miután a kezdeti adatkészlet rendelkezésre áll, előfeldolgozásra van szükség. Ez a folyamat címkézett pixelekkal rendelkező geometriai jeleneteket hoz létre, amelyek a kódoló bemeneti adataiként szolgálnak. Az adatkészletben szereplő környezetképekre további előfeldolgozási rutinokat alkalmazunk, mivel ezek többszörös konverziót igényelnek annak érdekében, hogy a kódolók által értelmezhető formátumra és értéktartományra alakíthatók legyenek.

Az automatikus jellemzők kinyerését követően az objektumosztály-kódoló meghatározza az elméleti modellben említett S komponens jellemzőit (lásd 4.1 szakasz). A környezeti kódoló feladata pedig a fényviszonyok és a környezeti fényjelenségek megtanulása lesz (az L és P komponensek leíróinak előállítása). A negyedik komponenst, amely a véletlenszerű zajt képviseli, a véletlenszerű zajkomponens biztosítja, és a mérete megegyezik a bemeneti környezeti kép felbontásával.

Az S , L , P és N komponensek a képtartomány konverziós lépés során jönnek létre. Utolsó lépésként pedig az egyes komponensek egyesítése

után az objektumfelismeréshez és a detektorok betanításához szükséges képadatok halmaza kerül előállításra.

A renderelést követően a lehető legréalisztikusabb képek előállítása kép-tartomány konverzióval valósítható meg. Az ilyen algoritmusok célja a $G : A \mapsto B$ leképezés megtanulása, azaz a forrás- és céltartomány közötti képi jellemzők transzformációja. Ennek megfelelően egy képből–kép fordításon alapuló módszert választottunk, amely képes a tükröződések és más felületi jelenségek vizuális mintázatainak megtanulására.

Mivel rendezett adathalmazzal dolgozunk, és minden képhez tartozik egy megfelelő pár, a párosított tanítási esetet alkalmaztuk. A képből–kép fordítás napjainkban számos mélytanulási architektúrával megvalósítható [1, 110]. Kísérleteinkhez a pix2pixHD alapú felépítést választottuk kiindulási modellként, mivel ez a megközelítés kiváló pontossággal képes fotoreális és nagy felbontású képek generálására. A generálási folyamatot a 4.2. ábra szemlélteti, amely bemutatja, hogyan konvertáljuk a renderelt képeket a detektorok betanításához szükséges képadatbázissá.



4.2. ábra. Képjelenetek tartományfordítása.

Ezenkívül számos kisebb-nagyobb módosítást hajtottunk végre a hálózati architektúra finomhangolása és a képzési eredmény javítása érdekében. Először az elsődleges hálózat kulcsfontosságú elemeit és azok kapcsolatait módosítottuk, hogy azok alkalmazkodjanak a fent említett ipari területhez. Ehhez az 4.1 ábrán bemutatott struktúrát használtuk alapul. Ennek megfelelően optimalizáltuk az elsődleges hálózatban található jellemzőkivonó alhálózatok (objektumosztály- és környezetkódoló) méretét és számát. Fontos megjegyezni, hogy a kódoló által generált jellemzővektorok hossza, amely a hálózat $feature_{num}$ paramétereként fog szolgálni, megegyezik a geometriai jelenetben található különböző objektumosztályok számával. Ezért módosítottuk a jellemzők klaszterméretét is, amelynek értéke megegyezik a geometriai jelenetben található összes objektumosztályhoz

tartozó alrégiók számával.

Ezt követően megdupláztuk a diszkriminátorok számát kettőről négyre, és a generátorban a lefelé mintavételi réteget négyről nyolcra. Emellett kiegészítettük az architektúrát Gauss-zaj komponenssel, ami növelte az eredmények pontosságát. Végül egy robusztus algoritmust építettünk be a tanítási folyamatba, hogy mérjük a szintetizált és a valós képek minőségét és hasonlóságát (a részletekért lásd a 3.4 szakaszt).

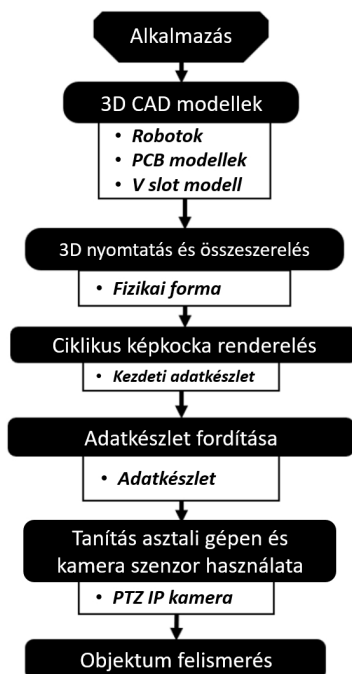
Munkánk során a CycleGAN [110] architektúrát is teszteltük, de mivel a pix2pixHD jelentősen jobb képminőséget eredményezett, csak ezt a módszert használtuk az adatkészletek konvertálására a képek szintézise során.

4.3. Adatkészletek létrehozása a valós és rendelt képekkel történő adatkészlet-fordítási módszer betanításához

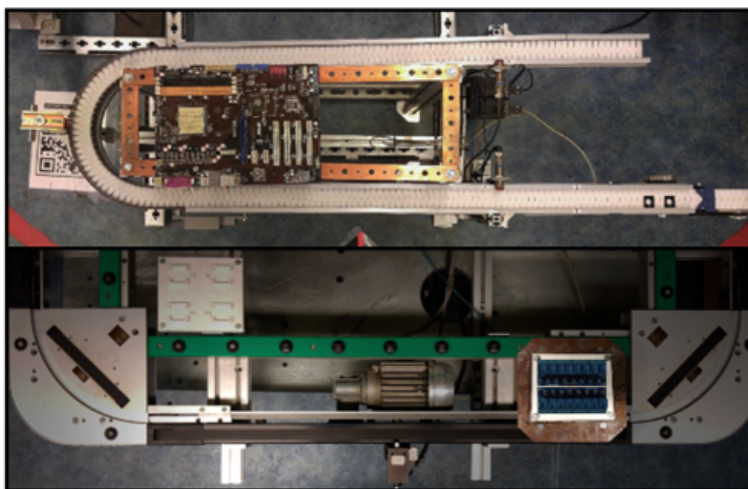
Ez a szakasz részletesen ismerteti az általunk javasolt adatkészlet-fordítási módszer betanításához szükséges adatkészlet létrehozásának folyamatát.

A v_{slot} elemek és a PCB relé táblán alapuló munkadarabok adatbázis-fordítási hálózatának betanításához két adatbázist kell létrehozni. Megoldásunk módszertanát a 4.3 ábra szemlélteti.

Az első adathalmaz, amely a munkadarabok valós képeit tartalmazza, egy Tapo C200 IP [129, 130] kamera telepítésével készült, amelyet a KUKA KR5 ipari robotkar munkaterületére szereltünk fel, és amely a teljes Flexlink XK szállítószalagot látja. A kamera felbontása 1080p Full HD (1920 × 1080 pixel), ami alkalmassá teszi nagy felbontású képek készítésére. A v_{slot} -alapú tesztdarabokat ezután a Flexlink XK szállítószalagra helyeztük. A Flexlink XK szállítószalagot ezután többször állandó sebességgel mozgattuk, miközben a képeket PNG formátumban rögzítettük későbbi értékelés céljából. A PCB -alapú tesztdarabot a Sony SCARA SRX-611 PARO QE 01 31-6000 szállítószalagon használtuk (lásd még a 4.4 ábrát). A fenti folyamatot követően összesen 14 576 kép állt rendelkezésre, amelyeket a Tapo C200 kamera rögzített.



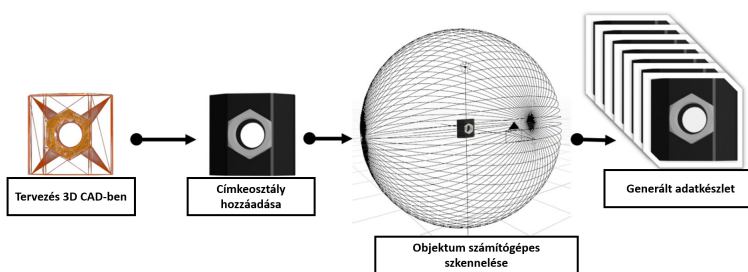
4.3. ábra. A módszertan keretdiagramja.



4.4. ábra. Munkadarabok a Flexlink XK és PARO QE szállítószalagokon.

A második adatkészlet tartalmazza a kezdeti jeleneteket, beleértve az egyes osztályokat és a hozzájuk tartozó alrégiókat. E képek előállítására további lépéseket igényelt.

Amint korábban említettük, a futószalagon mozgó tárgyak 3D CAD-modelljeinek tervezése két fő feladatot foglal magában. Az első a tárgyak tényleges legyártása 3D FDM technológiával, amely a későbbi összehasonlításokhoz szolgáltat referenciaanyagot. Miután a 3D CAD-modellek elkészültek, a következő lépés a neurális hálózat tanításához szükséges adatelemek előállításával digitális rendereléssel (lásd a 4.5. ábrát).



4.5. ábra. Szintetizált adatkészlet létrehozása.

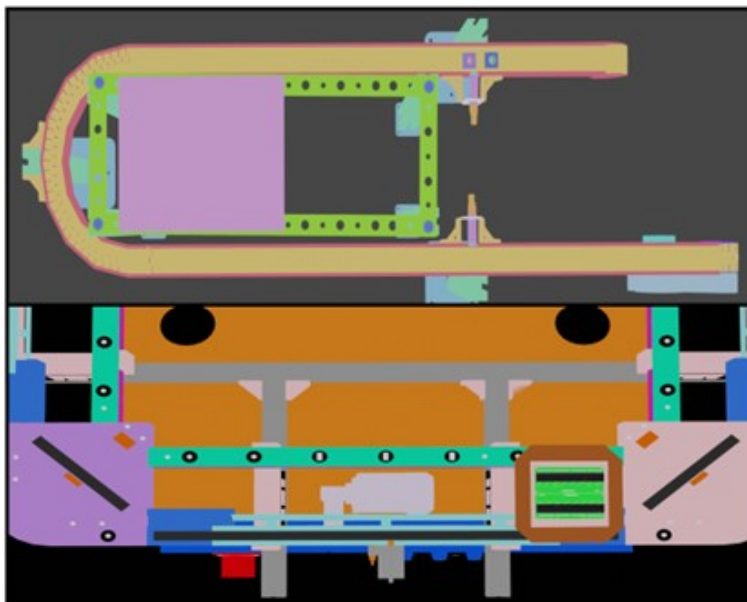
Esetünkben a megvalósítást a Blender szoftver biztosította, amelyet az iparban széles körben alkalmaznak vizualizációs és szimulációs feladatokhoz [131, 132]. A program nyílt forráskódú, és a GNU GPL licenc alatt érhető el [133].

Elkészítettük a Flexlink XK és PARO QE szállítószalagok 3D CAD-modelljeit is, majd poligoncsökkentést hajtottunk végre rajtuk a számítási igény mérséklése érdekében. Ezt követően virtuális kamerát helyeztünk el a 3D munkaterületen, a tesztarabokat pedig animáltuk a szállítószalagok mentén. A jeleneteket végül 1080p (Full HD) felbontásban rendereltük le (lásd a 4.6. ábrát).

Mivel a folyamat 3D CAD-modell alapú renderelésen alapul, az adatkészlethez tetszőleges számú elem előállítható. Ennek eredményeként ugyanannyi, 14 576 renderelt kép készült, mint a valós felvételek esetében.

A valós és a generált képek a fentiekben leírt módon PNG formátumban kerültek mentésre, azonban az objektumok felismeréséhez annotációs folyamatra is szükség volt. Ehhez a LabelImg szoftvert alkalmaztuk [134]. Első lépésben a valódi IP-kamerával rögzített fényképeket annotáltuk, ahol a határolókereteket az adott objektumok osztályneveihez rendeltük (lásd a 4.7. ábrát).

Ezt követően elvégeztük a 3D-ben renderelt képek annotálását is. Az



4.6. ábra. 3D CAD-alapú képalkotás.

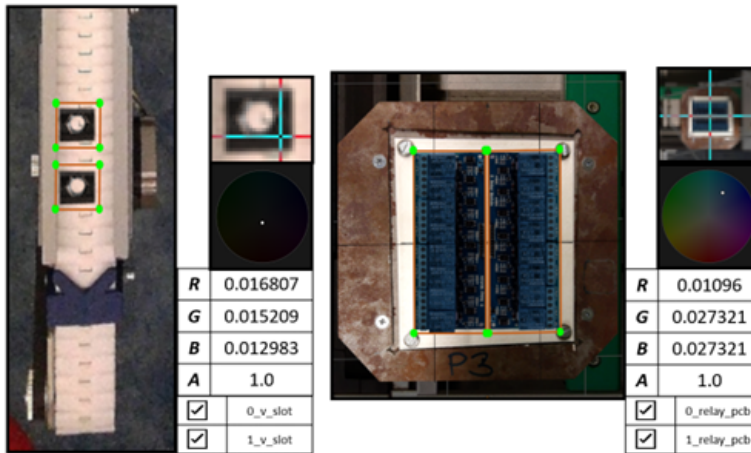
RGB színskála alapján jól látható, hogy a renderelt képek színei konzisztens eloszlást mutatnak, azaz nem tartalmaznak valós textúrákat vagy fényváltásokat. Ezzel két, egymással összevethető adatkészlet állt rendelkezésre a mélytanuláson alapuló algoritmusok betanításához.

A jelölési és renderelési folyamatokat a laboratórium hálózatához csatlakoztatott munkaállomáson végeztük. Az asztali számítógép konfigurációja a következő volt: Intel Xeon W-2123 CPU (4 mag, 8 szál), NVIDIA RTX 4060 Ti GPU és 64 GB RAM. [135].

4.4. A mélytanuláson alapuló detektor algoritmusok architektúrájának leírása

Ebben a szakaszban a robotegységek betanításához használt detektorok felépítését ismertetjük.

A. Bochkovskiy és munkatársai továbbfejlesztették a YOLO architektúrát mozaikadatokkal történő kiegészítéssel, valamint egy új, horgonymentes detektáló fej és egy módosított veszteségfüggvény bevezetésével



4.7. ábra. A valós képek annotációi.

[136]. 2020-ban az Ultralytics kiadta a YOLO egy újabb változatát, a YOLOv5 modellt [93]. Ennek eredményeként a hálózati architektúra jelentősen fejlődött, többek között a hiperparaméter-optimalizálás és az integrált kísérletkövetés funkcióinak beépítésével. Később Chien-Yao Wang és munkatársai [137] bemutatták a YOLOv7 modellt, amely a tanítási segédtechnikák csomagja megközelítésre épül. A modell újraparaméterezési és méretezési technikákat alkalmaz az architektúra pontosságának és sebességének további javítása érdekében. Az Ultralytics azóta további YOLO-változatokat is megjelentetett [93, 138, 139], amelyek új funkciókat és fejlesztéseket integrálnak a teljesítmény, a rugalmasság és a hatékonyság növelésére. Ezek a rendszerek már teljes körű, látásalapú mesterséges intelligencia-feladatokat támogatnak, beleértve az észlelést, a szegmentálást, a pozícióbecslést, a nyomon követést és az osztályozást is.

A fenti munkákat figyelembe véve két fő YOLO architektúrát [94, 114] választottunk ki a detektáláshoz, mivel ezek nagyon hatékonyak és gyorsan megvalósíthatók, és nagyon elfogadható szintű észlelési pontosságot érnek el.

Esetünkben a következő négy különböző altípusra fókuszáltunk az elsődleges beépített detektorok esetében: YOLOv3-tiny, YOLOv3-tiny-3l, YOLOv3-SPP és YOLOv3-5l. Mivel fő célunk az detektorok működőképességének és pontosságának ellenőrzése, nem végeztünk jelentős architektúrabeli módosításokat, hanem az adatszintézis révén létrehozott

adatkészlethez optimalizáltuk a detektorblokkokat. A 2.16 egyenlet alapján az egyes YOLO rétegek előtt álló konvolúciós rétegszűrők számát méreteztük, és eredményképpen 18-at kaptunk, mivel mind a két adatkészlet esetében egy-egy detektálandó osztályunk volt.

Végül minden detektor esetében újraszámoltuk a horgony értékeket a lehető legjobb detektálási pontosság elérése érdekében; ehhez a beépített klaszterezési algoritmust, a k-means ++ [115] algoritmust használtuk. A kapott értékeket a 4.1 táblázat tartalmazza.

4.1. táblázat. Újonnan kiszámított horgony-értékek.

Horgony	YOLOV3-Tiny	YOLOV3-tini-3l	YOLOV3-SPP	YOLOV3-5l
0	10×9	12×7	12×7	10×6
1	13×8	9×10	9×10	10×10
2	10×10	10×10	10×10	9×10
3	11×10	11×10	11×10	10×10
4	12×10	11×10	11×10	11×10
5	14×10	11×10	11×10	11×10
6	–	13×10	13×10	11×10
7	–	14×9	14×9	11×10
8	–	15×11	15×11	13×8
9	–	–	–	12×10
10	–	–	–	11×10
11	–	–	–	12×10
12	–	–	–	13×10
13	–	–	–	14×10
14	–	–	–	15×12

A második esetben a hálózat méretétől és a számítási igényektől függően a következő négy YOLOv8 alapértelmezett detektort választottuk: YOLOv8-nano, YOLOv8-small, YOLOv8-medium és YOLOv8-large.

Mivel a hálózati architektúra több beépített módszert és hiperparaméter-optimalizálást tartalmaz, más optimalizálási technikák és módszerek alkalmazása nem volt szükséges.

4.5. Tanítás és validálás

Ebben a szakaszban bemutatjuk a tanítási és értékelési folyamat során választott tanulási konfiguráció és paraméterek részleteit.

4.5.1. pix2pixHD

A szintetizált képek generálásához használt mélytanulási hálózat tanítási konfigurációját a következőképpen választottuk meg: Optimalizálóként az Adam algoritmust [46] használtuk, 0,0005 kezdeti tanulási sebességgel, amelyet az első 40 tanítási ciklusban változatlanul hagyunk, majd lineárisan csökkentettünk. Az algoritmust összesen 200 tanítási ciklus alatt tanítottuk be 400 kép párra, 2048×1024 pixel felbontással, 16-os tanító-csomag mérettel és pixel címkék nélkül, mivel csak a képfelvételek álltak rendelkezésre. A renderelt képekhez kép párokat hoztunk létre a valós helyszínen, a működés során jellemző fényviszonyok mellett. A tanítás során csak forgatást és tükrözést alkalmaztunk.

4.5.2. YOLO alapú detektorok

Az első lépésben létrehoztuk a kezdeti adatkészletet, amely a tanító, validációs és teszt részekre osztható. A tanító adatkészlet 400 képet tartalmazott, amelyeket szintén a kiválasztott pix2pixHD algoritmussal generáltunk, míg a validálási rész csak 100 valódi, rögzített képből állt. Mindkét esetben a képek felbontása 2048×1024 volt. A teszt adatkészlet a validációshoz hasonlóan szintén 100 valódi, rögzített tesztképből állt.

A YOLO-v3 esetében további adatdúsító algoritmusokat is alkalmaztunk, például a telítettség és az expozíció módosítása, a képek átméretezése és a színárnyalat eltolása [117], hogy megakadályozzuk a tanulási folyamat során a túlzott képzés kialakulását. A megfelelő paraméterekhez tartozó módszerek a 4.2. táblázatban találhatók.

4.2. táblázat. Alkalmazott képjavító algoritmusok és azok paraméterbeállításai.

Algoritmus neve	Paraméter értéke
Szaturáció	1.5
Expozíció	1.5
Átméretezés	1.5
Színeltolás	0.3

Az augmentációs folyamat leírása után az Adam-módszert [116]

választottuk megfelelő optimalizáló algoritmusként. Az Adam momentum- és tanulási ráta paramétereinek beállításait a 4.3. táblázat tartalmazza.

4.3. táblázat. A YOLOv3-alapú detektorok tanulási paramétere.

Detektor típusa	Tanulási ráta	Momentum	Csökkenés
YOLOV3-tiny	0.005	0.9	0.0005
YOLOV3-tiny-3l	0.0005	0.9	0.0005
YOLOV3-SPP	0.0003	0.9	0.0005

Az optimalizáló kiválasztását követően további tanulási paramétereket is konfigurálnunk kellett, például a kezdeti tanulási rátát és az ehhez kapcsolódó ütemezőt, amely minden iterációban dinamikusan módosítja a tanulási értéket. Ebben a munkában exponenciális ütemezőt alkalmaztunk (lásd a 3.1. egyenletet).

