



Diplomová práce

Integrace průmyslového robota do modelu výrobní linky

Studijní program:

N0714A270010 Mechatronika

Autor práce:

Bc. Pavel Müller

Vedoucí práce:

doc. Ing. Josef Černožorský, Ph.D.

Ústav mechatroniky a technické informatiky

Liberec 2024



Zadání diplomové práce

Integrace průmyslového robota do modelu výrobní linky

Jméno a příjmení:

Bc. Pavel Müller

Osobní číslo:

M22000049

Studijní program:

N0714A270010 Mechatronika

Zadávající katedra:

Ústav mechatroniky a technické informatiky

Akademický rok:

2023/2024

Zásady pro vypracování:

1. Seznamte se koncepcí řízení modelu průmyslové linky Festo.
2. Vytipujte vhodnou stanici pro nasazení průmyslového robota.
3. Proveďte systémovou integraci.
4. Ověřte funkčnost vašeho řešení.

Rozsah grafických prací: dle potřeby dokumentace
Rozsah pracovní zprávy: 40 až 50 stran
Forma zpracování práce: tištěná/elektronická
Jazyk práce: čeština

Seznam odborné literatury:

- [1] NOVOTNÝ, František; HOTAŘ, Vlastimil; HORÁK, Marcel; STARÁ, Marie a STARÝ, Michal. Úvod do automatizace a robotizace ve strojírenství. Liberec: Technická univerzita v Liberci, Fakulta strojní, Katedra sklářských strojů a robotiky, 2020. ISBN 978-80-7494-545-8.
- [2] KOLÍBAL, Zdeněk. *Roboty a robotizované výrobní technologie*. Brno: Vysoké učení technické v Brně – nakladatelství VUTIUM, 2016. ISBN 978-80-214-4828-5
- [3] NOVOTNÝ, František a Marcel HORÁK. *Efektory průmyslových robotů*. Liberec: Technická univerzita v Liberci, 2015. ISBN 978-80-7494-195-5

Vedoucí práce: doc. Ing. Josef Černožorský, Ph.D.
Ústav mechatroniky a technické informatiky

Datum zadání práce: 12. října 2023
Předpokládaný termín odevzdání: 14. května 2024

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

L.S.

doc. Dr. Ing. Jaroslav Hlava
garant studijního programu

V Liberci dne 12. října 2023

Prohlášení

Prohlašuji, že svou diplomovou práci jsem vypracoval samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Jsem si vědom toho, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má diplomová práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

Integrace průmyslového robota do modelu výrobní linky

Abstrakt

Tato diplomová práce zkoumá integraci robotického systému a kamerového inspekčního systému v rámci Festo Modulárního výrobního systému (MPS). Cílem výzkumu je rozšířit automatizované procesy zavedením pokročilých vizuálních inspekčních schopností, čímž by mohla být zvýšena přesnost a efektivita operací MPS.

Klíčová slova: Universal robot 3, Modulární výrobní systém, kamera, Festo, nástroj robota, zpracování obrazu

Using of industrial robot in model of production facility

Abstract

This master's thesis investigates the integration of a robotic system and a camera inspection system within a Festo Modular Production System (MPS). The research aims to augment automated processes by introducing advanced visual inspection capabilities, potentially enhancing both the precision and efficiency of MPS operations.

Keywords: Universal robot 3, Modular Production System, camera, Festo, robot tool, image processing

Poděkování

Rád bych poděkoval svému vedoucímu diplomové práce, doc. Ing. Josefu Černožorskému, Ph.D., jehož vedení a podpora byly klíčové pro dokončení této práce. doc. Ing. Josef Černožorský, Ph.D. nejen poskytl odborné rady, ale také usnadnil pořízení potřebného vybavení a přispěl 3D tiskem zásadních částí, které byly nezbytné pro dokončení této práce

Kromě toho jsem vděčný Bc. Romanu Dolenskému za aktivní účast a spolupráci.

Obsah

Seznam zkratek	13
1 Úvod	14
2 Rozbor možných řešení	15
2.1 Manipulator	15
2.2 Inspekce	16
3 Robot Universal robot 3	17
3.1 Kinematické vlastnosti	17
3.2 Dynamické vlastnosti	18
3.3 Elektrické vlastnosti	19
3.4 Způsoby ovládání robota	21
3.5 Aplikace	22
4 Efektory průmyslových robotů	23
4.1 Manipulační nástroje	24
4.1.1 Mechanická chapadla	25
4.1.2 Elektromagnetická chapadla	26
4.1.3 Vakuová chapadla	27
5 Inspekční systémy	28
5.1 Kamerové inspekční systémy	28
5.2 Laserové inspekční systémy	30
6 Modelová výrobní linka	32
6.1 Distribuční stanice	32
6.2 Spojovací stanice	34
6.3 Třídící stanice	36
6.4 Integrace robota k Modulárnímu výrobnímu systému	37
6.5 Funkce stroje	38
7 Komunikace mezi prvky	40
7.1 Komunikace s PLC	41

8	Pohyb robotem	42
8.1	Zpětná vazba	42
8.2	Použití Handshaku	43
8.3	Pozice robota	44
9	Nástroj robota	46
9.1	Pneumatický řetězec	46
9.2	Konečný návrh	47
10	Kamera	48
10.1	Poloha pro inspekci	49
10.2	Osvětlení	49
11	Předzpracování snímku	51
11.1	Hranová segmentace	51
11.1.1	Hranové detektory	52
11.1.2	Použití hranové detekce k segmentaci obrazu	53
12	Metody kontroly kvality	56
12.1	Průměrná hodnota RGB	56
12.2	Průměrná hodnota odstínů šedi a odstínu	57
12.3	Nekompaktnost	58
12.4	Minimum Average Correlation Energy	59
12.5	Zhodnocení metod	62
13	Závěr	64
	Použitá literatura	66
	Přílohy	67
N	Schéma zapojení	68
O	Testované obrázky	69
P	Zdrojové kódy	73
P.1	Robot	73
P.2	Zpracování obrazu	78

Seznam obrázků

3.1	Robot UR3	17
3.2	Graf zatížení [1]	19
3.3	Konektor nástroje [1]	20
3.4	Konfigurace I/O	21
3.5	Programování pomocí Teach pendant	21
4.1	Příklad paralelních chapadel [5]	25
4.2	Příklad prstových chapadel [6]	26
4.3	Příklad elektromagnetických chapadel [7]	26
4.4	Ukázka funkce vakuových savek	27
4.5	Příklad přísavek [[8], [9]]	27
5.1	Ukázka kamer	29
5.2	Ukázka kamer	30
6.1	Distribuční stanice	33
6.2	Moduly distribuční stanice	33
6.3	Spojovací stanice	34
6.4	Moduly spojovací stanice	35
6.5	Třídící stanice	36
6.6	Moduly třídící stanice	36
6.7	Uspořádání stoje	38
6.8	OK výrobky	38
6.9	Diagram průchodu materiálu strojem	39
7.1	Komunikace	40
8.1	Diagram zpětné vazby	43
8.2	Flowchart pohybu robota	45
9.1	Model nástroje	46
9.2	Vakuový ejektor	47
9.3	Konečný návrh nástroje	47
10.1	Basler acA 1300 30gc	48
10.2	Vyfocené výrobky	49
10.3	Vyfocené výrobky bez světla	50

10.4	Osvětlení	50
11.1	Výsledek Matlab Sobelova filtru	51
11.2	Použití masky	53
11.3	Postup segmentace	54
11.4	Segmentované výrobky	55
12.1	Nekompaktnost	58
12.2	Výsledné hodnoty nekompaktnosti	59
12.3	MACE filter	60
12.4	3d spektrum	61
12.5	2d spektrum	61
12.6	Oblast zhodnocení	62
12.7	NOK výrobky	62
N.1		68
O.1	Černý výrobky	69
O.2	Černý klasifikační obrázky	69
O.3	Červený výrobky	70
O.4	Červený klasifikační obrázky	70
O.5	Stříbrný výrobky	71
O.6	Stříbrný klasifikační obrázky	71
O.7	Žlutý NOK výrobek	72
O.8	Bílý NOK výrobek	72

Seznam tabulek

7.1	Rozdělení IP adres	40
7.2	Zapojení signálů	41
7.3	Zapojení signálů	41
11.1	Příklad konvolučních kernelů	53
12.1	Konverze mezi barevnými prostory	57
12.2	Zhodnocení metod pro inspekci barvy	63
12.3	Zhodnocení metod pro inspekci tvaru	63

Seznam zdrojových kódů

3.1	Příklad příkazu movej [2]	22
3.2	Příklad vyčtení aktuálního natočení kloubů [2]	22
8.1	Možnosti použití funkce	43
P.1	Handshake	73
P.2	kompletní program	74
P.3	Nastavení parametrů kamery	78
P.4	Zpracování obrazu	78

Seznam zkratek

UR3	Universal robot 3
OK	Conforming - Vyhovující výrobek
NOK	Non-Conforming - Nevyhovující výrobek
MPS	Modular Production System - Modulární výrobní systém
GUI	Graphical User Interface - grafické uživatelské rozhraní
TCP/IP	Transmission Control Protocol/Internet Protocol - protokol pro řízení přenosu/protokol pro propojení sítí
GigE	Gigabit Ethernet
HSV	Hue, Saturation, Value - odstín, saturace, jas
MACE	Minimum Average Correlation Energy - Minimální průměrná energie korelace
TP	True Positive - Pravý pozitiv
TN	True Negative - Pravý negativ
FP	False Positive - Falešný pozitiv
FN	False Negative-Falešný negativ
DFT	Discrete Fourier transform - Diskrétní fourierova transformace

1 Úvod

V rámci průmyslu 4.0 představuje integrace manipulačních a inspekčních systémů do výrobních linek významný krok směrem k pokročilé výrobě. Využitím synergie mezi precizní automatizací poskytovanou manipulátory a vysokou přesností nejmodernějších inspekčních technologií mohou výrobci dosáhnout bezproblémového a vysoce efektivního výrobního procesu. Tato konvergence nejen zlepšuje kontrolu kvality a snižuje provozní náklady, ale také výrazně zvyšuje rychlost a flexibilitu.

Tato diplomová práce zkoumá integraci robota Universal Robots UR3 a kamerového inspekčního systému v rámci Festo Modular Production System (MPS). Hlavním cílem je zajistit manipulaci s výrobky robotem, což umožní jejich přenos mezi určenými pozicemi pro vyzvednutí, inspekci, umístění a třídění nevyhovujících (NOK) výrobků. Vzhledem k tomu, že výrobky mohou být buď vyhovující (OK), nebo nevyhovující (NOK), je inspekční systém nezbytný k jejich rozlišení. Výrobky určené k inspekci jsou plastové kryty dostupné ve třech barvách: černé, červené a stříbrné.

Úvodní fáze této diplomové práce se zaměřuje na výběr vhodných modulů MPS. Vzhledem k rozsáhlému sortimentu dostupných modulů MPS je nezbytná analýza. Tato analýza identifikuje moduly nejvhodnější pro integraci na základě jejich schopností, kompatibility s dalšími komponentami systému a specifických požadavků zamýšlených aplikací. Proces výběru si klade za cíl zajistit, že vybrané moduly efektivně podpoří a rozšíří funkčnosti integrovaného systému.

Po identifikaci optimálních modulů následuje plynulá integrace robota UR3 s těmito moduly. Tato fáze zahrnuje zavedení komunikace mezi všemi zařízeními zapojenými do systému, aby bylo zajištěno synchronizované fungování a výměna dat. Dále tato fáze zahrnuje upevnění robota k modulům.

Následně bude navržen specializovaný robotický nástroj, který splňuje operační potřeby. Po vývoji nástroje bude vytvořen program pro zpracování obrazu. Tento software bude mít za úkol inspekci výrobků s využitím algoritmů pro hodnocení kvality, detekci vad nebo potvrzení specifikací na základě vizuálních dat získaných během inspekčních procesů.

Tato práce vznikla ve spolupráci s kolegou Bc. Romanem Dolenským, který měl na starost programování MPS modulů.

2 Rozbor možných řešení

Počáteční analýza potenciálních řešení je klíčová pro stanovení jasného směru vývoje projektu. V této diplomové práci bylo hlavním cílem předběžné analýzy výběr nejvhodnějšího manipulátoru a inspekčního systému. Tato základní analýza měla za cíl identifikovat klíčové specifikace a schopnosti potřebné pro efektivní integraci těchto systémů do Festo MPS.

Zaměřením na tyto klíčové komponenty pomohla analýza nastínit technické a funkční požadavky, čímž zajistila, že vybraná řešení budou v souladu s celkovými cíli diplomové práce. Tento přístup nejen nastavuje trajektorii projektu, ale také zajišťuje, že následné fáze vývoje jsou založeny na důkladném porozumění potřebám systému.

2.1 Manipulator

Vzhledem k provozním požadavkům na manipulaci s pracovními kusy je nezbytné zvolit vhodný manipulátor. Škála dostupných manipulátorů sahá od jednoduchých dopravníkových pásů po složitější robotické systémy, včetně hydraulických a pneumatických možností. Každý typ manipulátoru nabízí specifické výhody v závislosti na konkrétních potřebách aplikace, jako je přesnost, nosnost a adaptabilita.

Například hydraulické systémy poskytují významný výkon a jsou schopny manipulovat s těžkými břemeny, zatímco pneumatické systémy nabízejí jednoduchost a rychlost, což je výhodné pro lehčí a opakované úkoly. Nicméně s ohledem na dostupné vybavení laboratoře byl pro tento projekt vybrán kloubový robot. Kloubové roboty nabízejí flexibilitu a přesnost, což je činí vhodnými pro složité a různorodé manipulační úkoly.

Volba kloubového robota je dále ospravedlněna jeho kompatibilitou se stávající infrastrukturou a jeho schopností být naprogramován pro vysoce specifické úkoly, čímž zvyšuje efektivitu a účinnost procesu manipulace s pracovními kusy.

Vybraný robot je UR3 od společnosti Universal Robots, který bude ovládán prostřednictvím předem vytvořená knihovny v Matlabu. Vzhledem k tomu, že tato knihovna ve své původní verzi postrádá mechanismus zpětné vazby k potvrzení dokončení pohybu, bude nutné knihovnu upravit a tuto funkčnost do ní začlenit. Jakmile budou tyto úpravy implementovány, mohou být definovány specifické pohybové pozice pro robota. Následně bude modelován a vyvinut nástroj pro robota, který umožní robotu provádět zamýšlené úkoly.

2.2 Inspekce

V oblasti automatizované inspekce je k dispozici široká škála zařízení, která by mohla splnit požadavky úkolu, včetně pokročilých kamer a vizuálních senzorů. Navíc jeden z druhů výrobku má pokovený povrch, to by umožňovalo použití indukčních senzorů pro jejich detekci.

Nicméně v rozsahu této diplomové práce bylo explicitně definován použití kamerové technologie pro inspekční účely. Kritéria pro výběr kamery byla poměrně flexibilní, umožňující široký rozsah modelů od špičkových kamerových systémů až po dostupnější webové kamery. Zásadní požadavky pro vybranou kameru byly schopnosti detekce RGB barev a možnost propojení kamery s programovacím softwarem. Tato flexibilita zajišťuje, že kamera může zachytit detailní vizuální data nezbytná pro důkladnou inspekci a že může být bezproblémově integrována do automatizovaného systému řízeného prostřednictvím softwaru.

Nakonec vzhledem k její dostupnosti pro inspekční systém byla vybrána kamera modelu Basler ACE 1300-30gc. Vzhledem k tomu, že tato kamera nedisponuje zabudovaným softwarem pro kontrolu kvality, bude vyvinut vlastní software. Tento software bude obsahovat metody pro detekci barvy a tvaru výrobků a jejich vyhodnocení na základě zjištěných atributů. Tyto metody budou systematicky hodnoceny, aby se posoudil jejich výkon a účinnost při rozlišování mezi OK a NOK pracovními výrobky.

V počáteční úvaze inspekčního softwaru měl být naprogramován v Pythonu s využitím knihovny OpenCV, která je známá svými výkonnými schopnostmi zpracování obrazu. Nicméně s ohledem na potřeby integrace s robotickým systémem, který je již programován v Matlabu, bylo rozhodnuto vyvinout inspekční software pro kameru také v Matlabu. Toto rozhodnutí bylo motivováno výhodami udržování konzistentního programovacího prostředí mezi komponentami řízení robotů a inspekce kamerou. Tato integrace zjednodušuje spolupráci mezi různými částmi systému a umožňuje efektivnější sdílení dat a funkcionality mezi robotem a inspekčním systémem.

3 Robot Universal robot 3

Universal robot 3 (UR3) robot, vyvinutý společností Universal Robots, představuje významný pokrok v technologii kolaborativních robotů. Je navržen pro drobné automatizační úkoly, jeho lehká a kompaktní konstrukce umožňuje bezproblémovou integraci do stávajících pracovních procesů, často přímo ve spolupráci s lidskými operátory.



Obrázek 3.1: Robot UR3

Veškeré technické specifikace robota byly převzaty z manuálu robota [1].

3.1 Kinematické vlastnosti

Kinematický design robota UR3 je základním prvkem, který mu umožňuje provádět přesné a všestranné pohyby, nezbytné pro různé aplikace v průmyslových i výzkumných prostředích. UR3 je vybaven šesti rotačními klouby, z nichž každý přispívá k celkové obratnosti a schopnosti manipulace s objekty v jeho operačním prostoru. Dosah robota je 500 mm.

Konfigurace kloubů

- Stupně volnosti
- Rozsah pohybu

Stupně volnosti: Každý z šesti kloubů UR3 poskytuje jediný rotační stupeň volnosti, což robotu umožňuje orientovat jeho koncový efektor v jakékoli pozici v dosahu. To je klíčové pro úkoly vyžadující vysokou obratnost a přesnost.

Rozsah pohybu: Klouby se typicky mohou otáčet o $\pm 360^\circ$, což poskytuje výjimečnou flexibilitu. Nástrojový kloub se může otáčet o $\pm \infty$. Tento rozsáhlý rozsah pohybu je obzvláště výhodný v aplikacích, kde musí robot navigovat kolem překážek nebo pracovat v těsných prostorech..

Přesnost a opakovatelnost

- Opakovatelnost
- Rychlost a reaktivita

Opakovatelnost: Jednou z výrazných vlastností robota UR3 je jeho vysoká opakovatelnost $\pm 0,1$ mm, což zajišťuje, že se může vrátit na konkrétní pozici s vysokou přesností. To je zásadní pro aplikace vyžadující konzistentní kvalitu a přesnost, jako je montáž elektroniky nebo drobných mechanických výrobků.

Rychlost a reaktivita: Klouby jsou navrženy tak, aby rychle a hladce reagovaly na řídicí vstupy, což je nezbytné pro udržení vysoké produktivity a efektivity v dynamických provozních prostředích.

3.2 Dynamické vlastnosti

Dynamické vlastnosti robota UR3 odkazují na jeho schopnost řídit a přizpůsobovat své pohyby a operace v reakci na vnější síly, točivé momenty a různé zatížení. Tyto schopnosti jsou zásadní pro zajištění bezpečného a efektivního plnění úkolů robota, zejména v prostředích, kde je častá interakce s lidmi a dalšími stroji.

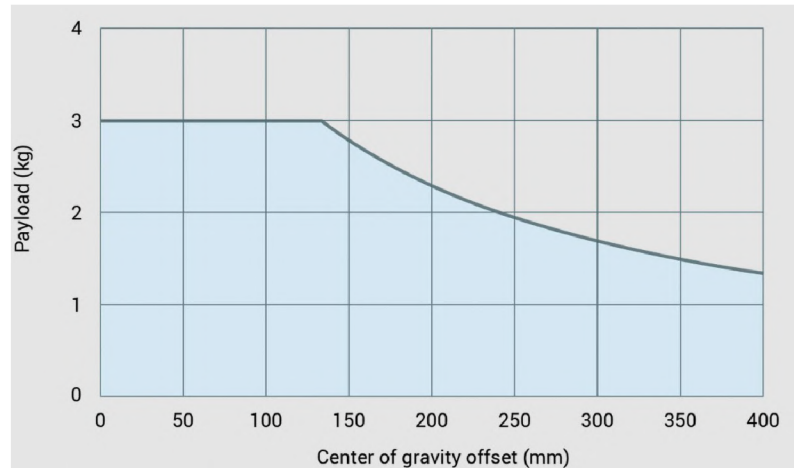
Zacházení se silou a zatížením

- Kapacita zatížení
- Citlivost na sílu

Kapacita zatížení:

UR3 je navržen tak, aby zvládal zatížení až do 3 kg, což je vhodné pro širokou škálu lehkých průmyslových úkolů, jako je sběr, umístování a montáž. Schopnost přesně zvládat různé zatížení je klíčová pro udržení kvality a konzistence operací.

