



# Možnosti využití 2,5D vision v robotických aplikacích

## Diplomová práce

*Studijní program:* **N0714A270010** Mechatronika  
*Studijní obor:* Mechatronika

*Autor práce:* **Bc. Stanislav Dvořák**  
*Vedoucí práce:* doc. Ing. Josef Černožský, Ph.D.  
Ústav mechatroniky a technické informatiky



## Zadání diplomové práce

# Možnosti využití 2,5D vision v robotických aplikacích

*Jméno a Příjmení:* **Bc. Stanislav Dvořák**  
*Osobní číslo:* M22000044  
*Studijní program:* N0714A270010 Mechatronika  
*Zadávací katedra:* Ústav mechatroniky a technické informatiky  
*Akademický rok:* 2023/2024

### Zásady pro vypracování:

1. Seznamte se s principy počítačového vidění, zejména v oblasti 2,5D a 3D vidění.
2. Seznamte se s možnostmi vision systému Keyence v laboratoři A102.
3. Navrhněte a realizujte integraci systému Universal Robot a vision systému Keyence.
4. Úspěšnou integraci demonstруйте na vhodné úloze

# Prohlášení

Byl(a) jsem seznámen(a) s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, zejména § 60 – školní dílo.

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

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

Diplomovou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum:

Podpis:

# Poděkování

Chtěl bych poděkovat svému vedoucímu doc. Ing. Josefovi Černoorskému, Ph.D. za odborné vedení, věcné rady a čas, který mi byl ochoten poskytnout během tvorby této práce.

# Anotace

Diplomová práce se zabývá možnostmi využití 2,5D vision systémů v robotických aplikacích. První kapitola je zaměřena na aktuálně používané principy v oblasti 2,5D a 3D strojového vidění. Druhá kapitola se věnuje obecným principům počítačového zpracování obrazu. Další kapitola pojednává o možnostech vision systémů od firmy Keyence. Dále následuje návrh a realizace integrace vision systému a kolaborativního robota UR3 od firmy Universal robots. Závěr shrnuje výsledky praktické části.

## Klíčová slova

systém vidění, robotika, průmysl, počítačové vidění, strojové vidění

## Annotation

The thesis deals with the possibilities of using 2.5D vision systems in robotic applications. The first chapter focuses on currently used principles in the field of 2.5D and 3D machine vision. The second chapter is devoted to general principles of computer image processing. The next chapter discusses the capabilities of vision systems from Keyence. This is followed by the design and implementation of the integration of the vision system and the UR3 collaborative robot from Universal robots. The conclusion summarizes the results of the practical section.

## Key words

vision system, robotics, industry, computer vision, machine vision

## Obsah

Seznam obrázků .....	7
Seznam zkratk .....	8
Úvod.....	9
<b>1. Principy počítačového vidění ve 2,5D a 3D .....</b>	<b>10</b>
<b>1.1 2,5D vidění.....</b>	<b>10</b>
<b>1.2 3D vidění.....</b>	<b>11</b>
1.2.1 Stereo vidění .....	11
1.2.2 Laserová triangulace.....	12
1.2.3 Strukturované světlo .....	13
1.2.4 Time-of-Flight .....	14
1.2.5 Časově kódované strukturované světlo .....	15
<b>2. Obecné principy počítačového zpracování obrazu .....</b>	<b>17</b>
<b>2.1 Snímání a digitalizace obrazu .....</b>	<b>17</b>
<b>2.2 Předzpracování obrazu .....</b>	<b>19</b>
<b>2.3 Segmentace.....</b>	<b>24</b>
<b>2.4 Popis objektů.....</b>	<b>26</b>
<b>2.5 Klasifikace.....</b>	<b>27</b>
<b>3. Vision systém Keyence.....</b>	<b>30</b>
<b>3.1 Proprietární funkce Vision systémů Keyence.....</b>	<b>30</b>
<b>3.2 Inspekční nástroje .....</b>	<b>32</b>
<b>3.3 Konfigurace vision systému .....</b>	<b>35</b>
<b>4. Návrh a realizace integrace systému Universal Robots a vision systému Keyence</b>	<b>39</b>
<b>4.1 Instalace komponent .....</b>	<b>40</b>
<b>4.2 Prvotní nastavení.....</b>	<b>44</b>
<b>4.3 Integrace robota a strojového vidění – úloha 1.....</b>	<b>45</b>
<b>4.4 Integrace robota a strojového vidění – úloha 2.....</b>	<b>56</b>
<b>5. Závěr.....</b>	<b>61</b>
<b>Literatura.....</b>	<b>62</b>
<b>Seznam příloh .....</b>	<b>65</b>

# Seznam obrázků

Obr. 1: 2,5D vidění zachycuje hloubku objektu.....	10
Obr. 2: Princip Aktivního stereo vidění .....	12
Obr. 3: Princip strukturovaného světla .....	14
Obr. 4: Princip Time-of-Flight .....	14
Obr. 5: Vysoce kvalitní 3D mračno bodů.....	16
Obr. 6: Korekce geometrického zkreslení .....	20
Obr. 7: Transformace jasové stupnice .....	22
Obr. 8: Ekvalizace histogramu .....	22
Obr. 9: Využití binární morfologie.....	25
Obr. 10: Výsledek dilatace .....	25
Obr. 11: Barvení oblastí .....	26
Obr. 12: Řetězový kód (8 směrů) .....	27
Obr. 13: Scale-space .....	29
Obr. 14: ZoomTrax.....	30
Obr. 15: Princip LumiTrax .....	31
Obr. 16: ShapeTrax 3 .....	32
Obr. 17: Inspekční nástroj Auto-Teach .....	33
Obr. 18: Rozdíl detekčních schopností kamery v závislosti na rozlišení .....	36
Obr. 19: Způsoby osvětlení .....	37
Obr. 20: Metoda doplňkové barvy.....	38
Obr. 21: Instalační skříň vision systému .....	40
Obr. 22: Řídicí jednotka systému a rozšiřující jednotka osvětlení .....	41
Obr. 23: Instalace kamery a osvětlení .....	42
Obr. 24: kobot UR3 s efektozem .....	43
Obr. 25: Kontrolér robota UR3 spolu s elektronikou .....	44
Obr. 26: Definování roviny .....	45
Obr. 27: Volba typu programu .....	46
Obr. 28: Program s komunikačním skriptem.....	47
Obr. 29: Volba typu úlohy.....	48

Obr. 30: Nastavení kamery a spojení s robotem .....	49
Obr. 31: ShapeTrax3 pro kalibraci nástroje .....	50
Obr. 32: Kalibrační pohybový vzorec .....	51
Obr. 33: Výsledek kalibrace .....	52
Obr. 34: Posun objektu od Master pozice .....	53
Obr. 35: Struktura programu .....	54
Obr. 36: Program pro 1.úlohu .....	55
Obr. 37: Výběr funkce Height Measurement .....	57
Obr. 38: Výsledek měření – hloubková mapa.....	58
Obr. 39: Řetězec výstupních dat .....	59
Obr. 40: Program a skript pro 2.úlohu .....	60

## Seznam zkratek

TCP	Tool Center Point
UI	User Interface
ToF	Time of Flight
CCD	Charged Coupled Device
CMOS	Complementary Metal Oxide Semiconductor
RPP	Random Pattern Projection
GIF	Graphics Interchange Format
PNG	Portable Network Graphics
RGB	Red, Green, Blue
HSI	Hue, Saturation, Intensity
JPEG	Joint Photographic Experts Group
SIFT	Scale-Invariant Feature Transform
SURF	Speeded Up Robust Features
IFR	International Federation of Robotics
LED	Light-Emmiting Diode
DC	Direct Current
PLC	Programmable Logic Controller



# Úvod

Pro lehčí práce na výrobních linkách s nízkým zatížením jako je manipulace s materiálem se používají většinou menší roboty, které mohou pracovat poblíž člověka – koboty. K efektivnímu plnění těchto úkolů je zapotřebí kromě různých efektorů a souvisejících periferních zařízení také systémů vidění, které umožní provádět vizuální kontrolu a rozpoznávání objektů. Jaký typ systémů vidění je však nejvhodnější pro nejširší škálu kolaborativních aplikací? Pro zodpovězení této otázky je důležité pochopit rozdíl mezi třemi hlavními typy systémů vidění, kterými jsou 2D, 2.5D a 3D.

2D systémy jsou nejlevnější, ale také nejméně univerzální, jsou schopny určit délku a šířku (osy X a Y), ale nedokážou určit výšku, což omezuje počet aplikací, které mohou podporovat. Dalším problémem těchto systémů je chybné určení polohy vlivem projektivního zkrácení (různá výška objektů).

Na druhém konci spektra jsou 3D systémy vidění. Tyto systémy poskytují robotovi vizuální informace ve všech třech osách, včetně otáčení objektu. 3D kamery nabízejí vysoce výkonné funkce, ale jsou nejdražší. Bývají to také složitá zařízení, která k integraci do robotů vyžadují hodně zkušeností. Kromě toho mají 3D kamery problémy se spolehlivostí, kvůli kterým se mnoho výrobců zdráhá tuto technologii přijmout.

2,5D systémy zaujímají zlatý střed mezi 2D a 3D, a to jak z hlediska nákladů, tak i možností. Narozdíl od 2D systémů jsou schopné určit výšku objektů, jsou tedy ideální pro scénáře, v nichž se objekty liší výškou, a pro případy, kdy je nutné předměty skládat na sebe. Kamery 2,5D jsou podstatně levnější než jejich 3D protějšky a nabízejí více možností než 2D kamery, takže se ideálně hodí pro širokou škálu kolaborativních aplikací. 2D a 3D systémy mají své místo ve spoustě aplikací, ale první z nich nabízejí omezenou funkčnost a implementace 3D systémů je drahá a pro většinu úloh možná až zbytečná.

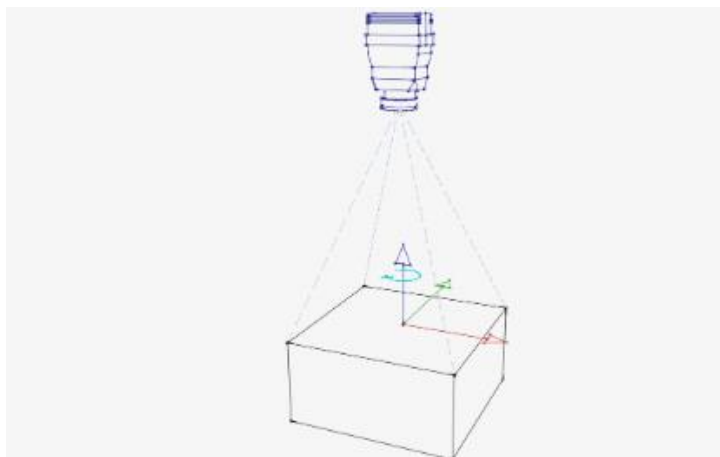
Stejně jako koboty nabízejí způsob automatizace bez nákladů a složitostí spojených s tradičními průmyslovými roboty, systémy vidění na bázi 2,5D nabízejí způsob, jak zavést pokročilé strojové vidění do automatizačního nasazení bez všech nákladů a složitostí spojených se systémy vidění na bázi 3D.

# 1. Principy počítačového vidění ve 2,5D a 3D

## 1.1 2,5D vidění

Systém 2,5D vidění v robotice označuje systém, který zachycuje a zpracovává vizuální informace pro zjednodušené trojrozměrné vnímání prostředí, obvykle pomocí hloubkových map. Na rozdíl od systémů 3D vidění, které vytvářejí kompletní objemovou reprezentaci scény, poskytují systémy 2,5D vidění praktičtější kompromis mezi 2D a 3D a nabízejí informace o hloubce bez výpočetní náročnosti zpracování kompletních 3D obrazů. [5] [6]

Pokud se bavíme o využití 2,5D vidění v technické praxi, tak tyto systémy využívají speciálních kamer a dalšího příslušenství a poté za použití nejrůznějších algoritmů vypočítávají údaje jako jsou výška, tvar, poloha a orientace objektu.



*Obr. 1: 2,5D vidění zachycuje hloubku objektu [5]*

V kontextu automatizace výrobních linek se 2,5D vidění často používá pro detekci a kontrolu kvality výrobků, měření jejich rozměrů ve všech třech osách, sledování pohybu na výrobní lince, orientaci dílu na dopravníku, nebo pro kontrolu správného umístění prvků při sestavování výrobků. Díky vysoké rychlosti a přesnosti jsou vision systémy 2,5D vidění velmi efektivní při zvyšování produktivity a řízení kvality v průmyslové výrobě. [5] [6]

## 1.2 3D vidění

System 3D vidění v robotice umožňuje strojům vnímat prostředí ve třech rozměrech a poskytovat podrobné prostorové informace o objektech a jejich okolí. Na rozdíl od 2,5D systémů, které aproximují 3D informace z hloubkových map. Obraz cílového objektu v systému 3D strojového vidění již není jen plochým obrazem. Nyní je to trojrozměrné mračno bodů s přesnými souřadnicemi, které ukazuje polohu každého pixelu v prostoru. Současně nabízí data v rovinách X, Y a Z a také informace o rotaci (kolem každé z os). [6] [7]

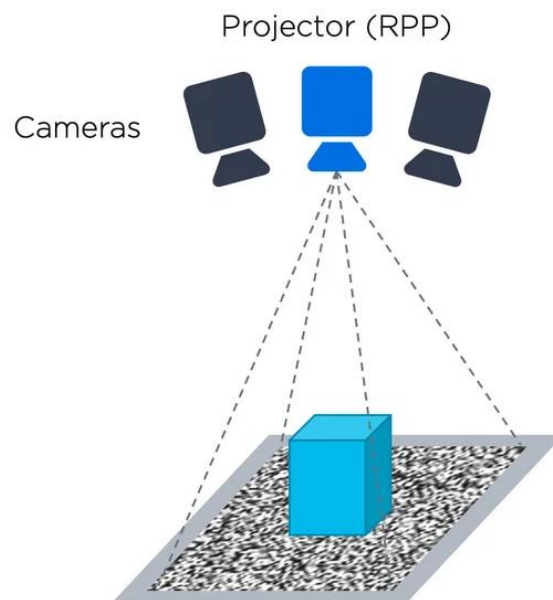
Primárním senzorem používaným v systémech 3D vidění je obvykle kamera snímající hloubku, například stereo kamera, Time of Flight kamera (ToF) nebo kamera se strukturovaným světlem. Tyto senzory přímo snímají 3D informace měřením vzdálenosti k objektům v prostředí a vytvářejí husté mračno bodů reprezentující scénu. Pořízená data mračna bodů se zpracovávají, aby se odstranil šum, odlehlé hodnoty a nerelevantní body, čímž se zajistí čistá a přesná reprezentace prostředí ve všech třech rozměrech. Existují čtyři hlavní principy, na kterých 3D systémy strojového vidění fungují: laserová triangulace, stereo vidění, Time of Flight (ToF) a strukturované světlo. Tři z těchto principů jsou založeny na prostorové oblasti, čtvrtý – ToF je princip založený na časové oblasti. [6] [7]

### 1.2.1 Stereo vidění

Technika napodobující lidské vidění, využívá dvojici kamer k zaznamenání stejného 2D pohledu na cílový objekt ze dvou různých úhlů. Při znalosti pevné relativní polohy obou kamer software porovnává odpovídající body na obou plochých snímcích, identifikuje rozdíly a pomocí triangulace vytváří úplné 3D mračno bodů. Výhody jsou rychlé pořizování snímků a velké zorné pole, nevyžaduje speciální osvětlení ani vestavěný zdroj světla. Stereo vidění je proto vhodné pro měření na velké vzdálenosti ve venkovním prostředí. Bohužel kvalita obrazu není srovnatelná s kvalitou lidského zraku. U objektů s texturou se ke stanovení bodových korespondencí (hledání stejného bodu na snímku z obou kamer) obvykle používají metody korelace obrazu. Jedná se o procesorově náročnou úlohu, která obvykle vyžaduje dlouhou dobu výpočtu. Hlavním problémem stereo vidění je stanovení korespondence mezi body na dvou nebo více snímcích z kamery. Charakteristické body, jako jsou rohy krychle, mohou být identifikovány jako odpovídající body, ale problémy

vznikají u bodů, které se nacházejí uprostřed hladkého povrchu bez textury. Tyto problémy mohou vést k řídce zaplněnému 3D obrazu (tj. menšímu počtu datových bodů v mračnu bodů). Začleněním laserové nebo strukturované světelné projekce pro přidání textury lze tyto problémy zmírnit. [7]

U stereo vidění je možné provést modifikaci – Aktivní stereo vidění je podobné klasickému, s tím rozdílem, že je rozšířeno o projektor náhodných obrazců (RPP). RPP je často laser s rozbíhavým, pseudonáhodným vzorem, který pomáhá stanovit shodu pixelů u objektů s malou nebo žádnou texturou. Promítaný vzor totiž účinně vyplňuje prosté oblasti obrazu texturou, čímž zmírňuje problém korespondence. Zavedení RPP však kromě toho, že celý systém komplikuje a prodražuje, neomezuje stávající problémy základního stereo vidění. [7]



Obr. 2: Princip Aktivního stereo vidění [7]

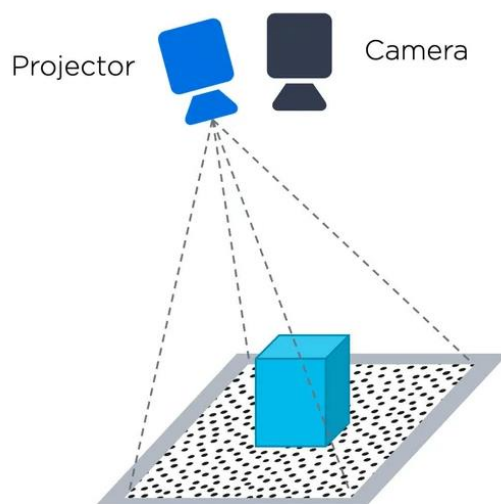
### 1.2.2 Laserová triangulace

Laserová triangulace je jednou z nejčastěji používaných technik 3D zobrazování a je nasazována v široké škále interiérových i exteriérových aplikací. Je jednoduchá, levná, rychlá a vysoce přesná. Technika laserové triangulace nebo laserového profilování zahrnuje pohyb cílového objektu paprskem (linií) laserového světla. Kamera umístěná v nastaveném úhlu k laserovému zářiči zaznamenává obraz laserového paprsku, jak je vychylován geometrickým tvarem objektu, čímž je zachycen přesný profil objektu v určitém časovém

okamžiku. Zaznamenávají se po sobě jdoucí profily objektu a porovnáním výškových odchylek od přímé, nedeformované laserové linie se vytvoří trojrozměrné mračno bodů. Při zpracování obrazů se používají algoritmy triangulace, které určují vzdálenosti mezi různými body na povrchu objektu a informace jsou pak použity k vytvoření detailního 3D modelu objektu. Laser je výkonný monochromatický zdroj světla a použitím pásmového filtru před měřicí kamerou lze systém učinit velmi odolným vůči okolnímu světlu. Toto nastavení neposkytuje informace o barvě a laserová triangulace může mít také problémy se získáváním dat z lesklých a tmavých povrchů objektů. Hlavní nevýhodou laserové triangulace je, že je relativně pomalá. Protože se jedná pouze o jednu linii, musí se nějakým způsobem skenovat napříč scénou, aby se vytvořil úplný obraz. Buď musí celý objekt projít laserovou linií konstantní rychlostí, nebo musí snímač "přejet" přes objekt. Potřeba zajistit stabilní způsob řízení pohybu pro snímání objektu může představovat problém, ačkoli vzhledem k této potřebě je proto laserová triangulace pravděpodobně nejvhodnější metodou pro 3D zobrazování objektů na pohybujícím se dopravníku. [6] [7]