Számos kísérletet végeztünk a tanulási ráta és a k paraméter optimális beállításának meghatározására. Ehhez logaritmikus keresést alkalmaztunk a paramétertartományon, amely eredményeként a legjobb érték $k = 1 \times 10^{-1}$ lett. A tanulási ráta legkedvezőbb értékeit a 4.3. táblázat foglalja össze.

Ezt követően további tanítási paramétereket is beállítottunk, többek között a maximális epoch értékét és az ún. burn-in paramétert. Utóbbi a tanítás kezdeti szakaszában biztosítja a stabil konvergenciát, ezért megfelelő megválasztása kulcsfontosságú. Esetünkben a legjobb eredményt minden detektor esetében 100-as érték adta.

Végül meghatároztuk a maximális tanítási iterációszámot a 2.17. egyenlet alapján, amely jelen esetben 2000 lett.

A YOLOv8 detektorok esetében a beépített augmentációs módszereket alkalmaztuk automatikusan beállított paraméterekkel, többek között mozaik-augmentációval és átméretezési technikákkal. A tanítás során használt fő hiperparamétereket az 4.4. táblázat tartalmazza.

4.6. Eredmények

Ez a szakasz ismerteti az összes mélytanuláson alapuló algoritmus eredményeit és az ezekhez tartozó értékelő mutatókat. Emellett elemeztük

4.4. táblázat. A YOLOv8-alapú detektorok tanulási paramétereit.

Paraméter neve	Értéke
kezdeti tanulási érték	0.01
végző tanulási érték	0.001
momentum	0.937
weight_decay	0.0005
warmup_epochs	3.0
warmup_momentum	0.8
warmup_bias_learning_rate	0.1

a valós és a szintetikus adatkészletek közötti különbséget is, amely kulcsfontosságú a detektorok sikeres betanításához. Az algoritmusok tanítását Intel Xeon W-2123 CPU (4 mag, 8 szál), NVIDIA RTX 4060 Ti GPU és 64 GB RAM konfigurációval rendelkező számítógépen végeztük.

4.6.1. pix2pixHD

A kiválasztott pix2pixHD algoritmus eredményeit a 4.5. és 4.6. táblázat foglalja össze, míg a tanítási folyamat során rögzített mintavételi képek a 4.8. ábrán láthatók. A táblázatok tartalmazzák a főbb értékelési mutatókat, azok szórását, valamint az egyes releváns tanítási ciklusokhoz tartozó minimális és maximális értékeket is.

Az értékelés során a komplex hullám szerkezeti hasonlósági mutatót (CWSSIM) [120] választottuk kiegészítő metrikaként, mivel ez a módszer robusztus különféle transzformációkkal – például forgatással vagy átméretezéssel – szemben, és a jelen vizsgálatban különösen hasznosnak bizonyult.

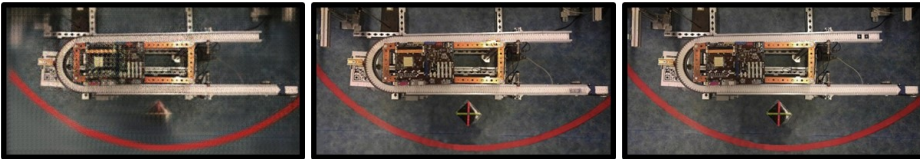
A képek és objektumok méretkülönbségei miatt a hagyományos strukturális hasonlósági (SSIM) módszer alkalmazásakor bizonyos hibák léphetnek fel, különösen az intenzitás- és kisebb geometriai torzulások esetén. Ezek minimalizálása érdekében több előfeldolgozási technikát alkalmaztunk: hisztogramkiegyenlítést, színkorrekciót, valamint geometriai torzuláskorrekciót. Utóbbi különösen fontos a torzításból eredő strukturális eltérések elkerüléséhez, ugyanakkor csak akkor alkalmazható, ha a kamera mátrixa és a lencse paramétereit ismertek.

Ezért az értékelési folyamat során kezelniük kellett ezeket a problé-



Eredmény a tanítás kezdé- Eredmény a tanítás köze- Eredmény a tanítási folya-
tén pénél mat végén

PCB-based Sony SCARA dataset



V slot-based Conveyor dataset

4.8. ábra. A pix2pixHD algoritmus tanítása során szintetizált képsorok kimenetei.

mákat, illetve törekednünk kellett a hatásuk minimalizálására. Az értékelést mind a teljes képekre, mind pedig az észlelendő objektumok régióira kiterjesztettük. A teljes képekre számított minőségi mutatók már a tanítás kezdeti szakaszában is jó pontosságot mutattak, és a tanítás végére az eredmények elérték azt a szintet, amely már megfelelő a detektálási feladatokhoz. Mivel a detektorok tanítása ezeken a régiókon alapul, ezek az értékek különösen relevánsak a valós és a szintetizált adatkészletek közötti maximális különbség meghatározásában, valamint a kívánt képminőség és hasonlóság elérésében. Az adatkészlet szórásértékeinek, illetve a legjobb és legrosszabb eredményeinek elemzése azt mutatta, hogy a teljes képek között csak minimális eltérések figyelhetők meg, míg az objektumrégiók esetében a különbségek jóval nagyobbak. Ez utóbbi esetben a magasabb és alacsonyabb minőségű régiók aránya közvetlenül befolyásolja a detektorok tanítási folyamatának pontosságát, vagyis a regionális minőségi ingadozások jelentős hatást gyakorolhatnak a végső teljesítményre.

4.6.2. YOLO alapú detektorok

A kiválasztott YOLO-alapú detektorok eredményeit a megfelelő mért minőségi értékekkel (pontosság, átlagos pontosság, visszahívás és F1-pontszám) a 4.7. és a 4.8, 4.9. táblázatokban foglaltuk össze (mAP50 az 0,5-ig terjedő átlagos pontosságot, mAP50-95 pedig a 0,5 és 0,95 között kapott eredményeket jelöli)

Referenciaként a detektorokat valós képeken is betanítottuk; ezek eredményei a táblázat első sorában találhatóak. A generált képadatbázis minden esetben 256 000 képet tartalmazott, míg a validációs adathalmaz 400 valós felvételtől állt.

A YOLOv3 detektorok esetében megfigyeltük, hogy a pontosság jelentősen csökken, amikor a strukturális hasonlóság 0,5 alá esik. Ez alapján meghatározható a valós és a szintetizált adatkészletek közötti maximális eltérés, amelyet a YOLO-alapú detektorok tanítása során figyelembe kell venni. Az értékelés továbbá kimutatta, hogy 0,6-os vagy annál nagyobb hasonlósági érték mellett a detektorok teljesítménye már megközelíti a valós képeken tanított modellek eredményeit.

Az objektumrégiók szórása szintén meghatározó tényező: amennyiben az adatkészlet nagyobb arányban tartalmaz alacsonyabb minőségű régiókat, a detektorok pontossága érzékelhetően romlik.

A YOLOv8 detektorok esetében az eredmények azt mutatták, hogy az architektúra robusztusabb a YOLOv3-hoz képest. Bár a hálózat gyengébb minőségű szintetikus képeken is megfelelő teljesítményt ért el, az átlagos pontosság hasonló mértékű csökkenést mutatott a hasonlóság csökkenésével, mint az előző esetben.

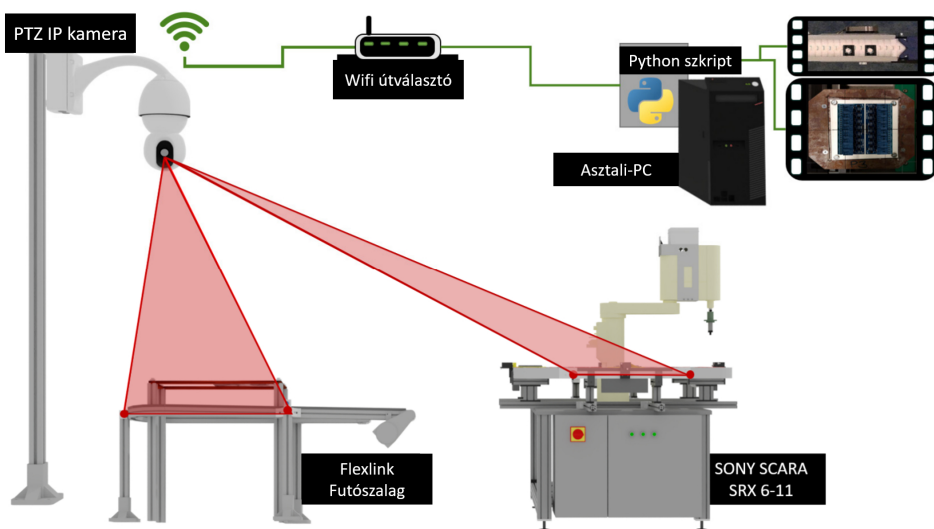
Összességében megállapítható, hogy a képek közötti strukturális hasonlóság mértéke közvetlenül befolyásolja a tanítási folyamat pontosságát, és kulcsszerepet játszik a szintetikus és valós adatok közötti átjárhatóság biztosításában.

4.7. A valós alkalmazás bemutatása

A betanított detektorokkal valós tesztek is végrehajtottunk több eszközön, például egy Nvidia RTX 4060Ti GPU kártyával felszerelt asztali

számítógépen és egy Nvidia Jetson Nano fejlesztői készleten [140].

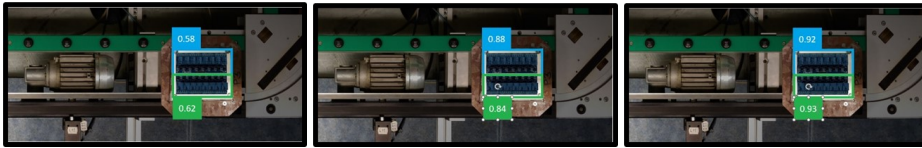
A megvalósított tesztkörnyezetünket a 4.9. ábra szemlélteti. A különböző képzési adatkészletek alapján kapott eredményeket pedig a 4.10. táblázat tartalmazza. Fontos megjegyezni, hogy mind a két fő típusú adatkészlet esetében tanítási, validációs és teszt részekre is kiértékeljük az eredményeket. A tesztek elvégzéséhez a következő két detektort választottuk ki számítási követelményeik és pontossági képességeik alapján: YOLOv3-tiny-3l a szállítószalag-adatkészlethez és YOLOv8-nano a Sony SCARA-adatkészlethez. A kapott eredmények alapján a szintetikus generált adatkészleteken tanított detektorok közel olyan pontos észleléseket érhetnek el, mint a valós képeken képzettek (lásd a 4.10. ábrát).



4.9. ábra. A megvalósított tesztkörnyezet.

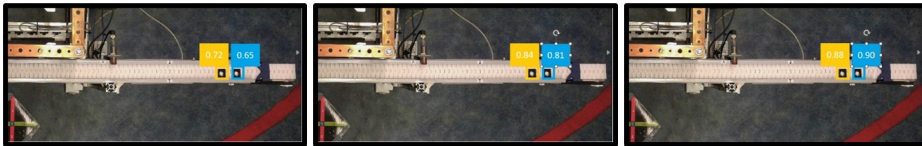
A tesztelés során mindkét eszközön megmértük a feldolgozás idejét is. A YOLOv3-tiny-3l esetében körülbelül 54,19 milliszekundumot, a YOLOv8-nano esetében pedig 76,29 milliszekundumot kaptunk, ami az Nvidia Jetson eszközön elfogadható eredménynek számít. Az észlelőket egy asztali RTX 4060Ti grafikus kártyán futtatva átlagosan 27,87 és 73,11 milliszekundumot értünk el.

Végül megemlítjük, hogy a javasolt adatkészlet-átalakítási architektúra más esetekben is alkalmazható, például egy régi képadatkészlet újbóli konvertálásakor. Ezenkívül a képeken az objektumok textúrájának cseréje



Kizárólag renderelt jelene- Szintetizált képeken beta- Kizárólag valós felvétele-
teken betanított. nított. ken betanított.

PCB-based Sony SCARA dataset



V slot-based Conveyor dataset

4.10. ábra. A pix2pixHD algoritmus tanítása során szintetizált képhalmazok kimenetei.

is lehetségessé válik. Ezekben az esetekben három fő lehetőség áll rendelkezésre. Az első a legegyszerűbb és leggyorsabb megoldás, de csak akkor alkalmazható, ha ugyanazokkal a kamera paraméterekkel és beállításokkal tudunk új képeket készíteni, és ha a jelenetek fő objektumainak elrendezése megegyezik a korábbiakkal. Az objektumok határainak és méreteinek is pixelpontosnak kell lenniük. A legtöbb esetben ez a megoldás hasznos a régi adatkészlet képeinek anyagainak és fényviszonyainak cseréjében. A második esetben további lépéseket kell tennünk.

Ha az első lehetőségben említett feltételek nem teljesülnek, előfeldolgozást kell végrehajtanunk a képzési adatok, beleértve a címkézett pixeleket is, generálásához. Az előfeldolgozás automatikusan elvégezhető klasszikus és gépi tanulási algoritmusok segítségével, az alkalmazási terület komplexitásától függően. Számos klaszterezési vagy szegmentálási algoritmus is hasznos lehet a szükséges címkézési térképek létrehozásában az adatfordítási módszer betanításához.

A harmadik esetben az előfeldolgozási algoritmusok helyettesíthetők vagy kombinálhatók kézzel készített módszerekkel vagy manuális annotációs folyamattal az osztályrégiók létrehozásához. Ezt a megoldást csak akkor javasoljuk, ha az első kettő nem alkalmazható, és az előfeldolgozás nem

ad pontos eredményeket. Ezenkívül elengedhetetlen, hogy az adatkészlet ne legyen túl bonyolult vagy nagy, mivel ez a folyamatok igen időigényes lehet.

4.8. A fejezet konklúzió

Az adatok szintéziséhez kép-kép fordítást és 3D modellezést alkalmaztunk. A szintézált képek hasonlóságát és minőségét komplex wavelet szerkezeti hasonlósági mutatóval értékeltük, és ezeknek az értékeknek az alapján végeztük el a YOLO detektorok képzését. Ennek eredményeként kifejlesztettünk egy képadatbázis-fordítási módszert, amely kép-kép fordítási technikát alkalmazott új adatbázisok létrehozására. Mérőszámok tekintetében elértük azt a minőségi szintet, amelyen az érzékelő-tanítás pontossága megközelíti a referencia-tanítás pontosságát. A megoldás lehetővé tette számunkra, hogy mélytanuláson alapuló neurális hálózatokat alkalmazzunk egy már meglévő ipari robotegységre és a hozzá tartozó adatkészletekre anélkül, hogy teljesen kicseréltük volna a vezérlőrendszerét, és további költségek nélkül. A következő második tézispont mutatja be a kapott eredményeket:

Második tézispont

Kidolgoztam egy újfajta képi adatkészlet fordító eljárást, amely képes az adott gyártási környezetet leíró virtuális modell alapján valósághű képfelvételek szintetizálására. Figyelembe véve a szintetizált képi adathalmaz információmennyiségének hasonlóságát a valós felvételekhez képest megállapítottam, hogy legalább 0,6-os értékű egyezőségre van szükség a mélytanuló detektorok szinte teljesen azonos pontosságú működéséhez, az érték megállapítása elvégzett kísérletek alapján került meghatározásra.

Kapcsolódó publikáció:

[J2]. Erdei T.I.; **Kapusi, T.P.**; Hajdu, A.; Husi, G. *Image-to-Image Translation-Based Deep Learning Application for Object Identification in Industrial Robot Systems*. *Robotics*, 2024, **13**, 88. <https://doi.org/10.3390/robotics13060088> Q1

4.5. táblázat. A pix2pixHD algoritmus minőségi eredményei a szállítószalag-adatkészlet alapján.

epochs	CWSSIM _{Image}	$\sigma_{CWSSIM_{Image}}$	minCWSSIM _{Image}	maxCWSSIM _{Image}	CWSSIM _{Obj}	$\sigma_{CWSSIM_{Obj}}$	minCWSSIM _{Obj}	maxCWSSIM _{Obj}
1	0.6134	9.5687×10^{-5}	0.6111	0.6156	0.3845	0.0347	0.1245	0.6289
2	0.6826	9.2365×10^{-4}	0.6806	0.6843	0.4005	0.0548	0.1869	0.6902
3	0.7254	8.9854×10^{-4}	0.7224	0.7279	0.4254	0.0654	0.2143	0.7216
4	0.7389	9.7564×10^{-4}	0.7368	0.7397	0.4493	0.0458	0.2110	0.7304
5	0.7826	8.7052×10^{-4}	0.7814	0.7843	0.4647	0.0599	0.2440	0.7477
6	0.8053	8.9652×10^{-4}	0.8034	0.8064	0.4886	0.0321	0.2312	0.7507
7	0.8115	7.4652×10^{-4}	0.8103	0.8131	0.4931	0.0245	0.2408	0.7633
8	0.8149	7.3254×10^{-4}	0.8121	0.8157	0.5026	0.0458	0.2655	0.7707
9	0.8204	6.7478×10^{-4}	0.8194	0.8212	0.4916	0.0501	0.2887	0.7798
10	0.8349	7.6548×10^{-4}	0.8324	0.8361	0.5134	0.0546	0.3247	0.7925
20	0.8353	6.2248×10^{-4}	0.8331	0.8369	0.5495	0.0496	0.3335	0.7811
30	0.8469	6.0654×10^{-4}	0.8447	0.8483	0.5766	0.0512	0.3469	0.7761
40	0.8579	5.6335×10^{-4}	0.8560	0.8591	0.6024	0.0564	0.3504	0.7848
50	0.8622	4.4256×10^{-4}	0.8603	0.8634	0.5889	0.0524	0.3475	0.7991
60	0.8676	5.2145×10^{-4}	0.8658	0.8689	0.6124	0.0530	0.3664	0.7948
70	0.8706	5.0125×10^{-4}	0.8692	0.8714	0.6065	0.0535	0.3586	0.7890
80	0.8765	4.7879×10^{-4}	0.8751	0.8783	0.6248	0.0509	0.3496	0.8122
90	0.8789	4.4575×10^{-4}	0.8762	0.8798	0.6424	0.0496	0.3789	0.8046
100	0.8795	4.1254×10^{-4}	0.8774	0.8909	0.6349	0.0524	0.3864	0.7899
140	0.8802	3.8873×10^{-4}	0.8789	0.8814	0.6401	0.0596	0.3941	0.7943
200	0.8815	3.7998×10^{-4}	0.8801	0.8832	0.6578	0.0552	0.4229	0.7873

4.6. táblázat. A Sony SCARA adatkészleten alapuló pix2pixHD algoritmus minőségi eredményei.

epochs	CWSSIM _{Image}	$\sigma_{CWSSIM_{Image}}$	minCWSSIM _{Image}	maxCWSSIM _{Image}	CWSSIM _{Obj}	$\sigma_{CWSSIM_{Obj}}$	minCWSSIM _{Obj}	maxCWSSIM _{Obj}
1	0.7542	4.2548×10^{-3}	0.7514	0.7586	0.4348	0.0601	0.3621	0.5408
2	0.7945	3.9547×10^{-3}	0.7926	0.7914	0.4963	0.0569	0.4090	0.6140
3	0.8002	3.6985×10^{-3}	0.7975	0.8042	0.5269	0.0354	0.4356	0.6432
4	0.8239	2.9645×10^{-3}	0.8210	0.8251	0.5048	0.0487	0.4541	0.6892
5	0.8477	2.4123×10^{-4}	0.8453	0.8490	0.5369	0.0369	0.4802	0.7013
6	0.8914	2.2658×10^{-4}	0.8902	0.8932	0.6087	0.0578	0.5402	0.7274
7	0.9001	1.5478×10^{-3}	0.8982	0.9024	0.6396	0.0469	0.5504	0.7364
8	0.9057	1.9625×10^{-4}	0.9034	0.9076	0.6498	0.0398	0.5666	0.7559
9	0.9111	1.4785×10^{-4}	0.9099	0.9127	0.7069	0.0352	0.5869	0.7624
10	0.9159	9.6582×10^{-4}	0.9135	0.9167	0.6874	0.0245	0.5764	0.7318
20	0.9209	7.6548×10^{-4}	0.9188	0.9213	0.7369	0.0269	0.6415	0.8236
30	0.9238	6.1458×10^{-4}	0.9220	0.9251	0.7846	0.0210	0.6592	0.8735
40	0.9293	6.6548×10^{-4}	0.9287	0.9321	0.7569	0.0289	0.6840	0.8833
50	0.9359	5.2145×10^{-4}	0.9340	0.9372	0.7789	0.0369	0.7248	0.9103
60	0.9448	5.9658×10^{-4}	0.9423	0.9465	0.8093	0.0203	0.7865	0.9245
70	0.9506	5.4568×10^{-4}	0.9485	0.9526	0.8309	0.0326	0.8249	0.9143
80	0.9548	5.1524×10^{-4}	0.9534	0.9564	0.8419	0.0454	0.8351	0.9351
90	0.9627	4.9854×10^{-4}	0.9604	0.9643	0.8829	0.0438	0.8517	0.9469
100	0.9743	4.5230×10^{-4}	0.9719	0.9754	0.9024	0.0334	0.8631	0.9514
140	0.9772	4.2530×10^{-4}	0.9751	0.9792	0.9110	0.0286	0.8724	0.9561
200	0.9772	6.1205×10^{-4}	0.9709	0.9744	0.9269	0.0295	0.8997	0.9569

4.7. táblázat. A YOLOv3-alapú detektorok teljesítményének összehasonlítása a $CWSSIM_{obj}$ függvényében.