Citlivost na sílu: UR3 je vybaven snímači síly, které umožňují detekci a reakci na vnější síly. Tato citlivost umožňuje provádění úkolů vyžadujících jemný dotek, jako je leštění, broušení nebo montáž křehkých komponent.



Obrázek 3.2: Graf zatížení [1]

Kontrola točivého momentu a rychlosti

- Snímače točivého momentu kloubů
- Nastavení rychlosti

Snímače točivého momentu kloubů: Každý kloub je vybaven snímači točivého momentu, které poskytují zpětnou vazbu v reálném čase o silách působících na robota a od robota. Tyto informace jsou nezbytné pro přesnou kontrolu točivého momentu, což zajišťuje, že robot pracuje v bezpečných a efektivních parametrech.

Nastavení rychlosti: UR3 může dynamicky upravovat svou rychlost na základě požadavků úkolu a vnějších podmínek. Pro složité úkoly, které vyžadují vysokou přesnost, může robot zpomalit, aby zajistil přesnost, nebo zrychlit, když je potřeba menší přesnost, čímž optimalizuje produktivitu.

3.3 Elektrické vlastnosti

Robot UR3 funguje efektivně v rozsahu napětí 100-240 VAC, přičemž během typického provozu spotřebuje průměrný výkon 200 wattů. Tato nízká energetická spotřeba činí z UR3 ekonomickou volbu pro firmy. Elektrický systém robota je navržen tak, aby zvládal výkyvy a poskytoval konzistentní výkon i za různých podmínek napájení.

Jeho elektrická architektura zahrnuje různé vstupy a výstupy pro usnadnění integrace s externím zařízením a senzory, čímž rozšiřuje jeho funkčnost a adaptabilitu na různé úkoly. Bezpečnostní prvky jsou na prvním místě, s vestavěným monitorováním proudu a možnostmi nouzového zastavení, které zajišťují jeho provoz v bezpečných parametrech a předcházejí nehodám a poškození. Signály I/O lze rozdělit do těchto skupin:

- Digitální I/O

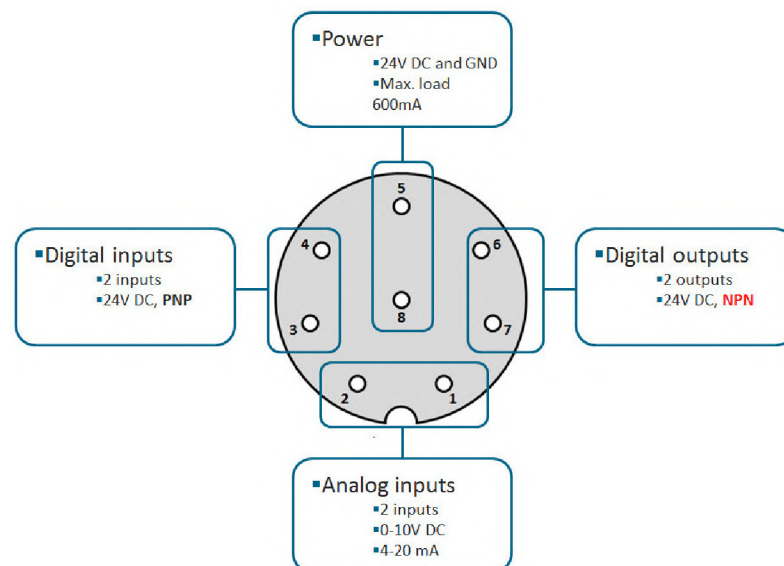
- Analogové I/O
- Bezpečnostní I/O
- I/O nástroje
- Konfigurovatelné I/O

Digitální I/O se vztahují na binární vstupní a výstupní signály, které UR3 používá pro jednoduchou kontrolu zap/vyp a rozhodování.

Analogové I/O zpracovávají proměnlivé signály.

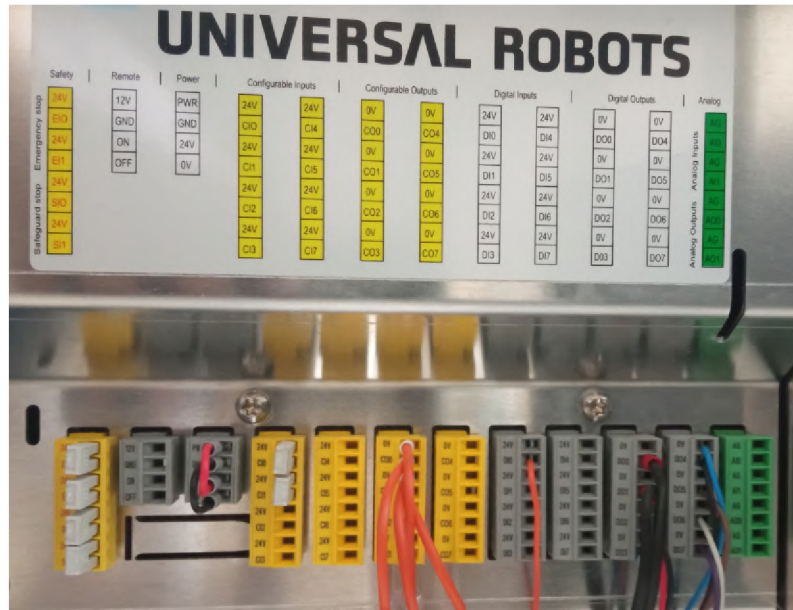
Bezpečnostní I/O jsou klíčové pro integraci bezpečnostních funkcí do operací robota, čímž zajišťuje jeho soulad s předpisy o bezpečnosti na pracovišti a bezpečný provoz vedle lidí.

I/O nástroje jsou specificky věnovány řízení a ovládání nástrojů připojených k koncovému efektoru robota. To je nezbytné pro úkoly vyžadující synchronizaci provozu nástroje s pohyby robota.



Obrázek 3.3: Konektor nástroje [1]

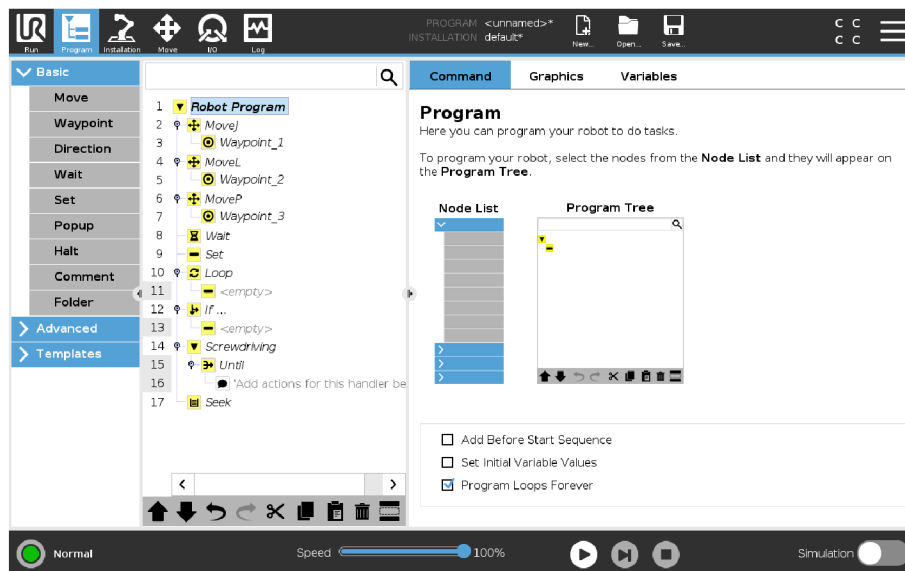
Konfigurovatelné I/O poskytují flexibilitu v tom, jak je každý I/O port použit, což umožňuje přizpůsobení na základě konkrétních požadavků úkolu nebo změn v provozním nastavení.



Obrázek 3.4: Konfigurace I/O

3.4 Způsoby ovládání robota

Robot UR3 lze ovládat několika způsoby, které vyhovují uživatelům s různou úrovní technické zkušenosti. Primární metodou je použití teach pendant. To je přenosné zařízení, které umožňuje operátorům programovat robota fyzickým pohybem jeho ramen do požadovaných pozic a zaznamenáváním těchto pohybů. Pro pokročilejší ovládání mohou uživatelé využít Polyscope GUI, proprietární programovací rozhraní společnosti Universal Robots.



Obrázek 3.5: Programování pomocí Teach pendant

Kromě toho UR3 podporuje programování prostřednictvím softwaru třetích stran a vlastních skriptů, které lze psát v programovacích jazycích, jako jsou Python, C++ nebo Matlab. Tyto programovací jazyky lze také použít k dálkovému ovládní robota přes TCP/IP spojení prostřednictvím Ethernetu.

[2]

Zdrojový kód 3.1: Příklad příkazu movej [2]

```
1 function pohybj(client,q,a,v,t,r)
2     q = q*pi/180;
3     cmd = ['movej([' ,num2str(q(1)) ,',' ,num2str(q(2)) ,',' ,
4           ,num2str(q(3)) ,',' ,num2str(q(4)) ,',' ,num2str(q
5           (5)) ,',' ,num2str(q(6)) ,'],' ,t=' ,num2str(t) ,',' ,r=' ,
6           num2str(r) ,')'];
7         writeline(client,cmd);
8     end
9 end
```

Zdrojový kód 3.2: Příklad vyčtení aktuálního natočení kloubů [2]

```
1 function output = kloub_natoceni_skut(client);
2     client = tcpclient(client.Address, client.Port);
3     offset = 30;
4     response = flip(read(client,UR3_lib.bytes));
5     for k = 1:1:6
6         count = length(response) - 4 -((k+offset)
7             *8);
8         item = response(count-7:count);
9         output(k) = typecast(uint8(item), 'double')
10            *180/pi;
11     end
12 end
```

3.5 Aplikace

Aplikace robota UR3 zasahují do různých průmyslových odvětví díky jeho všestrannosti, bezpečnosti a snadnému používání. Ve výrobě plní úkoly, jako je přesná montáž, pájení a malování, kde je jeho jemná kontrola a konzistentní výkon klíčové. Mimo průmyslové prostředí se UR3 nachází také ve výzkumných laboratořích pro automatizaci opakujících se úkolů, jako je manipulace s vzorky, které vyžadují vysokou přesnost a spolehlivost.

Jeho schopnost bezpečně pracovat vedle lidí bez potřeby rozsáhlých bezpečnostních uzávěr ho činí ideálním pro malé dílny a firmy, které se poprvé pouštějí do automatizace. Adaptabilita UR3 se rovněž rozšiřuje na kreativnější aplikace, jako je provádění živých demonstrací v učebnách.

4 Efektory průmyslových robotů

Koncové efektory robotů, jsou jednou z nejdůležitějších částí robota, které umožňují správné plnění jeho určených úkolů. Jsou to příslušenství nebo nástroje, které jsou montovány na konci robotického ramene k vykonávání specifických úkolů. Tyto nástroje se mohou výrazně lišit v závislosti na aplikaci a úkolech, které robot musí vykonávat:

- Nástroje pro zpracování materiálu
- Nástroje pro manipulaci materiálu
- Montážní nástroje
- Nástroje pro kontrolu a zajištění kvality

Nástroje pro zpracování materiálu jsou navrženy tak, aby modifikovaly fyzikální vlastnosti materiálů. Zahrnují svářecí nástroje, řezací nástroje, brusné nástroje, odstraňovače otřepů a malířské nástroje. Používají se ve výrobních procesech pro spojování materiálů, tvarování, odstranění nedokonalostí nebo aplikaci povrchových úprav. [3]

Nástroje pro manipulaci materiálu slouží k sbírání, umístování, přesouvání a balení objektů. Patří sem nástroje pro manipulaci, balení a paletizaci. Jsou zásadní pro automatizaci opakovaných úkolů spojených s manipulací s materiálem a zlepšení efektivity ve skladech, distribučních centrech a výrobních zařízeních. [3]

Montážní nástroje jsou určeny k bezpečnému a přesnému spojení částí. Mezi tyto nástroje patří upevňovací nástroje (šrouby, matice, šroubováky, nýty) a dávkovací nástroje (lepiva, těsnící hmoty, barvy). Jsou nezbytné pro montáž produktů v různých průmyslových odvětvích, zajišťují přesné zarovnání a správné spojení komponent. [3]

Nástroje pro kontrolu a zajištění kvality slouží k ověřování kvality produktů, rozměrů a dodržování specifikací. Tyto nástroje zahrnují kontrolní nástroje, jako jsou kamery, senzory a sondy, stejně jako měřicí nástroje, jako jsou posuvné měřidla a měřicí přístroje. Hrají klíčovou roli při zajišťování konzistence produktů, spolehlivosti a souladu s průmyslovými standardy. [3]

Kromě konkrétního použití robota je třeba zvážit další důležité faktory pro výběr správného koncového nástroje. Mezi tyto faktory patří:

- Minimální hmotnost

- Přesnost
- Spolehlivost a trvanlivost
- Rozměry a prostorové uspořádání
- Bezpečnost
- Nákladová efektivita

Minimální hmotnost Hmotnost nástroje robota společně s hmotností objektů, které manipuluje, hraje klíčovou roli pro správnou funkci a dlouhodobou životnost robotických systémů. [3]

Přesnost Koncové nástroje by měly poskytovat potřebnou přesnost pro provádění úkolů s požadovanou úrovní kvality. To je obzvláště důležité pro úkoly, jako je montáž, inspekce a obrábění, kde jsou nezbytné těsné tolerance. [3]

Spolehlivost a trvanlivost Koncové nástroje by měly být spolehlivé a trvanlivé, aby vydržely náročné podmínky nepřetržitého provozu v průmyslových prostředích. Měly by být vyrobeny z vysoce kvalitních materiálů a navrženy tak, aby minimalizovaly opotřebení a potřebu údržby. [3]

Rozměry a prostorové uspořádání ovlivňují manipulační možnosti robotu a mají vliv na silové zatěžování výstupního členu robotu. [3]

Bezpečnost Koncové nástroje musí splňovat bezpečnostní normy a předpisy, aby zajistily bezpečnost operátorů, dalšího zařízení a pracovního prostředí. To může zahrnovat funkce jako ochranné kryty, tlačítka pro nouzové zastavení a bezpečnostní závory. [3]

Nákladová efektivita Koncové nástroje by měly poskytovat dobrý poměr cena/výkon z hlediska počátečních nákladů, provozních nákladů a celkového návratu investic. To zahrnuje faktory jako cena nákupu, náklady na údržbu, spotřeba energie a zvýšení produktivity. [3]

4.1 Manipulační nástroje

Design koncového efektoru závisí na faktorech, jako jsou tvar, velikost, hmotnost a povrchová textura objektů, které má manipulovat. Některé koncové efektorů jsou jednoduché a tuhé, zatímco jiné jsou složitější a adaptabilní, schopné přizpůsobit se tvaru uchopovaného objektu. Koncové efektorů mohou obsahovat zařízení jako senzory, aktuátory a zpětnovazební systémy, které zlepšují jejich výkon a všestrannost. Například některá chapadla používají hmatové senzory k detekci přítomnosti a polohy objektů, zatímco jiná používají zpětnou vazbu síly k přizpůsobení síly úchopu podle hmotnosti nebo křehkosti objektu. Běžné typy koncových efektorů zahrnují paralelní chapadla, prstová chapadla, vakuová chapadla, magnetická chapadla, adaptabilní chapadla a specializovaná chapadla pro unikátní úkoly.

4.1.1 Mechanická chapadla

Mechanická chapadla se vyskytují v různých typech a konfiguracích, každé je navrženo pro specifické aplikace a požadavky. Design koncového efektoru závisí na faktorech, jako jsou tvar, velikost, hmotnost a povrchová textura objektů, které má manipulovat, stejně jako na požadované přesnosti a rychlosti úkolu. Některé koncové efektorů jsou jednoduché a tuhé, zatímco jiné jsou složitější a adaptabilní, schopné přizpůsobit se tvaru uchopovaného objektu. Běžné typy mechanických chapadel zahrnují:

- Paralelní Chapadla
- Prstová Chapadla
- Specializovaná Chapadla

Paralelní Chapadla se skládají ze dvou nebo více paralelních čelistí, které se pohybují proti sobě, aby uchopily objekt. Jsou všestranné a schopné manipulovat s širokou škálou tvarů a velikostí objektů. Paralelní chapadla se běžně používají v operacích vyzvednutí a umístění, montážních úkolech a aplikacích manipulace s materiálem. [4]



Obrázek 4.1: Příklad paralelních chapadel [5]

Prstová Chapadla mají prsty, které se zavírají kolem objektu, aby jej bezpečně uchopily. Často se používají pro jemné manipulační úkoly nebo pro uchopení objektů s nepravidelnými tvary. Prstová chapadla mohou být navržena s tuhými prsty, kompatibilními materiály nebo adaptabilními mechanismy pro různé objekty a aplikace. [4]



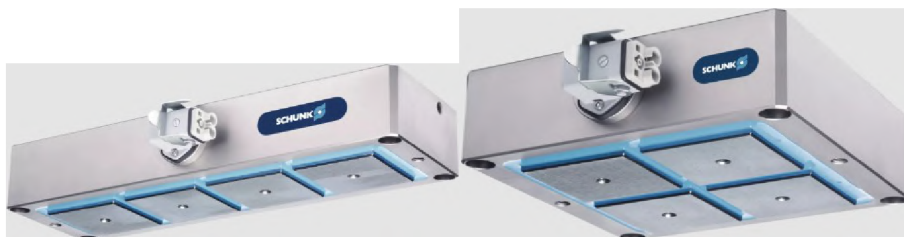
Obrázek 4.2: Příklad prstových chapadel [6]

Specializovaná Chapadla jsou navržena pro specifické úkoly a prostředí. Mohou zahrnovat chapadla pro manipulaci s křehkými objekty, chapadla pro manipulaci s nepravidelně tvarovanými objekty a chapadla pro použití v nebezpečných nebo extrémních podmínkách. [4]

4.1.2 Elektromagnetická chapadla

Elektromagnetické koncové efekty robotů využívají elektromagnetická pole k uchopení a manipulaci s feromagnetickými objekty. Tyto koncové efekty obvykle obsahují elektromagnety, zdroje napájení, řídicí elektroniku a úchopové plochy. [3]

Elektromagnety v koncovém efektoru generují magnetická pole, když jimi prochází elektrický proud. Selektivním aktivováním nebo deaktivováním elektromagnetů může koncový efektor ovládat sílu a směr magnetického pole. To umožňuje koncovému efektoru s přesností přitahovat nebo odpuzovat feromagnetické objekty. [3]

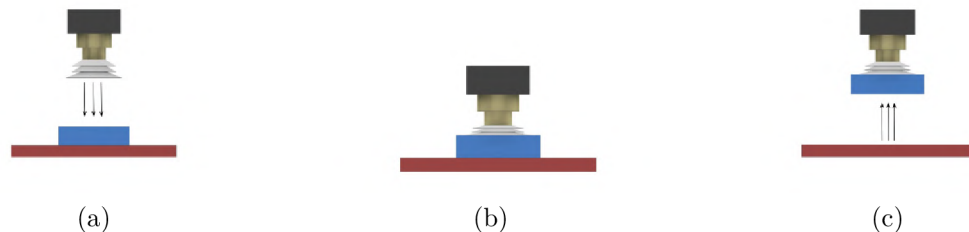


Obrázek 4.3: Příklad elektromagnetických chapadel [7]

Hlavním problémem efektivní aplikace magnetických úchopných hlavice je zbytečný magnetismus, který způsobuje zachycování feromagnetických nečistot na objektech manipulace. Tento problém lze vyřešit použitím demagnetizační cívky po položení objektu a pomocí dodatečného úsilí při čištění pracovního prostoru. Oba tyto návrhy však zvyšují finanční zátěž při výrobě stroje, ve kterém je robot používán.

