

Ovládání kuličky na nakloněné rovině

Bc. Marek Leopold

Diplomová práce
2024



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2023/2024

ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Bc. Marek Leopold
Osobní číslo: A20826
Studijní program: N0613A140022 Informační technologie
Specializace: Softwarové inženýrství
Forma studia: Kombinovaná
Téma práce: Ovládání kuličky na nakloněné rovině
Téma práce anglicky: Control of a Ball on an Inclined Plane

Zásady pro vypracování

1. Proveďte stručnou literární rešerši zaměřenou na embedded systémy pro ovládání náklonu.
2. Popište obecné způsoby detekce objektu. Zaměřte se na optické systémy a automatizované zpracování obrazu.
3. Vyberte komerčně dostupnou manuálně ovládanou nakloněnou rovinu umožňující náklon desky ve dvou osách. Případně sestavte vlastní model nakloněné roviny. Vyberte vhodné akční členy a řídicí jednotku pro přestavbu nakloněné roviny na automatizované ovládání.
4. Vyberte a sestavte vhodný hardware pro detekci pozice kuličky na nakloněné rovině.
5. Sestavte, naprogramujte a odlaďte celý systém pro regulaci pozice kuličky na nakloněné rovině.
6. Otestujte systém na zvolených úlohách a srovnajte jeho přesnost a rychlost s manuálním ovládáním.

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. IENNE, Paolo a Rainer LEUPERS. *Customizable embedded processors: design technologies and applications*. San Francisco, Calif.: Elsevier/Morgan Kaufmann, 2007, 1 online zdroj (xxviii, 497 p.). The Morgan Kaufmann series in systems on silicon. ISBN 9780080490984. Dostupné také z: https://proxy.k.utb.cz/login?url=http://app.knovel.com/hotlink/toc/id:kpCEPDTA06/customizable_embedded_processors__design_technologies_and_applications
2. ZHANG, Peng. *Advanced industrial control technology*. Amsterdam: William Andrew/Elsevier, 2010, 1 online zdroj (xxi, 842 pages). ISBN 9781437778076. Dostupné také z: https://proxy.k.utb.cz/login?url=http://app.knovel.com/hotlink/toc/id:kpAICT0003/advanced_industrial_control_technology
3. SPAČEK, Luboš. *Číslíkové řízení laboratorního modelu CE 151 Kulička na ploše*. 2016, 62 s. (58 175 znaků). Dostupné také z: <http://hdl.handle.net/10563/38786>
4. DAVIES, E. R. *Machine vision: theory, algorithms, practicalities*. 3rd edition. Amsterdam: Elsevier, Morgan Kaufmann, [2005], xxxiii, 934 s. ISBN 0122060938. Dostupné také z: <http://www.loc.gov/catdir/enhancements/fy0626/2005270146-t.html>
5. PETERS, James F. *Foundations of computer vision: computational geometry, visual image structures and object shape detection*. Cham, Switzerland: Springer, 2017, 1 online resource (xvii, 431 pages). Intelligent systems reference library. Dostupné z: <https://doi.org/10.1007/978-3-319-52483-2>.
6. HORNBERG, Alexander. *Handbook of machine and computer vision: the guide for developers and users*. Second, revised and updated edition. Weinheim, Germany: Wiley-VCH, 2017. Dostupné z: <https://doi.org/10.1002/9783527413409>.

Vedoucí diplomové práce:

Ing. Petr Chalupa, Ph.D.

Ústav automatizace a řídicí techniky

Datum zadání diplomové práce:

5. listopadu 2023

Termín odevzdání diplomové práce:

13. května 2024



doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan

prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 5. ledna 2024

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne

.....
podpis studenta

ABSTRAKT

Diplomová práce se zaměřuje na vývoj systému pro dynamické ovládání kuličky na nakloněné ploše. V práci jsou analyzovány a porovnány různé typy řídicích systémů včetně PID regulátoru a algoritmů založených na principu zpětné vazby. Jádrem systému je využití moderních jednodeskových počítačů a mikrokontrolérů pro řízení a monitorování stavu roviny. Experimentální část zahrnuje návrh a realizaci prototypu, jeho testování a optimalizaci na základě získaných dat.

Klíčová slova: ovládání pohybu, nakloněná rovina, jednodeskové počítače, mikrokontroléry, PID regulátor, zpětná vazba.

ABSTRACT

The thesis focuses on the development of a system for dynamic control of a ball on a tilted surface. It analyzes and compares various types of control systems including PID controllers and algorithms based on the principle of feedback. The core of the system utilizes modern single-board computers and microcontrollers for control and monitoring of the plane's status. The experimental part includes the design and implementation of a prototype, its testing, and optimization based on the data obtained.

Key words: motion control, inclined plane, single-board computers, microcontrollers, PID controller, feedback.

Chtěl bych vyjádřit svou hlubokou vděčnost panu Ing. Petru Chalupovi, PhD., za jeho odborné vedení diplomové práce, které bylo provázeno trpělivostí, ochotou a osobním přístupem. Jeho cenné rady byly klíčové pro úspěšné zpracování mé práce.

Dále bych chtěl poděkovat panu Davidu Obalovi za odbornou pomoc a poskytnutí potřebných prostředků, které umožnily zpracování mechanické a elektronické části mé práce.

Moje poděkování patří také mé rodině a blízkým, kteří mi poskytovali neustálou podporu během celého mého studia.

„Jednou z oblastí, která je pro nás typická, je neúspěch. Myslím, že jsme nejlepším místem na světě pro selhání. Máme v tom bohatou praxi. Selhání a vynález jsou neoddělitelná dvojčata. K vymýšlení nových věcí musíte experimentovat, a pokud předem víte, že to bude fungovat, není to experiment.“

- Jeff Bezos

Prohlašuji, že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD	8
I TEORETICKÁ ČÁST	9
1 ŘÍZENÍ NÁKLONU	10
1.1 JEDNODESKOVÉ POČÍTAČE	10
1.1.1 Arduino	11
1.1.2 Raspberry PI.....	12
1.1.3 BeagleBone	13
1.1.4 Odroid	14
1.1.5 BananaPI	15
1.2 PLC.....	16
1.3 MIKROKONTROLÉRY	17
2 OVLÁDÁNÍ POHYBU	18
2.1 AKTUÁTORY.....	18
2.1.1 Elektrické aktuátory	18
2.1.2 Pneumatické aktuátory	18
2.1.3 Hydraulické aktuátory	19
2.2 ROTAČNÍ MOTORY.....	20
2.2.1 Servomotory	20
2.2.2 Krokové motory	22
3 METODY DETEKCE OBJEKTŮ V OPTICKÝCH SYSTÉMECH A AUTOMATIZOVANÉ ZPRACOVÁNÍ OBRAZU	24
3.1 ZÁKLADNÍ PRINCIPY OPTICKÝCH SYSTÉMŮ	24
3.1.1 Typy a charakteristiky obrazových senzorů.....	24
3.1.2 Objektivy a optika	25
3.1.3 Zpracování signálu a obrazu	25
3.1.4 Kalibrace a nastavení systému	25
3.2 METODY AUTOMATIZOVANÉHO ZPRACOVÁNÍ OBRAZU	26
3.2.1 Předzpracování obrazu	26
3.2.2 Segmentace obrazu.....	26
3.2.3 Extrakce Rysů	27
3.2.3.1 Sobelův operátor	27
3.2.3.2 Cannyho Detektor Hran	28
3.2.4 Integrace a Aplikace.....	28
3.3 ALGORITMY DETEKCE OBJEKTŮ	29
3.3.1 Tradiční Metody Detekce.....	29
3.3.1.1 Hraniční a Konturové Detektory.....	29
3.3.1.2 Sliding Window (Klouzavé okno)	30
3.3.1.3 Histogram Orientovaných Gradientů (HOG)	31
3.3.2 Učení s Učitelem	33
3.3.2.1 Support Vector Machines (SVM).....	33
3.3.2.2 Random Forests	34
3.3.3 Hluboké Učení	35
3.3.3.1 Konvoluční neuronové sítě (CNN).....	35
3.3.3.2 R-CNN (Regions with Convolutional Neural Network Features).....	37

3.3.3.3	YOLO	38
4	PID REGULÁTOR	40
II	PRAKTICKÁ ČÁST	41
5	VÝBĚR NAKLÁPĚCÍ ROVINY	42
5.1	KOMERČNĚ DOSTUPNÉ MANUÁLNĚ ŘÍZENÉ NAKLÁPĚCÍ ROVINY	42
5.1.1	Woody Labyrint	42
5.1.2	Labyrint hra Brio 34000 dřevěné	43
5.2	VLASTNÍ ŘEŠENÍ	43
5.2.1	Konstrukce zařízení	43
5.2.2	Pohyb ramen	49
5.2.3	Snímání polohy kuličky	50
5.2.4	Řízení systému	51
5.2.5	Parametry řešení	51
6	REGULACE POLOHY KULIČKY	52
6.1	URČOVÁNÍ POLOHY KULIČKY	52
6.1.1	Rozpoznání objektů v kamerovém snímku	53
6.1.2	Určování polohy objektů na kamerovém snímku	54
6.2	NAKLÁPĚNÍ ROVINY	58
6.3	REGULÁTOR POLOHY	64
6.4	ŘÍZENÍ TRAJEKTORIE KULIČKY V BLUDIŠTI	66
6.4.1	Definice trajektorie	66
6.4.2	Kontrola dosažení bodu	66
6.4.3	Výpočet dalšího bodu a segmentace trajektorie	66
7	POPIS GUI APLIKACE	67
8	VIZUALIZACE POHYBU	69
9	SROVNÁNÍ AUTOMATICKÉHO A MANUÁLNÍHO OVLÁDÁNÍ	72
	ZÁVĚR	73
	SEZNAM POUŽITÉ LITERATURY	74
	SEZNAM OBRÁZKŮ	78
	SEZNAM PŘÍLOH	80

ÚVOD

Diplomová práce s názvem "Ovládání kuličky na nakloněné rovině" představuje komplexní studii zaměřenou na design, implementaci a optimalizaci systému pro dynamické řízení pohybu kuličky na nakloněné ploše. Cílem této práce je nejen praktické ověření teoretických konceptů v oblasti řídicích systémů, ale také rozvoj a aplikace moderních technologií v reálném prostředí.

V teoretické části je kladen důraz na porozumění základním principům řízení pohybu, dynamiky nakloněné roviny a využití jednodeskových počítačů a mikrokontrolérů, které tvoří základ řídicího systému. Zvláštní pozornost je věnována PID regulátorům a metodám zpětné vazby, jejichž správné nastavení a optimalizace jsou klíčové pro úspěšné dosažení stabilního a přesného ovládání kuličky.

Praktická část práce se zaměřuje na návrh a realizaci prototypu, který je následně podroben důkladnému testování a analýze. Součástí této fáze je i vývoj uživatelského rozhraní, které umožňuje interakci se systémem a sledování jeho výkonu v reálném čase. Výsledky testů a měření jsou pečlivě analyzovány a porovnány s teoretickými předpoklady, což přináší hlubší pochopení fungování systému a možností jeho dalšího vylepšení.

Diplomová práce tak představuje spojení teorie a praxe, kde teoretické znalosti z oblasti automatizace a řídicích systémů jsou aplikovány na konkrétní technické řešení s reálným užitím. Výsledkem je funkční model, který demonstruje možnosti moderních technologií a poskytuje základ pro další výzkum a rozvoj v této fascinující oblasti.

I. TEORETICKÁ ČÁST

1 ŘÍZENÍ NÁKLONU

Řízení náklonu roviny je klíčovým prvkem pro správné fungování zařízení, které pracuje s ovládáním pohybu kuličky. Práce se zaměřuje na implementaci programového řízení pro manipulaci s náklonem roviny za účelem ovládnutí pohybu kuličky. Programové řízení nabízí flexibilitu a možnost komplexního nastavení provozních parametrů podle požadavků.

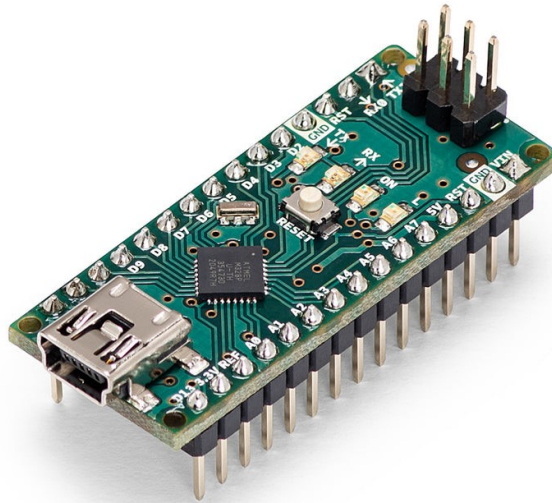
1.1 Jednodeskové počítače

Jednodeskový počítač, známý také jako SBC (Single Board Computer), je typ počítače, který je navržen tak, aby všechny jeho hlavní komponenty a funkce byly umístěny na jedné desce plošného spoje. Tato deska obvykle obsahuje procesor, paměť, vstupně-výstupní porty a další periferie jako napájení a často i možnost rozšíření pomocí různých konektorů.

Jednodeskové počítače jsou často malého fyzického rozměru a mají nízkou spotřebu energie, což je dělá ideálními pro různé projekty v oblastech jako jsou vzdělání, domácí automatizace, robotika, IoT (Internet věcí), hry a další.

Populárními příklady jednodeskových počítačů jsou například Raspberry Pi, Arduino, BeagleBone, Odroid a mnoho dalších. Tyto platformy mají širokou podporu komunity a jsou cenově dostupné, což umožňuje lidem a organizacím využít je pro své projekty a experimenty.

1.1.1 Arduino



Obrázek 1: Arduino Nano [1]

Arduino je open-source platforma pro vývoj elektronických projektů, která je založena na jednodeskovém počítači. Jedná se o malý počítačový modul obsahující mikrokontrolér, který je programovatelný pomocí jednoduchého vývojového prostředí. Arduino je oblíbené mezi elektroniky, studenty a nadšenci do technologie kvůli své jednoduchosti použití a rozsáhlé komunitě.

Hlavní rysy Arduina zahrnují:

- Mikrokontrolér: Arduino obsahuje mikrokontrolér, který řídí všechny funkce a operace. Tento mikrokontrolér je často založen na architektuře AVR nebo ARM.
- Vývojové prostředí: Pro programování Arduina se používá Arduino IDE (Integrated Development Environment), které je multiplatformní a snadno použitelné. Umožňuje psát kód v jazyce C/C++ a nahrávat ho přímo do Arduina.
- Široká dostupnost přídatných modulů (shields): Arduino je kompatibilní s mnoha rozšiřujícími moduly nazývanými "shields", které poskytují další funkce jako WiFi, Bluetooth, senzory, pohony motorů a další.
- Rozsáhlá komunita a dokumentace: Arduino má obrovskou komunitu uživatelů a je k dispozici mnoho návodů, knih a online materiálů, které pomáhají nováčkům i pokročilým uživatelům při vytváření projektů.

Díky těmto vlastnostem se Arduino stalo populárním nástrojem pro výuku elektroniky, vývoj prototypů a realizaci různých interaktivních projektů. [1]

1.1.2 Raspberry Pi



Obrázek 2: Raspberry Pi 3 [2]

Raspberry Pi je populární jednodeskový počítač, který byl vyvinut s cílem podporovat výuku počítačových věd a informatiky ve školách a rozvíjet zájem o počítačové programování. Nicméně se rychle rozšířil i do dalších oblastí, jako je domácí automatizace, média centra, herní konzole a mnoho dalších aplikací.

Zde jsou hlavní rysy Raspberry Pi:

- **Procesor:** Raspberry Pi obsahuje ARM procesor, což mu umožňuje běžet operační systém Linux a spouštět širokou škálu aplikací.
- **Operační systém:** Raspberry Pi podporuje různé operační systémy, včetně Raspbianu (speciální verze Linuxu navržená pro Raspberry Pi), Ubuntu, Windows 10 IoT Core a další.
- **Rozšiřitelnost:** Raspberry Pi má mnoho portů pro připojení různých zařízení a rozšiřujících modulů, jako jsou HDMI pro připojení k monitoru, USB porty, GPIO (General Purpose Input/Output) piny pro připojení k elektronickým komponentům a další.
- **Komunita a podpora:** Stejně jako Arduino, i Raspberry Pi má velkou komunitu uživatelů a širokou škálu dokumentace, tutoriálů a projektů k dispozici online.

Díky své flexibilitě a nízké ceně se Raspberry Pi stal oblíbeným nástrojem pro výuku programování a elektroniky, vývoj prototypů, domácí automatizaci a mnoho dalších aplikací, které vyžadují malý a cenově dostupný počítač. [2]

1.1.3 BeagleBone



Obrázek 3: BeagleBone Black [3]

BeagleBone je další jednodeskový počítač, který je navržen pro vývoj elektronických projektů a experimentování. Podobně jako Raspberry Pi a Arduino, nabízí BeagleBone robustní vývojové prostředí a flexibilitu pro různé aplikace.

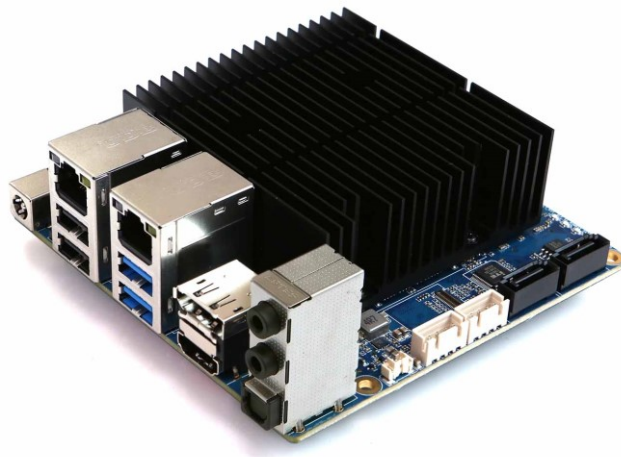
Zde jsou hlavní rysy BeagleBone:

- **Procesor:** BeagleBone je vybaven procesorem ARM, který umožňuje spouštět operační systém Linux a další aplikace.
- **Operační systém:** BeagleBone podporuje různé operační systémy, jako je například Debian, Ubuntu, Android a další.
- **Rozšiřitelnost:** BeagleBone má mnoho rozšiřujících konektorů a portů, včetně USB portů, Ethernetu, HDMI výstupu, a řady GPIO pinů, které umožňují připojení různých periférií a rozšiřujících modulů.
- **Integrace s cloudovými službami:** BeagleBone je integrován s různými cloudovými službami a platformami, což umožňuje vzdálený přístup k datům a řízení zařízení přes internet.
- **Podpora komunity:** BeagleBone má aktivní komunitu uživatelů a vývojářů, kteří sdílejí znalosti, projekty a nápady, a poskytují tak užitečnou podporu pro nové uživatele.

BeagleBone se často používá pro projekty, které vyžadují vyšší výpočetní výkon než Arduino, ale jsou méně náročné než některé aplikace, pro které se používá Raspberry Pi. Je

vhodný pro vývoj aplikací v oblastech jako jsou IoT, robotika, domácí automatizace a další. [3]

1.1.4 Odroid



Obrázek 4: ODROID-H3 [4]

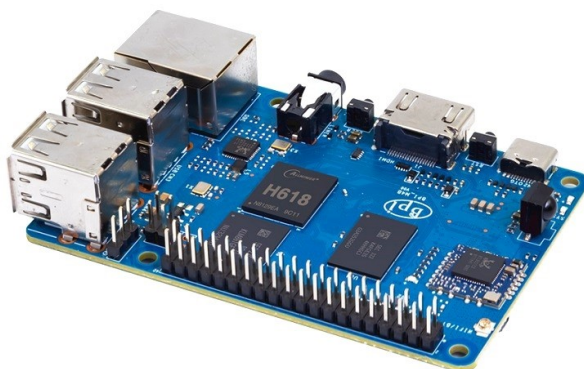
Odroid je další zajímavý jednodeskový počítač, který nabízí vysoký výkon a širokou škálu funkcí pro různé aplikace.

Zde jsou hlavní rysy Odroidu:

- **Výkonný procesor:** Odroid je vybaven výkonným procesorem založeným na architektuře ARM, což mu umožňuje provádět náročné výpočty a běžet náročné aplikace.
- **Operační systém:** Odroid podporuje různé operační systémy založené na Linuxu, včetně Ubuntu, Debianu a dalších. Taktéž je možné nainstalovat další alternativní operační systémy, včetně Androidu.
- **Rozšiřitelnost:** Odroid má bohatou sadu portů a konektorů, včetně USB portů, HDMI výstupu, Ethernetu, a dalších. Taktéž nabízí možnost rozšíření pomocí GPIO pinů a dalších rozšiřujících modulů.
- **Vysoká výkonost:** Odroid je navržen pro vysoký výkon a může být použit pro náročné aplikace, jako jsou multimediální centra, herní konzole, virtuální stroje a další.
- **Podpora komunity:** Odroid má aktivní komunitu uživatelů a vývojářů, kteří poskytují podporu, sdílejí nápady a projekty, a přispívají k rozvoji platformy.

