

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Petar Grabar

Zagreb, 2025.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

Automatska kalibracija vrha alata robotu korištenjem tehnika strojnog vida

Mentor:

doc. dr. sc. Filip Šuligoj, dipl. ing.

Student:

Petar Grabar

Zagreb, 2025.

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Zahvaljujem se mentoru doc.dr.sc Filipu Šuligoju i djelatnicima CRTA-e na ukazanoj pomoći i savjetima tokom izrade ovog rada. Želim se također zahvaliti i svim djelatnicima i asistentima fakulteta na ugodnom boravku i suradnji tokom diplomskog studija. Zahvaljujem se svim prijateljima i kolegama na suradnji i nezaboravnim iskustvima tokom trajanja studija. Posebno se zahvaljujem svojim roditeljima i braći te djevojci na podršci koju si mi ukazali tijekom cjelokupnog školovanja.

Petar Grabar



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE



Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za diplomske ispite studija strojarstva za smjerove:
Proizvodno inženjerstvo, inženjerstvo materijala, industrijsko inženjerstvo i menadžment,
mehatronika i robotika, autonomni sustavi i računalna inteligencija

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa: 602 - 04 / 25 - 06 / 1	
Ur.broj: 15 - 25 -	

DIPLOMSKI ZADATAK

Student: **Petar Grabar**

JMBAG: 0035220823

Naslov rada na hrvatskom jeziku: **Automatska kalibracija vrha alata robota korištenjem tehnika strojnog vida**

Naslov rada na engleskom jeziku: **Automatic calibration of robot tool center point using vision-based techniques**

Opis zadatka:

Cilj ovog rada je razviti sustav za automatsku kalibraciju vrha alata industrijskog robota koristeći tehnike strojnog vida s primjenom dvije 2D kamere. Sustav je dizajniran s ciljem omogućavanja preciznog određivanja i korekcije položaja i orijentacije vrha alata robota, čime se značajno povećava točnost i ponovljivost izvedbe robotskih operacija. U okviru ovog zadatka obuhvatit će se implementacija algoritama za analizu slike te upravljačkih metoda za iterativno podešavanje parametara kalibracije.

Glavni zadaci rada uključuju:

- Optimalno postavljanje kamera tako da omogućavaju kalibraciju u radnom prostoru robota.
- Korištenje algoritama strojnog vida za obradu slike za preciznu detekciju položaja i orijentacije vrha alata.
- Komunikacija s upravljačkom jedinicom robota i slanje naredbi radi kalibracije vrha alata.
- Kontrola kretanja robota radi utvrđivanja i verifikacije kalibracijskih parametara.
- Validacija kalibracijskog sustava testiranjem s različitim robotskim alatima kako bi se osigurala robustnost i pouzdanost.

U radu je potrebno navesti korištenu literaturu i eventualno dobivenu pomoć.

Zadatak zadan:

Datum predaje rada:

Predviđeni datumi obrane:

28. studeni 2024.

30. siječnja 2025.

6., 7. i 10. veljače 2025.

Zadatak zadao:

Predsjednik Povjerenstva:

Doc.dr.sc. Filip Šuligoj

Prof. dr. sc. Ivica Garašić

SADRŽAJ

SADRŽAJ	I
POPIS SLIKA	III
POPIS TABLICA.....	V
POPIS OZNAKA	VI
SAŽETAK.....	VII
SUMMARY	VIII
1. UVOD.....	1
2. VIZIJSKI SUSTAVI	2
2.1. Kamera	2
2.2. Softver vizijskih sustava (OpenCV)	3
2.3. Vrste vizijskih sustava	5
2.3.1. 2D vizijski sustavi.....	5
2.3.2. 3D vizijski sustavi.....	6
2.4. Kalibracija vizijskih sustava	7
3. OPREMA ZA RAZVOJ VIZIJSKOG SUSTAVA	9
3.1. Jetson Nano komplet za razvijanje.....	9
3.2. Kamera IMX 219-120.....	12
3.3. Integracija vizijskog sustava	13
4. PROJEKTIRANJE KALIBRACIJSKOG UREĐAJA	14
4.1. Kućište	14
4.2. Kalibracijska kutija	15
5. EKSPERIMENT ZA ODABIR BOJE POZADINE	19
6. KALIBRACIJA KAMERA.....	24
7. RAZVOJ ALGORITMA ZA DETEKCIJU VRHA ALATA ROBOTA.....	27
8. ALGORITAM KALIBRACIJE ALATA	34
8.1. Kalibracija pozicije vrha alata robota	34
8.2. Kalibracija orijentacije vrha alata robota	35
9. Komunikacija vizijskog sustava i robota.....	37
9.1. Socket komunikacija	37
9.2. Tijek komunikacije - server	38
9.3. Tijek komunikacije - klijent.....	39
9.3.1. Program robota.....	40
9.3.1.1. Inicijalizacija programa	40
9.3.1.2. Robot program	40
9.3.1.3. Thread komunikacije	44
10. IMPLEMENTACIJA I REZULTATI KALIBRACIJE ALATA	46
10.1. Postav za validaciju kalibracije.....	46
10.2. Prvi alat – kalibracijski šiljak.....	46

10.3. Drugi alat – svrdlo.....	50
10.4. Treći alat – vakuumska hvataljka.....	53
10.5. Četvrti alat – Alat pod kutem.....	55
11. ZAKLJUČAK.....	58
LITERATURA.....	59
PRILOZI.....	60

POPIS SLIKA

Slika 1. Pinhole model kamere[4]	2
Slika 2. Prikaz različitih zapisa slike.....	3
Slika 3. Utjecaj korištenja filtera na sliku uz dodavanje sol i papar šuma	4
Slika 4. Segmentacija korištenjem praga	4
Slika 5. Canny edge detekcija rubova na slici.....	5
Slika 6. 2D kamera [4]	6
Slika 7. Vrste 3D vizijskih sustava [4]	6
Slika 8. Vrste kalibracijskih ploča [4].....	7
Slika 9. Konfiguracije robot-kamera [4]	8
Slika 10. Jetson Nano sa priključcima [1].....	9
Slika 11. Karakteristike Jetson nano-a [1]	10
Slika 12. Korisničko sučelje Jetson nano-a [1]	11
Slika 13. Prikaz slika s kamera i sučelje Qt creator-a	11
Slika 14. IMX219-120 kamera [3]	12
Slika 15. Karakteristike kamere IMX 219-120 [3]	12
Slika 16. Vizijski sustav sa svim dijelovima.....	13
Slika 17. Donji dio (lijevo) i gornji dio (desno) kućišta.....	14
Slika 18. Prikaz načina spajanja kućišta.....	15
Slika 19. Dizajn kalibracijske kutije	16
Slika 20. Način spajanja kalibracijske kutije s kućištem	16
Slika 21. Cijeli sklop vizijskog sustava i kalibracijske kutije	17
Slika 22. Cijeli sklop vizijskog sustava i kalibracijske kutije	17
Slika 23. Simulirano dohvaćanje scene prve kamere	18
Slika 24. Simulirano dohvaćanje scene druge kamere	18
Slika 25. Postav za eksperiment	19
Slika 26. Predmeti segmentacije	19
Slika 27. Postupak segmentacije	20
Slika 28. Rezultati segmentacije bijele boje.....	20
Slika 29. Rezultati segmentacije crvene boje.....	21
Slika 30. Rezultati segmentacije plave boje.....	21
Slika 31. Rezultati segmentacije zelene boje	22
Slika 32. Rezultati segmentacije žute boje.....	22
Slika 33. Svi rezultati segmentacije	23
Slika 34. Kalibracijska ploča sa 8 x 6 kvadrata sa stranicama od 25mm.....	24
Slika 35. Kalibracijske slike za jednu od kamera.....	25
Slika 36. Prikaz detekcije rubova na zadnjoj kalibracijskoj slici.....	26
Slika 37. Intrinzični parametri jedne od kamera	26
Slika 38. Alat robota u Lab formatu (1), kanal L(2), kanal a(3), kanal b(4).....	27
Slika 39. Filtrirana slika korištenjem bilateralnog filtera.....	28
Slika 40. Segmentacija alata korištenjem OTSU metode	28
Slika 41. Segmentacija alata korištenjem adaptivnog praga	29
Slika 42. Pronađene konture alata OTSU(gornje) i adaptivni prag(donje)	29
Slika 43. Detekcija vrha alata OTSU(gornje) i adaptivni prag(donje).....	30
Slika 44. Udaljenost vrha alata od optičkih osi prema koordinatnim sustavima kamera.....	31
Slika 45. Položaj kalibracijske kutije u odnosu na bazu robota	32
Slika 46. Odnos koordinatnog sustava kamere 0 s osima baze robota (y, z)	32
Slika 47. Odnos koordinatnog sustava kamere 1 s osima baze robota (x, z)	33
Slika 48. Vrijednosti udaljenosti alata robota od optičkih centara.....	33

Slika 49. Arhitektura socket komunikacije[9].....	37
Slika 50. Tijek komunikacije na vizijskom sustavu	38
Slika 51. Dijagram izvođenja programa robota(klijenta).....	39
Slika 52. Inicijalizacija programa robota	40
Slika 53. Dio programa koji se izvodi ako se zatvori socket	40
Slika 54. Dio programa koji se izvodi za kontrol_var_1 = 1	41
Slika 55. Dio programa koji se izvodi kada je kontrol_var_1 = 2	41
Slika 56. Dio programa za mijenjanje orijentacije alata robota	42
Slika 57. Dio programa koji se izvodi za kontrol_var_1 = 3	43
Slika 58. Dio programa koji se izvodi za var_kraj = 1.....	44
Slika 59. Thread za komunikaciju s serverom	44
Slika 60. Postav na robotskoj stanici.....	46
Slika 61. Alat 1 - kalibracijski šiljak	46
Slika 62. Dovedeni alat u prostor vidnog polja kamera	47
Slika 63. Set slika za prvu kalibracijsku točku s obje kamere - alat 1	47
Slika 64. Set kalibracijskih točaka s obje kamere – prvi alat.....	48
Slika 65. Slike sa obje kamere za poravnanje osi alata sa z osi baze robota - prvi alat.....	49
Slika 66. Drugi alat - svrdlo	50
Slika 67. Set kalibracijskih točaka s obje kamere - drugi alat.....	51
Slika 68. Slike sa obje kamere za poravnanje osi alata sa z osi baze robota - drugi alat.....	52
Slika 69. Treći alat - vakuumska hvataljka	53
Slika 70. Slike sa obje kamere za poravnanje osi alata sa z osi baze robota - treći alat	53
Slika 71. Set kalibracijskih točaka s obje kamere - treći alat.....	54
Slika 72. Četvrti alat – improvizirani alat	55
Slika 73. Set kalibracijskih točaka s obje kamere - četvrti alat.....	56
Slika 74. Slike sa obje kamere za poravnanje osi alata sa z osi baze robota - četvrti alat	57

POPIS TABLICA

Tablica 1. Rezultati kalibracije - prvi alat	50
Tablica 2. Rezultati kalibracije – drugi alat	52
Tablica 3. Rezultati kalibracije - treći alat	55
Tablica 4. Rezultati kalibracije - četvrti alat	57

POPIS OZNAKA

Oznaka	Jedinica	Opis
x_{mm}	[mm]	Udaljenost vrha alata u milimetrima od optičke osi
x_p	[piksela]	Udaljenost vrha alata u pikselima od optičke osi
Z	[mm]	Dubina za pretvaranje piksela u milimetre
f	[piksela]	Fokalna duljina
T_{1i}, T_{1j}		Matrica transformacije od baze do prirubnice
Q		Matrica transformacije od vrha alata do prirubnice
TCP_x, TCP_y, TCP_z	[mm]	Vrijednosti vrha alata
a_{ij}, b_{ij}		Članovi matrica transformacija
\bar{t}_i	[mm]	Translacijski dio matrice transformacije
R_i	[rad]	Rotacijski dio matrice transformacije

SAŽETAK

U ovom radu razvijeno je korištenje vizijskog sustava pri kalibraciji alata robota. Ključni dijelovi razvijenog vizijskog sustava su kamera, hardver za procesuiranje slika i softver koji sadrži funkcije za obradu slike. U uvodu je dano osnovno o vizijskim sustavima i kameri te su prikazane vrste vizijskih sustava i vrste konfiguracija robota i kamera. Nadalje je pokazana oprema koja se koristi za procesuiranje slike i dohvaćanje scene. Za procesiranje i obradu slike korištena je Jetson Nano platforma za razvijanje s instaliranim OpenCV-om za obradu slike, a za dohvaćanje scene korištene su dvije IMX219-120 MIPI kamere. Kako bi se na sustavu moglo raditi potrebno je na njega priključiti periferiju što omogućavaju priključci Jetson platforme. Da bi se oprema zaštitila napravljeno je kućište za Jetson platformu i napravljena je kalibracijska kutija koja je odvojena od procesne jedinice radi sigurnosti i robusnosti segmentacije. Dalje je proveden eksperiment odabira boje pozadine scene kako bi se odredila najprikladnija boja za dobar kontrast u odnosu na alat. Odabirom optimalne boje pozadine omogućuju se robusniju rezultati segmentacije, koja je ključna za dobru kalibraciju. Kako bi se sa kamera dobila dobra informacija potrebno ju je kalibrirati pa je kratko objašnjena te provedena kalibracije kamere. Pokazan je način segmentacije alata od podloge sa uključenim svim radnjama obrade slike. Definiran je način detekcije vrha alata robota i izračuni gibanja robota kako bi se alat nalazio u sjecištu optičkih osi kamera. Za izračunavanje kalibracije alata odnosno transformacije koordinatnog sustava vrha alata robota u odnosu na prirubnicu robota, što je i cilj kalibracije, prikazani su korišteni algoritmi. Pokazan je i način komunikacije robota i vizijskog sustava gdje je robot klijent, a vizijski sustav server u socket arhitekturi komunikacije. Dani je programski kod na robotu za proces kalibracije alata korištenjem vizijskog sustava. Na kraju su prikazani rezultati kalibracije korištenjem vizijskog sustava i dana je usporedba takvog načina kalibracije alata robota naspram klasične metode kalibracije robotskog alata.

Ključne riječi: kalibracija alata robota, vizijski sustav, kamere, komunikacija

SUMMARY

In this paper, the use of a vision system during the calibration of robot tools was developed. The key parts of the developed vision system are the camera, the image processing hardware, and the software that contains the image processing functions. In the introduction, the basics about vision systems and cameras are given, and the types of vision systems and the types of configurations of robots and cameras are presented. Furthermore, the equipment used for image processing and scene retrieval is presented. A Jetson Nano development platform with OpenCV installed for image processing is used for processing of images, and two IMX219-120 MIPI cameras are used for scene acquisition. In order to be able to work on the system, it is necessary to connect the peripherals that enable the connections of the Jetson platform. In order to protect the equipment, a case was made for the Jetson platform and a calibration box was made that was separated from the processing unit for the safety and robustness of the segmentation. Next, an experiment was conducted to select the background color of the scene to determine the most suitable color for good contrast with the tool. Choosing the optimal background color enables robust segmentation of the results, which is essential for good calibration. In order to get good information from the camera, it needs to be calibrated, so the camera calibration was briefly explained and carried out. The method of segmentation of the tool from the background with all the actions of image processing included is shown. The method of detecting the tooltip of the robot and calculating the motion of the robot so that the tool is located at the intersection of the optical axes of the cameras are defined. Algorithms used to calculate the tool calibration, i.e. the transformation of the coordinate system of the tip of the robot tool in relation to the robot flange, which is the goal of the calibration, are presented. The method of communication between the robot and the vision system is also shown, where the robot is the client and the vision system is the server in the socket communication architecture. The programming code on the robot is given for the tool calibration process using the vision system. At the end, the results of the calibration using the vision system are presented and a comparison of this method of robot tool calibration versus the classic method of robot tool calibration is given.

Key words: calibration of robot tools, vision system, cameras, communication

1. UVOD

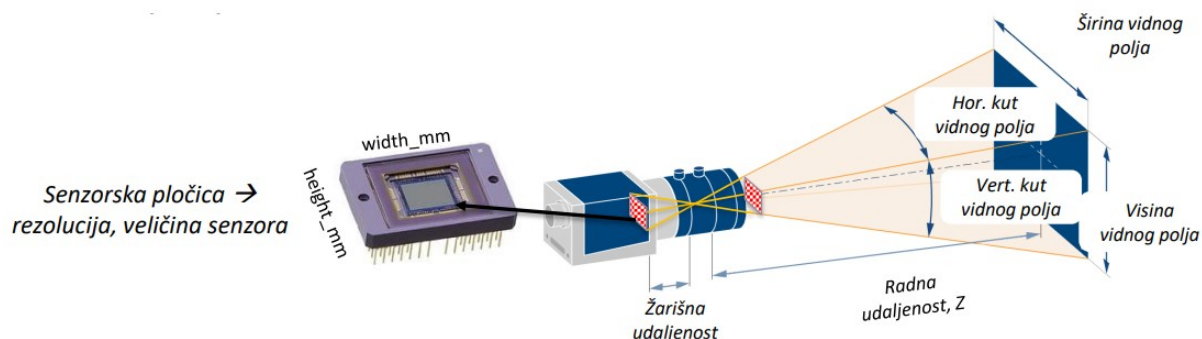
U ovom radu prikazati će se razvoj vizijskog sustava koji ima za zadatak napraviti kalibraciju alata robota. Korištenjem vizijskog sustava za kalibraciju alata robota dobiva se na brzini i jednostavnosti procesa kalibracije, a kalibracija ima veću preciznost zbog isključivanja faktora ljudske greške. U radu će se prikazati dijelovi vizijskog sustava kao i alati korišteni za razvoj, algoritam obrade slike za dohvaćanje vrha alata robota, komunikacija robota i vizijskog sustava i kalibracija alata robota. Vizijski sustavi su se razvili zbog svoje mogućnosti dohvaćanja scene i davanja funkcije istoj. Najveći iskorak dobiven je time što se slika pretvara u informaciju. Vizijski sustavi su omogućili rješenje velikog broja različitih problema obrađivanjem slike ili videa. U današnje vrijeme se sve više koriste baš iz tog razloga. Glavni dijelovi vizijskog sustava su kamera, softver i hardver kojim se radi procesuiranje slike. U ovom slučaju vizijski sustav nije vezan na robotsku ruku što znači da su robot i vizijski sustav u tzv. oko-ruka (eng. Eye to hand) konfiguraciji. Za zadatak kao što je kalibracija alata robota vizijskim sustavom potrebno je ostvariti komunikaciju između robota i vizijskog sustava što se odrađuje preko nekog od komunikacijskih protokola. Pošto je vizijski sustav odvojen od robota potrebno je napraviti kućište kojim će se osigurati zaštita komponenata sustava te osigurati pozicioniranje kamera kako bi one mogle obavljati funkciju. U radu je prikazano osnovno o vizijskim sustavima, navedena je oprema i software koji sustav koristi, prikazana je konfiguracija eye to hand, razvoj kućišta i kalibracijske kutije. Prikazani je način obrade slike za detekciju vrha alata robota. Na kraju su prikazani rezultati kalibracije za različite robotske alate. Kako bi proizvod, u čiji je proces izrade uključen robot, bio visoke kvalitete nužna je precizna kalibracija alata robota. Roboti se koriste u sve više grana industrije zbog automatizma radnji koje robot može obaviti sa velikom ponovljivosti. Robotika je zastupljena u primjenama koje zahtijevaju veliku ponovljivost kao što su npr. automobilska industrija, medicina i sl. Automatska kalibracija alata robota je pritom povoljna zbog pojednostavljenja procesa kalibracije i osiguranja točnijih rezultata kalibracije.

2. VIZIJSKI SUSTAVI

Vizijski sustavi su sustavi koji koriste prihvaćanje scene i obradu slike scene kako bi riješili neki problem. Kako je navedeno u uvodu, glavne komponente sustava su kamera koja služi kao senzor scene koja se promatra, hardver kojim se odrađuje procesuiranje slike te softver koji služi za definiranje načina obrade slike i davanje slici određenih vrijednosti koje su od interesa za rješavanje problema. Najmanji dio slike jest piksel kojem se pridodaje numerička vrijednost kako bi se odredila boja tj. intenzitet piksela. Ujedno je to prvi i najveći korak koji se morao napraviti kako bi vizijski sustavi mogli rješavati probleme. Nakon što je pikselu dodana numerička vrijednost slika se može obrađivati korištenjem raznih algoritama koji izmjenjuju sliku, a na temelju izmjena daju joj bolju funkcionalnost za rješavanje problema. Tako postoje najznačajnije metode obrade slike kao što su grayscale konverzija, prag slike (eng. thresholding), detekcija rubova, filtriranje, segmentacija i još mnoge druge.

2.1. Kamera

Kamera je uređaj koji služi za prihvaćanje slike. Ona je objašnjena na modelu otvor kamera (eng. pinhole camera) koji je prikazan na Slika 1.



Slika 1. Pinhole model kamere[4]

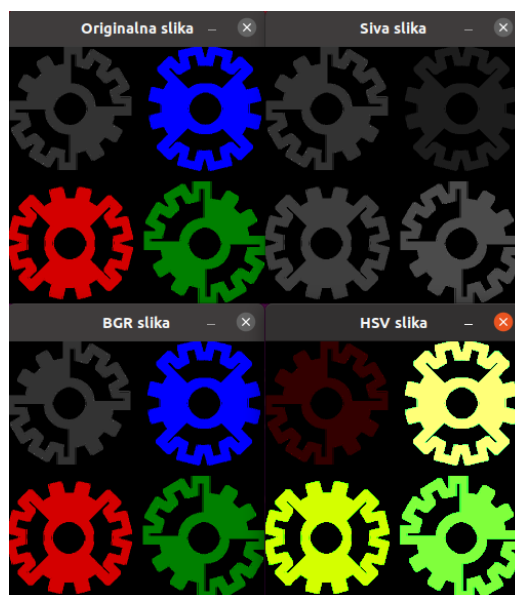
Pinhole model kamere je model koji radi na principu prolaska svjetlosnih zraka kroz otvor kamere i projicira invertiranu sliku scene na senzoru kamere koji svjetlost pretvara u električni signal. Svaki od senzora ima mrežu fotodioda kojima se mjeri intenzitet svjetla i pridodaje njegova vrijednost svakom od piksela. Kamera može zapisati i boju scene ukoliko je opremljena tzv. Bayerovim filterom, kojim se pridodaju vrijednosti intenziteta crvene, zelene i plave boje. Svaka takva slika koja se može shvatiti kao mreža piksela tada se zapisuje u memoriju. Kod kamera postoji nekoliko najvažnijih parametara koji su prikazani na Slika 1. Žarišna udaljenost predstavlja udaljenost senzora kamere i središta leće objektiva. Što je ona manja omogućuje

veći kut vidnog polja, a što je veća omogućuje usmjereniju sliku i veće povećanje objekta. Veličina senzora, odnosno širina i duljina senzora, je također važan parametar jer što je senzor veći ima bolja svojstva i dolazi se do kvalitetnije slike. Kvaliteta slike izražava se rezolucijom kamere što je veličina koja pokazuje koliko je kamera sposobna detektirati sitne detalje. Širina vidnog polja također je bitna kako bi se ona mogla rekonstruirati i provjeriti zadovoljava li kamera potrebe da se detektira cijela scena od interesa.