4.1.3 Vakuová chapadla

Vakuové koncové efektory robotů používají negativní tlak k uchopení a manipulaci s objekty. Tyto koncové efekory využívají princip vakuového sacího efektu, aby vytvořily bezpečný úchop na objektech s hladkými a plochými povrchy, jako jsou sklo, kov, plast nebo karton.



Obrázek 4.4: Ukázka funkce vakuových savek

Vakuové chapadla obvykle sestávají z vakuového generátoru, sacích savek, pružných hadic nebo trubek a řídicích ventilů. Vakuový generátor vytváří v systému negativní tlak nebo vakuum, které je přenášeno prostřednictvím hadic do vakuových savek. Vakuové savky jsou připevněny na povrch objektu a negativní tlak vytváří vzduchotěsné těsnění, které objekt bezpečně drží na místě.

Vakuové přísavky

Vakuové přísavky a podložky jsou rozhraním mezi vakuovým systémem a manipulovaným objektem. Jsou dostupné v různých velikostech, tvarech a materiálech, aby vyhověly různým tvarům, velikostem a texturám povrchů objektů. Vakuové savky a podložky mohou být vyrobeny z materiálů jako je guma, silikon nebo polyuretan.



Obrázek 4.5: Příklad přísavek [[8], [9]]

5 Inspekční systémy

Inspekční systémy v průmyslu zahrnují řadu nástrojů, zařízení a metodologií používaných k hodnocení kvality, integrity a souladu výrobků nebo procesů během výroby nebo montáže. Tyto systémy jsou navrženy tak, aby detekovali vady, odchylky nebo nesoulady a zajistily, zda výrobky splňují stanovené normy a požadavky. Inspekční systémy hrají klíčovou roli v udržování kvality výrobků, optimalizaci výrobních procesů a prevenci dodání vadných nebo substandardních produktů zákazníkům.

5.1 Kamerové inspekční systémy

Kamerové inspekční systémy jsou na popředí moderní průmyslové inspekční technologie, využívají kamery s vysokým rozlišením a sofistikované algoritmy zpracování obrazu k monitorování a analýze výrobního procesu. Tyto systémy jsou obzvláště cenné v odvětvích, kde je vizuální kontrola kvality klíčová, jako je výroba elektroniky, montáž automobilů a zpracování potravin. [10]

V jádru kamerových inspekčních systémů jsou kamery s vysokým rozlišením, které zachycují detailní snímky výrobku, jak postupuje po výrobní lince. Tyto snímky jsou následně zpracovávány pomocí pokročilých algoritmů, které mohou detekovat vady, měřit rozměry a ověřovat úplnost montáže. Systémy jsou často vybaveny různými konfiguracemi osvětlení, které zlepšují čistotu a kontrast obrazu, umožňující přesnou detekci vad i v náročných podmínkách. [10]

Hlavními výhodami kamerových inspekčních systémů je jejich rychlost a přesnost, které jsou výrazně vyšší než v případě metod manuální kontroly. Poskytují konzistentní výsledky, jsou schopné pracovat nepřetržitě 24/7 a významně snižují čas a náklady spojené s kontrolou kvality.

Nicméně implementace těchto systémů není bez výzev. Počáteční nastavení a kalibrace kamerových inspekčních systémů může být složitá a časově náročná. Vyžadují pečlivé konfigurace, aby byly v souladu s konkrétními výrobními potřebami, a musí být pravidelně udržovány, aby byla zajištěna jejich trvalá přesnost. Kromě toho tyto systémy mohou generovat velké objemy dat, které je třeba efektivně spravovat a analyzovat, aby byly užitečné.



(a) Keyence [11]

(b) Basler [12]

Obrázek 5.1: Ukázka kamer

Moderní kamerové inspekční systémy obvykle vyžadují specifické PC jednotky od výrobce a proprietární software. Tyto softwarové balíčky obsahují GUI, která uživatelům umožňují vybrat a konfigurovat různé parametry inspekce podle jejich konkrétních potřeb. Běžné inspekční parametry zahrnují geometrické specifikace, detekci hran a analýzu barev.

Tyto pokročilé kamerové systémy jsou všestranné a mohou být použity v širokém spektru aplikací, včetně:

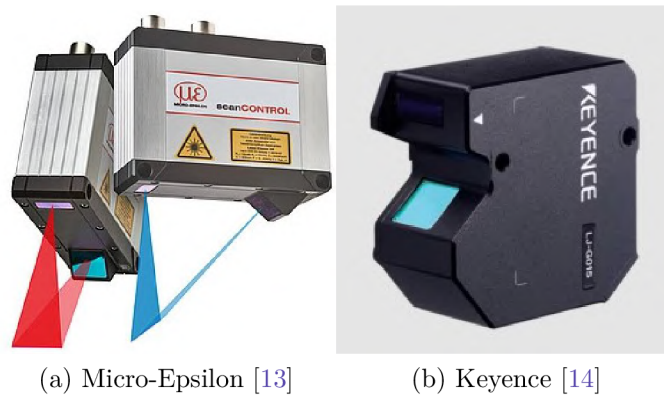
- **Inspekce barev a typu produktů:** Identifikace a ověřování barvy a typu produktů.
- **Inspekce přítomnosti:** Detekce přítomnosti nebo absence komponentů v sestavě.
- **Kontrola množství:** Počítání položek nebo ověřování množství v balení.
- **Pozicování/Vyrovnání:** Zajištění správného umístění a vyrovnání dílů.
- **Kontrola vzhledu:** Kontrola povrchových vad nebo kosmetických nedostatků.
- **Měření rozměrů:** Přesné měření rozměrů objektů.
- **Robotika s vizuálním naváděním:** Poskytování vizuální zpětné vazby pro řízení robotických operací.
- **Identifikace:** Čtení čárových kódů, QR kódů nebo rozpoznávání textů a symbolů.

Významní výrobci inspekčních kamer zahrnují společnosti Keyence, Cognex, Basler a Omron. Tyto společnosti jsou známé pro své inovativní řešení a robustní systémy, které jsou široce používány v různých odvětvích pro zlepšení kontroly kvality a automatizačních procesů.

5.2 Laserové inspekční systémy

Laserové inspekční systémy představují významný pokrok v oblasti technologií měření bez kontaktu. Jsou známé především svou přesností a rychlostí. Tyto systémy jsou široce používány v průmyslových odvětvích, kde je zásadní přesnost rozměrů a kvalita povrchu, jako jsou letecký průmysl, automobilový průmysl a výroba těžkých strojů.

Laserové inspekční systémy typicky využívají jeden nebo více laserových paprsků k prohledávání povrchu objektu. Laserové zařízení promítá paprsek na povrch a měří čas, který trvá světlu k návratu k senzoru, což je pak využito k přesnému výpočtu rozměrů objektu a detekci povrchových anomálií. Některé systémy využívají metody laserové triangulace, kde je úhel odraženého paprsku použit k určení měření, poskytující vysoce přesná data o geometrii objektu.



(a) Micro-Epsilon [13]

(b) Keyence [14]

Obrázek 5.2: Ukázka kamer

Laserové inspekční systémy jsou všestranné nástroje používané v různých aplikacích, včetně:

- **Celkové geometrické specifikace:**

- Měření tloušťky/šířky: Přesné měření tloušťky nebo šířky materiálů a komponentů.
- Měření výšky/výškových stupňů: Hodnocení výšky objektů nebo výškových rozdílů na vícestupňových površích.
- Měření vnějšího/vnitřního průměru: Určování vnějších a vnitřních průměrů válcových nebo kruhových objektů.

- **Kontrola polohování:**

- Délka zdvihu/polohování: Monitorování a ověřování délky zdvihu nebo přesného polohování komponentů v mechanických systémech.
- Odklon/decenrování: Detekce jakéhokoli odklonu nebo decenrování komponentů, aby bylo zajištěno, že zůstávají v rámci stanovených tolerancí.

- **Analýza radiální odchylky a vibrací:** Hodnocení radiálních odchylek a vibračních charakteristik rotujících komponentů, což je klíčové pro udržení rovnováhy a provozní stability.
- **Kontrola tvaru/profilu:** Analyzování tvaru a profilu objektů pro zajištění, že odpovídají specifikovaným geometrickým standardům.
- **3D inspekce:** Vytváření detailních trojrozměrných modelů objektů pro komplexní inspekci jejich geometrických vlastností a charakteristik povrchu.

Podobně jako kamerové inspekční systémy, laserové inspekční systémy vyžadují specifický software od výrobce, který umožňuje uživatelům programovat a přizpůsobovat požadované funkce. Tyto softwarové platformy obvykle poskytují uživatelsky přívětivá rozhraní, která usnadňují konfiguraci různých inspekčních parametrů a kritérií, čímž zajišťují, že systémy mohou být přizpůsobeny konkrétním průmyslovým potřebám.

Mezi přední výrobce v oblasti laserových inspekčních systémů patří: Hexagon Manufacturing Intelligence, Keyence Corporation, FARO Technologies, Carl Zeiss AG, Mitutoyo Corporation a Cognex Tyto společnosti jsou známé pro své pokročilé technologie a robustní inspekční systémy, které jsou široce používány v různých odvětvích, včetně automobilového průmyslu, výroby elektroniky a strojírenství, pro zajištění vysoké kvality a přesnosti výrobních procesů.

6 Modelová výrobní linka

Modulární výrobní systém (MPS) společnosti Festo poskytuje robustní vzdělávací platformu, která úzce napodobuje skutečné průmyslové pracovní postupy a nabízí komplexní přístup k učení. MPS učí studenty základním principům a reálným aplikacím automatizace a pokročilých výrobních procesů.

MPS se skládá z různých modulárních stanic, které simulují jednotlivé procesy nalezené na výrobní lince, včetně montáže, testování, třídění a manipulačních operací. Každý modul může fungovat nezávisle nebo může být propojen do soudržného a komplexního výrobního systému. Vrozená modularita MPS poskytuje významnou flexibilitu ve vzdělávacích prostředích, což umožňuje institucím přizpůsobit nastavení na základě jejich vzdělávacích cílů a prostorových omezení.

Pohony a sensorické schopnosti vestavěné do MPS od Festo jsou klíčové, zvyšují jak jeho funkční účinnost, tak vzdělávací dopad. Tyto komponenty jsou zásadní pro demonstraci dynamických interakcí mezi strojními prvky v realistickém průmyslovém kontextu a poskytují studentům pohlcující zkušenost s učením.

Aktuátory v rámci MPS zahrnují pneumatické válce, které se používají k posunu výrobků po výrobní lince. Dále vakuové savky usnadňují bezpečné manipulování s předměty, zatímco elektromagnet vybavený separátorem je používán k zastavení nebo oddělení výrobků podle potřeby. Pohyb dopravního pásu, nezbytný pro přepravu výrobků mezi jednotlivými stanicemi, je poháněn stejnosměrným převodovým motorem.

Pro účely zamýšlené funkce byly vybrány stanice MPS Festo: distribuční stanice, montážní stanice a třídící stanice.

6.1 Distribuční stanice

Distribuční stanice funguje jako podavač v rámci MPS. Podavače jsou klíčové komponenty určené k držení, třídění a podávání výrobků. Tyto jednotky navíc mají schopnost třídit výrobky na základě několika charakteristik, jako jsou tvar a hmotnost, což zvyšuje univerzálnost výrobního procesu. Distribuční stanice a její hlavní moduly lze vidět na obrázcích 6.1 a 6.2.



Obrázek 6.1: Distribuční stanice



(a) Modul stohovacího magazínu

(b) Modul dopravníku

Obrázek 6.2: Moduly distribuční stanice

Tato konkrétní stanice je zodpovědná za oddělování výrobků uložených v magazínové trubici stohovacího magazínu. Používá dvojité působící válec k individuálnímu vydávání výrobků. Následně modul dopravníku usnadňuje přepravu každého výrobku buď doprava nebo doleva, v závislosti na požadovaném směru. Stanice navíc obsahuje mechanismy pro zastavení a izolaci výrobků na dopravníku podle potřeby, což umožňuje přesnou manipulaci a zpracování. Distribuční stanice má 2 hlavní moduly:

- Modul dopravníku
- Modul stohovacího magazínu

Modul dopravníku slouží dvojímu účelu při transportu a ukládání výrobků v rámci výrobního systému. Pro zajištění přítomnosti a přesného umístění výrobků jsou strategicky umístěny optické blízkostní spínače vybavené optickými kabely,

a to před separátorem podavače a na konci dopravníku. Toto uspořádání umožňuje přesné sledování a řízení toku výrobků. [15]

Samotný dopravní pás je poháněn stejnosměrným převodovým motorem, který zajišťuje konzistentní a kontrolovaný pohyb výrobků po výrobní lince. Dále mohou být výrobky zastaveny a přesně odděleny pomocí připojeného elektromagnetu doplněného o separační mechanismus. Konečné pozice výrobků jsou pečlivě sledovány indukčními blízkostními spínači, které přispívají k přesnosti a efektivitě systému. [15]

Modul stohovacího magazínu je navržen tak, aby metodicky odděloval výrobky ze skladovacího magazínu. Tento magazín je schopen pojmout až sedm výrobků, které jsou v něm uloženy v libovolném pořadí. Modul využívá dvojitě působící válec k vytlačení spodního výrobku z gravitačně napájeného magazínu. Tento pohyb tlačí výrobek proti externí zarážce a přesně jej umísťuje do držáku. [15]

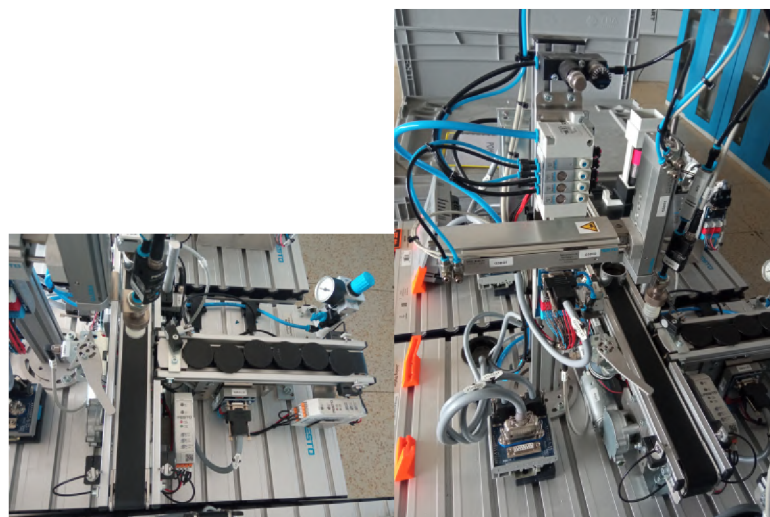
Na tomto držáku lze nainstalovat optoelektronický senzor pro detekci buď krytek nebo samotných výrobků, což zajišťuje správnou identifikaci a manipulaci.

6.2 Spojovací stanice

Spojovací stanice je pokročilá konfigurace, která vychází z základního designu separační stanice, rozšířená o modul pro vyzvedávání a umísťování (pick & place) a sběrníkový uzel. Modul pro vyzvedávání a umísťování je speciálně vybaven sacím pohárem, který je navržen tak, aby usnadnil spojování krytek výrobků s základními výrobky, čímž zajišťuje přesné montážní procesy. Spojovací stanici a její hlavní moduly lze vidět na obrázcích 6.3 a 6.4.



Obrázek 6.3: Spojovací stanice



(a) Distribuční stanice

(b) Spojovací stanice

Obrázek 6.4: Moduly spojovací stanice

Tato stanice je univerzální, s možností jak extrahovat základny výrobků z materiálového toku, tak je podle potřeby zpět integrovat. Během provozu je základna výrobku pevně držena ve spojovací pozici u odbočky, což stabilizuje výrobek pro přesnou montáž a integraci. Toto strategické uspořádání zvyšuje efektivitu výrobního procesu, což umožňuje nepřetržité a spolehlivé manipulace s komponentami. Spojovací stanice má 3 hlavní moduly:

- Modul dopravníku 1
- Modul dopravníku 2
- Modul pro vyzvedávání a umístování

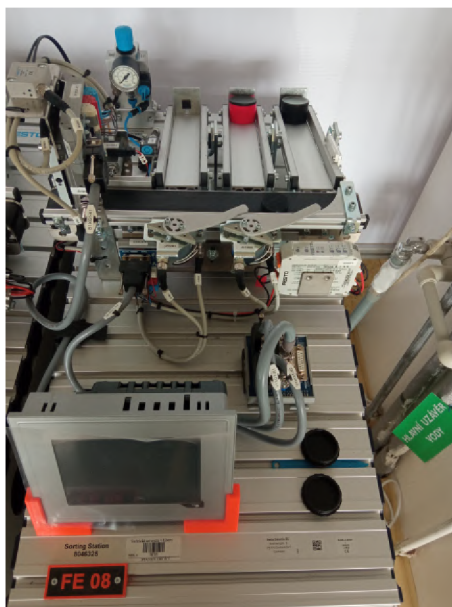
Modul dopravníku 1 a **Modul dopravníku 2** plní podobné funkční role, jak je vidět na distribuční stanici. Nicméně Modul dopravníku 2 je speciálně určen pro manipulaci s krytkami výrobků, což odlišuje jeho roli od manipulace se standardními výrobky, jak je tomu u Modulu dopravníku 1. [16]

Modul pro vyzvedávání a umístování představuje univerzální dvouosé manipulační zařízení, které se vyznačuje svou přesností a adaptabilitou. Tento modul se skládá z vysoce přesných posuvných jednotek, jejichž pohybové limity jsou monitorovány senzory přiblížení. Konfigurace těchto senzorů, včetně jejich montážní polohy a výšky, je plně nastavitelná, aby vyhovovala různým provozním potřebám. [16]

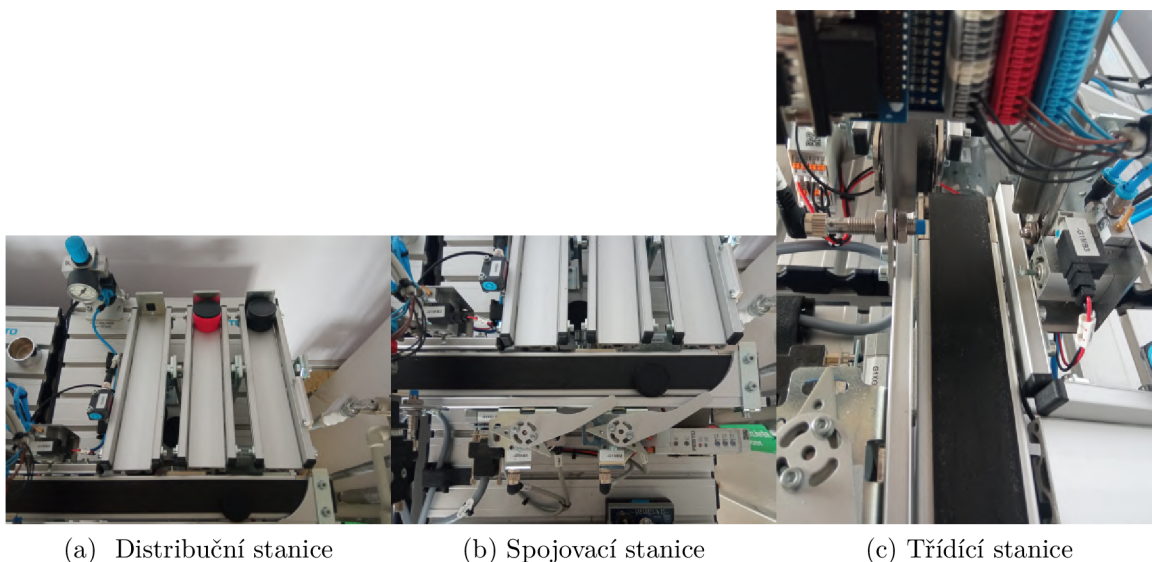
Dále modul obsahuje savky vybavený měchy pro bezpečné uchopení výrobků. Přímo k savce je připojený vakuový filtr, který hraje klíčovou roli v prevenci vnikání částic nečistot do vakuového generátoru, čímž zajišťuje čistotu a integritu systému. Tlakový spínač integrovaný do sestavy potvrzuje spolehlivé uchopení výrobků, což poskytuje zpětnou vazbu nezbytnou pro automatizované procesy. [16]