### **1.2.3 Strukturované světlo**

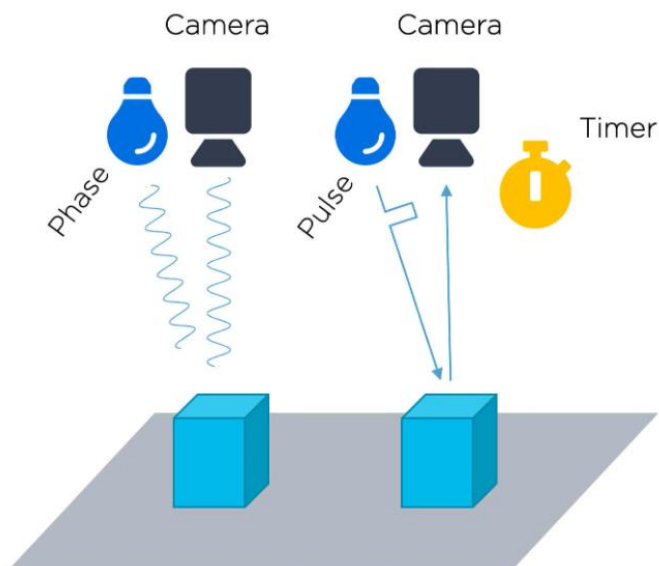
Technika strukturovaného světla je podobná laserové triangulaci, ale je rozšířena na celé pole. Poskytuje tedy celý 3D obraz objektu, a nikoli pouze jedinou linii průřezu. Zaznamenává geometrické zkreslení známého osvětlovacího obrazce promítaného na povrch statického cílového objektu. Na povrch cílového objektu se promítá většinou mřížkový vzor složený z několika tisíc čar vysoce kontrastního světla. Vzor je snímán kamerou pod nastaveným úhlem. Určí se zkreslení původního plochého vzoru, což umožní získat velmi přesné 3D zobrazení mračna bodů objektu. Strukturovaný vzor nemusí být nutně složen z čar, může mít podobu bodů apod. V každém případě musí poskytovat jedinečné informace pro každou oblast obrazu. Při použití pouze jednoho vzoru je strukturované světlo podobné aktivnímu stereu, ale bez druhé kamery. Protože je však vzor strukturovaný, a nikoli náhodný, je extrakce dat obecně méně náročná na procesor a strukturovaný vzor se vyhýbá problému korespondence. Negativní stránkou je, že prostorové strukturované světlo s jedním vzorem má stejné problémy jako stereo vidění, pokud jde o rozlišení a přesnost. Systémy strukturovaného světla mají obvykle také potíže, pokud jde o zrcadlové nebo lesklé povrchy nebo velmi tmavé a absorpční povrchy. Tyto povrchy obvykle vedou k chybějícím datům a chybám měření. [7]



Obr. 3: Princip strukturovaného světla [7]

### 1.2.4 Time-of-Flight

Někdy také nazývané jako systémy LIDAR nebo laserové radary, využívají k 3D zobrazování spíše časovou než prostorovou oblast. Měřením časového zpoždění mezi emitovaným laserovým světlem a odraženým laserovým světlem od povrchu objektu můžeme získat přesné měření vzdálenosti.



Obr. 4: Princip Time-of-Flight [7]

Na obr.4 můžeme vidět dva rozdílné přístupy:

- Systémy pulzního laserového světla měří časové zpoždění mezi vyslaným a přijatým laserovým impulsem, a protože čas je přímo úměrný vzdálenosti, určuje se relativní poloha.
- Fázové systémy pracují tak, že modulují sinusovou vlnu na emitovaném laserovém paprsku a měří fázový rozdíl mezi vlnou emitovanou zařízením a vlnou odraženou od objektu. Integrace v delším časovém úseku ve srovnání se systémem založeným na impulzech obvykle poskytuje lepší přesnost. Nevýhodou je menší rozsah měření, zvýšená citlivost na okolní osvětlení jako je sluneční světlo a odrazy.

Time-of-Flight kamery představují relativně novou technologii pro snímání 3D obrazů v reálném čase. Tyto kamery jsou velmi rychlé a snímají celý obraz namísto pouhého bodu. [7]

K osvětlení scény se používá blízké infračervené světlo (NIR) a fáze světla, které se od scény odrazí a vrátí se zpět, se měří pomocí speciálního snímače. Fázový posun je úměrný uražené vzdálenosti, což nám umožňuje měřit vzdálenosti od každého pixelu na snímku k odpovídajícím bodům na scéně. Kamery ToF jsou složité na výrobu a často se dodávají s nízkým prostorovým rozlišením, přičemž rozlišení VGA (640×480) je považováno za špičkové. Kvůli složitosti konstrukce a samotného principu měření mají kamery ToF menší teoretickou přesnost než systémy se strukturovaným světlem na vzdálenosti do deseti metrů. Nad deset metrů začínají být systémy ToF lepší. [7]

### **1.2.5 Časově kódované strukturované světlo**

V 3D technice zobrazování časově kódovaným strukturovaným světlem se kombinují přístupy v prostorové a časové oblasti. Namísto promítání jediného obrazce se promítá řada jedinečných obrazců a kamera pořídí několik snímků celé kódované řady. Intenzity pozorované v jednotlivých pixelech v různých časech se používají k určení korespondence mezi jednotlivými pixely v kameře a projektoru. Časově kódované strukturované světlo dosahuje až 100x vyšší přesnosti než jiné techniky, je velmi žádaným řešením pro 3D strojové vidění. Bohužel pořízení několika snímků vyžaduje více času a cílový objekt i





## 2. Obecné principy počítačového zpracování obrazu

Následující kapitola obsahuje stručný přehled známých metod pro jednotlivé kroky zpracování obrazu.

Počítačové vidění spočívá v napodobení schopnosti lidského oka pomocí technických prostředků. Nastává zde mnoho problémů jako je nutnost častého zpracování velkého množství dat, šum v obrazu, ztráta informací při převodu 3D scény do 2D atd. [1]

Posloupnost základních kroků při zpracování obrazu je následující:

- Snímání a digitalizace obrazu
- Předzpracování
- Segmentace obrazu
- Popis objektů
- Klasifikace

### 2.1 Snímání a digitalizace obrazu

Základem zpracování a rozpoznávání obrazu je nasnímání obrazu reálného světa a jeho následný převod do digitální formy vhodné pro další zpracování v počítači. První článek řetězce při snímání obrazu je optický systém tvořený objektivem. Objektiv je charakterizovaný svou ohniskovou vzdáleností, světelností a zorným úhlem a bývá tvořen soustavou čoček s různými funkcemi. Jeho úkolem je soustředit dopadající energii (fotony) na snímač, kde se poté vytváří obraz. [1]

Obrazové snímače jsou většinou založeny na principu světlocitlivosti polovodičů, díky které mohou přeměňovat dopadající světlo na elektrický náboj. Jsou to v podstatě matice polovodičových buněk citlivých na světlo, v nichž se po dobu expozice akumuluje elektrický náboj úměrný osvětlení buňky. V současnosti jsou nejpoužívanější snímače CMOS kvůli jejich nízké spotřebě a vysoké rychlosti. Dále je možné se setkat se staršími snímači typu

CCD, které mají výhody ve velké citlivosti a nízkém šumu, naopak ztrácejí v rozsahu intenzit a vzájemném ovlivňování nábojů v sousedních pixelech, tzv. blooming. CMOS snímače mají větší rozsah intenzit, ale také o řád větší šum v porovnání s CCD. CMOS snímače obsahují jak snímač, tak A/D převodník na jednom čipu. Oba typy pracují na principu vnitřního fotoelektrického jevu. [1]

Při převodu z analogové do digitální formy se signál vzorkuje a kvantuje. Vzorkování dat představuje převod spojitého signálu do matice  $M \times N$  obrazových bodů. Kvantování představuje převod spojitě jasové úrovně původního analogového signálu do  $K$  intervalů. Díky kvantování získá digitální obraz celočíselné jasové hodnoty. Při vzorkování dat je nutné dodržovat Shannonův – Nyquistův teorém. Ten říká, že vzorkovací frekvence musí být alespoň dvakrát větší než největší frekvence v původním signálu, jinými slovy interval vzorkování musí být menší nebo roven polovině nejmenšího detailu v obrazu. Vzorkovací mřížka použitá při vzorkování může být čtvercová, šestiúhelníková nebo trojúhelníková. Pokud není interval kvantování dostatečně jemný, dochází ke vzniku falešných obrysů – pro lidské oko začínají být patrné, pokud má digitální obraz méně než padesát jasových úrovní. Volba rozlišení obrazu je velmi zásadní, protože při nízkém rozlišení ztratíme informace o detailech, a naopak při příliš velkém rozlišení bude stoupat výpočetní náročnost u dalšího zpracování. [1] [4]

Dalším důležitým nositelem informace je barva. Existuje několik barevných modelů, které se liší podle složek. [3] [4]

- RGB

Tento barevný model je používán v zobrazovacích zařízeních jako jsou televize a počítačové monitory. Jeden pixel tvoří tři barevné složky – červená, zelená a modrá. Výsledná barva vznikne složením těchto tří složek. Pokud jsou hodnoty jednotlivých složek stejné, jedná se o odstíny šedi.

- Šedá škála

V určitých případech zpracování obrazu není barevná informace nutná a postačí nám odstíny šedi. Nejčastěji se používá osm bitů hloubky, tedy 256 různých odstínů. Využitím pouze odstínů šedi je možné uspořít značné množství dat v porovnání s barevnými prostory.

- CMY

CMY pro Cyan, Magenta, Yellow (azurová, purpurová, žlutá). Je to barevný model založený na subtraktivním míchání barev. Na rozdíl od RGB se zde barvy neskládají, ale odečítají od původně bílé. Čím je hodnota dané složky vyšší, tím více se blížíme k černé barvě. Tento barevný model se používá u reprodukčních zařízení jako jsou tiskárny.

- HSI

Složky tohoto barevného modelu nejsou tvořeny základními barvami, ale jejich vlastnostmi: Hue, Saturation a Intensity (barva, sytost a jas). Barva je udávána jako úhel v rozmezí 0-360° (0 resp. 360° = červená, 120° = zelená, 240° = modrá). Barvy tvoří uzavřený kruh. Pro určení požadované barvy stačí zadat libovolný zlomek úhlu. Sytost určuje množství přidané bílé složky a jas určuje, kolik světla daná barva odrazí. [3] [4]

### *Grafické formáty uložení obrazu*

Kvůli velkému množství obrazových dat se v počítačích používají různé techniky komprese. Tato komprese může být bezztrátová nebo ztrátová. Bezztrátové algoritmy používají efektivní kódování, kdy se v obrazu hledají podobnosti sousedních pixelů, např. každou posloupnost opakujících se čísel zapíšeme pouze jednou a k ní kolikrát se opakuje (run-length encoding), dále speciální kódy pro skupiny hodnot, které se často opakují (LZW algoritmus), atd. Tyto techniky využívají formáty jako jsou GIF nebo PNG. Naopak ztrátové algoritmy využívají toho, že lidské oko není tak citlivé a je tedy možné určitě informace v obrazu vypustit bez viditelného rozdílu. Zástupcem ztrátového formátu je JPEG, jež pro snížení objemu dat odstraňuje část barevné informace a vysoké frekvence. Pro strojové vidění je ale zapotřebí co nejvíce obrazových dat, tedy v bezztrátové kompresi. [3] [4]

## **2.2 Předzpracování obrazu**

Po úspěšné digitalizaci následuje předzpracování. Obraz může být různě zkeslen například kvůli nevhodným podmínkám při snímání. Existuje velké množství metod, které značně usnadňují další práci a analýzu obsahu obrazu, identifikaci objektů nebo jen zvýrazňují

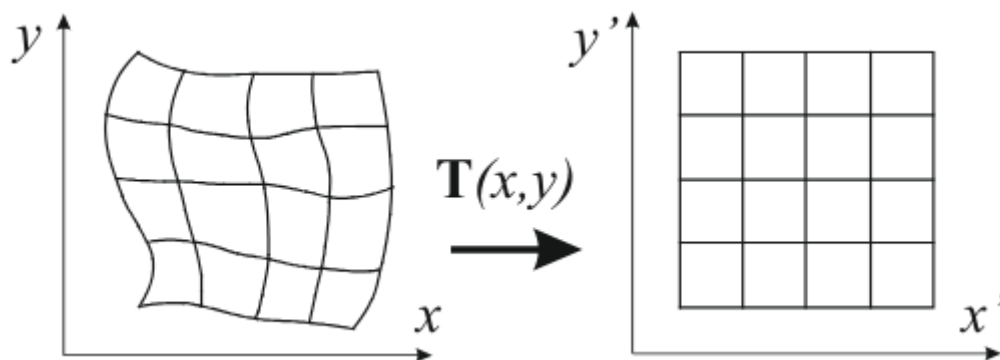
důležité rysy. Pro základní analýzu se používá obrazový histogram, který reprezentuje početní zastoupení jednotlivých jasových úrovní v digitálním obrazu. [2]

Mezi základní metody předzpracování obrazu se řadí:

- Geometrické transformace
- Jasové transformace
- Filtrace
- Ostření

### *Geometrické transformace*

Jedná se o zvětšení, posun, rotaci, zkosení či odstranění geometrického zkreslení



*Obr. 6: Korekce geometrického zkreslení [15]*

Geometrická transformace je vektorová funkce  $T$ , která mapuje polohu bodu se souřadnicemi  $(x, y)$  na nové souřadnice  $(x', y')$ . Pro rovinnou transformaci je vektorová funkce  $T$  (2.1) složena ze dvou dílčích transformací  $T_x$  (2.2) a  $T_y$  (2.3). [2] [4]

$$T = (T_x, T_y) \quad (2.1)$$

$$x' = T_x(x, y) \quad (2.2)$$

$$y' = T_y(x, y) \quad (2.3)$$

Operace jako zvětšování, posouvání, otáčení nebo zkosení se provádí podle předem známé

transformace, narozdíl od odstraňování geometrických zkreslení (Obr.6), u těch je většinou příslušná transformace odhadována z obrazu. Problémem geometrických transformací je, že digitální obraz poskytuje pouze vzorky v diskrétní mřížce, pokud bychom chtěli provést například otočení kolem počátku o konkrétní úhel  $\alpha = 30^\circ$  (2.4) (2.5), nové pozice pixelů by odpovídali bodům mimo diskrétní mřížku. Z tohoto důvodu je nutné provést interpolaci, nejběžnějšími metodami jsou metoda nejbližšího souseda, bilineární interpolace a bikubická interpolace. U metody nejbližšího souseda se přiřadí neznámé obrazové funkci hodnota nejbližšího známého pixelu. Další zmíněné interpolační metody poskytují značně kvalitnější výsledky za cenu zvýšené výpočetní náročnosti. [2] [4]

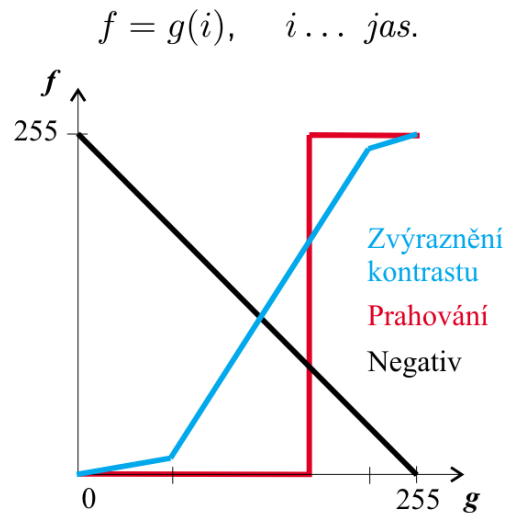
$$x' = x \cdot \cos(\alpha) - y \cdot \sin(\alpha) \quad (2.4)$$

$$y' = x \cdot \sin(\alpha) + y \cdot \cos(\alpha) \quad (2.5)$$

Při odstraňování geometrických zkreslení (projekce 3D objektů do 2D, čočky kamer atd.) hledáme takovou transformaci, která vede z daného souřadnicového systému do vzorového (máme obraz, u kterého předpokládáme, že je správný). Dalším problémem, který u transformací nastává je aproximace jasové funkce. Tato funkce hledá hodnotu jasu v celočíselné pozici, která nejvíce sedí pro novou neceločíselnou polohu  $x'$ ,  $y'$ . [2] [4]

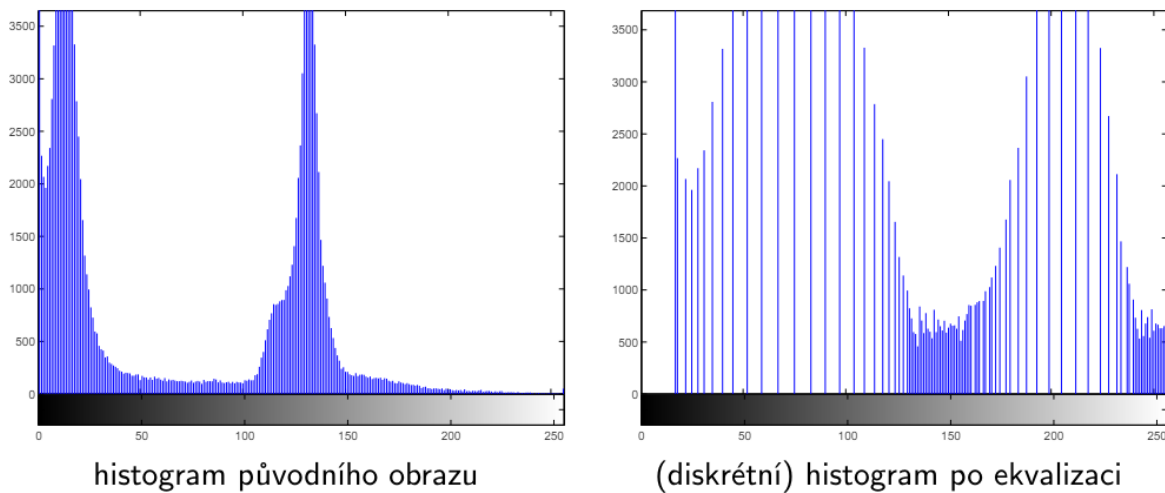
### *Jasové transformace*

Jasovou transformaci představuje opět transformační funkce, která mění hodnoty vstupní jasové funkce a vytváří tak funkci novou. Může se jednat o jasovou korekci, kterou je vhodné použít, pokud je snímek zatížen chybami jako je nadměrné osvětlení scény, přítomnost prachových částic na optice a snímacím prvku nebo pokud není snímací prvek citlivý ve všech bodech rovnoměrně. [2] [4] [15]



Obr. 7: Transformace jasové stupnice [15]

Jednou z používaných technik je ekvalizace histogramu, kde je snaha o rovnoměrné zastoupení všech jasových hodnot. Umožňuje tak v obrazu s celkově vysokým kontrastem zvýraznit špatně rozpoznatelné detaily s nízkým kontrastem, což se používá u málo kontrastních rentgenových snímků. [2] [4] [15]



Obr. 8: Ekvalizace histogramu [15]

### Filtrace šumu

Šum představují náhodné změny obrazové funkce, které jsou většinou nežádané. Zdroji šumu v obrazu jsou např. změny v osvětlení, šum světlocitlivého senzoru, důsledky

komprese obrazu a konečná přesnost souvisejících výpočtů. Často využívanou metodou je filtrace pohybujícím se okénkem za použití operace konvoluce (2.6). Nová hodnota se počítá z hodnot v malém okolí aktuálního pixelu. Toto okolí je reprezentováno filtračním okénkem (konvolučním jádrem)  $h$  udávajícím váhu hodnot obrazové funkce v okolí součtu. Je používáno několik variant těchto okének. [2] [16]

$$g(x, y) = (h * f)(x, y) = \sum_{(m,n) \in O} h(x + m, y + n) f(m, n) \quad (2.6)$$