2.2. Softver vizijskih sustava (OpenCV)

U ovom radu korištena je OpenCV biblioteka koja sadrži funkcije za obradu slike i metode računalnog vida. Pomoću ove biblioteke mogu se rješavati razni zadaci vizijskog sustava kao što su prepoznavanje objekata, računalna fotogrametrija, strojno učenje i sl. U ovom poglavlju će biti prikazane neke od najbitnijih metoda obrade slike koje se koriste u područjima koja zahtijevaju vizijske sustave.

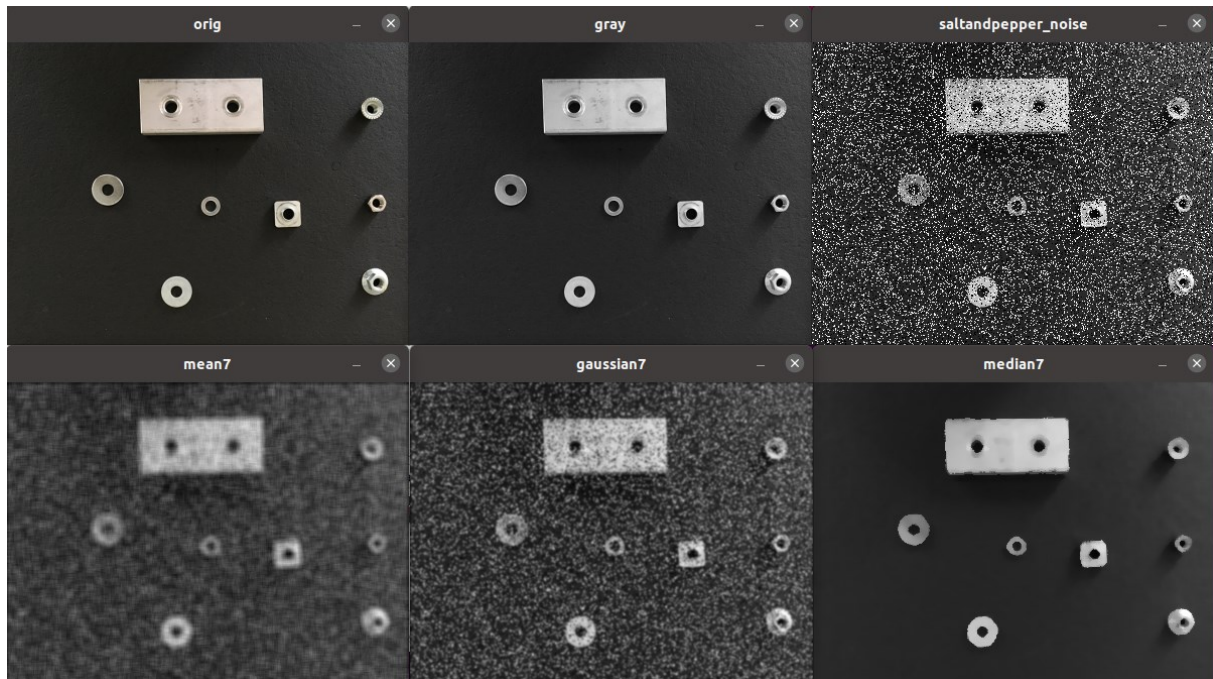
Pretvaranje slike u druge oblike zapisa kao što su grayscale (prikaz u sivoj skali), BGR slika i HSV slika koriste se za daljnju obradu slike. Slika 2 prikazuje pretvaranje slike u drugi oblik zapisa.



Slika 2. Prikaz različitih zapisa slike

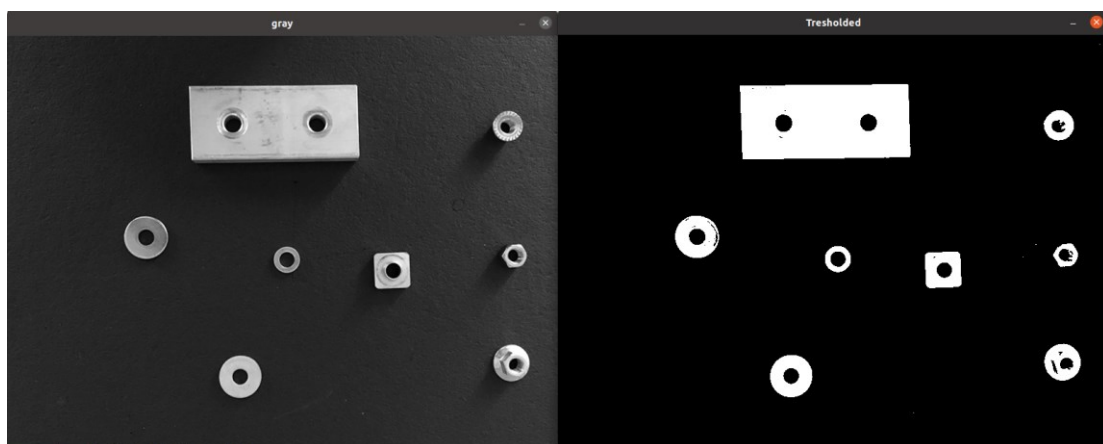
Ova metoda koristi se za dobivanje takvog zapisa slike kakav je potreban za neki zadatak. Siva slika koristi se kako bi se smanjila količina informacija o slici te se lakše obrađuje od onih u kojima su uključene boje. U slučaju da se treba detektirati određena boja koriste se BGR i HSV oblici zapisa slike. Druga bitna metoda koja se koristi jest zamućivanje slike i filtriranje, a služe kako bi se sa slike maknuo šum i za zaglađivanje slike zamućivanjem što poboljšava sliku u slučaju potrebe za detekcijom rubova. Postoji više filtera, a neki od njih prikazani su na Slika 3

sa prikazom kako koji filter utječe na originalnu sliku ako je slika degradirana tzv. sol i papar šumom.



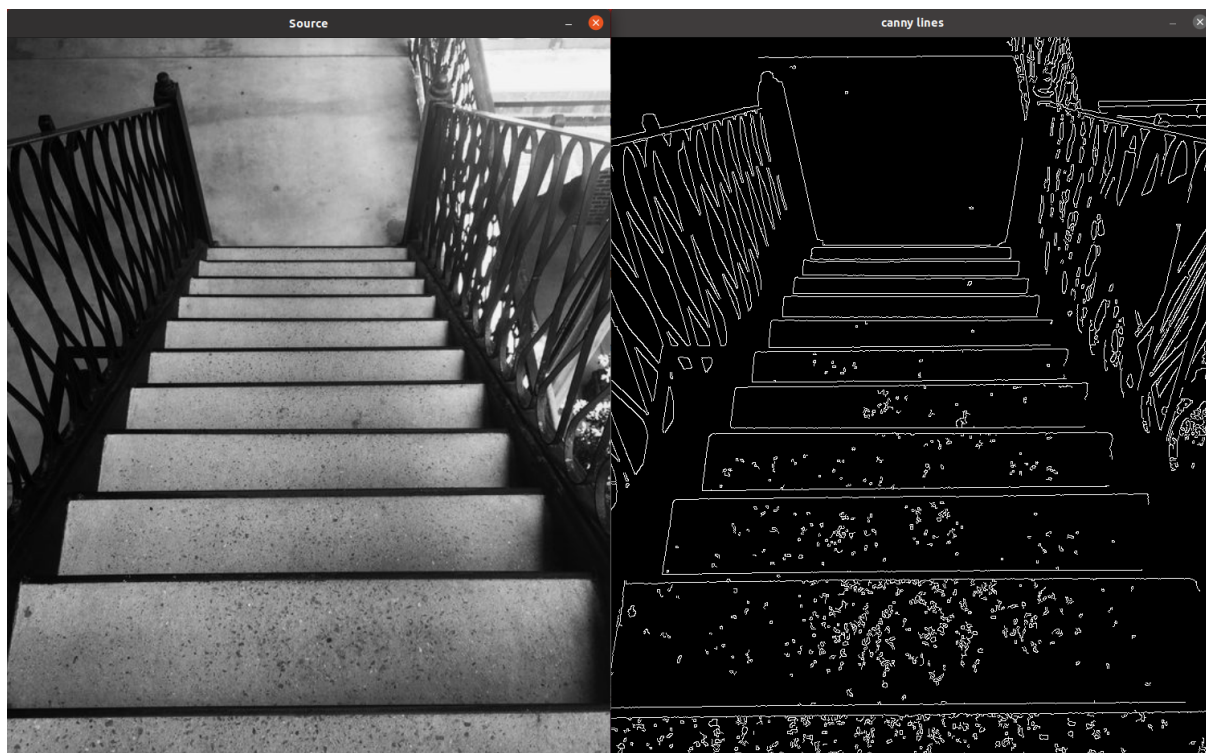
Slika 3. Utjecaj korištenja filtera na sliku uz dodavanje sol i papar šuma

Kao što je vidljivo sa slike potrebno je odabrati odgovarajući filter kako bi se dobila što bolja slika bez narušavanja rubova objekata od interesa. Nadalje vrlo je bitna i metoda segmentacije slike u kojoj se slika dijeli na više segmenata sa ciljem da se slika pojednostavi za analizu i lakše identificiranje granice i objekata na slici. Segmentacija se također može napraviti na više načina kao što su npr. postavljanje praga i klasteriranje. Slika 4 prikazuje segmentaciju slike korištenjem praga.



Slika 4. Segmentacija korištenjem praga

Ovdje se segmentacija radi tako da se slika dijeli na dva dijela odnosno dva intenziteta piksela te se time dobiva slika na kojoj bijela boja prikazuje objekte, a crna pozadinu. Jedan od najčešćih zadataka vizijskog sustava jest detekcija rubova, a algoritmi tj. metode koje se koriste najčešće traže mjesta nagle promjene intenziteta piksela. Najčešće korišten algoritam je eng. Canny edge algoritam koji je prikazan na Slika 5.



Slika 5. Canny edge detekcija rubova na slici

Ovdje je dan kratki pregled najbitnijih metoda u OpenCV-u kojima se služe vizijski sustavi, međutim postoji ih još nekoliko. Kombinacijom ovih bazičnih metoda dolazi se do složenijih aplikacija računalnog vida kao što je kalibracija alata robota.

2.3. Vrste vizijskih sustava

2.3.1. 2D vizijski sustavi

2D vizijski sustavi koriste kamere za hvatanje i analizu dvodimenzionalnih slika, a primarna im je svrha omogućiti prepoznavanje objekta prema obliku, boji ili nekom uzorku. Kamera dohvaća sliku scene ili objekta i zatim se, metodama računalnog vida, slika obrađuje kako bi 2D vizijski sustav napravio svoju funkciju. Neki od tipičnih zadataka 2D vizijskih sustava su inspekcija proizvoda, sortiranje, čitanje bar kodova, prepoznavanje teksta ili objekta, identifikacija nepravilnosti proizvoda i sl.

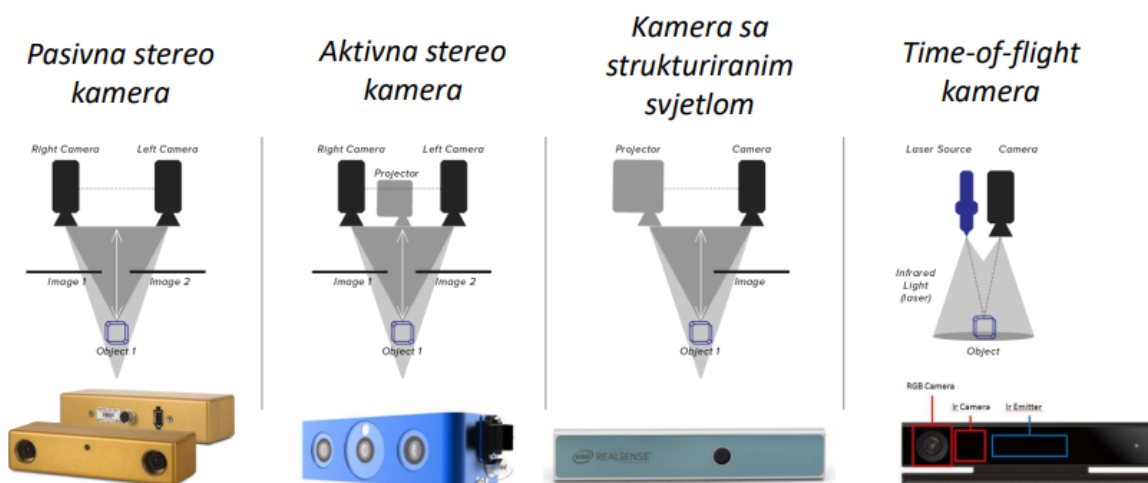
Slika 6 prikazuje tip 2D kamere koju takvi vizijski sustavi koriste.



Slika 6. 2D kamera [4]

2.3.2. 3D vizijski sustavi

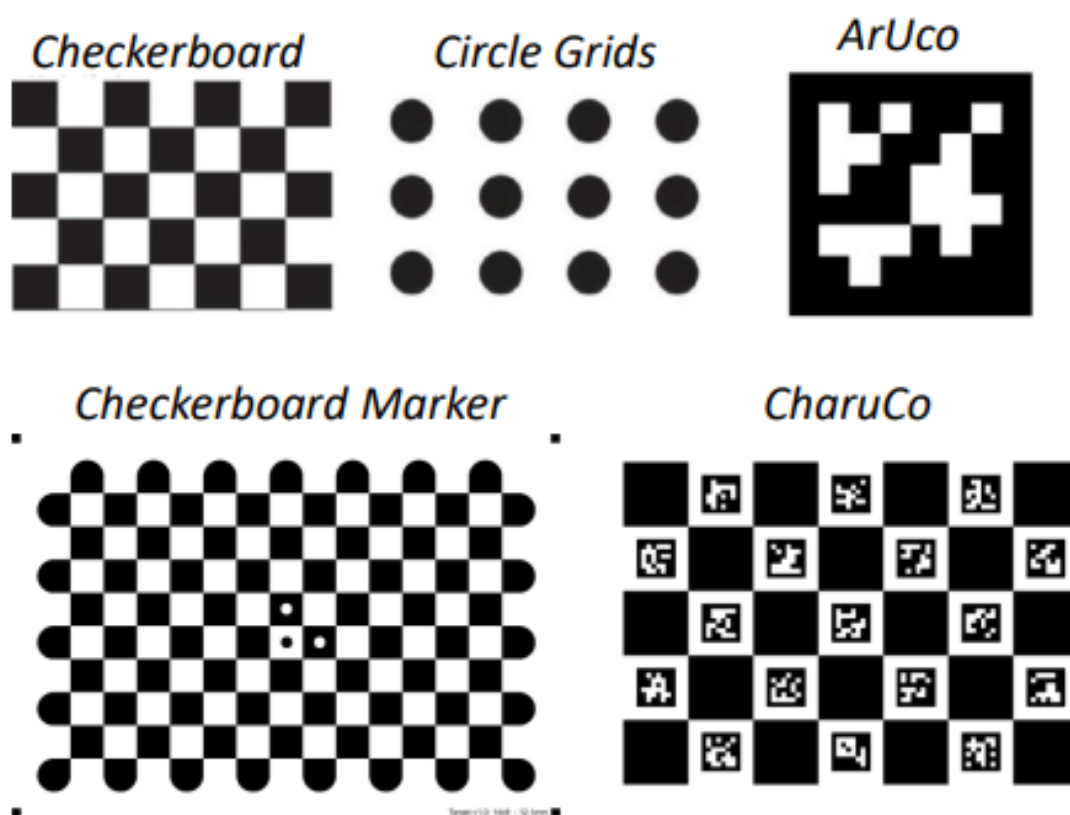
3D sustavi se koriste za dohvaćanje trodimenzionalne informacije o sceni ili objektu. Predostim je što se dobiva informacija o dubini, na temelju koje se mogu rješavati zahtjevniji zadaci kao što su hvatanje objekta u robotici i određivanje pozicije nekog objekta. Skup podataka koji se dobije korištenjem 3D vizijskih sustava tipično se naziva oblak točaka. Ovi sustavi obično koriste tehnike poput stereo vizije, laserskog skeniranja, strukturnog svjetla ili tzv. TOF (eng. time-of-flight) kamera kako bi prikupili podatke o dubini. Na Slika 7 prikazane su spomenute vrste 3D vizijskih sustava.



Slika 7. Vrste 3D vizijskih sustava [4]

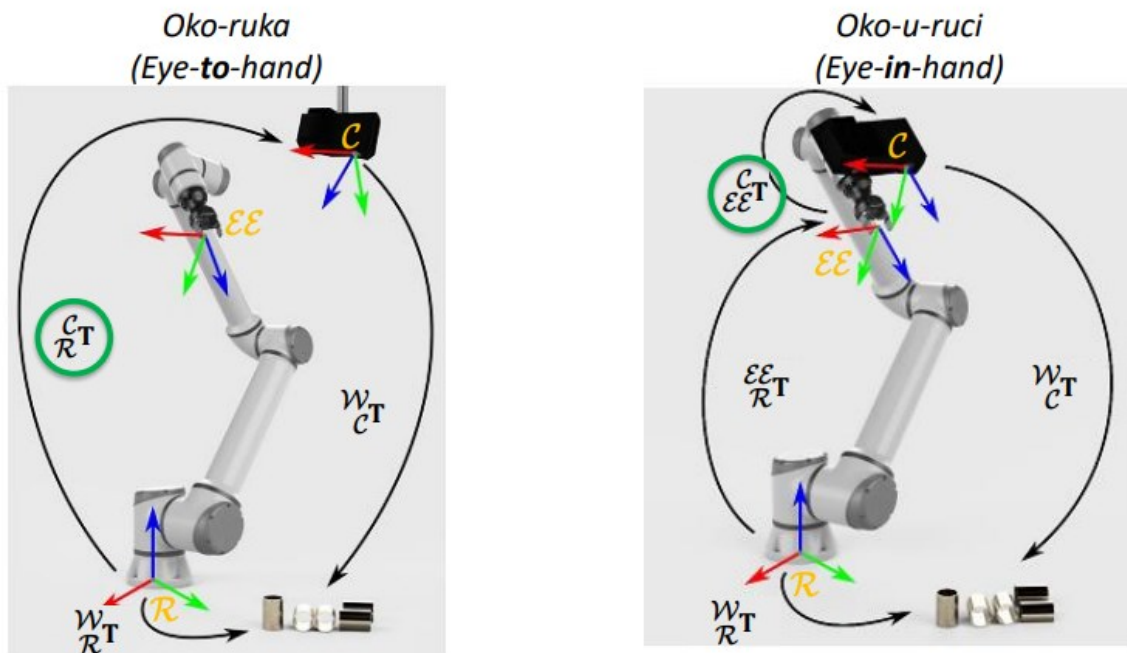
2.4. Kalibracija vizijskih sustava

Važno je napomenuti kako se svaki vizijski sustav treba kalibrirati. Kalibracija vizijskih sustava podijeljena je na dvije vrste, a to su ekstrinzična i intrinzična kalibracija. Cilj kalibracije je utvrditi intrinzične i ekstrinzične parametre kamere. Intrinzični se odnose na unutarnje karakteristike kamere kao što su fokalna duljina i koeficijenti distorzije leće i optičkog centra. Ekstrinzični parametri odnose se na poziciju i orijentaciju kamere u odnosu na radni prostor. Za kalibraciju se koriste specifični uzorci, najčešće nalik šahovskoj ploči. Slika 8 prikazuje različite kalibracijske ploče koje se koriste.



Slika 8. Vrste kalibracijskih ploča [4]

U robotici postoje dvije vrste konfiguracija kamera i robota. Prva je kada je kamera odvojena od robota i nalazi se na fiksnom mjestu u odnosu na robot, a naziva se oko-ruka konfiguracija (eng eye-to-hand). Druga konfiguracija jest ako je kamera pričvršćena za robotsku ruku te se s njom pomiče, a naziva se oko-u-ruci (eng. eye-in-hand). Slika 9 prikazuje dvije vrste konfiguracije robota i kamere.



Slika 9. Konfiguracije robot-kamera [4]

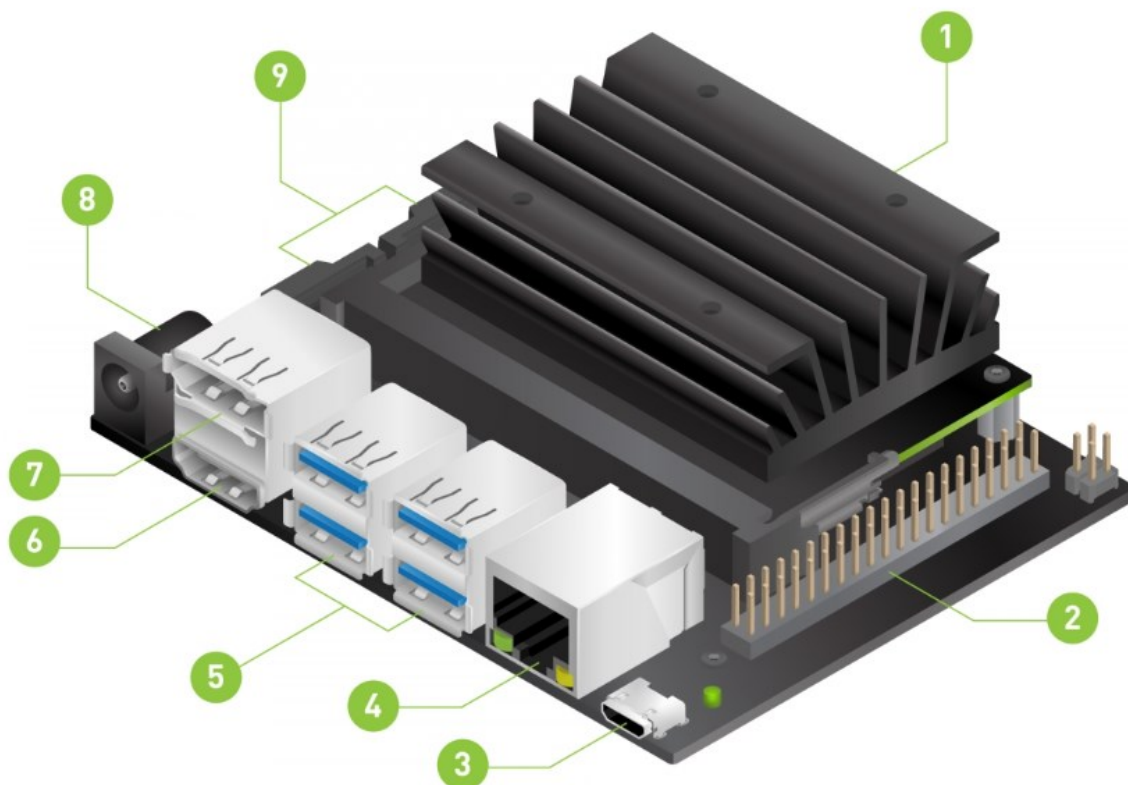
Vidljivo je da se kalibracija kamere razlikuje u ekstrinzičnoj kalibraciji. Razlikuju se prema homogenim transformacijama koje se traže kalibracijom. Tako se u slučaju oko-ruka konfiguracije traži homogena transformacija koordinatnog sustava kamere u odnosu na koordinatni sustav baze robota, dok se u slučaju oko-u-ruci traži homogena transformacija koordinatnog sustava kamere u odnosu na koordinatni sustav alata robota.