$CWSSIM_{obj}$	YOLOv3-tiny			YOLOv3-tiny-3l			YOLOv3-SPP			YOLOv3-5l						
	mAP	Pont.	ViSSzah.	F1-mutató	mAP	Pont.	ViSSzah.	F1-mutató	mAP	Pont.	ViSSzah.	F1-mutató	mAP	Pont.	ViSSzah.	F1-mutató
1.0000	0.9065	0.91	0.90	0.91	0.8468	0.85	0.87	0.86	0.8778	0.89	0.85	0.87	0.9296	1.00	0.94	0.96
0.3845	0.3145	0.44	0.25	0.29	0.1221	0.03	0.00	0.00	0.3949	0.42	0.40	0.43	0.4501	0.55	0.20	0.33
0.4005	0.3698	0.49	0.26	0.35	0.1469	0.26	0.05	0.15	0.4211	0.49	0.45	0.50	0.5068	0.67	0.40	0.54
0.4254	0.3846	0.52	0.22	0.34	0.3986	0.41	0.24	0.43	0.4801	0.52	0.47	0.49	0.5698	0.75	0.38	0.54
0.4931	0.4865	0.58	0.35	0.45	0.4425	0.56	0.40	0.46	0.5259	0.56	0.54	0.56	0.6421	0.81	0.49	0.65
0.5026	0.5469	0.62	0.56	0.57	0.5685	0.60	0.48	0.52	0.5458	0.63	0.61	0.63	0.7248	0.86	0.69	0.71
0.5134	0.5694	0.76	0.62	0.69	0.6405	0.64	0.54	0.58	0.6846	0.71	0.69	0.68	0.8694	0.89	0.72	0.79
0.5495	0.6485	0.78	0.73	0.75	0.6381	0.67	0.62	0.63	0.7954	0.81	0.78	0.79	0.8896	0.92	0.76	0.83
0.6401	0.8069	0.84	0.81	0.82	0.7954	0.81	0.77	0.78	0.8305	0.85	0.82	0.84	0.9100	0.96	0.84	0.91

4.8. táblázat. Results of YOLOv8-Nano and YOLOv8-Small detectors (Sony SCARA dataset).

CWSSIM ₀	YOLOv8-Nano					YOLOv8-Small					
	Pont.	Visszah.	mAP	mAP ₅₀	mAP ₅₀₋₉₅	Pont.	Visszah.	mAP	mAP ₅₀	mAP ₅₀₋₉₅	
1.000	0.999	1.000	0.995	0.835	0.999	1.000	0.995	0.843	0.993	1.000	0.995
0.4348	0.897	0.720	0.723	0.543	0.893	0.651	0.705	0.551	0.734	0.710	0.728
0.4963	0.880	0.860	0.884	0.613	0.896	0.775	0.731	0.667	0.778	0.759	0.796
0.5269	0.886	0.902	0.846	0.658	0.896	0.849	0.793	0.836	0.735	0.860	0.746
0.6396	0.898	1.000	0.896	0.712	0.928	0.990	0.825	0.746	0.889	0.890	0.882

4.9. táblázat. Results of YOLOv8-Medium and YOLOv8-Large detectors (Sony SCARA dataset).

CWSSIM ₀	YOLOv8-Medium						YOLOv8-Large								
	Pont.	Visszah.	mAP	mAP ₅₀	mAP ₅₀₋₉₅	Pont.	Visszah.	mAP	mAP ₅₀	mAP ₅₀₋₉₅	Pont.	Visszah.	mAP	mAP ₅₀	mAP ₅₀₋₉₅
1.000	0.836	0.998	1.000	0.995	0.845	0.998	1.000	0.995	0.845	0.998	1.000	0.995	0.845	0.995	0.845
0.4348	0.455	0.871	0.443	0.691	0.433	0.607	0.721	0.491	0.730	0.440	0.440	0.491	0.730	0.440	0.440
0.4963	0.653	0.779	0.525	0.710	0.607	0.681	0.889	0.884	0.703	0.598	0.598	0.884	0.703	0.598	0.598
0.6896	0.786	0.968	0.955	0.895	0.743	0.797	0.979	0.979	0.925	0.784	0.784	0.979	0.925	0.784	0.784
0.9187	0.827	0.998	1.000	0.995	0.814	0.827	0.998	1.000	0.995	0.814	0.827	0.998	0.995	0.814	0.814

4.10. táblázat. A különböző tanítási adatkészletekkel kapcsolatos detektálási eredmények.

Tanuló adatkészlet	Conveyor adatkészlet			Sony SCARA adatkészlet		
	Tanító.	Val.	Teszt	Tanító	Val.	Teszt
Csak renderelt	0.624	0.612	0.511	0.689	0.674	0.655
Szintetizált	0.795	0.784	0.778	0.802	0.782	0.775
Valós	0.847	0.823	0.814	0.835	0.822	0.802

5. fejezet

Szintetikus tanuláson alapú komponens–slot hozzárendelés

A fejezet egy olyan módszert mutat be, amely szintetikus adatok alapján képes megtanulni a komponensek és a foglalatok közötti hozzárendelést. Az adatszintézis moduláris módon történik, amelyet az Nvidia által fejlesztett Omniverse környezetben valósítunk meg. Összességében a bemutatott rendszer hozzájárul az ipari robotika kutatásához azzal, hogy új módszert kínál a komponens-helykiosztási probléma megoldására. A szintetikus adatgenerálás, a permutációinvariáns modellarchitektúra és a speciális veszteségfüggvények kombinált használata lehetővé teszi, hogy a SCARA robotok tanulásalapú vezérlése közelebb kerüljön a valódi ipari alkalmazásokhoz.

Az ipari robotok alkalmazása elengedhetetlenül fontos szerepet játszik a modern elektronikai gyártásban, különösen az összeszerelési folyamatokban. A SCARA (Selective Compliance Assembly Robot Arm) robotokat már régóta használják olyan feladatokhoz, mint az alkatrészek elhelyezése nyomtatott áramkörtáblák (PCB-k) lapokon, ahol nagy pontosság és ismételhetőség szükséges. A hagyományos programozási és szabályalapú vezérlési megközelítések azonban gyakran korlátokba ütköznek a változó számú alkatrészek kezelésének nehézségei, a zajos érzékelésből adódó bizonytalanságok és az összeszerelési sorrendek optimalizálása miatt. Ennek eredményeként a gépi tanulás és a mélytanuláson alapuló módszerek egyre nagyobb szerepet kapnak.

Az elmúlt években számos tanulmány vizsgálta a vizuális felismerés és a robotkar vezérlés közötti kapcsolatot, de kevés figyelmet fordítottak a komponensek és a helyek hozzárendelésének problémájára általánosított neurális hálózatok segítségével. Ez a feladat kombinatorikus jellegű: egy komponensek és helyek halmazának adott esetben a cél egy 1:1 hozzárendelés elérése, amely biztosítja, hogy minden komponens a megfelelő helyre kerüljön. A kihívás különösen összetett, ha a bemenet irreleváns vagy helytelen detektálásokat tartalmaz, és a komponensek sorrendje nincs előre meghatározva.

5.1. Releváns publikációk

Manapság a szintetikus adatgenerálás egyre nagyobb figyelmet kap a robotika és a gépi látás területén. Az úgynevezett tartomány véletlenszerűsítésén alapuló megközelítés [90] lehetővé teszi a neurális hálózatok számára, hogy szimulált környezetből tanuljanak, majd a valós világban is megbízhatóan működjenek. Ez kapcsolódik olyan ipari eszközökhöz, mint az NVIDIA Omniverse [2] és a Replicator [141], amelyek szabályalapú és fizikai validációt használnak a valósághű jelenetek létrehozásához. Kaigom E.G áttekintette, hogyan alkalmazhatók a metaverse és a digitális iker technológiák robotizált ipari környezetben [142]. A szerző hangsúlyozta, hogy a virtuális szimuláció és a mesterséges intelligencia integrációja hatékony eszköz lehet a robotok betanításához, a gyártási folyamatok optimalizálásához és az ember-gép együttműködés javításához. A cikk kifejezetten foglalkozik az Ipar 4.0 automatizálási fókuszával és az Ipar 5.0 emberközpontú, fenntarthatóságra orientált megközelítésével.

Pasanisi D. és társai bemutatták, hogyan lehet a tartomány randomizációt felhasználni szintetikus képek ipari objektumfelismerési feladatokra való alkalmazásához [143]. Két valós ipari alkalmazás segítségével szemléltetik, hogy a szintetikus képeken tanított modellek hogyan képesek hatékonyan felismerni az objektumokat valós környezetben. A CAD-modellekből generált szintetikus adatok jelentősen csökkenthetik a szimuláció és a valóság közötti különbséget, miközben minimalizálják az annotáció szükségességét is.

A komponens–foglatat párosítás a klasszikus hozzárendelési problé-

mán alapul, amelyet a Magyar-algoritmus optimálisan old meg [144]. Yu és társai egy gráfillesztési modellt mutattak be, amely csatornától független beágyazási réteggel és egy új típusú veszteségmechanizmussal működik [145]. A bemutatott megközelítésben a Magyar-algoritmus eredményeként egy Z hozzárendelési maszkot állítunk elő, amely a megjósolt és a valós párok súlyozását végzi. Ennek hatására a veszteségfüggvény nagyobb hangsúlyt helyez a helytelen illesztésekre, míg a nagy bizonyossággal helyes párok kisebb mértékben befolyásolják a tanulást.

Az elmúlt években megjelentek a differenciálható hozzárendelési módszerek, amelyek lehetővé teszik a neurális hálózatok közvetlen betanítását erre a feladatra. Garcia-Najera A. és munkatársai genetikusan alapuló megközelítést mutattak be a nyomtatott áramköri lapok összeszerelésének optimalizálására [146]. A módszer egyszerre oldja meg a komponensek helyének hozzárendelését és a felvétel-elhelyezés sorrendjének meghatározását, ezáltal minimalizálva az összeszerelési időt és költségeket. Az eredmények megerősítik, hogy az evolúciós optimalizálás hatékony alternatívát kínál a hagyományos ütemezési és hozzárendelési stratégiákhoz képest. Ahmadi R. és munkatársai a moduláris elhelyező gépeken történő optimális alkatrész-elosztás problémáját vizsgálták [147]. A szerzők kombinatorikus optimalizációs modellt és heurisztikus megoldást javasolnak, amely hatékonyan elosztja az alkatrészeket a gépmodulok között, minimalizálva ezzel a gyártási ciklusidőt. Az eredmények megerősítik, hogy a megfelelő alkatrész-elosztás közvetlenül növeli a termelékenységet és csökkenti az összeszerelési költségeket.

Crama Y. és társai a nyomtatott áramköri lapok összeszerelésének egyik legfontosabb feladatát, az alkatrészsadagolókat optimális elosztását tárgyalták [148]. A szerzők matematikai modelleket és algoritmusokat fejlesztettek ki a ciklusidő minimalizálása érdekében, és bebizonyították, hogy a megfelelő adagoló-elosztás jelentősen javítja a termelékenységet. Ez a munka egyértelmű példája annak, hogyan lehet a kombinatorikus optimalizálást felhasználni a PCB-gyártási folyamatok hatékonyságának növelésére. Li W. és társai bemutattak egy mélytanuláson alapuló AR-rendszert, amely valós idejű útmutatással támogatja a komplex kézi szerelési folyamatokat [149]. A rendszer mélytanulást használ az alkatrészek azonosítására és nyomon követésére, majd kiterjesztett valóságban vizuális utasításokat je-

lenít meg a munkások számára. Az eredmények azt mutatják, hogy az AR-alapú segítség jelentősen csökkenti a hibaarányt és növeli a szerelés hatékonyságát.

A transzformer-alapú modellek forradalmasították a szekvenciától független és halmazalapú bemenetek feldolgozását. Lee és társai kifejezetten ilyen problémákra tervezték, és jól alkalmazható a komponensek és a helyek közötti kapcsolatok tanulására [150]. Zhou W. és társai bemutattak egy Set Transformer és Knowledge-Assisted Network kombinációját acéllemezek hibáinak észlelésére [151]. A módszert kiegészíti egy jellemzők fontosságának elemzése, amely javítja az értelmezhetőséget és a pontosságot. Az eredmények megerősítik, hogy a javasolt megközelítés az ipari minőség-ellenőrzés terén felülmúlja a hagyományos neurális hálózatokat.

A DETR hasonló elveken alapul, és az objektumfelismerést feladatként fogalmazza meg [152]. A tanítás során a predikciókat egy Magyar alapú eljárás segítségével illesztik a valós objektumokhoz, amellyel meghatározzák, hogy melyik predikció melyik valós objektumhoz tartozik (egy-egy illesztés), majd erre alapozva számolják ki a veszteséget (osztályozási és keretező doboz veszteség). A H-DETR modell továbbfejleszti a DETR eredeti „egy-egyhez” illesztési stratégiáját: kiegészítésként bevezet egy egy-többhez segédágot a tanítási fázisban [153]. Ez lehetővé teszi, hogy több előrejelzés is tanulási jeleket kapjon ugyanazon valós objektumra, miközben megőrzi a végpontok közötti, egyedi hozzárendelési tulajdonságot a következtetés során.

A klasszikus inverz kinematikai algoritmusok korábban fontos szerepet játszottak a robotkarok mozgásának vezérlésében, de a közelmúltban megjelentek a neurális hálózatokon alapuló megközelítések, amelyek képesek közvetlenül tanulni a mozgásokat szintetikus vagy valós adatokból. Számos tanulmány alkalmazta a megerősítő tanulást vagy a felügyelt tanulási módszereket különböző felvevő-elhelyező és összeszerelő feladatokra, amelyeket gyakran szimulációs környezetben validáltak.

Alonso C. és társai [154] átfogó áttekintést adtak a robotmanipulátorok inverz kinematikájáról, vezérléséről és pályatervezési megközelítéseiről. A szerzők rendszerezik a mély neurális hálózatokon alapuló módszereket, összehasonlítják azokat a klasszikus numerikus algoritmusokkal, kiemelve a tanuláson alapuló megközelítések előnyeit és hátrányait. A cikk az

akadálymentes környezetben használt modelleket, valamint az akadályokat figyelembe vevő vezérlési stratégiákat tárgyalja, ahol a környezet komplexitása további kihívásokat jelent. Ezenkívül foglalkoznak a DNN-alapú megoldások főbb kihívásaival, mint például az általánosítási képesség, a számítási költségek és a biztonságos működés. A áttekintés egyértelműen mutatja, hogy a neurális hálózatok által támogatott IK és mozgástervezés ígéretes irányt jelent az ipari robotika jövőbeli fejlesztései számára.

Shihab M. és munkatársai kísérleti eredményeket mutattak be, amelyek a neurális hálózatokkal támogatott inverz kinematika alkalmazását vizsgálják különböző robotmanipulátorokon [155]. A szerzők rámutattak, hogy a mély neurális hálózatokon (DNN) alapuló megközelítések gyorsabb előrejelzést és jobb általánosítást biztosítanak, mint a klasszikus numerikus módszerek. Eredményeik megerősítik, hogy a mélytanulás ígéretes alternatívát kínál a nemlineáris munkaterületeket igénylő, komplex ipari robotikai feladatok megoldására.

Kovalchuk V. és társai megvizsgálták a mesterséges neurális hálózatok használatát a SCARA robotok inverz kinematikai problémáinak megoldásában [156]. A szerzők elemzik a különböző adatkészletek és optimalizáló algoritmusok hatását a hálózat pontosságára és konvergenciájára, és bemutatják, hogy a megfelelő kombinációval az ANN pontos és hatékony alternatívát nyújt a klasszikus IK-módszerekhez képest. Az eredmények rávilágítanak arra, hogy a tanulási stratégia kulcsfontosságú szerepet játszik a SCARA robotok vezérlési feladatainak megoldásában.

5.2. A robotkar térbeli munkaterülete

Legyen $\mathcal{W} \subset \mathbb{R}^2$ egy kétdimenziós tér, amely a robot eszköz munkaterületét jelenti. Ezen munkaterület három különböző, egymástól elkülönülő régióra osztható, amelyek az alkalmazásban különböző funkciókat látnak el.

A $\mathcal{R}_{bg} \subset \mathcal{W}$ nevű régiót háttérregiónak nevezzük, amelyen nem helyezhető el alkatrész, és a robotkar sem a kiindulási, sem a végpozícióba nem mozgatható, és tiltott zónaként szolgál.

A $\mathcal{R}_{comps} \subset \mathcal{W}$ régiót a komponensek régiójának nevezzük, amely tartalmazza az alkatrészek érvényes elhelyezési tartományait, és a követke-

zöképpen fejezhetünk ki:

$$\mathcal{R}_{\text{comps}} = \bigcup_{i=1}^K r_{\text{comp}}^{(i)} \quad \text{ahol} \quad r_{\text{comp}}^{(i)} \cap r_{\text{comp}}^{(j)} = \emptyset \quad \text{minden } i \neq j \text{ esetében.} \quad (5.1)$$

A (5.1) összefüggés alapján minden $r_{\text{comp}}^{(i)} \subset \mathcal{W}$ ($i = 1, \dots, K$) egy diszjunkt halmaz, amely egy adott típusú aktuális alkatrészt tartalmaz.

A harmadik $\mathcal{R}_{\text{slots}} \subset \mathcal{W}$ régiót célrégiónak nevezzük, és a PCB előre meghatározott részeit jelöli. Formálisan a következőképpen adható meg:

$$\mathcal{R}_{\text{slots}} = \bigcup_{j=1}^L r_{\text{slot}}^{(j)} \quad \text{ahol} \quad r_{\text{slot}}^{(j)} \cap r_{\text{slot}}^{(i)} = \emptyset \quad \text{minden } i \neq j \text{ esetében,} \quad (5.2)$$

A (5.2) összefüggés alapján minden $r_{\text{slot}}^{(j)} \subset \mathcal{W}$ ($j = 1, \dots, L$) egy diszjunkt halmaz, amely az egyes beültetések célpozícióját jelöli.