Díky své vysoké výkonové úrovni je Odroid vhodný pro pokročilé uživatele a projekty, které vyžadují vysoký výkon a rozšiřitelnost. Může být použit pro širokou škálu aplikací, včetně multimediálních center, herních konzolí, serverů, robotiky a dalších. [4]

1.1.5 BananaPI



Obrázek 5: Banana Pi BPI-M4 Berry [5]

Banana Pi je další jednodeskový počítač, který nabízí zajímavé možnosti pro vývoj a experimentování.

Hlavní rysy Banana Pi:

- **Procesor:** Banana Pi je vybaven výkonným procesorem založeným na architektuře ARM, což mu poskytuje dostatečný výkon pro běh náročnějších aplikací.
- **Operační systém:** Banana Pi podporuje různé operační systémy založené na Linuxu, jako je například Debian, Ubuntu a další. Taktéž může být použit s alternativními operačními systémy, včetně Androidu.
- **Rozšiřitelnost:** Banana Pi má širokou škálu portů a konektorů, včetně USB portů, HDMI výstupu, Ethernetu, a dalších. Podporuje také rozšíření pomocí GPIO pinů a dalších rozšiřujících modulů.
- **Multimediální možnosti:** Banana Pi je navržen s ohledem na multimediální aplikace, což znamená, že může být využit jako multimediální centrum, streamovací zařízení nebo dokonce jako mini herní konzole.
- **Podpora komunity:** Banana Pi má také aktivní komunitu uživatelů a vývojářů, kteří sdílejí své projekty, nápady a poskytují podporu novým uživatelům.

Díky svým vlastnostem je Banana Pi vhodný pro různé aplikace, včetně vývoje prototypů, domácí automatizace, multimediálních center a dalších projektů, které vyžadují malý a cenově dostupný počítač s dostatečným výkonem. [5]

1.2 PLC

PLC (Programmable Logic Controller), česky "programovatelný logický automat", je zařízení v automatizaci průmyslových procesů, které řídí a monitoruje různé vstupy a výstupy (senzory a akční prvky) za účelem provádění určitých úloh. Jedná se o robustní počítačový systém, který je programovatelný pomocí speciálních programovacích jazyků nebo grafických programovacích prostředí.

Základními prvky PLC jsou:

- Vstupy (Inputs): Zde jsou připojeny senzory a další zařízení, která poskytují informace o stavu okolního prostředí nebo stavu procesu, jako například tlačítka, snímače teploty, snímače polohy atd.
- Výstupy (Outputs): Jsou to akční prvky, jako jsou motory, ventily, světla atd., které PLC řídí na základě informací.
- Procesor (CPU): Je srdcem PLC. Zpracovává vstupní signály podle instrukcí v programu a ovládá výstupy podle nastavených podmínek.
- Paměť: Obsahuje operační systém PLC, uživatelské programy a data potřebná pro provoz.
- Programovací rozhraní: Umožňuje inženýrům vytvářet a upravovat programy pro PLC. To může být buď prostředí s textovým programováním (např. ladder logic, structured text) nebo grafické prostředí.

PLC je široce používán v průmyslu kvůli své spolehlivosti, flexibilitě a jednoduché programovatelnosti. Je schopen řídit různé procesy, od jednoduchých úloh, jako je ovládání osvětlení, až po složité průmyslové procesy, jako je řízení výrobních linek. Používá se v různých odvětvích, včetně automobilového průmyslu, potravinářského průmyslu, energetiky, chemického průmyslu a mnoha dalších. [6]

1.3 Mikrokontroléry

Mikrokontroléry jsou speciální typy integrovaných obvodů, které kombinují procesor, paměť a periferní zařízení do jednoho čipu. Jsou navrženy tak, aby poskytovaly řízení a správu různých zařízení a systémů. Typicky se používají v embedded systémech, jako jsou domácí spotřebiče, průmyslová zařízení, automobily, lékařská zařízení, a další.

Základními rysy mikrokontrolérů jsou:

- **Procesor (CPU):** Mikrokontroléry obsahují vestavěný procesor, který provádí výpočty a řídí chod zařízení. Tyto procesory jsou obvykle optimalizovány pro nízkou spotřebu energie a specifické aplikace, ve kterých jsou používány.
- **Paměť:** Mikrokontroléry mají integrovanou paměť pro ukládání programů a dat. Tato paměť může zahrnovat různé typy, jako jsou flash paměť pro programy a EEPROM nebo RAM pro ukládání dat za běhu.
- **Periferní zařízení:** Mikrokontroléry jsou vybaveny různými periferními zařízeními, jako jsou analogové a digitální vstupy/výstupy, sériové rozhraní (UART, SPI, I2C), časovače, přerušeni, AD převodníky, PWM (pulzně šířková modulace) a mnoho dalších. Tyto periferní zařízení umožňují mikrokontroléru komunikovat s okolním světem a provádět různé úlohy.
- **Nízká spotřeba energie:** Mnoho mikrokontrolérů je navrženo s důrazem na nízkou spotřebu energie, což je důležité pro aplikace, které běží na bateriovém napájení nebo mají omezenou přístupnost k napájecí síti.
- **Integrované vývojové nástroje:** Většina mikrokontrolérů je dodávána s vývojovým prostředím, které zahrnuje kompilátor, debugger a další nástroje, které usnadňují vývoj aplikací pro daný mikrokontrolér.

Díky svým vlastnostem jsou mikrokontroléry široce využívány v mnoha aplikacích, kde je potřeba řízení a správa elektronických zařízení. Jejich nízká cena, malé rozměry a nízká spotřeba energie je činí ideální volbou pro embedded systémy různých typů. [10]

2 OVLÁDÁNÍ POHYBU

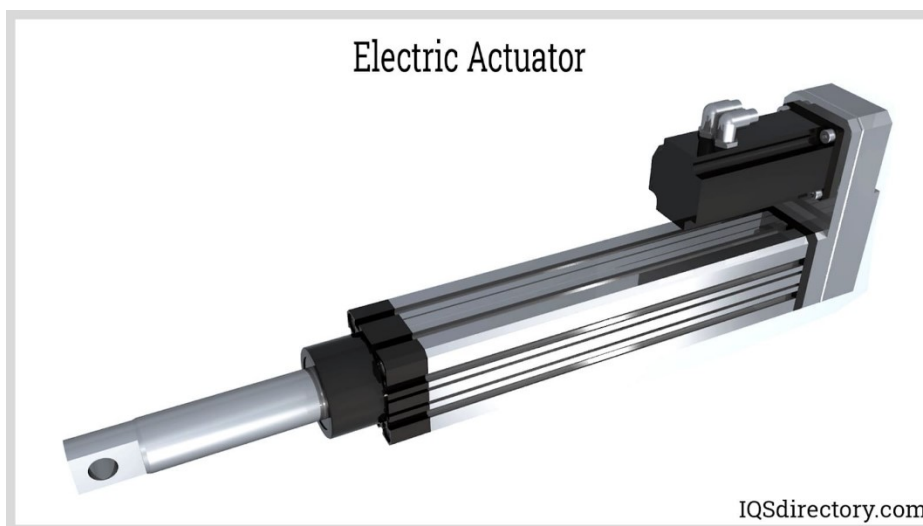
2.1 Aktuátory

Aktuátory jsou základními prvky, které umožňují pohyb v různých technických systémech. Přeměňují energii z jednoho typu na jiný, čímž vytvářejí mechanický pohyb. Tento pohyb je klíčový pro mnoho aplikací, které vyžadují manipulaci s objekty, změnu polohy nebo přenos síly.

2.1.1 Elektrické aktuátory

Elektrické aktuátory přeměňují elektrickou energii na mechanický pohyb. Mohou provádět lineární pohyb (například posouvání, zdvihání) nebo rotační pohyb (otočení kolem osy). Tyto aktuátory jsou často využívány tam, kde je důležitá přesná kontrola polohy nebo rychlosti.

Elektrické aktuátory obvykle poskytují vysokou přesnost, což je důležité pro aplikace, které vyžadují přesné polohování nebo opakovatelnost pohybu. Nicméně mohou být citlivé na vlivy jako je tření nebo zpětná vazba, což může ovlivnit jejich přesnost. [18]



Obrázek 6: Elektrický lineární aktuátor [18]

2.1.2 Pneumatické aktuátory

Pneumatické aktuátory používají stlačený vzduch k provádění mechanického pohybu. Jsou ideální pro aplikace, které vyžadují rychlý a silný pohyb, jako jsou například operace se spojovacími prvky nebo manipulace s materiály.

Pneumatické aktuátory obvykle poskytují nižší přesnost než elektrické aktuátory. Mohou být ovlivněny faktory jako je tlak vzduchu, teplotní vlivy nebo tření v pneumatickém systému, což může vést ke změně přesnosti pohybu. [19]



Obrázek 7: Pneumatický rotační aktuátor [20]

2.1.3 Hydraulické aktuátory

Hydraulické aktuátory využívají tekutiny, obvykle oleje, k provádění mechanického pohybu. Jsou schopny poskytnout velkou sílu a jsou často používány v těžkých průmyslových aplikacích, jako jsou například zdvihací mechanismy nebo hydraulické lisy.

Hydraulické aktuátory mohou poskytovat vysokou přesnost, zejména v případě použití přesně kalibrovaných ventilů a regulátorů. Nicméně mohou být citlivé na faktory jako je viskozita oleje nebo úniky tlakové tekutiny, což může ovlivnit jejich přesnost.[21]



Obrázek 8: Hydraulický lineární aktuátor [22]

2.2 Rotační motory

Rotační motory, které se řadí mezi aktuátory, představují klíčový prvek v mnoha aplikacích, které vyžadují přesný a kontrolovaný pohyb. Tyto motory přeměňují elektrickou energii na mechanický pohyb pomocí principu rotačního pohybu. V kontextu diplomové práce zaměřené na pohyb kuličky na nakloněné rovině jsou rotační motory klíčovým prvkem pro dosažení potřebného naklonění a pohybu podložky.

2.2.1 Servomotory

Servomotory jsou speciální typy rotačních motorů, které nabízejí vysokou přesnost, kontrolu a opakovatelnost pohybu. Jsou často využívány v různých aplikacích, včetně modelářství, průmyslové automatizace, robotiky a řízení pohybu. Klíčovým prvkem servomotorů je jejich schopnost udržovat určitou polohu nebo úhel, což je činí ideálními pro aplikace, které vyžadují přesný a stabilní pohyb.

Princip fungování:

Servomotory jsou obvykle vybaveny potenciometrem, enkodérem nebo jiným způsobem zpětné vazby, který jim umožňuje sledovat a udržovat svou polohu nebo úhel. Když je servomotor připojen k řídicí jednotce nebo mikrokontroléru, je možné mu poslat signál, který určuje požadovanou polohu nebo úhel. Servomotor pak automaticky přizpůsobí svůj pohyb tak, aby dosáhl této cílové polohy nebo úhlu. [14]



Obrázek 9: Servomotory [14]

Modelářská serva

V modelářství jsou servomotory často používány pro ovládání různých pohyblivých částí modelů, jako jsou řízení kormidla, pohyb křídel, podvozků nebo dalších mechanismů. Modelářská serva jsou obvykle malých rozměrů a lehkých, což je činí ideálními pro použití v modelářských aplikacích, kde je důležitá hmotnost a velikost.

Modelářská serva jsou často označována podle svých charakteristik, jako je moment, rychlost a přesnost. Tyto parametry jsou klíčové pro správný výběr servomotoru pro konkrétní aplikaci v modelářství. [15]

Servomotory nabízejí několik klíčových vlastností, které je činí atraktivními pro mnoho aplikací:

- Přesnost: Díky zpětné vazbě jsou servomotory schopny dosáhnout vysoké přesnosti polohování nebo úhlu.
- Rychlost: Mnoho servomotorů má vysokou rychlost reakce, což je užitečné pro aplikace vyžadující rychlý pohyb.

- Ovladatelnost: Servomotory lze snadno ovládat pomocí signálů z řídicí jednotky nebo mikrokontroléru, což umožňuje jednoduchou integraci do různých systémů.

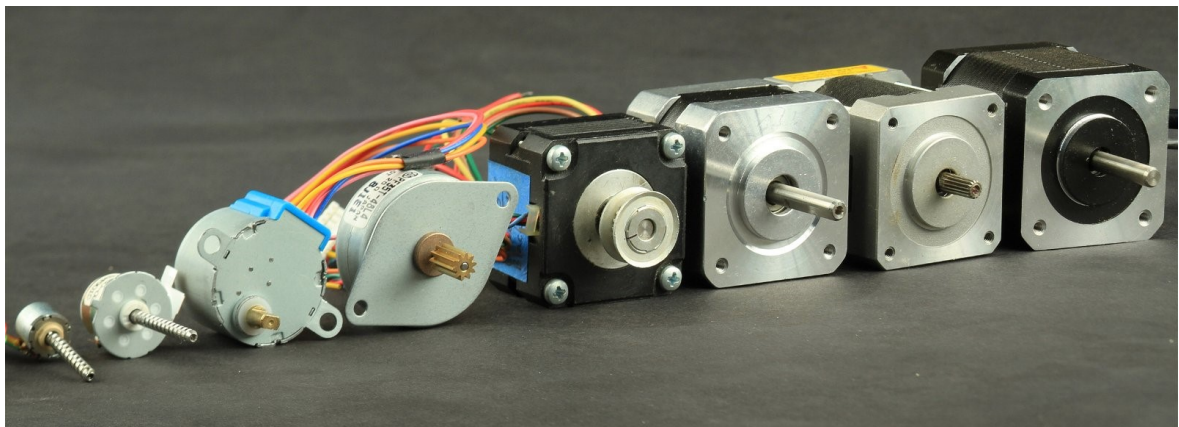
V modelářství jsou servomotory nezbytným prvkem pro řízení pohybu a chování modelů. Díky své přesnosti a rychlosti jsou modelářská serva často preferovanou volbou pro ovládání různých pohyblivých částí modelů letadel, automobilů, lodí a dalších. [15]

2.2.2 Krokové motory

Krokové motory jsou speciální typy rotačních motorů, které umožňují přesnou a kontrolovanou rotaci o určitý úhel pomocí elektrických impulsů. Jsou často využívány v aplikacích, které vyžadují vysokou přesnost a spolehlivost, jako jsou CNC stroje, tiskárny, robotické systémy a další zařízení, kde je důležitá opakovatelnost pohybu.

Krokový motor se skládá z rotoru, statoru a často také z mechanismu zpětné vazby, jako je enkodér. Rotor obsahuje permanentní magnety, zatímco stator je obklopen elektromagnetickými cívkami. Při přivedení elektrického proudu do jednotlivých cívek statoru se vytváří magnetické pole, které přitahuje permanentní magnety rotoru. Tímto způsobem se rotor posune o určitý úhel, nazývaný krok.

Krokový motor může být buď unipolární nebo bipolární v závislosti na tom, jak jsou cívky připojeny k napájecímu zdroji. Unipolární motory mají jednodušší řízení, zatímco bipolární motory poskytují vyšší výkon.



Obrázek 10: Krokové motory [16]

Existují různé typy krokových motorů, z nichž každý má své vlastní charakteristiky a aplikace:

- **Permanentní magnet (PM):** Tyto motory mají permanentní magnety na rotoru a elektromagnetické cívky na statoru. Jsou jednodušší a levnější, ale mohou trpět nedokonalostmi v krokování a momentu.
- **Variable Reluctance (VR):** Tyto motory využívají principu nejmenšího odporu k magnetickému poli. Jsou jednoduché a mají dobrý moment, ale obvykle mají nižší přesnost než PM nebo hybridní motory.
- **Hybridní:** Kombinují vlastnosti PM a VR (Variable Reluctance) motorů. Mají vylepšené vlastnosti, jako je vyšší moment a přesnost.

Vlastnosti a využití:

Krokové motory nabízejí několik klíčových vlastností, které je činí vhodnými pro mnoho aplikací:

- **Přesnost:** Krokové motory umožňují pohyb s vysokou přesností a opakovatelností, což je důležité pro aplikace vyžadující přesné polohování.
- **Ovladatelnost:** Řízení krokových motorů je relativně jednoduché pomocí posílání elektrických impulsů. To umožňuje snadnou integraci do různých systémů.
- **Moment:** Krokové motory mohou poskytovat vysoký moment, což je užitečné pro aplikace, které vyžadují pohyb velkých zátěží.

Krokové motory jsou často preferovanou volbou pro aplikace vyžadující přesný a opakovaný pohyb, jako jsou CNC stroje, 3D tiskárny, laboratorní automatizace a mnoho dalších. [16]

3 METODY DETEKCE OBJEKTŮ V OPTICKÝCH SYSTÉMECH A AUTOMATIZOVANÉ ZPRACOVÁNÍ OBRAZU

Detekce objektů je klíčovou technologií v mnoha oblastech aplikací, včetně robotiky, bezpečnostních systémů a autonomních vozidel. Tato kapitola se zaměřuje na principy a techniky detekce objektů s důrazem na optické systémy a automatizované zpracování obrazu. Cílem je poskytnout ucelený přehled o metodách detekce objektů, které využívají moderní optické systémy a algoritmy zpracování obrazu.

3.1 Základní principy optických systémů

Optické systémy jsou základem mnoha moderních technologií a aplikací, kde je kritické zachytávání a analýza obrazů. Tato sekce poskytuje přehled o základních komponentech a principy, které definují funkci optických systémů v kontextu detekce objektů.

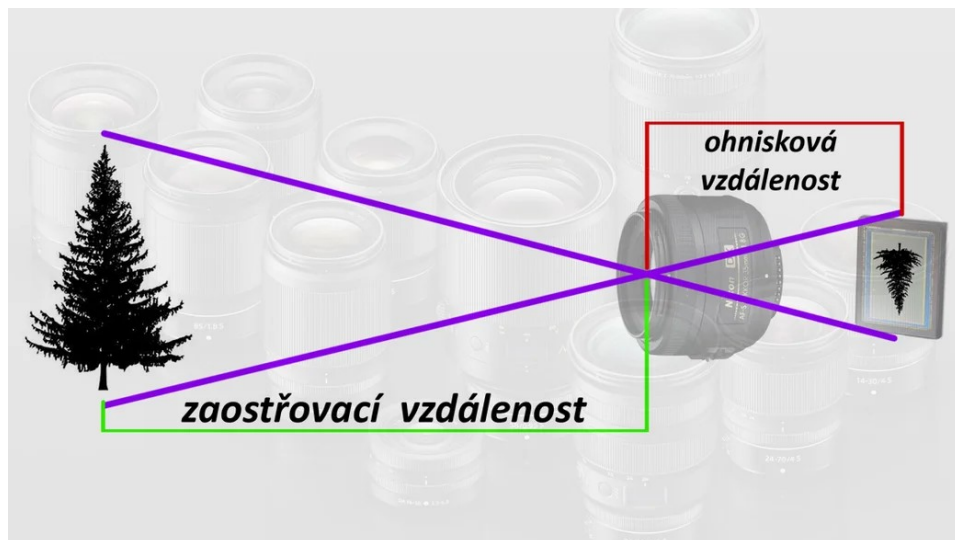
3.1.1 Typy a charakteristiky obrazových senzorů

Obrazové senzory jsou jedním z nejdůležitějších prvků optických systémů, které převádějí optický signál (světlo) na elektronický signál. Dva hlavní typy senzorů používaných v dnešních kamerách jsou CMOS a CCD:

- **CCD** (Charge-Coupled Device): Tyto senzory jsou známé svou vysokou kvalitou obrazu a schopností efektivně zachytávat světlo, což vede k lepšímu výkonu v nízkém osvětlení. CCD senzory transformují světlo přijaté na každém pixelu přímo na elektrický signál, což zajišťuje méně šumu v obraze. [23]
- **CMOS** (Complementary Metal-Oxide-Semiconductor): CMOS technologie umožňuje sensorům provádět některé zpracování signálu přímo na čipu, což umožňuje rychlejší zpracování obrazu a nižší spotřebu energie. I přesto, že kvalita obrazu může být někdy nižší ve srovnání s CCD, pokroky v technologii CMOS vedly k značnému zlepšení výkonu. [23]

3.1.2 Objektivy a optika

Objektivy jsou klíčové pro řízení toho, jaký obraz je promítán na senzor. Parametry jako ohnisková vzdálenost, clona a kvalita čoček mají přímý dopad na kvalitu a charakteristiky zachyceného obrazu. Ohnisková vzdálenost určuje zorný úhel kamery a tím i velikost zobrazené scény, zatímco clona ovlivňuje množství světla, které může procházet objektivem a dosáhnout senzoru. Toto je zvláště důležité v aplikacích, kde jsou požadavky na přesnou detekci objektů v různých světelných podmínkách. [24]



Obrázek 11: Objektiv – zaostřovací a ohnisková vzdálenost [25]

3.1.3 Zpracování signálu a obrazu

Po zachycení obrazu senzorem následuje zpracování signálu, které může zahrnovat různé úrovně filtrace, zesílení signálu a digitálního zpracování. Toto zpracování je zásadní pro transformaci surového obrazového datového toku na užitečný digitální obraz. Procesy jako demosaicing, korekce barevného posunu a redukce šumu jsou běžné a zásadní pro zajištění kvality obrazu potřebné pro další analýzu a detekci objektů.