6.3 Třídící stanice

Třídící stanice je navržena tak, aby kategorizovala výrobky do tří různých skluzavek na základě specifických kritérií. Jakmile výrobky vstoupí na stanici po dopravním pásu, jsou nejprve identifikovány optickým přibližovacím senzorem, který pro přesnou detekci využívá optický kabel. Třídící stanici a její hlavní moduly lze vidět na obrázcích 6.5 a 6.6. [17]



Obrázek 6.5: Třídící stanice



(a) Distribuční stanice

(b) Spojovací stanice

(c) Třídící stanice

Obrázek 6.6: Moduly třídicí stanice

Jakmile jsou výrobky detekovány, jsou dočasně zastaveny pneumatickou zarážkou, což umožňuje podrobné hodnocení jejich vlastností. Detekční modul na stanici je vybaven senzory, které přesně určují materiál a barvu každého výrobku, rozlišují mezi černou, červenou a stříbrnou. Po identifikaci jsou výrobky přemístěny do odpovídajících skluzavek pomocí elektricky ovládaných odboček na základě předem definovaných kritérií. Navíc jsou úrovně naplnění skluzavek neustále sledovány retro-reflexním senzorem, který pomáhá udržovat optimální provoz a zabrání přeplnění. [17]

Třídící stanice má 3 hlavní moduly :

- Detekční modul
- Modul dopravníku
- Modul skluzavky

Detekční modul je navržen tak, aby ověřil přítomnost a identifikoval charakteristiky výrobků na základě barvy a typu materiálu. Je schopen rozpoznávat červené, černé a kovové výrobky. Tento modul využívá tři senzory přiblížení vybavené digitálními výstupy k určení materiálu nebo barvy výrobků. Tato sofistikovaná senzorická technologie zajišťuje přesnou identifikaci a klasifikaci. [17]

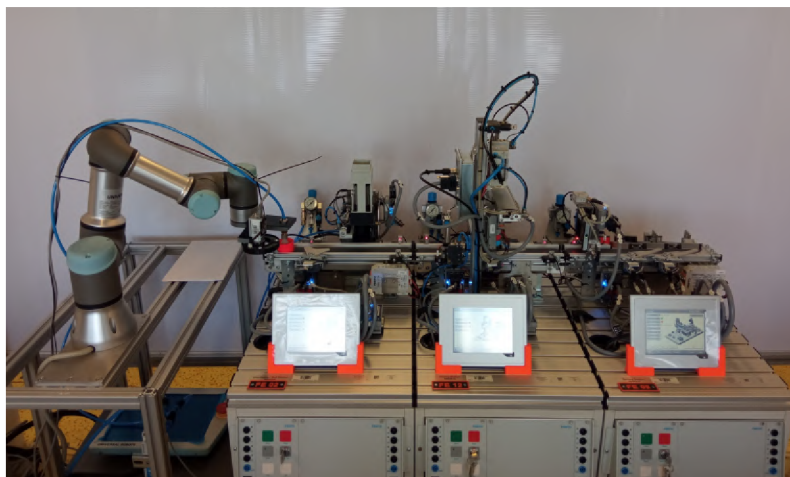
Modul dopravníku si zachovává stejnou funkčnost jako na distribučních a třídících stanicích.

Modul skluzavky je navržen pro transport nebo skladování výrobků a vyznačuje se univerzálním designem s variabilně nastavitelným sklonem a výškou, aby vyhovoval různým provozním potřebám. Modul může pojmout až pět výrobků. Na třídící stanici jsou využívány tři moduly skluzavek k ukládání výrobků. Toto uspořádání umožňuje organizovanou manipulaci a dočasné skladování výrobků, když jsou tříděny a směřovány k dalším fázím zpracování. [17]

6.4 Integrace robota k Modulárnímu výrobnímu systému

Pro integraci robota do MPS byl sestaven hliníkový rám. Tento rám je vybaven kolečky pro snadnější přepravu. Následně byly tři vybrané moduly MPS spolu s hliníkovým rámem s připevněným robotem zarovnané do přesné konfigurace. Toto uspořádání je znázorněno na obrázku 6.7.

Po správném zarovnání jsou moduly a rám zajištěny pomocí kovovou propojkou, která je navržen tak, aby zapadala do drážek hliníkových profilů. Tento způsob upevnění zajišťuje, že komponenty zůstávají pevně na svém místě, což udržuje stabilitu a integritu integrovaného systému.



Obrázek 6.7: Uspořádání stoje

6.5 Funkce stroje

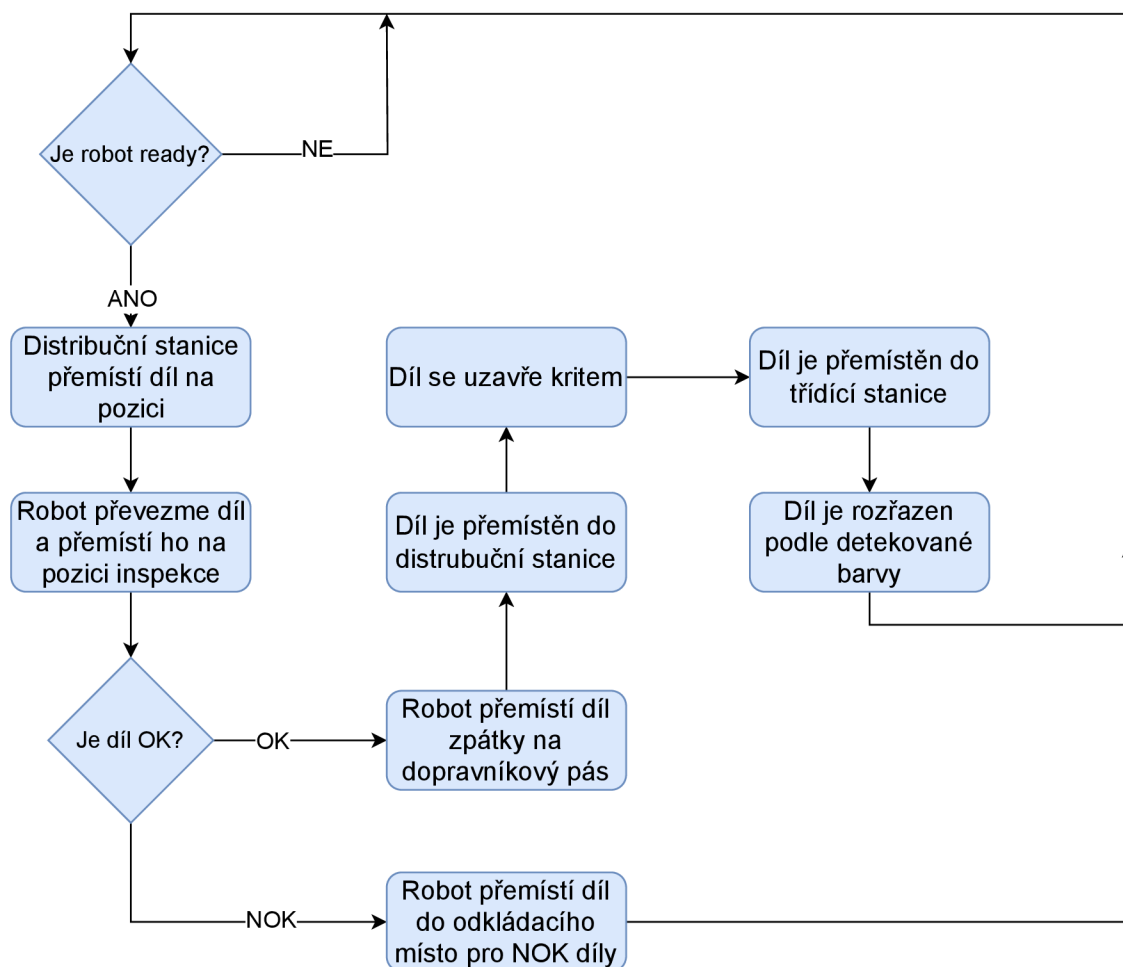
Na počátku PLC čeká na signál připravenosti od robota. Po přijetí tohoto signálu je výrobek vystřelen z modulu stohovacího magazínu v distribuční stanici a přepraven k hraně vybavené zarážkou. Následně je robotu odeslán signál, že je výrobek správně umístěn. Robot poté výrobek vyzvedne a umístí ho do pozice pro inspekci, kde je pořízen snímek pro hodnocení.



Obrázek 6.8: OK výrobky

Tento snímek je klíčový pro posouzení jak barvy, tak tvaru výrobku. Pokud je výrobek určen jako NOK, je robotem přesunut do vyhrazené oblasti pro NOK výrobky. Naopak, pokud je výrobek klasifikován jako OK, je vrácen na distribuční stanici. Odtud výrobek postupuje do spojovací stanice, kde je na něj umístěná krytka. Posledním krokem je přemístění výrobku do třídící stanice, kde je roztríděn podle barvy identifikované během inspekce kamerou.

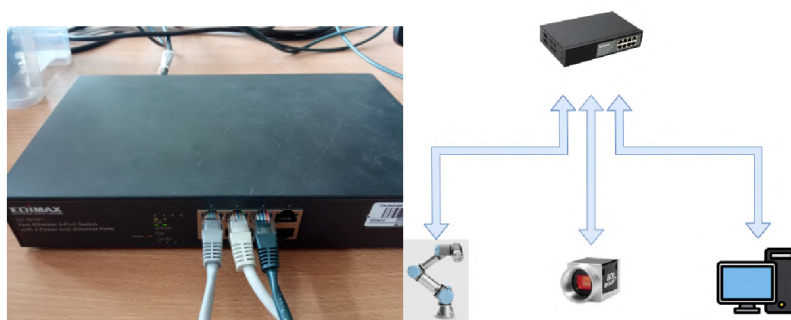
Tento cyklus se opakuje, dokud modul stohovacího magazínu nevyčerpá své zásoby výrobků. Diagram popisující funkci stroje lze vidět na obrázku 6.9. Zdrojový kód k funkci stroje lze vidět v příloze P.1.



Obrázek 6.9: Diagram průchodu materiálu strojem

7 Komunikace mezi prvky

Pro to, aby celý stroj mohl splnit své určené úkoly, je nezbytné zřídít komunikační kanál mezi jednotlivými zařízeními (robot, PC, kamera a PLC). Aby bylo možné tuto integraci usnadnit, jsou robot, kamera a PC propojeny prostřednictvím síťového switchu, konkrétně modelu EDIMAX ES-5844P. Tento přepínač podporuje technologii Power over Ethernet (PoE), která je využívána k napájení kamery. Tento přepínač je znázorněn na levé straně přiloženého obrázku 7.1. Diagram popisující komunikační cesty mezi těmito zařízeními je prezentováno na pravé straně obrázku 7.1.



Obrázek 7.1: Komunikace

Aby robot, PC a kamera mohly efektivně rozpoznávat a komunikovat mezi sebou, je zásadní, aby byly nakonfigurovány se stejnou výchozí bránou. V tomto scénáři je určená výchozí brána 169.254.153.1. Konkrétní IP adresy přiřazené jednotlivým zařízením jsou podrobně uvedeny v příložené tabulce 7.1. Připojení k robotovi a kamere je poté realizováno prostřednictvím Matlabu s využitím TPC/IP Toolboxu a Image Acquisition Toolboxu s rozhraním GigE Vision.

Tabulka 7.1: Rozdělení IP adres

Zařízení	IP adresa
Robot	169.254.153.50
PC	169.254.153.11
Kamera	169.254.153.53

7.1 Komunikace s PLC

Komunikace s PLC probíhá přes kabel. Jedná se o kabel 0,14×8. Ten je připojen přímo od karty PLC k ovládací jednotce robota.

Tabulka 7.2: Zapojení signálů

Barva	Zapojení robot	Zapojení PLC	Typ signálu z pohledu robota
Modrá	0 VDC	BC1	-
Černá	DO 7	DI 15	Výstup
Hnědá	DO 8	DI 16	Výstup
Bíla	DO 6	DI 7	Výstup
Fialová	DO 5	DI 8	Výstup
Oranžová	DI 1	DO 13	Vstup

Význam jednotlivých signálů

DO 7 a DO 8 Hlavní funkcí těchto výstupů je signalizovat PLC o stavu kontroly každého výrobku, rozlišující mezi kusy OK a NOK. Pokud je výrobek považován za přijatelný, jsou rovněž přenášeny informace o jeho barvě. Tyto údaje jsou kódovány pomocí 2-bitového systému, jak je popsáno v příložené tabulce 7.3.

Tabulka 7.3: Zapojení signálů

bit1	bit2	Výsledek
0	0	NOK
0	1	Červená
1	0	Černá
1	1	Stříbrná

DO 6 Před zahájením procesu vyzvednutí výrobku robot sděluje výrobní lince, že je připraven zahájit proces vyzvednutí. Po dokončení kontroly a přesunutí výrobku zpět na dopravní pás robot informuje výrobní linku, že byl výrobek bezpečně uložen, což umožňuje pokračování ve výrobním procesu.

DO 5 Robot sděluje výrobní lince, že výrobek byl vyzvednut.

DI 1 Před tím, než robot zahájí proces vyzvednutí, výrobní linka informuje robota, že je výrobek správně umístěn a připraven k vyzvednutí. Po dokončení potřebných úkolů výrobní linka komunikuje s robotem, že je určená pozice nyní dostupná, což zajišťuje, že výrobek může být bezpečně umístěn zpět.

Schéma zapojení lze vidět v příloze N.1.

8 Pohyb robotem

Robot je ovládán prostřednictvím Matlabu s využitím knihovny `UR3_lib`, která byla původně vyvinuta jako součást bakalářské práce. Tato knihovna usnadňuje získávání aktuálních dat z robota a jeho následnou manipulaci. Ovládací modul zahrnuje standardní robotické pohyby, jako jsou `movej`, `move1` a `movep`. Nicméně nedisponuje schopností poskytovat uživateli zpětnou vazbu po dokončení předepsaných pohybů. Absence takové zpětné vazby brání potvrzení dokončení úkolů robota, čímž se stává přesné ovládání pohybu robota napříč různými pozicemi nemožný.

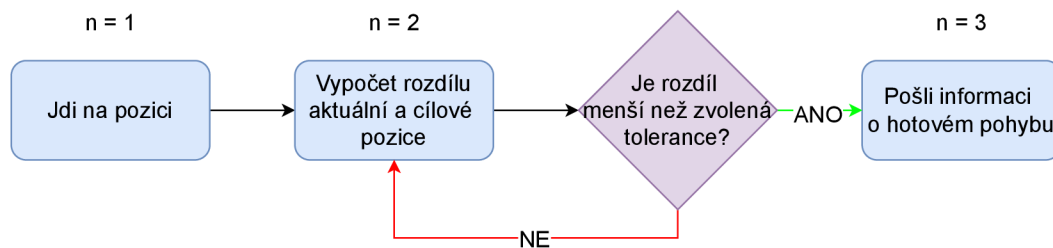
8.1 Zpětná vazba

Vzhledem k nutnosti spolehlivého pohybu mezi pozicemi bylo nezbytné provést úpravy knihovny `UR3_lib`, aby zahrnovala schopnosti zpětné vazby. Bohužel, data získaná z robota neuvádějí, zda dosáhl svého určeného cílového bodu nebo zda zastavil své pohyby. V důsledku toho je nezbytné získat tuto zpětnou vazbu z dostupných informací. Vzhledem k tomu, že zpětná vazba je klíčová po dokončení pohybu robota, nejvhodnějšími indikátory jsou jeho kinematické vlastnosti, konkrétně zrychlení, rychlost a poloha.

Co se týče zrychlení, kritickou hodnotou k identifikaci by bylo, když zrychlení je rovno nule, tedy $a = 0 \text{ m/s}$. Zrychlení během pohybu však může vykazovat výraznou proměnlivost, zůstávat konstantní nebo nabývat nulové hodnoty, což ztěžuje určení, zda nulové zrychlení doopravdy znamená ukončení pohybu. Z tohoto důvodu je zrychlení pro účely zpětné vazby nespolehlivé. Rychlost prochází podobnými fluktuacemi, ale stav, kdy rychlost je rovna nule, tedy $v = 0 \text{ m/s}$ obvykle nastane, když robot zastaví po pohybu. Tento stav by však mohl být také vyvolán nechtěným střetem s externím objektem. Ačkoliv rychlost může sloužit jako mechanismus zpětné vazby, je rozumné zvážit také poslední kinematickou míru pro optimální určení zpětné vazby. Data o poloze nabízejí přímý způsob, jak odvodit zpětnou vazbu výpočtem rozdílu mezi aktuální a cílovou pozicí. Toto je znázorněno ve vzorci 8.1. Zdrojový kód lze vidět v příloze P.1.

$$\text{Hotovo} = \begin{cases} \text{ANO} & \text{pokud } |\text{Cílová poz} - \text{Aktuální poz}| < x \\ \text{NE} & \text{Jindy} \end{cases} \quad (8.1)$$

Obrázek 8.1 znázorňuje blokový diagram funkce zpětné vazby. n má zde význam indexu ve funkci `switch`. Zdrojový kód lze vidět v příloze P.1.



Obrázek 8.1: Diagram zpětné vazby

8.2 Použití Handshaku

Příkaz pro použití funkce Handshaku je:

```
[potvrzeni, Ind] = UR3_lib.movej_with_feedback(client, Pozice, a, v, t, r, Ind)
```

V tomto příkazu:

Proměnná `potvrzeni` vrací potvrzovací zprávu „Done“ (hotovo) poté, co robot dokončí svůj pohyb. Tato zpráva slouží jako potvrzení, že zamýšlený pohyb byl úspěšně dokončen.

`Ind` je proměnná, která se při dokončení každého pohybu zvýší o 1. Toto zvýšení lze efektivně použít pro přechod mezi stavy v rámci `switch` funkce, čímž usnadňuje správu stavů a řízení toku v robotickém systému.

Tyto dva výstupní argumenty funkce jsou volitelné. Pokud je funkce volána bez těchto argumentů, při dokončení neposkytne žádnou odpověď. Přesto funkce zajistí, že aktuální pohyb bude plně proveden před zahájením jakýchkoli dalších příkazů pohybu. Tato funkce zaručuje, že každý pohyb je dokončen sekvenčně, čímž se udržuje integrita robotických operací. Varianty toho příkazu lze vidět ve zdrojovém kódu 8.1.

Zdrojový kód 8.1: Možnosti použití funkce

```

1 [potvrzeni, Ind] = UR3_lib.movej_with_feedback(
2     client, Pozice, a, v, t, r, Ind)
3 [potvrzeni] = UR3_lib.movej_with_feedback(client,
4     Pozice, a, v, t, r, Ind)
5 UR3_lib.movej_with_feedback(client, Pozice, a, v, t
6     , r, Ind)
7
8 UR3_lib.movej_with_feedback(client, Pozice, a, v)
9 UR3_lib.movej_with_feedback(client, Pozice, t)
  
```

Kromě volitelných výstupních argumentů, které byly dříve diskutovány, funkce také zahrnuje několik volitelných vstupních argumentů, které poskytují flexibilitu

při řízení pohybu robota. Tyto argumenty zahrnují **a** (zrychlení), **v** (rychlost), **t** (čas), **r** (poloměr) a **Ind** (index).

a a **v**: Tyto parametry jsou volitelné a určují zrychlení a rychlost, s jakou se robot pohybuje do určené pozice.

t: Tento parametr umožňuje uživateli specifikovat čas, během kterého by se měl robot přesunout do určené pozice.

r: Poloměr ohybu **r** definuje zakřivení, které robot následuje při přechodu z jednoho pohybu do druhého.