A fenti jelöléseket és összefüggéseket használva a háttérregió, a komponensregió és a célregió együttesen lefedi a teljes $\mathcal{W} \in R^2$ munkaterületet, és a következőképpen adható meg:

$$\mathcal{W} = \mathcal{R}_{\text{bg}} \cup \mathcal{R}_{\text{comps}} \cup \mathcal{R}_{\text{slots}}, \quad (5.3)$$

A háttérregió, a komponensregió és a célregió páronként diszjunktak lesznek.

$$\mathcal{R}_{\text{bg}} \cap \mathcal{R}_{\text{comps}} = \emptyset \quad (5.4)$$

$$\mathcal{R}_{\text{bg}} \cap \mathcal{R}_{\text{slots}} = \emptyset \quad (5.5)$$

$$\mathcal{R}_{\text{comps}} \cap \mathcal{R}_{\text{slots}} = \emptyset \quad (5.6)$$

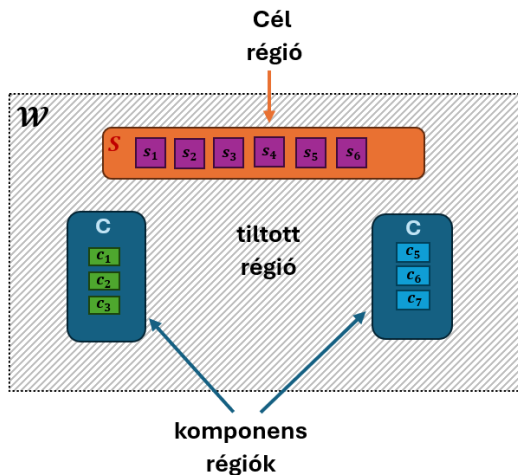
5.3. A komponens-foglalat hozzárendelés leírhatósága

Vegyünk figyelembe két konkrét véges részhalmazt a $\mathcal{W} \in R^2$ munkaterületen belül. Az első $C = \{c_1, c_2, \dots, c_K\}$ az észlelt komponensek egy részhalmazát jelöli, a második $S = \{s_1, s_2, \dots, s_L\}$ pedig a célhelyek helyőrzőit tartalmazza. A legegyszerűbb esetben tegyük fel, hogy $K = L$, így minden komponenshez pontosan egy hely tartozik.

A fenti jelöléseket használva a következő $f : C \mapsto S$ bijektív leképezés értelmezett, és az alábbi bináris mátrixszal fejezhető ki:

$$Y \in \{0, 1\}^{K \times L}, \quad Y_{ij} = \begin{cases} 1, & \text{ha } c_i \text{ hozzá van rendelve } s_j\text{-hez,} \\ 0, & \text{egyébként.} \end{cases} \quad (5.7)$$

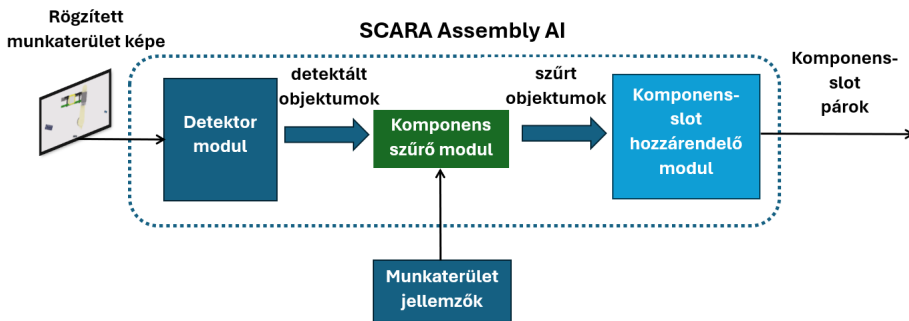
A fenti (5.7) egyenlet egyértelműen meghatározza a komponens-foglalat párosítás feltételeit (lásd a 5.1 ábrát). A későbbi szakaszokban ezen elméleti alapon terveztük meg neurális modellünket, amely megvalósítja a hozzárendelés predikcióját.



5.1. ábra. A robot eszköz munkaterülete és a megfelelő alrégiók.

5.4. A javasolt eljárás koncepciója

Az 5.2 és 5.3 alfejezetek részletesen ismertették és definiálták a probléma témakörét. A teoretikus háttér alapján belátható, hogy a probléma egyetlen lépésben nehezen vagy nemigen oldható meg, és többszintű architektúra kidolgozásra lesz szükség. Legelőször egy olyan alkomponensre lesz szükség, amely képes lesz felismerni a robotkar környezetében megtalálható tárgyakat és objektumokat lehetővé téve a tájékozódás képességét a robotkar számára. Ezt követően mindenképp kell egy olyan fő komponens, amely képes lesz ellátni a komponens-foglalt hozzárendelést. Ennek biztosításához ennek a komponensnek képesnek kell lennie megtanulni a robotkar környezetében megtalálható régiótípusokat, beleértve azok helyét és méretét, illetve a benne megtalálható komponensek altípusait és darabszámát is.



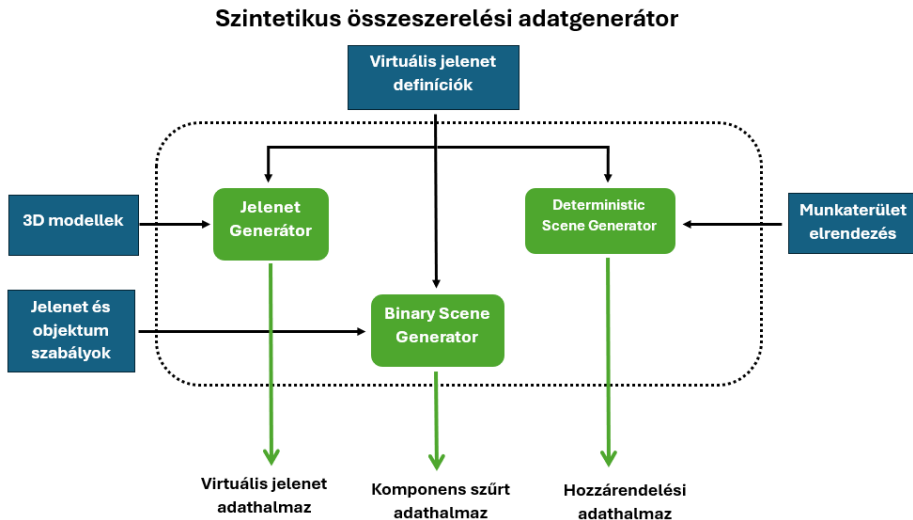
5.2. ábra. A javasolt módszerünk, az úgynevezett SCARA Assembly AI algoritmus architektúrája.

Az általunk javasolt és kidolgozott többszintű megközelítés felépítését mutatja be az 5.2. ábra. A környezetben megtalálható objektumok felismerését egy valós idejű működésre képes objektumdetektor fogja ellátni. A komponens-foglalt hozzárendelést pedig egy halmaz-transzformereken alapuló egyedi neurális hálózat fogja biztosítani. Mivel az objektumdetektor esetében előfordulhatnak duplikált vagy téves detektálások, mindenképp szükség lesz egy szűrőkomponensre, amely képes lesz az objektumdetektor kimenetét szétválogatni, és csak a helyes adatokat meghagyni. Erre a kom-

ponensre azért van szükség, mert mindenfajta hibás detektálás zajként fog megjelenni és megzavarhatja a hozzárendelési háló működését.

5.5. A szintetikus adatgeneráló modulrendszer bemutatása

Ez az alfejezet bemutatja az architektúra tanításhoz szintetikusan előállított adatkészletek létrehozásának folyamatát, valamint az ehhez szükséges, az Omniverse platformon megvalósított modulok részleteit. Az 5.3 ábra a szintetikus adatgenerálás folyamatának fő lépéseit mutatja be, és több generátort is tartalmaz.



5.3. ábra. A megvalósított szintetikus összeszerelési adatgenerátor (SADG) modul.

Legelső lépésként definiáljuk a virtuális jelenetet, amely magába foglalja a robotkar környezetét és a hozzá tartozó tartományokat és jellemzőket. Ezeket fogja mind a három generátor megkapni. Ezt követően történik meg az egyes adathalmazok generálása, amelyek szükségesek a három fő model tanításához.

A Jelenet generátor (FG) fogja biztosítani a képjeleneteket a detektor

számára. Ehhez plusz bemenetként megkapja a jelenetben megtalálható komponensek 3D modelljeit is. A generálás során ezeket a definiált környezetnek és kamerabeállításoknak megfelelően véletlenszerűen elhelyezi, és szintetikusán előállítja azt a képi adatmennyiséget az annotációkkal együtt, amely a detektor tanításához szükséges.

Jelölje $\mathcal{W} \subset R^2$ a fizikai munkaterületet (ahogy bevezettük a 5.2 szekcióban). Most definiáljuk annak virtuális megfelelőjét $\mathcal{W}_{virtual} = (\mathcal{R}, \mathcal{O})$ szintetikus jelenetként, ahol \mathcal{R} jelöli belül előre meghatározott régiók halmazát $\mathcal{W} \subset R^2$ munkaterületen belül, és $\mathcal{O} = (o_1, \dots, o_n)$ lesz a modellekhez és transzformációkhoz társított, példányosított 3D objektumok halmaza.

A fentiek alapján a $\text{Render} : \mathcal{W}_{virtual} \times \Theta \rightarrow \{(I_t, A_t)\}_{t=1}^T$ renderelő függvény leképezi a szintetikus jelenetet és a θ kamera-megvilágítási paramétereket egy I_t RGB színes képek halmazára a megfelelő A_t annotációkkal (b_i a határolókereteket és l_i osztálycímkéket jelöli). Az így kapott $\mathcal{D}_{scenes} = \{(I_t, A_t)\}_{t=1}^T$ adatkészletet a $\mathcal{W} \subset R^2$ munkaterület virtuális analógjának nevezzük.

A Bínáris jelenet generátor (BSG) feladata lesz a szűrő háló működéséhez szükséges tanuló adathalmaz előállítása. Ehhez további bemenetként megkapja azokat az objektum és régiómegszorításokat, amelyek pontosan definiálják azokat a szabályszerűségeket, melyekkel leírható, milyen esetek fordulhatnak elő a valóságban, és mik azok, amik tiltott állapotok. Továbbá kiegészítésként a generátor olyan adatokat is előállít, amellyel a detektor által adott téves eredmények is szűrhetővé válnak.

Legyenek a $\mathcal{C} = (c_1, c_2, \dots, c_K)$ komponenstípusok adottak, ahol mindegyik c_K a (w_k, h_k) fizikai méretekhez kapcsolódik.

Minden c_K komponenstípus esetében tartozik egy elfogadható elhelyezési régiókból álló $\mathcal{R}_{c_k} = \{r_{k,1}, r_{k,2}, \dots, r_{k,m_k}\}$ halmaz előre definiálva, ahol $r_{k,j}$ egy tengelyre igazított határoló régiót jelöl a $\mathcal{W} \subset R^2$ munkaterületen belül.

A BSG által generált x_i mintát az adott komponenstípus $x_i = [x, y, w, h, \text{label}_{id}, \text{IoU}_{max}] \in \mathbb{R}^6$ jellemzővektorának nevezzük, ahol (x, y) az objektum koordinátái, (w, h) az objektum dimenziói, label_{id} a komponens típusa, és IoU_{max} pedig a maximális átfedés lesz bármely más elhelyezett objektummal.

Minden x_i mintához bináris címke van hozzárendelve, amely $y_i = 1$ értéket vesz fel, ha az adott objektum érvényes (a kijelölt régió, tűréshatáron és IoU_{\max} belül van), ellenkező esetben $y_i = 0$ lesz, ha az objektum megsérti ezeket a szabályokat, és dummy-nak minősül.

A generátor kimenetét a $\mathcal{D}_{\text{bin}} = \{(x_i, y_i)\}_{i=1}^N$ komponenselemek bináris adatkészletének nevezünk, és a bináris szűrőhálózat betanításához alkalmazzuk.

A harmadik generátor, név szerint a Determinisztikus jelenet generátor (DSG) lesz a felelős a komponensek és foglalatok összepárosításához szükséges adathalmaz előállításáért. A korábban meghatározott $\mathcal{W}_{\text{virtual}}$ virtuális jelenet alapján bevezetjük a feladatspecifikus $\mathcal{H} = (\mathcal{P}, \mathcal{S}, \mathcal{M})$ struktúrát, ahol $\mathcal{P} = \{p_1, \dots, p_{N_p}\}$ az érvényes részek halmaza, $\mathcal{S} = \{s_1, \dots, s_{N_s}\}$ az előre meghatározott foglalatok halmaza, $\mathcal{M} \subseteq \{1, \dots, N_p\} \times \{1, \dots, N_s\}$ pedig a részek és foglalatok közötti valós alapigazság-hozzárendelési reláció.

Minden elemet egy $p_i = [x, y, w, h, c] \in \mathbb{R}^5$, $s_j = [x, y, w, h, c] \in \mathbb{R}^5$ ötdimenziós jellemzővektorral ábrázolunk, és a komponens-foglalat párosítás jellemzővektorainak nevezünk, ahol (x, y) térbeli koordináták, (w, h) a dimenzióbeli méretek, és c a típusazonosító lesznek. Fontos feltétel, hogy a párok tagjainál a típusazonosítónak azonosnak kell lennie.

A fent említett definíció esetében a dummy objektumokat már kiszűrjük, így minden rész pontosan egy foglalat-hoz lesz hozzárendelve, mivel $\forall p_i \in \mathcal{P} \exists! s_j \in \mathcal{S} : (i, j) \in \mathcal{M}$.

Formálisan a $\mathcal{D}_{\text{asg}} = \{(X^{(b)}, Y^{(b)})\}_{b=1}^B$ adatkészletet a komponens-foglalat párosítás jellemzővektorainak adatkészletének nevezzük, ahol minden minta $X^{(b)} = (\mathcal{P}^{(\ell)}, \mathcal{S}^{(\ell)})$, $Y^{(b)} = \mathcal{M}^{(\ell)}$ -ből áll.

A fenti definíció alapján $\mathcal{P}^{(\ell)} = \{p_i^{(b)}\}_{i=1}^{N_p}$ and $\mathcal{S}^{(\ell)} = \{s_j^{(b)}\}_{j=1}^{N_s}$ az alkatrészek és foglalatok halmaza lesz a $p_i^{(b)}, s_j^{(b)} \in \mathbb{R}^5$, and $\mathcal{M}^{(\ell)} \subseteq \{1, \dots, N_p\} \times \{1, \dots, N_s\}$ jellemzővektorokkal, amelyek a valós hozzárendelési párokat jelölik.

5.6. A javasolt eljárás architektúrákomponenseinek bemutatása

5.6.1. Detektor modul

A detektáló komponens megfelelő architektúrájának kiválasztásakor több fontos tényezőt is figyelembe kellett vennünk. Az egyik legfontosabb a pontosság, mivel az algoritmusnak képesnek kell lennie a kicsi, hasonló megjelenésű alkatrészek (pl. relék, mikrokontrollerek) megbízható megkülönböztetésére, és különösen fontos a sűrűn elhelyezkedő alkatrészek észlelése is, ahol az átfedések nagyon gyakoriak. A valós idejű működés biztosítása szintén kritikus fontosságú, mivel az összeszerelő robotok mozgása csak akkor vezérelhető, ha az detektálás valós időben történik. A késleltetés minimalizálása szintén szükséges a zárt hurkú vezérlés és a biztonságos működés érdekében. A modellnek továbbá képesnek kell lennie a globális kontextus figyelembevételére is, hogy elkerülje a helytelen hozzárendeléseket. Az elmúlt évtizedekben a detektáló architektúrák jelentős fejlődésen mentek keresztül, és a kifejlesztett módszerek három fő generációra oszthatók. A vizsgált módszerek első csoportját a kétfázisú detektorok architektúrája alkotja, amelyeket klasszikus megközelítésnek tekintenek. Működésük során először objektumjelölteket (proposals) generálnak, majd ezeket finomabb osztályozási és regressziós lépések követik. E kategóriába tartoznak többek között az R-CNN, Fast R-CNN, Faster R-CNN és Mask R-CNN alapú módszerek [157–160]. Ezeket a megoldásokat általában magas pontosság jellemzi, ugyanakkor a detektálási sebességük alacsonyabb, ezért erőforrás-igényesebbek és gyakran nem ideálisak valós idejű ipari alkalmazásokhoz. A második fő kategóriát az egyfázisú detektorok képezik, amelyek közvetlenül a jellemzőtérképről generálnak határolókereteket és osztályokat, így lényegesen gyorsabban működnek. Tipikus példák közé tartoznak a YOLO család, az SSD (Single Shot Detector) és a RetinaNet-alapú megoldások [161–163]. Fő előnyük a valós idejű működés, ugyanakkor gyakran utófeldolgozási lépésekre (pl. NMS – Non-Maximum Suppression) van szükség, mivel sűrűn elhelyezkedő vagy egymást átfedő objektumok esetén a pontosság csökkenhet. A harmadik fő kategóriát az elmúlt években megjelent transzformer-alapú detektorok

alkotják [164]. A hagyományos CNN-alapú módszerekkel ellentétben a transzformer-alapú detektorok egy önfigyelési mechanizmusra támaszkodnak, amely lehetővé teszi számukra a teljes kép globális kontextusának feldolgozását. Ezenkívül végpontok között taníthatóak, ami azt jelenti, hogy a határolókeretek és az osztálycímkék közvetlenül a modellből származnak, így nincs szükség külön utólagos feldolgozásra (pl. NMS). Tipikus példák a DETR, Deformable DETR, DN-DETR és RT-DETR architektúrák [152, 153, 165–167]. A fenti kritériumok és elvárások alapján a robotkar környezetének orientációját egy RT-DETR architektúrán alapuló detektor biztosítja [168], amely a javasolt módszer első fő komponense lesz.

Formálisan az RT-DETR modell a Magyar illesztéssel optimalizálja a detektálást: minden előre jelzett objektumot a legjobban illeszkedő valós dobozhoz rendel. A teljes veszteség a következő összetevőkből áll [168]:

$$\mathcal{L}_{\text{R-DT}} = \lambda_{\text{cls}} \cdot \mathcal{L}_{\text{cls}} + \lambda_{\ell_1} \cdot \mathcal{L}_{\ell_1} + \lambda_{\text{giou}} \cdot \mathcal{L}_{\text{GIoU}} \quad (5.8)$$

ahol \mathcal{L}_{cls} az osztályozási veszteség, \mathcal{L}_{ℓ_1} a doboz koordinátáinak L_1 regressziós hibája, $\mathcal{L}_{\text{GIoU}}$ az általánosított veszteséget jelöli, λ_{cls} , λ_{ℓ_1} és λ_{giou} pedig konstans súlyozási paraméterek.

A Magyar-algoritmus hozzárendelése a következőképpen fejezhető ki [168]:

$$\hat{\pi} = \underset{\pi \in S_N}{\operatorname{argmin}} \sum_{i=1}^N \mathcal{C}(y_i, \hat{y}_{\pi(i)}) \quad (5.9)$$

ahol π az N darab objektumok permutációját jelöli, y_i a doboz alapértéke, $\hat{y}_{\pi(i)}$ a doboz előrejelzése, \mathcal{C} pedig a költségfüggvény (az osztályozási és regressziós veszteségek kombinációja).

5.6.2. Szűrő modul

A szűrőhálózat komponensének feladata, hogy megbízhatóan megkülönböztesse a helyes és helytelen objektumadatokat a detektált komponensek között. Ez azért szükséges, mert sok olyan eset van, amely problémákat okozhat vagy nem felel meg a gyártási szabályoknak. Ide tartoznak például a rossz helyre elhelyezett, a megengedett területeken túlnyúló vagy más komponensekkel ütköző objektumok. Ha ezek a helytelen előfordulások

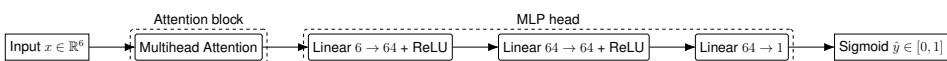
változatlanul maradnának a további feldolgozás során, akkor csökkentenék a foglalat-hozzárendelési modul teljesítményét és megbízhatóságát, mivel olyan kombinációkat kellene megtanulnia, amelyek a tényleges gyártási folyamatban nem fordulhatnak elő, és nehéz lenne megtalálni a helyes párosításokat az alkatrész foglalatok elhelyezésekor.

A szűrőkomponens tervezésekor számos tényezőt kellett figyelembe vennünk. Ezek közé tartozik az érvényességi kritériumok meghatározása, az érvénytelen komponensek megfelelő definiálása és a lehetséges téves észlelések leírása. Az első kritérium esetében fontos, hogy egy komponens akkor tekinthető érvényesnek, ha a típusához rendelt területen található, a terület nem haladhatja meg a jelenetben meghatározott százalékos értéket, és más komponensekkel való átfedése, amelyet az IoU mér, nem haladja meg a megadott küszöbértéket.

A második esetben minden olyan objektum, amely megsérti a tervezési kritériumokat vagy a jelenet korlátait, érvénytelen kategóriába sorolható. Ezeket a hálózatnak automatikusan el kell különítenie az érvényes komponensektől. A harmadik esetben a legfontosabb szempont az, hogy a kiválasztott detektorarchitektúra ismerete alapján meghatározzuk azokat a lehetséges kimeneti adatokat, amelyek problémákat okozhatnak a komponens-foglalat párosítás során. Ez felkészíti a szűrőkomponenst arra, hogy kiszűrje ezeket a hibás adatokat, és érvénytelen adatokként azonosítsa őket.