3. OPREMA ZA RAZVOJ VIZIJSKOG SUSTAVA

U ovom dijelu rada pokazat će se koja se oprema koristi kako bi se razvio vizijski sustav. Kako je navedeno u uvodu, vizijski sustav se sastoji od kamera i hardvera, kojim se procesira slika tj. informacija, i softvera koji sadrži funkcije i metode kojima vizijski sustav dolazi do rješenja. Oprema koja se koristi u ovom radu za vizijski sustav je Jetson Nano komplet za razvijanje kao hardver koji procesira informacije i na koji se instalira softver za razvoj vizijskog sustava. Za obradu slike koristi se OpenCV biblioteka. Na Jetson Nano-u postoje priključci preko kojih se može priključiti periferija za upravljanje i vizualizaciju softvera te priključak za ethernet kabel preko kojeg se Jetson spaja na mrežu.

3.1. Jetson Nano komplet za razvijanje

Jetson Nano komplet za razvijanje napravljen je od strane NVIDIA-e, a namijenjen je za razvoj i testiranje aplikacija koje koriste umjetnu inteligenciju i računalni vid. Na Slika 10 prikazan je Jetson Nano sa najvažnijim priključcima za periferiju i napajanje.



Slika 10. Jetson Nano sa priključcima [1]

Na slici su prikazani dijelovi Jetson Nano-a:

1. microSD utor za memoriju
2. 40-pinski priključak za proširenje
3. Micro-USB priključak za napajanje od 5V
4. Gigabitni ethernet priključak
5. 4 priključka USB 3.0
6. HDMI izlazni priključak
7. DisplayPort konektor
8. DC Barrel jack za ulaz napajanja
9. 2 MIPI CSI-2 konektora za kamere

Na Sliku 11 prikazane su karakteristike Jetson nano-a.

JETSON NANO	
AI Performance	472 GFLOPs
GPU	128-core NVIDIA Maxwell™ GPU
CPU	Quad-Core ARM® Cortex®-A57 MPCore
Memory	4GB 64-bit LPDDR4 Memory
Storage	Micro SD card slot
Video	Encode: 4K @ 30 (H.264/H.265) Decode: 4K @ 60 (H.264/H.265)
Connectivity	Gigabit Ethernet
CSI	12x CSI-2 D-PHY 1.1 (Up to 18 GB/s)
Display	Two Multi-Mode DP 1.2 eDP 1.4 HDMI 2.0 1x2 DSI (1.5Gbps/lane)
PCIe	Gen 2 1x4
Power	5W / 10W

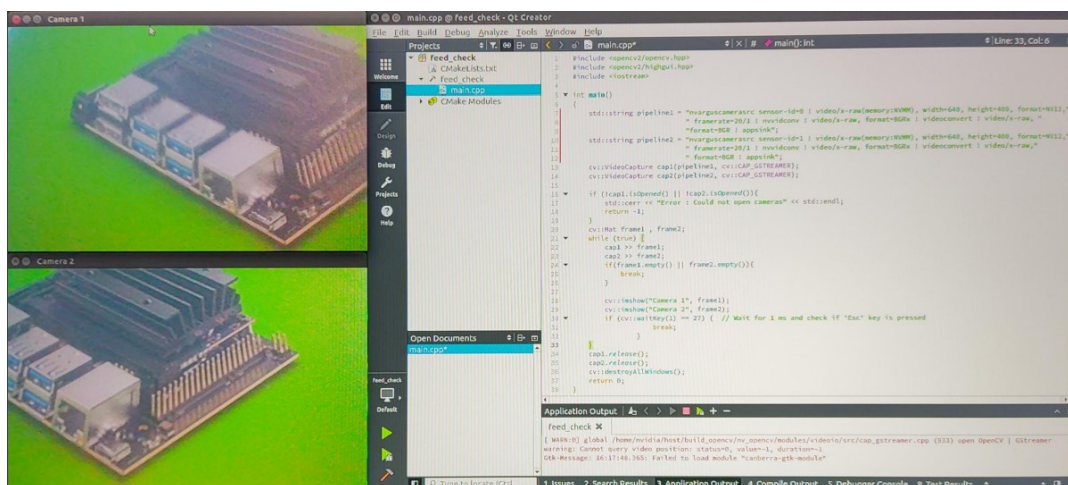
Slika 11. Karakteristike Jetson nano-a [1]

Kako bi se Jetson Nano osposobio potrebno je instalirati operativni sustav Ubuntu na mikro SD karticu. Na priključke se spaja periferija kao što su miš, tipkovnica i monitor za upravljanje sustavom. SD kartica sa instalacijom operativnog sustava umeće se u utor za SD karticu na Jetson-u. Nakon uspješne instalacije operativnog sustava potrebno se spojiti na internet i instalirati OpenCV biblioteku i softver za programiranje (Qt creator). Slika 12 prikazuje kako izgleda korisničko sučelje nakon instalacije.



Slika 12. Korisničko sučelje Jetson nano-a [1]

Nakon uspješne instalacije potrebno je priključiti kamere na CSI priključke i napraviti programski kod kojim će se provjeriti je li sve dobro instalirano i jesu li kamere učitane. Ukoliko je sve dobro instalirano i kamere su uspješno učitane, na ekranu monitora prikazuje se slika sa kamera. Na Slika 13 prikazane su slike sa kamera uz prikaz sučelja Qt creator-a.



Slika 13. Prikaz slika s kamera i sučelje Qt creator-a

3.2. Kamera IMX 219-120

Kamera koja se koristi za dohvaćanje slike jest IMX219-120 i prikazana je Slika 14.



Slika 14. IMX219-120 kamera [3]

Kamera za sučelje koristi CSI pa je kompatibilna sa Jetson nano-om, malih je dimenzija i širokog kuta vidnog polja što je dobro za vizijski sustav za kalibraciju alata robota. Kamera za senzor koristi IMX219 senzor sa rezolucijom 3280 x 2464 što znači da može detektirati sitne detalje što je potrebno za ovakvu aplikaciju. Slika 15 prikazuje sve specifikacije dane od proizvođača.

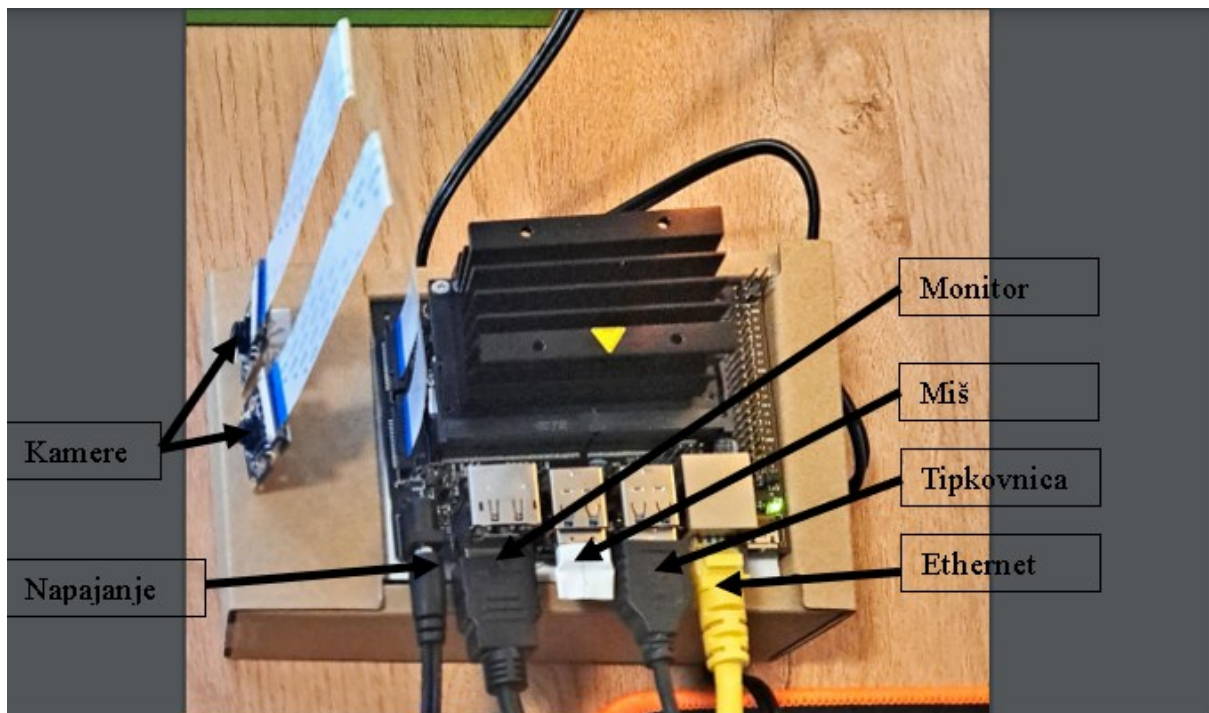
Parameters

- 8,000,000 pixels
- Photosensitive chip: IMX219
- Resolution: 3280 x 2464
- Camera parameters:
 - CMOS size: 1/4 inch
 - Aperture (F): 2.0
 - Focal Length: 1.79mm
 - Diagonal Field of View (FOV): 120° degrees
 - Distortion: <13.6%
 - Lens size: 6.5mm x 6.5mm
- 4 screw holes:
 - Available in a fixed position
 - Support 3.3V external power supply
- Dimensions: 25mm x 24mm

Slika 15. Karakteristike kamere IMX 219-120 [3]

3.3. Integracija vizijskog sustava

Vizijski sustav je u potpunosti definiran ranije opisanim komponentama. Sadrži kamere za dohvaćanje scene, hardver i softver koji su potrebni za obradu slike. Slika 16 prikazuje kako izgleda sustav kada je spojen na napajanje sa priključenim svim kabelima za periferiju koja je potrebna za rad istog.



Slika 16. Vizijski sustav sa svim dijelovima

Kada je sva oprema definirana, može se razviti kućište kako bi oprema bila zaštićena od vanjskih utjecaja, i kalibracijska kutija kako bi se kamere mogle fiksirati i odrađivati zadanu funkciju.

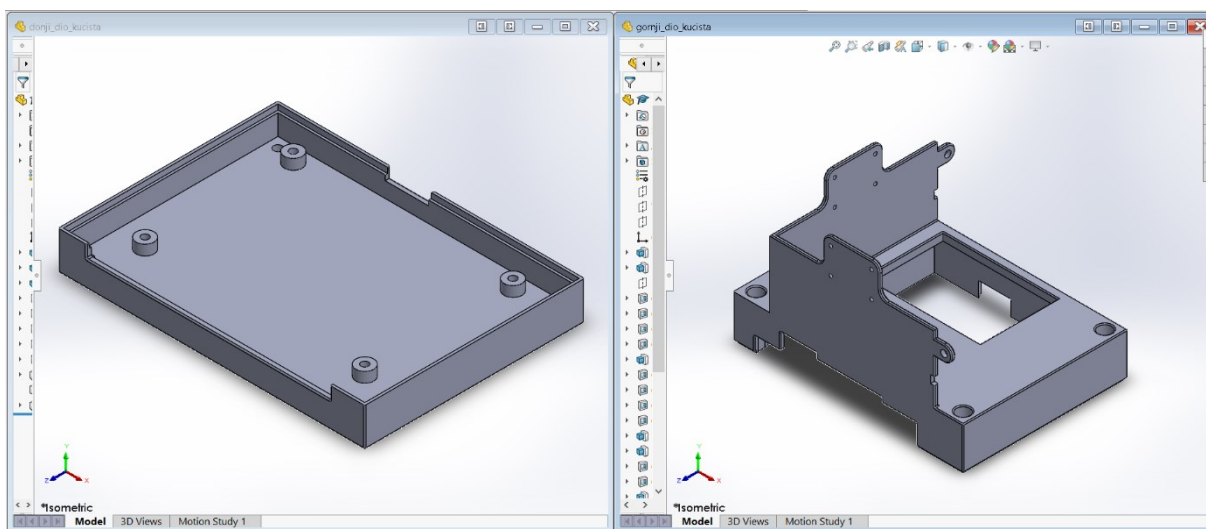
4. PROJEKTIRANJE KALIBRACIJSKOG UREĐAJA

4.1. Kućište

Razvoj kućišta treba zadovoljiti sljedeće zahtjeve:

- Treba biti omogućeno spajanje i odspajanje potrebne periferije i kamera kada je kućište u sastavljenom stanju
- Na kućištu treba postojati nosač za kamere gdje će se one pričvrstiti za kalibraciju istih
- Kućište mora biti moguće sastaviti i rastaviti bez oštećivanja opreme

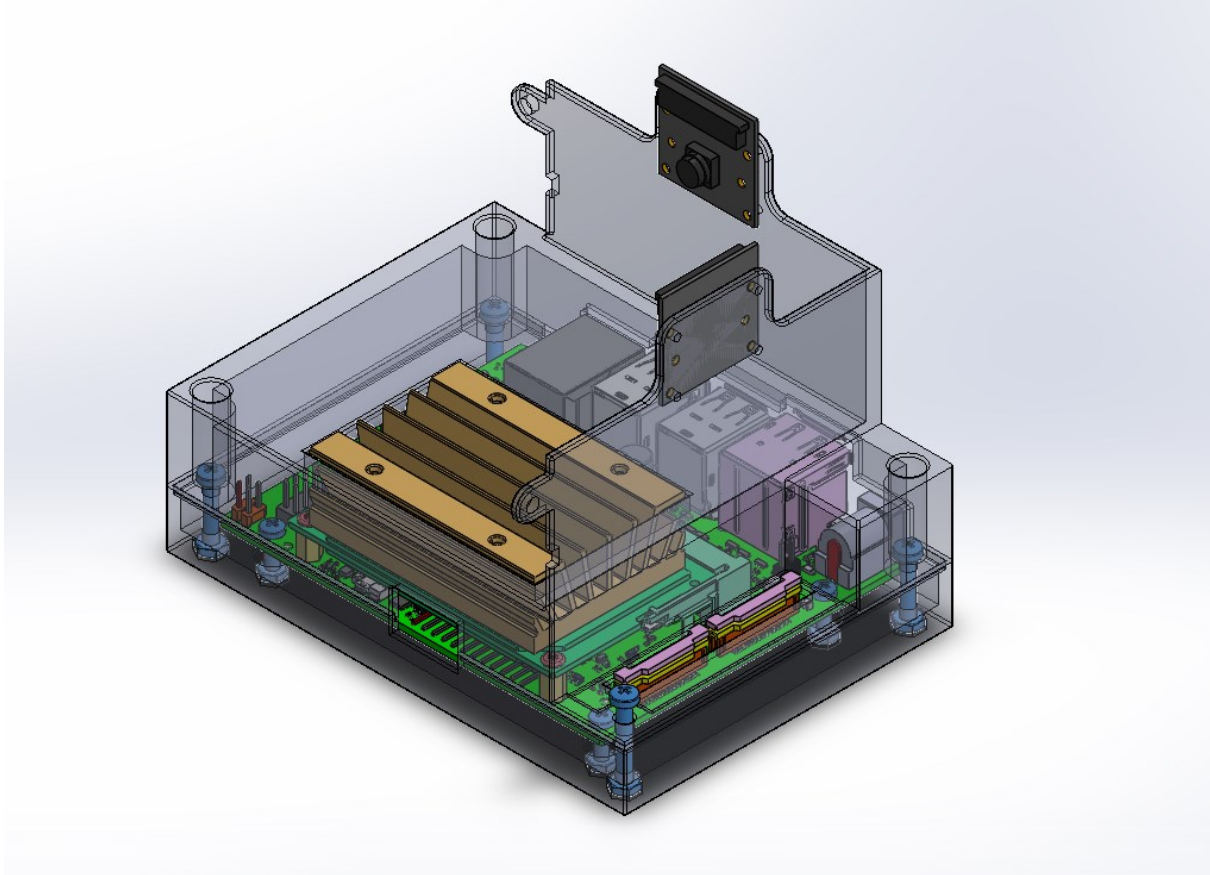
Prema navedenim zahtjevima kućište za vizijski sustav napravljeno je iz dva dijela, na donji dio se pričvršćuje Jetson nano sa četiri vijaka i matice pa se za isti pričvršćuje i gornji dio kućišta sa drugim setom četiriju vijaka i matica. Na gornjem dijelu kućišta napravljen je nosač na koji se učvršćuju kamere prilikom njihove kalibracije. Na Slika 17 prikazani su donji i gornji dio kućišta, modelirani u Solidworks CAD programu.



Slika 17. Donji dio (lijevo) i gornji dio (desno) kućišta

Kao što je vidljivo sa slike kućište je kompliciranije izvedbe pa se za tehnologiju izrade koristi 3D print.

Slika 18 prikazuje način spajanja donjeg i gornjeg kućišta.



Slika 18. Prikaz načina spajanja kućišta

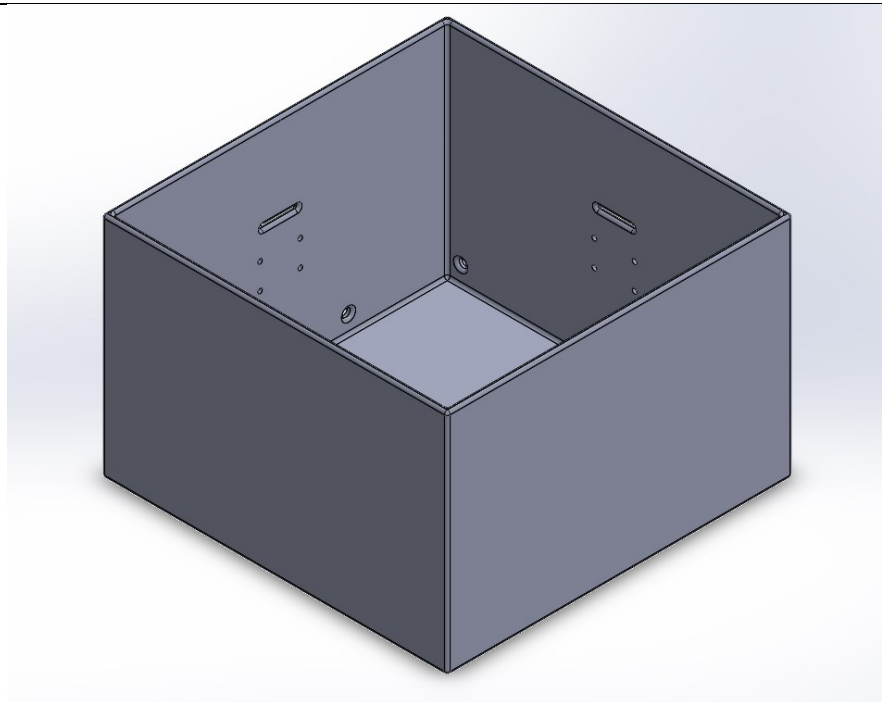
Spajanje je napravljeno putem vijaka i matica. Na donjem dijelu kućišta napravljeni su utori za matice kako bi se spriječilo odvrtnje, a na gornjem dijelu kućišta je napravljena rupa za vijke kojima se spajaju gornji i donji dio kućišta.

4.2. Kalibracijska kutija

Kako bi se olakšala segmentacija alata robota napravljena je kalibracijska kutija koja omogućava da pozadina slike bude uniformna. Izvedena je na način da se učvrsti vijcima i maticama za kućište i pritom su zadovoljeni sljedeći zahtjevi na dizajn kutije:

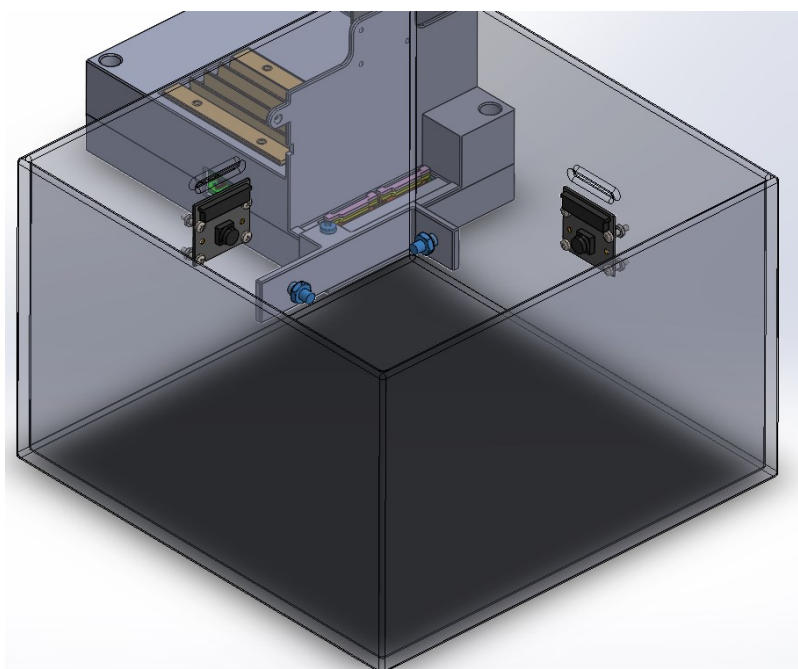
- Kutija je dovoljno velikih dimenzija da je alat u fokusu kamera koje snimaju scenu
- Kamere se postavljaju pod 90° stupnjeva kako bi se osiguralo da se osi kamera sijeku
- Kutija je dovoljno odvojena od kućišta kako bi se spriječilo oštećenje opreme
- Kutija je takve boje da osigurava dobar kontrast pozadine i alata robota
- Kutija je takvog dizajna da je moguće spojiti kamere na CSI portove

Slika 19 prikazuje dizajn kalibracijske kutije.