Použití těchto parametrů lze pozorovat v zdrojovém kódu 8.1, konkrétně na řádcích 5 a 6, které demonstrují, jak lze tyto argumenty aplikovat.

8.3 Pozice robota

Kritickým aspektem ovládání pohybu robota je vymezení klíčových pozic a určení pořadí, ve kterém robot přechází mezi nimi. Celkem bylo identifikováno pět zásadních pozic: výchozí pozice (Home position), pozice výrobku na dopravním pásu, pozice pro inspekci výrobku, pozice pro fotografování kamerou a pozice určená pro umístění nevyhovujících (NOK) výrobků.

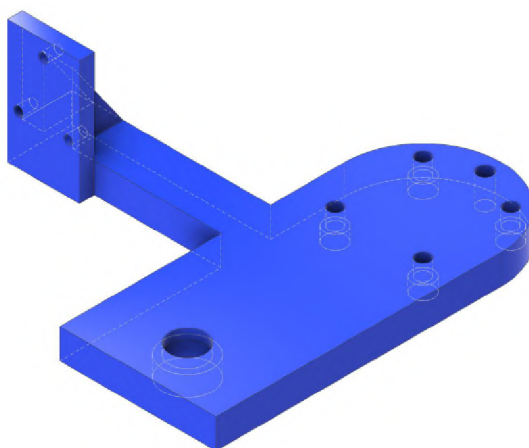
Proces přebírání výrobku z pozice nebo jeho ukládání do pozice představuje potenciální riziko kolize s jinými objekty v okolí. Aby se toto riziko minimalizovalo, tak bylo nutné se k těmto pozicím přiblížit shora, poté sestoupit svisle dolů, nebo naopak, vzestoupit svisle z těchto pozic. Tato metoda pomáhá předcházet pohybům, které by mohly vést ke kolizím. Předepsaná sekvence pohybů mezi těmito pozicemi je popsána v příloženém diagramu toku 8.2.



Obrázek 8.2: Flowchart pohybu robota

9 Nástroj robota

Pro návrh nástroje robota je nezbytné zvážit jeho funkční požadavky. Nástroj musí obsahovat mechanismus pro připojení sací přísavky a zároveň poskytnout možnost bezpečného upevnění kamery, aniž by hrozily kolize během manipulace s výrobkem. Kromě toho musí být design kompatibilní s přírubou robota, pro zajištění přesného a stabilního spojení. Model navrženého nástroje lze vidět na obrázku 9.1.

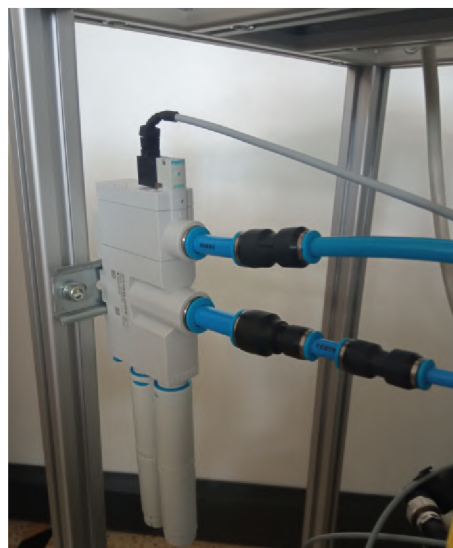
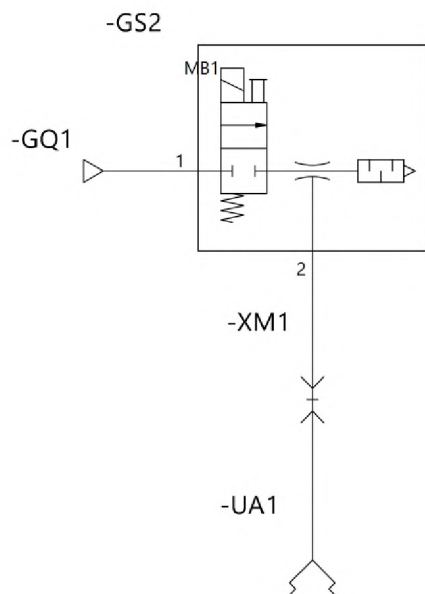


Obrázek 9.1: Model nástroje

9.1 Pneumatický řetězec

Pneumatický řetězec Festo se skládá z několika klíčových komponent: zdroje stlačeného vzduchu, generátoru vakua, spojovacích hadic a fitinek a sací přísavky. Speciﬁcký generátor vakua použitý v tomto uspořádání je model VN-20-H-T6-PQ4-VQ5-RO2-M. Tento model je dostatečně velký a robustní, což mu umožňuje pohodlně zvedat určené výrobky.

Provozní tlak vakuového ejektoru je nastaven na 3,5 baru. Vstupní připojení pro vakuový ejektor je specifikováno jako QS-10. Vzhledem k tomu, že zdroj stlačeného vzduchu má výstup QS-8, je nutné snížení velikosti hadice z průměru 10 na 8, aby byla zajištěna kompatibilita mezi komponenty. Dále je výstup vakuového ejektoru vybaven připojením QS-12. Aby se tento výstup sladil s požadavky systému, je nutné snížení velikosti hadice z průměru 12 na 6. Schéma zapojení pneumatického řetězce lze vidět na obrázku 9.2.



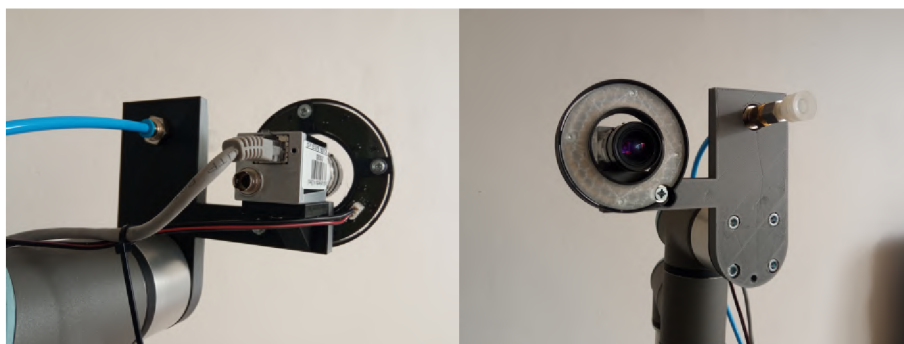
(a) Schéma zapojení

(b) Vakuový ejektor

Obrázek 9.2: Vakuový ejektor

9.2 Konečný návrh

Kvůli přehlednutí v počátečním návrhu nástroje robota, který neumožňoval instalaci osvětlení nezbytného pro kameru, byla nutná úprava. Tato úprava zahrnovala vyvrtání jednoho otvoru do nástroje, aby bylo možné bezpečně připevnit osvětlení šroubem. Finální verze nástroje robota, nyní vybavena a namontovaná na přírubě robota, je zobrazena na příloženém obrázku 9.3.



Obrázek 9.3: Konečný návrh nástroje

10 Kamera

Jak už bylo zmíněno použitá kamera je model Basler acA1300-30gc ze série ace firmy Basler, který je speciálně navržen pro průmyslové a strojové vizuální aplikace. Tato kamera je vybavena 1,3 megapixelovým CMOS senzorem, který poskytuje rozlišení 1296 x 966 pixelů, což zajišťuje čisté a detailní zachycení obrazu, které splňuje rozmanité požadavky průmyslového zobrazování. Nabízí vysokou snímkovou frekvenci až 30 snímků za sekundu, což je výhodné pro zachycení rychle se pohybujících procesů. Kamera využívá rozhraní Gigabit Ethernet (GigE) pro efektivní přenos dat. Navíc je schopna produkovat barevné obrázky, což je nezbytná funkce pro aplikace, kde je kritická diferenciací barev, jako jsou procesy kontroly kvality. Kamera je zobrazena na obrázku 10.1.



Obrázek 10.1: Basler acA 1300 30gc

Jak bylo dříve diskutováno v kapitole 7, připojení ke kamerě je umožněno pomocí nástroje `Image Acquisition Toolbox` v Matlabu. Pro navázání spojení je nejprve nutné vypsát dostupné kamery pomocí příkazu `gigecamlist`. Z výsledného seznamu lze pak připojit k požadované kamerě pomocí příkazu `gigecam`. Tento příkaz také vyžaduje specifikaci formátu pixelů. V tomto případě je to `BayerBG8`. Následujícím krokem je nastavení parametrů kamery, to je popsáno ve zdrojovém kódu P.3 v příloze. Jakmile je nastavení dokončeno, je možné pořizovat snímky buď pomocí příkazu `snapshot(Cam)` pro jednotlivé snímky, nebo příkazu `preview(Cam)` pro nepřetržitý živý přenos.

10.1 Poloha pro inspekci

Dalším klíčovým faktorem, který je třeba zvážit, je místo, kde je snímek pořízen. Ideálně by nejlepším řešením bylo pořídit snímek, zatímco výrobek zůstává na dopravním pásu. Tento přístup by urychlil proces snížením času potřebným pro manipulaci s výrobkem mezi jednotlivými pozicemi. Nicméně se objevuje významná výzva u jednoho z kontrolovaných výrobků, který je černý a bohužel se shoduje s barvou dopravního pásu. Tato shoda barev komplikuje úkol přesné a spolehlivé segmentace kontrolovaného výrobku od pozadí, což ovlivňuje efektivitu analýzy obrazu.

Aby se předešlo problému nedostatečného kontrastu při pořizování snímků, bylo nutné zřídit určenou oblast pro snímky. Tato oblast se nachází ve stejné jednotce jako robot. Povrch této oblasti je bílý, což výrazně zvyšuje kontrast pro černé, stříbrné a červené výrobky proti pozadí. Toto kontrastní uspořádání výrazně usnadňuje následné procesy segmentace barev a detekce hran, čímž zlepšuje přesnost a spolehlivost analýzy obrazu



(a) Černý výrobek

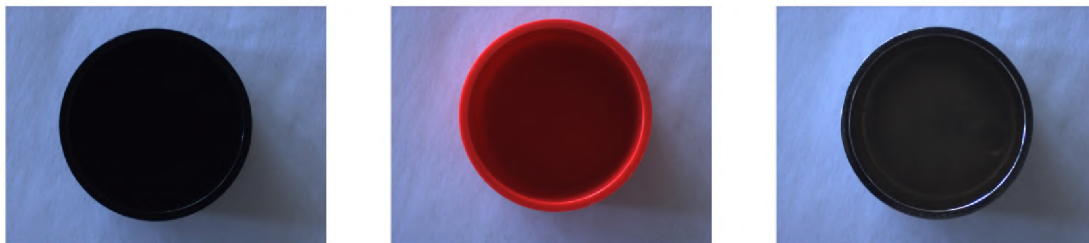
(b) Červený výrobek

(c) Stříbrný výrobek

Obrázek 10.2: Vyfocené výrobky

10.2 Osvětlení

Osvětlení hraje klíčovou roli v přesnosti analýzy obrazu, jak je ukázáno na obrázku 10.3, kde kontrolované výrobky vypadají uspokojivě i bez přídavného osvětlení. Počáteční pozorování by mohla naznačovat, že segmentace by mohla být možná i bez umělého osvětlení. Nicméně vznikají vlivy kvůli variacím přirozeného světla během dne. Tyto fluktuace mohou změnit stíny vrhané výrobky, což může komplikovat další segmentaci, zejména u stříbrných výrobků. Navíc změny v přirozeném osvětlení mohou ovlivnit celkový barevný tón obrázků, což vede k nesprávným výsledkům inspekce.



(a) Černý výrobek

(b) Červený výrobek

(c) Stříbrný výrobek

Obrázek 10.3: Vyfocené výrobky bez světla

Aby se zmírnily výzvy způsobené proměnlivým přirozeným osvětlením, byl na nástroji robota instalován LED pásek, umístěný pod kamerou v kruhovém rámu. Toto osvětlení je aktivováno pouze během procesu pořizování snímků, což zajišťuje konzistentní světelné podmínky, které jsou klíčové pro spolehlivou analýzu obrazu. Ovládání osvětlení je řízeno digitálním výstupem robota, konkrétně DO 3, což umožňuje přesné řízení zapojení světla. Snímky s osvětlením výrobků lze vidět v předešlé sekci na obrázku 10.2. Osvětlení lze vidět na obrázku 10.4.



Obrázek 10.4: Osvětlení

11 Předzpracování snímku

Před zahájením jakékoli inspekce musí být zachycený snímek nejprve podroben předzpracování. Tento proces začíná změnou velikosti snímku na polovinu, čímž se snižují výpočetní nároky pro následné analýzy. Po této redukci se provádí segmentace objektu. Segmentace může být provedena různými technikami, z nichž každá je vhodná pro různé vlastnosti obrazu. V této práci je použita metoda zahrnující segmentaci hran.

11.1 Hranová segmentace

Segmentace detekce hran je založena na detektorech hran, které jsou klíčové pro identifikaci obvodů kruhových výrobků v obraze. Jakmile je hrana detekována, lze ji zvýraznit pomocí morfologických operací. Tyto operace usnadňují vytvoření binární masky, kde je vnitřek kruhu reprezentován jedničkami a vnějšek nulami. Tato maska může být následně aplikována na původní obrázek pomocí násobení, čímž se izoluje oblast zájmu.

Existuje několik metod detekce hran, včetně Sobelovy, Prewittovy, Robertsovy, Laplacian of Gaussian (LoG) a Cannyho. Každá z těchto metod nabízí své vlastní výhody a je vhodná pro různé požadavky zpracování obrazu. Matlab Image Processing Toolbox zahrnuje implementace mnoha z těchto technik detekce hran, poskytující robustní rámec pro zvládnutí takových úkolů. Účinnost Sobelovy metody je demonstrován na výsledcích ukázaných na obrázku 11.1.



(a) Vstupní snímek

(b) Segmentovaný snímek

Obrázek 11.1: Výsledek Matlab Sobelova filtru

Jak je znázorněno na obrázku, segmentace dosažená detekcí hran je obecně účinná

ná, s výjimkou malé nedokonalosti – malého přesahu – pozorovaného v pravém dolním rohu segmentovaného obrazu. Kvůli této drobné nesrovnalosti byly prozkoumány alternativní metody, ale ty poskytly podobné nebo méně uspokojivé výsledky. V důsledku toho se vývoj vlastního algoritmu pro detekci hran jevil jako nejlepší alternativa.

11.1.1 Hranové detektory

Hrana v obraze je dána vlastnostmi obrazového elementu a jeho okolí. Je určena tím, jak náhle se mění hodnota obrazové funkce $f(x, y)$. Matematickým nástrojem pro studium změn funkce dvou proměnných jsou parciální derivace. Změnu funkce udává její gradient, vektorová veličina ∇ , určující směr největšího růstu funkce (směr gradientu) a strmost tohoto růstu (velikost, modul gradientu). Pixely s velkým modulem gradientu se nazývají hranami. Hlaváč a Sedláček [18]

Následující vzorce jsou převzaté od Gonzalez a Woods [19]. Gradient obrazové funkce $f(x, y)$ lze vyjádřit jako:

$$\nabla f \equiv grad(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (11.1)$$

Z toho je velikost gradientu, která je zásadní pro detekci hran, vypočítána pomocí vzorce:

$$M(x, y) = mag(\nabla f) = \sqrt{g_x^2 + g_y^2} \quad (11.2)$$

V digitalizovaném obraze aproximujeme parciální derivace diferencemi

$$\begin{aligned} \Delta_x g(x, y) &= g(x, y) - g(x - n, y), \\ \Delta_y g(x, y) &= g(x, y) - g(x, y - n) \end{aligned} \quad (11.3)$$

Operátory aproximují derivace pomocí diferencí, jejichž základní myšlenku jsme viděli ve vztahu 11.3. Některé operátory jsou invariantní vůči rotaci, např. Laplaceán, a mohou být počítány konvolucí s jedinou maskou. Jiné, aproximující první derivaci, využívají několik masek odpovídajících příslušné orientaci. Z nich se vybere ta, která nejlépe lokálně aproximuje obrazovou funkci. Výběrem jedné z masek je určen i směr gradientu (orientace). Hlaváč a Sedláček [18]

Sobelův hranový detektor

Sobelův hranový detektor využívá celkem osm konvolučních jader o velikosti 3×3 , které jsou odvozeny systematickým otáčením základního jádra. Tento proces rotace a výsledné konfigurace jader jsou podrobně popsány v tabulce 11.1. Pro účely výpočtů byla konkrétně vybrána jádra h_1 a h_3 .

Tato jádra efektivně zachytávají informace o hranách ve více směrech, čímž zvyšují schopnost detektoru přesně vymezit hrany v obraze. Volba h_1 a h_3 byla založena na jejich výkonu v testech, kde poskytovaly nejrelevantnější detaily hran pro další analýzu.

Tabulka 11.1: Příklad konvolučních kernelů

1	2	1
0	0	0
-1	-2	-1

h1

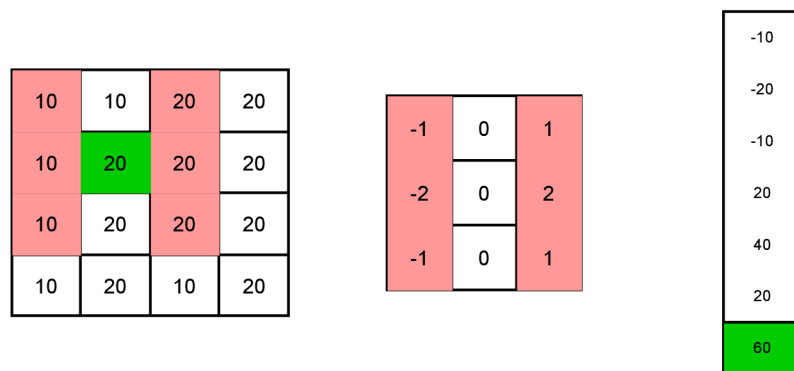
0	1	2
-1	0	1
-2	-1	0

h2

-1	0	1
-2	0	2
-1	0	1

h3

Grafické znázornění použití masky lze vidět na obrázku 11.2.



Obrázek 11.2: Použití masky

Při zaměření na konkrétní bod (x, y) v obraze lze operaci s maskou lokalizovat na malou oblast 3×3 kolem tohoto bodu, označovanou jako I_m . Reprezentace tohoto procesu může být matematicky přepsána následovně:

$$\begin{aligned} grad_x &= h_3 \odot I_m = (1 \cdot I_{m(1,3)} + 1 \cdot I_{m(2,3)} + 1 \cdot I_{m(3,3)}) - \\ &\quad (1 \cdot I_{m(1,1)} + 1 \cdot I_{m(2,1)} + 1 \cdot I_{m(3,1)}), \quad (11.4) \\ grad_y &= h_1 \odot I_m \end{aligned}$$

Pro výpočet velikosti gradientu při zpracování obrazu lze použít vzorec uvedený v rovnici 11.2. Po vypočítání velikostí gradientů následuje krok aplikace prahu na tyto hodnoty. Prahování je zásadní, protože pomáhá rozlišovat hrany nastavením hraniční hodnoty.

$$img(x, y) = \begin{cases} 1 & \text{Pokud } M(x, y) \geq 0, 1 \\ 0 & \text{Jindy} \end{cases} \quad (11.5)$$

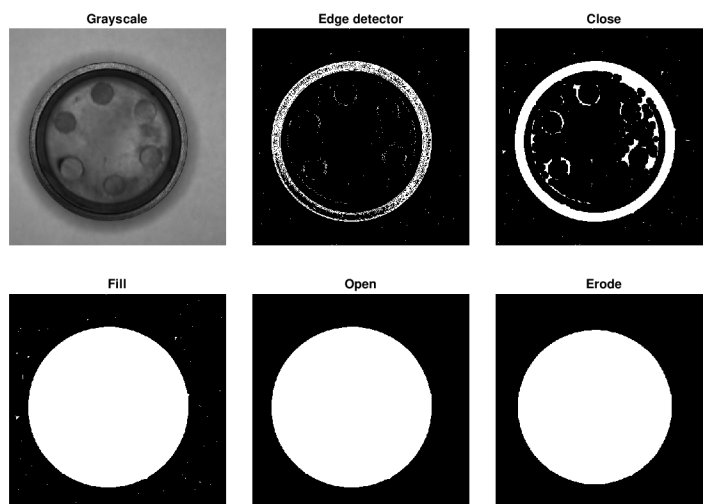
11.1.2 Použití hranové detekce k segmentaci obrazu

Segmentace hran s využitím Sobelova detektoru hran zahrnuje několik klíčových kroků. Nejprve je obrázek převeden na odstíny šedi, což je nezbytný krok pro de-

tekci hran. Po převedení na odstíny šedi je aplikován Sobelův detektor hran, aby identifikoval potenciální hrany zdůrazněním oblastí s významnou změnou intenzity.