A szintetikus adatgenerálásról szóló alfejezet alapján (lásd a 5.5 alfejezetet a további részletekért) minden objektum leírható egy $x_i = [x, y, w, h, \text{label}_{\text{id}}, \text{IoU}_{\text{max}}] \in \mathbb{R}^6$ hatdimenziós jellemzővektorral. A hálózat célja egy bináris döntés meghozatala: minden mintánál egyértelműen jelzi, hogy az adott komponens érvényes ($y_i = 1$) vagy hamis ($y_i = 0$). Ez kizárja a hibás objektumokat a további feldolgozásból, és a hozzárendelési modul csak a valódi alkatrész-hely párosításokból tanul.

A fenti kritériumokat figyelembe véve kifejlesztettünk egy önfigyelésen alapuló többrétegű perceptron architektúrát. Ennek felépítését a 5.4 ábra szemlélteti.



5.4. ábra. A komponensszűrő neurális hálózat felépítése.

A szűrőhálózat bemenete minden komponenshez egy hatdimenziós jellemzővektor lesz, amely tartalmazza a pozíciót, méretet, típust és átfedés mértékét. A vektorokat először egy többfejű figyelemblokk dolgozza fel, amely képes figyelembe venni a különböző komponensek közötti kapcsolatokat. A figyelmi réteg kimenete ezután egy többretegű perceptron (MLP) fejhez kerül, amely két rejtett rétegen keresztül egyetlen skalár logit értéket állít elő. A végső kimenetet egy szigmoid aktiválás valószínűséggé alakítja, amely meghatározza, hogy az adott komponens az érvényes vagy az érvénytelen kategóriába tartozik-e.

A szűrőhálózat költségfüggvénye a bináris keresztentropia kiszámításán alapul. Formálisan, az $\mathcal{D}_{\text{train}} = \{(x_i, y_i)\}_{i=1}^N$, $x_i \in \mathbb{R}^6$ adatsor és a $\hat{y}_i = \sigma(f_{\theta}(x_i))$ predikció alapján a veszteség a következőképpen definiálható:

$$\mathcal{L}_{\text{WC}}(\theta) = -\frac{1}{N} \sum_{i=1}^N (w_1 y_i \log \hat{y}_i + w_0 (1 - y_i) \log(1 - \hat{y}_i)) \quad (5.10)$$

ahol $w_1, w_2 > 0$ az érvényes és az érvénytelen mintákat kiegyensúlyozó osztálysúlyok.

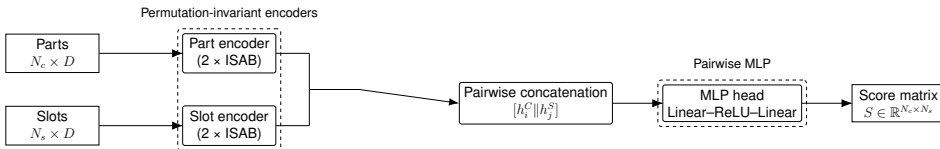
5.6.3. Hozzárendelő modul

A hozzárendelési modul képezi az összeszerelési módszer architektúrájának gerincét. Ezért ennek a modulnak kell a legrobosztusabbnak lennie. Fő feladata az lesz, hogy a felismert és szűrt objektumjellemzők és a PCB elrendezés alapján meghatározza a megfelelő alkatrészpárok párosítását, és megtalálja a helyes foglalat pozíciót. A 4. szakaszban szereplő definíciók alapján a probléma úgy írható le, hogy két halmaz között bijektív leképezést kell találni, azaz egy-egy leképezést kell találni a C komponenshalmaz és az S foglalat-halmaz között. Ezeknek a halmazoknak változó számú eleme lehet, és az elemek száma mintáról mintára változhat. Ezenkívül sorrendjük tetszőleges lehet. Ezért olyan neurális hálózatot kell tervezni, amely megfelel a következő fontos kritériumoknak.

A hálózatnak permutációinvariánsnak kell lennie a bemeneti jellemzővektorok tekintetében, és képesnek kell lennie változó számú elem kezelésére is. Egy másik fontos szempont az adatkészlet globális konzisztenciája,

amely kritikus fontosságú lesz az alkalmazási terület specifikus jellemzőiből adódó kihívások kezelésében, mint például a sűrűn elhelyezkedő alkatrészek, a minimális eltérésekkel rendelkező, hasonló méretű alkatrészek nagy száma stb.

A fenti szempontokat figyelembe véve megterveztük a hozzárendelési neurális hálózatot, amelynek felépítését a 5.5. ábra mutatja.



5.5. ábra. A hozzárendelő neurális hálózat felépítése.

A komponens-hozzárendelési feladat megoldásához halmaztranszformer-alapú architektúrát alkalmaztunk. A bemenetet a korábban említett két halmaz (komponensek és foglalatok) adja, amelyek elemei $p_i = [x, y, w, h, c] \in \mathbb{R}^5$ ötelemű jellemzővektorokkal írhatók le. A két bemeneti halmazt külön-külön dolgozzák fel a halmazkódoló blokkok (ISAB), biztosítva a permutációs függetlenséget és a változó elemszámok kezelését. Az így kapott rejtett reprezentációkból a vektorok összefűzésével páronkénti jellemzőket hoztunk létre minden lehetséges komponens–foglalat párhoz. Ezeket egy többrétegű perceptron (MLP) fejrész értékeli, amely minden párhoz pontszámot rendel.

Az összesített kimenet egy $S = [s_{ij}] \in \mathbb{R}^{N_c \times N_s}$ dimenziós pontszám-mátrix, ahol $s_{ij} \in \mathbb{R}$ az i -edik komponens és a j -edik foglalat összetartozásának erősségét fejezi ki. A tanítás és az értékelés során a pontszámokból egy egy-egy megfeleltetést (hozzárendelést) állítunk elő a Magyar párosító algoritmussal.

Jelölje $\mathcal{M}_{gt} \subseteq \{1, \dots, N_c\} \times \{1, \dots, N_s\}$ a valós (címkézett) párok halmazát. Az alapvető folytonos Magyar veszteségfüggvényt a következőképpen definiáljuk:

$$\mathcal{L}_{\text{Magyar_basic}}(S, \mathcal{M}_{gt}) = -\frac{1}{|\mathcal{M}_{gt}|} \sum_{(i,j) \in \mathcal{M}_{gt}} \log \sigma(s_{ij}), \quad (5.11)$$

ahol $\sigma(\cdot)$ a szigmoid függvény.

Az (5.11) egyenletben szereplő alapvető veszteség csak a valós párok pontszámainak növelését ösztönzi, azonban nem bünteti explicit módon (i) a hozzárendelés által kiválasztott, de helytelen párokat, illetve (ii) azokat a valós párokat, amelyeket a hozzárendelés kihagy. Ennek kezelésére először meghatározzuk a Magyar algoritmus által adott hozzárendelést. Mivel a Magyar algoritmus klasszikusan minimalizálási feladatot old meg, a maximalizálandó pontszámok helyett a negatívjukat adjuk meg költségként:

$$M_h = \text{Magyar}(-S).$$

Itt M_h a kiválasztott párok halmaza, $M_h \subseteq \{1, \dots, N_c\} \times \{1, \dots, N_s\}$, és egy-egy hozzárendelést reprezentál (egy index legfeljebb egy párban szerepelhet mindkét oldalon). Mivel a feladatban minden komponens legfeljebb egy foglalathoz, és minden foglalathoz legfeljebb egy komponenshez tartozik, a valós párok halmaza \mathcal{M}_{gt} szintén 1:1 hozzárendelést reprezentál, ezért $|M_h| = |\mathcal{M}_{gt}| = K$, ahol $K = \min(N_c, N_s)$.

A kiválasztott és a valós párok viszonyát három, egymástól diszjunkt halmazzal írjuk le: $A = M_h \cap \mathcal{M}_{gt}$ (helyes választások), $B = M_h \setminus \mathcal{M}_{gt}$ (kiválasztott, de helytelen párok), és $C = \mathcal{M}_{gt} \setminus M_h$ (kihagyott valós párok). Fontos megjegyezni, hogy a C halmaz pontosan azokat a valós párokat tartalmazza, amelyek nem szerepelnek a Magyar által kiválasztott M_h halmazban; ez nem „kétszeresen kihagyott” párokat jelent, hanem egyszerű halmazkülönbséget.

A kiválasztott párok minőségét bináris keresztentropiával értékeljük úgy, hogy az A elemei pozitív, a B elemei negatív címkét kapnak:

$$\mathcal{L}_{\text{match}}(S, M_h, \mathcal{M}_{gt}) = -\frac{1}{|M_h|} \left(\sum_{(i,j) \in A} \log \sigma(s_{ij}) + \sum_{(i,j) \in B} \log(1 - \sigma(s_{ij})) \right). \quad (5.12)$$

A kihagyott valós párok külön büntetőtagot kapnak:

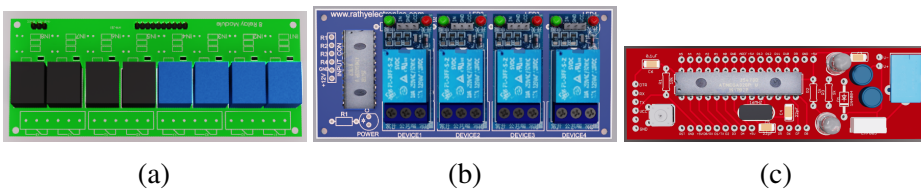
$$\mathcal{L}_{\text{miss}}(S, M_h, \mathcal{M}_{gt}) = -\frac{1}{|\mathcal{M}_{gt}|} \sum_{(i,j) \in C} \log \sigma(s_{ij}). \quad (5.13)$$

A végső veszteség a két tag súlyozott összege:

$$\mathcal{L}_{\text{Magyar_assembly}} = \mathcal{L}_{\text{match}}(S, M_h, \mathcal{M}_{gt}) + \alpha \mathcal{L}_{\text{miss}}(S, M_h, \mathcal{M}_{gt}), \quad \alpha \geq 0. \quad (5.14)$$

5.7. Tanítási és validációs eredmények

Ez az alfejezet az architektúra modellek tanításával és validálásával kapcsolatos részleteket tartalmazza. Az összes modellt, beleértve a kapcsolódó szoftvermodulokat és adatkészlet-kezelési rutinokat, PyTorch Lightning [169] segítségével valósítottuk meg. Az objektumdetektor képzéséhez virtuális testjeleneteket is összeállítottunk az Nvidia Omniverse programban [2]. A kapcsolódó leírásokat és paramétereket JSON adatstruktúrában tároltuk. A modellek betanításához három különböző testjelenetet állítottunk össze, amelyek különböző összetettségű és alkatrészszámú PCB-ket tartalmaztak, valamint a megfelelő számú régiót, hogy megvizsgálhassuk az architektúra általánosítási képességét (lásd a 5.6 ábrát).



5.6. ábra. A PCB panelek: (a) egyszerű felépítésű, (b) közepes bonyolultságú, (c) komplex felépítésű.

5.7.1. A detektor modul

Az RT-DETR architektúrán [168] alapuló detektor modell tanításához három fő komponenst használtunk, amelyek sorrendben a Frame Generator (lásd a Szintetikus adatgenerálás alfejezetben), a kapcsolódó adatkészlet modul és a tanító modul. A Képkocka Generátor (FG) számára a robot környezetében található összes komponenst és alkatrészt, valamint magát a robotkar modellt is biztosítottuk, hogy az detektor megfelelően legyen betanítva. Első lépésként az FG beolvasta a JSON struktúrában definiált 3D

modelleket és a környezet leírását, majd olyan jeleneteket generált, amelyek realisan modellezték a gyártási környezetet. Ezután az Nvidia Omniverse [2] Replicator modulját [141] használtuk a sikeres betanításhoz szükséges adatmennyiség és annotációk generálásához. Az érzékelő architektúra képzéséhez 10 000 képet és a hozzájuk tartozó annotációkat szintetizáltunk minden jelenethez. A szintetizálás során a Replicator modul [141] véletlenszerűen alkalmazta a következő transzformációkat: keverés, forgatás, különböző pozíciókba mozgatás és $\pm 5\%$ méretváltoztatás.

A következő lépésben az adatkészlet-modul beolvasta ezt a szintetizált adatkészletet, és létrehozta a tanításhoz szükséges tanuló, validációs és tesztelési egységeket. A képek kiegészítéséhez csak vízszintes és függőleges tükrözést, $\pm 5\%$ kontrasztot és színeltolást használtunk. Ezenkívül összevontuk a három jelenettípus adatkészleteit, majd létrehoztunk egy egységes tanuló és validációs készletet. Ez biztosította, hogy a modell a tanulási fázisban mindhárom jelenettípusból találkozzon mintákkal, és általános képességeket szerezzen az alkatrészek felismeréséhez. Az értékelést azonban három különálló tesztjeleneten végeztük, amelyek különböző összetettségű és különböző számú régióval rendelkező PCB-eket tartalmaztak. Ez lehetővé tette a modell azon képességének vizsgálatát, hogy változó összetettségű összeszerelési környezetekben is stabil teljesítményt nyújtson. A detektor tanítása során használt konfigurációt és paramétereket a 5.1. táblázat foglalja össze. A táblázatban feltüntetett értékeket Optuna csomagon alapuló hiperparaméter-optimalizáló eljárással választottuk ki [170].

5.1. táblázat. A detektormodell képzési konfigurációja és paraméterei.

Paraméter	Alapértelmezett érték(ek)
Cikluszsám	100
Kötegméret	16
Képméret (imgsz)	640
Optimalizáló	Adam
Kezdeti tanulási ráta (lr0)	0.00025 (backbone: 0.000025)
Momentum / béták	(0.9, 0.999)
Súlycsökkentés	0.05
Warmup iterációk	körülbelül 1000 iteráció
LR ütemező	cosine decay
Súlyértékek	cls = 1.0, bbox = 5.0, GIoU = 2.0

A detektor tanításának eredményei, valamint a hozzá tartozó metrikák a 5.2. táblázatban szerepelnek. Ezen értékek alapján megállapítható, hogy az architektúra választása megfelelő volt, és hogy a betanított modell képes megbízhatóan felismerni a jelenetben látható alkatrészeket és tárgyakat.

5.2. táblázat. A detektor modell képzési és validációs eredményei.

Metódus neve	Tanító adathalmaz	Val. adathalmaz	Teszt adathalmaz
mAP50	0.9982	0.9982	0.9963
mAP50-95	0.9795	0.9640	0.9640
Pontosság	0.9865	0.9704	0.9655
Visszahívás	0.9597	0.9467	0.9303

5.7.2. A szűrő modul

A szűrőmodell betanításához két fő komponenst implementáltunk, amelyeket Bináris jelenetgenerátor és Komponensszűrő-betanító néven fogunk említeni. A Bináris jelenetgenerátor egy adatkészlet-modul volt, amely betanítási, validációs és tesztelési adagokat generált a betanításhoz. Annak érdekében, hogy a modell betanítása kellően robusztus legyen, számos JSON-konfigurációt hoztunk létre, amelyek leírják az összes lehetséges esetet, beleértve a valós környezetben előforduló régiókat és komponenseket is. Ezenkívül egy JSON fájlban megadtuk a tiltott állapotokat tartalmazó szabályok definícióit is. Ennek alapján eldönthető, hogy mely komponensek érvényesek, és mely észlelt eredményeket kell kiszűrni.

A komponensszűrő modell tanítása során használt konfigurációt és paramétereket a 5.3. táblázat foglalja össze, míg a képzés eredményeit az 5.4. táblázat mutatja be. Az eredmények alapján megállapítható, hogy a modell képes volt a várt pontossággal megtanulni az összes lehetséges esetet, és kiszűrni az érvényes komponenseket. A paraméterek kiválasztása szintén az Optuna hiperparaméter-optimalizáló eljárással történt [170]. [170].

5.7.3. A hozzárendelő modul

Bevezettünk két fő komponenst is a feladatmodell tanításához, az úgynevezett Determinisztikus Jelenetgenerátort(DSG) és az „Hozzárendelő

5.3. táblázat. A szűrőháló konfigurációs paramétereit.

Paraméter	Alapértelmezett érték
Bemeneti dimenziók	6 ($x, y, w, h, label_id, iou_max$)
Rejtett dimenziók	64
Miniköteg mérete	32
Munkaszálak száma	4
Optimalizáló	Adam
Tanulási ráta	0.001
Veszteségfüggvény	Bináris keresztentrópia logitekkel (BCEWithLogitsLoss)
Osztály-aránytalanság kezelése	<code>class_weight</code> kiszámítva

5.4. táblázat. A komponensszűrő modell tanítási és validálási eredményei.

Adatkészlet neve	Tanítási veszteség	Validálási veszteség	Tanítási helyesség	Validálási helyesség
Adatkészlet A	0.0736	0.0905	0.966	0.955
Adatkészlet B	0.0425	0.0720	0.979	0.957
Adatkészlet C	0.1320	0.1470	0.923	0.893

tanító” modul. A DSG szintén egy adatkészlet-modul lesz, amelynek feladata a tanításhoz szükséges komponens-helyérték párok generálása.

5.5. táblázat. A feladatmodell képzési konfigurációja és paramétereit.

Parameter	Default Value
Bemeneti dimenziók	5 ($x, y, w, h, label_id$)
Rejtett	128
Miniköteg mérete	16
Munkaszálak száma	4
Optimalizáló	Adam
Tanulási ráta	0.00001
LR ütemező	ExponentialLR ($\gamma = 0.95$, per epoch)
Veszteségfüggvény	Folyamatos Magyar
Metrikák	Magyar-párosítási helyesség, Osztályozási helyesség

A modell betanítása során alkalmazott konfigurációkat és paramétereiket a 5.5. táblázat tartalmazza, míg a betanítás során kapott eredményeket a 5.6. táblázat foglalja össze. A paraméterek kiválasztása itt is ugyanúgy az

Optuna hiperparaméter-optimalizáló eljárással történt [170].

5.6. táblázat. A hozzárendelő modell tanítási és validálási eredményei.

Adatkészlet neve	Tanítási veszteség	Validálási veszteség	Tanítási helyesség	Validálási helyesség
Adatkészlet A	0.1120	0.1430	1.000	1.000
Adatkészlet B	0.1656	0.1824	0.9880	0.9780
Adatkészlet C	0.1820	0.2435	0.9340	0.9134

5.8. Szimuláció Omniverse platformon

A kidolgozott módszertan egyik legfontosabb lépése a bemutatás és tesztelés. Ennek megfelelően ez a szakasz lépésről lépésre bemutatja ezt a részt. Maga a teszt az Omniverse környezetben lett szimulálva. Fő célunk az volt, hogy a betanított architektúra kimenetét integráljuk egy olyan környezetbe, ahol a robotkar szimulált módon valóság-hű mozdulatokat hajthat végre.



5.7. ábra. A járműgyártási laboratórium és kutatóközpont virtuális laboratóriumi helyszíne.

Az NVIDIA Omniverse platform [2] egy valós idejű, fizikaalapú 3D szimulációs és együttműködési platform, amely az USD (Universal Scene

Description) formátumon [171] alapul. Lehetőséget nyújt komplex virtuális környezetek és rendszerek létrehozására, ahol különböző szoftverek és modellek kezelhetők egy egységes keretrendszerben.

A szoftvertechnológia maga több területen is alkalmazható. Ezek közül az ipari robotika lesz számunkra különösen releváns. A platform rendelkezik egy beépített ISAAC Sim [172] nevű applikációval, amelyet kifejezetten robotkarok, mobil robotok és autonóm rendszerek valósághű fizikai szimulációjának biztosítására fejlesztettek ki. Első lépésként három virtuális PCB-t alkalmaztunk tesztként. Ehhez kiindulási pontként az előző alfejezetben betanított modelleket és környezetet használtuk, amelyek elrendezését a 5.6. ábra mutatja. A teljes jeleneteket USD formátumban [171], Xform struktúrában [173] tároltuk (lásd a 5.7. ábrát). A szimulációs modul képes olvasni a transzformációs mátrixokat, majd azokat a pozíció és a forgás szerint módosítani. Ez lehetővé teszi a robotkar ízületeinek szögének és a komponensek aktuális állapotának pontos beállítását.