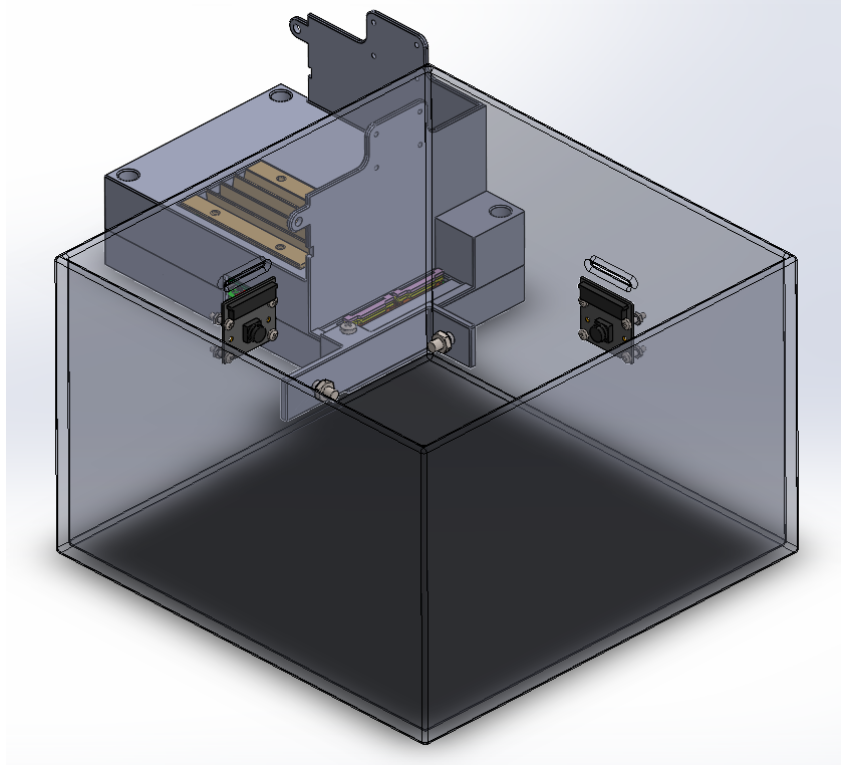
Slika 19. Dizajn kalibracijske kutije

Na gornjoj slici vidljivo je da su kamere postavljene pod kutom od 90° stupnjeva, te postoji utor za CSI kabel kojim se one spajaju na Jetson nano. Dodatno su napravljena dva provrta kojima se putem dodatnog dijela omogućava spajanje kutije i kućišta. Način spajanja kalibracijske kutije s kućištem prikazan je na Slika 20.

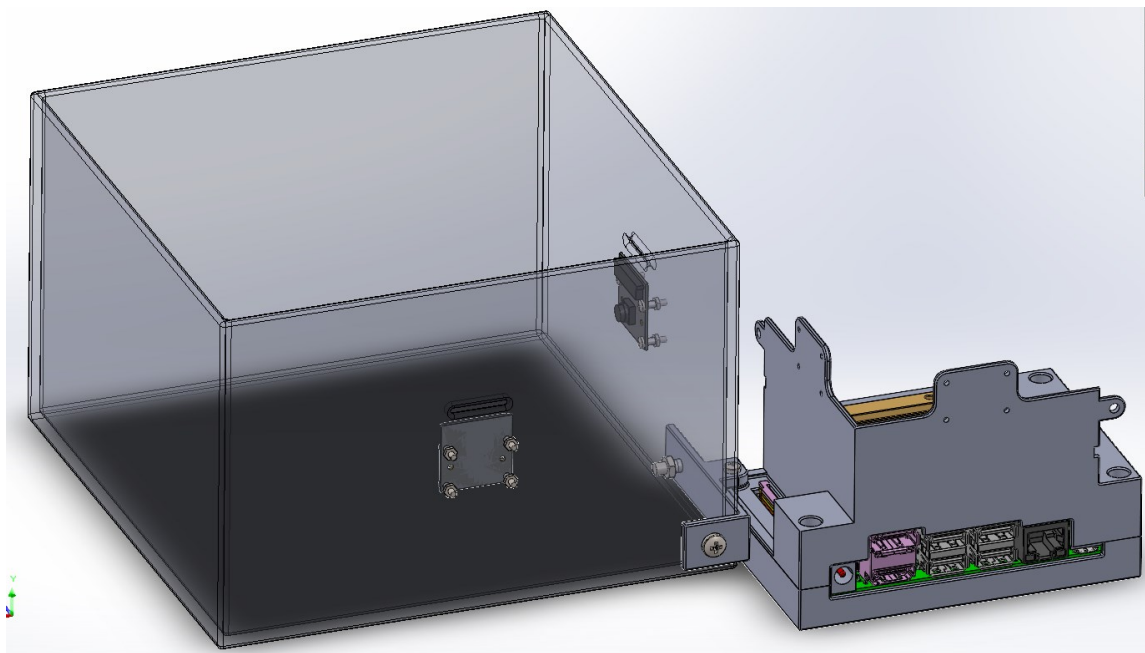


Slika 20. Način spajanja kalibracijske kutije s kućištem

Ovime je završen fizički razvoj kućišta Jetson-a i kalibracijske kutije. Krajnji izgled cijelog sklopa vizijskog sustava s kalibracijskom kutijom prikazan je na Slika 21 i Slika 22.

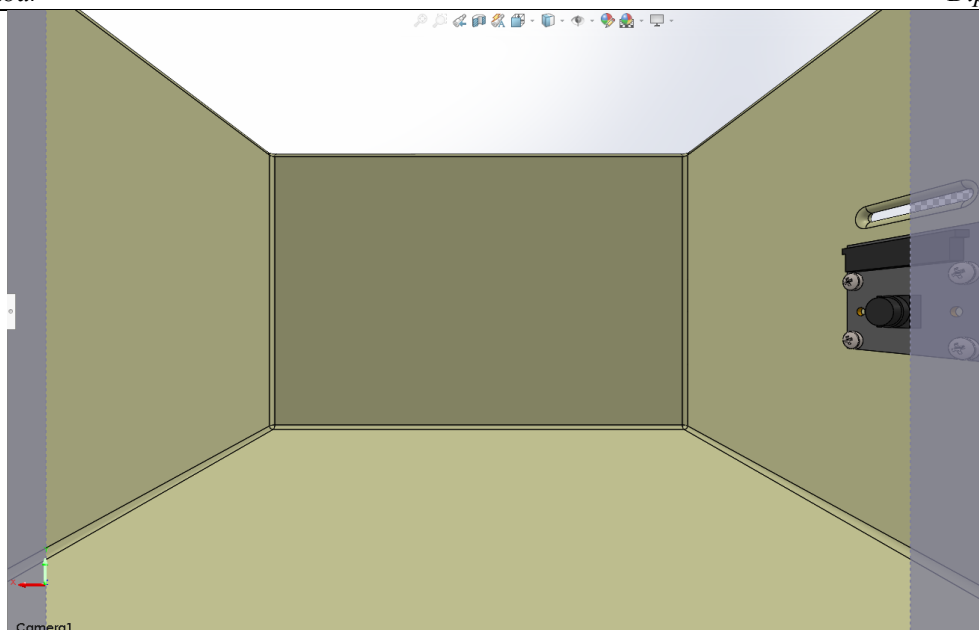


Slika 21. Cijeli sklop vizijskog sustava i kalibracijske kutije

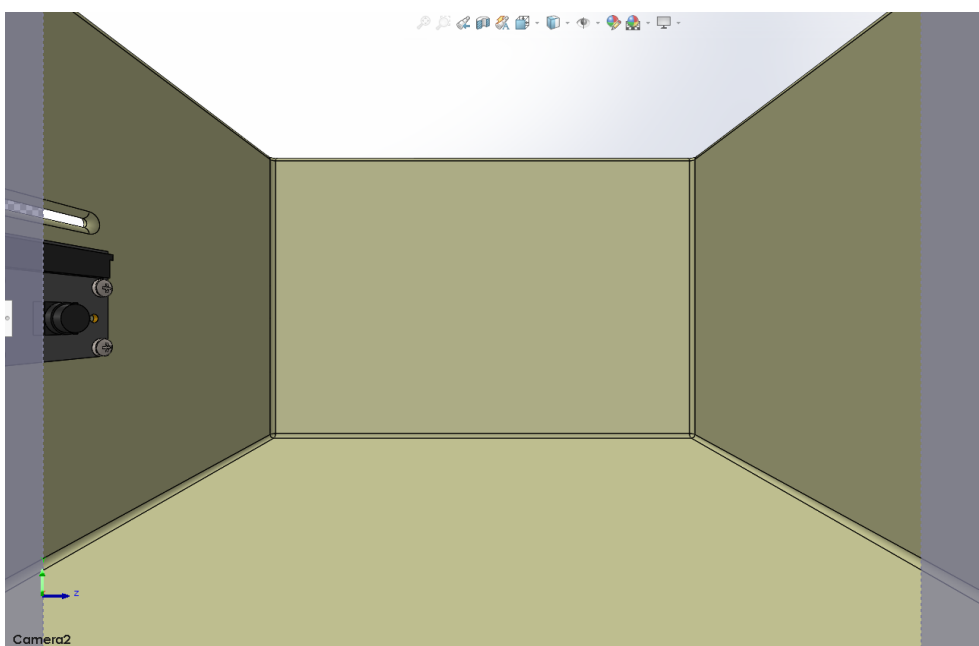


Slika 22. Cijeli sklop vizijskog sustava i kalibracijske kutije

Solidworks omogućuje dodavanje parametara kamera i na taj način može se simulirati dohvaćanje scene sa kamera. Slika 23 i Slika 24 prikazuju simulirane slike scene dohvaćene kamerama.



Slika 23. Simulirano dohvaćanje scene prve kamere



Slika 24. Simulirano dohvaćanje scene druge kamere

Kako bi se kućište moglo izraditi potrebno je definirati izgled pozadine kalibracijske kutije. To je ostvareno eksperimentom na modelu kalibracijske kutije, a cilj je definirati boje pozadine za koje postoji dobar kontrast alata i pozadine tj. dobra segmentacija slike.

5. EKSPERIMENT ZA ODABIR BOJE POZADINE

Napravljeni model umanjene kalibracijske kutije i postav prikazan je na Slika 25.



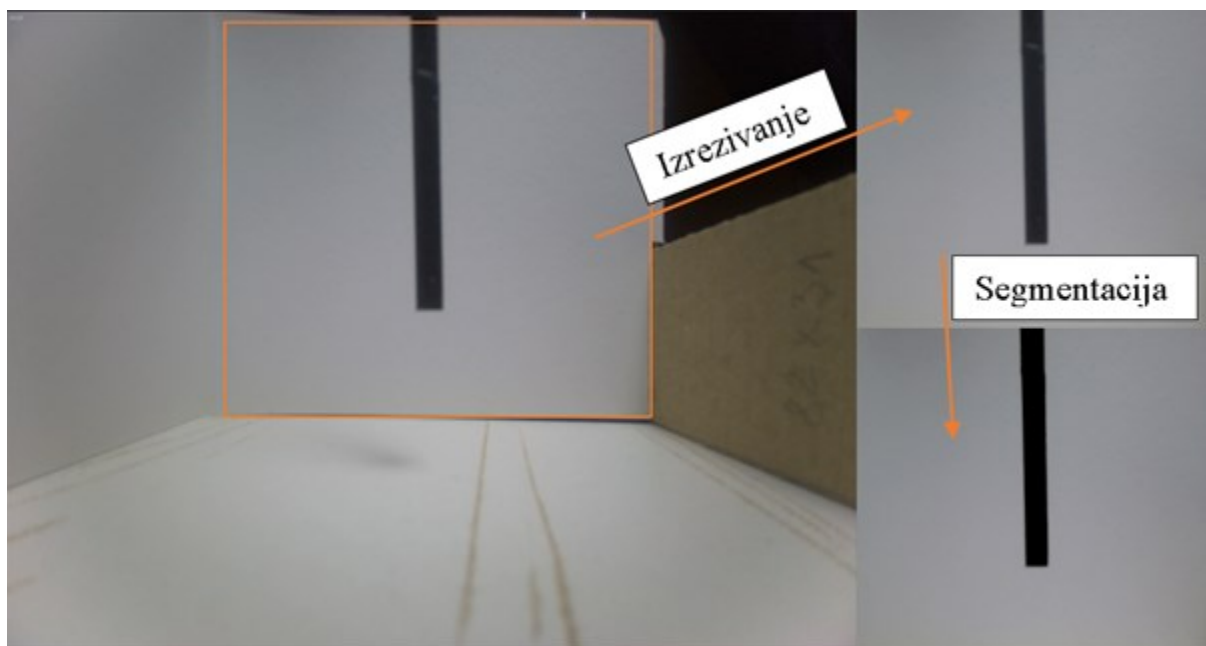
Slika 25. Postav za eksperiment

Predmeti segmentacije u eksperimentu prikazani su na Slika 26. Prvi je uniformne boje bez velike mogućnosti reflektiranja svjetlosti, dok je drugi nesavršeni i reflektira svjetlost kao i mnogi alati napravljeni od metala.



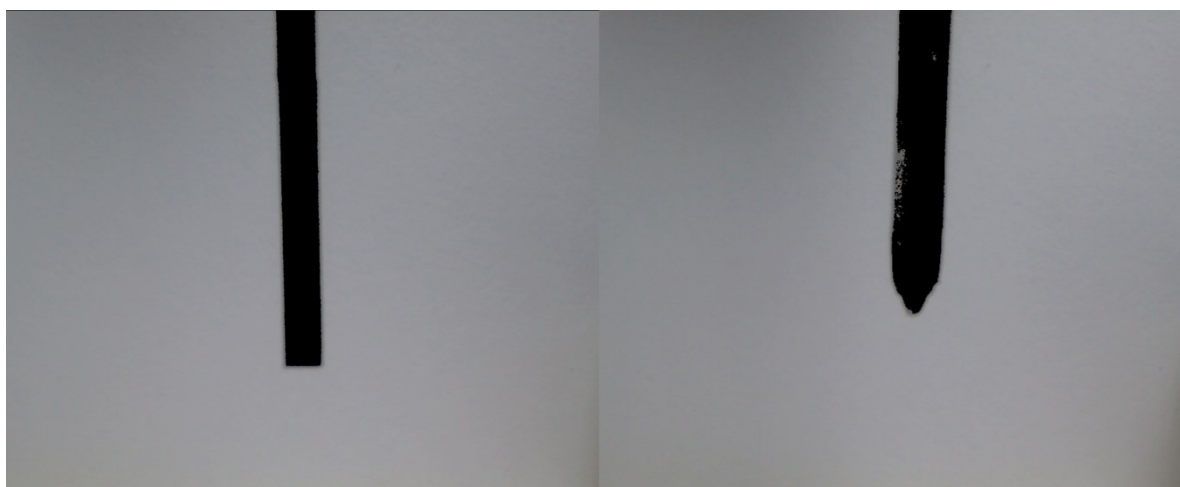
Slika 26. Predmeti segmentacije

U eksperimentu je odabrano pet različitih boja pozadine kalibracijske kutije. Segmentacija se izvodi na način da se prvo učitava slika sa kamere, a potom se ona izrezuje na dio slike koji je od interesa (pozadina kutije i predmet). Potom se slika obrađuje na način da se zagladi filterima i prebaci u HSV format pošto se segmentacija radi na temelju boje pozadine, a ne alata. Provodi se traženje piksela koji odgovaraju boji pozadine te se tvori maska kojom se slika segmentira morfološkim operacijama i dobiva se rješenje segmentacije. Postupak je prikazan Slika 27.



Slika 27. Postupak segmentacije

Za rješenja segmentacije dani su rezultati segmentacije oba predmeta po bojama i iznosi pragova boja u HSV formatu. Rezultati segmentacije oba alata za bijelu boju pozadine sa donjim pragom (0,0,100) i gornjim pragom (180,40,255) dani su na Slika 28.



Slika 28. Rezultati segmentacije bijele boje

Rezultati segmentacije za crvenu boju pozadine sa prvim donjim pragom (0,100,100) i prvim gornjim pragom (10,255,255), a drugim donjim pragom (170,100,100) i drugim gornjim pragom (180,255,255) dani su na Slika 29.



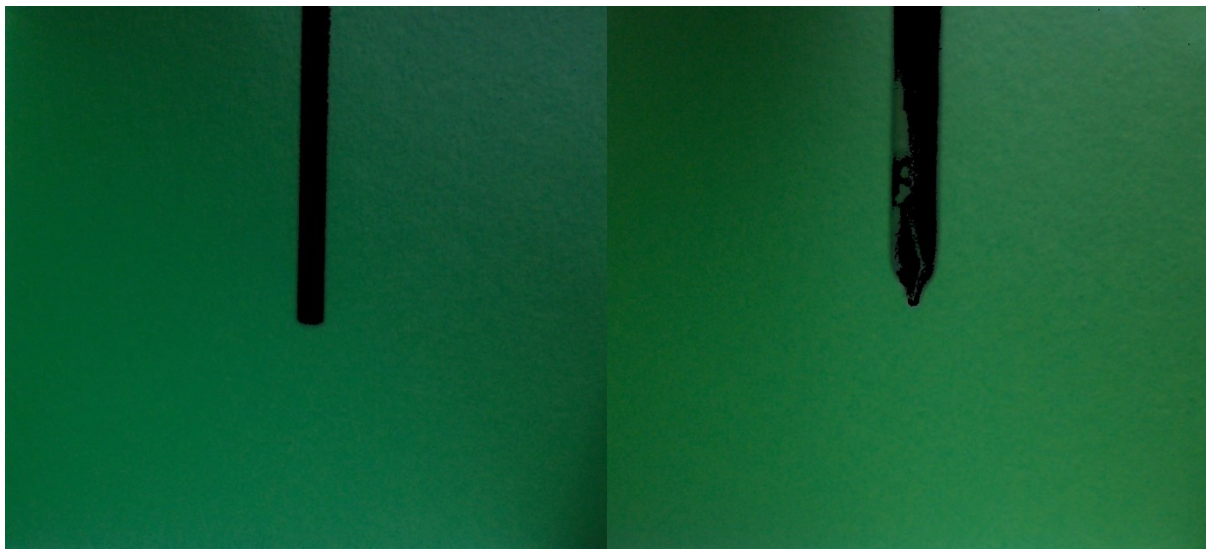
Slika 29. Rezultati segmentacije crvene boje

Rezultati segmentacije za plavu boju pozadine sa donjim pragom (100,100,80) i gornjim pragom (120,255,255) dani su na Slika 30.



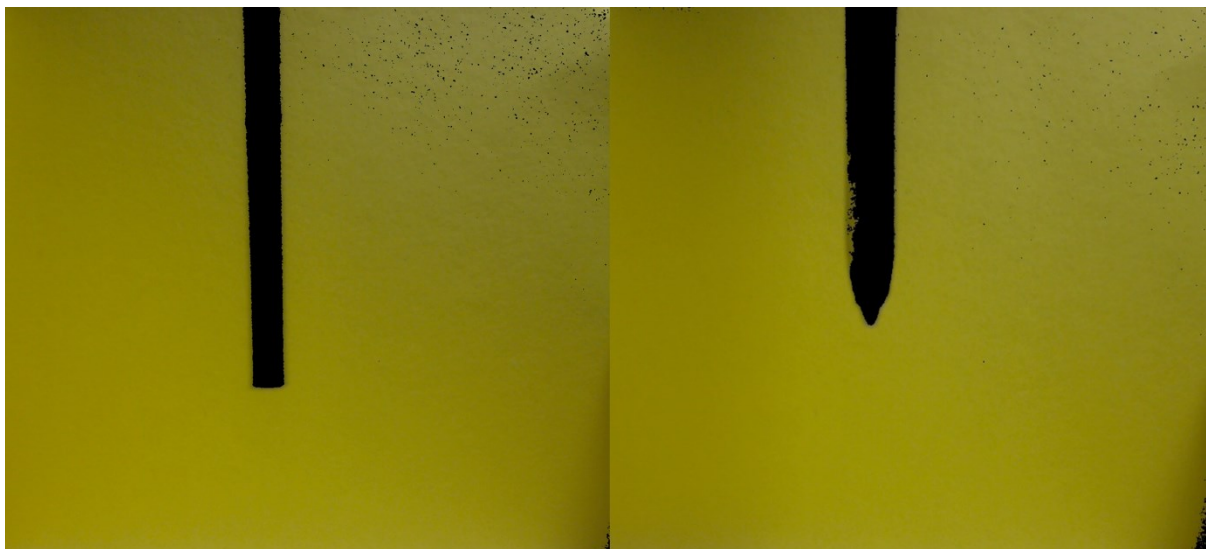
Slika 30. Rezultati segmentacije plave boje

Rezultati segmentacije za zelenu boju pozadine sa donjim pragom (35,100,50) i gornjim pragom (85,255,255) dani su na Slika 31.



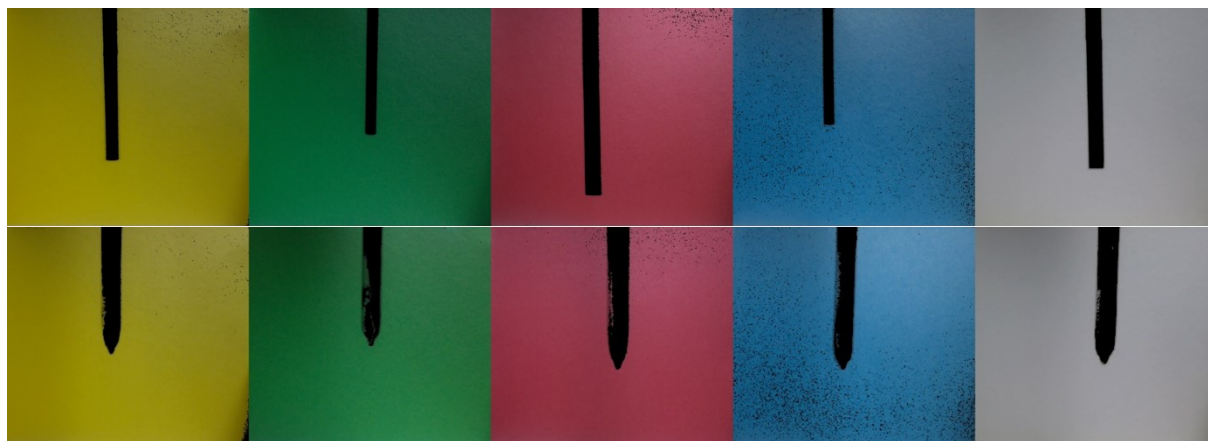
Slika 31. Rezultati segmentacije zelene boje

Rezultati segmentacije za žutu boju pozadine sa donjim pragom (20,100,100) i gornjim pragom (30,255,255) dani su na Slika 32.



Slika 32. Rezultati segmentacije žute boje

Radi bolje usporedbe rezultata dana je Slika 33 koja prikazuje sve rezultate segmentacije.