Metoda prostého průměrování pouze průměruje hodnoty původních jasů ve zvoleném okolí, celé okolí bodu může mít stejnou váhu nebo se např. střednímu pixelu přiřadí dvojnásobná váha. Nevýhodou je rozmazání hran. Další z variant je metoda rotující masky, ta oproti prostému průměrování nerozmazává hrany a mírně ostří. Masky rotuje kolem aktuálního pixelu (celkem 8 poloh) a z každé masky se vypočte rozptyl jasů, poté je vybrána ta s nejmenším rozptylem a z této masky se spočte aritmetický průměr, který je výslednou hodnotou jasu pro nový pixel. Filtrování pomocí mediánu je také jednou z metod pro filtrace šumu. Snižuje rozmazání hran. Nevýhoda je, že obdélníková maska způsobuje porušení tenkých čar, z toho důvodu se používá jiný tvar okolí. [2] [16]

### *Detekce hran*

Hrany jsou místa v obrazu, kde se náhle a značně mění hodnota jasu, což naznačuje přítomnost objektů nebo hranic mezi objekty. V místě hrany bude tedy velká hodnota derivace jasové funkce. Směr největší změny (kolmo na hranu) vyjadřuje gradient. K aproximacím derivací se používají operátory využívající konvoluci s vhodným jádrem. Pokud se při detekci hran uvažuje 8-okolí (po 45°), existuje osm variant těchto masek, každý směr se vypočte zvlášť a ten s největší odezvou udává směr hrany. Základním operátorem je Robertsův operátor, je velmi citlivý na šum, protože okolí použité pro aproximaci je malé (obecně platí, že čím větší maska, tím větší odolnost proti šumu). Dalšími příklady operátorů jsou operátor Prewittové, Sobelův, Robinsonův, Cannyho. Mezi nejlepší se řadí Cannyho hranový detektor, který dále aplikuje prahování s hysterezí. Práhování má za účel odlišit hrany od šumu a prahování s hysterezí znamená pokročilejší metodu, kdy se nastaví dva

různé prahy (T1 a T2), mezi kterými může gradient kolísat. Pokud hodnota gradientu pixelu leží nad prahem T2, tak je označen jako hranový. Pokud vyhodnocujeme bod, jehož hodnota je mezi prahy T1 a T2, pak je uvažován jako hrana jediné pokud sousedí s bodem, který už byl jako hrana označen dříve. Nevýhoda operátorů využívajících aproximaci prvních derivací diferencemi v malém okolí je velká závislost na konkrétním obrázku, velikost masky musí totiž odpovídat velikosti detailů v obraze. Spolehlivějším řešením je využití metod založených na průchodech druhé derivace nulou. [2] [17]

$$\begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} \quad \begin{pmatrix} -1 & -1 & -1 \\ 1 & -2 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Prewittové operátor, Robinsonův operátor (pro jeden směr)

### *Ostření*

Probíhá zvýrazněním vysokých frekvencí. K tomu je vhodné použít Laplacián, neboť tyto frekvence zdůrazňuje a je všesměrový. Nejdříve se vypočte konvoluce původního obrazu s Laplaceovým jádrem a poté se vypočtená konvoluce sečte s původním obrazem. [4] [17]

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

Diskrétní Laplacián

## **2.3 Segmentace**

Jedním z nejtěžších kroků zpracování obrazu je segmentace. Cílem segmentace je rozdělení obrazu na oblasti, které mají společné vlastnosti, typicky oddělení popředí od pozadí. Základním způsobem segmentace je metoda prahování. Spočívá v hodnocení jasů každého pixelu. Principem metody je najít takovou hranici v histogramu, aby pixely pod touto hranicí tvořily pozadí a pixely s jasem nad touto hranicí tvořily popředí. S rostoucím podílem šumu v obraze je stanovení správného prahu obtížnější, v extrémních případech nemožné. Pokročilejší metodou je například watershed (Povodí). Tato metoda segmentace je

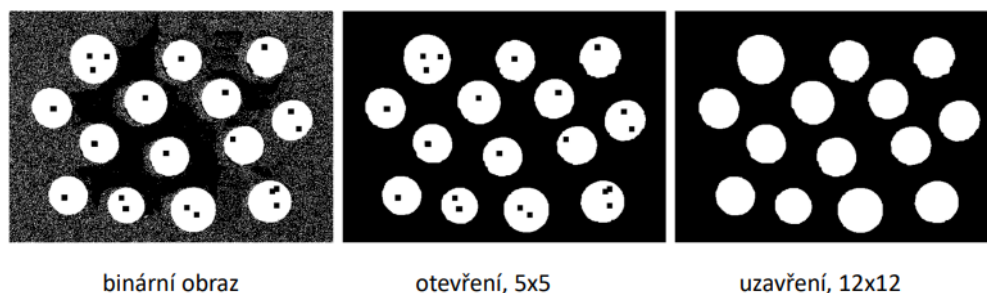


inspirována topografií povrchu a říčním systémem. Obraz je chápán jako topografická mapa – jas pixelu je chápán jako nadmořská výška a hrany chápány jako horské hřbety. Algoritmus pak simuluje zaplavování povrchu a rozděljuje ho na segmenty podle vrcholů a minimálních údolí. [3] [4]

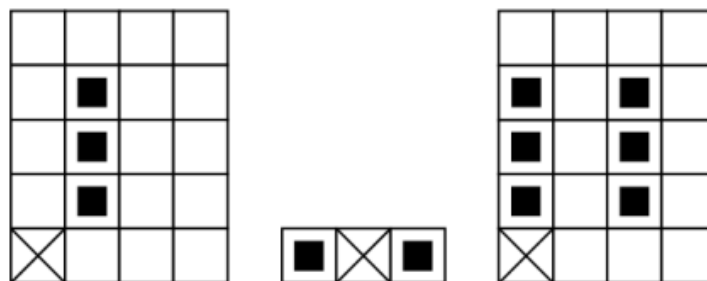
Po segmentaci obrazu prahováním nebo jinou metodu obvykle v obrazu zůstávají pixely představující zbytkový šum. V takových případech se aplikuje matematická (binární) morfologie. Morfologická transformace představuje relaci mezi obrazem a strukturálním elementem. Tento element je systematicky posouván po obrazu a výsledek transformace je zapisován do výsledného binárního obrazu. [3] [4]

Mezi základní operace binární matematické morfologie patří:

- Dilatace – zvětšuje objekty, používána pro zaplnění malých děr a úzkých zálivů
- Eroze – zjednoduší struktury objektů, složitější objekt se rozdělí na jednodušší
- Otevření – eroze následovaná dilatací, rozděljuje objekty spojené úzkou šíjí
- Uzavření – dilatace následovaná erozí, spojuje blízké objekty, vyhlazuje tvar



Obr. 9: Využití binární morfologie [23]



Obr. 10: Výsledek dilatace [23]

Po segmentaci a následných úpravách je vhodné provést identifikaci oblastí. To se provádí metodou barvení. Po aplikování této metody je každá oblast v obraze označena unikátním číslem. Po obarvení oblastí je možné vypočítat těžiště oblastí a zjistit tak souřadnice každého objektu. [3] [4]

0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	1	1	0	1	0
0	1	1	1	1	1	1	0	0	1	0	0	1	0
0	0	0	0	1	0	1	0	0	0	0	0	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	1	1	1	1	1	1	1	1	1	0
0	1	1	0	0	0	1	0	1	0	0	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0

### Binární obraz po segmentaci

0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	2	2	0	1	0
0	1	1	1	1	1	1	0	0	2	0	0	1	0
0	0	0	0	1	0	1	0	0	0	0	0	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	1	1	1	1	1	1	1	1	1	0
0	3	3	0	0	0	1	0	1	0	0	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0

### Výsledek

Obr. 11: Barvení oblastí [22]

Mimo binární morfologie se využívá také šedotónová morfologie. Tu je možné použít na šedotónové obrazy se zachováním šedé intenzity pixelů. [3] [4]

## 2.4 Popis objektů

Popis objektů je proces získávání příznaků ze segmentovaných dat. Tyto příznaky pak dále slouží pro klasifikaci objektů, je nutné tedy vystihnout jejich charakteristické rysy. Popis objektů může být založený například na číselných charakteristikách jako je velikost, kompaktnost, průměrná jasová úroveň, obvod apod. Jednou z takových metod je granulometrie. Je to metoda používaná pro analýzu velikosti a tvaru částic nebo objektů. Nejběžnější způsob provedení je použití série morfologických erozí s různými velikostmi

strukturálního elementu. Zjednodušeně se dá říci, že granulometrie je postupné prosívání sítem s rostoucí velikostí děr. Začínáme s hromadou kamenů (granulí) o různých velikostech a postupně zjišťujeme kolik kamenů patří do jednotlivých tříd daných velikostí, výsledkem je diskretní funkce (granulometrické spektrum), která udává velikost děr v sítu (nezávislá proměnná) a počet kamenů příslušné velikosti (závislá proměnná). Další metodou popisu může být například řetězový (Freemanův) kód. Ten popisuje hranici objektu řetězcem symbolů s určenými směry, konkrétně čtyři symboly pro 4-okolí nebo osm symbolů pro 8-okolí. Freemanův kód je invariantní vůči posunutí, ne však vůči rotaci. Pro nezávislost vůči rotaci existuje modifikace Freemanova řetězového kódu a tou je diferenciální řetězový kód, který reprezentuje relativní změny směru mezi sousedními segmenty. [4] [9]



Obr. 12: Řetězový kód (8 směrů) [9]

## 2.5 Klasifikace

Finálním krokem při zpracování obrazu je klasifikace (rozpoznání obrazu). Často se jedná o zařazení objektů nalezených v obrazu do skupiny předem známých tříd. V těch nejjednodušších případech se objekty řadí do kategorií podle jejich velikosti, v trochu složitějších může jít o řazení do tříd jako kulaté / hranaté. Mezi dnes hojně využívané a zároveň nejnáročnější způsoby rozpoznávání objektů patří umělé neuronové sítě. Ty jsou založeny na učení, kdy se neuronová síť postupně zlepšuje v hledání daného objektu. K učení neuronové sítě je často zapotřebí velké množství vzorových dat a výpočetního výkonu. Mezi další metody patří ty, které jsou založené na detekci klíčových bodů v obraze

tzv. body zájmu (Feature Detection). Metody fungují tak, že se na vstupním a vzorovém obraze najdou body zájmu a každému z těchto bodů je přiřazen určitý deskriptor popisující bod samotný a jeho okolí. Dle podobnosti bodů (jejich deskriptorů) je pak vytvořena množina dvojic odpovídajících bodů, mezi kterými je poté hledána geometrická transformace. Na závěr se provede verifikace prahovým klasifikátorem. [4] [11] [12]

K dispozici je několik různých algoritmů pro detekci klíčových bodů, které se liší ve způsobu identifikace bodů a výpočtu jejich deskriptorů. Jedním takovým algoritmem je SIFT, celý proces detekce klíčových bodů pomocí tohoto algoritmu je navržen tak, aby byl invariantní vůči změnám rotace, měřítka, posunu, šumu, osvětlení nebo kontrastu, což z něj dělá mocný nástroj pro rozpoznávání objektů nebo sledování pohybu. [3] [4] [13]

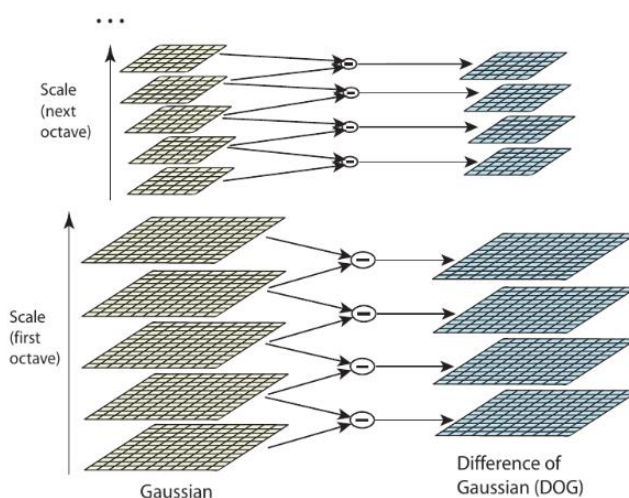
Pro nalezení zajímavých bodů využívá algoritmus tzv. scale-space. Scale-space reprezentuje obraz ve formě série rozmazaných kopií. Každý z obrazů v sérii je získán aplikací určitého rozostření původního obrazu. Toto rozostření je provedeno za použití Gaussovského filtru, který simuluje rozostření v různých měřících. Scale-space  $L(x, y, \sigma)$  snímku s obrazovou funkcí  $I(x, y)$  se vypočte konvolucí s Gaussovou funkcí. [11] [12] [13]

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (2.7)$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (2.8)$$

Algoritmus pro rychlé nalezení extrémů využívá aproximaci LoG (Laplacian of Gaussian) obyčejnými diferencemi – DoG (Difference of Gaussians).

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (2.9)$$



Obr. 13: Scale-space [13]

Dále se postupně projde každý bod a porovná se jeho hodnota s okolními sousedy v aktuální vrstvě a také ve vrstvách nad i pod ním. Pokud je zkoumaný bod maximum/minimum, označí se za možný klíčový bod. Některé tyto kandidáty je nutné vyfiltrovat, protože mohou mít malý kontrast k okolí nebo být na hraně objektu. Diskrétní obrazová funkce nám udává hodnoty jen ve vzorkovaných bodech, může se ale stát, že extrém bude mezi těmito vzorkovanými body (mezipixelové okolí). Hodnoty v mezipixelovém okolí určíme Taylorovou aproximací druhého řádu v blízkém okolí kandidáta. Po filtraci nevhodných bodů přiřadíme těm vhodným velikost a orientaci (kvůli rotační invarianci). To se provádí výpočtem velikosti a směru gradientu z okolí bodu. Výsledný deskriptor klíčového bodu se vytvoří tak, že vezmeme 16x16 okolí daného bodu a opět spočteme velikost a orientaci gradientu. Od orientace gradientu ale tentokrát odečteme orientaci bodu zjištěnou v předchozím kroku. Deskriptor je ve výsledku histogramem těchto gradientů skládajících se ze 128 hodnot. Pro porovnání podobnosti vektorů mezi dvěma obrazy využijeme např. klasickou eukleidovskou metriku. [3] [4] [11]

SURF představuje další z přístupů v detekci klíčových bodů. Tato metoda je vylepšenou verzí metody SIFT. SURF se vyznačuje výrazně vyšší rychlostí, přičemž poskytuje minimálně stejně kvalitní výsledky. Pro urychlení tohoto procesu nahrazuje složité výpočty jejich jednoduššími aproximacemi. [11]

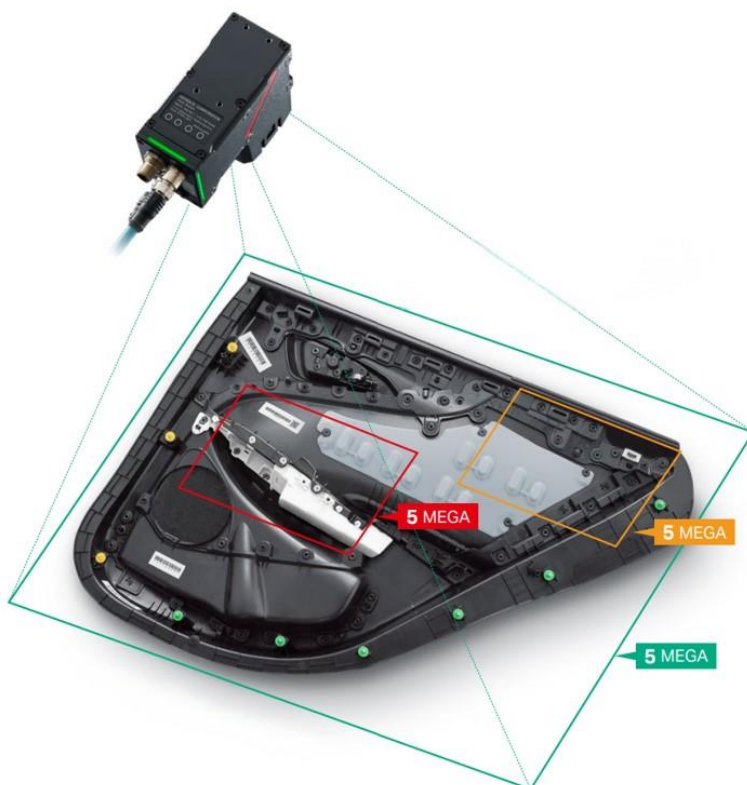
### 3. Vision systém Keyence

Keyence je Japonská firma, která se zabývá vývojem a výrobou zařízení a celkových řešení pro automatizaci jako jsou například senzory, měřicí přístroje, systémy vidění, čtečky čárových kódů a digitální mikroskopy. Firma nabízí ve svém portfoliu několik řad vision systémů.

#### 3.1 Proprietární funkce Vision systémů Keyence

##### *ZoomTrax*

ZoomTrax je proprietární řešení optického přiblížení bez ztráty rozlišení. Řešení spočívá v implementaci několika čoček do jedné kamery což dovoluje rozmanité možnosti v instalaci. Je možné tak volně nastavit zorné pole bez potřeby měnit optiku. Kamery se ZoomTrax technologií firma nabízí v rozlišeních od 1,6 do 15 Mpx. [18]



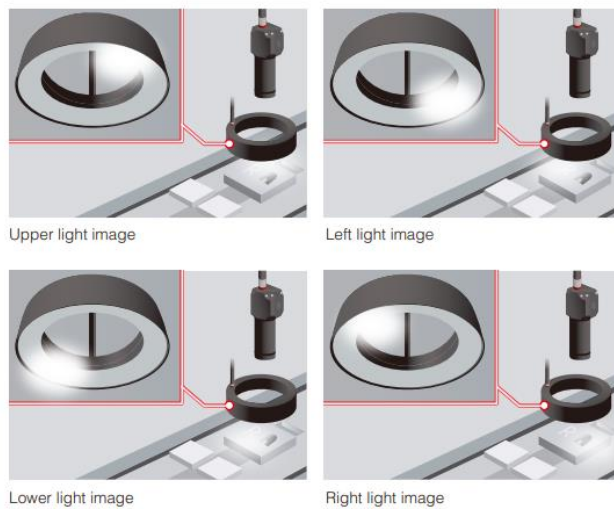
Obr. 14: ZoomTrax [18]

## LumiTrax

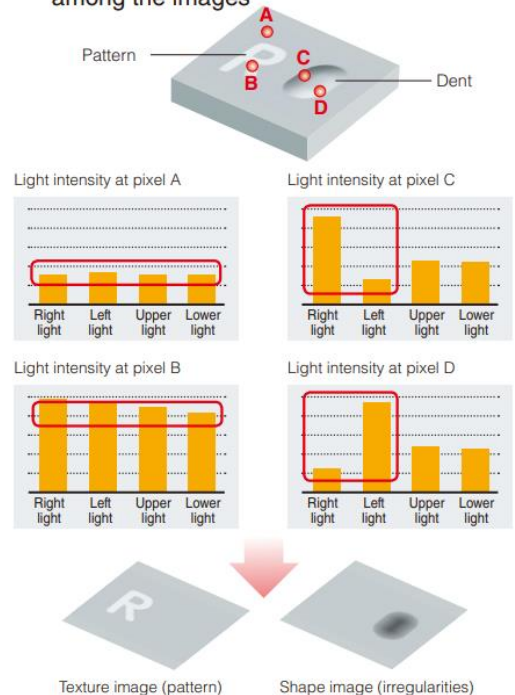
Pro případy, kdy kamerový systém selže v inspekci z důvodů jako jsou nepříznivé povrchové podmínky na zkoumaném objektu (lesk) nebo zkreslení díky odrazu okolního světla vyvinula firma Keyence systém LumiTrax. Tento systém funguje na základě spolupráce vysokorychlostní kamery a speciálního kruhového LED světla, které osvětluje zkoumaný objekt po částech z několika směrů a tím vytváří několik snímků ze kterých složí výsledný obraz na kterém jsou eliminovány odlesky a další neduhy způsobené okolním prostředím. [19]

Systém také disponuje zpětnou vazbou, která využívá fotodiody, jež kontroluje svítivost diod a zachovává tak stabilní detekci objektů i při stárnutí LED.