3.1.4 Kalibrace a nastavení systému

Pro maximální efektivitu detekce objektů je klíčová správná kalibrace optických systémů, včetně kalibrace objektivů a senzorů. Kalibrace zahrnuje nastavení zaostření, expozice a dalších parametrů, aby obrazy byly co nejpřesnější a umožňovaly efektivní detekci a rozpoznávání objektů.

3.2 Metody automatizovaného zpracování obrazu

Automatizované zpracování obrazu je nezbytné pro interpretaci a analýzu obrazových dat v mnoha aplikacích. Tato kapitola se zaměřuje na klíčové kroky zpracování, od předzpracování obrazu po extrakci rysů, které jsou nezbytné pro úspěšnou detekci objektů.

3.2.1 Předzpracování obrazu

Předzpracování obrazu je prvním krokem v procesu zpracování a je zaměřeno na zlepšení kvality obrazu tak, aby následující kroky zpracování byly efektivnější a přesnější. Mezi běžné techniky předzpracování patří:

- **Normalizace intenzity:** Aby bylo zajištěno, že obrazová data jsou reprezentativní napříč různými podmínkami osvětlení, provádí se normalizace, která sjednotí rozsah intenzit pixelů.
- **Filtrace šumu:** Digitální obrazy často obsahují šum způsobený hardwarem nebo vnějšími vlivy. Aplikace filtrů, jako jsou mediánový filtr nebo Gaussovy filtry, pomáhá redukovat šum a zlepšit kvalitu obrazu.
- **Ostření:** Za účelem zvýraznění důležitých rysů obrázku se používají techniky ostření, které zlepšují viditelnost hran a dalších detailů. [26]

3.2.2 Segmentace obrazu

Segmentace obrazu je proces rozdělení obrazu na jeho konstituentní části nebo objekty. Tento krok je zásadní pro lokalizaci objektů v obraze. Mezi přístupy k segmentaci patří:

- **Prahování:** Jednoduchá, ale efektivní metoda pro segmentaci založená na intenzitě pixelů, kde pixely nad určitou hranicí intenzity jsou klasifikovány jako objekt, zatímco ostatní jako pozadí.
- **Metody založené na clusterech:** Techniky jako K-means clustering nebo segmentace pomocí algoritmu Watershed rozdělují obraz do segmentů na základě podobnosti v barvě, intenzitě nebo textuře. [26]

3.2.3 Extrakce Rysů

Po segmentaci následuje extrakce rysů, která identifikuje a izoluje užitečné informace z obrazu, které mohou být použity pro další analýzu. Mezi klíčové rysy patří:

- Hrany a obrysy: Detekce hran pomocí operátorů jako Sobel nebo Canny umožňuje identifikaci hranic objektů v obraze.
- Textura a barva: Analýza textury a barvy může pomoci rozlišit mezi různými objekty na základě jejich vizuálních charakteristik.
- Invariantní momenty: Tyto statistické metriky jsou užitečné pro popis tvaru a struktury objektů a jsou robustní proti změnám v měřítku a orientaci. [26]

3.2.3.1 Sobelův operátor

Sobelův operátor je technika numerické aproximace gradientu, která se používá k detekci hran v obrazových oblastech s vysokým gradientem intenzity. Operátor používá dvě 3x3 konvoluční masky, které jsou aplikovány na obraz pro výpočet přibližných derivací podle horizontální a vertikální osy.

- Horizontální Sobelova maska:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (1)$$

- Vertikální Sobelova maska:

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & 1 \end{bmatrix} \quad (2)$$

Obě masky se aplikují na obraz, a výsledné hodnoty z obou filtrů se kombinují pro každý pixel tak, aby se získala celková míra gradientu v daném bodě. Výsledek je pak obvykle převeden do formy, kde silné gradienty signalizují hrany. Tato metoda je efektivní v odhalování hrubých hran v obrazu, ale může být citlivá na šum.[17]

3.2.3.2 Cannyho Detektor Hran

Cannyho detektor hran je pokročilejší nástroj pro detekci hran, který poskytuje robustnější a spolehlivější výsledky. Byl vyvinut Johnem F. Canny v roce 1986 a stále je považován za jeden z nejefektivnějších detektorů hran. Cannyho algoritmus zahrnuje několik kroků:

- **Filtrace šumu:** Obraz je nejprve vyhlazen pomocí Gaussova filtru, aby se snížila citlivost na šum.
- **Výpočet gradientu:** Podobně jako u Sobelova operátoru, Canny používá filtry pro výpočet gradientu obrazu ve vertikálním a horizontálním směru.
- **Potlačení non-maxim:** V této fázi algoritmus projde gradientní magnitudou obrazu a "ztenčuje" hrany tím, že potlačí všechny pixely na hranách, které nejsou lokálním maximem.
- **Dvojitý práh a sledování hran pomocí hysterze:** Canny používá dva prahy, jeden pro zjištění silných hran a druhý pro slabé hrany. Silné hrany jsou okamžitě považovány za součást finálního hraničního obrazu, zatímco slabé hrany jsou akceptovány pouze tehdy, když jsou spojeny se silnou hranou.

Výsledný detektor hran Canny je vysoce účinný v detekci širokého rozsahu hran v obrazu a je méně náchylný k ovlivnění šumem než mnoho jiných metod detekce hran. Je široce používán v mnoha aplikacích zpracování obrazu, kde je potřeba přesná lokalizace hran a kontur objektů. [27]

3.2.4 Integrace a Aplikace

Po extrakci rysů následuje klasifikace nebo další zpracování, které využívá identifikované rysy pro rozhodování o přítomnosti a typu objektů. Tento krok může zahrnovat strojové učení nebo hluboké učení, což umožňuje systémům učit se z příkladů a zlepšovat svou schopnost detekce objektů.

3.3 Algoritmy Detekce Objektů

Algoritmy detekce objektů jsou klíčové pro mnoho aplikací, od bezpečnostních kamer po autonomní vozidla. Existují různé metody a techniky, které lze použít pro identifikaci a lokalizaci objektů v digitálních obrázcích.

3.3.1 Tradiční Metody Detekce

Tradiční metody zahrnují techniky, které se spoléhají na ručně navržené rysy a algoritmické přístupy bez využití učení.

3.3.1.1 Hraniční a Konturové Detektory

Hraniční a konturové detektory jsou základní techniky používané pro identifikaci hranic objektů v obrázcích. Hrany v obrazu obvykle odpovídají náhlým změnám v intenzitě pixelů, které často signalizují důležité události a změny v obrazových vlastnostech. Tyto techniky jsou klíčové pro mnoho aplikací, včetně segmentace obrazu, rozpoznávání objektů a počítačového vidění obecně.

Detekce Hran

Detekce hran je proces identifikace míst v obrazu, kde dochází k výrazné změně jasu nebo barvy. Tyto změny obvykle indikují hranice mezi různými objekty nebo mezi objektem a pozadím. Zde jsou některé z nejpoužívanějších operátorů pro detekci hran:

- **Sobelův operátor:** Jak bylo zmíněno dříve, Sobelův operátor detekuje hrany pomocí dvou konvolučních masek (horizontální a vertikální), které aproximují první derivaci obrazu ve vertikálním a horizontálním směru.
- **Prewittův operátor:** Podobný Sobelovu operátoru, Prewittův operátor používá jiné konvoluční masky pro výpočet gradientů. Má tendenci být méně citlivý na šum než Sobelův operátor.
- **Operátor Robertsův kříž:** Tento operátor používá diagonální masky pro výpočet gradientu a je zvláště užitečný pro detekci diagonálních hran v obrazu.
- **Cannyho detektor hran:** Tento algoritmus je považován za jeden z nejefektivnějších detektorů hran, protože kombinuje několik kroků zpracování k maximalizaci detekce a minimalizaci chyb. Využívá dvojitého prahování a potlačení non-maximů pro čistější a přesnější výsledky. [28]

Detekce Kontur

Detekce kontur je pokročilý proces, který nejen identifikuje hrany, ale také sleduje hraniční linie objektů v obrazu. Detekce kontur je obvykle založena na metodách, které rozpoznávají spojitost hraničních bodů. Příklady zahrnují:

- **Sledování kontur:** Jedná se o techniku, která začíná na hraničním bodě a pokračuje v sledování celé kontury objektu, dokud se nevrátí zpět k výchozímu bodu.
- **Algoritmus Suzuki85:** Tento algoritmus je populární pro nalezení kontur v binárních obrazích a je implementován v několika knihovnách počítačového vidění, jako je OpenCV. Pomáhá identifikovat vnitřní a vnější kontury objektů. [30]

3.3.1.2 Sliding Window (Klouzavé okno)

Metoda klouzavého okna zahrnuje posunování malého okna přes celý obraz, krok po kroku, v každém kroku analyzující obsah okna k určení, zda obsahuje objekt zájmu. Tento přístup může být použit s různými klasifikátory, které rozhodují, zda je v aktuálním okně přítomen hledaný objekt.

- **Volba velikosti okna:** Velikost okna by měla být zvolena na základě očekávané velikosti objektu. Často je potřeba použít okna různých velikostí pro zachycení objektů různých rozměrů.
- **Krokování okna:** Okno se posune po obrazu v obou horizontálních a vertikálních směrech. Velikost kroku (posun mezi jednotlivými polohami okna) může ovlivnit jak přesnost detekce, tak výpočetní náročnost. Menší kroky zvyšují šanci na detekci objektů, ale také značně zvyšují počet vyhodnocení a tím celkový výpočetní čas.
- **Klasifikace:** V každém kroku se použije klasifikátor (například podpůrné vektory, neuronové sítě, nebo rozhodovací stromy) k určení, zda okno obsahuje objekt. Klasifikátor je trénován na základě trénovacích dat obsahujících příklady objektů a pozadí.
- **Výběr Okna:** Pro každé okno, ve kterém klasifikátor identifikuje objekt, se zaznamená jeho poloha. V případě, že se několik okenních detekcí překrývá, může být potřeba další logika pro zjednodušení a výběr nejlepší detekce.

Výhody metody:

- Jednoduchost implementace.
- Flexibilita v použití s různými klasifikátory.
- Efektivní pro obrazy s omezeným počtem objektů a relativně jednoduchými pozadími.

Nevýhody metody:

- Výpočetně náročné, zejména při vysokém rozlišení obrazů a malých krocích.
- Může být méně efektivní pro detekci objektů, které jsou výrazně různých velikostí, pokud se nepoužívají více škálové okna.
- Vysoký počet falešných pozitiv, pokud klasifikátor není dostatečně robustní.

Metoda klouzavého okna byla populárním přístupem k detekci objektů, dokud nebyla ve větší míře nahrazena modernějšími technikami založenými na hlubokém učení, které dokáží lépe zvládat výzvy spojené s různorodostí a komplexností reálných scén. Nicméně, stále se nachází aplikace, kde může být tento přístup užitečný. [29]

3.3.1.3 *Histogram Orientovaných Gradientů (HOG)*

Histogram orientovaných gradientů (HOG) je technika extrakce rysů, která byla původně navržena pro účely detekce chodců v obrazových datech. Metoda HOG je založena na počítání a statistickém vyhodnocení směrů gradientů jasu obrázku, které popisují lokální struktury obrazu. Tato metoda je obzvláště účinná pro úlohy, kde je třeba rozpoznat objekty nebo jejich části na základě jejich tvaru a vzhledu.

HOG extrahuje rysy z obrazu tím, že sleduje distribuci směrů gradientů (změn intenzity) a velikosti gradientů v lokálních oblastech obrázku. Základní kroky extrakce HOG rysů jsou následující:

1. **Normalizace jasu:** Obraz je nejprve normalizován pro snížení vlivu změn osvětlení na výpočet gradientů.
2. **Výpočet gradientů:** Pro každý pixel v obrázku se vypočítají gradienty jasu v horizontálním a vertikálním směru, obvykle pomocí jednoduchých diferencovacích masek, jako jsou $[1, 0, -1]$ a její transpozice.
3. **Vytvoření buněk:** Obraz je rozdělen do malých prostorově propojených oblastí zvaných buňky (například 8×8 pixelů), uvnitř kterých se vypočítají histogramy gradientů.

4. **Histogram gradientů:** Pro každou buňku se vypočítá histogram gradientních orientací. Běžně se používá 9 až 18 intervalů (binů), které pokrývají 0 až 180 stupňů (nebo 0 až 360 stupňů pro neorientované gradienty).
5. **Normalizace bloků:** Aby byly rysy robustnější vůči změnám osvětlení a stínům, buňky jsou sdruženy do větších jednotek zvaných bloky. Histogramy v bloku jsou normalizovány podle nějaké normy (např. L2-norma), což zlepšuje invarianci vůči osvětlení a stínování.
6. **Sestavení deskriptoru:** Konečný HOG deskriptor je sestaven spojením normalizovaných histogramů všech bloků obrázku. Tento deskriptor poskytuje komplexní a kompaktní reprezentaci tvaru objektů v obrázku.

HOG se široce používá pro detekci a rozpoznávání objektů v různých aplikacích počítačového vidění, včetně:

- **Detekce chodců:** HOG byl původně navržen pro identifikaci chodců v urbanistických scénách a stále se používá jako robustní rys pro tuto aplikaci.
- **Rozpoznání vozidel:** Stejně jako chodci, i vozidla mají charakteristické obrysy, které HOG efektivně zachycuje.
- **Biometrické systémy:** Tvar a obrysy lidských obličejů mohou být také extrahovány pomocí HOG, což je užitečné pro účely rozpoznávání tváří.
- **Robotika a autonomní systémy:** HOG pomáhá autonomním systémům v rozpoznávání objektů a navigaci.

Histogram orientovaných gradientů (HOG) je silný a efektivní nástroj pro extrakci rysů z obrazu, který umožňuje detekci a rozpoznání objektů na základě jejich tvaru. Jeho robustnost vůči změnám v osvětlení a schopnost zachytit důležité morfologické charakteristiky činí HOG populární volbou v mnoha aplikacích počítačového vidění. [29]

3.3.2 Učení s Učitelem

Metody učení s učitelem se spoléhají na anotovaná trénovací data k "naučení" modelu, jak rozpoznat objekty.

3.3.2.1 *Support Vector Machines (SVM)*

Support Vector Machines (SVM) je dozorovaný strojový učící model, který je široce využíván pro klasifikační a regresní úlohy. SVM je zvláště efektivní v případech, kde je dimenze vstupních dat vysoká, a je dobře znám pro svou schopnost modelovat složité hranice rozhodování s relativně malým množstvím trénovacích dat.

Základní princip SVM spočívá ve vytváření hyperrovin (nebo rozhodovacích hranic), které efektivně oddělují různé třídy dat v prostoru vstupních proměnných. Cílem je najít hyperrovinu, která maximálně zvětšuje okraj mezi různými třídami, což zvyšuje odolnost klasifikátoru vůči šumu a přesnosti predikce.

Proces SVM

- **Maximalizace okraje:** SVM hledá hyperrovinu, která maximalizuje okraj mezi nejbližšími datovými body různých tříd, známými jako podpůrné vektory. Okraj je definován jako vzdálenost mezi hyperrovinou a nejbližšími body na každé straně této hranice.
- **Kernel Trick:** Pro data, která nejsou lineárně separovatelná ve svém původním prostoru, SVM používá tzv. jádrové funkce. Tyto funkce transformují data do vyššího dimenzionálního prostoru, kde mohou být lineárně separovatelná, aniž by bylo nutné explicitně provádět tuto transformaci, což šetří výpočetní zdroje.
- **Řešení optimalizačního problému:** Algoritmus řeší optimalizační problém, který najde hyperrovinu s maximálním okrajem. Tento proces zahrnuje kvadratické programování, které může být řešeno pomocí různých numerických metod.

Typy Jádrových Funkcí

- **Lineární jádro:** Nejjednodušší jádro, používané, když data jsou lineárně separovatelná.
- **Polynomiální jádro:** Umožňuje modelování ne-lineárních hranic. Stupeň polynomu určuje složitost modelu.
- **Radiální bázová funkce (RBF) nebo Gausiánské jádro:** Jedno z nejpopulárnějších a nejuniverzálnějších jader, které dobře funguje v mnoha praktických případech.

- **Sigmoidní jádro:** Převádí vstupní vektory na hyperbolické tangenty jejich předobrazů.

SVM se používá v široké škále aplikací, od textové klasifikace a rozpoznávání obrazů po bioinformatické klasifikace a mnoho dalších oblastí. Díky své robustnosti a efektivitě při zpracování vysoce dimenzionálních dat je SVM jedním z nejoblíbenějších modelů v machine learningu.

SVM jsou klíčovou komponentou strojového učení, poskytující mocný nástroj pro klasifikaci a regresi. Díky jejich flexibilitě v použití různých jádrových funkcí a schopnosti zvládnout velké dimenze dat jsou extrémně užitečné pro širokou škálu problémů v reálném světě. [31]

3.3.2.2 *Random Forests*

Random Forests je souborová metoda strojového učení, která využívá množství rozhodovacích stromů k provedení predikcí. Vytváření mnoha stromů a jejich kombinace zvyšuje přesnost a robustnost modelu ve srovnání s jednotlivým rozhodovacím stromem, přičemž se snižuje riziko přeučení (overfitting).

Random Forests funguje na základě následujících principů:

- **Bagging (Bootstrap Aggregating):** Každý strom v lese je trénován na náhodně vygenerované podmnožině trénovacích dat s opakováním, což znamená, že některé instance se mohou v trénovacím setu každého stromu objevit vícekrát, zatímco jiné mohou být vynechány.
- **Výběr náhodných rysů:** Při každém rozdělení v trénovacím procesu každého stromu se náhodně vybírá podmnožina dostupných rysů (featureů). Toto zajišťuje, že jednotlivé stromy jsou rozdílné a že jsou schopny lépe generalizovat, protože nejsou závislé na konkrétních rysech.
- **Agregace predikcí:** Konečná predikce Random Forests modelu je získána agregací predikcí od všech stromů v lese. Pro klasifikační úlohy se obvykle používá hlasování většiny, zatímco pro regresi se výsledky jednotlivých stromů průměrují.

Výhody Random Forests

- **Vysoká přesnost:** Random Forests obvykle poskytuje vysokou prediktivní přesnost díky kombinaci predikcí z mnoha stromů.

- **Odolnost vůči přeučení:** Díky náhodnosti ve výběru dat a rysů mají stromy menší tendenci k přeučení než jednotlivé rozhodovací stromy.
- **Flexibilita:** Algoritmus lze použít jak pro klasifikační, tak pro regresní úlohy.
- **Důležitost proměnných:** Random Forests poskytuje odhady důležitosti jednotlivých rysů, což může být užitečné pro pochopení dat.

Nevýhody Random Forests

- **Výpočetně náročnější:** Trénování mnoha stromů vyžaduje více výpočetních zdrojů, což může být náročné pro velmi velké datové sady.
- **Méně interpretabilní:** Na rozdíl od jednoho rozhodovacího stromu, který lze snadno vizualizovat a interpretovat, je les stromů jako celek těžší pochopit a vysvětlit.
- **Pomalá predikce v reálném čase:** Kvůli potřebě provést predikce z mnoha stromů může být doba potřebná pro predikci v reálném čase delší než u jiných modelů.

Random Forests se používá v široké škále oblastí, jako jsou například na biomedicínský výzkum, finanční analýzy, strojové vidění a mnoho dalších. Jeho schopnost zvládat velké a složité datové sady s vysokou přesností a robustností činí z Random Forests jednoho z nejoblíbenějších algoritmů strojového učení. [32]

3.3.3 Hluboké Učení

Metody hlubokého učení se staly de facto standardem pro detekci objektů díky jejich výkonnosti a schopnosti učit se složitým vzorcům z dat.