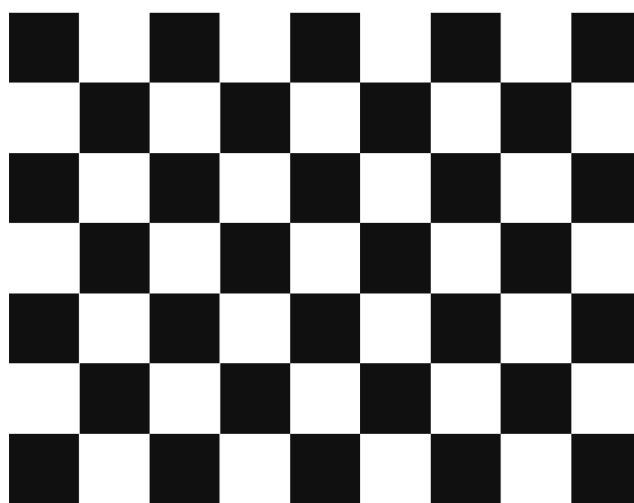


Slika 33. Svi rezultati segmentacije

Kao što je vidljivo sa Slika 33 crvena, plava i žuta boja osjetljive su na svjetlost što rezultira pojavom sol i papar šuma na slici, dok se kod zelene i bijele boje taj šum ne pojavljuje. Kod zelene i bijele boje vidi se da imaju mali problem sa reflektirajućim predmetom što se očituje kao pojava tih boja na predmetu zbog refleksije svjetlosti s njih. Bijela boja sama po sebi poprilično je osjetljiva na pojavu sjena tako da je odabrana zelena pozadina kalibracijske kutije zbog dobrog kontrasta između predmeta i pozadine. Time je završen eksperiment i odabrana je zelena boja pozadine kalibracijske kutije.

6. KALIBRACIJA KAMERA

S obzirom da je cilj koristiti vizijski sustav za postavljanje alata robota u istu točku, potrebno je definirati referentnu točku. Ta točka se nalazi na sjecištu optičkih osi kamera. Kako bi se moglo odrediti koliko je alat udaljen od referentne točke potrebno je napraviti kalibraciju kamera da bi se moglo iste koristiti za mjerenje udaljenosti. Kalibracija kamera je proces kojim se određuju intrinzični parametri kamere kao što su fokalna duljina i koeficijenti distorzije. U ovom slučaju za kalibraciju se koristi šahovska ploča prikazana Slika 34.

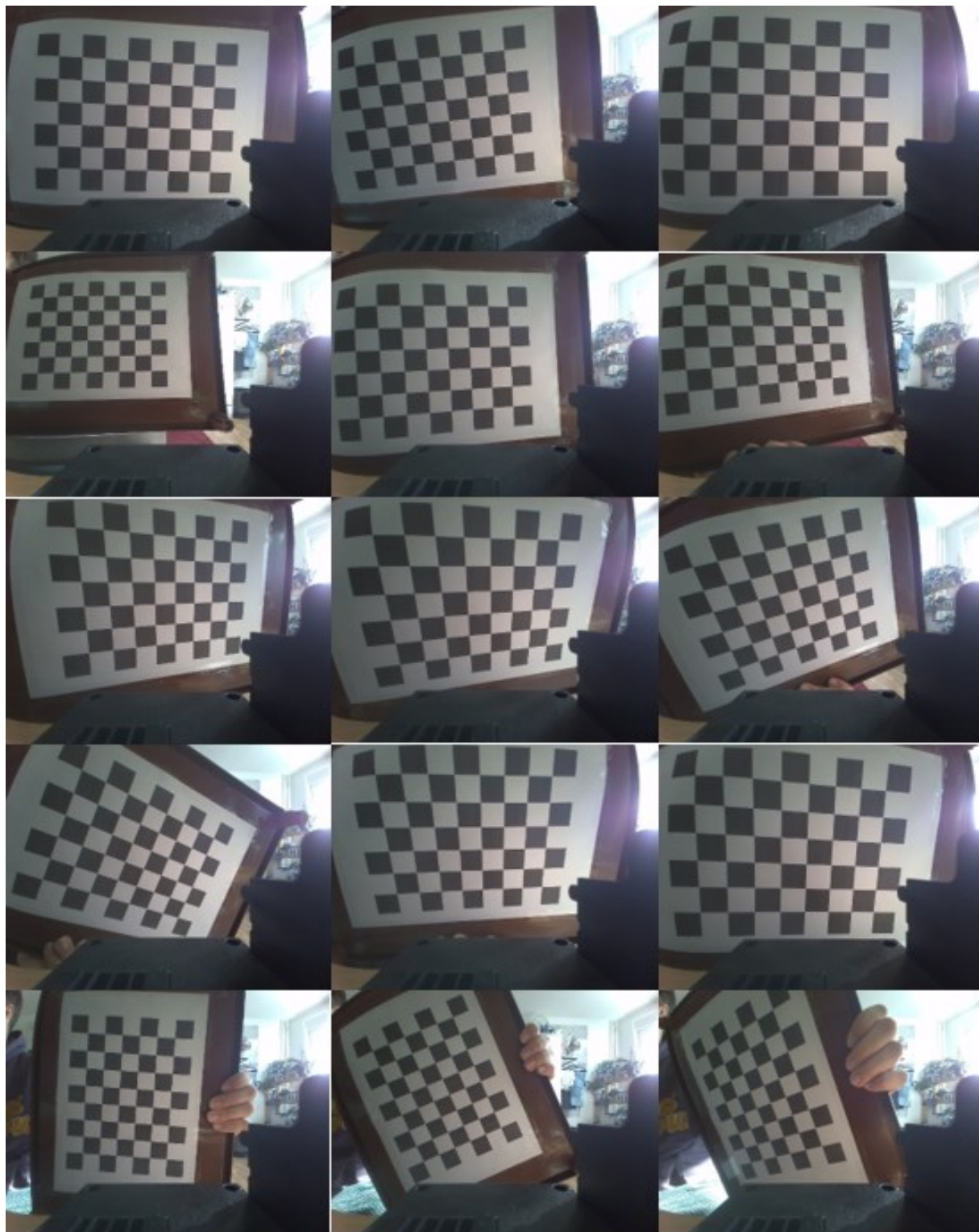


Slika 34. Kalibracijska ploča sa 8 x 6 kvadrata sa stranicama od 25mm

Kako bi kalibracija bila zadovoljavajuća, potrebno je snimiti mnogo slika kalibracijske ploče sa raznim pozicijama i orijentacijama prema kameri koja se kalibrira. Dodatno, za dobru kalibraciju preporučeno je snimiti od 10 do 20 slika gdje kalibracijski uzorak čini minimalno 50% slike. Kalibracija zahtijeva i definiciju kalibracijskog uzorka programski pa je napravljena skripta koja se koristi za kalibraciju. Za dobivanje intrinzičnih parametara kamera korištena je Open-CV-ova funkcija `calibrateCamera()`. Dio procesa kalibracije u kojem se snimaju slike sa kamera odvojena je programska skripta. Ukupan proces se odvija na način da se prvo snimaju slike i zatim se spremne, a spremljene slike se učitaju u skriptu za kalibraciju kamera. U ovom radu snimljeno je 16 slika za kalibraciju jedne kamere. Na svakoj od slika traži se odgovarajući vrh kvadrata na šahovskoj ploči i uvijek istim redoslijedom svi ostali u nizu bez obzira na

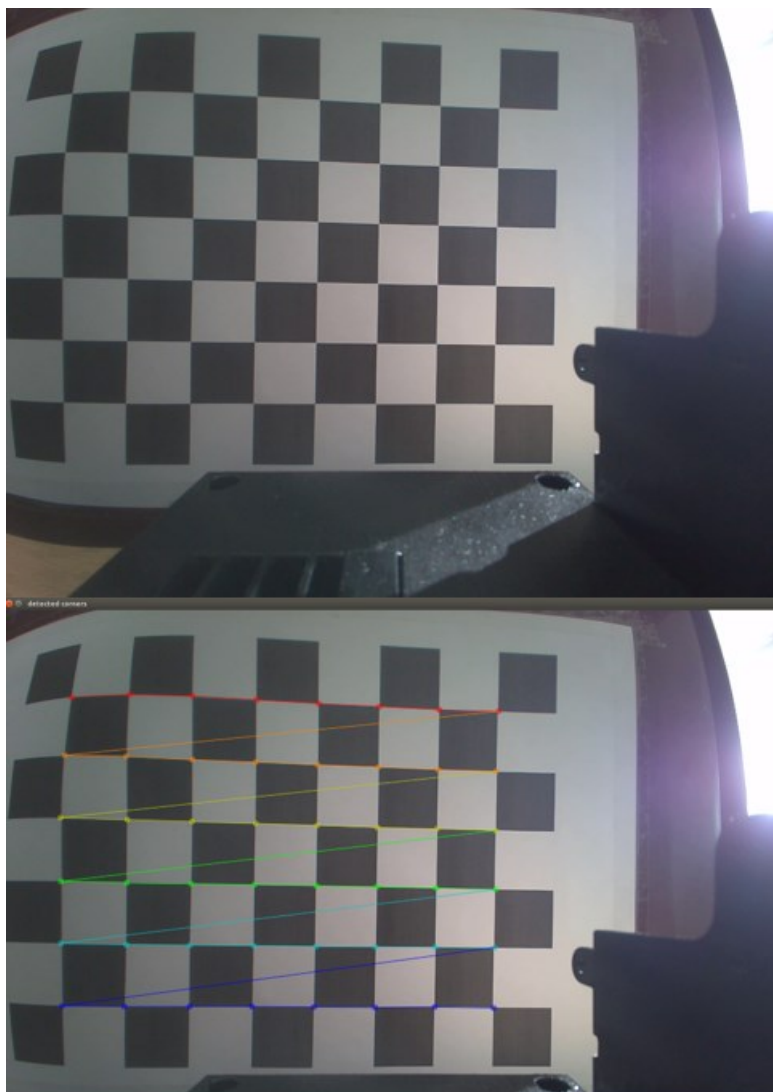
orijentaciju šahovske ploče. Na temelju detektiranih vrhova radi se kalibracija kojom se definiraju intrinzični parametri kamere i zapisuju se u tekstualnu datoteku.

Kao primjer, prikazane su kalibracijske slike za jednu od kamera koje se kalibriraju. Slika 35 prikazuje prvih 15 kalibracijskih slika za jednu od kamera.



Slika 35. Kalibracijske slike za jednu od kamera

Za primjer detekcije vrhova kvadrata uzeta je zadnja kalibracijska slika iz seta od 16 slika. Na Slika 36 prikazana je detekcija rubova kvadrata jedne kalibracijske slike.



Slika 36. Prikaz detekcije rubova na zadnjoj kalibracijskoj slici

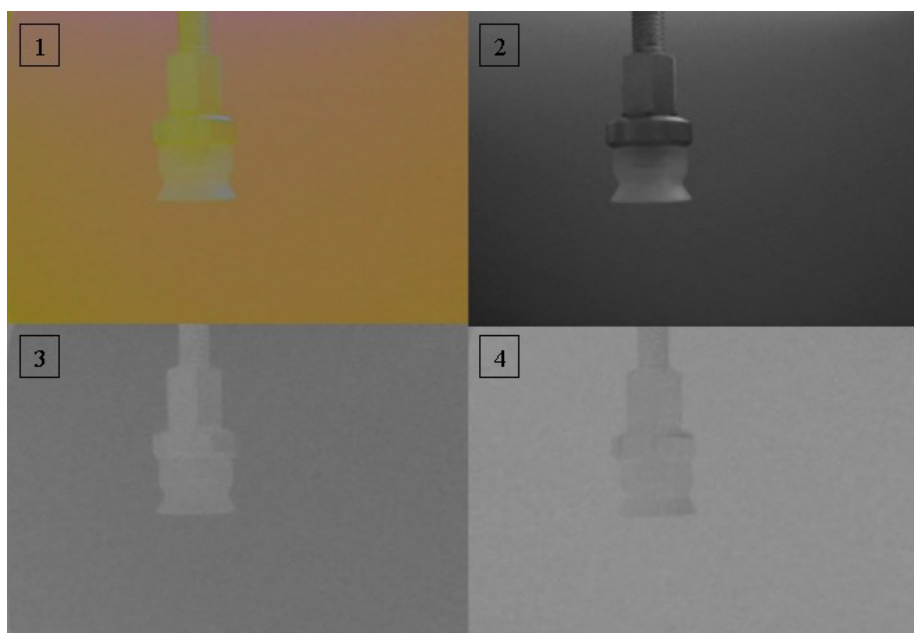
Nakon što se napravi kalibracija kamere u tekstualnu datoteku se spremne podaci o matrici kamere i koeficijentima distorzije. Za ovakav set kalibracijskih slika za jednu od kamera dobivena je tekstualna datoteka u kojoj su sadržani intrinzični parametri, njen sadržaj prikazan je Slika 37.

```
matrica kamere:  
[814.3097531833109, 0, 810.6401441916403;  
0, 813.3822440603992, 626.6564417158986;  
0, 0, 1]  
  
koeficijenti distorzije:  
[0.0110281691408063, 0.06642369117113196, 0.001098443458176403, -0.0006956327194863606, -0.09539373780159867]]
```

Slika 37. Intrinzični parametri jedne od kamera

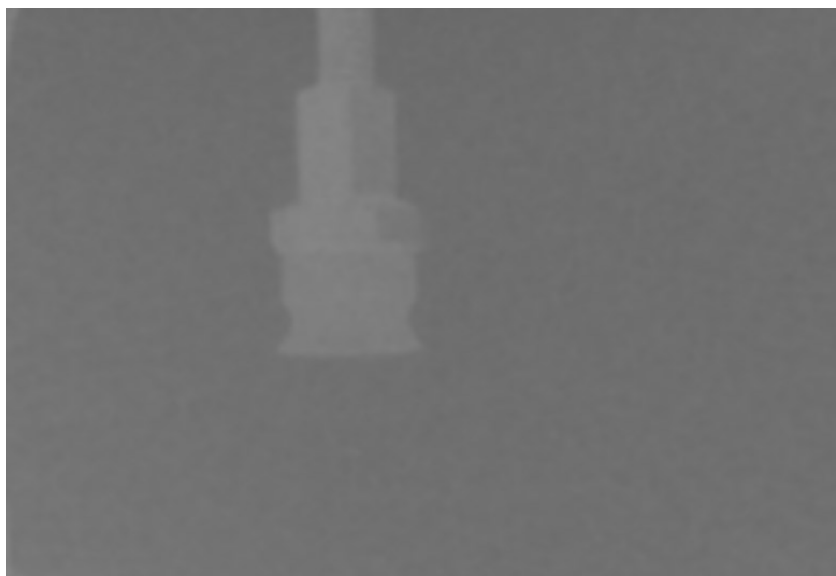
7. RAZVOJ ALGORITMA ZA DETEKCIJU VRHA ALATA ROBOTA

Da bi vizijski sustav bio u mogućnosti pomoći u kalibraciji alata robota, potrebno je napraviti njegovu detekciju. Zadatak vizijskog sustava jest detekcija vrha alata koji se želi kalibrirati i izračun greške trenutne pozicije alata u odnosu na referentnu točku. Kao što se vidjelo kod korištenja metalnog predmeta dolazi do tzv. lažnog obojenja na površini predmeta, što otežava segmentaciju alata robota. Zbog toga postoji potreba za segmentacijom koja može segmentirati alat i kada se na njegovoj slici, zbog refleksije, prikazuju pikseli koji su slične boje kao pozadina. Zbog ovog problema segmentacija je napravljena koristeći Lab format prostora boja umjesto HSV-a. Lab prostor boja sastoji se od 3 kanala. Prvi je kanal svjetlosti gdje su jako osvijetljeni pikseli prikazani jakim intenzitetom, koji odgovara bijeloj boji, a oni slabije osvijetljeni prigodnim manjim intenzitetom (siva i crna boja). Drugi kanal prikazuje intenzitet zelene i crvene boje gdje je zelena prikazana sa manjim intenzitetom, a crvena većim. Treći kanal prikazuje intenzitete plave i žute boje gdje je plava prikazana manjim intenzitetom. Obrada slike ima isti slijed kako je ranije opisano u poglavlju 5. Prvo se sa kamera dohvaća scena u kojoj se nalazi alat, a potom se slika izrezuje na način da se dobije isječak sa uniformnom pozadinom. Ta slika se nadalje prebacuje u Lab format prostora boja. Slika se rastavlja na kanale koji ju tvore, a time je napravljena priprema slike za daljnju obradu. Slika 38 prikazuje izrezanu sliku alata robota u Lab formatu i u svakom od kanala.



Slika 38. Alat robota u Lab formatu (1), kanal L(2), kanal a(3), kanal b(4)

Od interesa je kanal „a“ na broju tri Slika 38 jer on sadržava intenzitet zelene boje. Pošto je alat napravljen od metala, dijelovi alata koji su lažno obojani zelenom bojom postižu intenzitet slabiji od onog pozadine čime je osigurana dovoljno dobra segmentacija pozadine. Dodatno, ovakva slika se dalje koristi u obradi, ali se prije segmentacije ona filtrira bilateralnim filterom kako bi se naglasili rubovi alata i zamutila slika. Slika 39 prikazuje sliku kanala a nakon filtriranja bilateralnim filterom.



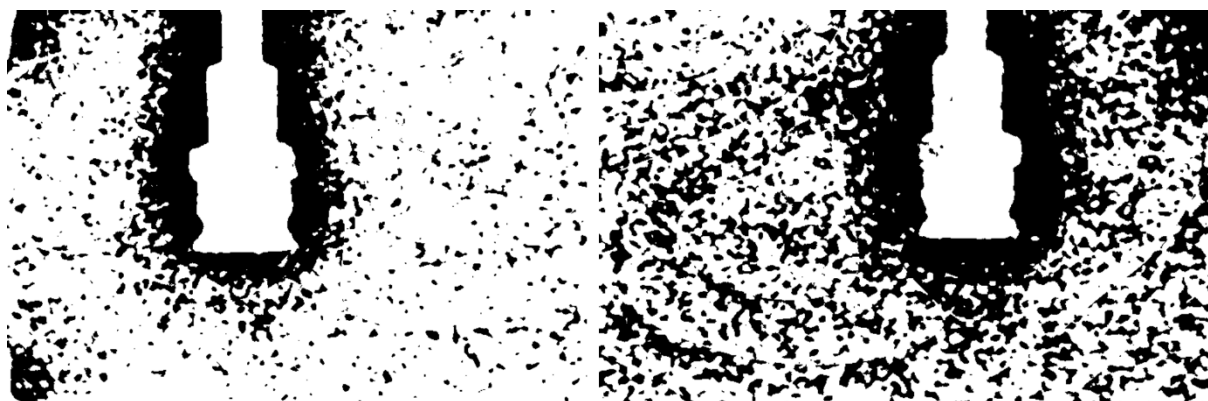
Slika 39. Filtrirana slika korištenjem bilateralnog filtera

Kada je slika filtrirana može se odraditi segmentacija na temelju pozadine. Korištene su dvije metode za segmentaciju alata sa korištenjem praga za razlikovanje piksela . Prva metoda je takozvana OTSU metoda. Njome se na slici traže dva najzastupljenija intenziteta piksela. Svaki od piksela na slici pridružuje se onom intenzitetu kojem je bliže. Potpuno odvajanje pragova dobiveno je na način da se svi pikseli manjeg intenziteta postavljaju na intenzitet 0 i prikazani su kao crna boja, dok su drugi intenziteta 255 i predstavljaju bijelu boju na slici. Na Slika 40 prikazana je segmentacija alata sa obje kamere korištenjem OTSU metode.



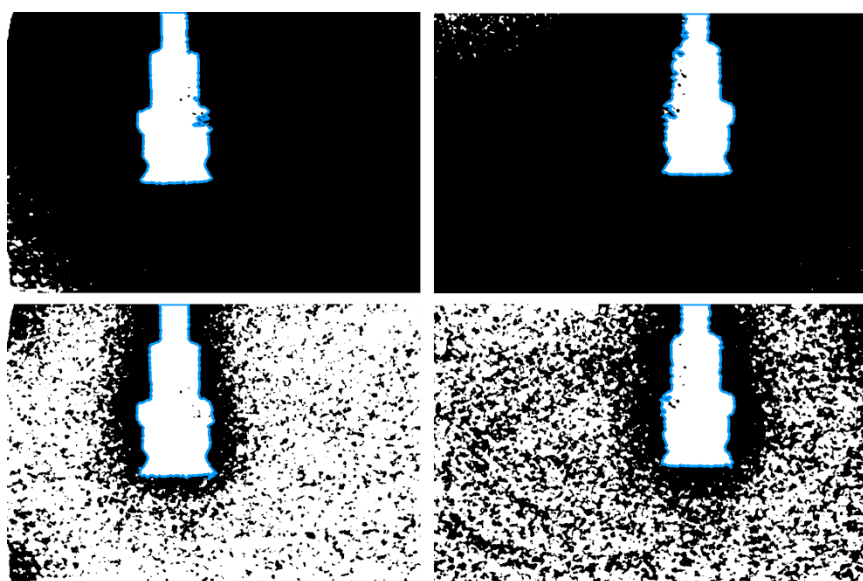
Slika 40. Segmentacija alata korištenjem OTSU metode

Druga metoda koja se koristi zove se adaptivni prag. Kod ove metode slika se dijeli na manje dijelove i u svakom se dijelu radi pridruživanje intenziteta na temelju određenog praga. Postupak se provodi na cijeloj slici, a pikseli su na kraju opet podijeljeni na one većeg intenziteta i one manjeg koristeći bijelu (255) i crnu (0) boju. Slika 41 prikazuje segmentaciju alata korištenjem adaptivnog praga.



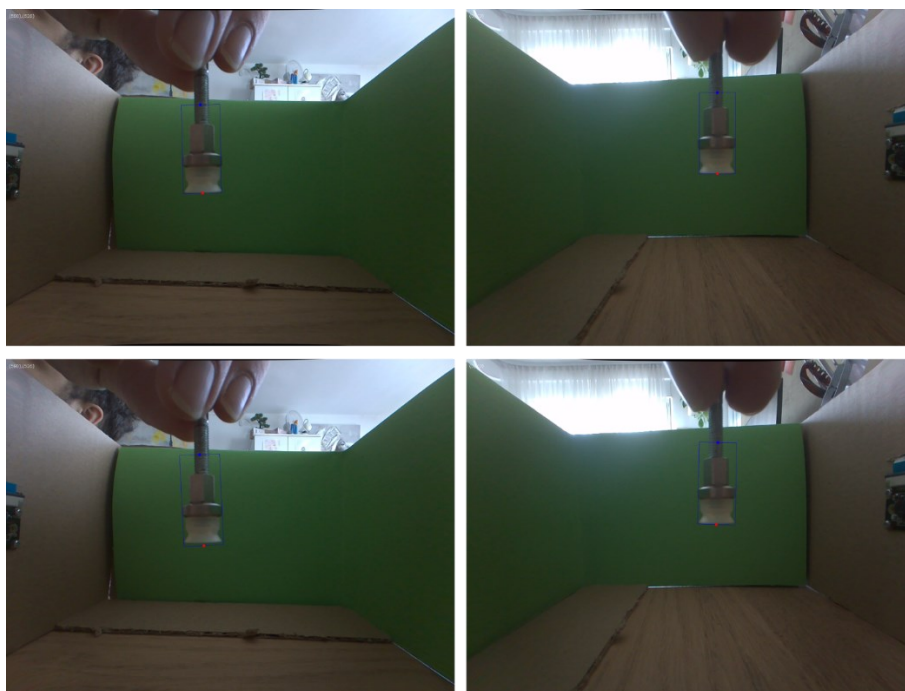
Slika 41. Segmentacija alata korištenjem adaptivnog praga

Kao što je vidljivo iz prethodnih slika, nijedna od ovih metoda nije potpuno uspješna u segmentaciji alata. OTSU metodom gubi se dio alata koji je lažno obojan zelenom bojom, dok adaptivnom metodom na slici nije potpuno segmentiran isključivo alat, ali je dobro segmentiran u svojem području. Budući da je za detekciju vrha alata potrebna kontura alata, njezina definicija ovisi o metodi segmentacije. Tako se za OTSU metodu traži najveća kontura na slici, dok se kod adaptivne metode traži druga najveća kontura na slici. Na Slika 42 prikazane su nađene konture alata u oba slučaja segmentacije.