Algorithm 1: SCARA Assembly AI algoritmus

Bemenet: Jelenet JSON fájlok, konfiguráció szabályrendszerek,
betanított modellek súlyfájljai

Kimenet: Mozgási szekvencia szögekben

Inicializáció:

$jelenet \leftarrow loadJSON()$

$detektor, szuro, hozzarendelo \leftarrow modellekBetoltese(sulyok)$

Folyamat:

$objectumok \leftarrow detektor(jelenet);$ // Objektum
detektalas

$helyes \leftarrow filter(objectumok);$ // Helytelen elemek
szures

$parok \leftarrow hozzrendel(helyes, foglalatok);$
// Komponens-foglalat hozzárendelés

Inverz kinematika:

foreach (c, s) **in** $parok$ **do**

$szogek \leftarrow IK(c.pozicio, s.pozicio)$
 $szekvencia.hozza.fuzes(szogek)$

Kimenet:

A szekvencia visszaadása a környezetnek

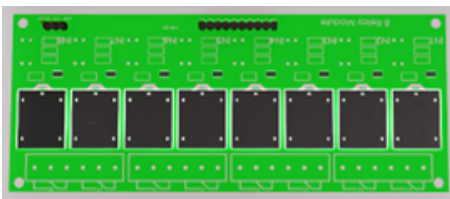
Következő lépésként a SCARA Assembly AI algoritmusának működése került beépítésre, amelynek a főbb lépéseit az előző oldalon látható pszeudokód mutat be:

1. A jelenet inicializálása: Ebben a fő lépésben történik meg a jelenetet leíró JSON állományok, a hozzá kapcsolódó modellfájlok, és az elrendezési szabályrendszert tartalmazó konfigurációs állomány betöltése is.
2. Algoritmus inicializálása: itt történik meg az előzőleg betanított modellek súlyfájljainak betöltése.
3. Inference és hozzárendelés: Az adott képkockán megtörténik az algoritmus lefuttatása, és az egyes modellkomponensek feldolgozzák és átadják egymásnak az adatstruktúrákat.
4. Szekvencia számítás: Az előző lépésben kiszámított hozzárendelési lista alapján alapján inverz kinematika segítségével kiszámítjuk a

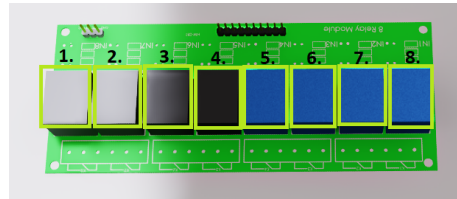
komponens-foglalat párokhoz tartozó csukló szögértékeket, és szekvenciát állítunk össze.

5. Legutolsó lépésként pedig visszaadjuk a környezetnek a kiszámított értékeket, és ellenőrizzük, hogy a komponensek ténylegesen a megfelelő pozíciókba kerültek-e.

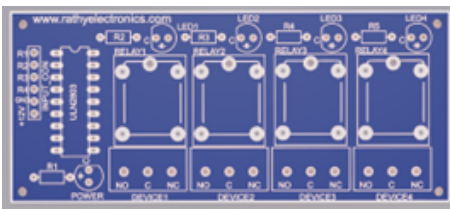
Az algoritmus által vizualizált hozzárendelési eredményeket a 5.8 ábracsoport szemlélteti.



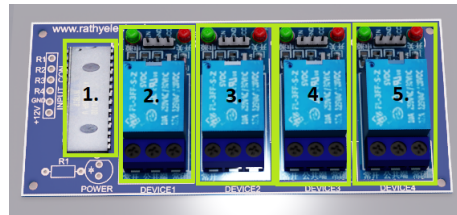
(a)



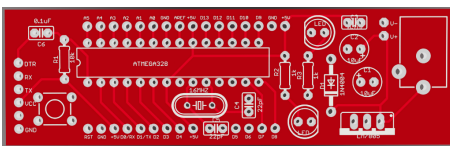
(b)



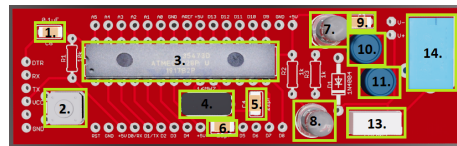
(c)



(d)



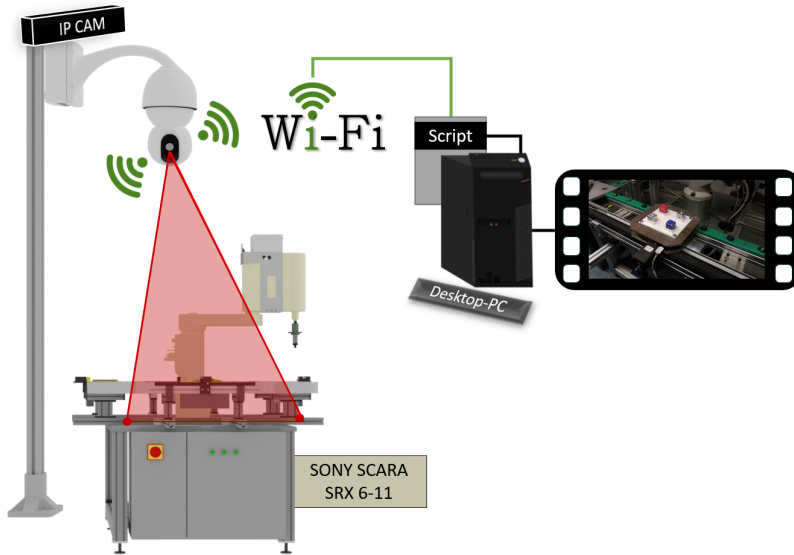
(e)



(f)

5.8. ábra. A SCARA Assembly AI algoritmus eredményei: (a,c,e) Az előre meghatározott slotokkal rendelkező üres PCB-k, (b,d,f) A PCB-k beültetett változata kijelölt alkatrészekkel.

A helytelen elemek szűrőjét és a hozzárendelési hálózatot a jelenlegi munkában minden jelenethez meghatározott szabályok alapján tanítottuk be, ezért a modell teljesítményét három különálló testjeleneten értékeltük. Az architektúra azonban nem korlátozódik erre a megközelítésre: egy átfogó, általános jelenet alapján a rendszer új testjeleneteken is képes működni, ahol a régiók és komponensek csak részben vannak jelen. Ez a kiterjesztés



5.9. ábra. A valós környezetben végzett validációhoz használt SCARA-alapú szerelőállomás kísérleti összeállítása. Az IP-kamera rögzíti a komponenselhelyezési képeket, és Wi-Fi-n keresztül továbbítja azokat a vezérlő- és adatfeldolgozó szkripteket futtató asztali számítógépre. A SONY SCARA SRX-611 a detektált komponens–foglalat párok alapján hajtja végre a pick-and-place műveleteket.

lehetőséget nyújt a generalizáció és a robusztusság még átfogóbb értékelésére, mivel a modell alkalmazhatóvá válna olyan új elrendezésekre is, amelyeket a képzési adatok nem fednek le teljes mértékben. A módszernek vannak bizonyos korlátai. Bonyolultabb elrendezések esetén az adatgenerálás nagyobb számítási erőforrásokat igényel, és a hozzárendelés pontossága erősen függ a képzési adatok változékonyságától. A jövőbeli munkatervek középpontjában a módszer kiterjesztése háromdimenziós feladatokra, valamint a megerősítéses tanuláson alapuló vezérlési stratégiák integrálása áll.

5.9. A valós roboton végzett kísérleti validáció

Az összeállított valós tesztkörnyezet felépítését szemlélteti a 5.9 ábra. A munkaterület fölé 1,5 m magasságban egy Tapo C200 RGB kamerát

Modul	Környezet	Kulcsmetrika	Teljesítmény	Megjegyzés
Detektálás (YOLOv8s-Seg)	Omniverse és valós	Pontosság	97.9%	Stabil domain transzfer mellett
Szűrőhálózat	Csak szintetikus	Pontosság	97.4%	Robusztus vizuális zajjal szemben
Hozzárendelő hálózat	Szintetikus és valós	Pontosság	96.9%	Legjobb $\alpha = 1.0$ konfigurációban
Szimulációs inferencia	Omniverse	Késleltetés/átbocsátás	16 ms / ≈ 62 FPS	Valós idejű működésre képes
Valós környezeti teszt	SCARA SRX-611	Sikerráta	98%	Fizikai validáció robot hardveren

5.7. táblázat. A SCARA Assembly AI keretrendszer összesített kiértékelésének összefoglalója a detektálási, szűrési, hozzárendelési és valós környezeti validációs fázisokon keresztül.

rögzítettünk, amelyet merőlegesen igazítottunk a szerelési felülethez.

A 5.7 táblázat az összesített kiértékelést mutatja, és a teljes SCARA Assembly AI algoritmus következetesen magas teljesítményt nyújt valamennyi észlelési és döntéshozatali modulban. A detektor szintetikus és valós körülmények között egyaránt 97,9%-os átlagos pontosságot ér el, míg a Vizuális Szűrőhálózat a domain-randomizált vizuális zaj ellenére is 97,4%-os pontosságot tart fenn.

A SONY SCARA SRX-611 platformon végzett fizikai validáció 98%-os sikeres szerelési arányt igazolt, ami azt mutatja, hogy a betanított modellek közvetlenül telepíthetők ipari hardverre is.

5.10. Konklúzió

A fejezet egy moduláris architektúrát ismertetett, amelynek célja egy ipari SCARA robot (SONY SRX-611) modellezése és betanítása szerelési feladatokra szintetikus adatok, mélytanulási modellek és szimulációs környezetek integrálásával. Az alábbi tézispontok összegzi a fejezetben ismertetett eredményeket.

Harmadik tézispont

A SCARA Assembly AI nevű algoritmushoz kapcsolódóan kidolgoztam és implementáltam egy Set-transformers architektúrán alapuló mélytanuló neurális hálózatot, amely képes sorrend függetlenül az alkatrészek és a hozzátartozó célrégiók bijektív leképezésének megtanulására és megvalósítására. Kísérletileg igazoltam, hogy különböző méretű és komplexitású nyomtatott áramköri lapkák esetében az architektúra képes legalább 91%-os pontosság elérésére. A kidolgozott háromlépcsős mélytanuló architektúrán alapuló eljárás pedig képes sorrend függetlenül dinamikusan megtanulni és megvalósítani SCARA típusú roboteszközöket vezérelve ipari beültetési folyamatokat ellátni. Az algoritmus működésének pontosságát kísérletileg igazoltam, egy az Omniverse ISA-AC Sim applikációban felépített virtuális környezetben. A kidolgozott módszertan validálása valós ipari környezetben is megtörtént SONY SCARA SRX-611 gépegységen, ahol a sikerességi szerelési arány 98%-os értéket ért el.

Tézishez kapcsolódó publikáció: [J1]. Kapusi, T.P.; Erdei, T.I.; Abdullah, M.; Husi, G.; Hajdu, A. SCARA Assembly AI: The Synthetic Learning-Based Method of Component-to-Slot Assignment with Permutation-Invariant Transformers for SCARA Robot Assembly. Robotics, 2025, 14, 175. <https://doi.org/10.3390/robotics14120175> Q1

Negyedik tézispont

Megterveztem egy újfajta adatszintetizáló szoftvermodult, amely az NVIDIA által fejlesztett Omniverse platformon alapulva képes valósághű adatkészletek előállítására és ipari robotika rendszerek környezetének és komponenseinek lemodellezésére. Kísérletileg igazoltam a szintetizált adatkészletek alkalmasak mélytanuló architektúrák betanítására.

Tézishez kapcsolódó publikáció: [J1]. Kapusi, T.P.; Erdei, T.I.; Masuk, A.; Husi, G.; Hajdu, A. SCARA Assembly AI: The synthetic learning-based method of Component-to-Slot Assignment with Permutation-Invariant Transformers for SCARA Robot Assembly. Q1 Submitted

Összefoglalás

Ez a disszertáció széleskörű áttekintést nyújtott a jelenlegi ipari gyártási folyamatok digitalizációjához és modernizációjához kapcsolódó kulcsfontosságú kutatási irányokkal kapcsolatban, valamint újfajta kidolgozott módszertanokat, és alkalmazásokat ismertetett. A bevezetést követően a második fejezetben részletesen kifejtésre kerültek a megvalósított eljárásokhoz szükséges hardveres eszközök, illetve a megtervezésükhöz és alkalmazásokhoz szükséges elméleti hátterek is a hozzákapcsolódó témaköröket beleértve. A harmadik fejezetben egy új, ipari robotikai rendszereken alkalmazott képből kép alapú fordítási módszert dolgoztam ki és részleteztem. Továbbfejlesztve azt ezt megelőző renderelésen alapú megközelítést, minél valóságosabb adatkészlet előállítására törekedtem, amely minden szükséges képi információjellemezőt magába foglal az alkalmazandó gyártási környezetnek megfelelően. A negyedik fejezetben részletezett eljárás a harmadik fejezetben ismertetett megközelítés tapasztalatai alapján került kidolgozásra folytatásként. Ebben a munkában egy újfajta képi adatkészlet fordító eljárást készítettem. Az ötödik fejezetben pedig egy újfajta mélytanuláson alapuló algoritmust dolgoztam ki, amelynek fő alkalmazási területe az ipari beültetési folyamatok. Itt is definiáltam a tervezéshez szükséges összefüggéseket, a roboteszköz munkakörnyezetét, és az azt alkotó főbb háromfajta régiótípust, amelyek együttesen leírják a környezet felépítését és határait. Ezt követően dolgoztam ki egy adatgeneráló eljárást, amely ennek megfelelően szintetizálja az adatkészleteket. Ennek az eljárásnak három fő komponense lett, amelyek név szerint a képjelenet generátor, a bináris jelenet generátor, és a hozzárendelő generátorok lettek. Erre ráépülve terveztem meg a háromlépcsős mélytanuló architektúrát, amely képes a célfeladatként definiált beültetési folyamat kiszámítására és előállítására a roboteszköz számára értelmezhető formátumban.

Köszönetnyilvánítás

Először is szeretném nagyon megköszönni Prof. Dr. Hajdu András témavezetőmnek, hogy vállalta a témavezetésem még a mesterképzés idején, és lehetőséget biztosított a kutatásokban való részvételre, és a tématerület megismerésében való elmélyülésben. Nagyon hálás vagyok minden iránymutatásért, támogatásért és segítségért amit kaptam az évek során, és igyekszem mindig ennek megfelelően színvonalas és megbízható munkát végezni folyamatosan.

Szintén szeretném megköszönni kollégámnak és jóbarátomnak, Erdei Timóteus Istvánnak minden segítséget és támogatást a PhD képzés kezdete óta, hiszen támogatása és szaktudása nélkül nem valósulhatott volna meg egyik jelenlegi munka és kutatás sem.

Közlemények listája

5.10.1. A PhD értekezés alapjául szolgáló közlemények

- [J1] **Kapusi, T.P.;** Erdei, T.I.; Masuk, A.; Husi, G.; Hajdu, A. *SCARA Assembly AI: The synthetic learning-based method of Component-to-Slot Assignment with Permutation-Invariant Transformers for SCARA Robot Assembly*. *Q1 Submitted*
- [J2] Erdei, T.I.; **Kapusi, T.P.;** Hajdu, A.; Husi, G. *Image-to-Image Translation-Based Deep Learning Application for Object Identification in Industrial Robot Systems*. *Robotics*, 2024, **13**, 88. <https://doi.org/10.3390/robotics13060088> **Q1**
- [J3] **Kapusi, T.P.;** Erdei, T.I.; Husi, G.; Hajdu, A. *Application of image translation-based approach in industrial manufacturing system for object identification*. *Int. Rev. Appl. Sci. Eng.*, 2025, 1–11. **Q2**

5.10.2. További közlemények

- [J4] Szántó, A., Abdullah, M., Kapusi, T., Diós, S. S. *Dynamic Modelling and Simulation of a Permanent Magnet Synchronous Motor (PMSM) Applied in a Prototype Race Car and the Comparison of Its Performance with BLDC Motor*. *Modelling*, 6 (3), 104, 1–21, 2025, **Q2**.
- [J5] Kapusi, T., Erdei, T. I., Husi, G., Hajdu, A. *Application of Deep Learning in the Deployment of an Industrial SCARA Machine for Real-Time Object Detection*. *Robotics*, 11 (4), 69, 1–20, 2022, **Q1**.
- [J6] Kapusi, T., Kovács, L., Hajdu, A. *Deep learning-based anomaly detection for imaging in autonomous vehicles*. In: *2022 IEEE 2nd Conference on Information Technology and Data Science (CITDS)*, IEEE, Piscataway (NJ), 142–147, 2022. ISBN 978-1-6654-9652-0.
- [J7] Józsa, R. O., Erdei, T. I., Kapusi, T., Tóth, S., Husi, G. *MP-9S Industrial Manipulator Handling and Scrapping Tasks in Industrial Environments*. *Műszaki Tudományos Közlemények*, 15 (1), 38–42, 2021.
- [J8] Tóth, J., Kapusi, T., Harangi, B., Tomán, H., Hajdu, A. *Accelerating the Optimization of a Segmentation Ensemble Using Image Pyramids*. In: *11th International Symposium on Image and Signal Processing*

and Analysis (ISPA 2019), IEEE, Piscataway (NJ), 43–48, 2019. ISBN 978-1-7281-3140-5.

- [J9] Krucsó, L., Erdei, T. I., Kapusi, T., Husi, G. *Designing an ATmega328 Microcontroller-Based Gesture-Controlled IoT UGV Unit and Creating a Camera System Using Linux Distribution. Recent Innovations in Mechatronics*, 6 (1), 1–7, 2019.
- [J10] Varga, T. A., Kapusi, T., Mankovits, T. *Celluláris anyagok modellezése digitális képek alapján / Modelling of Cellular Materials Based on Digital Images*. In: *XXII. Fiatal Műszakiak Tudományos Ülésszaka*, Erdélyi Múzeum Egyesület, Kolozsvár, 399–402, 2017. ISBN 978-963-449-018-0.
- [J11] Varga, T. A., Kapusi, T. *Digital Image Analysis of Metal Foam Specimens*. In: *4th International Scientific Conference on Advances in Mechanical Engineering (ISCAME 2016)*, University of Debrecen, Debrecen, 584–587, 2016. ISBN 978-963-473-944-9.



Nyilvántartási szám: DEENK/37/2026.PL
Tárgy: PhD Publikációs Lista

Jelölt: Kapusi Tibor Péter
Doktori Iskola: Informatikai Tudományok Doktori Iskola
MTMT azonosító: 10084802

A PhD értekezés alapjául szolgáló közlemények

Idegen nyelvű tudományos közlemények hazai folyóiratban (1)

1. **Kapusi, T.**, Erdei, T. I., Husi, G., Hajdu, A.: Application of image translation-based approach in industrial manufacturing system for object identification.
Int. rev. appl. sci. eng. [Epub ahead of print], 1-11, 2025. ISSN: 2062-0810.
DOI: <http://dx.doi.org/10.1556/1848.2025.01005>

Idegen nyelvű tudományos közlemények külföldi folyóiratban (2)

2. **Kapusi, T.**, Erdei, T. I., Abdullah, M., Husi, G., Hajdu, A.: SCARA Assembly AI: The Synthetic Learning-Based Method of Component-to-Slot Assignment with Permutation-Invariant Transformers for SCARA Robot Assembly.
Robotics. 14 (12), 1-36, 2025. ISSN: 2218-6581.
DOI: <http://dx.doi.org/10.3390/robotics14120175>
IF: 3.3 (2024)
3. Erdei, T. I., **Kapusi, T.**, Hajdu, A., Husi, G.: Image-to-Image translation-based deep learning application to object identification in industrial robot systems.
Robotics. 13 (6), 1-21, 2024. ISSN: 2218-6581.
DOI: <http://dx.doi.org/10.3390/robotics13060088>
IF: 3.3

További közlemények

Idegen nyelvű, külföldi könyvrészletek (1)

4. **Kapusi, T.**, Kovács, L., Hajdu, A.: Deep learning-based anomaly detection for imaging in autonomous vehicles.
In: 2022 IEEE 2nd Conference on Information Technology and Data Science (CITDS): Proceedings . Szerk.: Fazekas István, IEEE, Piscataway (NJ), 142-147, 2022. ISBN: 9781665496520





Magyar nyelvű tudományos közlemények külföldi folyóiratban (1)

5. Józsa, R. O., Erdei, T. I., **Kapusi, T.**, Tóth, S., Husi, G.: MP-9S ipari manipulátor anyagmozgatási és selejtezési feladatok ellátása ipari környezetben.
Műsz. tud. közl. 15, 38-42, 2021. ISSN: 2393-1280.
DOI: <http://dx.doi.org/10.33895/mtk-2021.15.08>

Idegen nyelvű tudományos közlemények hazai folyóiratban (1)

6. Krucsó, L., Erdei, T. I., **Kapusi, T.**, Husi, G.: Designing an ATmega328 microcontroller based gesture-controlled IoT UGV unit and creating a camera system using Linux distribution.
Recent Innov. Mechatron. 6 (1), 1-7, 2019. EISSN: 2064-9622.