### 1. Ultra high-speed image capture with lighting from different directions



### 2. Image creation based on the analysis of the difference in light intensity at each pixel among the images



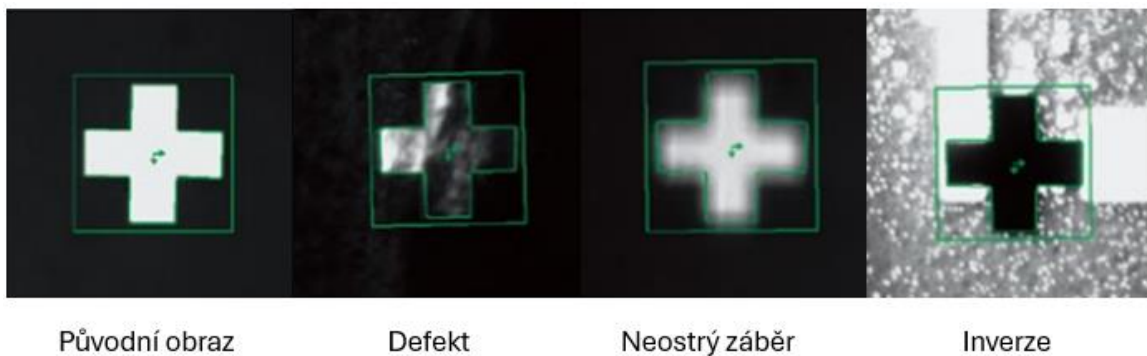
Obr. 15: Princip LumiTrax [19]

Systém LumiTrax nemusí nutně používat pouze klasické bílé LED světlo, ale pro komplikovanější případy, kdy je nutné například rozpoznat text, který splývá s pozadím, je

možnost využít multispektrálního osvětlení, jež disponuje LED světly, která dokážou svítit osmi různými vlnovými délkami barevného světla. [19]

### *ShapeTrax*

Přesněji ShapeTrax 3 je speciální vyhledávací nástroj s vysokou rychlostí a přesností i za špatných podmínek. Algoritmus je schopen najít vzor i pokud se snížil kontrast nebo došlo ke změně velikosti či orientace. Nástroj je používán u robotických aplikací. [20]



Obr. 16: ShapeTrax 3 [20]

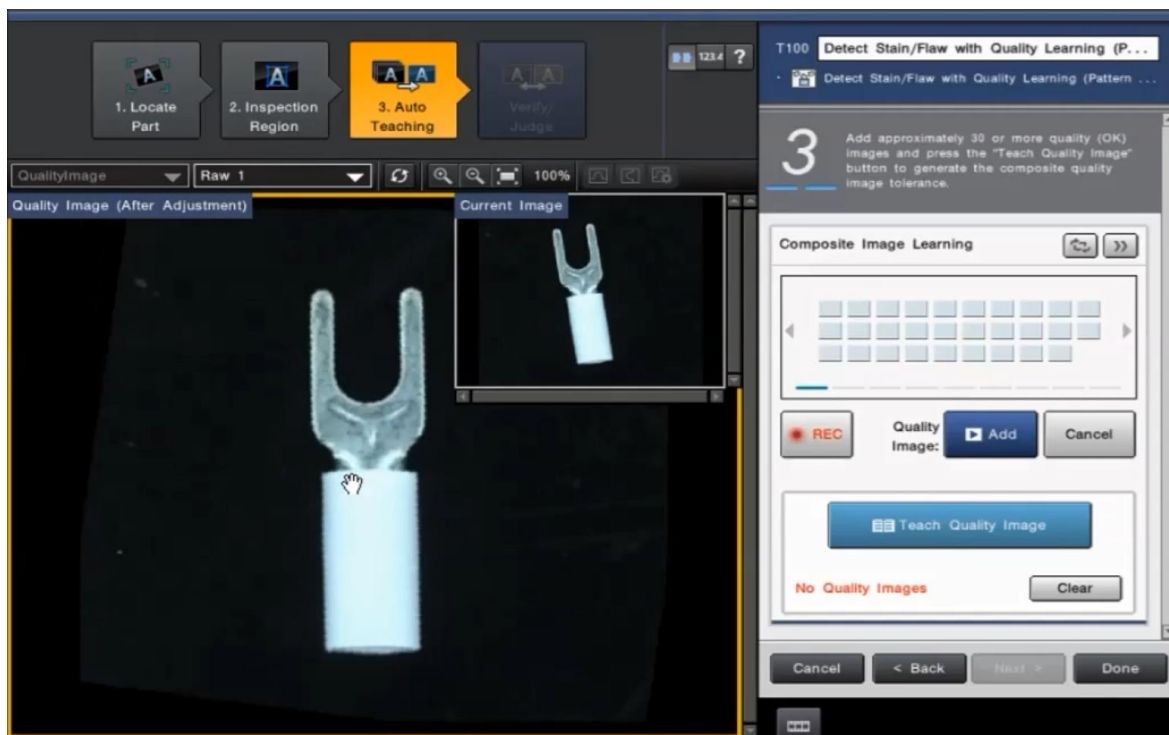
## **3.2 Inspekční nástroje**

Následující kapitola obsahuje výčet funkcí (inspekčních nástrojů) řídicí jednotky CV-X420F a rovněž stručný popis vybraných funkcí.

### *Auto-Teach Inspection*

Tato funkce umožňuje nastavit kontrolu dílu pouze pořízením několika vzorových snímků (vzorky OK) na jejichž základě se kontrolér vision systému „naučí“ co považovat za správný díl bez defektu a naopak.





Obr. 17: Inspekční nástroj Auto-Teach

Pokud bychom chtěli na dílu na obr.17 hledat různé vady jako jsou třeba skvrny nebo špatný tvar, museli bychom bez využití Auto-Teach funkce použít kombinaci několika nástrojů. S funkcí Auto-Teach tato potřeba odpadá. Při nastavování se provede několik kroků. První krok spočívá v definování kontrolovaného dílu, jinými slovy, jaký vzor bude vision systém na snímku hledat, je možné nastavit i úhel o kolik bude moci být díl otočený, dále je nutné definovat oblast ve které bude funkce hledat chybu – na výběr jsou oblasti typu obdélník, kruh, ovál, mezikruží nebo oblouk. Je zde i možnost označit části dílu, které nebudou kontrolovány (pro úsporu času zpracování). Další částí je samotné učení, jak má správný díl vypadat, pro optimální funkci je nutné do systému vložit vzorové snímky alespoň třiceti bezchybných dílů, ale platí zde pravidlo čím více, tím lépe.

### *Color Component*

Tento nástroj změří barevnou informaci zkoumaného objektu. Je možné měřit v barevném prostoru RGB nebo HSB. Na základě hodnot jednotlivých barevných kanálů je možné nastavovat podmínky a omezení a podle toho vyvodit, zda zkoumaný objekt vyhovuje či nevyhovuje.

### *Color-Specific Area*

Slouží ke změření velikosti plochy konkrétní barvy. V principu funguje tak, že uživatel vybere barvu na snímku, systém obraz převede na binární a spočítá počet bílých (černých) pixelů. Po vybrání barvy lze rozsah dále upravit – každou složku (HSB) je možné manuálně nebo automaticky rozšířit tak, aby se například započítaly některé odstíny zvolené barvy nebo kompenzoval šum.

### *Edge Detection*

Pro identifikaci hran v obraze. Výsledkem jsou zvýrazněné oblasti, kde dochází k prudké změně jasu. Vhodné pro analýzu tvarů a kontur.

### *Barcode Reading*

Nástroj umožňuje automatické čtení a dekodování čárových kódů. Systém detekuje čárové kódy různých formátů a provádí jejich dekodování pro identifikaci produktů nebo informací.

### *Total Status*

Pokud se v konkrétním programu nachází více inspekčních nástrojů, například kontrola barvy a délky hrany objektu, je výsledek dané kontroly určený logickým součinem (AND). Jinými slovy, pokud je jedna část kontroly označena jako NG (No Good), je takto označen i celkový výsledek daného programu.

## *Výpis dostupných inspekčních nástrojů*

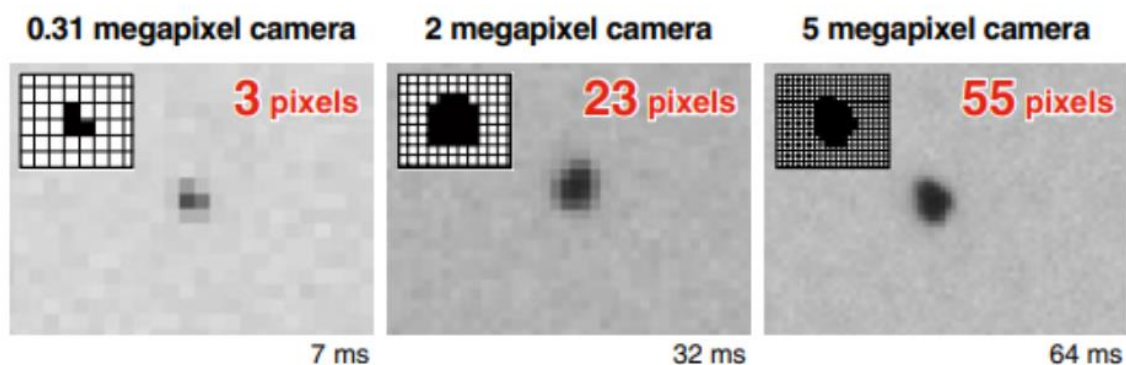
- 3D přítomnost / nepřítomnost
  - 3D porovnání / srovnání plochy řezů / porovnání obrysů / porovnání výšky
  - Porovnání profilu
- Výška
  - Měření výšky / trend měření výšky / měření profilu
- Přítomnost / nepřítomnost
  - Černá/bílá – specifická oblast
  - Shoda vzorců / stínování
- Detekce závad
  - Celková plocha závady / jednotlivé závady / kontrast s pozadím
  - Vada na lince / kruhu / křivce
- Zarovnání
  - Pozice / sklon hrany
- Měření a rozměry
  - Vzdálenost bodů / vzdálenost úseček / vzdálenost mezi hranami
  - Průměr / velikost mezery / šířka / úhel mezi úsečkami
  - Střed čtyřúhelníku / střed mezi dvěma body
- Množství
  - Počet hran / počet shod se vzorem
- ID & OCR/OCV
  - Rozpoznávání znaků / 1D kód / 2D kód
- Funkce
  - ShapeTrax3 / detekce profilu / pozice profilu

### **3.3 Konfigurace vision systému**

Vision systém se skládá z kontroléru, což je centrální jednotka zpracovávající obraz, dále kamery zachytávající obraz a v neposlední řadě také čočky s osvětlením. Pro optimální výběr všech těchto komponent je nutné si nejdříve ujasnit jaké objekty bude vision systém vlastně kontrolovat či detekovat.

Výběr kontroléru závisí na požadovaných funkcích. Keyence nabízí několik řad vision systémů, každé s trochu jiným zaměřením. Každá konkrétní řada se dále dělí na další řady, které se liší výbavou – max. množstvím připojených kamer, max. rozlišení kamer, výpočetní výkon (čas zpracování), 3D funkce, proprietární výpočetní algoritmy atd. [21]

Výběr kamery, respektive jejího rozlišení závisí na nejmenším objektu (čárové kódy, různé vady a praskliny), který bude kamera schopná stabilně detekovat. Pokud uvažujeme vzorový příklad, kdy máme dvě kamery, první s rozlišením 0,31 Mpx (640x480), druhou s 2 Mpx (1600x1200) a šířku zorného pole třicet milimetrů, tak kamera s rozlišením 0,31 Mpx bude schopná detekovat objekty s velikostí 0,25 milimetru a kamera s rozlišením 2 Mpx zvládne detekci objektů až do 0,1 milimetru. Rozlišení kamery neovlivňuje pouze rozlišovací schopnosti, ale také dobu zpracování. Rozdíl doby zpracování mezi 0,3 Mpx kamerou a 5 Mpx kamerou může být až desetinásobný. [21]

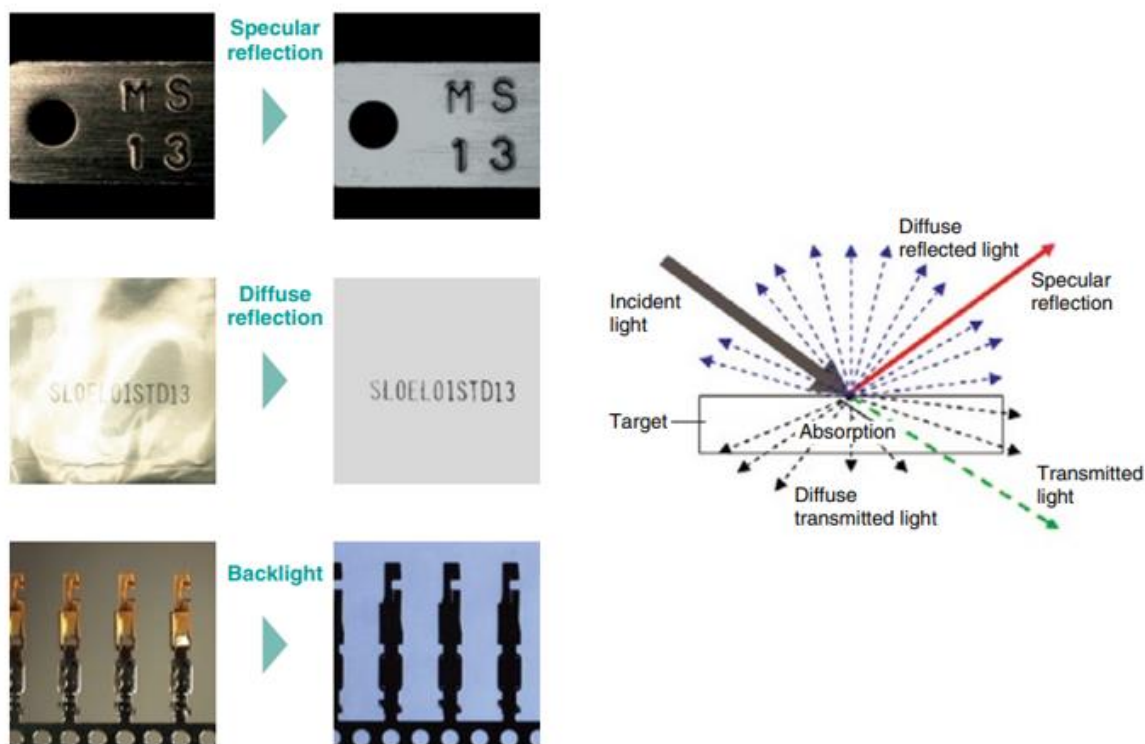


Obr. 18: Rozdíl detekčních schopností kamery v závislosti na rozlišení [21]

Po kameře přichází na řadu volba čočky. Ta ovlivňuje pozorovací úhel / šířku zorného pole. Je nutné uvažovat nejen nad velikostí detekovaného dílu, ale případně jestli nedochází například u dílů na dopravníku ke změnám v pozici (nepřesně daná pozice dílu), v tom případě je potřeba uvažovat s určitou rezervou. [21]

Výběr osvětlení znatelně ovlivňuje schopnosti a stabilitu detekce. Způsob osvětlení lze rozdělit na tři hlavní typy:

- Zrcadlový odraz (Specular reflection) – Systém, ve kterém je světlo zrcadlově odražené od cíle přijímáno čočkou. Efektivní pro povrchy kovových nebo skleněných desek.
- Difúzní odraz (Diffuse reflection) - Systém, v němž se zamezí zrcadlovému odrazu světla od cíle a objektiv přijímá rovnoměrné světlo. Účinné pro lesklé povrchy nebo průhledné fólie, u nichž může snadno dojít k oslnění.
- Průhledný typ (Backlight) – Systém, ve kterém je cíl osvětlen zezadu a silueta je detekována pomocí procházejícího světla. Účinné pro případy, kdy odražené světlo neposkytuje ostrý kontrast, nebo pro detekci složitých profilů. [21]



Obr. 19: Způsoby osvětlení [21]

Kvalitu detekce kromě metody osvětlování ovlivňuje také použití optimální barvy během osvětlování. Využitím metody doplňkové barvy (obr.20) lze zajistit vysoký kontrast. Příklad aplikace této metody je osvětlení oranžové lepicí pásky modrou barvou (doplňková barva k oranžové). Při osvětlení objektu jeho doplňkovou barvou lze získat monochromatický obraz objektu se zvýšeným kontrastem. [21]



Obr. 20: Metoda doplňkové barvy [21]

## 4. Návrh a realizace integrace systému Universal Robots a vision systému Keyence

### *Universal Robots*

Universal Robots je dánský výrobce menších průmyslových kolaborativních robotů (kobotů) se sídlem v Dánsku. Společnost jako první uvedla na trh kolaborativní robot, který může bezpečně pracovat vedle zaměstnanců, čímž odpadá nutnost používat bezpečnostní klece nebo oplocení. Od roku 2022 byla společnost Universal Robots lídrem na trhu kolaborativních robotů, připisuje se jí 40-50% podíl na trhu.

### *Kobot*

Kolaborativní robot, jinými slovy kooperativní robot nebo kobot je průmyslový robot, jehož konstrukce je taková, aby mohl bezpečně pracovat v bezprostřední blízkosti lidí. Této konstrukce může být dosaženo například sníženou hmotností nebo eliminací ostrých hran. Kolaborativní robot je dále vybaven bezpečnostními čidly a zároveň jsou rychlosti a síly jednotlivých pohonů omezeny takovým způsobem, aby člověka nemohl zranit.

Mezinárodní Federace pro Robotiku (IFR) definuje čtyři úrovně spolupráce mezi průmyslovými roboty a lidskými pracovníky:

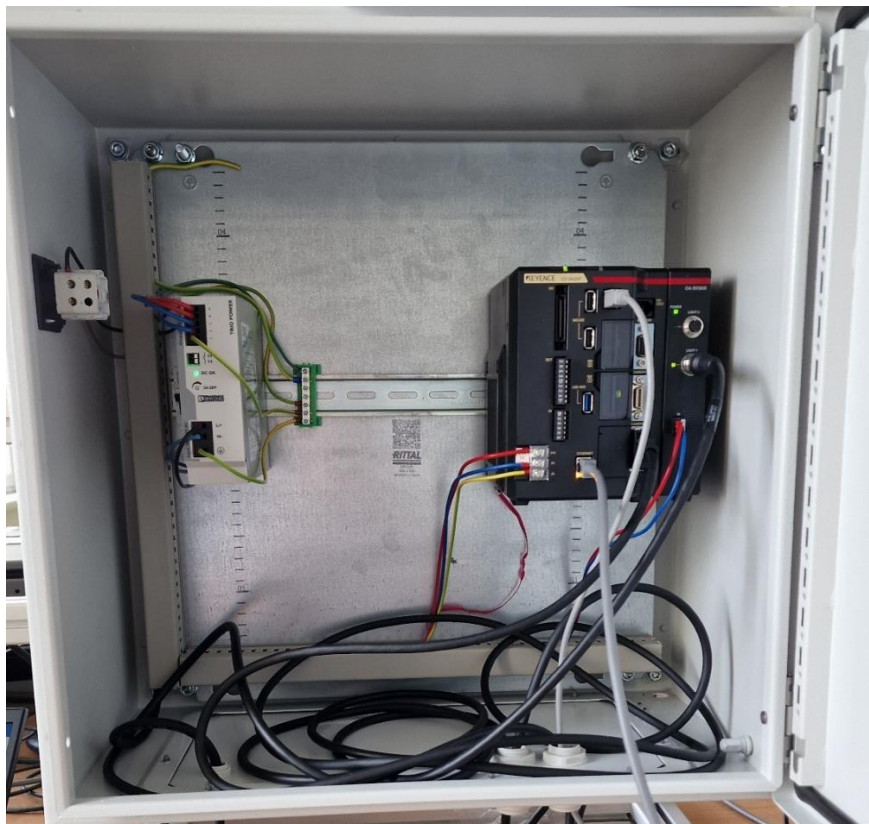
- **Koexistence:** Člověk a robot pracují vedle sebe bez oplocení, ale bez společného pracovního prostoru.
- **Sekvenční spolupráce:** Člověk a robot jsou aktivní ve sdíleném pracovním prostoru, ale jejich pohyby jsou sekvenční – nepracují na dílu současně.
- **Spolupráce:** Robot a člověk pracují na stejném dílu současně, přičemž oba jsou v pohybu.
- **Reaktivní spolupráce:** Robot reaguje v reálném čase na pohyb lidského pracovníka.