Slika 42. Pronađene konture alata OTSU(gornje) i adaptivni prag(donje)

Na temelju kontura koje su pronađene traži se pravokutnik najmanje površine koji tu konturu može opisati. U radu se promatraju samo simetrični alati, uz pretpostavku da je njihova os simetrije dulja od širine alata. Imajući ovu pretpostavku u vidu, zaključuje se da se vrh alata nalazi na polovištu jedne od kraćih stranica pravokutnika koji opisuje alat. Rezultat detekcije vrha alata dobiven je na način da se na manjim stranicama pravokutnika odredila pozicija polovišta tih stranica, a zbog orijentacije kamera, vrh alata uvijek se nalazi na nižoj točki od dvije definirane. Slika 43 prikazuje detektirane točke pravokutnika u originalnoj slici snimljenoj s kamera za oba slučaja segmentacije.



Slika 43. Detekcija vrha alata OTSU(gornje) i adaptivni prag(donje)

Na slikama je vidljivo da obje metode daju zadovoljavajuće rezultate detekcije vrha alata robota. Iz segmentacije je vidljivo da OTSU metoda može izgubiti dio alata koji je lažno obojan i u slučaju gdje je lažno obojenje veliko može prouzročiti velike greške u detekciji vrha alata. Kod takvih slučaja potrebno je segmentaciju napraviti metodom adaptivnog praga kojom će se ti dijelovi alata bolje prepoznati i dati bolje rezultate. Sljedeći korak jest definirati referentnu točku i udaljenosti vrha alata od nje. Kako je prije rečeno, referentna točka nalazi se na sjecištu osi kamera, a kako su one jedna u odnosu na drugu postavljene pod kutom od 90° ta točka se svakoj od kamera nalazi u centru slike koja se snima. S obzirom da se sjecište optičkih osi kamera nalazi na poznatoj udaljenosti od svake kamere i iznosi 82 mm, može se odrediti udaljenost vrha alata od optičkih osi kamera. Način mjerenja udaljenosti opisan je jednadžbom (1).

$$x_{mm} = \frac{x_p \cdot Z}{f} \quad (1)$$

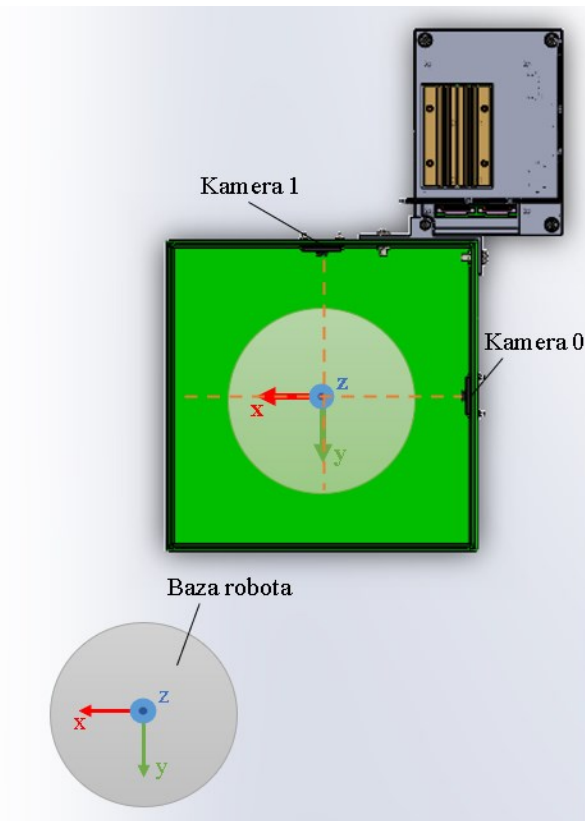
Kako je vidljivo, da bi se kamerom izmjerila udaljenost potrebno je udaljenost dviju točaka izraženu u pikselima pomnožiti s udaljenosti objekta od kamere što je u ovom slučaju sjecište optičkih osi. Dobiveni umnožak se dijeli sa pripadajućom fokalnom duljinom iz matrice intrinzičnih parametara kamere. Na ovaj način dobiva se pomak koji robot mora ostvariti kako bi vrh alata došao do sjecišta optičkih osi kamere. Ideja je da alat prilazi toj točki iterativno kako bi se smanjila greška koja se dobiva pošto se alat ne nalazi na točnoj udaljenosti od kamere. Što je alat bliže sjecištu rade se manji i točniji ispravci pozicije alata. Udaljenosti se određuju na temelju osi kamere na način da se izračuna broj piksela u x smjeru od vrha alata do optičkog središta, a isto se napravi i za y smjer. Slika 44 prikazuje udaljenost vrha alata (crveno) od optičkih središta kamere (žuto) u smjerovima x i y osi koordinatnih sustava kamere.



Slika 44. Udaljenost vrha alata od optičkih osi prema koordinatnim sustavima kamere

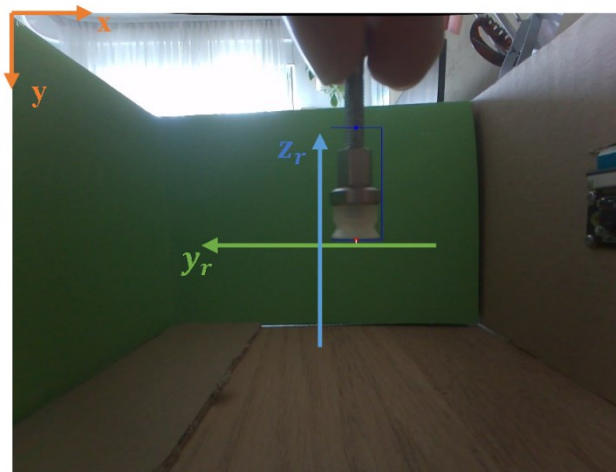
Da bi se robotu mogle zadati vrijednosti pomaka prema sjecištu optičkih osi potrebna je logika prema kojoj će se osim udaljenosti odrediti i smjer gibanja robota. Najbolji pristup i ujedno najjednostavniji jest pozicioniranje kalibracijske kutije na način da su osi koordinatnih sustava kamere prema osima baze robota postavljene paralelno. Time se na svakoj od kamere udaljenosti mogu direktno povezati sa gibanjem po osima baze robota bez dodatnog preračunavanja gibanja.

Kako bi se odredio smjer kretanja potrebno je unaprijed odrediti poziciju kalibracijske kutije u odnosu na bazu robota. Slika 45 prikazuje položaj kalibracijske kutije u odnosu na bazu robota.



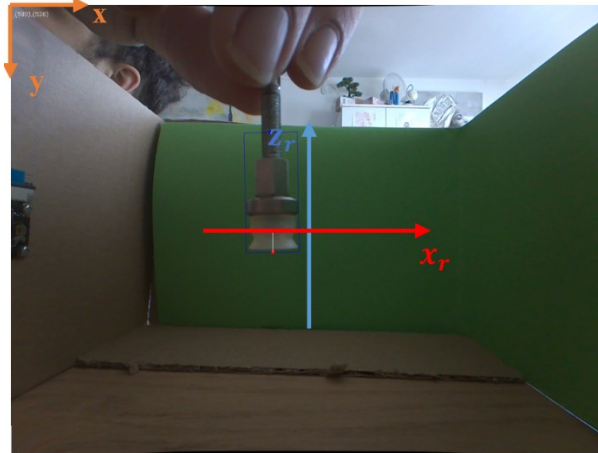
Slika 45. Položaj kalibracijske kutije u odnosu na bazu robota

Kao što je vidljivo sa slike, orijentacije osi kamere 0 poklopljene su sa z i y osima baze robota, dok su orijentacije kamere 1 poklopljene sa x i z osima baze robota. Za smjerove gibanja potrebno je problem sagledati iz perspektive slika sa kamera. Na Slika 46 prikazan je odnos koordinatnog sustava kamere 0 sa osima baze robota poklopljene s njima.



Slika 46. Odnos koordinatnog sustava kamere 0 s osima baze robota (y, z)

Prema prošloj slici očekuje se da će robot za pomicanje ići u pozitivnom smjeru osi y baze robota i u negativnom smjeru osi z baze robota kako bi alat došao u sjecište optičkih osi. Na Slika 47 prikazan je odnos koordinatnog sustava kamere 1 sa osima baze robota poklopljene s njima.



Slika 47. Odnos koordinatnog sustava kamere 1 s osima baze robota (x, z)

Kako je vidljivo iz slike, očekuje se da će robot za pomicanje ići u pozitivnom smjeru osi x baze robota i u pozitivnom smjeru osi z baze robota kako bi alat došao u sjecište optičkih osi. Slika 48 prikazuje vrijednosti udaljenosti vrha alata robota od optičkih centara sa predznakom prema kojem se određuje smjer gibanja robota.

```
y os udaljenost od sredine(cam0 - x):97pikesela, 9.75388mm
z os udaljenost od sredine(cam0 - y):-14pikesela, -1.40869mm
x os udaljenost od sredine(cam1 - x):101.989pikesela, 10.2701mm
z os udaljenost od sredine(cam0 - y):57.4881pikesela, 5.79558mm
```

Slika 48. Vrijednosti udaljenosti alata robota od optičkih centara

Ova kretanja ostvariti će se relativnim pomacima alata robota prema trenutnoj poziciji istog, a vrijednosti pomaka poslani će se robotu kao greške koje se izračunavaju pri prihvaćanju scene. Na sličan način radi se i promjena orijentacije alata pri postavljanju alata tako da se njegova os poravnava sa z osi baze robota za kalibraciju orijentacije alata. U tom slučaju mjeri se razlika udaljenosti x i y koordinata točke vrha alata i točke osi alata. Na temelju tih koordinata računaju se kutovi zakreta preko jednostavne trigonometrije pravokutnog trokuta. Kako bi se mogla odraditi kalibracija alata potrebno je još napraviti komunikaciju između robota i vizijskog sustava i definirati metode kalibracije alata.

8. ALGORITAM KALIBRACIJE ALATA

8.1. Kalibracija pozicije vrha alata robota

Algoritam kalibracije preuzet je iz [13] gdje se na temelju više različitih pozicija prirubnice robota računa vektor pozicija vrha alata u odnosu na prirubnicu. Taj vektor je ujedno i cilj kalibracije vrha alata robota. Metoda kaže da, ako se definira matrica transformacije od prirubnice robota do vrha alata robota, ona ima sljedeću formu kako prikazuje jednačba (2).

$$Q = \begin{bmatrix} 1 & 0 & 0 & TCP_x \\ 0 & 1 & 0 & TCP_y \\ 0 & 0 & 1 & TCP_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

Q matrica prikazuje matricu transformacije od prirubnice robota do vrha alata robota gdje varijable TCP_x , TCP_y i TCP_z prikazuju poziciju točke vrha alata u koordinatnom sustavu prirubnice. Nadalje, zna se da je zadovoljena jednačba (3) za svaku matricu transformacije T_{1i} od prirubnice robota do baze robota ukoliko se alat nalazi u istoj točki prostora.

$$T_{1i}Q = T_{1j}Q \quad (3)$$

Gdje je T_{1i} matrica transformacije od baze do prirubnice robota za jednu orijentaciju alata, T_{1j} je matrica transformacije od baze do prirubnice robota za drugačiju orijentaciju alata, a Q je matrica transformacije od prirubnice do alata robota te je ona ista za obje orijentacije alata. Potrebno je naglasiti da su $i, j \in [1 \dots N]$ i $i \neq j$. Dalje je potrebno prikazati kako izgledaju matrice transformacija od baze robota do prirubnice robota. One su definirane kako pokazuju jednačbe (4) i (5):

$$T_{1i} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_a & \bar{\mathbf{t}}_a \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (4)$$

$$T_{1j} = \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \\ b_{41} & b_{42} & b_{43} & b_{44} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_b & \bar{\mathbf{t}}_b \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

Gdje su a_{ij} i b_{ij} članovi matrica transformacija od baze do prirubnice robota, \mathbf{R}_a i \mathbf{R}_b su rotacijski dijelovi te matrice, a $\bar{\mathbf{t}}_a$ i $\bar{\mathbf{t}}_b$ su translacijski dijelovi te matrice. Redak niza nula i jedinice nužan je za korištenje matrica transformacija u izračunima kako bi matematika matrica

bila zadovoljena. Za razumijevanje tvorbe sustava jednadžbi za računanje kalibracije vrha alata dana je analiza na temelju prvog reda iz jednadžbe (3). On je prikazan jednadžbom (6).

$$\begin{aligned} a_{11}TCP_x + a_{12}TCP_y + a_{13}TCP_z + a_{14} \\ = \\ b_{11}TCP_x + b_{12}TCP_y + b_{13}TCP_z + b_{14} \end{aligned} \quad (6)$$

Gdje su a_{ij} i b_{ij} članovi matrica transformacija od baze do prirubnice robota, a varijable TCP_x, TCP_y i TCP_z prikazuju poziciju točke vrha alata u koordinatnom sustavu prirubnice. Iz tog izraza rješavanjem dalje dobije se jednadžba (7) koja predstavlja jedan redak iz jednadžbe (3).

$$(a_{11} - b_{11})TCP_x + (a_{12} - b_{12})TCP_y + (a_{13} - b_{13})TCP_z = - (a_{14} - b_{14}) \quad (7)$$

Dalje za svaku od jednadžbi (3) dobije se sustav triju jednadžbi kako je prikazano jednadžbom (7). Svaki sustav jednadžbi može se zapisati kako prikazuje jednadžba (8).

$$[(\mathbf{R}_a - \mathbf{R}_b)] \begin{bmatrix} TCP_x \\ TCP_y \\ TCP_z \end{bmatrix} = - [\bar{\mathbf{t}}_a - \bar{\mathbf{t}}_b] \quad (8)$$

Gdje su \mathbf{R}_a i \mathbf{R}_b rotacijski dijelovi matrice transformacije od baze do prirubnice robota, vektor $[TCP_x \ TCP_y \ TCP_z]^T$ prikazuje translacijske vrijednosti matrice transformacije od prirubnice do alata robota, a $\bar{\mathbf{t}}_a$ i $\bar{\mathbf{t}}_b$ su translacijski dijelovi matrice transformacije od baze do prirubnice robota. Korištenjem drugih orijentacija dobije se sustav jednadžbi kako je prikazano jednadžbom (8). Tada se pozicije alata robota u odnosu na prirubnicu robota dobiju kao linearno rješenje najmanjeg kvadrata sustava jednadžbi.

8.2. Kalibracija orijentacije vrha alata robota

Algoritam je preuzet iz [15] gdje se na temelju postavljanja alata tako da njegova z os bude paralelna sa z osi baze računa orijentacija alata robota u odnosu na njegovu prirubnicu. Metoda se naziva „Metoda paralelnih osi“. Matrica rotacija alata u odnosu na bazu prikazana je jednadžbom (9).

$$\mathbf{R}_{baza,TCP} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (9)$$

Gdje je $\mathbf{R}_{baza,TCP}$ rotacijski dio matrice transformacije od baze do alata robota kako bi se zadovoljio uvjet da su osi alata i baze poravnani.

Veza relativnih orijentacija osi koordinatnih sustava baze robota, prirubnice robota i vrha alata robota dana je jednadžbom (10).

$$\mathbf{R}_{baza,TCP} = \mathbf{R}_{baza,prirubnica} \mathbf{R}_{prirubnica,TCP} \quad (10)$$

Gdje je $\mathbf{R}_{baza,TCP}$ rotacijski dio matrice transformacije od baze do alata robota, $\mathbf{R}_{baza,prirubnica}$ je rotacijski dio matrice transformacije od baze do prirubnice robota, a $\mathbf{R}_{prirubnica,TCP}$ je rotacijski dio matrice transformacije od prirubnice do alata robota. Rotaciju od prirubnice do vrha alata robota može izračunati korištenjem inverza rotacijske transformacije od baze do prirubnice i rotacijske transformacije od baze do vrha alata robota kako prikazuje jednadžba (11).

$$\mathbf{R}_{prirubnica,TCP} = \mathbf{R}_{baza,prirubnica}^{-1} \mathbf{R}_{baza,TCP} \quad (11)$$

Gdje je $\mathbf{R}_{prirubnica,TCP}$ rotacijski dio matrice transformacije od prirubnice do alata robota, $\mathbf{R}_{baza,prirubnica}^{-1}$ je inverz rotacijskog dijela matrice transformacije od baze do prirubnice robota, a $\mathbf{R}_{baza,TCP}$ je rotacijski dio matrice transformacije od baze do alata robota.

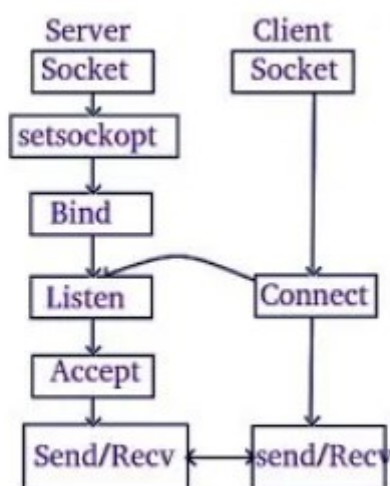
Na ovaj način dobiva se krajnje kalibrirani alat robota sa translacijama i rotacijama vrha alata robota do prirubnice robota.

9. Komunikacija vizijskog sustava i robota

Komunikacija između robota i vizijskog sustava napravljena je putem socketa jer on omogućava slanje i primanje podataka odnosno komunikaciju u dva smjera. Vizijski sustav u ovom slučaju predstavlja server na kojeg se robot spaja kao klijent putem socketa. Oni komuniciraju sa stringovima koji služe za upravljanje tijekom izvođenja programa.

9.1. Socket komunikacija

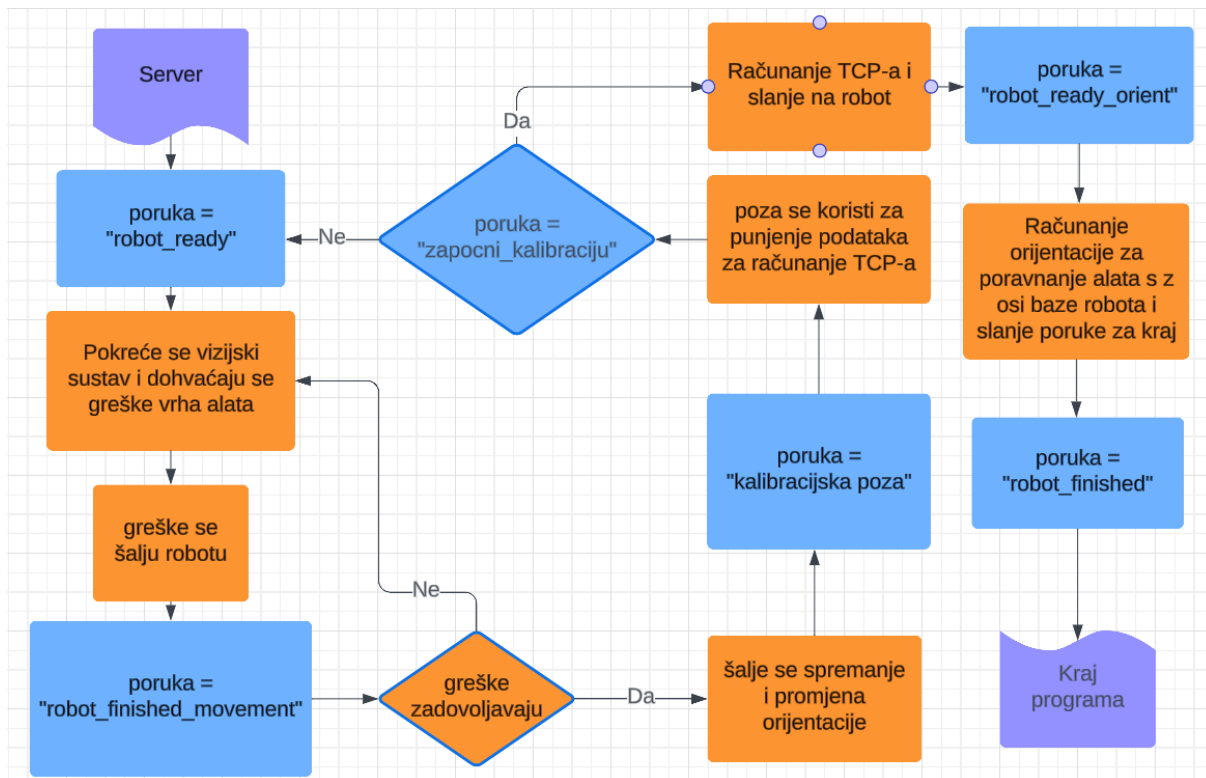
Socketi djeluju kao krajnje točke za slanje i primanje podataka između programa koji se izvode na različitim uređajima. Socket je krajnja točka dvosmjerne komunikacijske veze između dva programa koji se izvode na mreži. Socketi nude mehanizam za međuprocesnu komunikaciju uspostavljanjem imenovanih kontaktnih točaka kroz koje se odvija komunikacija. U suštini, omogućuju različitim uređajima i aplikacijama razmjenu podataka preko mreže, bilo da se radi o lokalnoj mreži ili internetu. Protokol kontrole prijenosa (eng. Transmission Control Protocol (TCP)) je protokol koji se najčešće koristi. TCP radi dvosmjerno, što znači da primatelj šalje paket odgovora natrag pošiljatelju, potvrđujući primitak poruke. Ako se ne primi točan odgovor, pošiljatelj će ponovno poslati poruku. Osim toga, TCP provodi provjeru pogrešaka na paketima kako bi osigurao integritet podataka [9]. Arhitekturu socket komunikacije prikazuje Slika 49.



Slika 49. Arhitektura socket komunikacije[9]

9.2. Tijek komunikacije - server

Slika 50 prikazuje tijek komunikacije i procese koji se odvijaju u radu sustava tokom kalibracije alata robota.