Idegen nyelvű tudományos közlemények külföldi folyóiratban (3)

7. Szántó, A., Abdullah, M., **Kapusi, T.**, Diós, S. S.: Dynamic Modelling and Simulation of a Permanent Magnet Synchronous Motor (PMSM) Applied in a Prototype Race Car and the Comparison of Its Performance with BLDC Motor.
Modelling. 6 (3), 1-21, 2025. EISSN: 2673-3951.
DOI: <http://dx.doi.org/10.3390/modelling6030104>
IF: 1.5 (2024)
8. **Kapusi, T.**, Erdei, T. I., Husi, G., Hajdu, A.: Application of Deep Learning in the Deployment of an Industrial SCARA Machine for Real-Time Object Detection.
Robotics. 11 (4), 1-20, 2022. ISSN: 2218-6581.
DOI: <https://doi.org/10.3390/robotics11040069>
IF: 3.7
9. Józsa, R. O., Erdei, T. I., **Kapusi, T.**, Tóth, S., Husi, G.: MP-9S Industrial Manipulator Handling and Scrapping Tasks in Industrial Environments. utánközlés nyelvi változat,
Műsz. tud. közl. 15 (1), 38-42, 2021. ISSN: 2601-5773.
DOI: <http://dx.doi.org/10.33894/mtk-2021.15.08>

Magyar nyelvű konferencia közlemények (1)

10. Varga, T. A., **Kapusi, T.**, Mankovits, T.: Celluláris anyagok modellezése digitális képek alapján / Modelling of Cellular Materials Based on Digital Images.
In: A XXII. Fiatal Műszakiak Tudományos Ülésszak előadásai = Proceedings of the XXIIth International Scientific Conference of Young Engineers. Szerk.: Bitay Enikő, Erdélyi Múzeum Egyesület (EME), Kolozsvár, 399-402, 2017. ISBN: 9789634490180

Idegen nyelvű konferencia közlemények (2)

11. Tóth, J., **Kapusi, T.**, Harangi, B., Tomán, H., Hajdu, A.: Accelerating the Optimization of a Segmentation Ensemble using Image Pyramids.
In: 11th International Symposium on Image and Signal Processing and Analysis (ISPA 2019). Eds.: S. Lončarić, R. Bregović, M. Carli, M. Subašić, Institute of Electrical and Electronics Engineers (IEEE), Piscataway, NJ, USA, 43-48, 2019. ISBN: 9781728131405





12. Varga, T. A., **Kapusi, T.**: Digital image analysis of metal foam specimens.

In: Proceedings of the 4th International Scientific Conference on Advances in Mechanical Engineering (ISCAME 2016). Ed.: Sándor Bodzás, Tamás Mankovits, University of Debrecen Faculty of Engineering, Debrecen, 584-587, 2016. ISBN: 9789634739449

A közlő folyóiratok összesített impakt faktora: 11,8

**A közlő folyóiratok összesített impakt faktora (az értekezés alapjául szolgáló közleményekre):
6,6**

A DEENK a Jelölt által a Tudóstérbe feltöltött adatok bibliográfiai és tudományometriai ellenőrzését a tudományos adatbázisok és a Journal Citation Reports Impact Factor lista alapján elvégezte.

Debrecen, 2026.01.30.



Irodalomjegyzék

- [1] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, „High-resolution image synthesis and semantic manipulation with conditional gans,” 2018. Available online.
- [2] N. Corporation, „Nvidia omniverse platform – a simulation and collaboration framework for 3d workflows,” 2024. Available online.
- [3] F. Caviggioli and E. Ughetto, „The four smarts of industry 4.0: Evolution of ten years of research and future perspectives,” Technological Forecasting and Social Change, vol. 168, p. 120784, 2021.
- [4] H. Kagermann, W. Wahlster, and J. Helbig, Recommendations for Implementing the Strategic Initiative Industrie 4.0. Frankfurt, Germany: acatech – National Academy of Science and Engineering, 2013.
- [5] L. M. Kipper, L. B. Furstenu, D. Hoppe, R. Frozza, and S. Iepesen, „Industry 4.0: An overview of technologies and applications,” Computers and Industrial Engineering, vol. 151, p. 106990, 2020.
- [6] A. Kusiak, „Smart manufacturing must embrace big data,” Nature, vol. 554, pp. 509–511, 2018.
- [7] F. Tao, M. Zhang, A. Liu, and A. Nee, „Digital twin-driven smart manufacturing: Connotation, reference model, applications and research issues,” Robotics and Computer-Integrated Manufacturing, vol. 61, p. 101837, 2018.
- [8] J. Bokrantz, A. Skoogh, C. Berlin, T. Wuest, and J. Stahre, „Smart maintenance: An empirical study of maintenance performance in in-

- dustry 4.0,” International Journal of Production Economics, vol. 228, p. 107843, 2020.
- [9] L. S. Dalenogare, G. B. Benitez, N. F. Ayala, and A. G. Frank, „The expected contribution of industry 4.0 technologies for industrial performance,” International Journal of Production Economics, vol. 204, pp. 383–394, 2018.
- [10] S. Wang, J. Wan, D. Li, and C. Zhang, „Implementing smart factory of industrie 4.0: An outlook,” International Journal of Distributed Sensor Networks, vol. 12, no. 1, p. 3159805, 2016.
- [11] E. Kaasinen, F. Schmalfuß, C. Öztürk, S. Aromaa, M. Boubekur, J. Heilala, T. Kuula, M. Liinasuo, S. Mach, and T. Walter, „Empowering and engaging industrial workers with operator 4.0 solutions,” Computers and Industrial Engineering, vol. 139, p. 105678, 2020.
- [12] D. Romero, J. Stahre, T. Wuest, O. Noran, P. Bernus, Fast-Berglund, and D. Gorecky, „Towards an operator 4.0 typology: A human-centric perspective on the fourth industrial revolution technologies,” in Proceedings of the International Conference on Computers and Industrial Engineering (CIE46), pp. 1–11, 2016.
- [13] D. Romero, P. Bernus, O. Noran, J. Stahre, and Fast-Berglund, „The operator 4.0: Human cyber-physical systems and adaptive automation towards human–automation symbiosis work systems,” Computers and Industrial Engineering, vol. 139, p. 106128, 2020.
- [14] C. Cimini, A. Boffelli, A. Lagorio, M. Kalchschmidt, and R. Pinto, „How human–robot collaboration is changing job design in manufacturing: A multiple case study,” International Journal of Production Research, vol. 58, no. 7, pp. 2136–2154, 2020.
- [15] Y. Cohen, M. Faccio, F. Pilati, and X. Yao, „Design and management of digital manufacturing and assembly systems in the industry 4.0 era,” International Journal of Production Research, vol. 57, no. 6, pp. 1607–1621, 2019.

- [16] P. Fantini, M. Pinzone, and M. Taisch, „Placing the operator at the centre of industry 4.0 design: Modelling and assessing human activities within cyber-physical systems,” Computers and Industrial Engineering, vol. 139, p. 105722, 2020.
- [17] Segura, M. Hincapié, and C. Rusu, „Smart working: Human-technology collaboration in industry 4.0,” Applied Sciences, vol. 10, no. 13, p. 4573, 2020.
- [18] I. Zolotová, P. Papcun, E. Kajáti, M. Miškuf, and J. Mocnej, „Smart and cognitive solutions for operators in industry 4.0: Human–machine collaboration perspective,” Computers and Industrial Engineering, vol. 139, p. 105684, 2020.
- [19] M. Pinzone, P. Fantini, S. Perini, S. Garavaglia, and M. Taisch, „A framework for assessing sustainability in human–machine collaboration within the manufacturing sector,” Sustainability, vol. 12, no. 17, p. 6750, 2020.
- [20] S. Fletcher, P. Webb, J. Wilson, and S. Sharples, „A framework for human factors integration in adaptive manufacturing systems of the future,” Applied Ergonomics, vol. 89, p. 103204, 2020.
- [21] A. G. Frank, L. S. Dalenogare, and N. F. Ayala, „Industry 4.0 technologies: Implementation patterns in manufacturing companies,” International Journal of Production Economics, vol. 210, pp. 15–26, 2019.
- [22] A. G. Frank, L. S. Dalenogare, and N. F. Ayala, „Industry 4.0 technologies: Implementation patterns in manufacturing companies,” International Journal of Production Economics, vol. 210, pp. 15–26, 2019.
- [23] G. Büyüközkan and F. Göçer, „Digital supply chain: Literature review and a proposed framework for future research,” Computers in Industry, vol. 97, pp. 157–177, 2018.
- [24] J. O. Strandhagen, L. R. Vallandingham, G. Fragapane, J. Strandhagen, E. Stangeland, and N. Sharma, „Logistics 4.0 and emerging

- sustainable business models,” Advances in Manufacturing, vol. 5, no. 4, pp. 359–369, 2017.
- [25] G. F. Frederico, J. A. Garza-Reyes, A. Anosike, and V. Kumar, „Supply chain 4.0: Concepts, maturity and research agenda,” Supply Chain Management: An International Journal, vol. 25, no. 2, pp. 262–282, 2019.
- [26] D. Ivanov, A. Dolgui, and B. Sokolov, „The impact of digital technology and industry 4.0 on the ripple effect and supply chain risk analytics,” International Journal of Production Research, vol. 54, no. 23, pp. 6917–6938, 2016.
- [27] L. Barreto, A. Amaral, and T. Pereira, „Industry 4.0 implications in logistics: An overview,” Procedia Manufacturing, vol. 13, pp. 1245–1252, 2017.
- [28] A. G. Frank, L. S. Dalenogare, and N. F. Ayala, „Smart manufacturing and logistics using autonomous mobile robots and cyber–physical systems,” Procedia Manufacturing, vol. 39, pp. 202–209, 2019.
- [29] H. Kahle, F. Teuteberg, and M. Böhm, „Digitalization in manufacturing: The role of smart products and smart services in value creation,” Procedia CIRP, vol. 93, pp. 578–583, 2020.
- [30] M. E. Porter and J. E. Heppelmann, „How smart, connected products are transforming competition,” Harvard Business Review, vol. 92, no. 11, pp. 64–88, 2014.
- [31] M. Ardolino, M. Rapaccini, N. Saccani, P. Gaiardelli, G. Crespi, and C. Ruggeri, „The role of digital technologies for the service transformation of industrial companies,” International Journal of Production Research, vol. 56, no. 6, pp. 2116–2132, 2017.
- [32] J. Cenamor, M. Rönnerberg Sjö Dahl, and V. Parida, „Adopting a platform approach in servitization: Leveraging digitalization for business model innovation,” International Journal of Production Economics, vol. 192, pp. 54–65, 2017.

- [33] N. F. Ayala, C. A. Paslauski, A. Ghezzi, and A. G. Frank, „Knowledge sharing dynamics in product-service system development: Addressing the transdisciplinary approach,” International Journal of Production Economics, vol. 183, pp. 179–188, 2017.
- [34] B. A. Kadir and O. Broberg, „Industry 5.0 – past, present, and near future,” Procedia Computer Science, vol. 218, pp. 343–350, 2023.
- [35] A. Orea-Giner, L. Fuentes-Moraleda, T. Villacé-Molinero, et al., „Does the implementation of robots in hotels influence the overall tripadvisor rating? a text mining analysis from the industry 5.0 approach,” Tourism Management, vol. 93, p. 106915, 2022.
- [36] E. Kaasinen, A. Anttila, P. Heikkilä, et al., „Smooth and resilient human–machine teamwork as an industry 5.0 design challenge,” Sustainability, vol. 14, no. 5, pp. 1–20, 2022.
- [37] European Commission, „Industry 5.0.” https://ec.europa.eu/info/research-and-innovation/research-area/industrial-research-and-innovation/industry-50_en, 2021. Accessed: 2025-10-12.
- [38] V. Popov, E. Kudryavtseva, N. Katiyar, et al., „Industry 4.0 and digitalisation in healthcare,” Materials, vol. 15, no. 6, pp. 1–15, 2022.
- [39] E. Coronado, T. Kiyokawa, G. Ricardez, et al., „Evaluating quality in human–robot interaction: A systematic search and classification of performance and human-centered factors, measures and metrics towards an industry 5.0,” Journal of Manufacturing Systems, vol. 63, pp. 392–410, 2022.
- [40] E. Carayannis, K. Christodoulou, P. Christodoulou, et al., „Known unknowns in an era of technological and viral disruptions—implications for theory, policy, and practice,” Journal of the Knowledge Economy, vol. 13, no. 1, pp. 587–610, 2022.
- [41] M. Maier, A. Ebrahimzadeh, A. Beniiche, et al., „The art of 6g (tao 6g): How to wire society 5.0 [invited],” Journal of Optical

Communications and Networking, vol. 14, no. 2, pp. A101–A113, 2022.

- [42] A. Duggal, P. Malik, A. Gehlot, et al., „A sequential roadmap to industry 6.0: Exploring future manufacturing trends,” IET Communications, vol. 16, no. 5, pp. 521–531, 2022.
- [43] M. Shahbakhsh, G. Emad, and S. Cahoon, „Industrial revolutions and transition of the maritime industry: The case of seafarer’s role in autonomous shipping,” Asian Journal of Shipping and Logistics, vol. 38, no. 1, pp. 10–18, 2022.
- [44] P. Maddikunta, Q. Pham, P. Boopathi, et al., „Industry 5.0: A survey on enabling technologies and potential applications,” Journal of Industrial Information Integration, vol. 26, p. 100257, 2022.
- [45] Z. Fatima, M. Tanveer, Waseemullah, et al., „Production plant and warehouse automation with iot and industry 5.0,” Applied Sciences, vol. 12, no. 4, pp. 1–15, 2022.
- [46] G. Paul, N. Abele, and K. Kluth, „A review and qualitative meta-analysis of digital human modeling and cyber-physical systems in ergonomics 4.0,” IIESE Transactions on Occupational Ergonomics and Human Factors, vol. 9, no. 3–4, pp. 111–123, 2021.
- [47] B. Pereira Guimarães, C. da Silva Fernandes, D. Amaral de Figueiredo, et al., „Cutting temperature measurement and prediction in machining processes: Comprehensive review and future perspectives,” International Journal of Advanced Manufacturing Technology, vol. 120, no. 5–6, pp. 2849–2878, 2022.
- [48] R. Sindhvani, S. Afridi, A. Kumar, et al., „Can industry 5.0 revolutionize the wave of resilience and social value creation? a multi-criteria framework to analyze enablers,” Technology in Society, vol. 68, p. 101887, 2022.
- [49] A. Verma, P. Bhattacharya, and N. Madhani, „Blockchain for industry 5.0: Vision, opportunities, key enablers, and future directions,” IEEE Access, vol. 10, pp. 69160–69199, 2022.

- [50] S. Nahavandi, „Industry 5.0: A human-centric solution,” Sustainability, vol. 11, no. 16, p. 4371, 2019.
- [51] D. Gürdür Broo, O. Kaynak, and S. Sait, „Rethinking engineering education at the age of industry 5.0,” Journal of Industrial Information Integration, vol. 25, p. 100311, 2022.
- [52] E. Alvarez-Aros and C. Bernal-Torres, „Technological competitiveness and emerging technologies in industry 4.0 and industry 5.0,” Anais da Academia Brasileira de Ciencias, vol. 93, no. 1, pp. 1–20, 2021.
- [53] Y. Lu, „The current status and developing trends of industry 4.0: A review,” Information Systems Frontiers, 2021.
- [54] A. Kaklauskas, N. Lepkova, S. Raslanas, et al., „Covid-19 and green housing: A review of relevant literature,” Energies, vol. 14, no. 8, pp. 1–25, 2021.
- [55] V. Roblek, M. Meško, and I. Podbregar, „Mapping of the emergence of society 5.0: A bibliometric analysis,” Organizacija, vol. 54, no. 4, pp. 293–305, 2021.
- [56] T. Nguyen, Q. Duong, T. V. Nguyen, et al., „Knowledge mapping of digital twin and physical internet in supply chain management: A systematic literature review,” International Journal of Production Economics, vol. 244, p. 108381, 2022.
- [57] S. de Miranda, A. Córdoba-Roldán, F. Aguayo-González, et al., „Neuro-competence approach for sustainable engineering,” Sustainability, vol. 13, no. 8, pp. 1–20, 2021.
- [58] KUKA Roboter GmbH, Augsburg, Germany, KR 5 Arc specification, 2011. Issued: 21.03.2011.
- [59] A. Talli and A. Giriapur, „Kinematic analysis and simulation of industrial robot based on roboanalyzer,” 2021. Available online.

- [60] C. Andreucci, E. Fonseca, and R. Jorge, „3d printing as an efficient way to prototype and develop dental implants,” BioMed Informatics, vol. 2, pp. 671–679, 2022.
- [61] M. Korol, J. Vanca, V. Majstorovic, M. Kocisko, P. Baron, J. Torok, A. Vodilka, and S. Hlavata, „Study of the influence of input parameters on the quality of additively produced plastic components,” in Proceedings of the 2022 13th International Conference on Mechanical and Aerospace Engineering (ICMAE), (Bratislava, Slovakia), pp. 39–44, 2022.
- [62] E. Engineers, „Abs plastic filament engineering information.” Accessed on 12 October 2023.
- [63] C. Choe, M. Choe, and S. Jung, „Run your 3d object detector on nvidia jetson platforms: A benchmark analysis,” Sensors, vol. 23, no. 8, p. 4005, 2023.
- [64] J. Lee, B. Varghese, R. Woods, and H. Vandierendonck, „Tod: Transprecise object detection to maximise real-time accuracy on the edge,” in 2021 IEEE 5th International Conference on Fog and Edge Computing (ICFEC), pp. 53–60, 2021.
- [65] E. Humes, M. Navardi, and T. Mohsenin, „Squeezed edge yolo: Onboard object detection on edge devices,” arXiv preprint, 2023.
- [66] S. Liu, J. Zha, J. Sun, Z. Li, and G. Wang, „Edgeyolo: An edge-real-time object detector,” arXiv preprint, 2023.
- [67] M. Hullin, E. Eisemann, H.-P. Seidel, and S. Lee, „Physically-based real-time lens flare rendering,” ACM Transactions on Graphics, vol. 30, p. 108, 2011.
- [68] S. Lee and E. Eisemann, „Practical real-time lens flare rendering,” Computer Graphics Forum, vol. 32, p. 7, 2013.
- [69] D. Seland, „An industry demanding more: Intelligent illumination and expansive measurement volume sets the new helix apart from other 3-d metrology solutions,” Quality, vol. 50, pp. 22–24, 2011. Available online.

- [70] P. Martinez, D. Ahmad, and M. Al-Hussein, „A vision-based system for pre-inspection of steel frame manufacturing,” Automation in Construction, vol. 97, pp. 151–163, 2019.
- [71] Y. Wu, Q. He, T. Xue, R. Garg, J. Chen, A. Veeraraghavan, and J. Barron, „How to train neural networks for flare removal,” in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), (Montreal, QC, Canada), pp. 2239–2247, 2021.
- [72] S.-T. Chen, C. Cornelius, J. Martin, and D. Chau, „Robust physical adversarial attack on faster r-cnn object detector,” arXiv preprint arXiv:1804.05810, 2018.
- [73] T. Kapusi, L. Kovács, and A. Hajdu, „Deep learning-based anomaly detection for imaging in autonomous vehicles,” in 2022 IEEE 2nd Conference on Information Technology and Data Science (CITDS), pp. 142–147, 2022.
- [74] M. He, J. Liao, D. Chen, L. Yuan, and P. V. Sander, „Progressive color transfer with dense semantic correspondences,” ACM Transactions on Graphics (TOG), vol. 37, no. 6, pp. 1–15, 2018.
- [75] M. Avi-Aharon, A. Arbelle, and T. Riklin Raviv, „Deephist: Differentiable joint and color histogram layers for image-to-image translation,” arXiv preprint arXiv:2005.03995, 2020.
- [76] M. Avi-Aharon, A. Arbelle, and T. Riklin Raviv, „Hue-net: Intensity-based image-to-image translation with differentiable histogram loss functions,” arXiv preprint arXiv:1912.06044, 2019.
- [77] S. Chen, B. Li, Z. Li, J. Xu, and Y. Zhang, „Auto-encoders in deep learning—a review with new insights,” Mathematics, vol. 11, no. 8, p. 1777, 2023.
- [78] Y. Chen, X. Zhang, and Y. Zhu, „Histogram-based color transfer for image stitching,” Journal of Imaging, vol. 3, no. 3, p. 38, 2017.
- [79] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, „A review on generative adversarial networks: Algorithms, theory, and applications,” IEEE

Transactions on Knowledge and Data Engineering, vol. 35, no. 4, pp. 3313–3332, 2020.