Ve většině dnešních průmyslových aplikací kobotů sdílejí kobot a lidský pracovník stejný prostor, ale úkoly plní nezávisle či postupně (koexistence nebo sekvenční spolupráce). [24]

## 4.1 Instalace komponent

Vision systém Keyence použitý pro praktickou část této diplomové práce se skládá z následujících komponent:

- Řídicí jednotka CV-X420F
- Rozšiřující jednotka pro řízení LED osvětlení CA-DC60E
- Strukturované osvětlení s projekcí vzorů CA-DQP12X
- 2 Mpx monochromatická kamera CA-H200MX



*Obr. 21: Instalační skříň vision systému*

Na obr.21 lze vidět instalační skříň obsahující řídicí jednotku CV-X420F spolu s rozšiřující jednotkou CA-DC60E pro řízení LED s podporou strukturovaného světla. Vše je napájeno 24V/10A zdrojem od Phoenix Contact.



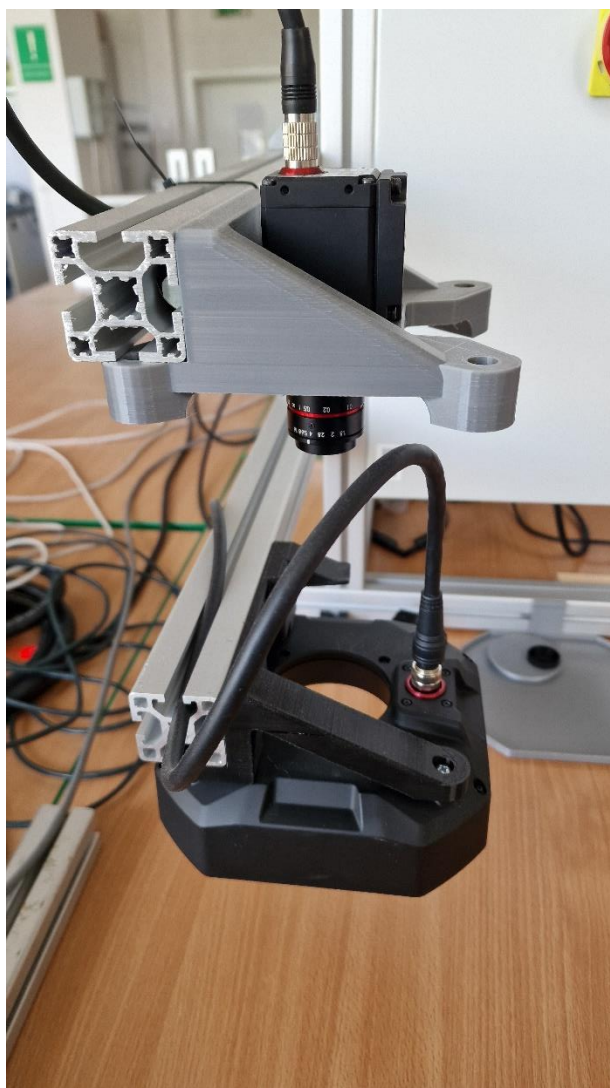


Obr. 22: Řídicí jednotka systému a rozšiřující jednotka osvětlení

Do řídicí jednotky (obr.22) je kromě napájení přiveden USB kabel pro komunikaci s PC, Ethernetový kabel pro komunikaci s robotem UR3 a také datový kabel od kamery. Dále je do jednotky vložena SD karta sloužící jako hlavní úložiště (možnost také připojit klasický HDD) pro ukládání programů, referenčních snímků a dalších snímků pro vyhodnocení statistik daného programu (poměr počtu OK a NG snímků). Do rozšiřující jednotky osvětlení je přivedeno napájení a samotné propojení s osvětlením.

Řada řídicích jednotek CV-X od společnosti Keyence je charakteristická svým přívětivým uživatelským prostředím a širokou nabídkou funkcí jako je například podpora kamer až do rozlišení 64 Mpx pro velmi detailní kontroly, možnost multispektrálních snímků, strukturované osvětlení s projekcí vzorců přidávající údaje o výšce zkoumaného objektu a stabilní kontrolu bez vlivu okolního prostředí či stavu povrchu objektu. Dále proprietární algoritmy LumiTrax a ShapeTrax 3. Řada CV-X zahrnuje čtyři modelové řady (100, 200, 300 a 400) lišící se v několika parametrech. Algoritmus ShapeTrax3 ovšem není podporovaný u všech řad.

Konkrétně použitý model CV-X420F má 3-jádrový DSP procesor, podporuje připojení až čtyř kamer současně a zvládá nejvyšší rozlišení kamer 2 Mpx.



*Obr. 23: Instalace kamery a osvětlení*

Kamera CA-H200MX (obr.23) disponuje monochromatickým CMOS senzorem (1600x1200) s elektronickou závěrkou nastavitelnou v rozmezí 1/15–1/50000 (celkem 16 kroků). Strukturované osvětlení CA-DQP12X (obr.23) obsahuje 72 bílých LED jako zdroj pro projekci vzorů z osmi směrů s doporučeným rozsahem v ose Z – 100 až 150 milimetrů. Opakovatelnost měření je 0,1 milimetru (závisí i na ostatních komponentech systému). Dále obsahuje 36 LED pro mód snímání LumiTrax. Spotřeba energie se pohybuje kolem 36W. K řídicí jednotce se připojuje 12-kolíkovým kulatým konektorem.

Řídicí jednotka osvětlení CA-DC60E pro řízení strukturovaného osvětlení se vyznačuje odezvou pod 1 ms, 1024stupňovým rozsahem intenzit a spotřebou proudu 4,6 A při napájení 24 V.



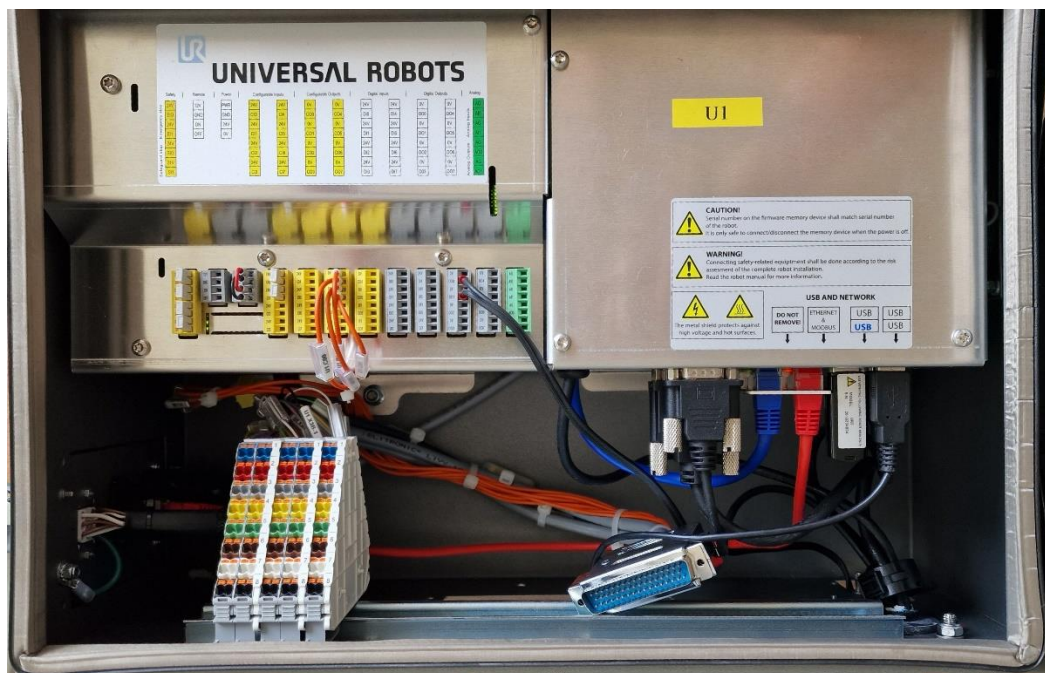
*Obr. 24: kobot UR3 s efektorem*

Robot, přesněji kolaborativní robot, použitý pro tuto konkrétní integraci se systémem strojového vidění je model UR3 od firmy Universal Robots. Je to šestiramenný manipulátor (šest stupňů volnosti). Na prvních pěti kloubech jsou možné rotace v rozsahu 360°, na posledním kloubu (s koncovým efektem) je dostupná nekonečná rotace.

Vybrané parametry modelu UR3:

- Opakovatelnost –  $\pm 0,1$  mm
- Příkon – Min. 90 W, typicky 125 W, max. 250 W
- Užitečné zatížení – 3 kg
- Dosah – 500 mm
- Stupně volnosti – 6 otočných kloubů
- Programování – Grafické uživatelské rozhraní Polyscope s 12“ dotykovým displejem
- Klasifikace IP – IP64

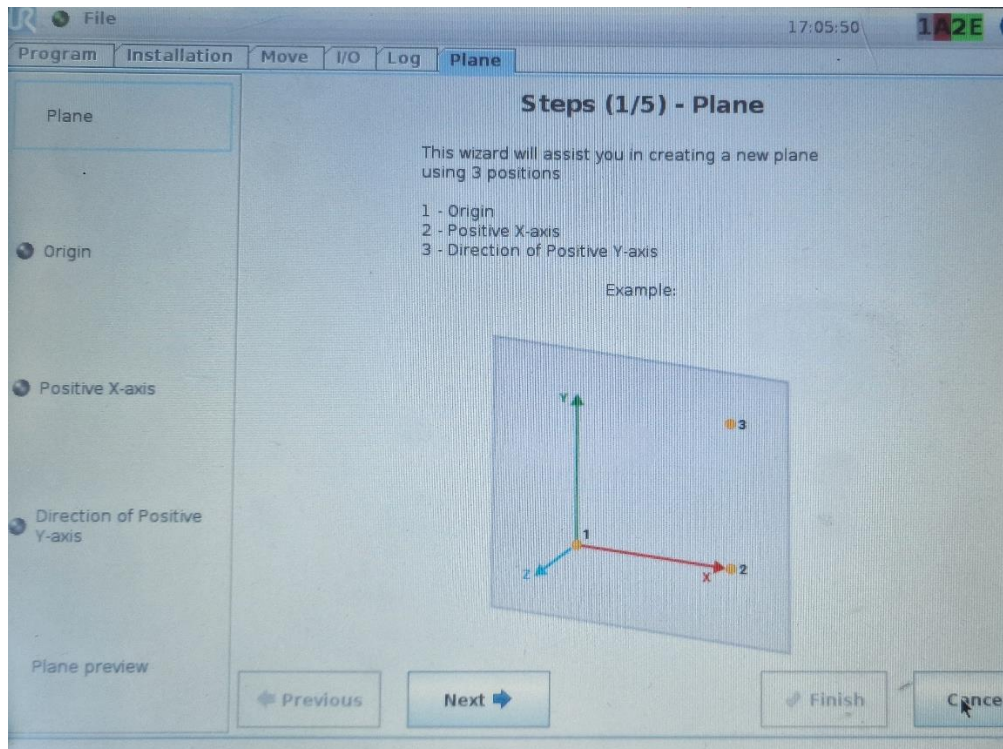
Jako efektor je použito paralelní chapadlo EHPS-16-A od firmy Festo. Pohon je elektrický DC motor se šnekovým převodem pracující na napětí 24 V. Pro efektor byl vytisknut adaptér na 3D tiskárně.



*Obr. 25: Kontrolér robota UR3 spolu s elektronikou*

## 4.2 Prvotní nastavení

Příprava na straně robota spočívá převážně v nastavení pracovní roviny a TCP (Tool Center Point). Pracovní rovina nám definuje podobu souřadnicového systému, tzn. která osa bude směřovat jakým směrem a který směr konkrétní osy bude kladný. V uživatelském prostředí UR3 je k tomuto nastavení uzpůsoben jednoduchý průvodce. V prvním kroku se zvolí libovolný počáteční bod fungující jako počátek souřadného systému, dále se zvolí druhý bod definující kladný směr osy X, v posledním kroku stačí zvolit bod, který definuje kladný směr osy Y.



Obr. 26: Definování roviny

Definování TCP neboli středového bodu nástroje probíhá tak, že se TCP přesune do stejného bodu ze čtyř různých úhlů.

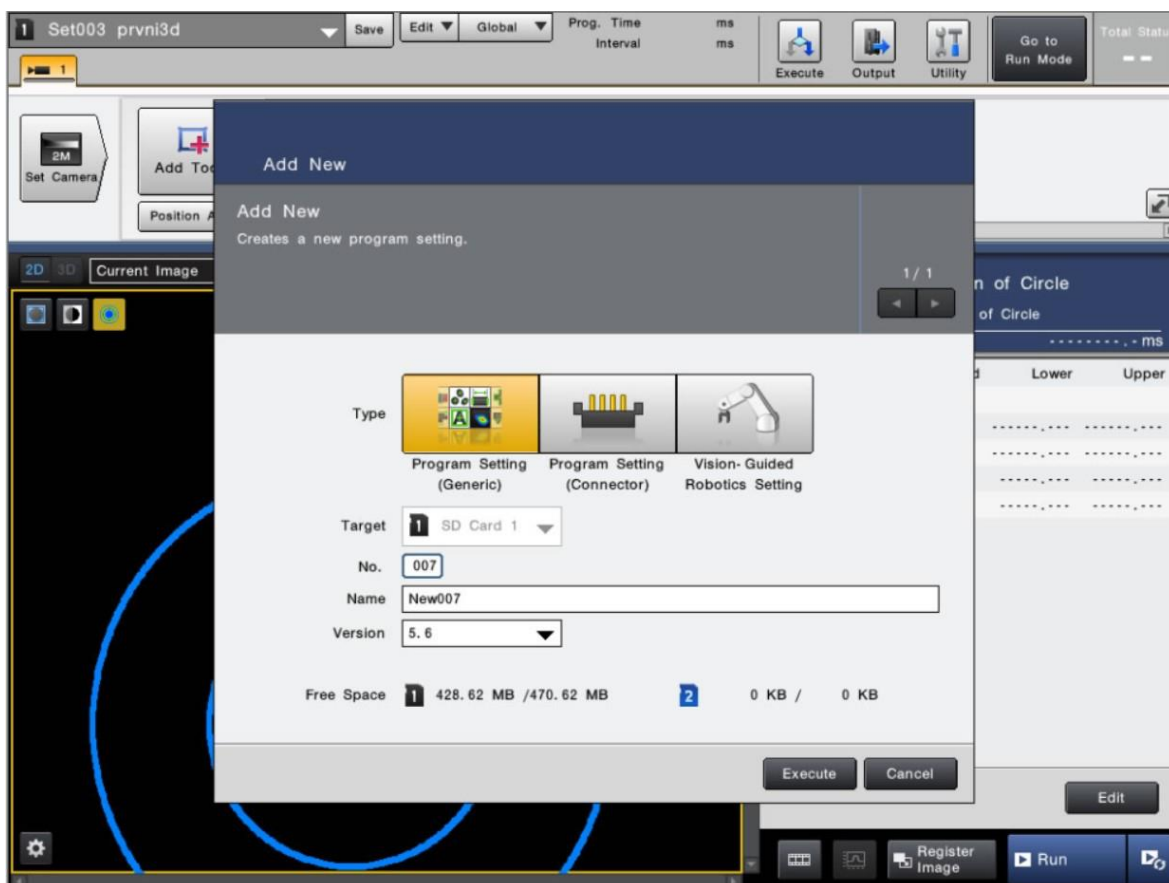
Pro přístup do uživatelského rozhraní kontroléru CV-X420F je možné použít přímého zapojení monitoru a myši do kontroléru nebo využít program CV-X Series Terminal pro operační systémy Windows. Ten je ke stažení na stránkách Keyence a přes USB zprostředkovává přenos obrazu z kontroléru do PC a signál myši jako vstup do kontroléru. Další možností, kterou program nabízí je připojení vzdáleně pomocí IP adresy.

Po připojení do uživatelského prostředí kontroléru je možné začít vytvářet program. Před vytvářením programu je ale potřeba seřídit obraz z kamery. To se dělá pomocí otočných ovladačů clony a zaostření přímo na kameře.

### 4.3 Integrace robota a strojového vidění – úloha 1

Při vytvoření nového programu je na této konkrétní specifikaci systému možnost volit mezi třemi různými typy programů – Generic, Connector a Vision-Guided Robotics. Typ Generic

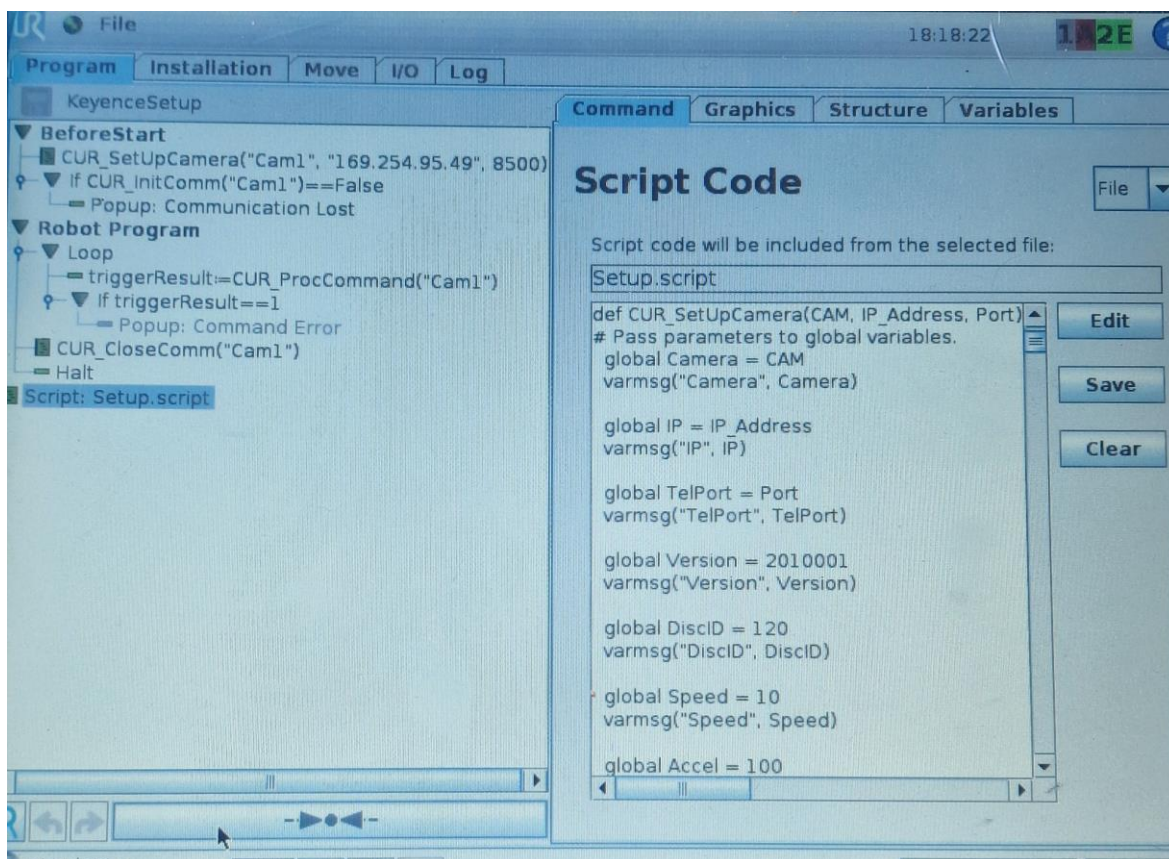
slouží k obecným kontrolám a jsou v něm dostupné inspekční nástroje zmíněné v kapitole 4.2. Connector je speciální typ programu, který je svými funkcemi uzpůsoben pro kontrolu různých parametrů konektorů, mezi tyto parametry patří piny, rozteč pinů, sklon hran atd. Posledním typem programu je Vision-Guided Robotics, v překladu – viděním naváděná robotika. V této kapitole je použit program typu Vision-Guided Robotics.