3.3.3.1 Konvoluční neuronové sítě (CNN)

Konvoluční neuronové sítě (CNN) jsou jedním z klíčových nástrojů v moderním počítačovém vidění a byly úspěšně aplikovány na širokou škálu úloh, od rozpoznávání obrazu a klasifikace až po pokročilé aplikace v analýze videa a přirozeném zpracování jazyka. CNN vynikají v zachycování prostorových a časových závislostí v datech díky použití relevantních filtrů.

Architektura CNN je navržena tak, aby automaticky a efektivně zpracovávala datové matice, jako jsou obrazy. Typická architektura CNN obsahuje několik vrstev, které společně pracují na extrakci a transformaci rysů ze vstupních dat. Pořadí těchto vrstev má zásadní význam pro úspěšné zpracování vstupů a výsledné predikce modelu:

1. **Konvoluční vrstvy:** Tyto vrstvy používají sadu učitelných filtrů, které automaticky detekují různé rysy ve vstupních datech. Každý filtr generuje aktivační mapu, která ukazuje, kde byly určité rysy detekovány ve vstupních datech. Vrstvy se obvykle střídají s aktivačními funkcemi, jako je ReLU, pro zavedení nelinearity.
2. **ReLU vrstvy (Rectified Linear Unit):** ReLU je nejčastěji používaná aktivační funkce v CNN, která přidává nelinearitu do modelu tím, že aplikuje elementární nelineární transformaci, jako je $\max(0, x)$.
3. **Pooling vrstvy (Subsampling):** Tyto vrstvy redukuje rozměry aktivačních map z předchozích konvolučních vrstev, čímž zvyšují odolnost sítě vůči drobným posunům a deformacím ve vstupních datech. Nejběžnější typ pooling vrstvy je max pooling.
4. **Plně propojené vrstvy (Fully Connected Layers):** Po sérii konvolučních a pooling vrstev následují jedna nebo více plně propojených vrstev, které slouží k integraci rysů do konečných predikcí. Výstup z těchto vrstev je často zpracován softmax funkcí pro klasifikaci.

Každá vrstva v CNN má specifickou roli v procesu učení a transformace dat. Vrstvy jsou uspořádány tak, aby postupně zvyšovaly abstrakci a redukovaly prostorovou dimenzi dat, což umožňuje efektivní a smysluplnou interpretaci vstupních dat pro klasifikační a detekční účely. Toto pořadí vrstev je klíčem k efektivnímu naučení modelu a dosažení dobrých výsledků na složitých úlohách zpracování dat.

Trénování CNN zahrnuje optimalizaci vah filtrů tak, aby byly minimalizovány chyby v predikcích sítě. To se obvykle děje pomocí metody zpětného šíření chyb (backpropagation) a gradientního sestupu (nebo jeho variant), které postupně upravují váhy v závislosti na chybě predikce.

Aplikace CNN:

- **Rozpoznávání obrazu a klasifikace:** CNN jsou základem pro mnoho systémů pro automatické rozpoznávání obrazu a video analýzu.
- **Detekce objektů:** Systémy jako YOLO (You Only Look Once) a SSD (Single Shot MultiBox Detector) využívají CNN k detekci objektů v reálném čase.
- **Segmentace obrazu:** CNN mohou být také použity pro segmentaci obrazu, což je proces rozdělení obrazu na několik segmentů, které reprezentují různé objekty.
- **Analýza videa:** CNN umožňují sledování objektů a rozpoznávání činností ve videu, což je užitečné v bezpečnostních aplikacích a pro interaktivní hry.

Přes své úspěchy čelí konvoluční neuronové sítě (CNN) výzvám, jako jsou vysoké požadavky na výpočetní zdroje a úložiště, zejména při zpracování velkých datových sad nebo velmi hlubokých sítí. Běžnou praxí je využívání předtrénovaných modelů, které se doplní o několik specifických vrstev adaptovaných na konkrétní úkoly, což může výrazně zredukovat náročnost a čas potřebný pro trénování nových modelů. Budoucí výzkum se zaměřuje na vytváření efektivnějších architektur, redukci požadavků na náročnost trénování a lepší generalizaci modelů mimo trénovací sady. [33]

3.3.3.2 R-CNN (Regions with Convolutional Neural Network Features)

R-CNN (Regions with Convolutional Neural Network features) představuje průlomový přístup v oblasti detekce objektů, který kombinuje regionální návrhy s výkonem konvolučních neuronových sítí. Tato metoda byla poprvé představena v roce 2014 a značně zlepšila přesnost detekce objektů oproti tehdy dostupným technikám.

R-CNN funguje ve dvou hlavních fázích: návrhu regionů a extrakci rysů pomocí CNN. Tento dvoustupňový proces umožňuje efektivní a přesné lokalizování a klasifikaci objektů:

- **Generování Regionálních Návrhů:** Prvním krokem v R-CNN je identifikace potenciálních objektů v obrázku. To se dosahuje pomocí algoritmu, jako je Selective Search, který systematicky spojuje přilehlé superpixely na základě různých kritérií (barva, textura, velikost, atd.) a generuje tisíce regionálních návrhů, které mohou obsahovat objekty.
- **Extrakce Rysů pomocí CNN:** Každý navržený region je následně zpracován konvoluční neuronovou sítí, která extrahuje rysy pro klasifikaci. Regiony jsou převedeny na pevnou velikost (například 227x227 pixelů), aby byly kompatibilní s CNN. Extrahované rysy jsou pak využity klasifikátorem, obvykle SVM (Support Vector Machine), pro určení, zda a jaký objekt region obsahuje.

Výhody R-CNN:

- **Vysoká Přesnost:** R-CNN značně zlepšuje přesnost detekce díky efektivní kombinaci regionálních návrhů a hlubokého učení.
- **Flexibilita:** R-CNN může být trénována na různých datasetech a snadno přizpůsobena novým úlohám detekce objektů.

Nevýhody R-CNN:

- **Výpočetní Náročnost:** R-CNN je poměrně pomalá, protože vyžaduje zpracování tisíců regionálních návrhů pro jediný obrázek, což zahrnuje mnohonásobné aplikace CNN.
- **Skladování Rysů:** R-CNN vyžaduje uložení extrahovaných rysů pro každý návrh regionu, což může vyžadovat značné množství paměti.

Po uvedení R-CNN bylo vyvinuto několik variant a vylepšení, které se snaží překonat jeho omezení, zejména co se týče rychlosti a efektivity:

- **Fast R-CNN:** Tato varianta zlepšuje efektivitu tím, že používá celý obrázek pro extrakci rysů jediným průchodem přes CNN před navrhováním regionů, čímž se sníží redundance v extrakci rysů.
- **Faster R-CNN:** Další zlepšení, Faster R-CNN, integruje proces generování regionálních návrhů přímo do CNN pomocí Region Proposal Network (RPN), což výrazně zrychluje proces detekce.

R-CNN a jeho vylepšení poskytují mocný framework pro detekci objektů využívající hluboké učení. Přestože původní R-CNN má určité nevýhody v rychlosti a výpočetní efektivitě, jeho pokračovatelé, jako Fast a Faster R-CNN, nabízejí řešení, která jsou praktičtější pro reálné aplikace. R-CNN tak nadále zůstává základním kamenem v oblasti počítačového vidění a má zásadní vliv na vývoj technik detekce objektů. [34]

3.3.3.3 YOLO

YOLO (You Only Look Once) je moderní systém detekce objektů, který je významný pro svou rychlost a efektivitu. Na rozdíl od tradičních metod detekce objektů, které analyzují několik částí obrázku několikrát, YOLO detekuje objekty v obrázku v jediném průchodu analýzy, což umožňuje jeho použití v reálném čase a v aplikacích vyžadujících rychlé zpracování.

YOLO zpracovává celý obrázek a předpovídá ohraničující rámečky a pravděpodobnosti tříd pro tyto rámečky. Tento přístup zahrnuje několik klíčových kroků:

- **Síťová architektura:** YOLO využívá jedinou konvoluční neuronovou síť (CNN), která přímo predikuje více ohraničujících rámečků a pravděpodobnostní skóre tříd pro každý rámeček simultánně.

- **Predikce:** Vstupní obraz je rozdělen na SxS mřížku. Pro každou buňku mřížky YOLO predikuje B ohraničujících rámečků a pravděpodobnosti C tříd. Každý rámeček má přidruženou pravděpodobnost, že obsahuje objekt a offsety polohy.
- **Trénování:** Model je trénován na celých obrázcích s přímou optimalizací detekčního výkonu. YOLO používá funkci ztráty, která penalizuje chyby v předpovědi rámečku a klasifikační chyby.

Výhody YOLO:

- **Rychlost:** YOLO je extrémně rychlý, což ho činí ideálním pro aplikace v reálném čase, jako je sledování vozidel, robotika nebo bezpečnostní systémy.
- **Přesnost:** Přestože je rychlejší než mnoho tradičních metod, YOLO dosahuje srovnatelné nebo lepší přesnosti v mnoha běžných datasetech a benchmarkových testech.
- **Generalizace:** YOLO je schopen lépe generalizovat na nové nebo neznámé objekty než jiné systémy, které se zaměřují pouze na specifické regiony.

Nevýhody YOLO:

- **Chyby lokalizace:** Kvůli svému jednoduchému přístupu může mít YOLO potíže s přesnou lokalizací některých objektů, zejména malých.
- **Citlivost na velikost objektu:** V některých případech může YOLO přehlédnout velmi malé objekty nebo objekty, které se nevejdou dobře do mřížkové struktury.

Od svého původního vydání bylo vyvinuto několik verzí YOLO, včetně YOLOv2 a YOLOv3, které postupně zlepšují přesnost, rychlost a schopnost detekovat malé objekty. Každá nová verze přináší inovace v architektuře sítě a trénovacích technikách.

YOLO je významným milníkem v oblasti detekce objektů a jeho vývoj nadále formuje budoucnost aplikací počítačového vidění. Jeho schopnost provádět rychlé a efektivní detekce v reálném čase otevírá nové možnosti pro mnoho technologických aplikací. [35]

4 PID REGULÁTOR

PID regulátor, neboli regulátor s proporcionální, integrační a derivační složkou, je jedním z nejčastěji používaných typů zpětnovazebních regulátorů v průmyslové automatizaci a kontrolních systémech. Základní účel PID regulátoru je udržet výstupní veličinu systému (například teplotu, rychlost, tlak apod.) co nejbližší předem definované cílové hodnotě, tzv. referenční nebo žádané hodnotě.

PID regulátor skládá z tří složek:

1. **Proporcionální složka (P)** - Reaguje na velikost aktuální chyby, která je rozdílem mezi aktuální hodnotou a referenční hodnotou. Proporcionální složka zvyšuje nebo snižuje výstupní hodnotu regulátoru přímo úměrně k velikosti chyby.
2. **Integrační složka (I)** - Akumuluje chyby, které se vyskytly v čase, a integruje je do výstupního signálu regulátoru. Tato složka pomáhá eliminovat trvalé chyby, které nemohou být vyrovnány pouze proporcionální složkou, protože se snaží eliminovat i malé, ale trvalé rozdíly mezi aktuální a referenční hodnotou.
3. **Derivační složka (D)** - Reaguje na rychlost změn chyby, což znamená, že hodnotí, jak rychle se chyba mění. Derivační složka pomáhá anticipovat budoucí chyby a zasahuje předem, což zlepšuje stabilitu a rychlost odezvy systému.

Funkce PID regulátoru:

Vstupní signál do PID regulátoru je chyba (e), a výstupní signál je akční zásah (u), který se snaží chybu minimalizovat. Výstup PID regulátoru je kombinací všech tří složek a vypočítává se podle následující rovnice:

$$u(t) = K_P \cdot e(t) + K_I \cdot \int e(t)dt + K_D \cdot \frac{de(t)}{dt} \quad (3)$$

kde K_P , K_I a K_D jsou konstanty určující váhu každé složky regulátoru. Tyto konstanty se musí pečlivě naladit podle konkrétních potřeb a dynamiky daného systému. [36]

II. PRAKTICKÁ ČÁST

5 VÝBĚR NAKLÁPĚCÍ ROVINY

5.1 Komerčně dostupné manuálně řízené naklápěcí roviny

Všechny komerčně dostupné platformy, které by potenciálně mohly odpovídat požadavkům stanoveným v této diplomové práci, vykazují podobné charakteristiky a představují řadu faktorů, které omezují jejich využití. Mechanismus naklápění platformy je realizován pomocí navíjení provázku na hřídel, která je spojena s bočními kolečky, čímž je do systému vnášena značná míra vůle. Tato konstrukce vede k nestabilitě platformy při rychlém pohybu. Způsob uložení platformy je navržen s využitím ocelových kolíků vložených do plastových pouzder, což rovněž vytváří významnou vůli. Dále je uložení hřídelek s bočními kolečky provedeno přímo ve dřevěné konstrukci.

5.1.1 Woody Labyrint

Výrobce Woody uvedl na trh Woody Labyrint, což je bludiště, které lze pomocí postranních koleček naklápět ve dvou směrech. [36]



Obrázek 12: Woody Labyrint [37]

5.1.2 Labyrint hra Brio 34000 dřevěné

Dalším výrobcem obdobné hry je Brio



Obrázek 13: Labyrint hra Brio 34000 dřevěné [38]

5.2 Vlastní řešení

Nedostatky komerčně dostupných řešení, bohužel, překonávají jejich potenciální přínosy a značně omezují možnost jejich adaptace pro využití v mechatronickém systému. Vzhledem k tomuto zjištění jsem se rozhodl pro vývoj vlastní naklápěcí platformy, která je sice inspirována dostupnými komerčními modely, avšak je navržena s ohledem na specifické potřeby tohoto projektu.

5.2.1 Konstrukce zařízení

Konstrukční koncept naklápěcí roviny je inspirován principy kinematiky delta, jak jsou aplikovány v oblasti robotických manipulátorů. Esence tohoto přístupu spočívá v umístění tro-

jice motorů na stabilní podložku, které jsou propojeny s naklápěcí platformou prostřednictvím ramen. Tato konfigurace zajišťuje efektivní přenos síly a poskytuje vysokou míru přesnosti a adaptability v pohybu naklápěcí roviny.



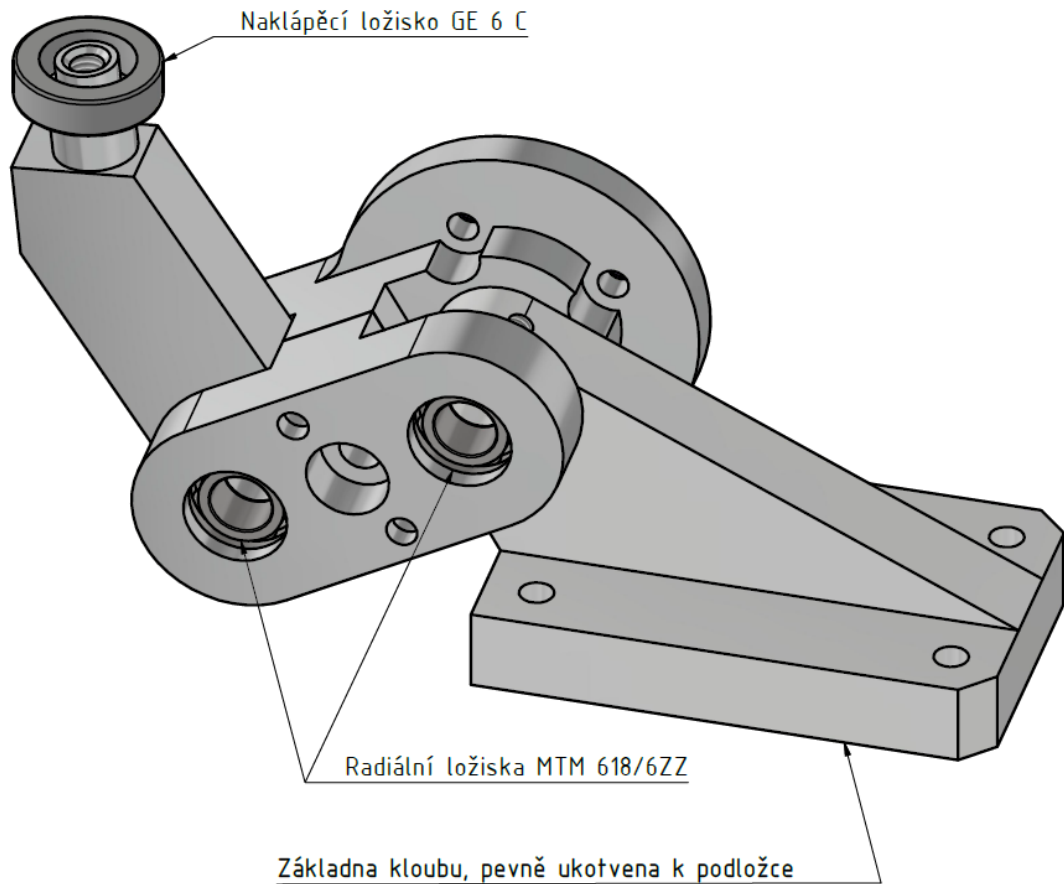
Obrázek 14: Fotografie sestrojeného zařízení

Podpora naklápěcí roviny je zajištěna třemi motorizovanými rameny, kde každé z nich je poháněno RC servomotorem HBL550 [11]. Tyto motory byly zvoleny pro svou schopnost poskytovat dynamický pohyb s vysokou rychlostí a točivým momentem. Ramena jsou konstruována z hliníkové slitiny EN AW 6082 [9], materiálu, který byl specificky vybrán pro jeho vynikající tuhost a schopnost odolávat nechtěnému prohnutí při pohybu v jiných osách. Kromě toho, tato slitina je výrazně lehčí oproti tradičním kovům, což přispívá ke zvýšení celkové efektivity a snížení zátěže na pohybový systém.

Spojení mezi jednotlivými klouby ramen je řešeno pomocí dvou kuličkových ložisek MTM 618/6ZZ [8], což minimalizuje odpor a vůle v kloubu a umožňuje plynulý a přesný pohyb. Toto řešení je fundamentální pro zachování vysoké efektivity a přesnosti pohybu.

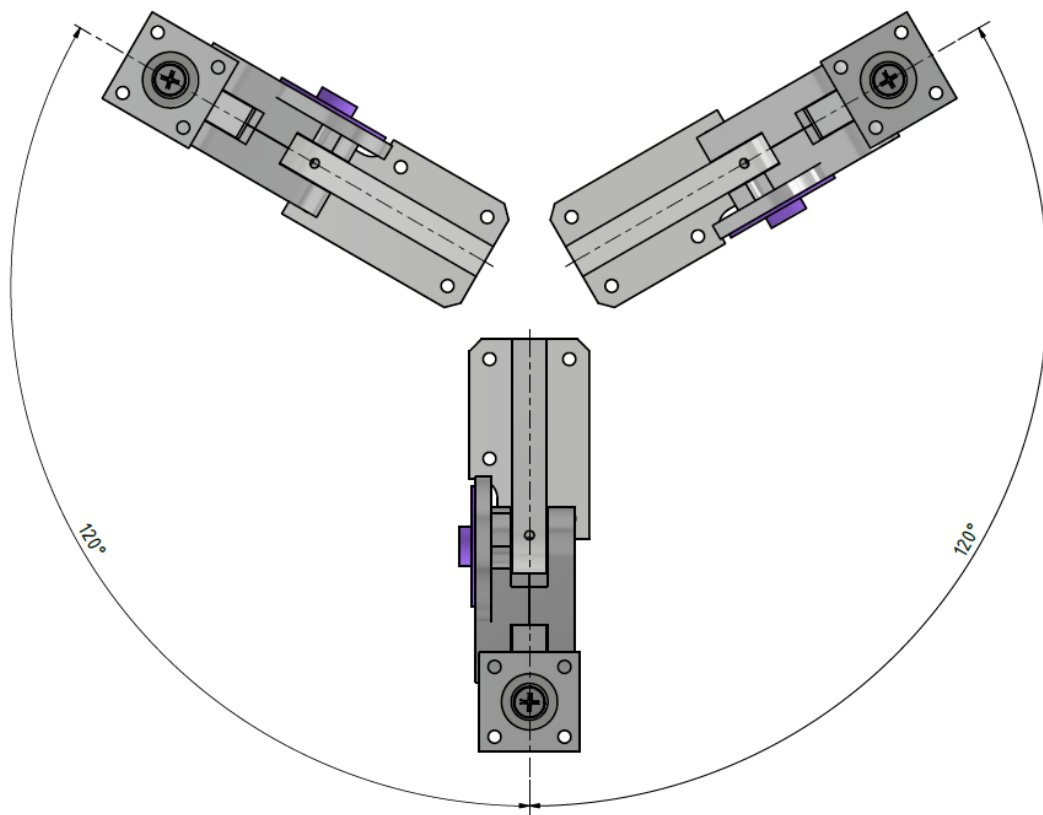
Pro spojení roviny s rameny je využito kloubového ložiska GE 6 C [7], které umožňuje vyklonění až o 13° od osy. Tato flexibilita je nezbytná pro přesné nastavení úhlu náklonu platformy.