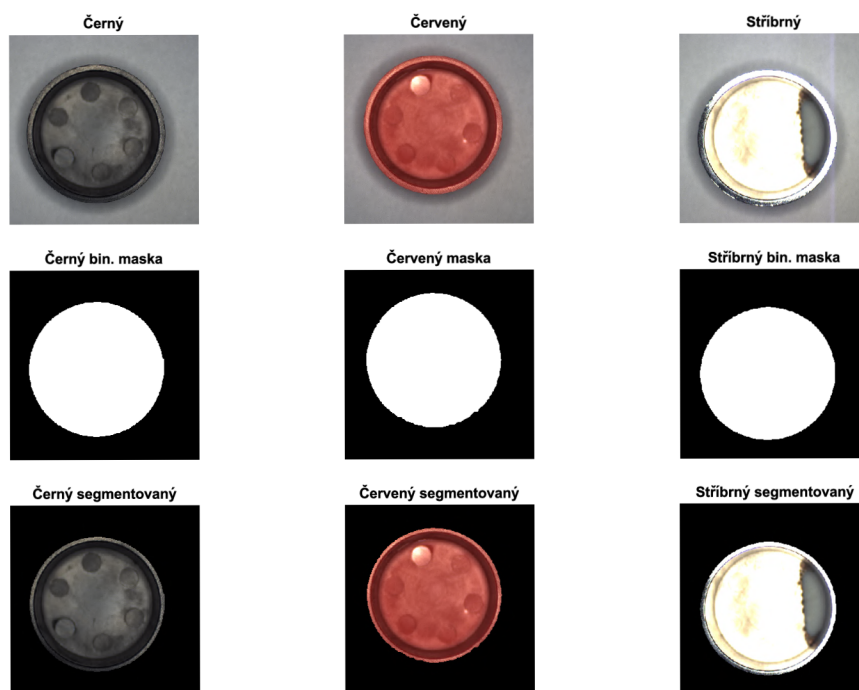
Tato aplikace však typicky vede k fragmentovaným reprezentacím hran, které často vypadají jako izolované tečky místo nepřerušovaných čar. Aby se tento problém vyřešil, používají se morfologické operace k vylepšení segmentace. První aplikovaná morfologická operace je **uzavření** (close), která má za cíl přemostit mezery mezi izolovanými tečkami v bezprostřední blízkosti. Tato operace pomáhá vytvořit souvislý vnější kruh, který naznačuje kruhový tvar objektu.

Po operaci uzavření je implementována operace **vyplnění** (fill), která zaplní uzavřenou oblast kruhu a zpevní tak segmentaci kruhu. Protože reziduální šum ve formě malých teček může stále existovat mimo hlavní okrajový kruh, používá se operace **otevření** (open) k odstranění těchto nadbytečných prvků. Nakonec je provedena operace **eroze** (erode) ke zmenšení rozměrů detekovaného kruhu, čímž se zajistí, že segmentace je pevně vázána na skutečný objekt.



Obrázek 11.3: Postup segmentace

Výsledná maska vytvořená těmito kroky je následně vynásobena s původním obrázkem. Toto vynásobení efektivně izoluje objekt tím, že eliminuje pozadí a zanechává ve snímku viditelný pouze výrobek. Tato metoda zajišťuje, že se pozornost soustředí výhradně na významné rysy výrobku pro další analýzu nebo zpracování.



Obrázek 11.4: Segmentované výrobky

12 Metody kontroly kvality

Tato kapitola představuje průzkum několika metod používaných pro inspekci obrazu, přičemž každá byla vybrána pro její relevanci a účinnost v konkrétních aplikacích. Diskutované metody zahrnují:

- Průměrná hodnota RGB
- Průměrná hodnota odstínů šedi a odstínu
- Nekompaktnost
- MACE

12.1 Průměrná hodnota RGB

Tato metoda spočívá ve výpočtu průměrných hodnot červeného, zeleného a modrého kanálu napříč obrazem. Poskytuje základní statistickou reprezentaci celkové barevné intenzity a distribuce v obraze, což může být zásadní pro úkoly, kde informace o barvě hrají klíčovou roli.

Průměrnou hodnotu RGB lze vypočítat jako:

$$R_{avg} = \frac{1}{L} \cdot \sum_{i=1}^L R_i, \quad G_{avg} = \frac{1}{L} \cdot \sum_{i=1}^L G_i, \quad B_{avg} = \frac{1}{L} \cdot \sum_{i=1}^L B_i \quad (12.1)$$

Výsledný RGB_{avg} vektor lze napsat jako:

$$\mathbf{RGB}_{avg} = \begin{pmatrix} R_{avg} \\ G_{avg} \\ B_{avg} \end{pmatrix} \quad (12.2)$$

Nalezená průměrná hodnota výrobků je:

$$\mathbf{\check{C}ern\acute{y}} = \begin{pmatrix} 72 \\ 70 \\ 69 \end{pmatrix}, \quad \mathbf{\check{C}erven\acute{y}} = \begin{pmatrix} 166 \\ 73 \\ 65 \end{pmatrix}, \quad \mathbf{St\acute{r}ibrn\acute{y}} = \begin{pmatrix} 223 \\ 221 \\ 210 \end{pmatrix}$$

Po výpočtu průměrných hodnot pro všechny výrobky je kolem každé průměrné hodnoty stanoveno interval ± 10 . Během hodnotícího procesu se průměrná hodnota každého testovaného výrobku zkontroluje, aby se zjistilo, zda spadá do některého











z těchto předem definovaných intervalů. Pokud průměrná hodnota testovaného výrobku nespadá do žádného ze specifikovaných intervalů, je výrobek klasifikován jako NOK.

12.2 Průměrná hodnota odstínů šedi a odstínu

Vzhledem k výpočetním nárokům spojeným se zpracováním velkých obrázků nebo vysokého objemu obrázků s použitím RGB barevného schématu je často praktické převést tyto obrázky do jiného barevného schématu, jako je odstín šedi. Tento převod se dosahuje přepočtem RGB hodnoty každého pixelu na hodnotu v odstínech šedi, jak je definováno rovnicí 12.3. Tento proces zahrnuje promítání 3D matice (reprezentující RGB hodnoty) do 2D matice (hodnoty v odstínech šedi), což nevyhnutelně vede ke ztrátě informací. V důsledku toho mohou různé RGB barvy poskytnout identické ekvivalenty v odstínech šedi, což znamená, že pouhé odstíny šedi nejsou spolehlivé pro detekci barev. Ukázka převodů z RGB do šedo-tónové a HSV oblasti lze vidět v tabulce 12.1.

$$Gray = 0,3R + 0,59G + 0,11B \quad (12.3)$$

Tabulka 12.1: Konverze mezi barevnými prostory

Barva	RGB	Grayscale	H	S	V
	(0,0,0)	0	0	0 %	0 %
	(255,255,255)	255	0	0 %	100 %
	(255,0,0)	76	0	100 %	100 %
	(0,255,0)	150	120	100 %	100 %
	(0,0,255)	28	240	100 %	100 %
	(255,255,0)	226	60	100 %	100 %
	(0,255,255)	178	180	100 %	100 %
	(255,0,255)	104	-60	100 %	100 %
	(191,191,191)	191	0	0 %	74.9 %
	(128,128,128)	128	0	0 %	50.2 %

Pro zmírnění tohoto problému a zlepšení informací o barvě se často zahrnuje dodatečná barevná složka. V tomto konkrétním případě byla vybrána složka Hue z barevného schématu HSV, která doplňuje hodnoty v odstínech šedi. Tato kombinace účinně transformuje data obrázku z matice $(m, n, 3)$ na matici $(m, n, 2)$. Výpočty pro převod RGB na HSV, včetně určení složky Hue, jsou popsány v rovnicích od 12.4 do 12.7.

Následující vzorce jsou převzaté od Gonzalez a Woods [19].

$$\begin{aligned} C_{max} &= \max(R, G, B) \\ C_{min} &= \min(R, G, B) \\ \Delta &= C_{max} - C_{min} \end{aligned} \quad (12.4)$$

$$H = \begin{cases} 0^\circ & \delta = 0 \\ 60^\circ \times \left(\frac{G-B}{\Delta} + 0\right) & , C_{max} = R \\ 60^\circ \times \left(\frac{B-R}{\Delta} + 2\right) & , C_{max} = G \\ 60^\circ \times \left(\frac{R-G}{\Delta} + 4\right) & , C_{max} = B \end{cases} \quad (12.5)$$

$$S = \begin{cases} 0 & , C_{max} = 0 \\ \frac{\Delta}{C_{max}} & , C_{max} \neq 0 \end{cases} \quad (12.6)$$

$$V = C_{max} \quad (12.7)$$

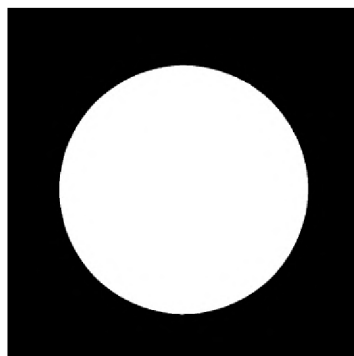
Tato adaptace umožňuje přesnější detekci barev při zachování výpočetní efektivity. Zvláště integrace složky Hue poskytuje klíčový rozměr informací o barvě, který se ztrácí při převodu RGB na pouhé odstíny šedi.

Inspekce testovaného výrobku následuje podobný postup jako dříve popsaná metoda. Nicméně v tomto případě průměrný vektor obsahuje pouze dvě hodnoty.

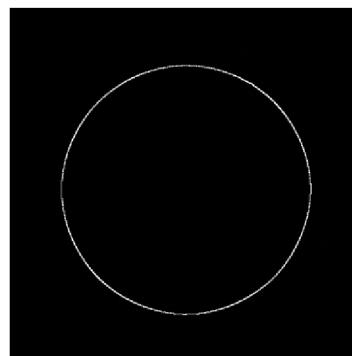
12.3 Nekompaktnost

Tato metoda hodnotí geometrické vlastnosti detekovaných prvků na obrázku, zejména se zaměřuje na aspekt, jak jsou tvary roztříštěné nebo koncentrované. Nekompaktnost se vypočítává na základě obvodu a plochy tvarů, což poskytuje informace o pravidelnosti a rozložení prvků v obraze. Na obrázku 12.1 lze vidět oblasti použité při výpočtu. Na obrázku 12.2 jsou znázorněné výsledné hodnoty nekompaktnosti pro OK a NOK výrobky.

$$\text{Nekompaktnost} = \frac{\text{Pouze hranice}^2}{\text{Celý výrobek}} \quad (12.8)$$

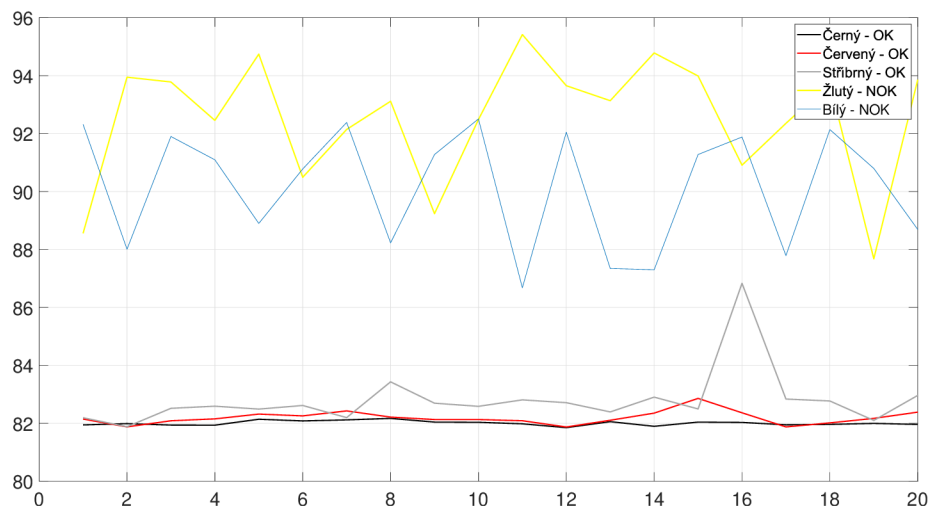


(a) Celý výrobek



(b) Pouze hranice

Obrázek 12.1: Nekompaktnost



Obrázek 12.2: Výsledné hodnoty nekompaktnosti

12.4 Minimum Average Correlation Energy

Minimum Average Correlation Energy (MACE) filtr je sofistikovaný nástroj používaný ve zpracování obrazu k měření podobnosti mezi porovnávanými obrázky prostřednictvím korelačních technik. V tomto kontextu vyšší korelační hodnoty znamenají větší podobnost mezi obrázky. Hlavním cílem MACE filtru je minimalizovat energii napříč korelační rovinou pro všechny vzory, které nejsou cílem. Tato minimalizace je klíčová, protože pomáhá snižovat pozadí a vliv prvků, které nejsou cílové, ve vizuálních datech. [18]

Zároveň je MACE filtr navržen tak, aby zajistil výrazný vrchol v korelační rovině, když je přítomen cílový vzor. Tento charakteristický vrchol slouží jako jasný ukazatel, efektivně odlišuje cíl od okolního šumu a ostatních ne-cílových vzorů.

Korelace mezi filtrem h a obrazem x může být reprezentována jako:

$$g_i(n) = h(n) * x_i(n) \quad (12.9)$$

Vzhledem k výpočetně náročné povaze přímých korelačních operací se jako efektivnější alternativa využívá diskrétní Fourierova transformace (DFT). DFT zjednodušuje proces korelace převedením dat prostorové domény do frekvenční domény, kde násobení nahrazuje konvoluci, čímž se snižuje výpočetní složitost. Energie v korelační rovině pak může být vypočítána jako:

$$\begin{aligned}
E_i &= \frac{1}{d} \sum_j |G_i(j)|^2 \\
&= \frac{1}{d} \sum_j |H(j)|^2 |X_i(j)|^2 \\
&= \frac{1}{d} h^T D_i h
\end{aligned} \tag{12.10}$$

Kde D_i je definovaná jako diagonální matice spočítaná jako:

$$D_{jj} = \sum_{i=1}^L |x_i(j)|^2 \quad j = 1, \dots, N \tag{12.11}$$

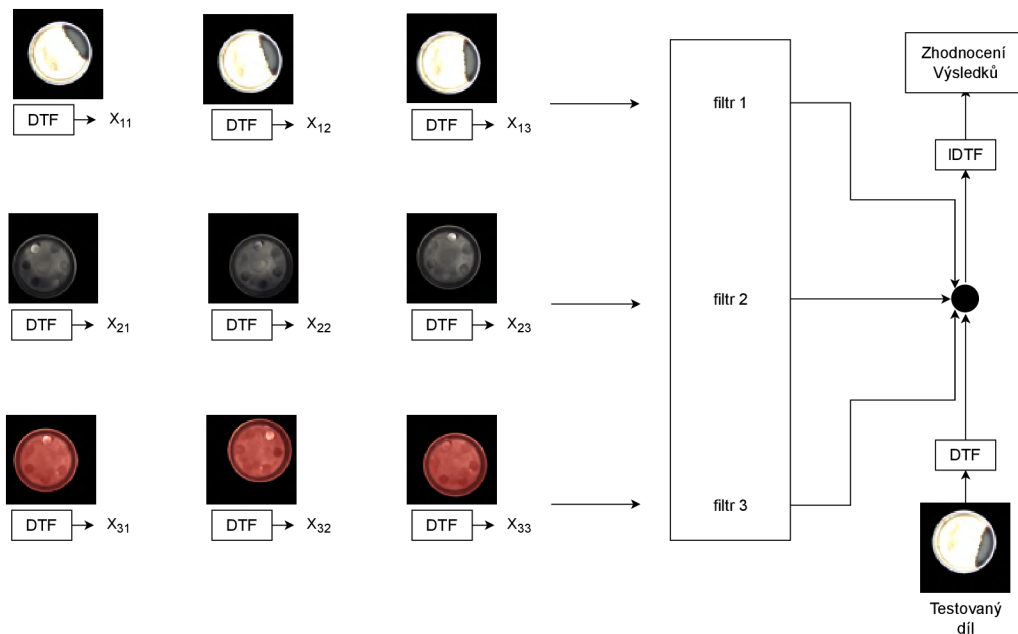
Podle metody Lagrangeových multiplikátorů, jak ji popisuje Mahalanobis (1987) [20], lze výsledný filtr v kontextu zpracování obrazu účinně přepsat jako:

$$H = D^{-1} X (X^T D^{-1} X)^{-1} u \tag{12.12}$$

Předchozí vzorce byly převzaté od Mahalanobis et al. [20].

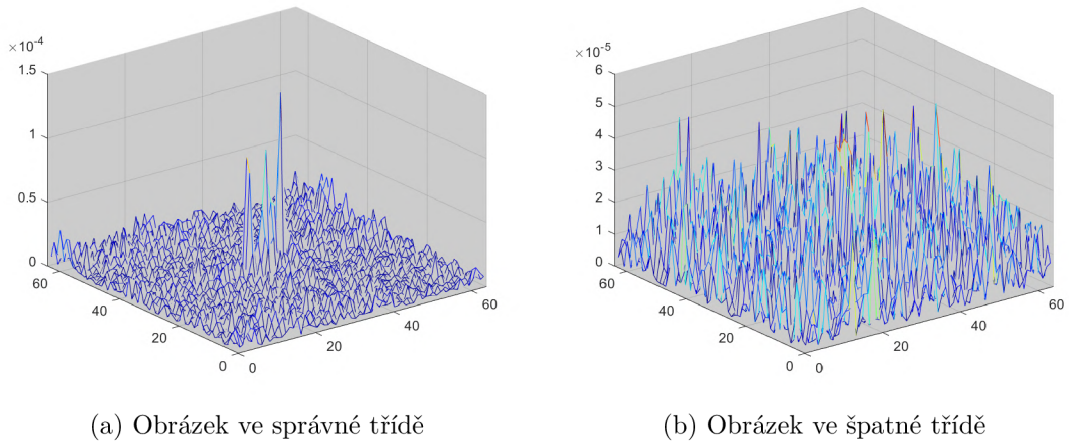
V této práci jsou obrázky výrobků zmenšeny na rozlišení 64×64 pixelů. Pro vytvoření filtru byly ke každému výrobku použity tři odlišné obrázky výrobků, z nichž byla vytvořena matice X o rozměrech 4096×3 , kde každý sloupec představuje vektor zmenšeného obrázku. Dále, matice D má rozměry 4096×4096 . Vektor u má rozměry 3×1 , a výsledný filtr vektor H má rozměry 4096×1 .

Proces klasifikace, který demonstruje aplikaci MACE filtru při identifikaci a rozlišování mezi různými typy výrobků, je znázorněn na obrázku 12.3.