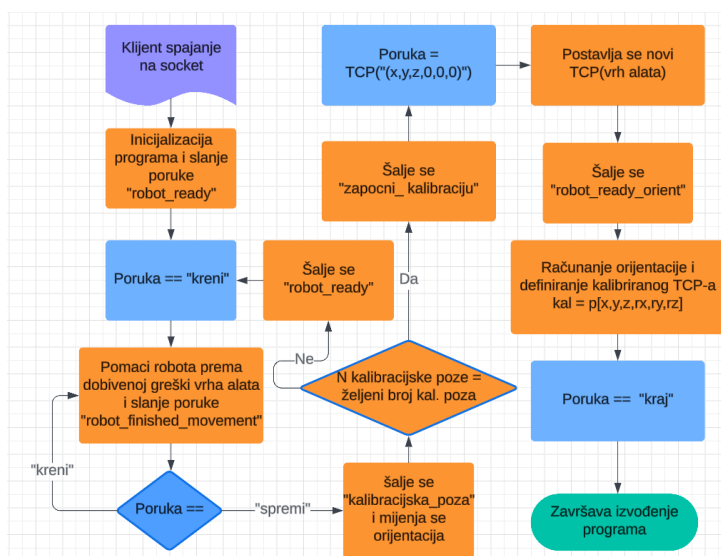
Slika 50. Tijek komunikacije na vizijском sustavu

Na slici su poruke koje dolaze sa robota prikazane plavim okvirima. Svaka poruka aktivira neki proces koji je prikazan u narančastim okvirima. U rombovima su prikazani dijelovi procesa kada se aktivira provjera koja je naznačena u njima. U početku programa otvara se server sa parametrima prikazanim na Slika 49 i čeka se spajanje klijenta i dolazak poruka od njega. Prva poruka koja se očekuje na robotu glasi „robot_ready“ što znači da je robot doveden u prostor vidnog polja kamera te da je spreman čekati pokretanje kamera, snimanje slika scene i obradu slika dok mu se ne pošalje greška vrha alata robota prema kojoj mora napraviti pomake. Po završetku tog gibanja robot šalje poruku koja glasi „robot_finished_movement“. U tom trenutku vizijski sustav provjerava vrijednosti zadnje poslane greške i, ukoliko su one veće od definiranih, ponavlja proces dohvaćanja novih grešaka vrha alata i ponovno ih šalje robotu. To se odvija sve dok greške ne postanu zadovoljavajuće. U radu su se zadovoljavajuće greške smatrale greške manje od pola milimetra. Zadovoljavajućim greškama robotu se šalje zahtjev za spremanje poze koja postaje kalibracijskom. Ta poza se šalje natrag vizijском sustavu kroz poruku „kalibracijska_poza“ te se ona koristi za punjenje podataka za računanje vrha alata.

Ukoliko je robot spremio već dovoljan broj poza, kojih minimalno mora biti 4, a njihov broj definiran je proizvoljno, robot šalje poruku „zapocni_kalibraciju“. Ona aktivira proces računanja vrha alata (eng. Tool Center point - TCP) te ga šalje natrag robotu. Ukoliko robot nije dostigao broj poza koje su mu definirane šalje se poruka „robot-ready“ te se cijeli opisani proces ponavlja. Jednom kada robot dobije podatke o kalibriranom TCP-u šalje vizijskom sustavu poruku koja glasi „robot_ready_orient“ čime se na sustavu putem kamera provjerava koliko se alat treba rotirati kako bi njegova os bila poravnana sa bazom robota. Nakon što se ti podaci izračunaju, šalju se robotu, a nakon toga robotu se šalje poruka za kraj programa. Robot tada šalje poruku da je program završio izvođenje „robot_finished“, čime završava izvođenje programa. Bitno je napomenuti da je potrebno bilo koje poruke koje se šalju na drugi sustav dobro formatirati kako bi se osiguralo da se ona može dohvatiti. Tako na primjer, kada se robotu šalju pomaci koje mora ostvariti, podaci moraju biti formatirani kao brojevi u zagradama i odvojeni zarezom. Tako se podatak o pomacima šalje na sljedeći način : (broj1,broj2,broj3,broj4,broj5,broj6).

9.3. Tijek komunikacije - klijent

Prije nego se program tj. vizijski sustav spoji sa stvarnim robotom i napravi sve radnje za kalibraciju vrha alata uvijek je bolje napraviti simulaciju. Prednost simulacije je da se njome može provjeriti cijela komunikacija i tok izvođenja komandi, bez rizika oštećenja opreme. Simulacija je napravljena unutar programa Polyscope koji služi kao simulator za UR robote. Robot koji je korišten u simulaciji jest robot UR3e. U ovom poglavlju osim komunikacije objasniti će se i prikazati program koji se izvodi na robotu za kalibraciju alata. Za početak Slika 51dan je dijagram toka izvođenja programa na robotu.



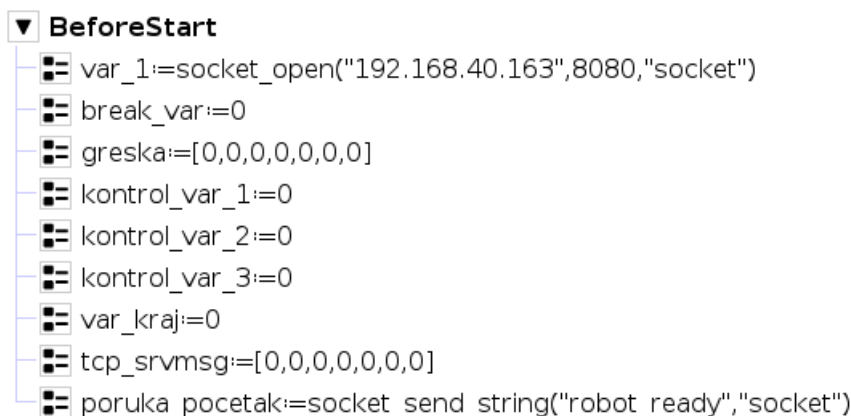
Slika 51. Dijagram izvođenja programa robota(klijenta)

Kako bi se bolje razumjelo što se događa na strani klijenta dat će se opis cijelog programa robota sa svim funkcijama unutar njega.

9.3.1. Program robota

9.3.1.1. Inicijalizacija programa

Kod inicijalizacije prikazan je Slika 52.

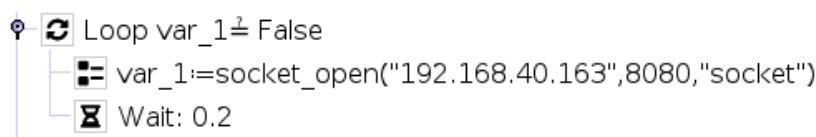


Slika 52. Inicijalizacija programa robota

Kao što je vidljivo sa slike, unutar inicijalizacije programa napravljena je cjelokupna priprema potrebnih varijabli i otvaranje komunikacije sa serverom. Varijabla `var_1` koristi `socket_open` funkciju UR-a da bi otvorila socket prema serveru tj. vizijskom sustavu. Potrebno je definirati na kojoj je IP adresi vizijski sustav i na kojem portu osluškuje potencijalne klijente. Dodatno je potrebno imenovati socket kako bi se znalo koji socket se koristi pri dohvaćanju poruka ili slanju istih. Varijabla `break_var` može poslužiti za zaustavljanje programa ukoliko je to potrebno. Varijable `kontrol_var` služe za kontrolu tijeka programa odnosno definiranje kada se koji dio programa izvodi. Za označavanje kraja programa i za završne radnje prije kraja programa koristi se varijabla `var_kraj`. Varijable `greska` i `tcp_srvmsg` inicijalizirane su jer će se u njih spremati poruke koje su `ascii_float` formata.

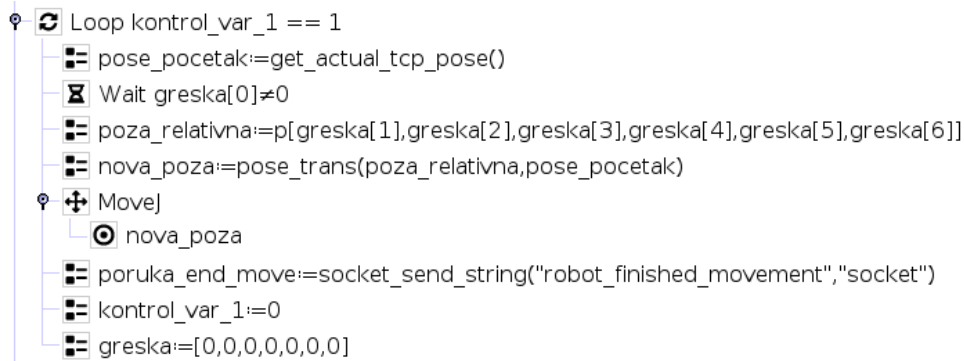
9.3.1.2. Robot program

Glavni program robota je podijeljen prema funkcionalnim dijelovima koda po redu. Slika 53 prikazuje dio programa koji se izvodi ukoliko se socket prekine.



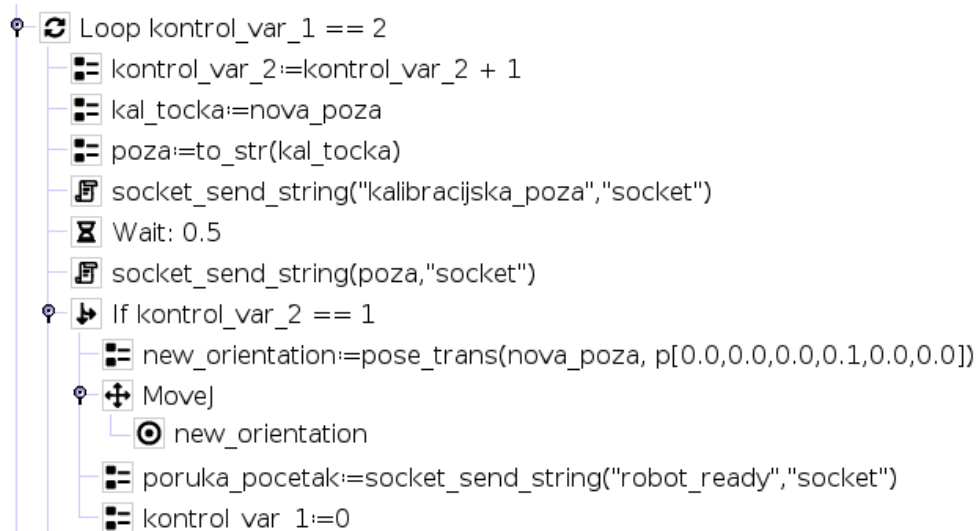
Slika 53. Dio programa koji se izvodi ako se zatvori socket

Kako je sa slike vidljivo, u ovom dijelu pokušava se ponovno otvoriti socket komunikacija. Nakon toga se čeka 0,2 sekunde i ukoliko je otvaranje socketa neuspješno ponavlja se dok god se ne uspostavi komunikacija. Na Slika 54 prikazan je dio programa koji se izvodi kada je kontrol_var_1 jednaka jedinici.



Slika 54. Dio programa koji se izvodi za kontrol_var_1 = 1

Robot se prije pokretanja programa prema zadatku treba dovesti u položaj vidnog polja kamera. Pošto je to definirano ta se poza robota može spremirati kao pose_pocetak. Nakon toga se čeka poruka od vizijskog sustava o greškama vrha alata. Vrijednosti greške zapisuju se u varijablu poza_relativna. Na temelju početne poze i grešaka radi se relativni pomak od početne poze za vrijednosti grešaka, kako bi alat došao u optički centar kamera. Šalje se poruka da je robot završio s kretanjem, a varijable kontrol_var_1 i greska vraćaju se u početno stanje. Na Slika 55 je prikazan dio programa koji se izvodi kada je prva kontrolna varijabla jednaka dva.



Slika 55. Dio programa koji se izvodi kada je kontrol_var_1 = 2

Kada je prva kontrolna varijabla jednaka dva prvo se za jedan povećava iznos druge kontrolne varijable. Prva kontrolna varijabla koristi se za definiranje dijela programa koji se izvodi na temelju poruka sa servera, dok druga kontrolna varijabla služi za definiciju promjene

orijentacije alata robota koja mora biti drugačija za svaku od kalibracijskih poza. Kada se povećala vrijednost druge kontrolne varijable poza koja je spremljena u zadnjoj iteraciji prijašnjeg dijela programa sprema se kao kalibracijska točka pa se zatim ona pretvara u string format. Prvo se šalje poruka „kalibracijska_poza“ kako bi se server mogao pripremiti za čitanje i obradu poze. Čeka se pola sekunde i nakon toga se šalje kalibracijska točka serveru. S obzirom da se druga kontrolna varijabla povećala za jedan, i sada iznosi jedan, definira se relativna poza sa drugačijom orijentacijom i robot se pomiče prema njoj. Nakon toga serveru se šalje poruka „robot_ready“ kako bi se opet dobila nova kalibracijska točka. Na kraju se prva kontrolna varijabla postavlja na nulu. Kada se sljedeći put krene izvoditi taj dio programa druga kontrolna varijabla će iznositi dva, te će se opet od robota zahtijevati da promjeni orijentaciju. Orijentacija se mijenja na način da se negira ona prošla i zada nova. U ovom slučaju program je napravljen za četiri različite poze, ali njih može biti proizvoljno sve dok ih je više od tri. Slika 56 prikazuje kako se mijenjaju orijentacije s obzirom na promjenu druge kontrolne varijable i kako završava proces mijenjanja orijentacija alata za zadnju kalibracijsku pozu.



Slika 56. Dio programa za mijenjanje orijentacije alata robota

Sa slike je vidljivo da definiranjem zadnje kalibracijske poze i slanjem na vizijski sustav robot postavlja prvu kontrolnu varijablu na trojku, drugu kontrolnu varijablu postavlja u početno stanje i uključuje treću kontrolnu varijablu postavljanjem iste na jedinicu. Uz ovo, vizijskom sustavu šalje poruku „zapocni_kalibraciju“ kako bi se znalo da je robot poslao onoliko poza koliko ih je bilo namijenjeno za kalibraciju alata. Slika 57 prikazuje dio programa koji se izvodi kada je prva kontrolna varijabla jednaka trojci.

```

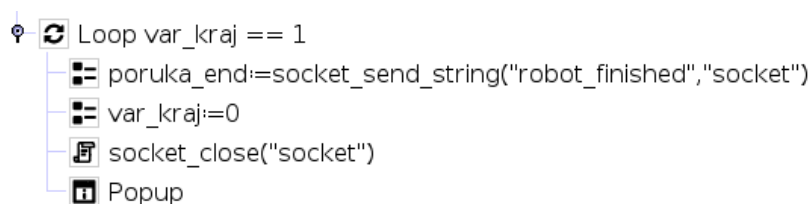
Loop kontrol_var_1 == 3
  Wait tcp_srvmsg[0] ≠ 0
  tcp:=p[tcp_srvmsg[1],tcp_srvmsg[2],tcp_srvmsg[3],tcp_srvmsg[4],tcp_srvmsg[5],tcp_srvmsg[6]]
  set_tcp(tcp)
  socket_send_string("robot_ready_orient","socket")
  kontrol_var_3:=0
  Wait greska[0] ≠ 0
  pos_1:=get_actual_tcp_pose()
  rel_pos:=p[greska[1],greska[2],greska[3],greska[4],greska[5],greska[6]]
  pos_new:=pose_trans(rel_pos, pos_1)
  rot:=p[pos_1[0],pos_1[1],pos_1[2],pos_new[3],pos_new[4],pos_new[5]]
  MoveJ
  tcp_b:=get_actual_tcp_pose()
  tcp_br:=p[tcp_b[0],tcp_b[1],tcp_b[2],3.1416,0,0]
  set_tcp(p[0,0,0,0,0,0])
  Wait: 1.0
  flange_b:=get_actual_tcp_pose()
  kal_orient:=pose_trans(pose_inv(flange_b), tcp_br)
  kal:=p[tcp[0],tcp[1],tcp[2],kal_orient[3],kal_orient[4],kal_orient[5]]
  set_tcp(kal)
  kontrol_var_1:=0

```

Slika 57. Dio programa koji se izvodi za kontrol_var_1 = 3

Na slici je vidljivo da se čeka poruka od strane servera koja sadrži podatke o kalibriranom alatu robota. Nakon što se poruka primi, u varijablu tcp spremaju se podaci koje ona sadrži te se ti podaci koriste za postavljanje alata robota na kalibrirane parametre translacije. Potom se vizijском sustavu šalje poruka „robot_ready_orient“ čime se definira da je robot spreman za kalibraciju orijentacije. Treća kontrolna varijabla postavlja se na nulu i čeka se poruka robota o potrebnim zakretima za postavljanje osi alata u takvu poziciju da je ona poravnana sa z osi baze robota. Zakreti dobiveni od vizijского sustava se spremaju i robot se postavlja u zadani položaj, ne provodeći translacijska gibanja već samo rotaciju. Nakon rotacije se zapisuje trenutni položaj alata u odnosu na bazu u varijablu tcp_b. Kako bi bio zadovoljen uvjet iz jednadžbe (9) u poglavlju 8.2 definira se potreban odnos orijentacija alata u varijablu tcp_br. Nakon definiranja varijable tcp_br alat se postavlja na prirubnicu kako bi se dobila rotacija prirubnice u odnosu na bazu i zapisuje se u varijablu flange_b. Varijabla kal_orient sadržava rotacijski dio transformacije kalibriranog alata u odnosu na prirubnicu robota prema jednadžbi (11). Kalibrirani alat je tada kombinacija ranije dobivenih translacija i rotacija dobivenih u ovom dijelu programa, a sadržan je u varijabli kal, na koju se postavlja alat robota. Nakon izvođenja svih radnji prva kontrolna varijabla se ponovno postavlja na nulu. Ovime je

kalibracija alata robota završena. Slika 58 prikazuje dio programa koji se izvodi kada server pošalje poruku „kraj“ i kada je varijabla var_kraj postavljena na jedinicu.

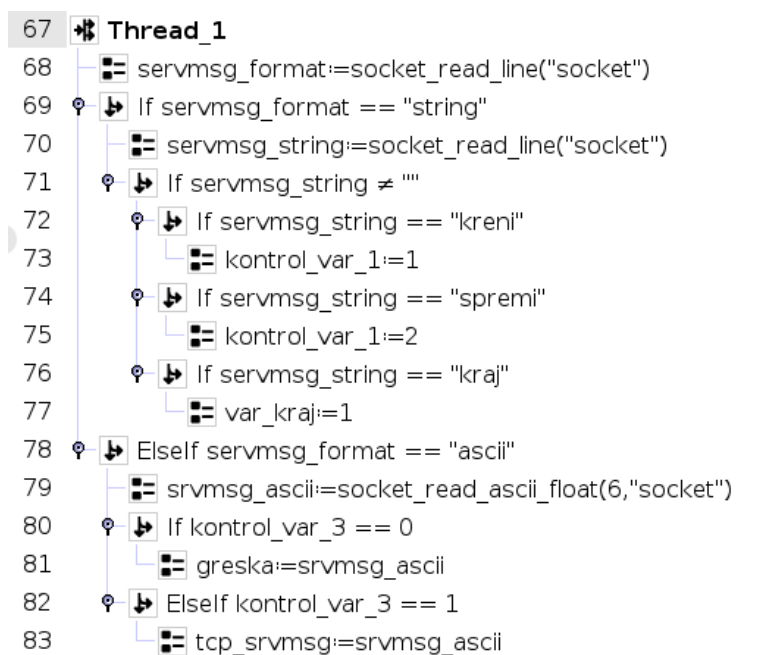


Slika 58. Dio programa koji se izvodi za var_kraj = 1

Poruka koja se šalje serveru glasi „robot_finished“ koja označuje da je on primio podatke o kalibriranom alatu. Tada se varijabla var_kraj postavlja na nulu i zatvara se komunikacija prema serveru. Na kraju se postavlja skočni prozor koji na ekran javlja podatke o definiranom kalibriranom alatu robota. U prozoru se može zaustaviti izvođenje programa pritiskom na tipku „Stop program“.

9.3.1.3. Thread komunikacije

Ostaje prikazati logiku komunikacije robota sa serverom. Kako bi robot mogao u bilo kojem trenutku primiti poruke, napravljen je tzv. thread koji se izvodi cijelo vrijeme u pozadini bez obzira na glavni dio programa robota. Slika 59 prikazuje thread za komunikaciju.



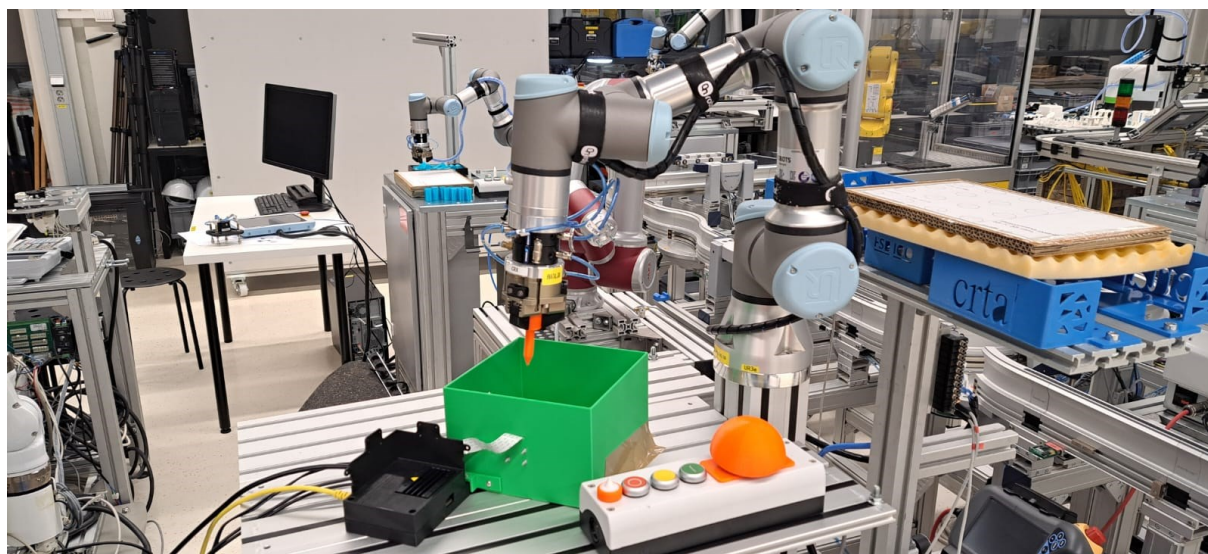
Slika 59. Thread za komunikaciju s serverom

U ovom dijelu programa se cijelo vrijeme provjerava dolazi li poruka sa servera i trenutna poruka sprema se u varijablu `servmsg_format` jer svaka poruka koja se prima sa servera prvo šalje format poruke, a zatim funkcionalni dio poruke. Ukoliko nijedan od očekivanih formata nije sadržaj poruke ili ako server nije poslao nikakvu poruku ne događa se ništa, već se ponovo čeka poruka koja zadovoljava sadržajem. Ukoliko je sadržaj `servmsg_format` varijable „string“ znači da se sa servera šalje poruka koja sadržava string. Nakon toga se naredbom `socket_read_line` čita sadržaj poruke i sprema se u varijablu `servmsg_string`. Kada je sadržaj poruke „kreni“, prva kontrolna varijabla postavlja se na jedinicu i u glavnom programu se izvodi dio programa kojem je to uvjet. Isto tako, izvodi se drugi dio programa, ali samo ako je sadržaj poruke „spremi“ da se prva kontrolna varijabla postavi na dva. Ukoliko je sadržaj poruke „kraj“ varijabla `var_kraj` postavlja se na jedan i izvodi se dio programa predviđen za taj uvjet. S druge strane, `servmsg_format` može imati sadržaj „ascii“ što znači da se sa servera šalje lista brojeva formata float te se oni mogu pročitati jedino naredbom `socket_read_ascii_float`. Ukoliko je treća kontrolna varijabla jednaka nuli poruka se sprema u varijablu `greska`, a ukoliko je ona jednaka jedinici poruka se sprema u varijablu `tcp_srvmsg`.