K naklápěcímu ložisku je dále ukotvena naklápěcí rovina



Obrázek 15: Schéma ramene

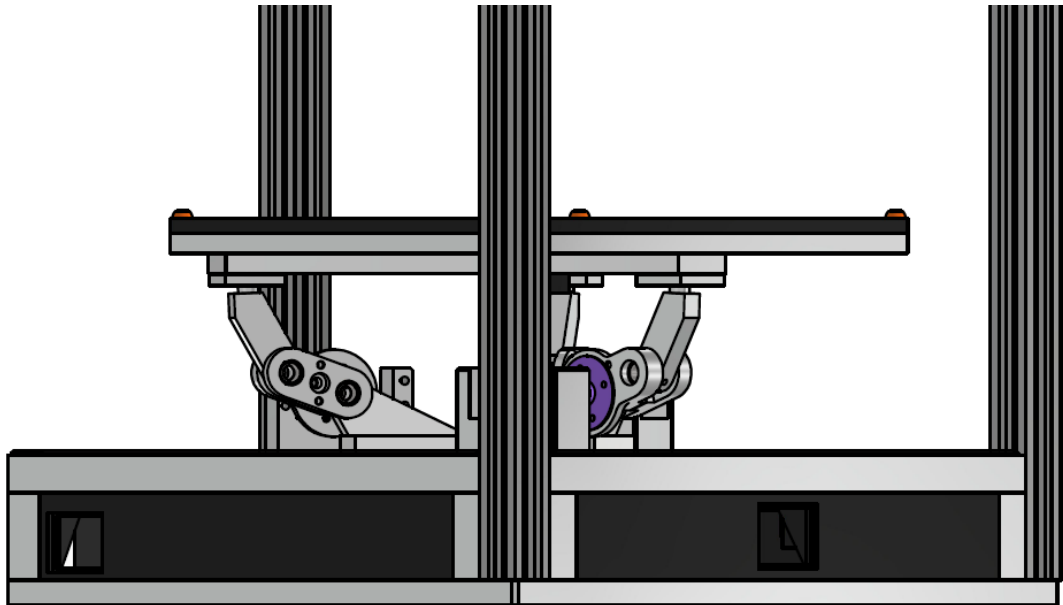
Pro hlubší pochopení geometrické konfigurace mechanismu je klíčové zdůraznit, že ramena jsou vůči sobě vzájemně orientována pod úhlem 120° . Tato precizní úhlová dispozice je nezbytná pro zajištění rovnoměrné distribuce síly a stabilizaci naklápěcí platformy. Toto uspořádání umožňuje, aby každé z ramen přenášelo zátěž a pohybovalo se synchronizovaně s ostatními, čímž se dosahuje optimální rovnováhy a efektivity v pohybových operacích. Specifické úhlové rozmístění je navrženo tak, aby maximálně využívalo principy kinematiky delta, které jsou základem celé konstrukce.



Obrázek 16: Umístění ramen

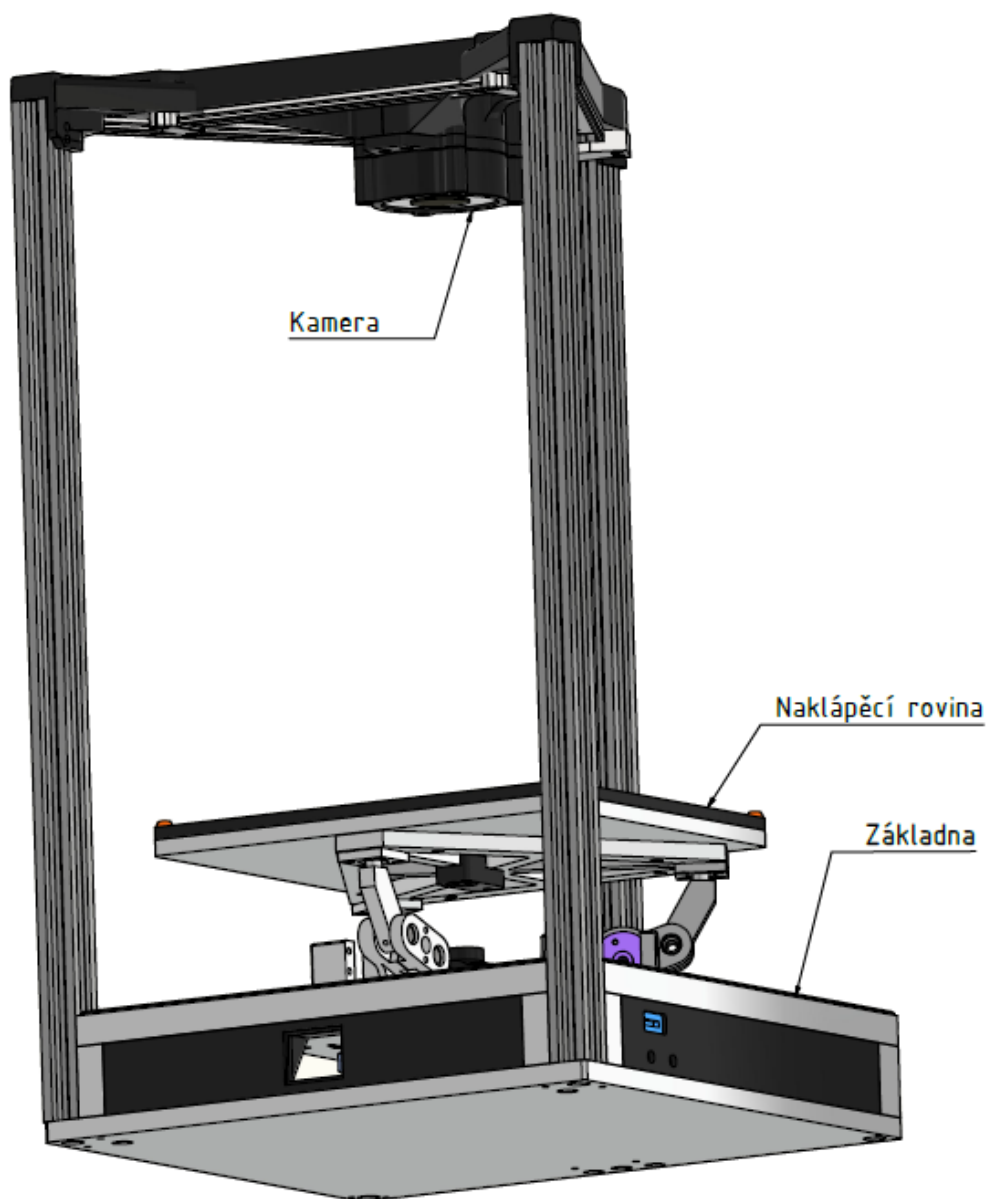
Vrchní část každého z ramen je prostřednictvím vyklápěcího ložiska efektivně spojena s herní platformou, což umožňuje její přesné a kontrolované naklápění. Herní platforma, stejně jako ramena, je vyrobena z hliníkové slitiny EN AW 6082 [9].

Důležitým aspektem této konstrukce je, že díky vysoké tuhosti spojení a kvalitnímu materiálu dochází k minimálnímu chvění nebo vibracím během provozu. To je zásadní pro přesné řízení polohy kuličky na herní platformě, neboť jakékoli nežádoucí pohyby by mohly způsobit nepředvídatelné změny v trajektorii kuličky a tím negativně ovlivnit celkovou přesnost a spolehlivost systému. Stabilita a absence vibrací tak přispívají k vysoké kvalitě ovládaní a zajišťují, že herní platforma reaguje pouze na záměrné pohybové příkazy, což je klíčové pro dosažení přesné regulace polohy kuličky.



Obrázek 17: Ukotvení roviny k ramenům

Kamera, klíčový prvek pro vizuální sledování a analýzu, je strategicky umístěna nad naklápací rovinou, což jí umožňuje získávat nepřerušovaný a celkový pohled na dění na platformě. Toto umístění je zásadní pro efektivní monitorování polohy kuličky a umožňuje precizní zpracování obrazových dat v reálném čase. Elektronické komponenty, nezbytné pro správnou funkci a regulaci celého systému, jsou inteligentně integrovány do základny zařízení. Tímto způsobem jsou skryty před zraky uživatelů a zároveň chráněny před vnějšími vlivy, což zajišťuje nejen estetickou, ale i funkční integritu zařízení.



Obrázek 18: Popis zařízení

5.2.2 Pohyb ramen

Ramena jsou poháněna servomotory typu HBL550, jež jsou významné díky své vysoké rychlosti a točivému momentu, což je zásadní pro precizní a efektivní manipulaci v rámci aplikací vyžadujících dynamický pohyb. Tyto motory jsou napájeny napětím 7,4 V, což je klíčové pro dosažení jejich vynikajících výkonových charakteristik, konkrétně točivého momentu 36 kg-cm. Tento moment umožňuje motorům aplikovat významnou sílu, což je nezbytné pro přesné pohybové operace.

Dále, schopnost motorů realizovat otočku o 60° v časovém úseku 0,094 sekundy je přímým důsledkem jejich vysoké rychlosti. Tato rychlostní charakteristika přispívá k rychlé reakci systému na příkazy a umožňuje rychlé provedení pohybových sekvenčních úkolů s vysokou přesností. Výběr servomotorů HBL550 pro pohon ramen tak reflektuje potřebu spojení vysoké síly, rychlosti a přesnosti, aby bylo zajištěno efektivní a spolehlivé řízení pohybu. [11]



Obrázek 19:
Servo HBL550 [11]

5.2.3 Snímání polohy kuličky

V procesu snímání polohy kuličky je klíčovou součástí využití vysokokvalitní kamery, konkrétně modelu Ausdom AF660. Tato kamera disponuje rozlišením 1920×1080 pixelů, což odpovídá standardu Full HD, a poskytuje detailní a ostré obrazové záznamy. Její zorný úhel 75° umožňuje široké pokrytí monitorovaného prostoru, což je nezbytné pro efektivní sledování a lokalizaci kuličky v rámci experimentálního prostředí. Kamera je připojena a napájena pomocí USB portu. Dále pracuje s frekvencí 30 snímků za sekundu (FPS), což zajišťuje plynulý a kontinuální záznam pohybu kuličky bez ztráty kritických detailů. [12]

Je třeba podotknout, že výběr této kamery byl motivován jejími výjimečnými vlastnostmi, které se harmonicky skloubily s požadavky projektu.



Obrázek 20:
Ausdom AF660 [12]

5.2.4 Řízení systému

V rámci implementace systému je využito mikrokontroléru Arduino Nano, který funguje v podřízené (slave) roli. Hlavní úlohou tohoto mikrokontroléru je realizace synchronního pohybu servomotorů na základě instrukcí přijatých z nadřazeného počítače (master). Komunikace mezi mikrokontrolérem Arduino a počítačem je realizována prostřednictvím sériové linky, což umožňuje spolehlivý a efektivní přenos dat. Veškeré komplexní výpočty určující přesnou polohu a natočení servomotorů jsou prováděny na straně počítače, což počítač předurčuje k roli výpočetního jádra systému. Arduino zde zajišťuje bezprostřední řízení servomotorů podle specifikovaných parametrů. Tento přístup umožňuje efektivní rozdělení výpočetního zatížení a optimální využití obou zařízení ve svých specifických rolích.

5.2.5 Parametry řešení

Maximální náklon roviny:

- Technicky možný rozsah náklonu: $\pm 7^\circ$ v obou osách naklápění.
- Softwarově stanovené omezení: $\pm 5^\circ$ v obou osách naklápění.
- Při provozu s automatickým řízením: $\pm 3^\circ$ v obou osách naklápění.

Rychlost náklonu:

- Časový úsek potřebný pro pohyb od $A-5^\circ$; $B-5^\circ$ do $A+5^\circ$; $B+5^\circ$ je 0,128 sekundy.
- Časový úsek potřebný pro pohyb od $A-3^\circ$; $B-3^\circ$ do $A+3^\circ$; $B+3^\circ$ je 0,068 sekundy.

Uvedené časy jsou teoretické odhady.

Rychlost regulace:

- Rychlost regulace je omezena snímkovou frekvencí kamery snímající zařízení, která je 30 snímků za sekundu.

Rozměry herní platformy:

- 223,5 mm x 187,5 mm

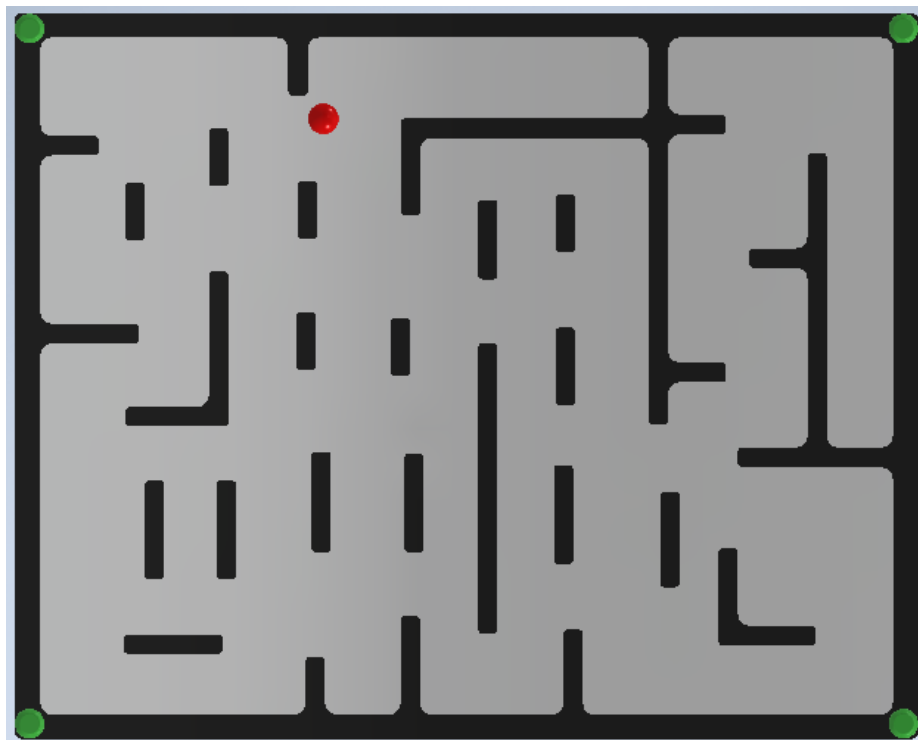
Rozměry zařízení (bez kabelů)

- Šířka: 310 mm
- Hloubka: 290 mm
- Výška: 522 mm

6 REGULACE POLOHY KULIČKY

6.1 Určování polohy kuličky

Poloha kuličky je snímána prostřednictvím kamery instalované nad herním polem. Tato kamera provádí snímání herní plochy s frekvencí 30 snímků za sekundu (FPS). Každý z těchto snímků je následně analyzován a na základě analýzy je generován regulační zásah, který je využit pro úpravu polohy kuličky v rámci herního prostoru.



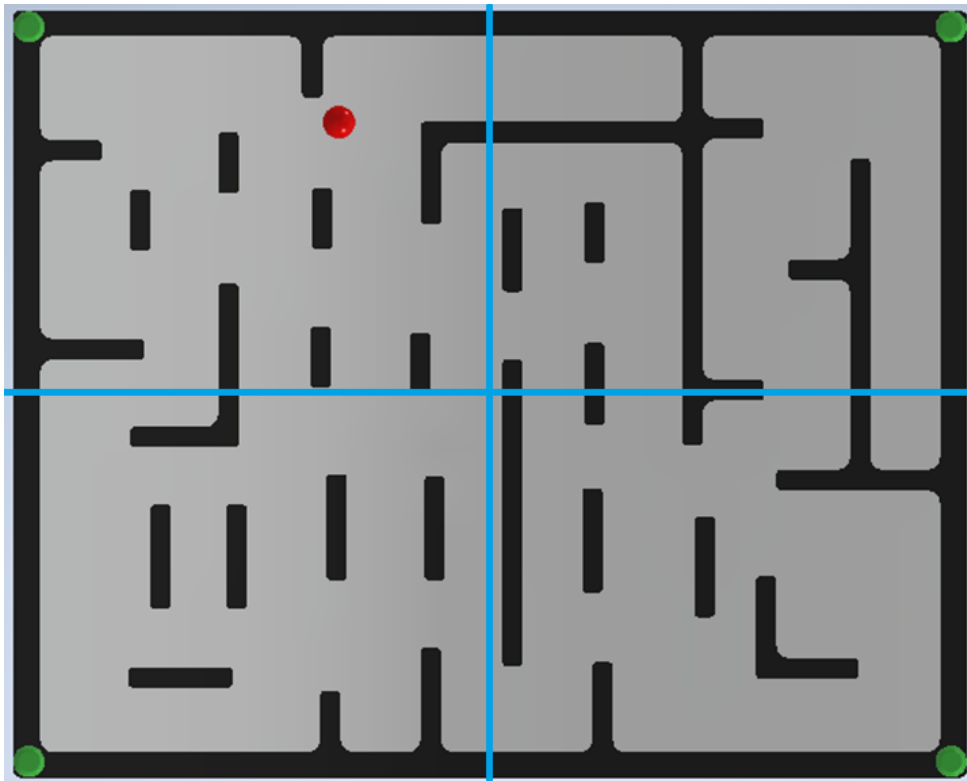
Obrázek 21: Kamerový snímek

6.1.1 Rozpoznání objektů v kamerovém snímku

V konstrukční fázi zařízení byl kladen důraz na výběr barev v oblasti sledované kamerou, což umožňuje identifikaci objektů výhradně na základě barevných rozdílů. Základní plocha a mantinely bludiště byly navrženy v černé barvě, referenční body na herní ploše v zelené a kulička v červené.

Veškeré referenční body mají uniformní barvu, nicméně jejich pozice je předvídatelná. Algoritmus vyhledává tyto body vždy v určitém kvadrantu snímku z kamery a na základě toho rozděluje detekované body.

Pro evaluaci barev byl zvolen barevný model HSB (Hue, Saturation, Brightness), který poskytuje lepší možnosti filtrace ve srovnání s RGB (Red, Green, Blue) modelem.



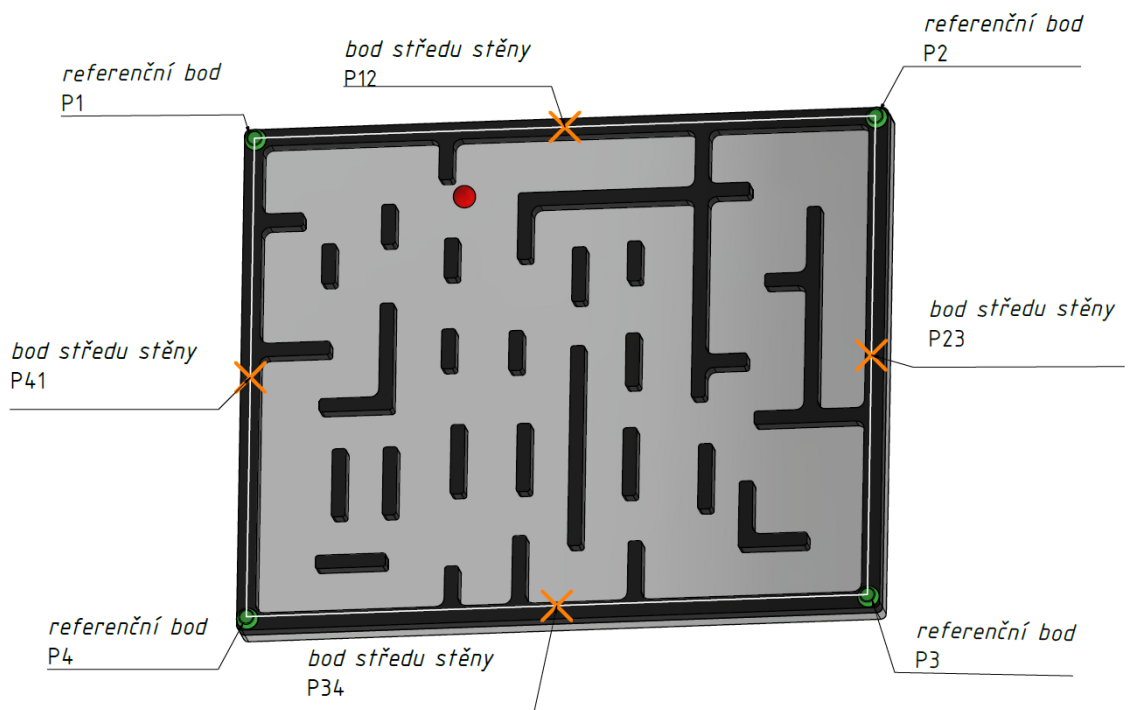
Obrázek 22: Kamerový snímek rozdělený na kvadranty

6.1.2 Určování polohy objektů na kamerovém snímku

Jednotlivé objekty jsou nejprve uloženy jako shluk poloh pixelů. Pro určení jejich středu vůči levému hornímu okraji snímku lze provést jednoduchý aritmetický průměr všech pixelů jednotlivých objektů.