Obr. 27: Volba typu programu

Před začátkem nastavování Vision-Guided Robotics programu je zapotřebí nejdříve provést pár kroků týkajících se samotné komunikace mezi kontroléry vision systému a robota UR3. Komunikace probíhá přes Ethernet, proto je prvním krokem nastavení vzájemných IP adres. Na straně robota je zapotřebí v nastavení sítě nastavit IP adresu, masku podsítě a změnit typ adresy na statickou. Stejně tak na straně vision systému je nutné v nastavení sítě zvolit vhodnou IP adresu. Vzhledem k nastavené masce sítě je důležité, aby se první tři čísla IP adres robota a vision systému shodovaly. To zaručí, že budou obě zařízení ve stejné podsíti.

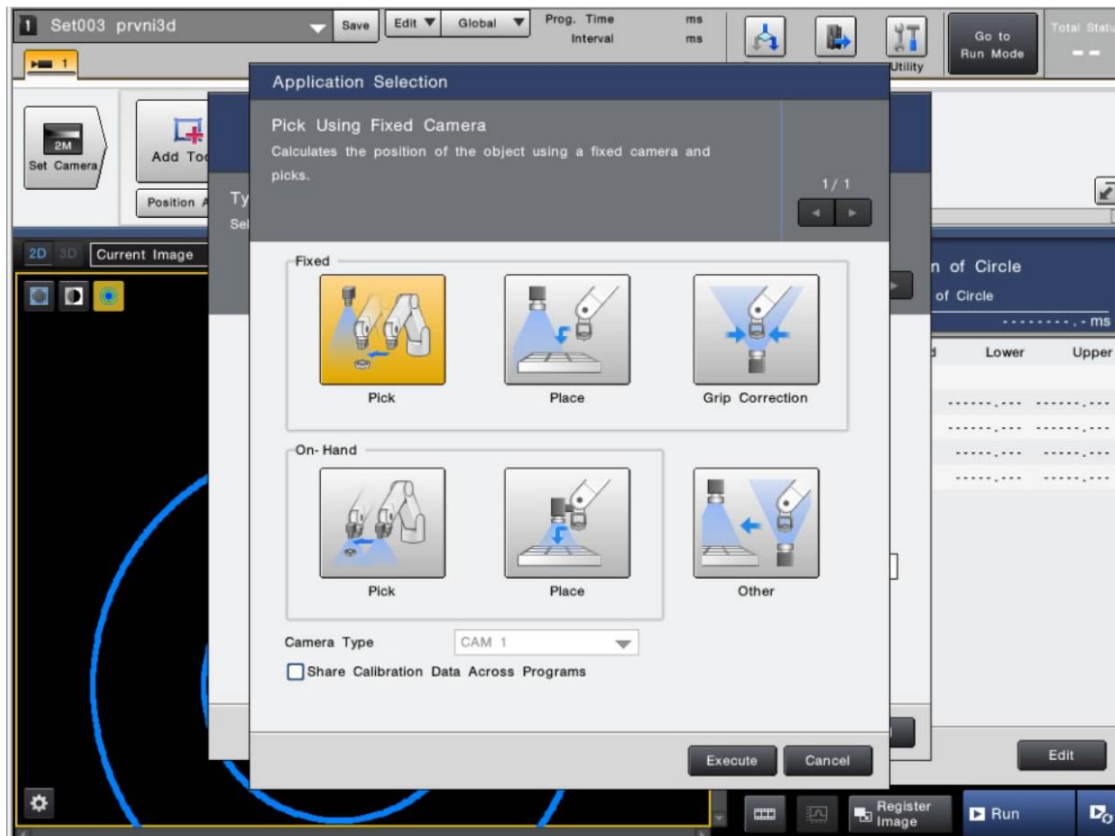
K úspěšné komunikaci je dále potřeba komunikační skript, ten je ke stažení na stránkách firmy Universal Robotics. Skript zajišťuje procesy jako jsou zahájení komunikace, příkazy k ovládání pohybu od kontroléru vision systému nebo předání aktuálních souřadnic a orientace robota. Ve skriptu stačí nastavit aktuální IP adresu vision systému.



Obr. 28: Program s komunikačním skriptem

Poslední věc, kterou je nutné upravit je podoba výstupního řetězce z vision systému. V nastavení každého programu je totiž možné upravit jaká data se posílají po Ethernetu (nebo jiném způsobu komunikace) ven z vision systému například do robota nebo do PLC. Mimo užitečná data jako jsou výška objektu, barva objektu atd. je nutné na začátek výstupního řetězce přidat znak „(“ a na konec „0)“, tyto znaky zaručí správný tvar komunikačního řetězce. Výsledný tvar výstupního řetězce tedy v tomto konkrétním případě obsahuje nejdříve znak „(“, dále hodnotu, zda je detekce OK/NG, poté následuje šest hodnot představující souřadnice X,Y,Z a rotace Rx, Ry, Rz a poté je řetězec ukončen znaky „0)“. Lze také zvolit oddělovač dat, konkrétně je na výběr čárka a středník.

Po provedení všech kroků popsaných výše a úspěšném navázání spojení mezi kontroléry je možné začít s nastavováním samotného Vision-Guided Robotics programu. Tento typ programu umožňuje vybrat si ze dvou hlavních typů úloh – Pick nebo Place. Obě jsou dostupné v konfiguraci se stacionární kamerou a také s kamerou umístěnou na robotovi. Tato práce popisuje stacionární variantu Pick úlohy.



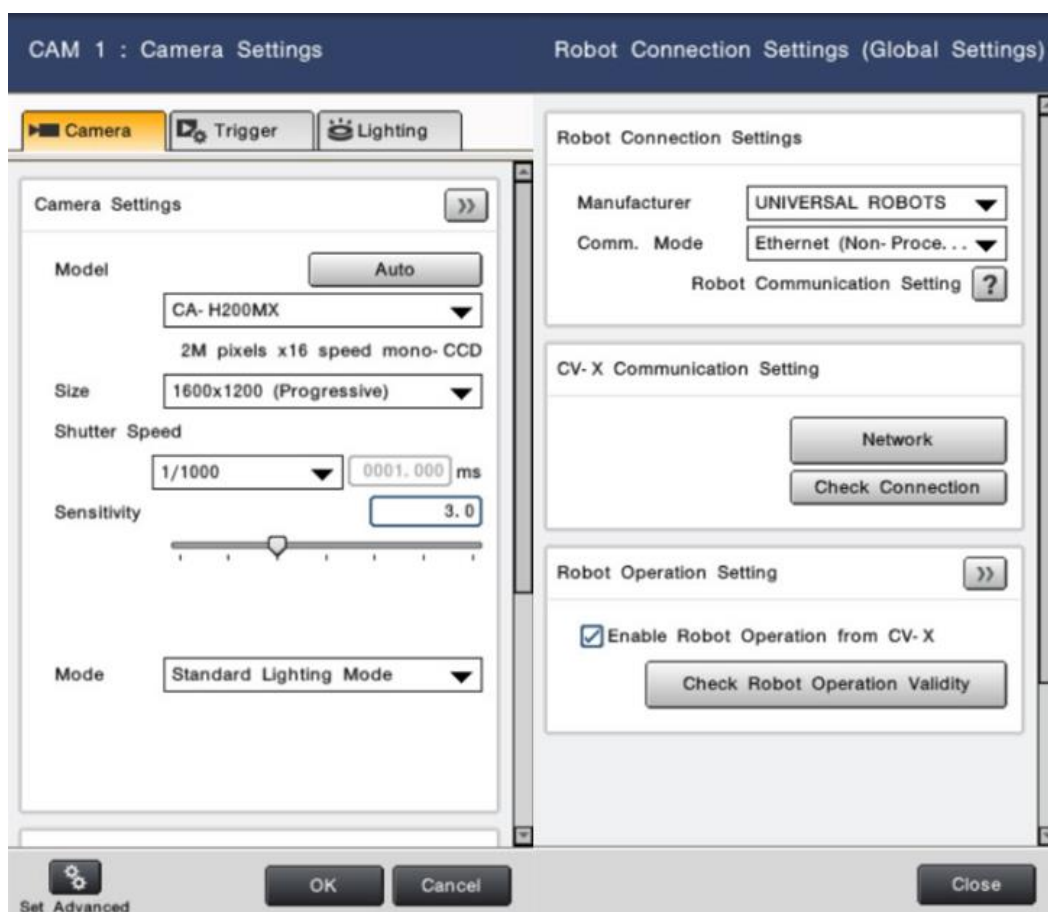
Obr. 29: Volba typu úlohy

Nastavovací proces se skládá z pěti dílčích kroků:

- Nastavení kamery
- Nastavení spojení s robotem
- Kalibrace
- Nastavení nástroje ShapeTrax3 pro detekci požadovaného objektu
- Nastavení a kalibrace operace úchopu objektu



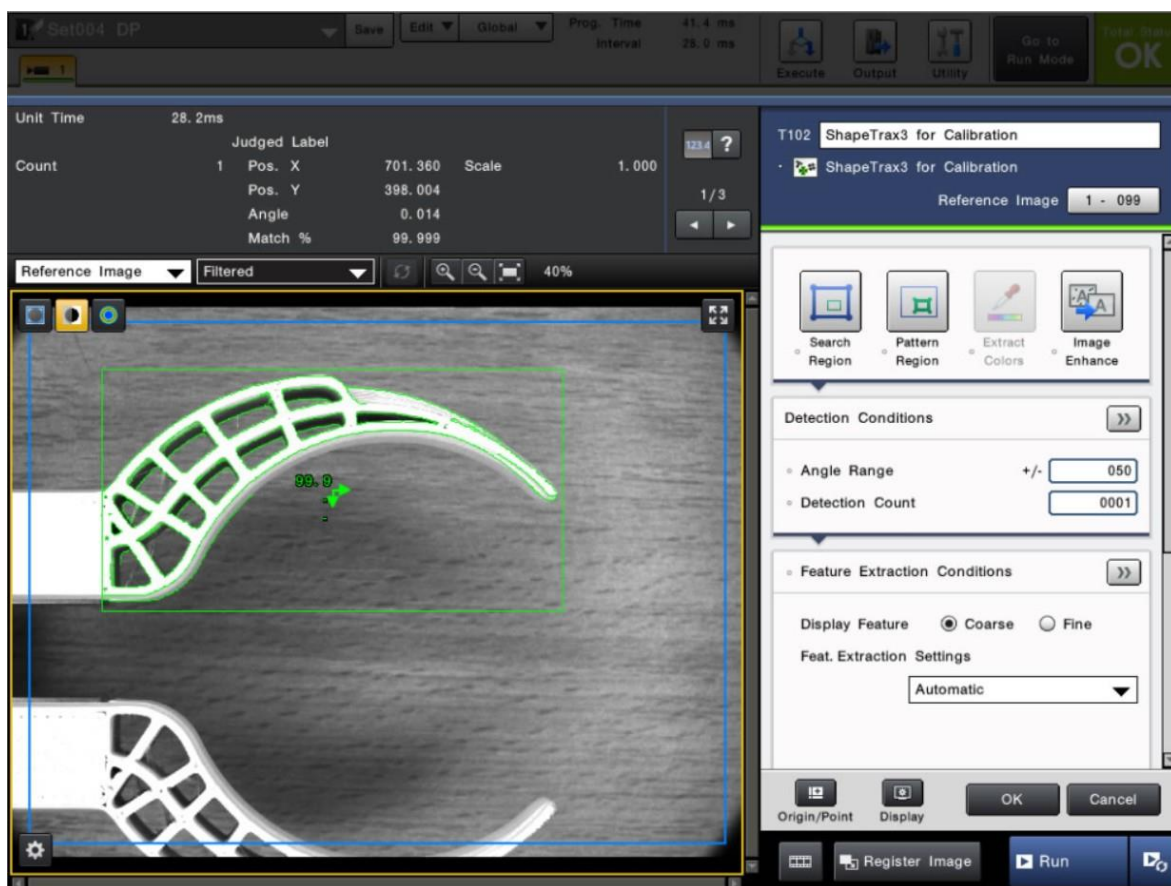
Nastavení kamery spočívá ve výběru modelu kamery (pro případ, že je ke kontroléru připojeno více kamer), rozlišení kamery a ladění obrazu pomocí rychlosti závěrky a citlivosti. Citlivostí je možné získat vyšší úroveň detailů za cenu zvýšeného šumu. U nastavení spojení s robotem se volí typ výrobce, způsob komunikace, ověřuje se, zda je robot připojen (ping na robota) a zda je spuštěn komunikační skript.



Obr. 30: Nastavení kamery a spojení s robotem

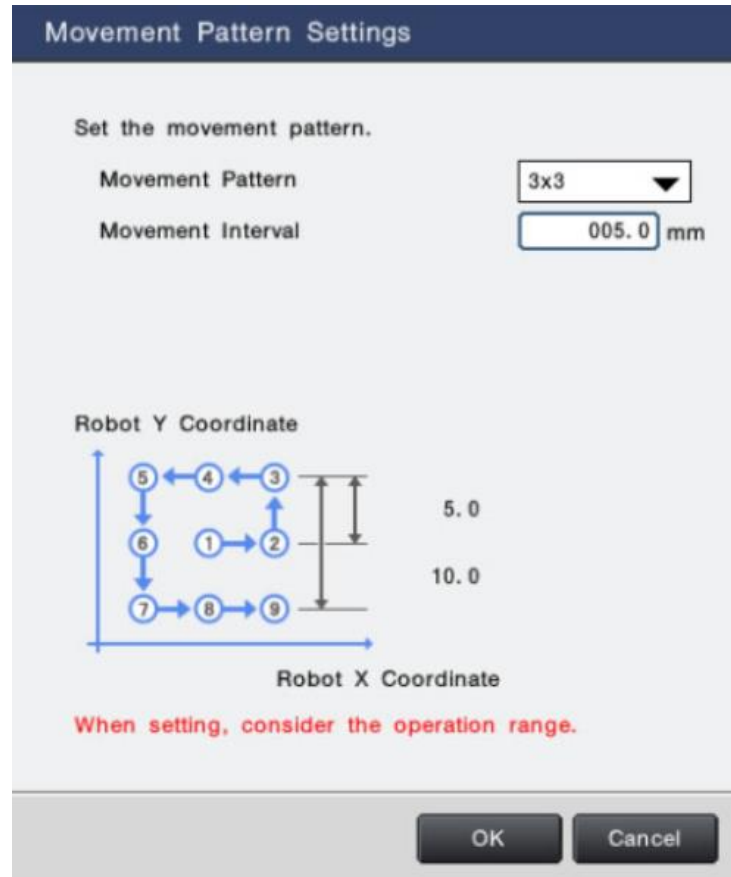
Samotný kalibrační proces je možné udělat manuálně nebo automaticky. Kalibrace se skládá ze čtyř kroků. Prvním krokem je startovací pozice, kdy se musí robot dostat do takové pozice, aby byl v záběru kamery. V této pozici se zvolí možnost „Get Robot Coordinates“ a kontrolér vision systému si od robota vyžádá aktuální pozici a orientaci. Připomínám, že po celou dobu procesu kalibrace musí na robotu běžet komunikační skript. Po uložení pozice robota je možné přejít na další krok – nastavení detekčního nástroje pro kalibraci. Detekční nástroj je v tomto případě již dříve zmiňovaný ShapeTrax3. Nastavení se provede přesunem

efektoru robota do záběru kamery, uloží se referenční snímek a na tomto snímku se pak zvolí charakteristická oblast, kterou bude ShapeTrax3 detekovat během kalibrace v následujících krocích. Jak lze vidět na obr.31, charakteristickou oblastí byla zvolena jedna část čelistí efektoru (zelený rámeček). V nastavení tohoto nástroje je možné upravit hranici míry shody s referenčním snímkem za které bude detekce ještě vyhodnocena jako úspěšná.



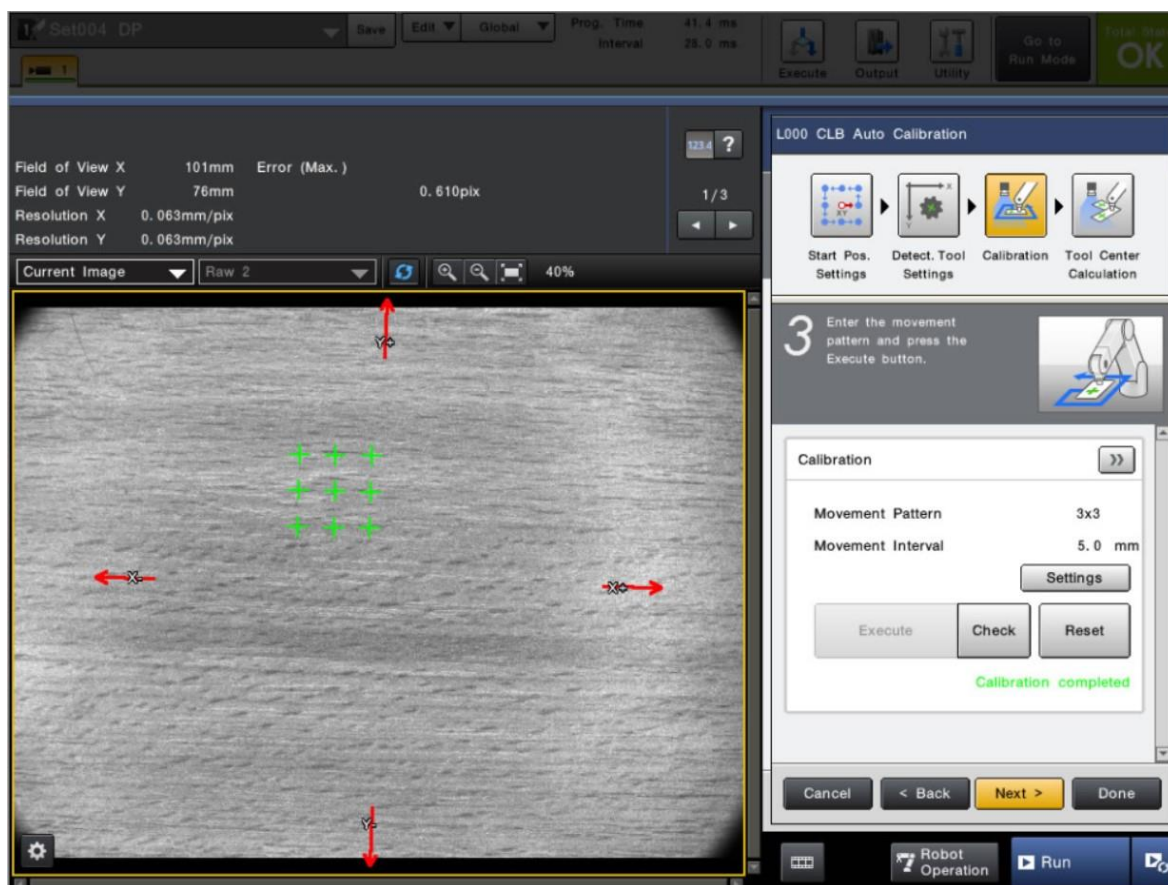
Obr. 31: ShapeTrax3 pro kalibraci nástroje

Následuje samotná kalibrace. Princip je takový, že se vezme výchozí pozice zvolená v prvním kroku a nastaví se pohybový vzorec, který vision systém sám provede. Pohybový vzorec může být mřížka 3x3 nebo 5x5 bodů. Kromě mřížky se také nastavuje, jak jsou od sebe dané body vzdáleny. Vzhledem k omezenému prostoru, který kamera zabírá jsem zvolil pět milimetrů.