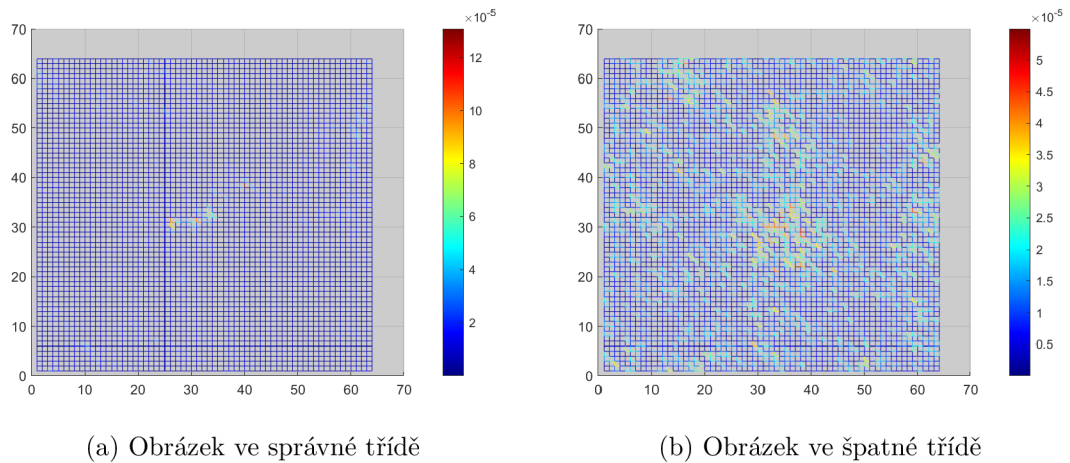


Obrázek 12.3: MACE filter

Vizualizace funkce MACE filtru je zobrazená na obrázcích 12.4 a 12.5.



Obrázek 12.4: 3d spektrum

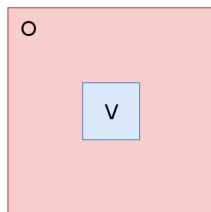


Obrázek 12.5: 2d spektrum

K hodnocení této metody se využívá poměr definovaný rovnicí 12.13, kde specifické oblasti jsou znázorněny na obrázku 12.6. V tomto kontextu V představuje oblast kolem vrcholu, která je umístěna ve středu obrázku a je velká 10×10 pixelů. Oblast O zahrnuje okolí, s výjimkou centrální oblasti V , a má velikost 20×20 pixelů.

Hodnotící proces zahrnuje výpočet poměru RES pro každý testovaný výrobek. Testovaný výrobek je pak zařazen do kategorie, se kterou dosáhne nejvyšší hodnoty RES . [21]

$$RES = \frac{\max(V) - \text{mean}(O)}{\text{std}(O)} \quad (12.13)$$



Obrázek 12.6: Oblast zhodnocení

12.5 Zhodnocení metod

Pro důkladné vyhodnocení účinnosti popsaných metod zpracování obrazu byly vyrobeny dva nevyhovující (NOK) výrobky pomocí 3D tisku. Tyto výrobky jsou znázorněny na obrázku 12.7. Aby byla vytvořena komplexní sada dat pro analýzu, bylo u každého výrobku pořízeno několik snímků. Konkrétně bylo pro každý OK a NOK výrobek pořízeno 20 snímků.

Cílem této analýzy je ověřit přesnost a spolehlivost technik zpracování obrazu při identifikaci OK výrobků, což je zásadní pro kontrolu kvality ve výrobních prostředích. Tento přístup zajišťuje, že metody zpracování obrazu jsou účinně testovány a optimalizovány pro reálné výrobní scénáře, což umožňuje přesné rozlišení mezi OK a NOK výrobky.



(a) Bílý NOK výrobek (b) Žlutý NOK výrobek

Obrázek 12.7: NOK výrobky

Zhodnocení se provádí výpočtem přesnosti (accuracy), preciznosti (precision), úplnosti (recall) a F1 skóre. Přesnost měří množství správných klasifikací, preciznost měří množství správně klasifikovaných pozitivních případů, úplnost měří schopnost detektoru najít všechny relevantní objekty a F1 skóre měří účinnost klasifikátoru.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (12.14)$$

$$Precision = \frac{TP}{TP + FP} \quad (12.15)$$

$$Recall = \frac{TP}{TP + FN} \quad (12.16)$$

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (12.17)$$

Termíny TP, TN, FP, FN označují:

- TP - True positive - počet správně identifikovaných pozitivních případů
- TN - True negative - počet správně identifikovaných negativních případů
- FP - False positive - počet nesprávně identifikovaných pozitivních případů (nesprávně označené jako pozitivní)
- FN - False negative - počet nesprávně identifikovaných negativních případů (nesprávně označené jako negativní)

Výsledky zhodnocení jsou vypsány v tabulkách 12.2 a 12.3.

Tabulka 12.2: Zhodnocení metod pro inspekci barvy

Metoda	TP	TN	FP	FN	Acc %	Prec %	Recall %	F1 %
avg RGB								
Černý	20	40	0	0	1	1	1	1
Červený	20	40	0	0	1	1	1	1
Stříbrný	15	40	0	5	0,91	1	0,75	0,85
avg Gray + Hue								
Černý	18	40	0	2	0,96	1	0,9	0,95
Červený	20	40	0	0	1	1	1	1
Stříbrný	18	40	0	2	0,96	1	0,9	0,95
MACE								
Černý	14	34	6	6	0,8	0,7	0,7	0,7
Červený	13	33	7	7	0,76	0,65	0,65	0,65
Stříbrný	17	37	3	3	0,9	0,85	0,85	0,85

Tabulka 12.3: Zhodnocení metod pro inspekci tvaru

Metoda	TP	TN	FP	FN	Acc %	Prec %	Recall %	F1 %
Nekompaktnost	59	40	0	1	0,99	1	0,98	0,99
MACE	60	40	0	0	1	1	1	1

Veškeré zdrojové kódy použité pro zpracování obrazu jsou v příloze P.2. Snímky použité pro testování jsou v příloze O.

13 Závěr

Tato diplomová práce byla zaměřená na integraci robotického systému a kamerového inspekčního systému v rámci Festo Modular Production System (MPS). Práce začala úvodním přehledem vybraných komponent: robot UR3, robotické nástroje, inspekční systémy a rámec Festo MPS.

Ovládání robota UR3 bylo zajištěno prostřednictvím stávající knihovny Matlab, která byla původně vyvinuta jako součást bakalářské práce. Protože bylo nutné rozšířit tuto knihovnu, byla provedena úprava pro začlenění zpětnovazebního signálu po dokončení pohybu robota. Toto vylepšení bylo klíčové pro přesné řízení a synchronizaci v systému.

V dalším kroku byl navržen specializovaný robotický nástroj, který podporuje dvě funkce: přísavku pro manipulaci s materiálem a kameru pro inspekční účely. Systém přísavky byl doplněn o vakuový ejektor, zatímco inspekční kamera byla vybavena osvětlením pro optimalizaci viditelnosti a přesnosti během inspekčního procesu.

Pro tuto práci byla vybrána kamera Basler ACE 1300-30gc. Jelikož tato kamera nepřícházela s nativním softwarem schopným programovat požadované inspekce, byl v Matlabu vyvinut vlastní program. Prvním krokem inspekčního procesu bylo použití Sobelova detektoru hran pro segmentaci kontrolovaného výrobku od jeho pozadí. Následné morfologické operace – otevření, uzavření a eroze – byly aplikovány k doladění segmentace, což vyústilo ve vytvoření binární masky, která byla použita společně s původním obrazem k izolaci kontrolovaného výrobku.

Inspekční parametry se zaměřily na dva hlavní aspekty: barvu a tvar. Pro detekci barvy byly naprogramované metody pro analýzu průměrných RGB hodnot a kombinaci odstínů šedi a Hue. Z hlediska tvaru byly použity metody založené na nekompaktnosti a filtru Minimum Average Correlation Energy (MACE).

Hodnocení těchto metod bylo provedeno na základě metrik pravých pozitiv (TP), pravých negativ (TN), falešných pozitiv (FP) a falešných negativ (FN). Metoda detekce RGB barev dosáhla úspěšnosti 100 % pro červené a černé kusy a 85 % pro stříbrné kusy. Metoda odstínů šedi a odstínů dosáhla úspěšnosti 95 % pro černé a stříbrné kusy a 100 % pro červené kusy. Filtr MACE byl také vyzkoušen pro detekci barev, ale jeho výkon byl mnohem horší než předchozí dvě metody. Pro detekci tvaru dosáhla metoda nekompaktnosti úspěšnosti 99 %, zatímco metoda filtru MACE dosáhla 100 %. Všechny metody byly testovány na 20 vyhovujících a 20 nevyhovujících kusech každého typu, aby byla zajištěna robustnost a spolehlivost zjištění.

Integrace robotického a inspekčního systémů do Festo MPS byla provedena umís-

těním jednotlivých modulů vedle sebe. Po zarovnání byly tyto moduly pevně spojeny, aby se předešlo jakémukoli posunu během provozu. Tato fyzická integrace byla zásadní pro zajištění stabilního a soudržného uspořádání, které umožňuje synchronizovanou interakci mezi různými komponentami MPS.

Komunikace mezi robotem UR3 a MPS byla usnadněna prostřednictvím přímého kabelového spojení, které propojovalo I/O porty robota s I/O porty PLC. Toto pevné připojení umožnilo bezproblémovou koordinaci mezi robotickými pohyby a procesy řízenými PLC. V budoucnu by však mohlo dojít k úpravě nastavením TCP/IP spojení mezi všemi zařízeními.

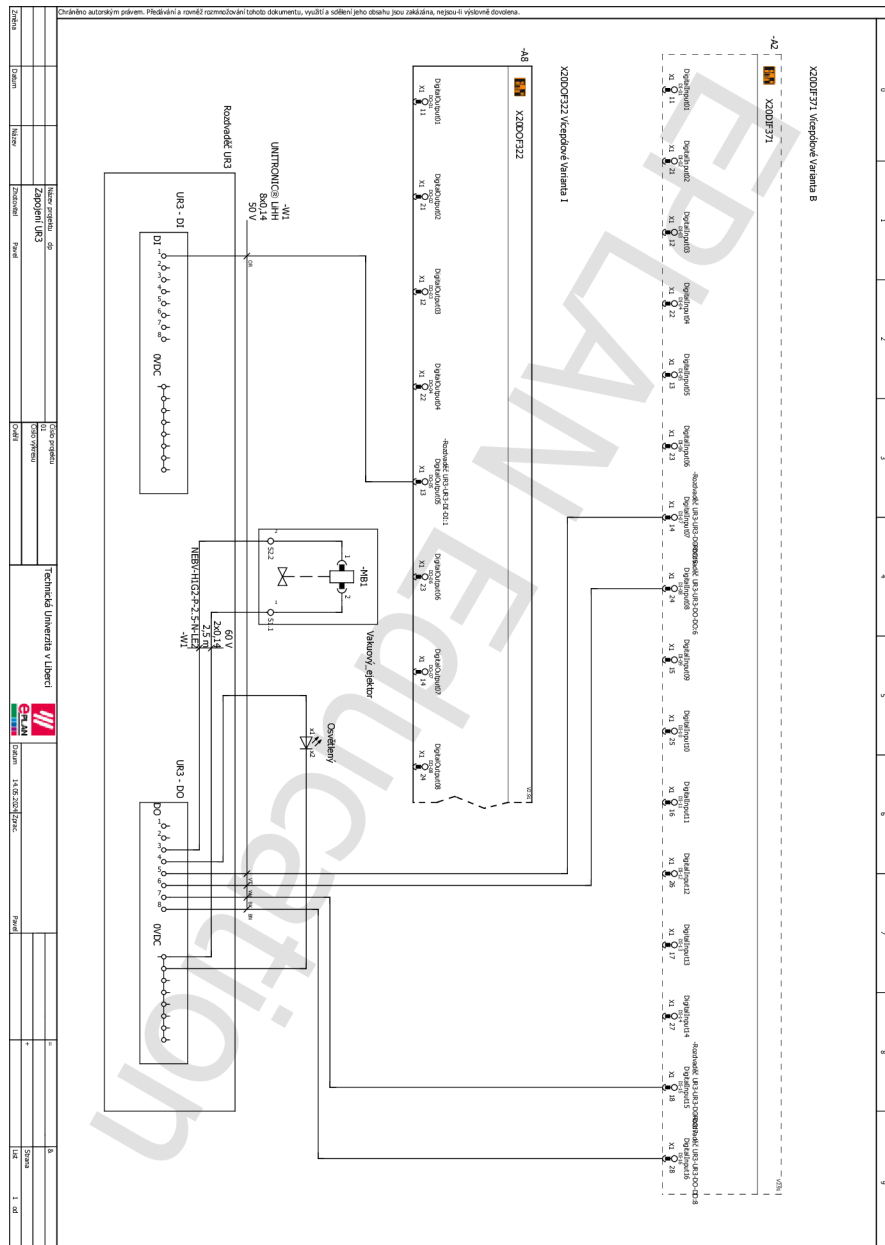
Inspekční proces prováděný integrovaným kamerovým systémem se ukázal jako vysoce účinný, přesně identifikoval charakteristiky každého výrobku. Úspěšné rozlišení mezi vyhovujícími a nevyhovujícími výrobky vedlo k jejich správnému rozřídění. Tento výsledek nejen potvrdil účinnost kamerového inspekčního systému, ale také prokázal robustnost celého integrovaného systému při plnění svých určených úkolů.

Použitá literatura

- [1] ROBOTS, Universal. *UR3 User Manual* [online]. [cit. 2024-04-13]. Dostupné z: https://s3-eu-west-1.amazonaws.com/ur-support-site/21984/UR3_User_Manual_en_Global.pdf.
- [2] MÜLLER, Pavel. *Ovládání Universal Robotu z nadřazeného programu*. Liberec: Technická univerzita v Liberci, 2022.
- [3] NOVOTNÝ, František a Marcel HORÁK. *Efektory průmyslových robotů*. Vydání 1. Liberec: Technická univerzita v Liberci, 2015. ISBN 978-80-7494-195-5.
- [4] NOVOTNÝ, František et al. *Úvod do automatizace a robotizace ve strojírenství*. Vydání 1. Liberec: Technická univerzita v Liberci, Fakulta strojní, Katedra sklářských strojů a robotiky, 2020. ISBN 978-80-7494-545-8.
- [5] DIDACTIC, Festo. *Chapadla* [online]. [cit. 2024-04-20]. Dostupné z: https://www.festo.com/cz/cs/c/vyroby/prumyslova-automatizace/chapadla-id_pim227/.
- [6]), robotiq inc. *Hand-E Adaptive Gripper* [online]. [cit. 2024-04-26]. Dostupné z: <https://robotiq.com/products/hand-e-adaptive-robot-gripper>.
- [7] INC., Schunk. *EGM Catalog Section* [online]. [cit. 2024-04-23]. Dostupné z: https://schunk.com/us/en/gripping-systems/magnetic-gripper/egm/c/PGR_3014.
- [8] INC., robotiq. *Vacuum Grippers* [online]. [cit. 2024-04-26]. Dostupné z: https://robotiq.com/products/vacuum-grippers?ref=nav_product_new_button.
- [9] DIDACTIC, Festo. *Vakuové přísavky* [online]. [cit. 2024-04-20]. Dostupné z: https://www.festo.com/cz/cs/c/vyroby/prumyslova-automatizace/vakuova-technika/vakuove-prisavky-id_pim113/?page=0.
- [10] HOTAŘ, Vlastimil. *Úvod do problematiky strojového vidění, část 1*. Vydání 1. V Liberci: Technická univerzita, 2015. ISBN 978-80-7494-156-6.
- [11] BELGIUM, KEYENCE International. *CV-X100 Series User's Manual (English) OP-M1845* [online]. [cit. 2024-04-16]. Dostupné z: <https://www.keyence.eu/products/vision/vision-sys/cv-x100/>.
- [12] AG, Basler. *aca1300-60gc* [online]. [cit. 2024-04-16]. Dostupné z: <https://docs.baslerweb.com/aca1300-60gc>.

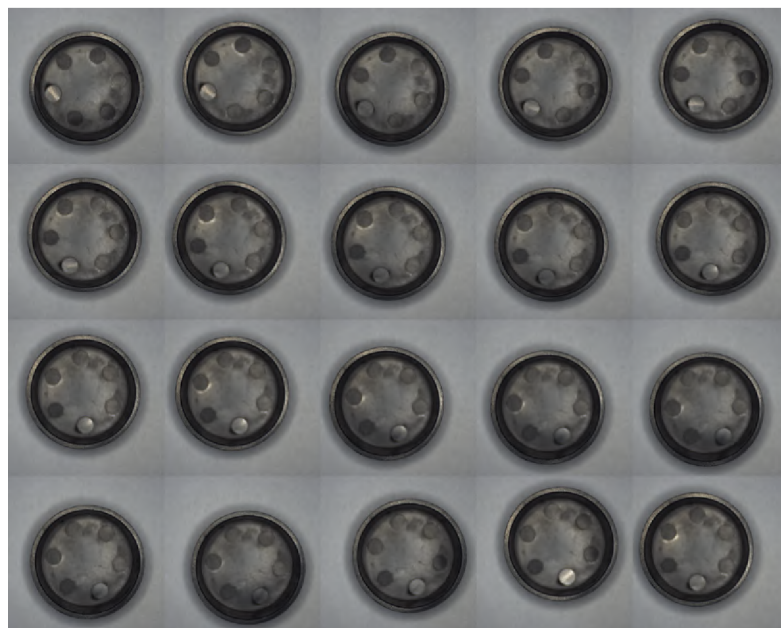
- [13] MICRO-EPSILON. *scanCONTROL: Laser Profile Sensors* [online]. [cit. 2024-04-18]. Dostupné z: <https://www.micro-epsilon.com/fileadmin/download/products/cat--scanCONTROL--en.pdf>.
- [14] BELGIUM, KEYENCE International. *LJ-X8000 Series 2D/3D Laser Profiler Catalogue* [online]. [cit. 2024-04-16]. Dostupné z: <https://www.keyence.eu/mykeyence/?ptn=001>.
- [15] DIDACTIC, Festo. *MPS-D Distributing station Getting started* [online]. [cit. 2024-04-20]. Dostupné z: https://www.festo.com/cz/cs/p/stanice-distribuce-id_PROD_DID_8034566/?page=0.
- [16] DIDACTIC, Festo. *MPS-D Joining station Getting started* [online]. [cit. 2024-04-20]. Dostupné z: https://www.festo.com/cz/cs/p/stanice-spojovani-id_PROD_DID_8063910/?page=0.
- [17] DIDACTIC, Festo. *MPS-D Sorting Station Getting Started* [online]. [cit. 2024-04-20]. Dostupné z: https://www.festo.com/cz/cs/p/stanice-trideni-id_PROD_DID_8046325/?page=0.
- [18] HLAVÁČ, Václav a Miloš SEDLÁČEK. *Zpracování signálů a obrazů*. Vyd. 1. Praha: Vydavatelství ČVUT, 2000. ISBN 80-01-02114-9.
- [19] GONZALEZ, Rafael C. a Richard E. WOODS. *Digital image processing*. 3rd ed. Upper Saddle River: Pearson, c2008. ISBN 9780131687288.
- [20] MAHALANOBIS, Abhijit, B. V. K. Vijaya KUMAR a David CASASANT. Minimum average correlation energy filters. 1987, roč. 26. Dostupné z DOI: [10.1364/AO.26.003633](https://doi.org/10.1364/AO.26.003633).
- [21] SAVVIDES, Marios, B. KUMAR a Pradeep KHOSLA. Face verification using correlation filters. *3rd IEEE Automatic Identification Advanced Technologies*. 2002. Dostupné také z: https://www.researchgate.net/publication/228603657_Face_verification_using_correlation_filters.