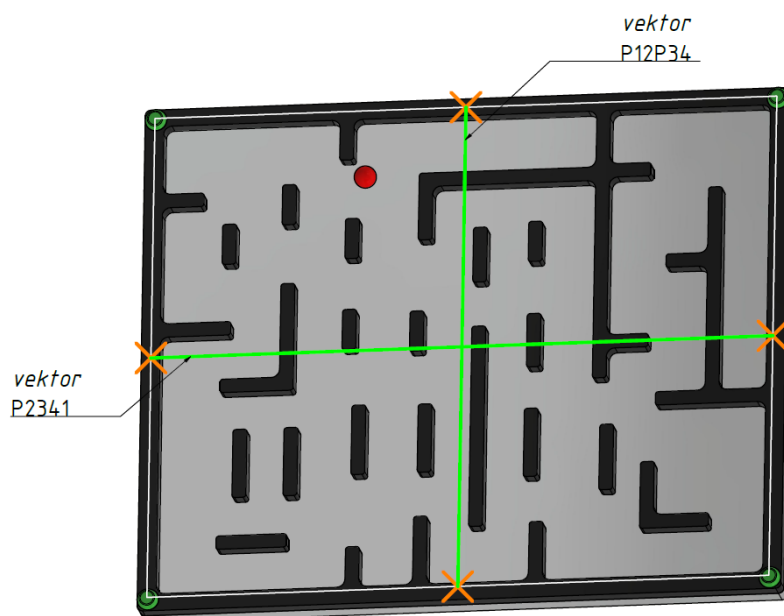
Při určování polohy kuličky vůči středu herní plochy je nutné pracovat i se skreslením rozměrů herní plochy vlivem náklonu.

V prvním kroku jsou nalezeny středy mezi referenčními body stěn (viz obrázek 26. zelené body)



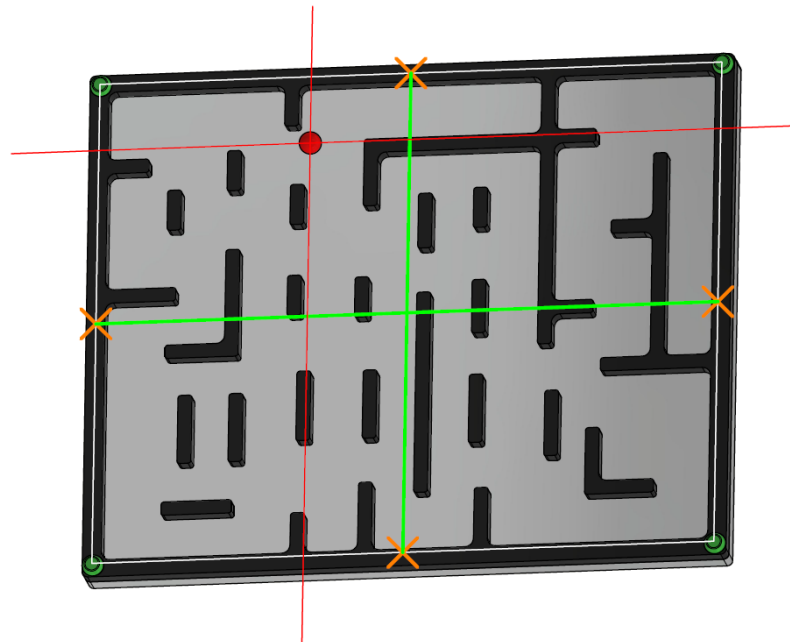
Obrázek 23: Kamerový snímek, nalezení středů stěn

V další fázi analytického procesu dochází ke generování směrových vektorů, které se zakládají na středových bodech stěn. To se provádí vytvořením prvního směrového vektoru mezi body P12 a P34 a druhého mezi body P23 a P41, jak je ilustrováno na obrázku 24.



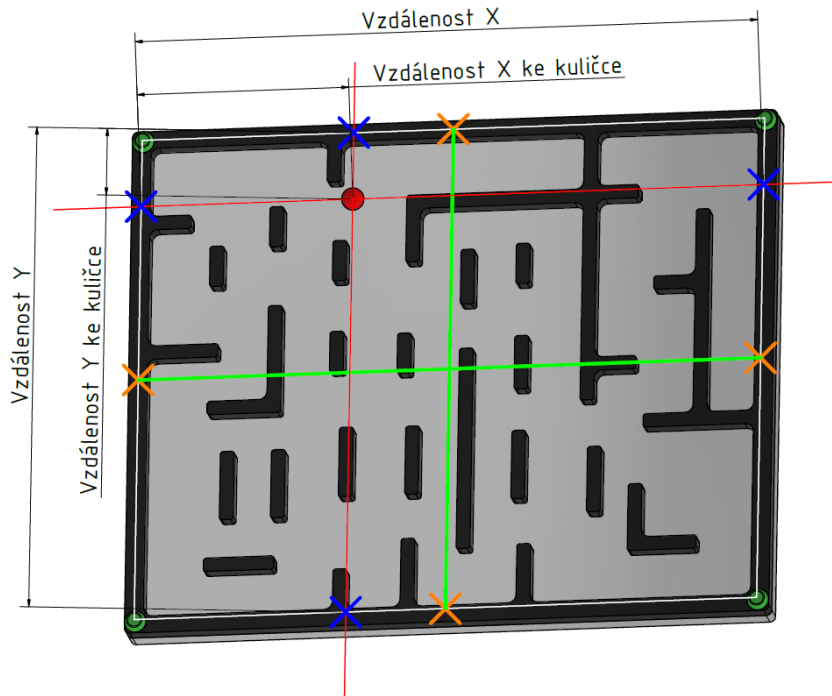
Obrázek 24 Kamerový snímek, tvorba vektorů

Postup pokračuje konstrukcí přímek procházejících středem kuličky, jež budou rovnoběžné s předchozími směrovými vektory, jak je zaznamenáno na obrázku 25.



Obrázek 25: Kamerový snímek, tvorba přímek

Využitím zmíněných přímek se identifikují jejich průsečíky s dalšími přímkami definovanými referenčními body, čímž lze určit relevantní vzdálenosti.



Obrázek 26: Kamerový snímek, vzdálenosti

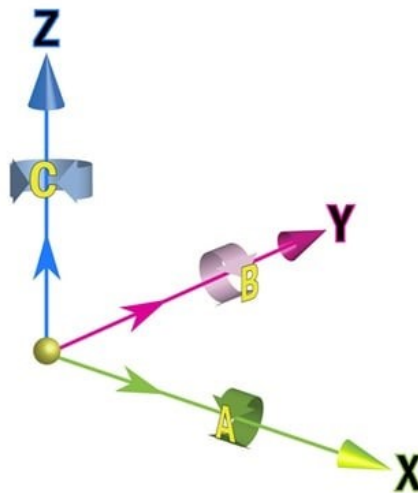
S předem známými skutečnými vzdálenostmi mezi referenčními body lze přejít k výpočtu polohy relativně ke středu herní plochy, jak je popsáno následujícím matematickým postupem:

$$polohaX = \frac{\text{Skutečná vzdálenost ref. bodů } X}{\text{Vzdálenost } X} * \left(\frac{\text{Vzdálenost } X}{2} - \text{Vzdálenost } X \text{ ke kuličce} \right) \quad (4)$$

$$polohaY = -\frac{\text{Skutečná vzdálenost ref. bodů } Y}{\text{Vzdálenost } Y} * \left(\frac{\text{Vzdálenost } Y}{2} - \text{Vzdálenost } Y \text{ ke kuličce} \right) \quad (5)$$

6.2 Naklápění roviny

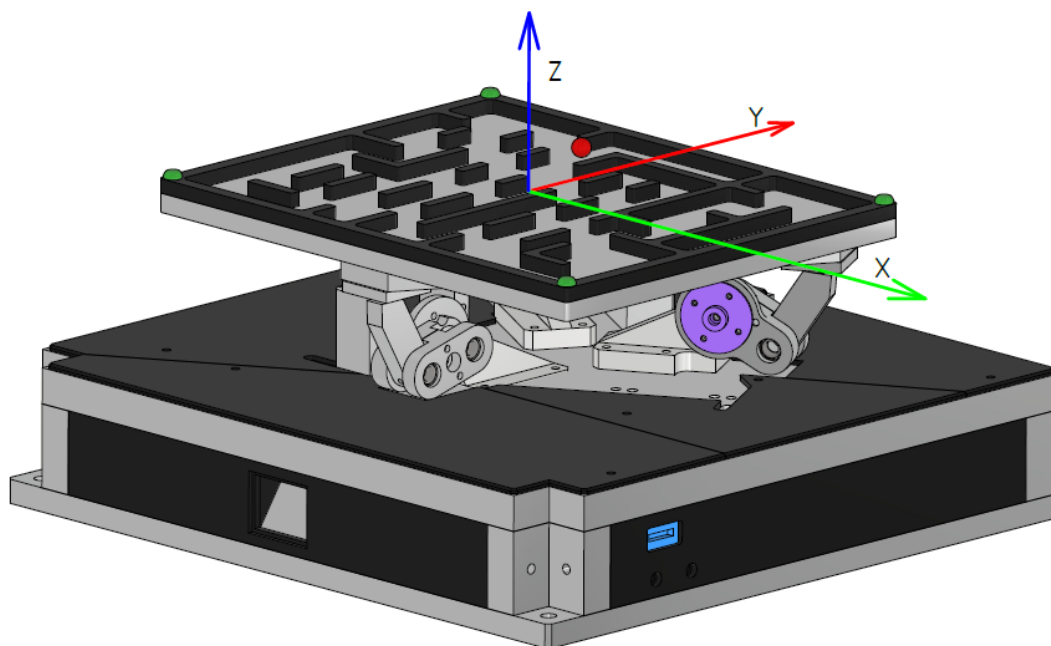
Implementací roviny podpírané třemi rameny, která jsou umístěna pod úhlem 120° vůči sobě, je vyžadováno, aby naklonění v jednotlivých osách řídily všechny tři motory. Systém byl navržen tak, aby umožňoval nezávislou manipulaci s polohou kuličky v osách X a Y, což si vyžaduje samostatnou regulaci pro každou osu. Vstupy pro výpočty tvoří naklonění kolem osy X, označené jako A, a naklonění kolem osy Y, označené jako B.



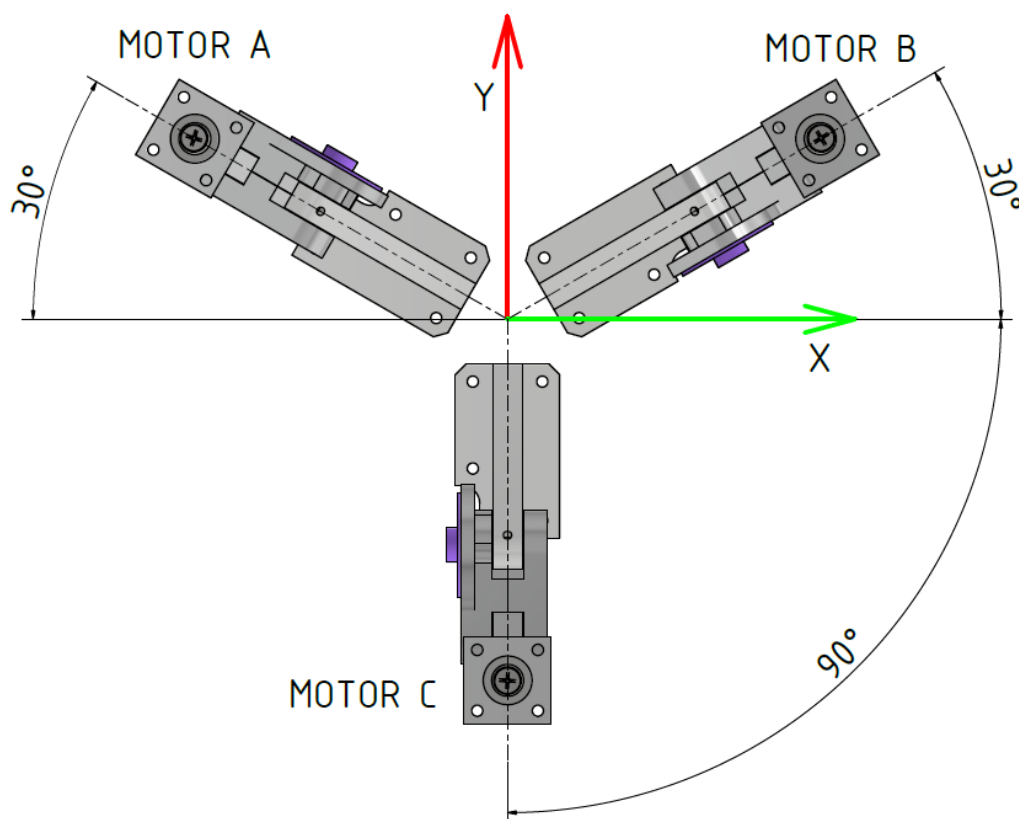
Obrázek 27:

Osový kříž s rotacemi [13]

Výstupy z výpočtů poskytují hodnoty naklonění pro jednotlivé motory, které jsou označeny jako „Motor A“, „Motor B“ a „Motor C“.



Obrázek 28: Osový kříž zařízení



Obrázek 29 Označení motorů

Vstupní úhly jsou převedeny na rotační matice:

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(A) & -\sin(A) \\ 0 & \sin(A) & \cos(A) \end{bmatrix} \quad (6)$$

$$R_y = \begin{bmatrix} \cos(B) & 0 & \sin(B) \\ 0 & 1 & 0 \\ -\sin(B) & 0 & \cos(B) \end{bmatrix} \quad (7)$$

Kde R_x reprezentuje rotaci kolem osy X a R_y kolem osy Y. Rotační matice platformy je vypočítána maticovým násobením:

$$R = R_x \cdot R_y \quad (8)$$

Pro motory A a B je dále nutné matice R rotovat kolem osy Z:

Rotační matice pro Motor A:

$$R_{za} = \begin{bmatrix} \cos(240) & -\sin(240) & 0 \\ \sin(240) & \cos(240) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (9)$$

Rotační matice pro Motor B:

$$R_{zb} = \begin{bmatrix} \cos(120) & -\sin(120) & 0 \\ \sin(120) & \cos(120) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (10)$$

Výsledná matice pro Motor A a Motor B je vypočítána násobením:

$$R_A = R_{za} \cdot R \quad (11)$$

$$R_B = R_{zb} \cdot R \quad (12)$$

Natočení roviny v prostoru vzhledem k ose motoru je vypočítáno pro:

- Motor A:

$$\alpha_A = \arctan\left(\frac{RA[1; 2]}{RA[2; 2]}\right) \quad (13)$$

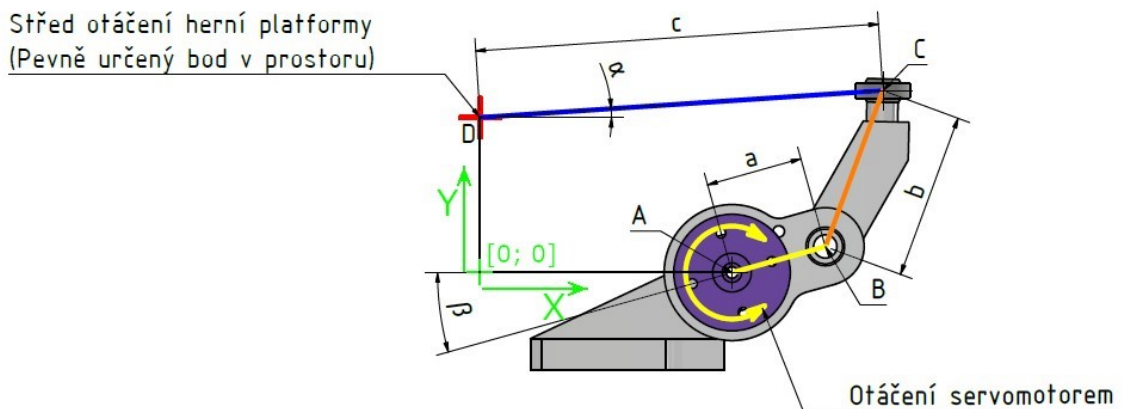
- Motor B:

$$\alpha_B = \arctan\left(\frac{RB[1; 2]}{RB[2; 2]}\right) \quad (14)$$

- Motor C:

$$\alpha_C = \arctan\left(\frac{R[1; 2]}{R[2; 2]}\right) \quad (15)$$

Další výpočty se soustředí na orientaci servomotoru, pro který je využit lokální souřadnicový systém aplikovaný individuálně na každý kloub, jak je znázorněno na Obrázku 30. Neutrální pozice servomotoru je definována jako stav, kdy úhel β je roven nule, což je ilustrováno na Obrázku 30 označeném jako "Kloub". Pro každý kloub se vypočítá úhel natočení, přičemž se používají úhly α , které jsou specifikovány pro každý motor (A, B, C).



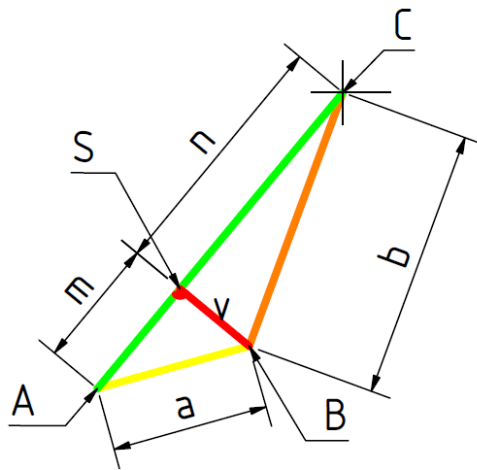
Obrázek 30: Kloub

Během konstrukce byly specifikovány body $D=[D_X; D_Y]=[0; 40]$ a $A=[A_X; A_Y]=[65; 0]$, kde bod D je centrum rotace platformy a bod A označuje osu rotace servomotoru. Další body a úhly jsou určeny následujícími výpočty.

Výpočet polohy bodu C:

$$C = [\cos(\alpha) * c + D_X; \sin(\alpha) * c + D_Y] \quad (16)$$

Výpočet polohy bodu B:



Obrázek 31: Schema části kloubu

$$|AC| = \sqrt{(C_X - A_X)^2 + (C_Y - A_Y)^2} \quad (17)$$

$$a^2 = v^2 + m^2 \quad (18)$$

$$b^2 = v^2 + n^2 \quad (19)$$

$$a^2 - b^2 = m^2 - n^2 = |AC| \cdot (m - n) \quad (20)$$

$$m = \frac{(a^2 - b^2)}{2|AC|} + \frac{|AC|}{2} \quad (21)$$

$$v = \sqrt{a^2 - m^2} \quad (22)$$

$$S = \left[A_X + \left(\frac{m}{|AC|} \right) \cdot (C_X - A_X); A_Y + \left(\frac{m}{|AC|} \right) \cdot (C_Y - A_Y) \right] \quad (23)$$

$$B = [S_x - \left(\frac{v}{|AC|}\right)(A_y - C_y); S_y + \left(\frac{v}{|AC|}\right) \cdot (A_x - C_x)] \quad (24)$$

Výpočet úhlu β , reprezentujícího natočení servomotoru:

$$\beta = \tan^{-1} \left(\frac{B_y - A_y}{B_x - A_x} \right) \quad (25)$$

6.3 Regulátor polohy

Regulace polohy probíhá prostřednictvím PID regulátoru, přičemž každá osa je vybavena regulátorem nezávislým na druhé ose. Experimentální ověření naznačuje, že oba regulátory by měly sdílet identické parametry.

Regulátor polohy využívá následujícího matematického modelu:

$$G_c(s) = Kp + \frac{Ki}{s} + \frac{N * Kd}{1 + \frac{N}{s}} \quad (26)$$

Systém byl namodelován v programu Matlab jako ideální systém druhého řádu (harmonický oscilátor) a v něm určená konstanta k pomocí měření na reálném systému. Konstanta k odpovídá 125.