Obr. 32: Kalibrační pohybový vzorec

Proces samotný probíhá tak, že robot začíná v bodě 1 viz. obr.32 a postupuje dále podle schématu, přičemž mezi body je nastavená vzdálenost pět milimetrů. V každém bodě se udělá snímek, který je analyzován vyhledávacím nástrojem ShapeTrax3 a hledá na něm vzor nastavený v předchozím kroku, pokud ho najde, uloží se aktuální souřadnice a orientace robota a k tomu pozice nalezeného vzoru na snímku. Tímto způsobem se „sladí“ jaká vzdálenost v souřadnicích robota odpovídá jaké vzdálenosti, konkrétněji kolika pixelům na snímku kamery. Stejný proces se ještě opakuje s rotací – zvolí se úhel a provedou se tři rotace po pěti stupních v ose Z. Výsledek kalibrace je možné vidět na obrázku níže.

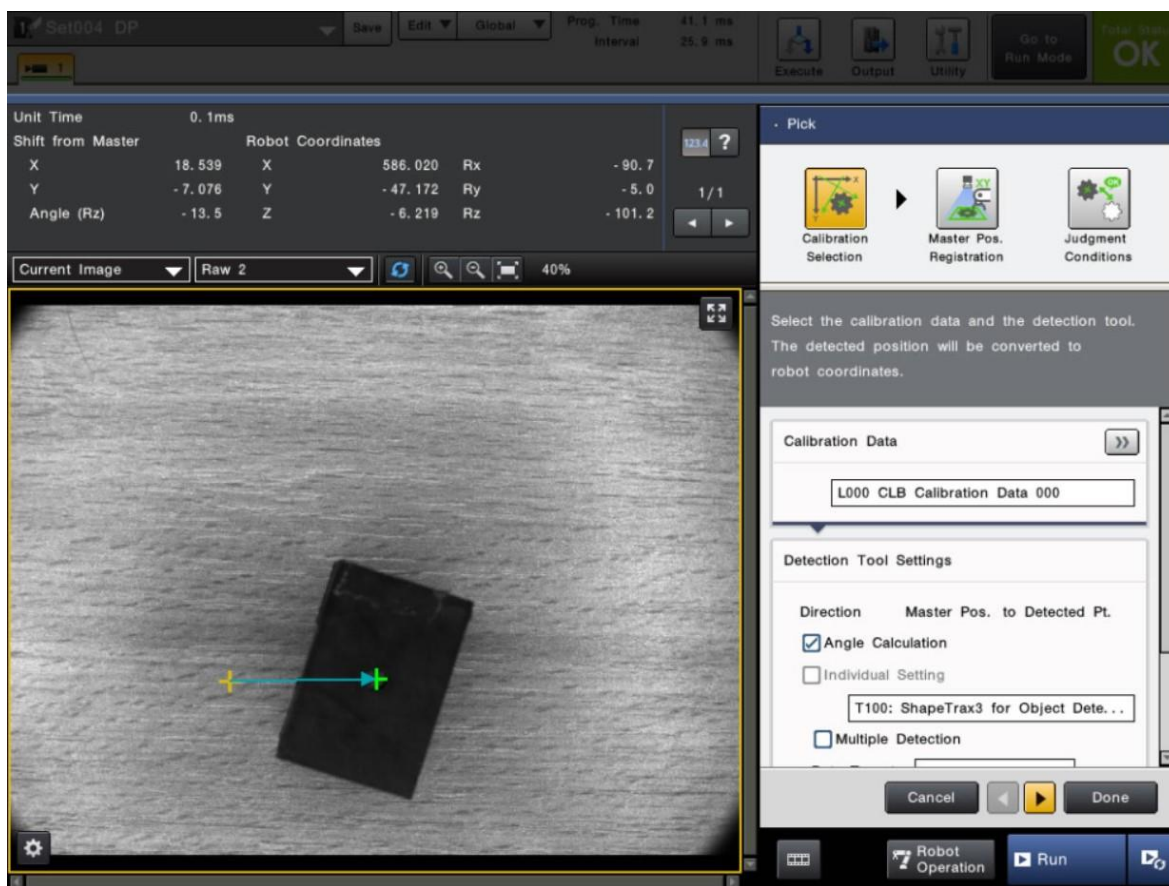


Obr. 33: Výsledek kalibrace

Po kalibraci přichází na řadu čtvrtý krok – nastavení ShapeTrax3 pro detekci hledaného objektu. Princip je stejný jako při nastavování během kalibrace – do záběru kamery se vloží objekt, který budeme robotem uchopovat, uloží se snímek (referenční) a na daném snímku se označí oblast ve které se samotný objekt nachází. Nastavení také dovoluje zvolit úhel o kolik bude moci být objekt otočený vůči vzoru nebo jestli má algoritmus na snímku hledat více kusů. Je možné také aplikovat řadu filtrů popisovaných v předchozích kapitolách – operace šedobílé morfologie, filtrování mediánem, průměrování, izolaci šumu, Laplacián, ostření, Sobel, Prewitt atd.

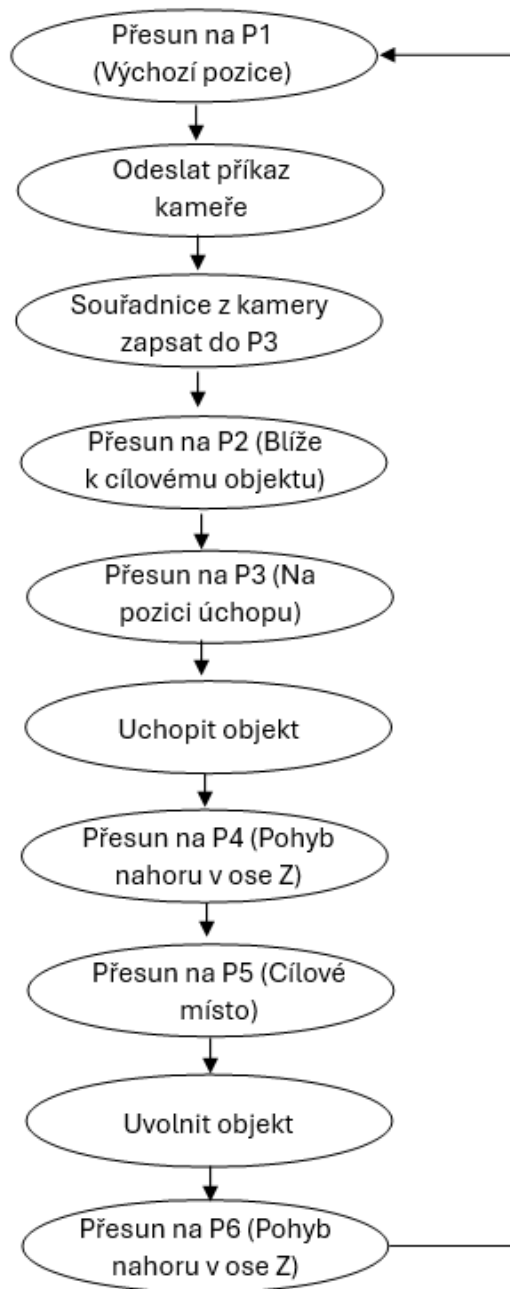
Posledním krokem je kalibrace samotné operace úchopu. Skládá se ze tří dílčích kroků. Prvním je volba kalibračních dat. V kontroléru se totiž může nacházet více kalibračních dat pro více kamer. Dalším krokem je registrace Master pozice. Postup je následující – efektor robota se přesune do záběru kamery, uchopovaný předmět se vloží na takové místo, aby ho

efektor robota mohl uchopit. V tuto chvíli se uloží referenční snímek a vision systém může vypočítat vzájemný poziční vztah mezi objektem a efektozem – má totiž k dispozici souřadnice robota a díky v předchozím bodě nastavenému ShapeTrax3 i pozici objektu na snímku. Poslední krok spočívá v pohybu efektoru pryč ze záběru, přičemž uchopovaný objekt zůstává na stejném místě jako v předchozím kroku. Uloží se referenční snímek a vypočte se poloha objektu. Tímto je procedura dokončena a vision systém nyní bude vypočítávat posuvy od této referenční pozice.



Obr. 34: Posun objektu od Master pozice

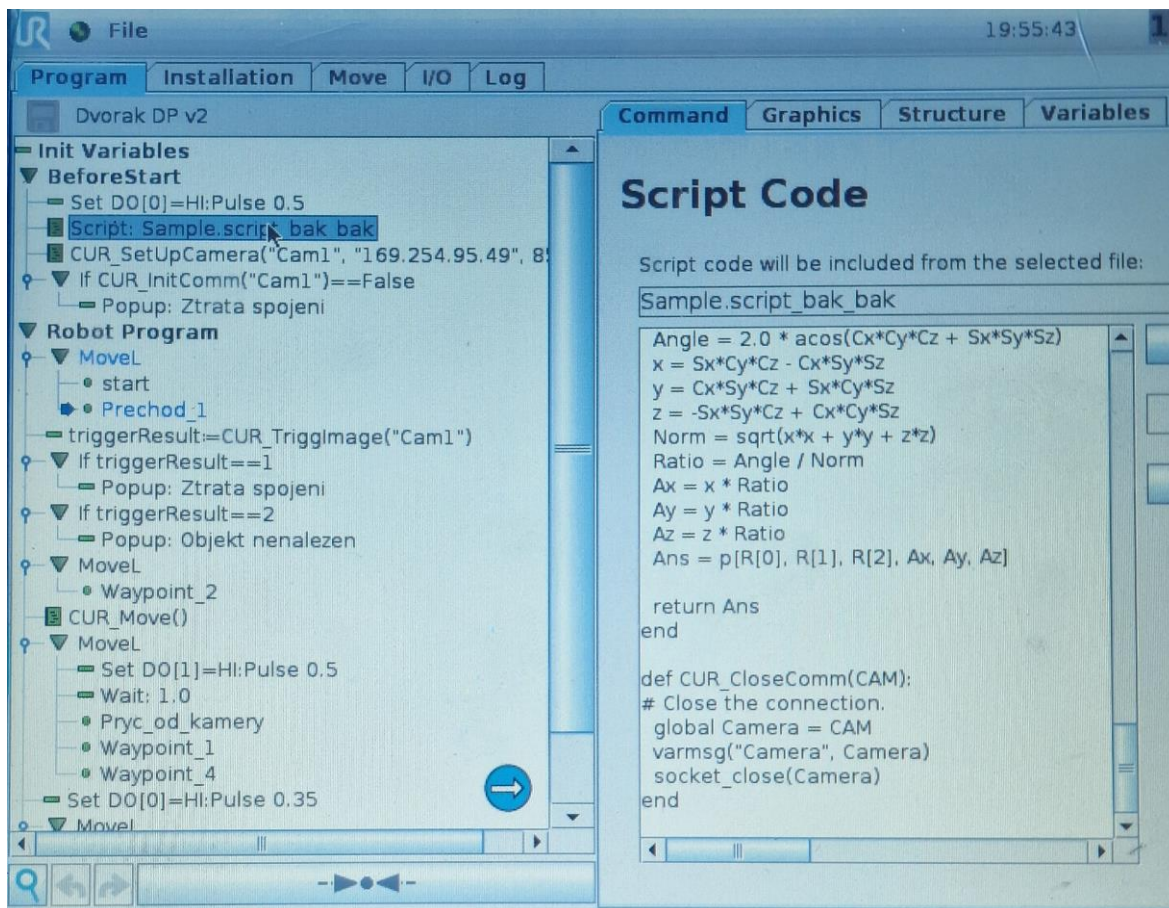
Nyní je dokončena celá kalibrační procedura ve vision systému a na řadu přichází tvorba programu pro robota UR3.



*Obr. 35: Struktura programu*

Na obr.35 lze vidět strukturu programu – robot začíná ve výchozí pozici, odešle příkaz do vision systému – ten pořídí snímek, ShapeTrax3 ho prohledá, najde objekt, vypočte se jeho posun od Master pozice, data se pošlou zpět do robota, ten pomocí skriptu popsaného níže vypočte souřadnice a uloží je jako budoucí pozici P3. Robot dále najíždí na pozici blíže k objektu, poté přichází na řadu vypočtené souřadnice z vision systému. V uchopovací

pozici se aktivuje elektro-efektor, uchopí objekt, provede se pohyb v ose Z směrem od země, robot se přesouvá pryč od kamery na cílovou pozici, kde pokládá objekt a poté se vrací na startovní pozici. Program takto běží ve smyčce, dokud vision systém vrácí správná data, pokud objekt nebyl nalezen, program se přerušuje.



Obr. 36: Program pro 1.úlohu

Program běžící na robotu UR3 na obr.36 principiálně funguje podle diagramu na obr.35. Na samotném začátku, před spuštěním jakéhokoliv pohybu, se otevřou čelisti efektoru, aby byly připraveny na úchop předmětu. Dále dojde k navázání spojení s kamerou vision systému, to probíhá tak, že se zavolá funkce „CUR\_InitComm“, funkci se předají vstupní argumenty jako název kamery, IP adresa a číslo portu. Pokud funkce vrátí hodnotu „False“, spojení nebylo navázáno a program skončí, v opačném případě se přejde na hlavní část programu. Úvodní pohyb je přesun robota na výchozí pozici, dále robot odešle kameře příkaz k pořízení snímku a vyhodnocení získaných dat, toto probíhá voláním funkce „CUR\_TriggImage“.

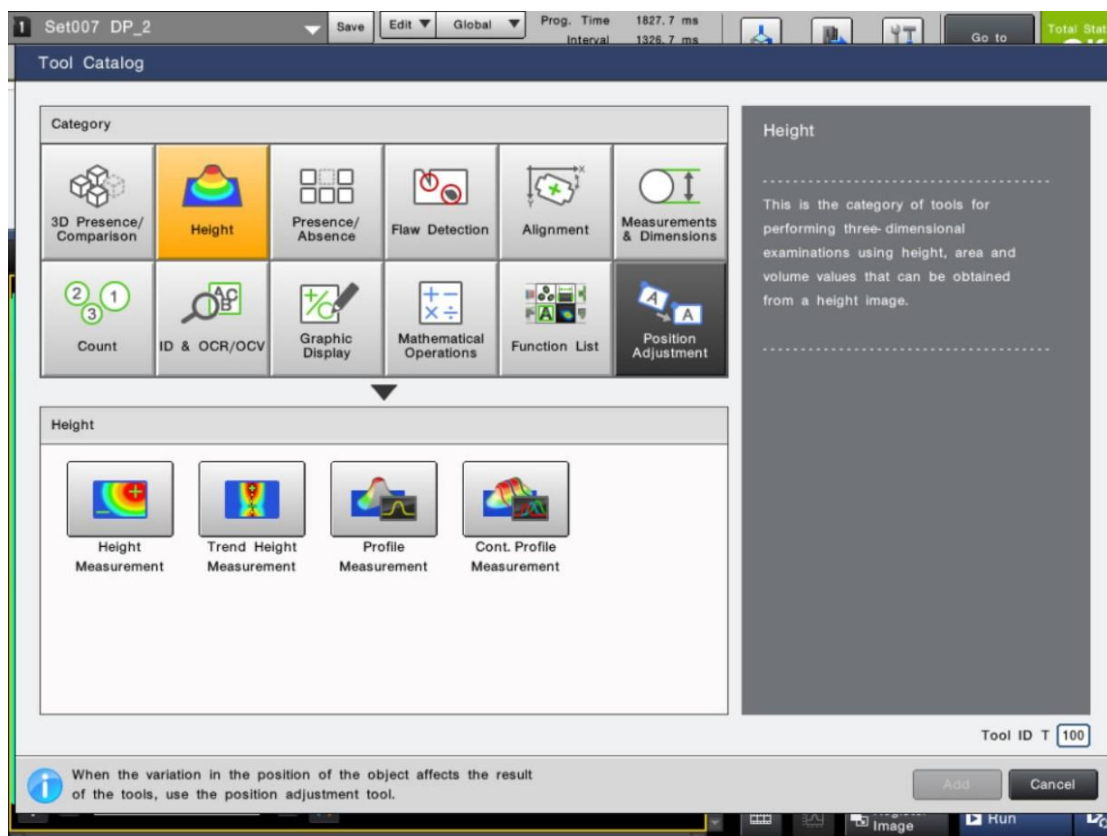
Tato konkrétní funkce zajišťuje odeslání požadavku do kamery, příjem datového řetězce z kamery a následnou práci s přijatými daty. Druhý bajt datového řetězce z kamery označuje, zda vyhodnocení snímku v kontroléru vision systému proběhlo jako OK nebo NG. Podle hodnoty tohoto bajtu se rozhoduje, zda program pokračuje nebo bude ukončen. Pokud je snímek OK (hledaný objekt nalezen), následuje práce s daty představující konverzi jednotek – vision systém vrací milimetry a stupně, robot ale potřebuje metry a radiány. Následuje přiblížení k objektu, konkrétně do prostoru kamery a volání funkce „CUR\_Move“. Tato funkce vykonává pohyb na pozici úchopu objektu. Před samotným pohybem na finální pozici je ale nutné data z kamery týkající se rotace v podobě Roll, Pitch a Yaw převést na popis pomocí způsobu osa-úhel pro robota. Po přiblížení na pozici úchopu je objekt uchopen elektro-efektorem, zvednut ze země a přesunut na cílové místo, kde je uvolněn z čelistí efektoru. Následuje přesun na výchozí pozici a program se opakuje.

#### **4.4 Integrace robota a strojového vidění – úloha 2**

Druhá úloha integrace vision systému Keyence a kobota UR3 od Universal Robots bude namísto navádění kobota kamerou využívat 2,5D funkci „Height Measurement“ měřící výšku objektu zjištěnou pomocí strukturovaného světla. Pro využití této funkce je nutné zvolit typ programu General namísto Vision-Guided Robotics.

Princip této úlohy spočívá v třídění objektů podle jejich výšky – je pevně dána pozice, kde se bude objevovat objekt, vision systém určí jeho výšku a robot ho na základě této hodnoty přesune na příslušnou pozici.