- [80] A. Aggarwal, V. Sounderajah, G. Martin, D. S. Ting, A. Karthikesalingam, D. King, H. Ashraffian, and A. Darzi, „Generative adversarial network: An overview of theory and applications,” International Journal of Data Science and Analytics, vol. 12, no. 3, pp. 245–268, 2021.
- [81] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, „Unpaired image-to-image translation using cycle-consistent adversarial networks,” in Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 2223–2232, 2017.
- [82] T. Karras, S. Laine, and T. Aila, „A style-based generator architecture for generative adversarial networks,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4401–4410, 2019.
- [83] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, „Image-to-image translation with conditional adversarial networks,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1125–1134, 2017.
- [84] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, „Normalizing flows for probabilistic modeling and inference,” Journal of Machine Learning Research, vol. 22, no. 57, pp. 1–64, 2021.
- [85] J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, and S. Ganguli, „Deep unsupervised learning using nonequilibrium thermodynamics,” in Proceedings of the 32nd International Conference on Machine Learning (ICML), vol. 37 of Proceedings of Machine Learning Research, pp. 2256–2265, PMLR, 2015.
- [86] J. Ho, A. Jain, and P. Abbeel, „Denoising diffusion probabilistic models,” in Advances in Neural Information Processing Systems, vol. 33, pp. 6840–6851, 2020.

- [87] J. Song, C. Meng, and S. Ermon, „Denoising diffusion implicit models,” arXiv preprint arXiv:2010.02502, 2020.
- [88] C. Saharia, W. Chan, H. Chang, C. A. Lee, J. Ho, T. Salimans, D. J. Fleet, and M. Norouzi, „Palette: Image-to-image diffusion models,” in SIGGRAPH ’22: Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings, 2022.
- [89] M. Wang and W. Deng, „Deep visual domain adaptation: A survey,” Neurocomputing, vol. 312, pp. 135–153, 2018.
- [90] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, „Domain randomization for transferring deep neural networks from simulation to the real world,” 2017.
- [91] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, „Domain-adversarial training of neural networks,” Journal of Machine Learning Research, vol. 17, no. 59, pp. 1–35, 2016.
- [92] J. Redmon and A. Farhadi, „Yolov3: an incremental improvement,” CoRR, vol. 1804, no. 02767, 2018. Available online.
- [93] B. Gašparović, G. Mauša, J. Rukavina, and J. Y. Lerga, „Yolov6, yolov7, and yolov8 in underwater environment: Is there real improvement?,” in Proceedings of the 2023 8th International Conference on Smart and Sustainable Technologies (SpliTech), Split/Bol, (Croatia), p. 1–4, 2023.
- [94] J. Redmon and A. Farhadi, „Yolov3: An incremental improvement.” arXiv 2018, arXiv:1804.02767. Available online.
- [95] R. Padilla, S. L. Netto, and E. A. B. da Silva, „A survey on performance metrics for object-detection algorithms,” Proceedings of the IEEE Access, vol. 9, pp. 71618–71637, 2021. Comprehensive comparison and definition of IoU, Precision, Recall, F1, and mAP.
- [96] M. Sokolova and G. Lapalme, „A systematic analysis of performance measures for classification tasks,” Information Processing

and Management, vol. 45, no. 4, pp. 427–437, 2009. Defines Precision, Recall, and F1-score metrics for evaluation.

- [97] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, „Microsoft coco: Common objects in context,” in Proceedings of the European Conference on Computer Vision (ECCV), pp. 740–755, 2014. Introduces the COCO evaluation metric $mAP@[0.5:0.95]$.
- [98] T. Erdei, Z. Molnár, N. Obinna, G. Husi, Z. Xu, X. Zhou, H. Wu, X. Li, and S. Li, „A novel design of an augmented reality based navigation system & its industrial applications,” ACTA IMEKO, vol. 7, p. 57, 2018.
- [99] D. Al-Alimi, Y. Shao, R. Feng, M. Al-qaness, M. Elaziz, and S. Kim, „Multi-scale geospatial object detection based on shallow-deep feature extraction,” Remote Sensing, vol. 11, no. 21, 2019. Available online.
- [100] Z. Xu, X. Zhou, and S. Li, „Deep recurrent neural networks based obstacle avoidance control for redundant manipulators,” Frontiers in Neurorobotics, vol. 13, p. 7, 2019.
- [101] Z. Xu, X. Zhou, H. Wu, X. Li, and S. Li, „Motion planning of manipulators for simultaneous obstacle avoidance and target tracking: An rnn approach with guaranteed performance,” IEEE Transactions on Industrial Electronics, vol. 99, pp. 1–1, 2021.
- [102] T. Kapusi, T. Erdei, G. Husi, and A. Hajdu, „Application of deep learning in the deployment of an industrial scara machine for real-time object detection,” Robotics, vol. 11, no. 4, 2022. Available online.
- [103] V. Branytskyi, M. Golovianko, D. Malyk, and V. Terziyan, „Generative adversarial networks with bio-inspired primary visual cortex for industry 4.0,” Procedia Computer Science, vol. 200, pp. 418–427, 2022.

- [104] S. Mei, W. Yudan, and G. Wen, „Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model,” Sensors, vol. 18, p. 1064, 2018.
- [105] L. Eversberg and J. Lambrecht, „Generating images with physics-based rendering for an industrial object detection task: realism versus domain randomization,” Sensors, vol. 21, p. 7901, 2021.
- [106] J. Li, P.-L. Götvall, J. Provost, and K. Åkesson, „Training convolutional neural networks with synthesized data for object recognition in industrial manufacturing,” in 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), (Zaragoza, Spain), pp. 1544–1547, 2019.
- [107] X. Yang, X. Fan, J. Wang, and K. Lee, „Image translation based synthetic data generation for industrial object detection and pose estimation,” IEEE Robotics and Automation Letters, vol. 7, no. 3, pp. 7201–7208, 2022.
- [108] J. Hu, F. Xiao, Q. Jin, G. Zhao, and P. Lou, „Synthetic data generation based on rdb-cycleGAN for industrial object detection,” Mathematics, vol. 11, p. 4588, 2023.
- [109] S. Kaji and S. Kida, „Overview of image-to-image translation by use of deep neural networks: denoising, super-resolution, modality conversion, and reconstruction in medical imaging,” Radiological Physics and Technology, vol. 12, p. 6, 2019.
- [110] J.-Y. Zhu, T. Park, P. Isola, and A. Efros, „Unpaired image-to-image translation using cycle-consistent adversarial networks,” in Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017.
- [111] M. Zamora, J. Vargas, J. Azorin-Lopez, and J. Rodríguez, „Deep learning-based visual control assistant for assembly in industry 4.0,” Computers in Industry, vol. 131, p. 103485, 2021.

- [112] L. Yu, J. Zhu, Q. Zhao, and Z. Wang, „An efficient yolo algorithm with an attention mechanism for vision-based defect inspection deployed on fpga,” Micromachines, vol. 13, no. 7, 2022. Available online.
- [113] X. Zhou, X. Xu, W. Liang, Z. Zeng, S. Shimizu, L. Yang, and Q. Jin, „Intelligent small object detection for digital twin in smart manufacturing with industrial cyber-physical systems,” IEEE Transactions on Industrial Informatics, vol. 18, no. 2, pp. 1377–1386, 2022.
- [114] P. Adarsh, P. Rathi, and M. Kumar, „Yolo v3-tiny: object detection and recognition using one-stage improved model,” 2020.
- [115] D. Arthur and S. Vassilvitskii, „K-means++: The advantages of careful seeding,” in Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), (New Orleans, LA, USA), pp. 1027–1035, Society for Industrial and Applied Mathematics, 2007.
- [116] D. Kingma and J. Ba, „Adam: A method for stochastic optimization,” 2014. Available online.
- [117] L. Perez and J. Wang, „The effectiveness of data augmentation in image classification using deep learning,” CoRR, vol. 1712, no. 04621, 2017. Online]. Available:.
- [118] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, „Efficient backprop,” in Neural Networks: Tricks of the Trade (G. B. Orr and K.-R. Müller, eds.), vol. 1524 of Lecture Notes in Computer Science, pp. 9–50, Springer, 1998.
- [119] Z. Li and S. Arora, „An exponential learning rate schedule for deep learning,” CoRR, vol. 1910, no. 07454, 2019. Online]. Available:.
- [120] M. Sampat, Z. Wang, S. Gupta, A. Bovik, and M. Markey, „Complex wavelet structural similarity: a new image similarity index,” IEEE Trans. Image Process, vol. 18, no. 11, pp. 2385–401., 2009.

- [121] A. Rikalovic, N. Suzic, B. Bajic, and V. Piuri, „Industry 4.0 implementation challenges and opportunities: A technological perspective,” IEEE Syst. J., vol. 16, p. 2797–2810.
- [122] C. Pascal, L.-O. Raveica, and D. Panescu, „Robotized application based on deep learning and internet of things,” in Proceedings of the 2018 22nd International Conference on System Theory, Control and Computing (ICSTCC, (Sinaia, Romania), p. 646–651, 2018.
- [123] A. Ayub and A. Wagner, „F-siol-310: A robotic dataset and benchmark for few-shot incremental object learning,” in Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA, (Xi’an, China), p. 13496–13502, 2021.
- [124] P. Jiang, Y. Ishihara, N. Sugiyama, J. Oaki, S. Tokura, A. Sugahara, and A. Ogawa, „Depth image–based deep learning of grasp planning for textureless planar-faced objects in vision-guided robotic bin-picking,” Sensors, vol. 20, p. 706, 2020.
- [125] A. Lobbezoo, Y. Qian, and H.-J. Kwon, „Reinforcement learning for pick and place operations in robotics: A survey,” Robotics, vol. 10, p. 105, 2021.
- [126] M. Sumanas, A. Petronis, V. Bucinskas, A. Dzedzickis, D. Virzonis, and I. Morkvenaite Vilkonciene, „Deep q-learning in robotics: Improvement of accuracy and repeatability,” Sensors, vol. 22, p. 3911, 2022.
- [127] M. Imad, O. Doukhi, D. Lee, J. Kim, and Y. Kim, „Deep learning-based nmpc for local motion planning of last-mile delivery robot,” Sensors, vol. 22, p. 8101, 2022.
- [128] K. Robotics, „Official documentation of industrial arc welder robot arm.” Accessed on 1 May 2025.
- [129] E. Chatzoglou, G. Kambourakis, and C. Smiliotopoulos, „Let the cat out of the bag: Popular android iot apps under security scrutiny,” Sensors, vol. 22, p. 513, 2022.

- [130] „Tapo c200 – pan / tilt home security wi-fi camera.” <https://www.tp-link.com/en/product/smart-camera/tapo-c200/>, 2025. Accessed: 2025-10-18.
- [131] B. O. Community, „Blender – a free and open-source 3d creation suite,” 2024. Accessed: 2025-10-18.
- [132] Y. Du, H. Sun, Q. Tian, S. Zhang, and C. Wang, „Design of blender imc control system based on simple recurrent networks,” in Proceedings of the 2009 International Conference on Machine Learning and Cybernetics, (Baoding, China), pp. 1048–1052, 2009.
- [133] T. Takala, M. Mäkäräinen, and P. Hamalainen, „Immersive 3d modeling with blender and off-the-shelf hardware,” in Proceedings of the Conference: 3D User Interfaces (3DUI), 2013 IEEE Symposium, (Orlando, FL, USA), 2013.
- [134] J. Li, L. Meng, B. Yang, C. Tao, L. Li, and W. Zhang, „An automated toolbox to make deep learning samples from remote sensing images,” Remote Sens, vol. 13, p. 12064, 2021.
- [135] Lenovo, „Thinkcentre m93 tower.” accessed on 2 September 2025).
- [136] A. Bochkovskiy, C. Wang, and H. Liao, „Yolov4: Optimal speed and accuracy of object detection.” arXiv 2020, arXiv:2004.10934. [Online]. Available online:.
- [137] C.-Y. Wang, A. Bochkovskiy, and H.-Y. Liao, „Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,” in Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), (Vancouver, BC, Canada), pp. 7464–7475, 2023.
- [138] A. Afdhal, K. Saddami, S. Sugiarto, Z. Fuadi, and N. Nasarudin, „Real-time object detection performance of yolov8 models for self-driving cars in a mixed traffic environment,” in Proceedings of the 2023 2nd International Conference on Computer System, Information Technology, and Electrical Engineering (COSITE), (Banda Aceh, Indonesia), pp. 260–265, 2023.

- [139] C.-Y. Wang, I.-H. Yeh, and H.-Y. Liao, „Yolov9: Learning what you want to learn using programmable gradient information.” arXiv 2024, arXiv:2402.13616. Available online.
- [140] NVIDIA, „Jetson nano developer kit,” 2024. Available online.
- [141] N. Corporation, „Omniverse replicator: Synthetic data generation framework,” 2024. Available online.
- [142] E. Kaigom, „Potentials of the metaverse for robotized applications in industry 4.0 and industry 5.0,” Procedia Computer Science, vol. 232, pp. 1829–1838, 2024.
- [143] D. Pasanisi, E. Rota, M. Ermidoro, and L. Fasanotti, „On domain randomization for object detection in real industrial scenarios using synthetic images,” Procedia Computer Science, vol. 217, pp. 816–825, 2023.
- [144] H. Kuhn, „The hungarian method for the assignment problem,” Naval Research Logistics Quarterly, vol. 2, pp. 83–97, 1955.
- [145] T. Yu, J. Ma, H. Yang, C. Xu, Z. Wang, and J. Liu, „Deep graph matching with channel-independent embedding and hungarian attention,” in Proceedings of the International Conference on Learning Representations (ICLR 2020), (Addis Ababa, Ethiopia), 2020.
- [146] A. Garcia-Najera and C. Brizuela, „Pcb assembly: An efficient genetic algorithm for slot assignment and component pick and place sequence problems,” in Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC’05), (Edinburgh, UK), pp. 1485–1492, 2005.
- [147] R. Ahmadi and I. Osman, „Component allocation for printed circuit board assembly using modular placement machines,” International Journal of Production Research, vol. 40, pp. 3545–3562, 2002.
- [148] Y. Crama, J. Klundert, and F. Spijksma, „Optimal feeder assignment for pcb assembly by placement machines,” International Journal of Production Research, vol. 40, pp. 223–241, 2002.

- [149] W. Li, A. Xu, M. Wei, W. Zuo, and R. Li, „Deep learning-based augmented reality work instruction assistance system for complex manual assembly,” Journal of Manufacturing Systems, vol. 73, pp. 307–319, 2024.
- [150] J. Lee, Y. Lee, J. Kim, A. Kosiorek, S. Choi, and Y. Teh, „Set transformer: A framework for attention-based permutation-invariant neural networks,” in Proceedings of the 36th International Conference on Machine Learning (ICML 2019), (Long Beach, CA, USA), pp. 3744–3753, 2019.
- [151] W. Zhou, X. Yi, C. Zhou, C. Li, Z. Ye, Q. He, X. Gong, and Q. Lin, „Feature importance evaluation-based set transformer and kan for steel plate fault detection,” IEEE Transactions on Instrumentation and Measurement, pp. 1–1, 2025.
- [152] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, „End-to-end object detection with transformers,” in Proceedings of the 16th European Conference on Computer Vision (ECCV 2020), (Glasgow, UK), pp. 213–229, 2020.
- [153] C. Jia, H. Liu, X. Wang, Y. Zhang, Z. Zhang, and L. Zhang, „Detrs with hybrid matching,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2023), (Vancouver, BC, Canada), pp. 19702–19712, 2023.
- [154] C. Alonso, D. Alvarez, F. Sánchez, and F. Vázquez, „A review on inverse kinematics, control and planning for robotic manipulators with and without obstacles via deep neural networks,” Algorithms, vol. 18, p. 23, 2025.
- [155] M. Shihab, A. Tarigan, R. Batubara, N. Yusof, and N. Rahim, „Inverse kinematics for robotic manipulators via deep neural networks: Experiments and results,” Applied Sciences, vol. 15, p. 7226, 2025.
- [156] V. Kovalchuk, A. Kasatkina, and A. Zavalniuk, „Ann approach for scara robot inverse kinematics solutions with diverse datasets and optimisers,” Applied Computer Systems, vol. 29, pp. 24–34, 2024.

- [157] R. Girshick, J. Donahue, T. Darrell, and J. Malik, „Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2014), (Columbus, OH, USA), pp. 580–587, 2014.
- [158] R. Girshick, „Fast r-cnn,” in Proceedings of the IEEE International Conference on Computer Vision (ICCV 2015), (Santiago, Chile), pp. 1440–1448, 2015.
- [159] S. Ren, K. He, R. Girshick, and J. Sun, „Faster r-cnn: Towards real-time object detection with region proposal networks,” in Proceedings of the 28th International Conference on Neural Information Processing Systems (NeurIPS 2015), (Montreal, QC, Canada), pp. 91–99, 2015.
- [160] K. He, G. Gkioxari, P. Dollár, and R. Girshick, „Mask r-cnn,” in Proceedings of the IEEE International Conference on Computer Vision (ICCV 2017), (Venice, Italy), pp. 2961–2969, 2017.
- [161] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, „You only look once: Unified, real-time object detection,” 2016.
- [162] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. Berg, „Ssd: Single shot multibox detector,” in Proceedings of the European Conference on Computer Vision (ECCV 2016), (Amsterdam, The Netherlands), pp. 21–37, 2016.
- [163] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, „Focal loss for dense object detection,” in Proceedings of the IEEE International Conference on Computer Vision (ICCV 2017), (Venice, Italy), pp. 2980–2988, 2017.
- [164] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, and Y. Xu, „Object detection with transformers: A review.” arXiv 2023, arXiv:2306.04670. Available online.

- [165] F. Li, H. Zhang, S. Liu, J. Guo, L. Ni, C. Zhang, B. Ni, L. Wang, H. Lu, and H. Hu, „Dn-detr: Accelerate detr training by introducing query denoising,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022), (New Orleans, LA, USA), pp. 13619–13627, 2022.
- [166] W. Lv, G. Song, H. Yu, C. Ma, Y. Pang, C. Zhang, and Y. Wei, „Dino-detr: Detr with improved denoising anchor boxes for end-to-end object detection,” in Proceedings of the International Conference on Learning Representations (ICLR 2023), (Kigali, Rwanda), 2023.
- [167] Y. Wei, W. Lv, C. Ma, Y. Liu, C. Zhang, Y. Zhang, Y. Pang, and G. Song, „Rt-detr: Real-time detr with efficient hybrid encoder.” arXiv 2023, arXiv:2304.08069. Available online.
- [168] Z. Zhao, J. Jia, W. Lv, C. Ma, Y. Pang, G. Song, and Y. Wei, „Detrs beat yolos on real-time object detection,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2024), (Seattle, WA, USA), pp. 16965–16974, 2024.
- [169] W. Falcon and T. P. L. Team, „Pytorch lightning: A lightweight pytorch wrapper for high-performance ai research,” in NeurIPS 2019 Workshop on ML Systems, 2019. Available online.
- [170] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, „Optuna: A next-generation hyperparameter optimization framework,” in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '19, (Anchorage, AK, USA), pp. 2623–2631, Association for Computing Machinery, 2019.
- [171] P. A. Studios, „Universal scene description (usd) specification,” 2016. Available online.
- [172] N. Corporation, „Omniverse isaac sim: Robotics simulation platform,” 2024. Available online.

[173] P. A. Studios, „Usd xform schema: Transforming prims,” 2016.
Available online.