Přenosová funkce systému:

$$G(s) = \frac{k}{s^2} = \frac{125}{s^2} \quad (27)$$

Přenosová funkce regulátoru:

$$Gr(s) = Kp + \frac{Ki}{S} + N * \frac{Kd}{1 + \frac{N}{s}} = \frac{q_2 * s^2 + q_1 * s + q_0}{s^2 + p_1 * s} \quad (28)$$

Rovnice pro součin zadané přenosové funkce systému a přenosové funkce regulátoru:

$$AP + BQ = D \quad (29)$$

Kde:

$$A = s^2(s^2 + p_1s) \quad (30)$$

$$B = k \quad (31)$$

$$Q = q_2s^2 + q_1s + q_0 \quad (32)$$

$$D = (s + \alpha)^4 \quad (33)$$

Specifikujeme α na hodnotu 2,2 a vypočteme D, což je $(s + 2,2)^4$.

$$D = (s + 2,2)^4 = s^4 + 8,8s^3 + 29,04s^2 + 42,592s + 23,4256 \quad (34)$$

Nyní se porovnají koeficienty v rovnici $AP + BQ = D$ a umístí se póly PID regulátoru.

$$s^2 * (s^2 + p_1s) + k * (q_2 * s^2 + q_1 * s + q_0) = s^4 + D_3s^3 + D_2s^2 + D_1s + D_0 \quad (35)$$

Porovnáním koeficientů získáme:

$$D_3 = 8,8 \quad (36)$$

$$D_3 = 8,8 \quad (37)$$

$$D_2 = 29,04 \quad (38)$$

$$D_1 = 42,592 \quad (39)$$

$$D_0 = 23,4256 \quad (40)$$

Nyní lze vypočítat hodnoty p_1 , q_2 , q_1 a q_0

$$p_1 = D_3 = 8,8 \quad (41)$$

$$q_2 = \frac{D_2}{k} = 0,23232 \quad (42)$$

$$q_1 = \frac{D_1}{k} = 0,340736 \quad (43)$$

$$q_0 = \frac{D_0}{k} = 0,187405 \quad (44)$$

Po dopočítání p_1 , q_2 , q_1 a q_0 se dopočítají hodnoty pro PID regulátor:

$$N = p_1 = 8,8 \quad (45)$$

$$Ki = \frac{q_0}{N} = 0,021296 \quad (46)$$

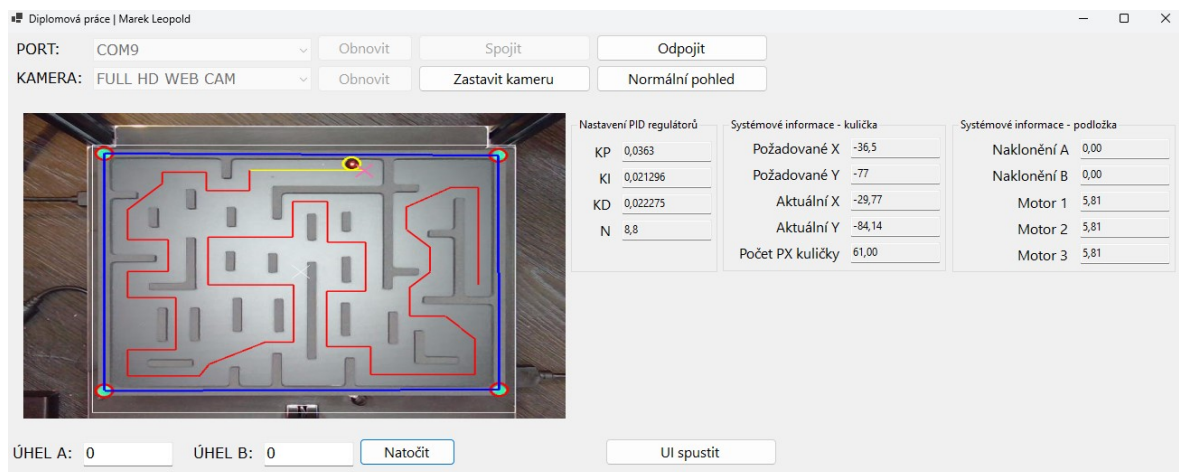
$$Kp = \frac{q_1 - Ki}{N} = 0,0363 \quad (47)$$

$$Kd = \frac{q_2 - Kp}{N} = 0,022275 \quad (48)$$

6.4 Řízení trajektorie kuličky v bludišti

6.4.1 Definice trajektorie

Trajektorie kuličky v bludišti je definována souborem klíčových bodů, které určují změny směru jejího pohybu na nakloněné rovině. Tyto body jsou strategicky umístěny na místech, kde je očekávána změna směru kuličky, a slouží jako navigační milníky pro její průchod bludištěm. Tento systém bodů umožňuje přesné vedení kuličky po předem určené trajektorii.



Obrázek 32: Zobrazená trajektorie v GUI

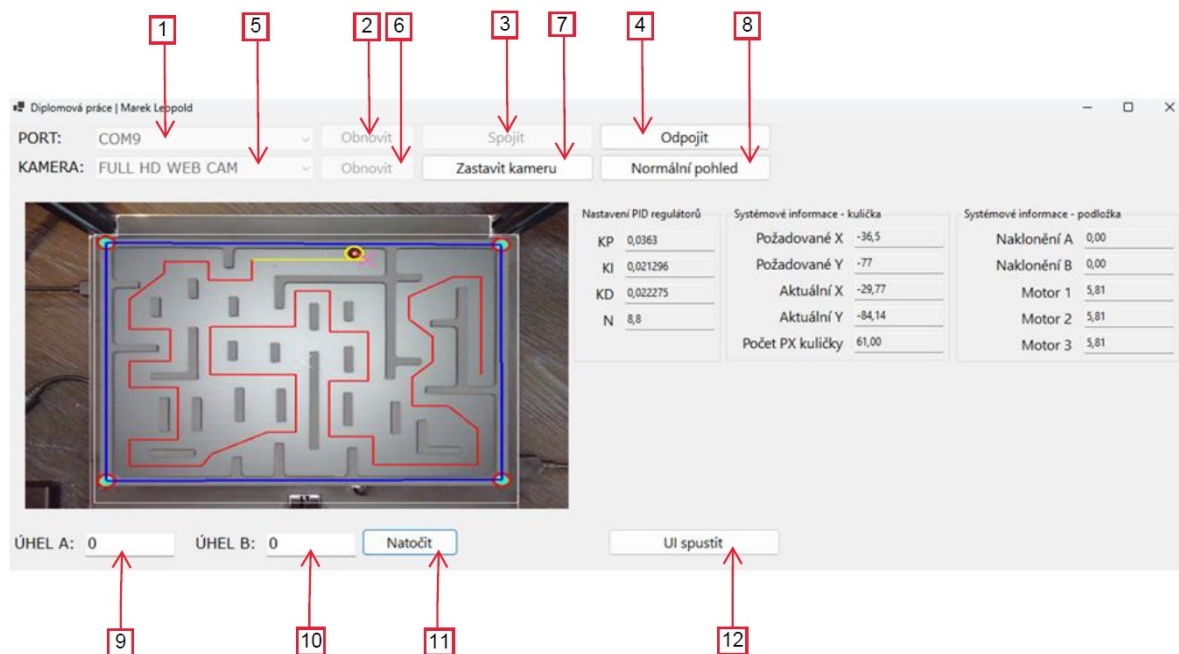
6.4.2 Kontrola dosažení bodu

Systém sleduje polohu kuličky a porovnává ji s jejím aktuálním cílovým bodem. Kontrola probíhá pomocí určení, zda se kulička nachází v předem definované oblasti okolo cílového bodu. Pokud ano, systém registruje dosažení bodu. Po několikrát opakovaném dosažení bodu, což značí stabilní pozici kuličky, systém aktualizuje cílový bod na další v pořadí. Tento proces zajišťuje, že kulička postupuje po trajektorii v souladu s plánem.

6.4.3 Výpočet dalšího bodu a segmentace trajektorie

Po úspěšném dosažení cílového bodu systém vypočítá vzdálenost k následujícímu bodu a rozdělí trajektorii na kratší segmenty, jejichž maximální délka odpovídá stanovenému kroku. Tato segmentace je klíčová pro regulaci pohybu kuličky, jelikož předchází vstupu příliš velkých hodnot do PID regulátoru. Příliš velké hodnoty by mohly vést k razantním a méně předvídatelným reakcím regulátoru, což by mohlo negativně ovlivnit stabilitu a přesnost pohybu kuličky. Segmentace trajektorie tedy přispívá k jemnější a stabilnější regulaci, což je zásadní pro udržení kuličky na plánované cestě s vysokou přesností.

7 POPIS GUI APLIKACE



Obrázek 33: GUI

1. **PORT:** Výběr sériového portu pro komunikaci s mikropočítačem
2. **Obnovit:** Obnovení výběru portu
3. **Spojit:** Zahájení komunikace s mikropočítačem
4. **Odpojit:** Ukončení komunikace s mikropočítačem
5. **KAMERA:** Výběr kamery zařízení
6. **Obnovit:** Obnovení výběru kamer
7. **Spustit kameru / Zastavit kameru:** Spuštění / vypnutí kamery zařízení
8. **Normální pohled / UI pohled:** Přepínání pohledu – zobrazení identifikovaných prvků a trajektorie kuličky
- 9., 10., 11 **ÚHEL A, ÚHEL B, Natočit:** Při deaktivovaném UI řízení lze rovinu naklápět zadáním hodnot a stisknutím natočit.
12. **UI spustit / UI vypnout:** Spuštění automatického řízení

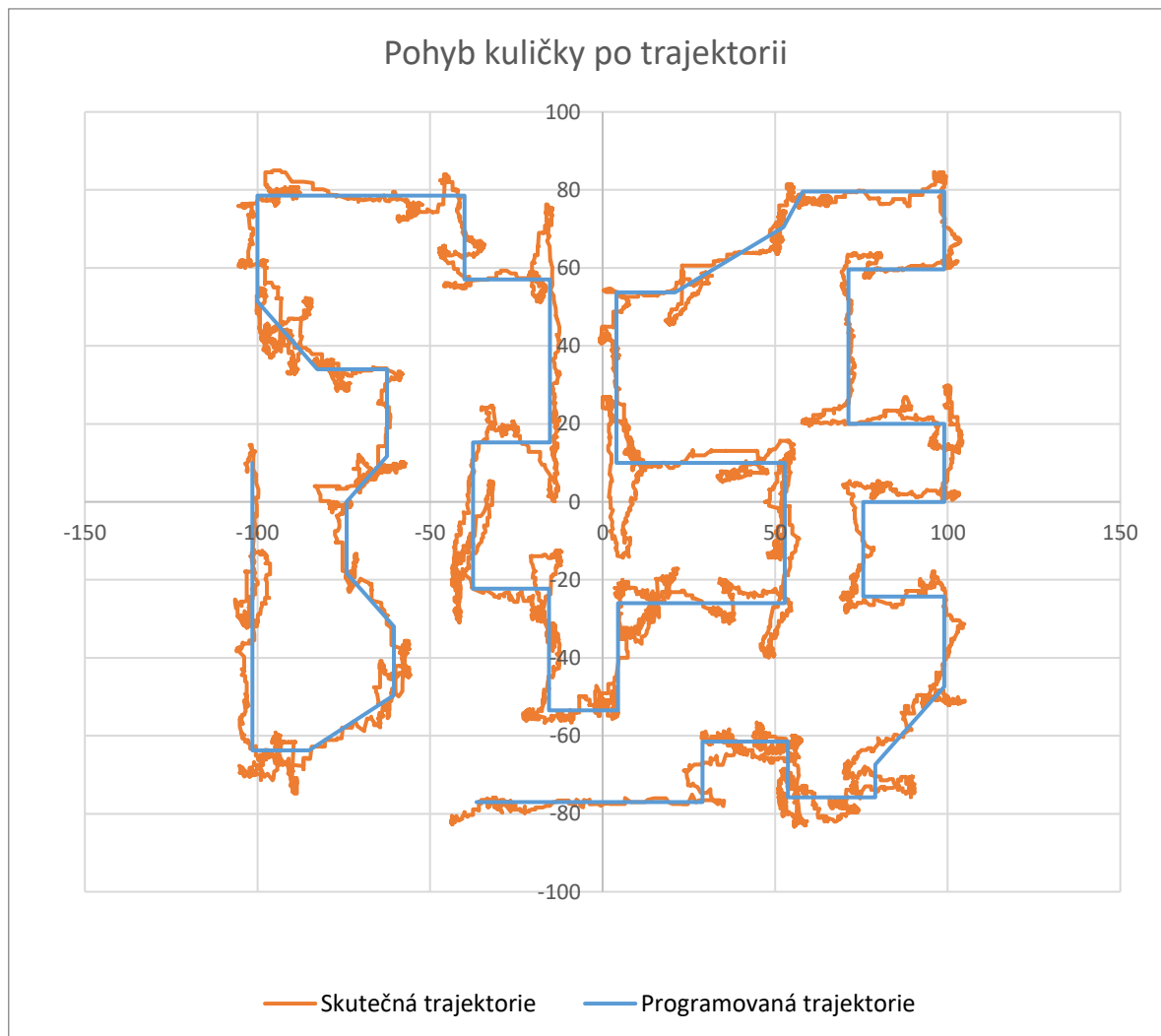
Oblast „Nastavení PID regulátorů“: Zobrazuje statické informace ze zdrojového kódu, zobrazuje nastavení obou PID regulátorů

Oblast „Systémové informace – kulička“: Zobrazuje požadovanou a aktuální polohu kuličky a počet pixelů, které byly vyhodnoceny jako kulička od kamery

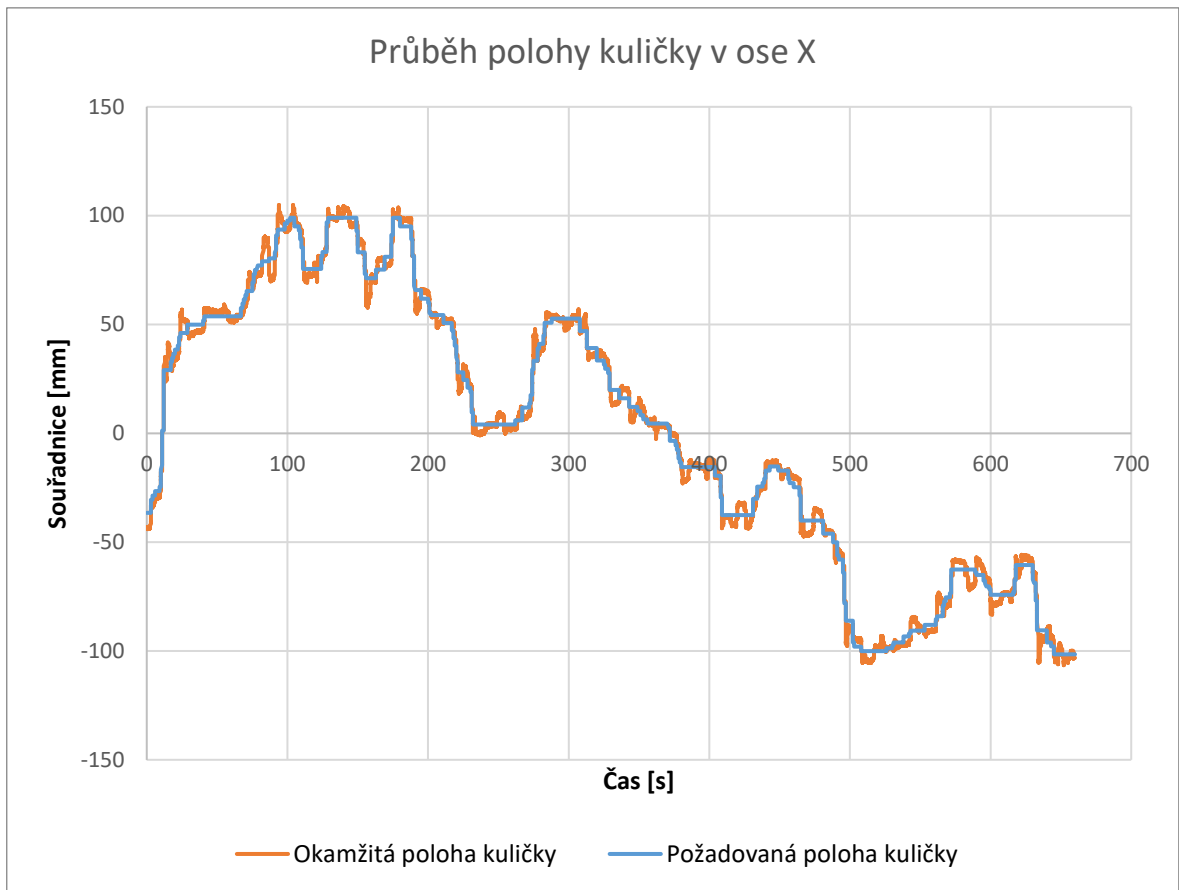
Oblast „Systémové informace – podložka“: Zobrazuje aktuální natočení podložky a natočení jednotlivých motorů

8 VIZUALIZACE POHYBU

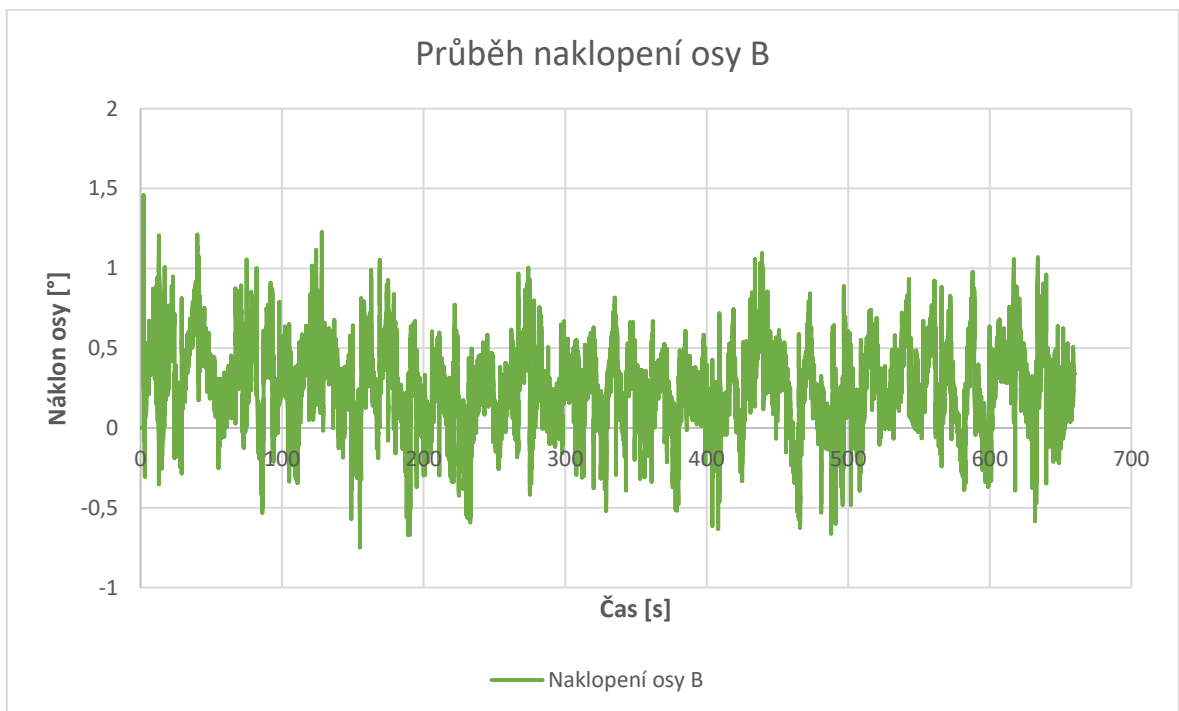
Ve vizualizaci prezentované na obrázku 34 je ilustrována poloha kuličky, která byla vzorkována v intervalu 0,033 sekundy (reprezentována oranžovou čarou), a to v kontrastu s naprogramovanou trajektorií (značenou modrou čarou). Obrázek 35 demonstruje průběh polohy kuličky na ose X ve srovnání s předem definovanou trajektorií, zatímco obrázek 36 ukazuje reakci systému na odchylky od této trajektorie pomocí naklápění roviny B. Analogicky, obrázek 37 zobrazuje průběh polohy kuličky na ose Y v kontrastu s požadovanou trajektorií a obrázek 38 ilustruje reakci systému na pozorované odchylky od předem definované trajektorie, a to prostřednictvím naklápění roviny A.