N Schéma zapojení

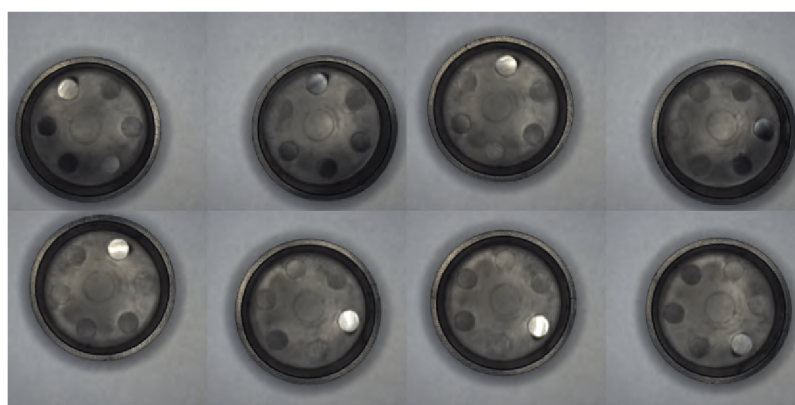


Obrázek N.1:

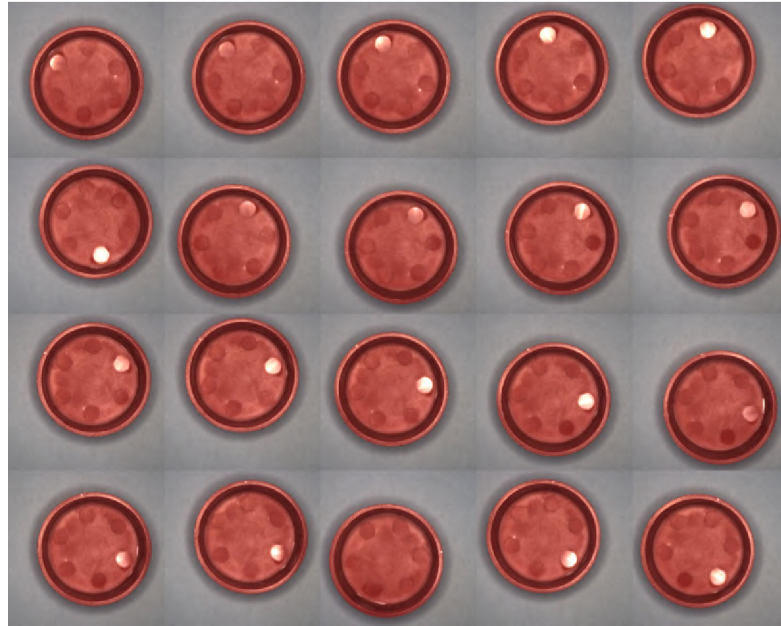
O Testované obrázky



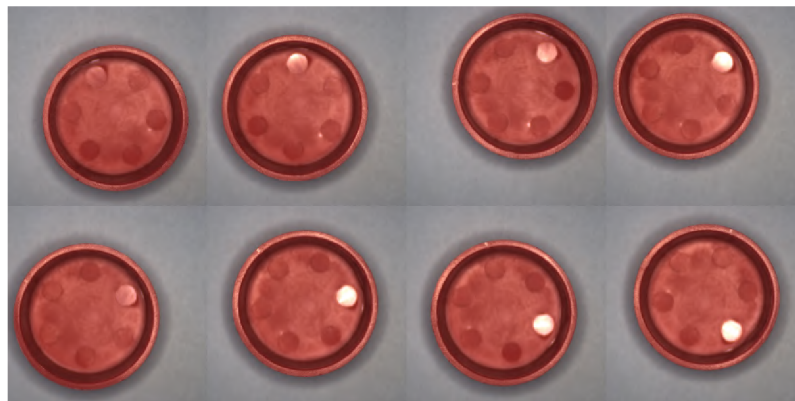
Obrázek O.1: Černý výrobky



Obrázek O.2: Černý klasifikační obrázky



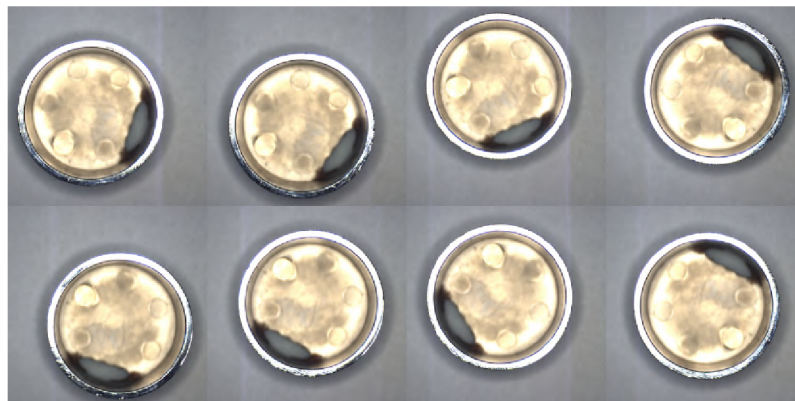
Obrázek O.3: Červený výrobky



Obrázek O.4: Červený klasifikační obrázky



Obrázek O.5: Stříbrný výrobky



Obrázek O.6: Stříbrný klasifikační obrázky



Obrázek O.7: Žlutý NOK výrobek



Obrázek O.8: Bílý NOK výrobek

P Zdrojové kódy

P.1 Robot

Zdrojový kód P.1: Handshake

```
1     function [out,ind] = movej_with_feedback(client,
2         pozice, zrychleni, rychl,t,r,ind)
3     if isempty(zrychleni)
4         zrychleni = 0.5;
5     end
6     if isempty(rychl)
7         rychl = 1;
8     end
9     if isempty(t)
10        t = 0;
11    end
12    if isempty(r)
13        r = 0;
14    end
15    if isempty(ind)
16        ind = 0;
17    end
18    n = 1;
19    while true
20        switch n
21            case 1
22                UR3_lib.pohybj(client, pozice, zrychleni, rychl,t,r
23                    );
24                n = 2;
25            case 2
26                while true
27                    client = tcpclient(client.Address, client.Port);
28                    current =UR3_lib.kloub_natoceni_cil(client);
29                    dif = abs(current -pozice);
30                    if dif < 0.005
31                        n = 3;
32                    break
```

```

31     end
32     end
33     case 3
34     out = "Done";
35     ind = ind +1;
36     break
37     end
38     end
39     if nargout < 2
40     clear ind;
41     end
42     if nargout < 1
43     clear out;
44     end
45     end

```

Zdrojový kód P.2: kompletní program

```

1     ip = '169.254.153.50';
2     port = 30003;
3     client = tcpclient(ip, port);
4
5     cam_list = gigeclist;
6     Cam = gigeclist(1, 'PixelFormat', 'BayerBG8');
7     Cam = cam_param(Cam);
8
9     rychl = 2.2;
10    zrychleni = 0.5;
11    n = 0;
12    while true
13    switch n
14    case 0
15    UR3_lib.set_dig_out(client,6, 'True')
16    pause(0.2)
17    [feed, n] =UR3_lib.movej_with_feedback(client,
18        home_pos, zrychleni, rychl, 0,0,n);
19
20    case 1
21
22    dig_in = UR3_lib.dig_vstupy(client);
23    if dig_in == 769
24    UR3_lib.set_dig_out(client,6, 'False')
25    pause(0.2)
26    n = 2;
27    end
28    case 2

```

```

28     [feed, n] =UR3_lib.movej_with_feedback(client,
29         pos_apc,zrychleni,rychl,0,0,n);
30     case 3
31     [feed, n] =UR3_lib.movej_with_feedback(client,
32         pos_pc,zrychleni,rychl,0,0,n) ;
33     case 4
34     pause(0.2)
35     UR3_lib.set_dig_out(client,2, 'True')
36     pause(0.2)
37     n = 5;
38     case 5
39     [feed, n] =UR3_lib.movej_with_feedback(client,
40         pos_apc,zrychleni,rychl,0,0,n) ;
41     case 6
42     UR3_lib.set_dig_out(client,7, 'True')
43     pause(0.1)
44     [feed, n] =UR3_lib.movej_with_feedback(client,
45         pos_acp,zrychleni,rychl,0,0,n) ;
46     case 7
47     [feed, n] =UR3_lib.movej_with_feedback(client,
48         pos_acp1,zrychleni,rychl,0,0,n) ;
49     case 8
50     [feed, n] =UR3_lib.movej_with_feedback(client,
51         pos_cp,zrychleni,rychl,0,0,n) ;
52     case 9
53     UR3_lib.set_dig_out(client,2, 'False')
54     pause(0.2)
55     UR3_lib.set_dig_out(client,7, 'False')
56     if UR3_lib.dig_vystup(client) == 768
57     n = 10;
58     end
59     case 10
60     [feed, n] =UR3_lib.movej_with_feedback(client,
61         pos_acp1,zrychleni,rychl,0,0,n) ;
62     case 11
63     [feed, n] =UR3_lib.movej_with_feedback(client,
64         pos_acp,zrychleni,rychl,0,0,n) ;
65     case 12
66     [feed, n] =UR3_lib.movej_with_feedback(client,
67         pos_ts,zrychleni,rychl,0,0,n) ;
68     case 13
69     UR3_lib.set_dig_out(client,3, 'True')
70     pause(1)

```

```

64     img = snapshot(Cam);
65     pause(0.4)
66     UR3_lib.set_dig_out(client,3,'False')
67     [im] = img_prog.segm_edge(img);
68     result = img_prog.simple_RGB_compare(im);
69     pause(0.2)
70     if result == "Red"
71     pause(0.5)
72     UR3_lib.set_dig_out(client,4,'True')
73     pause(0.5)
74     UR3_lib.set_dig_out(client,5,'False')
75     pause(0.5)
76     n = 14;
77     elseif result == "Black"
78     pause(0.5)
79     UR3_lib.set_dig_out(client,4,'False')
80     pause(0.5)
81     UR3_lib.set_dig_out(client,5,'True')
82     pause(0.5)
83     n = 14;
84     elseif result == "Silver"
85     pause(0.5)
86     UR3_lib.set_dig_out(client,4,'True')
87     pause(0.5)
88     UR3_lib.set_dig_out(client,5,'True')
89     pause(0.5)
90     n = 14;
91     else
92     pause(0.5)
93     UR3_lib.set_dig_out(client,4,'False')
94     pause(0.5)
95     UR3_lib.set_dig_out(client,5,'False')
96     pause(0.5)
97     n = 14;
98     end
99     case 14
100    [feed, n] =UR3_lib.movej_with_feedback(client,
        pos_acp,zrychleni,rychl,0,0,n)  ;
101    case 15
102    [feed, n] =UR3_lib.movej_with_feedback(client,
        pos_acp1,zrychleni,rychl,0,0,n)  ;
103    case 16
104    [feed, n] =UR3_lib.movej_with_feedback(client,
        pos_cp,zrychleni,rychl,0,0,n)  ;
105    case 17

```

```

106     UR3_lib.set_dig_out(client,2,'True')
107     pause(0.2)
108     n = 19;
109     case 19
110     [feed, n] =UR3_lib.movej_with_feedback(client,
111         pos_acp1,zrychleni,rychl,0,0,n) ;
112     case 20
113     [feed, n] =UR3_lib.movej_with_feedback(client,
114         pos_acp,zrychleni,rychl,0,0,n) ;
115
116     case 21
117     if result == "Others"
118     [feed, n] =UR3_lib.movej_with_feedback(client,
119         pos_ta,zrychleni,rychl,0,0,49);
120     else
121     if UR3_lib.dig_vstupy(client) == 769
122     [feed, n] =UR3_lib.movej_with_feedback(client,
123         pos_apc,zrychleni,rychl,0,0,n);
124     end
125     end
126     case 22
127     [feed, n] =UR3_lib.movej_with_feedback(client,
128         pos_pc,zrychleni,rychl,0,0,n) ;
129     case 23
130     UR3_lib.set_dig_out(client,2,'False')
131     pause(0.5)
132     UR3_lib.set_dig_out(client,6,'False')
133     pause(0.5)
134
135     n = 24;
136     case 24
137     [feed, n] =UR3_lib.movej_with_feedback(client,
138         pos_apc,zrychleni,rychl,0,0,n);
139
140     case 25
141     UR3_lib.set_dig_out(client,6,'True')
142     pause(0.5)
143     UR3_lib.set_dig_out(client,6,'False')
144     pause(0.5)
145
146     UR3_lib.set_dig_out(client,4,'False')
147     pause(0.5)
148     UR3_lib.set_dig_out(client,5,'False')
149     pause(0.5)
150     n = 0;

```

```

145
146     case 50
147     UR3_lib.set_dig_out(client,2,'False')
148     pause(0.5)
149     n = 25;
150     end
151     end

```

P.2 Zpracování obrazu

Zdrojový kód P.3: Nastavení parametrů kamery

```

1     function Cam = cam_param(Cam)
2     Cam.ExposureAuto = 'off';
3     Cam.ExposureTimeAbs = 900;
4     Cam.TriggerDelayAbs = 8000;
5     Cam.Width = 900;
6     Cam.Height = 900;
7     Cam.CenterY = "True";
8     Cam.CenterX = "True";
9     Cam.AcquisitionFrameRateEnable = "True";
10    Cam.Timeout = 200;
11    Cam.AcquisitionFrameRateAbs = 8;
12    Cam.GainAuto='Off';
13    Cam.GainRaw = 300;
14    Cam.GevSCPSPacketSize = 576;
15    Cam.BalanceWhiteAuto='Off';
16    Cam.GainSelector = "All";
17    Cam.GammaEnable = "True";
18    Cam.BalanceRatioSelector = "Red";
19    Cam.BalanceRatioRaw = 121;
20    Cam.BalanceRatioSelector = "Green";
21    Cam.BalanceRatioRaw = 96;
22    Cam.BalanceRatioSelector = "Blue";
23    Cam.BalanceRatioRaw = 238;
24    end

```

Zdrojový kód P.4: Zpracování obrazu

```

1     function [img_proc] = segm_edge(img)
2     img = imresize(img,0.5);
3     se= strel('disk',13);
4     gray=rgb2gray(img);
5     filt = stdfilt(gray);

```

```

6     BW = img_prog.Sobel(filt);
7     close = imclose(BW,se);
8     fill = imfill(close,'holes');
9     open=imopen(fill,se);
10    erode=imerode(open,se);
11    img_proc = img.*uint8(erode);
12    end
13
14    function out = edge_komp(img)
15    range = [81.5 83.6];
16    img=rgb2gray(img);
17    img(img>0) = 1;
18    outer = img_prog.Sobel(img);
19    sum_img = sum(img,"all");
20    sum_outer = sum(outer,"all");
21    ratio = sum_outer^2 / sum_img;
22    if (ratio > range(1)) && (ratio < range(2))
23    out = 1;
24    else
25    out = 0;
26    end
27
28    end
29
30    function img_out = Sobel(img)
31    img = double(img);
32    Gx = [-2 -1 0; -1 0 1; 0 1 2];
33    Gy = [0 -1 -2;
34    1 0 -1;
35    2 1 0];
36    [rows, cols] = size(img);
37    grad_x = zeros(rows, cols);
38    grad_y = zeros(rows, cols);
39    for i = 2:rows-1
40    for j = 2:cols-1
41    subImage = img(i-1:i+1, j-1:j+1);
42    grad_x(i, j) = sum(Gx .* subImage,"all");
43    grad_y(i, j) = sum(Gy .* subImage,"all");
44    end
45    end
46    img_out = sqrt(grad_x.^2 + grad_y.^2);
47    img_out = mat2gray(img_out);
48    img_out(img_out<0.1) = 0;
49    img_out(img_out>0) = 1;
50    end

```

```

51
52
53     function result = simple_RGB_compare(img)
54     black = [ 65 65 65 ; 80 80 80];
55     red = [ 155 60 55 ; 172 77 70];
56     silver = [ 215 210 195 ; 230 225 215];
57     avg = [mean(nonzeros(img(:,:,1))));
58            mean(nonzeros(img(:,:,2))));
59            mean(nonzeros(img(:,:,3)))]];
60     if (avg(1) > red(1,1)) && (avg(1) < red(2,1)) && (
61         avg(2) > red(1,2)) && (avg(2) < red(2,2)) && ...
62         (avg(3) > red(1,3)) && (avg(3) < red(2,3))
63         result = "Red";
64     elseif (avg(1) > black(1,1)) && (avg(1) < black
65         (2,1)) && (avg(2) > black(1,2)) && (avg(2) <
66         black(2,2)) && ...
67         (avg(3) > black(1,3)) && (avg(3) < black(2,3))
68         result = "Black";
69     elseif (avg(1) > silver(1,1)) && (avg(1) < silver
70         (2,1)) && (avg(2) > silver(1,2)) && (avg(2) <
71         silver(2,2)) && ...
72         (avg(3) > silver(1,3)) && (avg(3) < silver(2,3))
73         result = "Silver";
74     else
75         result = "Others";
76     end
77     end
78
79     function result = GH_compare(img)
80     black = [ 60 21 ; 80 25 ];
81     red = [ 86 2 ; 106 4 ];
82     silver = [ 210 13 ; 230 16 ];
83     gray = rgb2gray(img);
84     [ h,s,v] = rgb2hsv(img);
85     h= rad2deg(h);
86     avg = [mean(nonzeros(gray));
87            mean(nonzeros(h))];
88     if (avg(1) > red(1,1)) && (avg(1) < red(2,1)) && (
89         avg(2) > red(1,2)) && (avg(2) < red(2,2))
90         result = "Red";
91     elseif (avg(1) > black(1,1)) && (avg(1) < black
92         (2,1)) && (avg(2) > black(1,2)) && (avg(2) <
93         black(2,2))
94         result = "Black";
95     elseif (avg(1) > silver(1,1)) && (avg(1) < silver

```



```

        (2,1)) && (avg(2) > silver(1,2)) && (avg(2) <
        silver(2,2))
88 result = "Silver";
89 else
90 result = "Others";
91 end
92 end
93
94 function steepness = MACE(img,imgs)
95 img = imresize(img,[64 64]);
96 img = rgb2gray(img);
97 img = reshape(fft2(img).',1,[]);
98 imgs2 = [];
99 for i = 4:7
100 imgs_t = imresize(imgs{i},[64 64]);
101 imgs_t= (rgb2gray(imgs_t));
102 imgs2 = [imgs2 reshape(fft2(imgs_t).',1,[])];
103 end
104 filt =img_prog.calc(imgs2);
105 matcha = img_prog.match(img,filt);
106 steepness = img_prog.compare(matcha);
107 end
108
109
110 function out= convert(color)
111 g = 0.3*color(1) + 0.59*color(2)+0.11*color(3);
112 if color(1) == max(color)
113 h =60*(0+(color(2)-color(3))/(max(color)-min(color)
114 ));
115 elseif color(2) == max(color)
116 h =60*(2+(color(3)-color(1))/(max(color)-min(color)
117 ));
118 elseif color(3) == max(color)
119 h =60*(4+(color(1)-color(2))/(max(color)-min(color)
120 ));
121 end
122 if max(color) > 0
123 s = (max(color) - min(color))/max(color);
124 else
125 s = 0;
126 end
127 v = max(color);
128 out = [g h s v/255];
129 end

```

```

128     function out= convert2(img)
129     gray = rgb2gray(img);
130     [ h,s,v] = rgb2hsv(img);
131     out = [gray,rad2deg(h)];
132     end
133
134     function H = calc(x)
135     x = transpose(x);
136     D= diag(mean(abs(x).^2));
137     Xp = (conj(x));
138     c_size = size(x,1);
139     u = ones(c_size,1);
140     Dm = inv(D);
141     M0 = Dm*transpose(x);
142     M1 = Xp*Dm;
143     M2 = M1*transpose(x);
144     M2 = inv(M2);
145     M3 = M0*M2;
146     H = M3*u;
147     end
148
149     function out = compare(img)
150     center_x = 32;
151     center_y = 32;
152     V = img(center_x - 10: center_x + 10, center_y -
153           10: center_y + 10);
154     O_without_V = img(1:center_x - 10, :);
155     out = (max(V(:)) - mean(O_without_V(:))) / std(
156           O_without_V(:));
157     end
158     function out = match(img,filt)
159     img = double(img);
160     v = reshape(img,[4096,1]) .* filt;
161     v = reshape(v.',1,[]);
162     b = reshape(v,[64,64]);
163     out = fftshift(abs(ifft2(b)));
164     end
165     function out = eval(in)
166     acc= (in(1) + in(2))/(sum(in));
167     prec= in(1)/(in(1)+in(3));
168     recall= in(1)/(in(1)+in(4));
169     F1= (2*prec*recall)/(prec+recall);
170     out = [acc, prec, recall, F1];
171     end

```