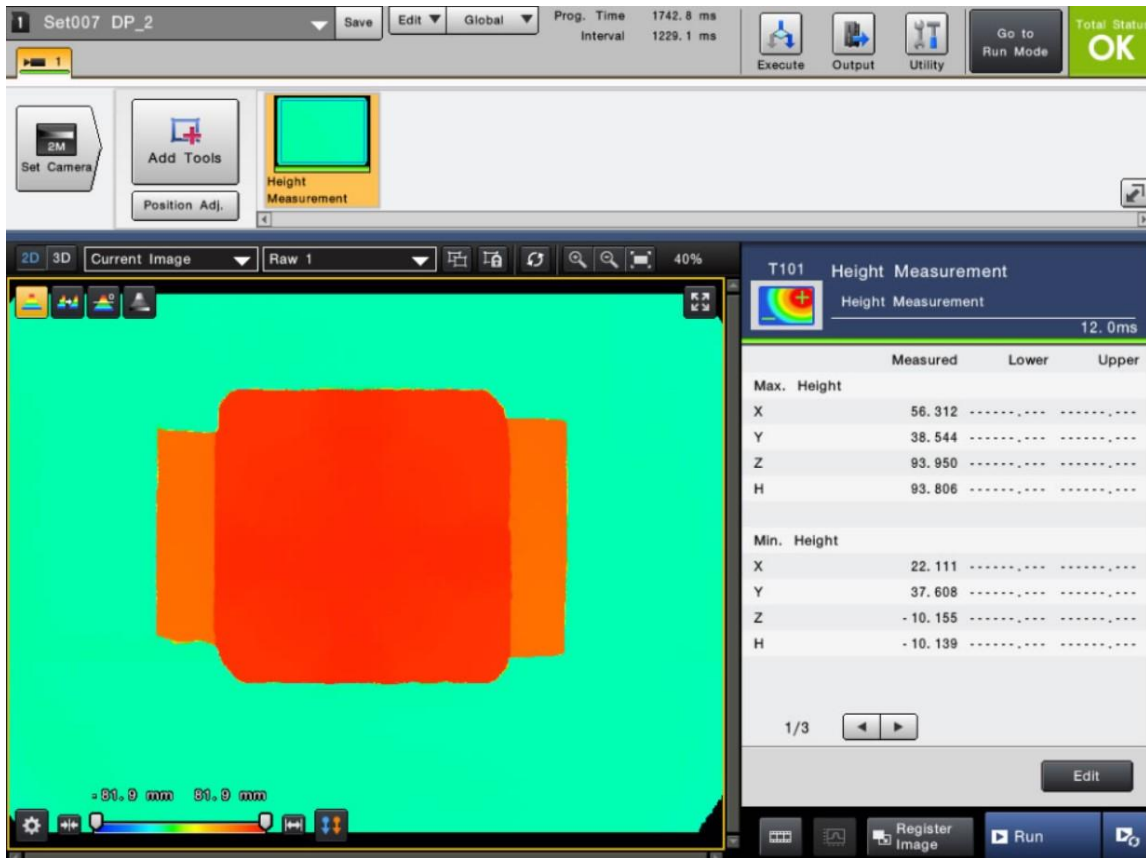




Obr. 37: Výběr funkce Height Measurement

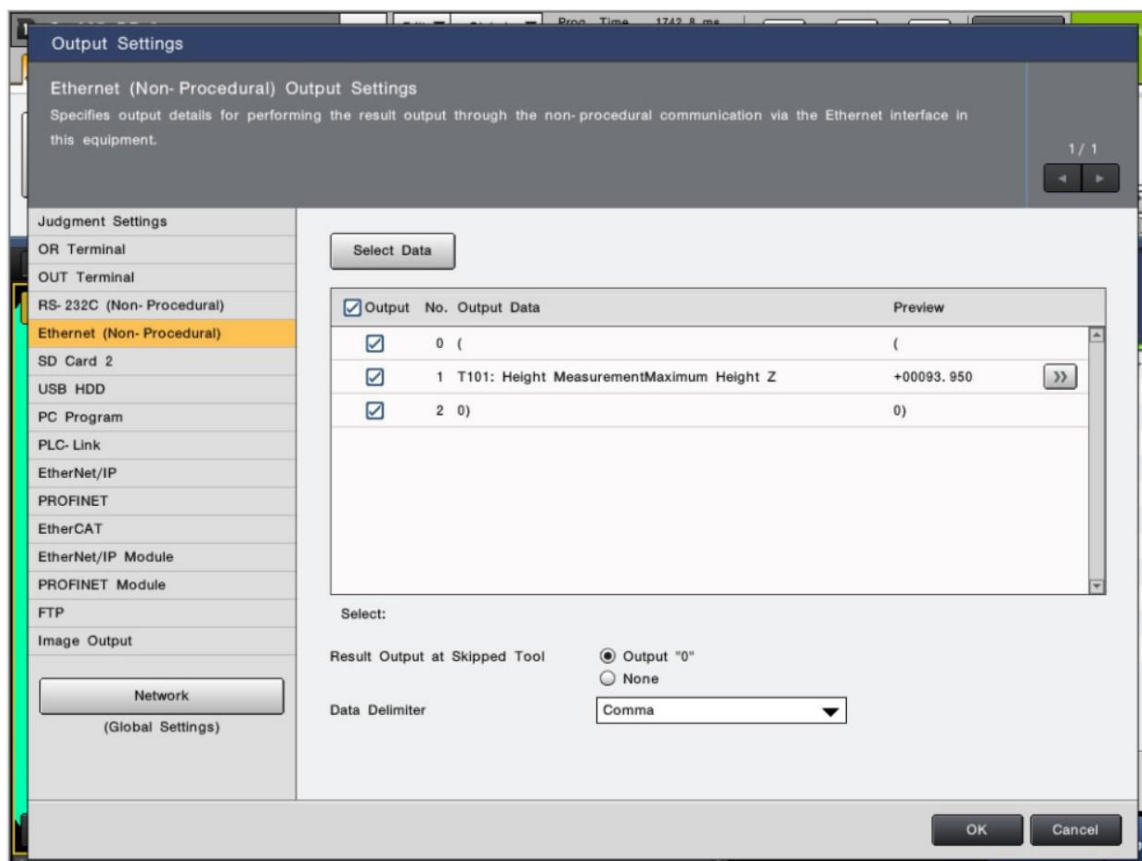
Pro správnou funkci nástroje je zapotřebí změnit mód kamery do „3D Capture“ a projít kalibračním procesem. Ten se skládá ze čtyř kroků. Prvním je nastavení rychlosti závěrky a citlivosti senzoru pro klasický 2D obraz. Snímek v náhledu by měl mít optimální kontrast s co největším množstvím detailů. Další krok spočívá v nastavení citlivosti, intenzity osvětlení a rychlosti závěrky strukturovaného světla. Zde je proces zcela automatický – do vyznačené oblasti v záběru kamery se umístí libovolný objekt a spustí se auto-tuning, ten postupně projde různými kombinacemi nastavení a zvolí to optimální. Následujícím krokem je samotná kalibrace, ta vyžaduje speciální kalibrační desku dodávanou spolu s osvětlovací jednotkou. Na začátku se podložka umístí do vyznačené oblasti a opět se spustí auto-tuning, který doplní nastavení rychlosti závěrky použitou v následujících krocích. Deska má na spodní straně mechanismus, který ji umožňuje lehce vyvýšit (celkem tři úrovně). Je zapotřebí zvlášť pořídit snímek pro každou úroveň. Po získání těchto snímků je kalibrace hotová a zbývá poslední krok, kde je možné provést lehké finální úpravy. Těmito úpravami jsou různé filtrace šumu nebo speciální nastavení, pokud bychom měřili například objekt

s dírami. Na závěr nastavení je možné změnit počet promítaných vzorů z osmi na čtyři s cílem uspořit čas zpracování.



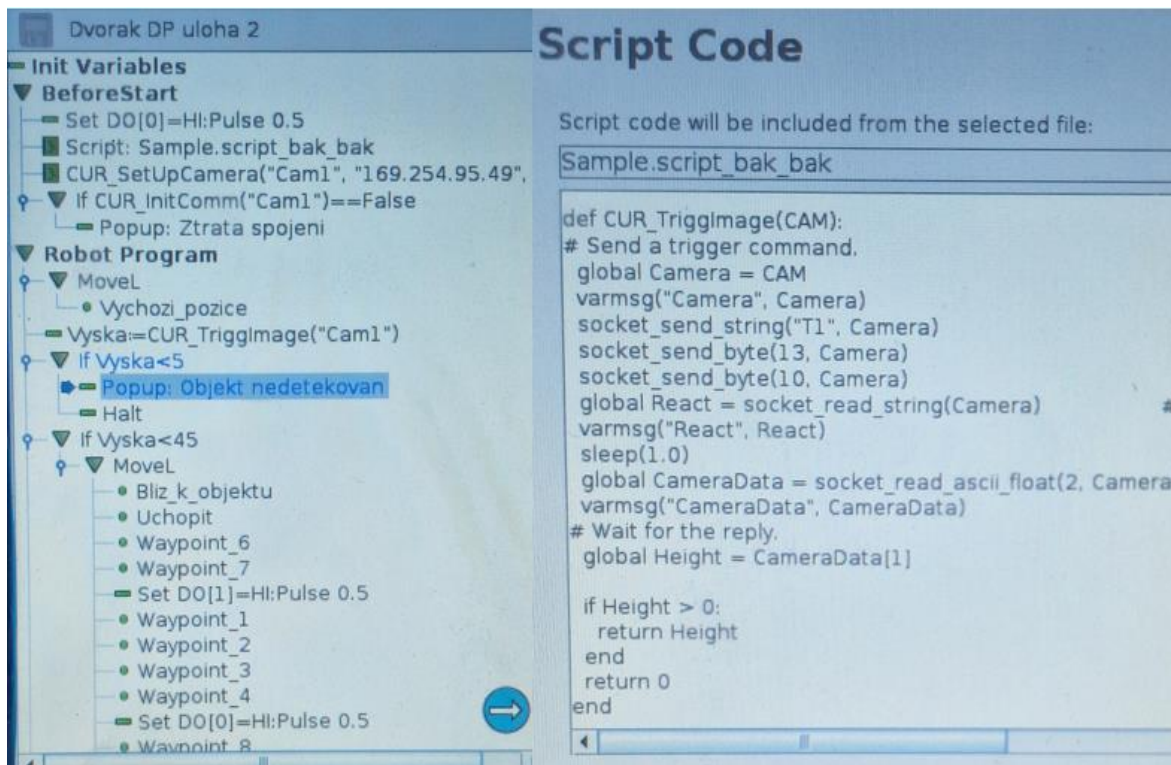
Obr. 38: Výsledek měření – hloubková mapa

V nastavení výstupu dat z vision systému je nutné specifikovat jaká data chceme po Ethernetu odesílat do kontroléru robota. V tomto konkrétním případě budeme odesílat nejvyšší naměřenou výšku objektu v ose Z.



Obr. 39: Řetězec výstupních dat

Co se týče přesnosti měření, odchylka od skutečnosti se pohybovala do  $\pm 4\%$ . Jeden z měřených objektů měl na výšku 93 milimetrů, přičemž vision systém vracel, při opakovaném měření, hodnoty od 91 do 96 milimetrů. Přesnost by bylo možné ještě zvýšit změnou některých hodnot během procesu kalibrace, zejména pokud by se pro kalibraci nepoužíval automatický režim, ale manuální. Pro účely této úlohy bylo ale nutné rozlišit objekty, které se lišily o desítky milimetrů, přesnost byla tedy dostačující.



Obr. 40: Program a skript pro 2.úlohu

Program na začátku otevře kleště efektoru a naváže spojení s kamerou, stejně jako v předchozí úloze. Dále se robot přesouvá na výchozí pozici, odesílá příkaz vision systému pro pořízení snímků scény a vyhodnocení prostorových dat. Vision systém odesílá výslednou hodnotu do kontroléru robota, následuje rozhodovací proces. Pokud je naměřená výška menší než pět milimetrů, předpokládá se, že objekt není v prostoru kamery a program dále nepokračuje. Pokud je změřená výška mezi 5 a 45 milimetry, objekt se považuje za menší a přesouvá se do prostoru vpravo od kamery. Přesahuje-li výška 45 milimetrů, objekt je klasifikován jako větší a robot ho přesouvá vlevo od kamery. V programu je tento rozhodovací proces realizován pomocí příkazu „IF-ELSE“, kde každá z výše popsaných variant má svou posloupnost pohybů a příkazů. Pro příjem dat z vision systému byla upravena funkce „CUR\_TriggerImage“, kterou lze vidět na obr.40.

## 5. Závěr

Cílem této práce bylo navrhnout a realizovat integraci robota UR3 od Universal Robotics a systém 2,5D strojového vidění řady CV-X od firmy Keyence a demonstrovat tuto integraci na vhodné úloze. Původním záměrem bylo vytvořit jednu úlohu, kde by vision systém naváděl robota a zároveň podával prostorové informace o výškách objektů. Během vytváření úlohy bylo však zjištěno, že u použitého systému strojového vidění Keyence je pro navádění robotů a využívání 2,5D funkcí nutno použít odlišné typy programů, které mezi sebou nelze kombinovat. Výsledná praktická část této práce je tedy prezentována v kapitole čtyři a v souladu se zadáním popisuje postup tvorby integrace jako dvě samostatné úlohy, z nichž první demonstruje schopnosti navádění robota vision systémem úlohou typu „Pick“, kde je robot schopný spolehlivě uchopit objekt, který neustále mění svou pozici a orientaci v záběru kamery. Druhá úloha funguje na základě pevně dané pozice objektu bez jakéhokoliv navádění, avšak využívá 2,5D funkcí vision systému, který komunikuje s robotem a vrací mu informace o výškách objektů, robot následně objekty přesouvá na dané pozice dle jejich výšky. Měřicí funkce použité v druhé úloze využívají strukturované světlo, a kromě výšky objektů vrací i jeho pozici v osách X a Y. Za určitých podmínek a při velmi přesné kalibraci by bylo možné orientovat se na základě těchto hodnot, jak ale bylo popsáno v předchozí kapitole, odchylky měření stejného předmětu byly až tři milimetry, což je v technické praxi už poměrně velká hodnota. Musím ale zdůraznit, že zkušenějším uživatelům se kalibrace může podařit lépe a výsledná přesnost může být značně vyšší. Dalším důležitým rozdílem je absence jakékoli zpětné vazby od robota v porovnání se specializovaným typem programu pro navádění použitého v první úloze. Za specifických podmínek a nenáročných požadavcích na přesnost by pravděpodobně bylo možné zkombinovat tyto 2,5D funkce vision systému Keyence a navádění robota do jedné úlohy.

# Literatura

- [1] HOTAŘ, Vlastimil. Úvod do problematiky strojového vidění. Část 1, Základní principy a hardware. Liberec: Technická univerzita v Liberci, 2015. ISBN 978-80-7494-156-6.
- [2] HOTAŘ, Vlastimil. Úvod do problematiky strojového vidění. Část 2, Základy zpracování obrazu. Liberec: Technická univerzita v Liberci, 2015. ISBN 978-80-7494-202-0.
- [3] SZELISKI, Richard. Computer vision: algorithms and applications. Texts in computer science. London: Springer-Verlag, 2011. ISBN 978-1-84882-934-3.
- [4] ŠONKA, Milan; HLAVÁČ, Václav a BOYLE, Roger. Image processing, analysis, and machine vision. Fourth Edition. Australia: Cengage Learning, [2015]. ISBN 978-1-133-59369-0.
- [5] Vision technology – 2.5D [online]. [cit. 2024-05-06]. Dostupné z: <https://www.dvs-technology.com/en/rbc-robotics/component-recognition/vision-25d>
- [6] NOVOTNÝ, František, Vlastimil HOTAŘ, Marcel HORÁK, Marie STARÁ a Michal STARÝ. ÚVOD DO AUTOMATIZACE A ROBOTIZACE VE STROJÍRENSTVÍ [online]. Technická univerzita v Liberci, 2020 [cit. 2024-05-06]. ISBN 978-80-7494-545-8. Dostupné z: <https://publi.cz/download/publication/1275?online=1>
- [7] Basic 3D machine vision techniques and principles [online]. [cit. 2024-05-06]. Dostupné z: <https://www.zivid.com/3d-vision-technology-principles>
- [8] Impact of 3D vision in vision-guided robotic applications [online]. [cit. 2024-05-06]. Dostupné z: <https://blog.zivid.com/impact-of-3d-vision-in-vision-guided-robotic-applications>
- [9] HORÁK, Karel. Popis objektů [online]. [cit. 2024-05-06]. Dostupné z: [http://vision.uamt.feec.vutbr.cz/POV/lectures/08\\_Popis\\_objektu.pdf](http://vision.uamt.feec.vutbr.cz/POV/lectures/08_Popis_objektu.pdf)
- [10] MACH, Lukáš. SIFT: Scale Invariant Feature Transform [online]. [cit. 2024-05-06]. Dostupné z: <https://cgg.mff.cuni.cz/~pepca/ref/SIFT.pdf>

- [11] PALEČEK, Karel. Rozpoznávání vzorů v obraze [online]. Liberec, 2009 [cit. 2024-05-06]. Dostupné z: <https://dspace.tul.cz/server/api/core/bitstreams/7c280432-fb86-4c07-9b3b-4575d2e70e0b/content>. Diplomová práce. TECHNICKÁ UNIVERZITA V LIBERCI. Vedoucí práce Ing. Josef Chaloupka, Ph.D.
- [12] FESSL, Lukáš. Detekce vzorů v obrazech a užití pro analýzu kulturního dědictví [online]. České Budějovice, 2016 [cit. 2024-05-06]. Dostupné z: <https://theses.cz/id/jc79y8/DP2016.pdf>. Diplomová práce. Jihočeská univerzita v Českých Budějovicích. Vedoucí práce Ing. Jiří Jelínek, CSc.
- [13] SLÁDEČEK, Martin. 3D MODEL [online]. Brno, 2017 [cit. 2024-05-06]. Dostupné z: [https://www.vut.cz/www\\_base/zav\\_prace\\_soubor\\_verejne.php?file\\_id=146340](https://www.vut.cz/www_base/zav_prace_soubor_verejne.php?file_id=146340). Diplomová práce. Vysoké učení technické v Brně. Vedoucí práce Ing. Miloslav Richter, Ph.D.
- [14] HLAVÁČ, Václav. Jasové a geometrické transformace [online]. [cit. 2024-05-06]. Dostupné z: <https://people.ciirc.cvut.cz/~hlavac/TeachPresCz/11DigZprObr/18BrightGeomTxCz.pdf>
- [15] HLAVÁČ, Václav a Jan KYBIC. Jasové transformace [online]. [cit. 2024-05-06]. Dostupné z: [https://cw.fel.cvut.cz/old/\\_media/courses/a6m33zmo/jasovetransformace.pdf](https://cw.fel.cvut.cz/old/_media/courses/a6m33zmo/jasovetransformace.pdf)
- [16] HLAVÁČ, Václav. Předzpracování obrazů v prostoru obrazů, operace v lokálním sousedství [online]. [cit. 2024-05-06]. Dostupné z: <https://people.ciirc.cvut.cz/~hlavac/TeachPresCz/11DigZprObr/21ImagPreprocCz.pdf>
- [17] HLAVÁČ, Václav. Hrany, hranové body a ostření obrazu [online]. [cit. 2024-05-06]. Dostupné z: <https://people.ciirc.cvut.cz/~hlavac/TeachPresCz/11DigZprObr/22EdgesInImagesCz.pdf>
- [18] KEYENCE. Vision System with Built-in AI [online]. [cit. 2024-05-06]. Dostupné z: <https://www.keyence.eu/downloads/?mode=tg>

- [19] KEYENCE. LumiTrax™ Vision System Proposal [online]. [cit. 2024-05-06].  
Dostupné z: <https://www.keyence.eu/downloads/?mode=ca>
- [20] KEYENCE. Intuitive Vision System CV-X Series [online]. [cit. 2024-05-06].  
Dostupné z: <https://www.keyence.eu/downloads/?mode=ca>
- [21] KEYENCE. Selecting Your First Vision System [online]. [cit. 2024-05-06].  
Dostupné z: <https://www.keyence.eu/downloads/?mode=tg>
- [22] CHALOUPKA, Josef. Úvod do zpracování obrazů: Prezentace přednášky č. 7 -  
Segmentace obrazu, popis oblastí [online]. In: . [cit. 2024-05-07]. Dostupné z:  
<https://elearning.tul.cz/>
- [23] CHALOUPKA, Josef. Úvod do zpracování obrazů: Prezentace přednášky č. 8 -  
Matematická morfologie [online]. In: . [cit. 2024-05-07]. Dostupné z:  
<https://elearning.tul.cz/>
- [24] I, Cobot: Future collaboration of man and machine [online]. [cit. 2024-05-11].  
Dostupné z: <https://www.themanufacturer.com/articles/i-cobot-future-collaboration-of-man-and-machine/>



## **Seznam příloh**

**Příloha A** video 1.úlohy k obhajobě

**Příloha B** video 2.úlohy k obhajobě