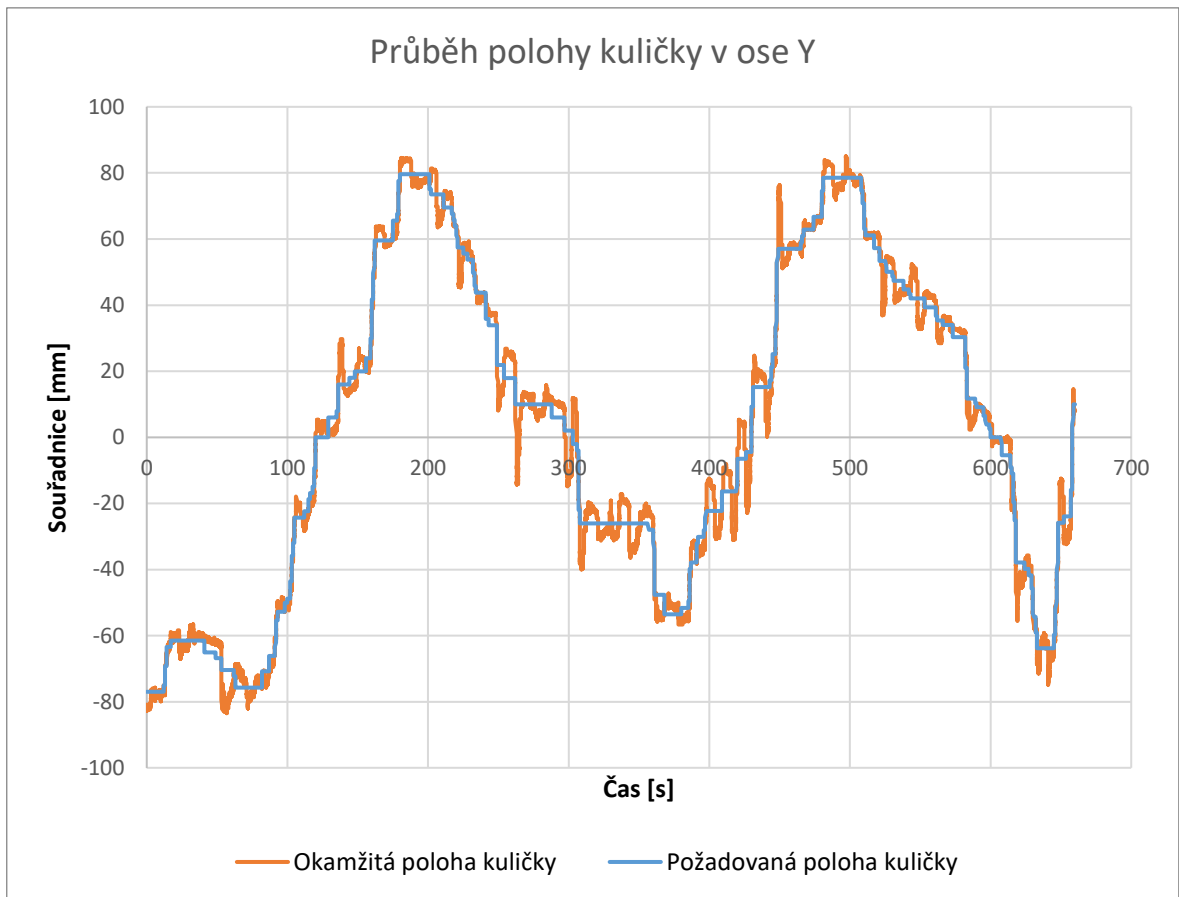
Obrázek 34: Pohyb kuličky po trajektorii



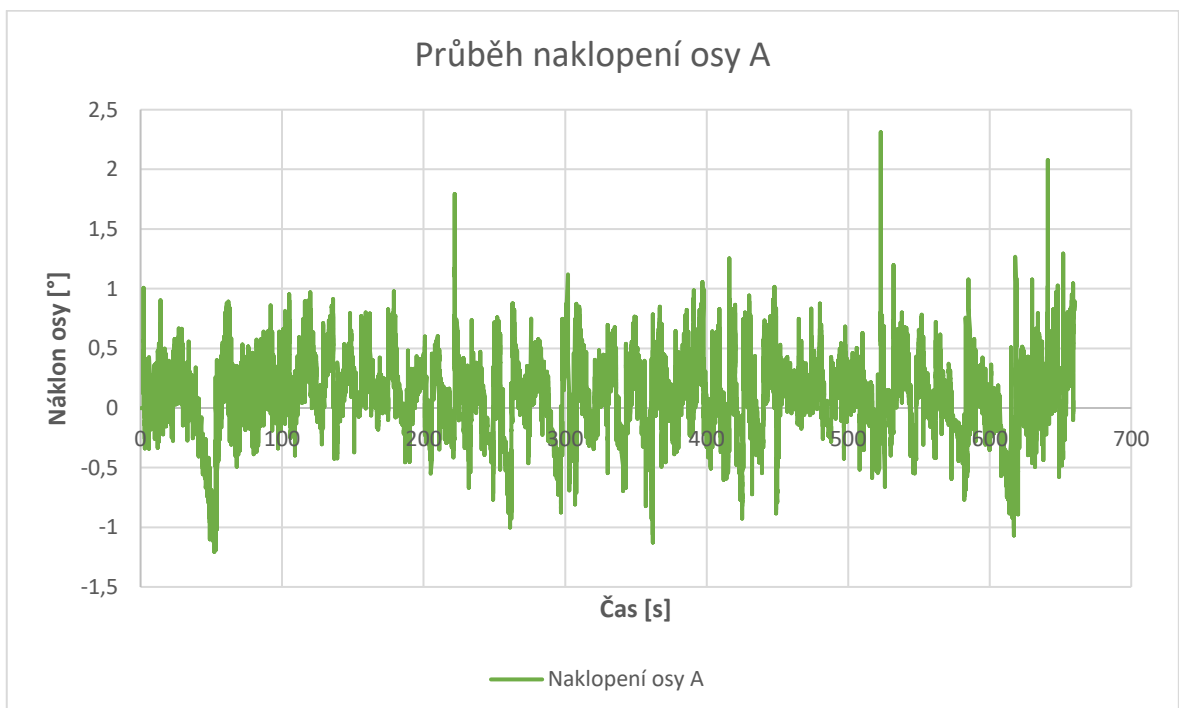
Obrázek 35: Průběh polohy kuličky v ose X



Obrázek 36: Průběh náklonu osy B



Obrázek 37: Průběh polohy kuličky v ose Y



Obrázek 38: Průběh náklonu osy A

9 SROVNÁNÍ AUTOMATICKÉHO A MANUÁLNÍHO OVLÁDÁNÍ

V rámci této diplomové práce bylo provedeno srovnání pohybu kuličky na mechanicky ovládané a automaticky ovládané nakloněné rovině s cílem porozumět rozdílům v jejich chování a účinnosti.

Automaticky ovládaná nakloněná rovina, vybavená motorizovaným systémem, demonstruje schopnost dosáhnout cíle s kuličkou přibližně za 11 minut. Tento výsledek představuje významnou výhodu v porovnání s manuálně ovládanou rovinou. Avšak, je důležité poznamenat, že automatická nakloněná rovina byla navržena s určitými zjednodušeními, včetně absence překážek ve formě děr, které by mohly ovlivnit pohyb kuličky. Tato absenci překážek je třeba vzít v úvahu při interpretaci výsledků, neboť zvyšují náročnost.

Naopak, manuálně ovládaná nakloněná rovina, Woody Labyrint z bodu 5.1.1, která je závislá na lidské interakci, neposkytla očekávané výsledky, neboť se nepodařilo dostat kuličku do cíle. Tento výsledek naznačuje náročnost a obtížnost ručního ovládnání nakloněné roviny, což může být způsobeno nedostatečnou přesností a kontrolou nad pohybem kuličky. Tento výsledek také zdůrazňuje význam automatického ovládnání při dosahování požadovaných cílů v podobných experimentálních prostředích.

V závěru lze říci, že mechanicky ovládaná a automaticky ovládaná nakloněná rovina představují odlišné přístupy k manipulaci s pohybem kuličky. Zatímco automatické ovládnání nabízí výhodu v přesnosti a konzistenci, manuální ovládnání může být náchylné k chybám a obtížím v dosažení požadovaných cílů. Tyto poznatky mohou poskytnout cenné informace pro další výzkum v oblasti automatizace a řízení pohybu na nakloněných rovinách.

ZÁVĚR

V diplomové práci byl úspěšně navržen a implementován systém pro řízení kuličky na nakloněné rovině. Výsledky testování ukázaly, že kombinace PID regulátoru a sofistikovaného systému detekce poskytuje efektivní metodu pro přesné a stabilní řízení pohybu kuličky. Dále bylo prokázáno, že využití jednodeskového počítače je efektivní a ekonomicky výhodné řešení pro takový typ aplikace. Doporučení pro budoucí práci zahrnují integraci pokročilejších algoritmů zpracování obrazu a rozšíření funkčnosti systému pro použití v složitějších aplikacích.

SEZNAM POUŽITÉ LITERATURY

- [1] ARDUINO. Arduino. Online. C2024. Dostupné z: <https://www.arduino.cc/>. [cit. 2024-02-21].
- [2] RASPBERRY PI. Raspberry Pi. Online. C2024. Dostupné z: <https://www.raspberrypi.com/>. [cit. 2024-02-24].
- [3] BeagleBoard [online]. c2024 [cit. 2024-02-24]. Dostupné z: <https://www.beagle-board.org/>
- [4] HARDKERNEL CO., LTD. ODROID. Online. C2019. Dostupné z: <https://www.hardkernel.com/>. [cit. 2024-02-25].
- [5] BANANA PI TEAM. Banana Pi. Online. C2022. Dostupné z: <https://www.banana-pi.org/>. [cit. 2024-02-28].
- [6] INNOVATIVE AUTOMATION INC. What is PLC Programming and Automation? Online. (c)2022. Dostupné z: <https://www.innovativeautomation.com/plc-programming-automation/>. [cit. 2024-03-29].
- [7] SKF. GE 6 C. Online. Dostupné z: <https://www.skf.com/cz/productinfo/productid-GE%206%20C>. [cit. 2024-04-24].
- [8] PROPRUMYSL.CZ. MTM 618/6 ZZ (628/6-2Z) kuličkové ložisko. Online. (c)2024. Dostupné z: <https://www.proprumysl.cz/mtm-618-6-zz-628-6-2z-kulickove-lozisko/>. [cit. 2024-04-24].
- [9] NPS PROAL OSTRAVA. EN AW-6082 (AlSi1MgMn). Online. (c)2019. Dostupné z: <https://proal.cz/hlinik/slitiny-hliniku/en-aw-6082/>. [cit. 2024-04-24].
- [10] ČÍSLICOVÁ A MIKROPROCESOROVÁ TECHNIKA. Online, učební text a návody do cvičení, vedoucí Ing. Michal Prauzek, Ph.D. Ostrava: Vysoká škola báňská – Technická univerzita Ostrava, 2013. Dostupné z: https://home1.vsb.cz/~pra132/files/CMT_prauzek_final_1_2.pdf. [cit. 2024-03-29].
- [11] MARK STAR. HBL550. Online. (c)2024. Dostupné z: <https://mks-servo.com/HBL550>. [cit. 2024-03].
- [12] ALZA A.S. Ausdom AF660. Online. (c) 1994 - 2024. Dostupné z: <https://www.alza.cz/ausdom-af660-d6345154.htm>. [cit. 2024-04-02].
- [13] COPE, Mike. 5-AXIS CNC MACHINE CONFIGURATIONS: DESIGN PROS AND CONS. Online. HURCO. 5-AXIS CNC MACHINE CONFIGURATIONS: DESIGN PROS AND CONS. Dostupné z:

- <https://blog.hurco.com/blog/bid/328587/5-axis-cnc-machine-configurations-the-pros-and-cons>. [cit. 2024-04-11].
- [14] RAVEO. Jak funguje servomotor a zpětnovazební řízení - RAVEO - World of motion. Online. Dostupné z: <https://www.raveo.cz/aktualita/jak-funguje-servomotor-a-zpetnovazebni-rizeni/>. [cit. 2024-04-13].
- [15] BIG HOBBY.CZ. RC Serva. Online. (c)2016-2024. Dostupné z: https://www.bighobby.cz/rc-serva/?gad_source=1&gclid=Cj0KCQJw2uiwB-hCXARIsACMvIU20YRuwuDwAR1tm_SnVn0bNh-0GC-OHdKIcpo99tIgWsOteePPwWzwaAjrEALw_wcB. [cit. 2024-04-13].
- [16] BOTLAND. Vše o krokových motorech - konstrukce, princip činnosti, aplikace Botland - obchod s robotikou. Online. (c)2024. Dostupné z: <https://botland.cz/content/220-vse-o-krokovych-motorech-konstrukce-princip-cinnosti-aplikace>. [cit. 2024-04-13].
- [17] FISHER, Robert; PERKINS, Simon; WALKER, Ashley a WOLFART, E. HIPR. Online. (c)2003. Dostupné z: <https://homepages.inf.ed.ac.uk/rbf/HIPR2/sobel.htm>. [cit. 2024-04-14].
- [18] IQS DIRECTORY. Electric Actuators: Types, Applications, Benefits, and Design. Online. (c)2024. Dostupné z: <https://www.iqsdirectory.com/articles/linear-actuator/electric-actuators.html>. [cit. 2024-04-15].
- [19] EETECH MEDIA, LLC. Pneumatic Actuator Response | Basic Principles of Control Valves and Actuators | Textbook. Online. Dostupné z: <https://control.com/textbook/control-valves/pneumatic-actuator-response/>. [cit. 2024-04-15].
- [20] ACTUATION VALVE. Rack and Pinion Actuators | By Actuation Valve & Control. Online. Dostupné z: <https://www.actuation.co.uk/rack-and-pinion-actuators/>. [cit. 2024-04-15].
- [21] HYDRAULIC CYLINDER MANUFACTURING & REPAIR - MAVERICKMACHINE.CA. Hydraulic Actuators 101 | What is it, Types & Comparison. Online. (c)2024. Dostupné z: <https://www.maverickmachine.ca/news/hydraulic-actuators-101/>. [cit. 2024-04-15].
- [22] IMI CRITICAL ENGINEERING. Hydraulic Powered Actuator (HPA) | IMI Critical Engineering. Online. (c)2024. Dostupné z: <https://www.imi-critical.com/product/hydraulic-powered-actuator-hpa/>. [cit. 2024-04-15].

- [23] NETCAM.CZ. Obrazové snímače CCD vs. CMOS. Online. Dostupné z: <https://netcam.cz/encyklopedie-ip-zabezpeceni/obrazove-snimace-ccd-cmos.php>. [cit. 2024-04-15].
- [24] MEGAPIXEL S.R.O. Jak se vyznat v názvech a zkratkách u objektivů | Megapixel [online]. (c) 2001-2024 [cit. 2024-04-15]. Dostupné z: <https://www.megapixel.cz/jak-se-vyznat-v-zkratkach-u-objektivu>
- [25] MEGAPIXEL S.R.O. Jak fotit digitální zrcadlovkou (DSLR) a bezzrcadlovkou: 26. díl - OHNISKOVÁ VZDÁLENOST OBJEKTIVU | Megapixel. Online. (c) 2001-2024. Dostupné z: <https://www.megapixel.cz/jak-fotit-digitalni-zrcadlovkou-dslr-a-bezzrcadlovkou-26-dil-ohniskova-vzdalenost-objektivu>. [cit. 2024-04-15].
- [26] MIKŠÍK, Ondřej. Praktické využití metod digitálního zpracování obrazu. Kroměříž: Gymnázium Kroměříž, Masarykovo náměstí 496, 2007.
- [27] JANDA, Miloš. CANNYHO OPERÁTOR A DALŠÍ POUŽÍVANÉ HRANOVÉ DETEKTORY. Bakalářská práce. Brno: Vysoké učení technické v Brně, 2008.
- [28] TSANKASHVILI, Nika. Comparing Edge Detection Methods. Online. 2018. Dostupné z: <https://medium.com/@nikatsanka/comparing-edge-detection-methods-638a2919476e>. [cit. 2024-04-17].
- [29] jun94. Comparing Edge Detection Methods. Online. 2020. Dostupné z: <https://medium.com/jun94-devpblog/cv-9-object-detection-with-sliding-window-and-feature-extraction-hog-cf1820c86b46>. [cit. 2024-04-17].
- [30] OPENCV. Structural Analysis and Shape Descriptors. Online. (c)2011. Dostupné z: https://www.opencv.org.cn/opencvdoc/2.3.2/html/modules/imgproc/doc/structural_analysis_and_shape_descriptors.html#suzuki85. [cit. 2024-04-17].
- [31] TRILOBYTE STATISTICAL SOFTWARE. SVM - Support Vector Machines. Online. (c)2024. Dostupné z: <https://www.trilobyte.cz/QC-Expert/SVM-Support-Vector-Machines.html>. [cit. 2024-04-22].
- [32] HRUBÝ, Filip. Posouzení možností využití různých predikčních modelů pro dostupná dopravní data. Diplomová práce. Praha: České Vysoké Učení Technické v Praze, 2023.
- [33] MITRENGA, Michal. KONVOLUČNÍ NEURONOVÁ SÍŤ PRO SEGMENTACI OBRAZU. Bakalářská práce. Brno: Vysoké učení technické v Brně, 2018.

- [34] ANANTH, Shilpa. R-CNN for object detection. Online. 2019. Dostupné z: <https://towardsdatascience.com/r-cnn-for-object-detection-a-technical-summary-9e7bfa8a557c>. [cit. 2024-04-22].
- [35] KEITA, Zoumana. YOLO Object Detection Explained. Online. (c)2024. Dostupné z: <https://www.datacamp.com/blog/yolo-object-detection-explained>. [cit. 2024-04-22].
- [36] KEITA, Zoumana. What is a PID Controller : Working & Its Applications. Online. (c)2013-2024. Dostupné z: <https://www.elprocus.com/the-working-of-a-pid-controller/>. [cit. 2024-04-22].
- [37] ALLEGRO RETAIL A.S. Woody Labyrint s naklápěcími rovinami. Online. (c) 2000–2023. Dostupné z: <https://www.mall.cz/drevene-hry-hlavolamy/woody-labyrint-s-naklapecimi-rovinami>. [cit. 2024-04-22].
- [38] ODKARLA.CZ. Labyrint hra Brio 34000 dřevěné. Online. Dostupné z: <https://www.odkarla.cz/labyrint-hra-brio-34000-drevene~p266202>. [cit. 2024-04-22].

SEZNAM OBRÁZKŮ

Obrázek 1: Arduino Nano [1]	11
Obrázek 2: Raspberry Pi 3 [2]	12
Obrázek 3: BeagleBone Black [3]	13
Obrázek 4: ODROID-H3 [4]	14
Obrázek 5: Banana Pi BPI-M4 Berry [5].....	15
Obrázek 6: Elektrický lineární aktuátor [18]	18
Obrázek 7: Pneumatický rotační aktuátor [20]	19
Obrázek 8: Hydraulický lineární aktuátor [22].....	20
Obrázek 9: Servomotory [14]	21
Obrázek 10: Krokové motory [16].....	22
Obrázek 11: Objektiv – zaostřovací a ohnisková vzdálenost [25].....	25
Obrázek 12: Woody Labyrint [37].....	42
Obrázek 13: Labyrint hra Brio 34000 dřevěné [38]	43
Obrázek 14: Fotografie sestrojeného zařízení	44
Obrázek 15: Schéma ramene	45
Obrázek 16: Umístění ramen	46
Obrázek 17: Ukotvení roviny k ramenům	47
Obrázek 18: Popis zařízení	48
Obrázek 19: Servo HBL550 [11].....	49
Obrázek 20: Ausdom AF660 [12]	50
Obrázek 21: Kamerový snímek	52
Obrázek 22: Kamerový snímek rozdělený na kvadranty	53
Obrázek 23: Kamerový snímek, nalezení středů stěn.....	54
Obrázek 24 Kamerový snímek, tvorba vektorů	55
Obrázek 25: Kamerový snímek, tvorba přímek.....	56
Obrázek 26: Kamerový snímek, vzdálenosti	57
Obrázek 27: Osový kříž s rotacemi [13].....	58
Obrázek 28: Osový kříž zařízení	59
Obrázek 29 Označení motorů	59
Obrázek 30: Kloub.....	61
Obrázek 31: Schema části kloubu.....	62
Obrázek 32: Zobrazená trajektorie v GUI	66

Obrázek 33: GUI.....	67
Obrázek 34: Pohyb kuličky po trajektorii.....	69
Obrázek 35: Průběh polohy kuličky v ose X.....	70
Obrázek 36: Průběh náklonu osy B.....	70
Obrázek 37: Průběh polohy kuličky v ose Y.....	71
Obrázek 38: Průběh náklonu osy A.....	71

SEZNAM PŘÍLOH

Příloha P I: Model zařízení

Příloha P II: Firmware pro jednodeskový počítač

Příloha P III: Zdrojový kód software

Příloha P IV: Video dokumentace

Příloha P V: Data pro grafy