10. IMPLEMENTACIJA I REZULTATI KALIBRACIJE ALATA

10.1. Postav za validaciju kalibracije

Kako je prije rečeno, robot koji se koristi je UR3e. Na njegovoj stanici fiksirani su kalibracijska kutija, tako da njen položaj odgovara osima baze robota, i Jetson-nano sa kamerama. Alati koji su korišteni za validaciju pričvršćeni su za robot putem hvataljke, a također su simetrični i cilindričnog oblika. Slika 60 prikazuje postav za validaciju kalibracije.



Slika 60. Postav na robotskoj stanici

10.2. Prvi alat – kalibracijski šiljak

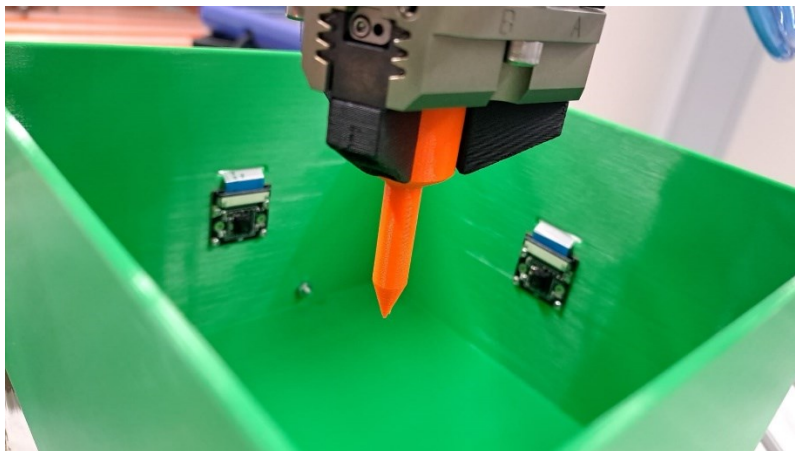
Prvi alat na kojem se radi validacija kalibracije prikazan je Slika 61.



Slika 61. Alat 1 - kalibracijski šiljak

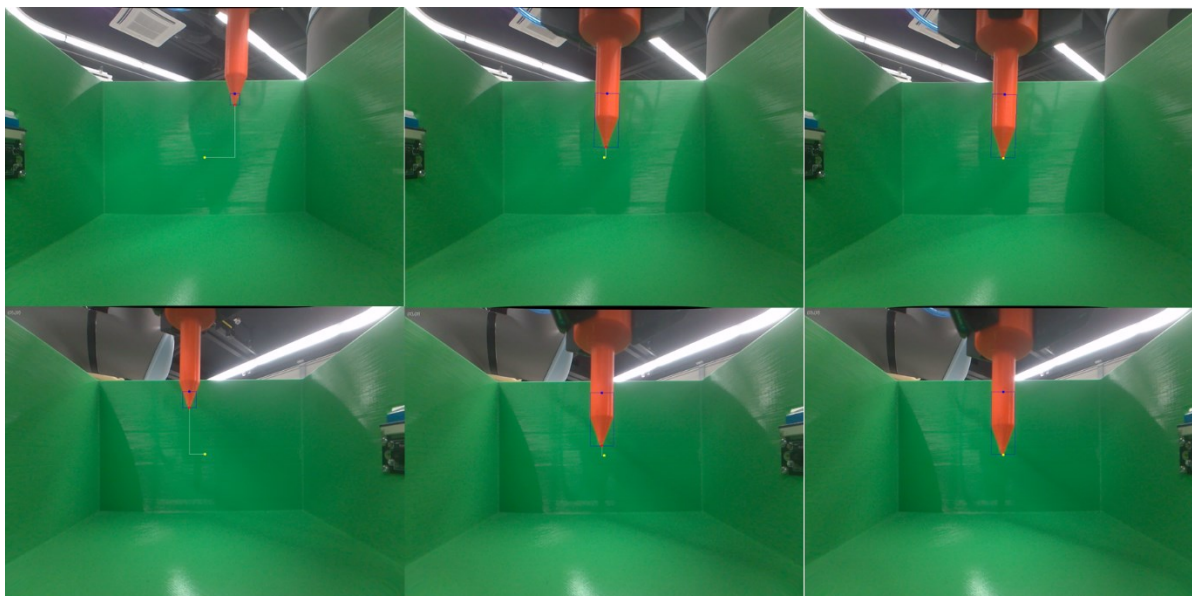
Ovaj alat karakterizira jednobojnost i jasno definirani vrh alata s obzirom da se radi o šiljku. Alat nije zahtijevan za segmentaciju i jasna je razlika u boji alata i pozadine slike. Translacijski parametri kalibracije bi trebali biti mali u x i y smjeru te značajni za z smjer alata robota pošto je alat praktički u osi prirubnice. Orijentacije će naravno ovisiti o orijentaciji prirubnice, ali bez obzira na to, z os alata mora biti u istoj osi kao i os simetrije alata.

Prvi korak kalibracije je ručno dovođenje alata robota u vidno polje kamera za kalibraciju. Slika 62 prikazuje taj korak.



Slika 62. Dovedeni alat u prostor vidnog polja kamera

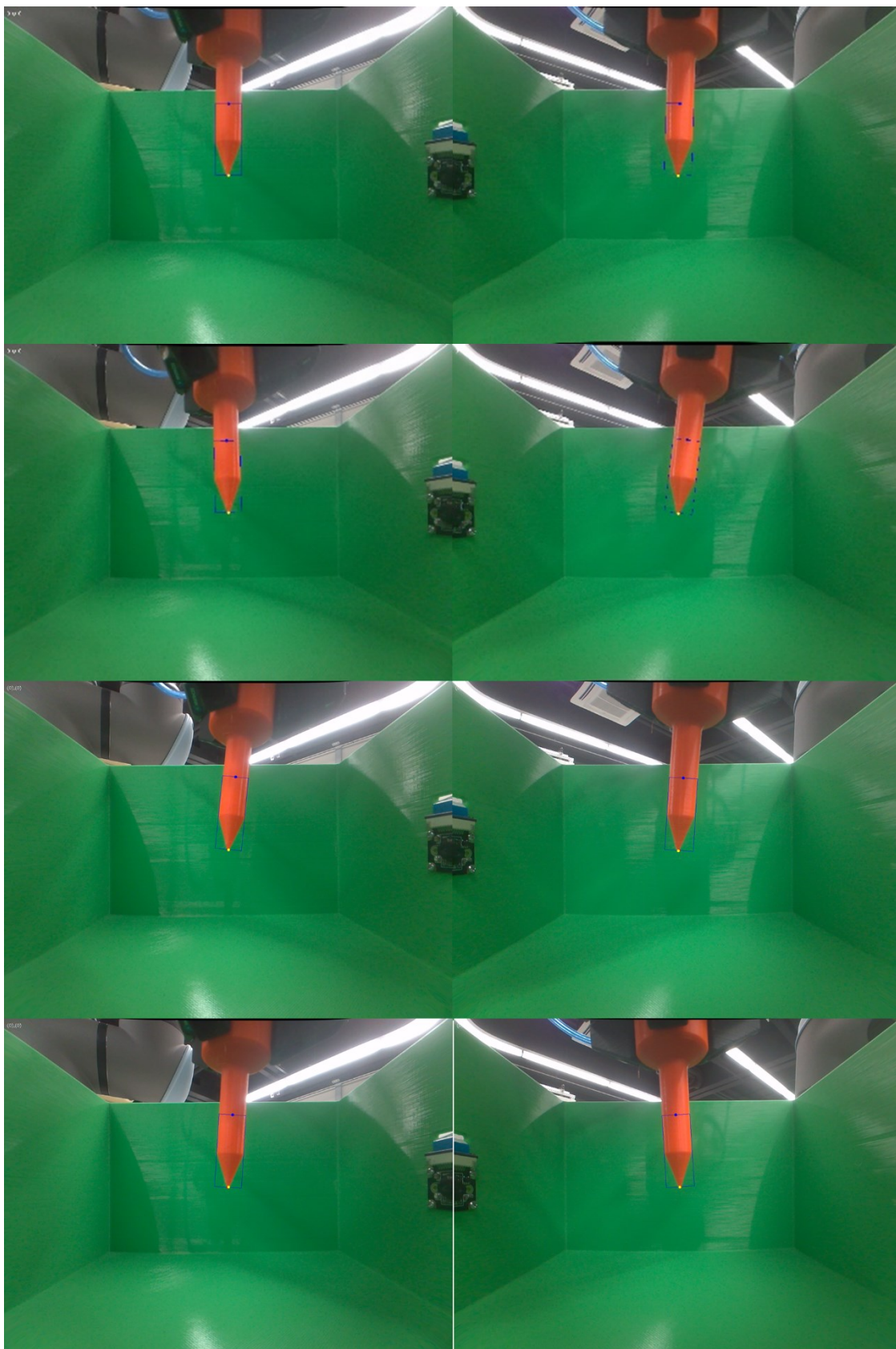
Na temelju trenutne orijentacije robot se pomiče dok alat ne dođe dovoljno blizu sjecišta optičkih osi kamera. To se radi iterativno kako je prije objašnjeno u poglavlju 7. Tako se za prvu orijentaciju dobivaju sljedeće snimljene slike, prikazane Slika 63.



Slika 63. Set slika za prvu kalibracijsku točku s obje kamere - alat 1

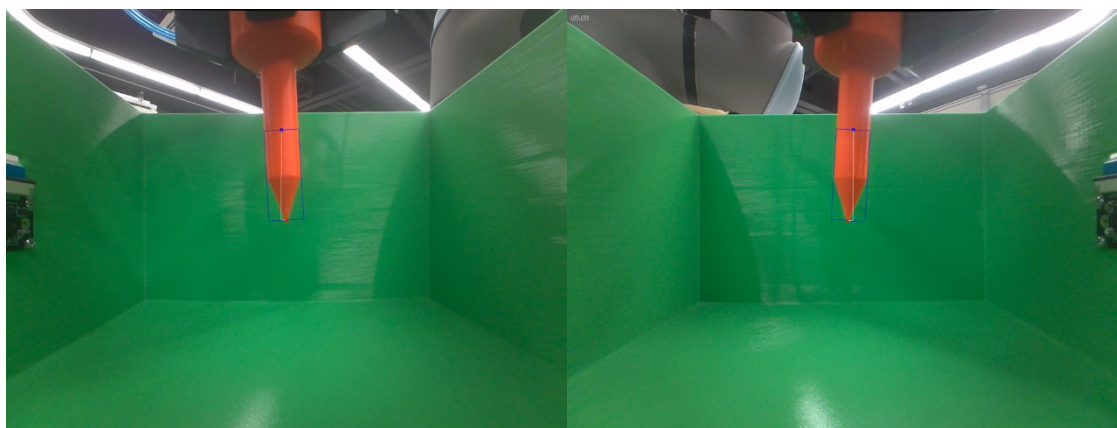
Sa slike se vidi da za prvu kalibracijsku točku alat dolazi u točnu poziciju u tri koraka. Zbog načina na koji se mjeri udaljenost alata od kalibracijske točke on dolazi do nje u više koraka, međutim što je alat bliže kalibracijskoj točki rade se manji i točniji ispravci pozicije alata. U ovom slučaju definirano je da će broj kalibracijskih točaka biti četiri. Zbog toga se od robota zahtijeva da pristupi istoj točki u prostoru sa četiri različite orijentacije da se zadovolji uvjet.

Slika 64 prikazuje set kalibracijskih točkaka za prvi alat po redu za obje kamere.



Slika 64. Set kalibracijskih točkaka s obje kamere – prvi alat

Orijentacija alata mjeri se opet na temelju slika sa kamera gdje se gleda razlika pozicije vrha alata i druge točke koja je na istoj osi, odnosno osi simetrije alata. Na temelju tih pozicija računaju se kutovi za koje se robot treba zakrenuti kako bi alat bio poravnat sa z osi baze robota i da bi se mogla napraviti kalibracija orijentacije. Slika 65 prikazuje slike preko kojih se računa zakret robota kako bi se os alata poravnala sa z osi baze robota.



Slika 65. Slike sa obje kamere za poravnanje osi alata sa z osi baze robota - prvi alat

Nakon što se na temelju ovih slika izračuna zakret, on se šalje na robot. Alat robota se tada poravnava sa z osi baze robota te se zapisuje pozicija vrha alata u odnosu na bazu. Nakon poravnanja se zapisuje pozicija prirubnice robota u odnosu na bazu robota i računa se orijentacija alata u odnosu na prirubnicu kako je pokazano u poglavlju 9.2. Kao provjera rezultata kalibracije napravljena je kalibracija korištenjem vizijskog sustava i klasična kalibracija alata na robotu. Klasična kalibracija alata robota definirana je od strane proizvođača. Za UR robote ona se radi na način da se alat robota dovede u istu točku prostora sa četiri različite orijentacije kako bi se dobili translacijski parametri alata. Kalibracija orijentacije radi se na način da se alat postavlja tako da njegova z gleda u smjeru osi z neke definirane reference kao što su baza robota, ploha definirana od strane korisnika i sl.

Tablica 1 prikazuje usporedbu kalibracije vizijskim sustavom i klasične kalibracije alata robota.

Tablica 1. Rezultati kalibracije - prvi alat

	Kalibracija – vizijski sustav	Kalibracija - robot	Razlika	
x[mm]	1,68	1,69	0,01	Translacija
y[mm]	-0,94	-0,96	0,02	
z[mm]	177,42	177,49	0,07	
Rx[rad]	0,0440	0,0444	0,0004	Rotacija
Ry[rad]	-0,0082	0,0087	0,0005	
Rz[rad]	2,0311	2,0311	0,0000	

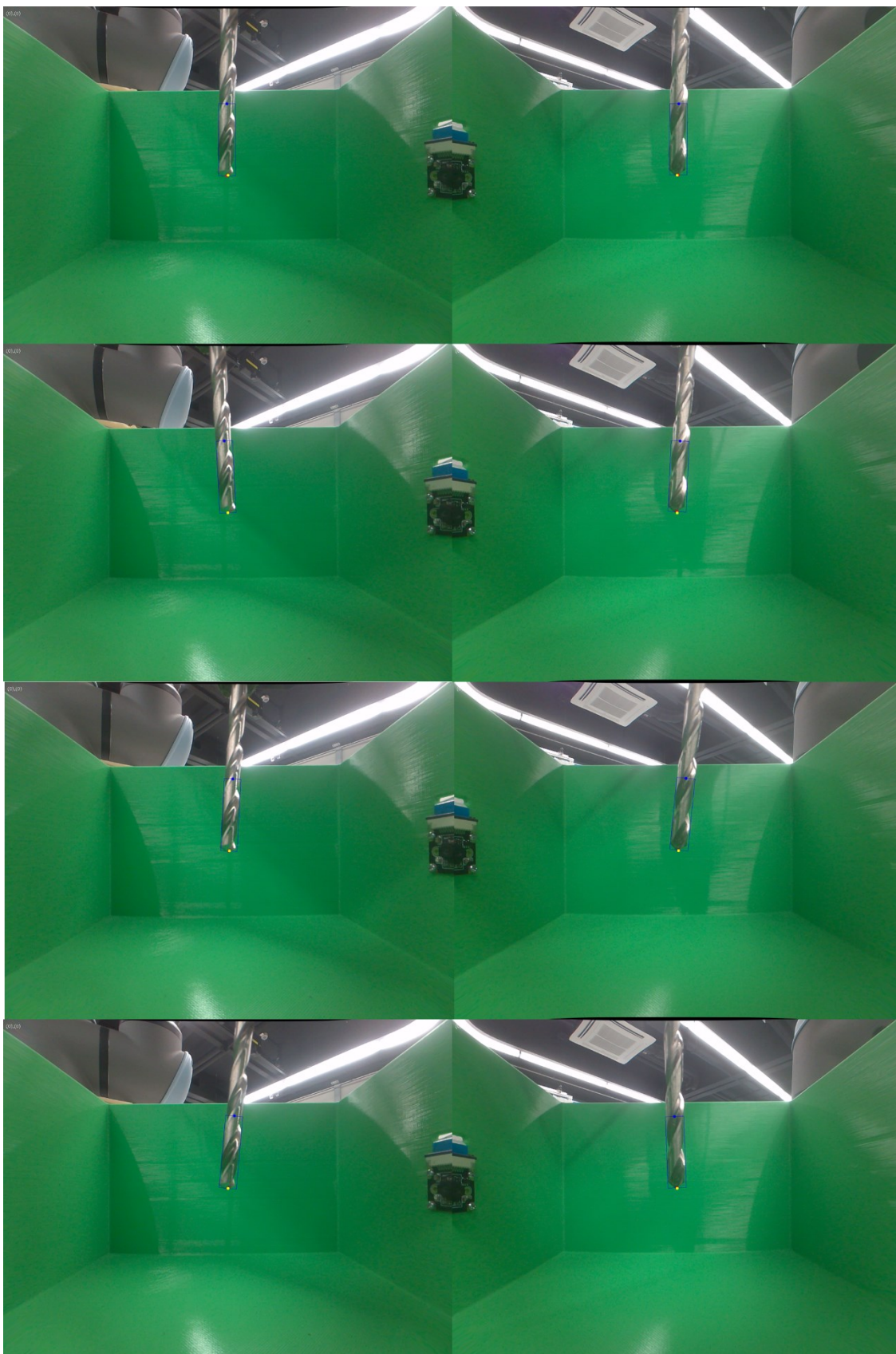
Iz tablice se vidi da su rezultati kalibracije gotovo identični čime se može zaključiti da kalibracija vizijskim sustavom dobro radi. Male razlike u kalibraciji pokazuju da je klasična kalibracija dovoljno dobro odrađena što se i može očekivati za ovakvu vrstu alata kojemu je vrh jasno istaknut. U slučajevima gdje je teže odrediti vrh alata te svaki put dovesti robot u istu točku prostora očekuje se veća razlika rezultata. Za ostale alate prikazat će se izgled alata, kalibracijske točke i slike za računanje zakreta robota radi kalibracije orijentacije alata. Na kraju se prikazuje razlika kalibracije alata korištenjem vizijskog sustava i klasične kalibracije alata robota.

10.3. Drugi alat – svrdlo

Drugi alat korišten za validaciju jest svrdlo što predstavlja metalni alat na kojem se može pojaviti lažna boja na površini alata, što otežava pronalazak točke vrha alata. Zbog tog problema mogu se generirati greške u kalibraciji alata koja mora biti precizna za primjenu u industriji. Sami alat jednostavne je izvedbe. Slika 66 prikazuje drugi alat korišten za validaciju.

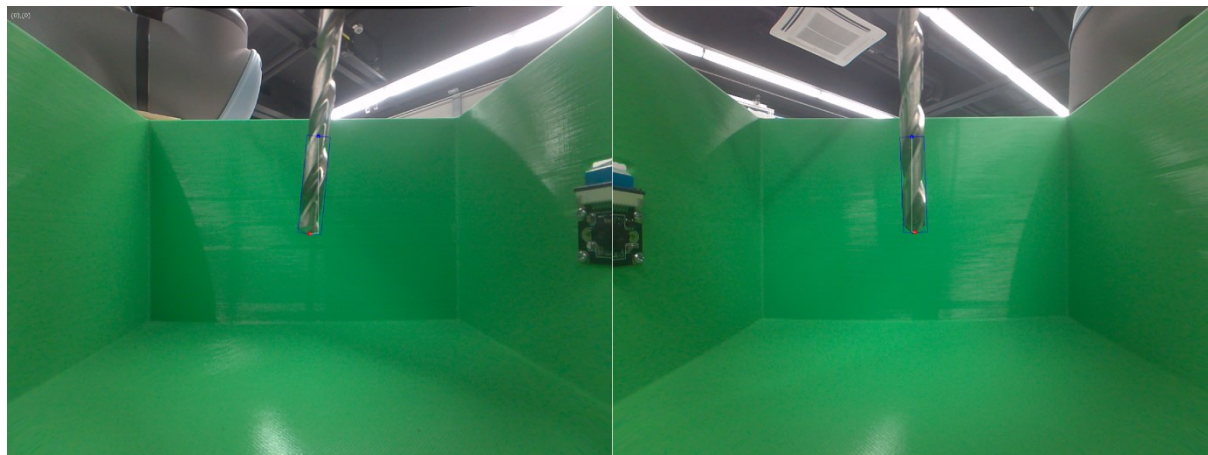
**Slika 66. Drugi alat - svrdlo**

Slika 67 prikazuje set kalibracijskih točaka za drugi alat.



Slika 67. Set kalibracijskih točaka s obje kamere - drugi alat

Slika 68 prikazuje slike preko kojih se računa zakret robota kako bi se os drugog alata poravnala sa z osi baze robota.



Slika 68. Slike sa obje kamere za poravnanje osi alata sa z osi baze robota - drugi alat

Tablica 2 prikazuje usporedbu kalibracije drugog alata sa korištenjem vizijskog sustava i klasične kalibracije alata robota.

Tablica 2. Rezultati kalibracije – drugi alat

	Kalibracija – vizijski sustav	Kalibracija - robot	Razlika	
x[mm]	-0,05	-2,46	2,41	Translacija
y[mm]	-1,50	0,57	2,07	
z[mm]	206,11	208,18	2,07	
Rx[rad]	-0,0002	-0,0048	0,0046	Rotacija
Ry[rad]	-0,0368	0,0120	0,0488	
Rz[rad]	2,6356	2,6029	0,0327	

Kao što je vidljivo iz tablice razlika kalibracija iznosi oko dva milimetra, čime se može pretpostaviti mogućnost generiranja greške uvođenjem faktora ljudske greške u klasičnoj kalibraciji alata robota. Greška može biti generirana i korištenjem vizijskog sustava za ovu primjenu.

10.4. Treći alat – vakuumska hvataljka

Vakuumska hvataljka predstavlja tip alata koji je kompliciranije izvedbe, ima metalne dijelove koji se mogu lažno obojati, što nije pogodno za segmentaciju, a time ni za detektiranje vrha alata korištenjem trenutne metode. Slika 69 prikazuje treći alat korišten za validaciju kalibracije.



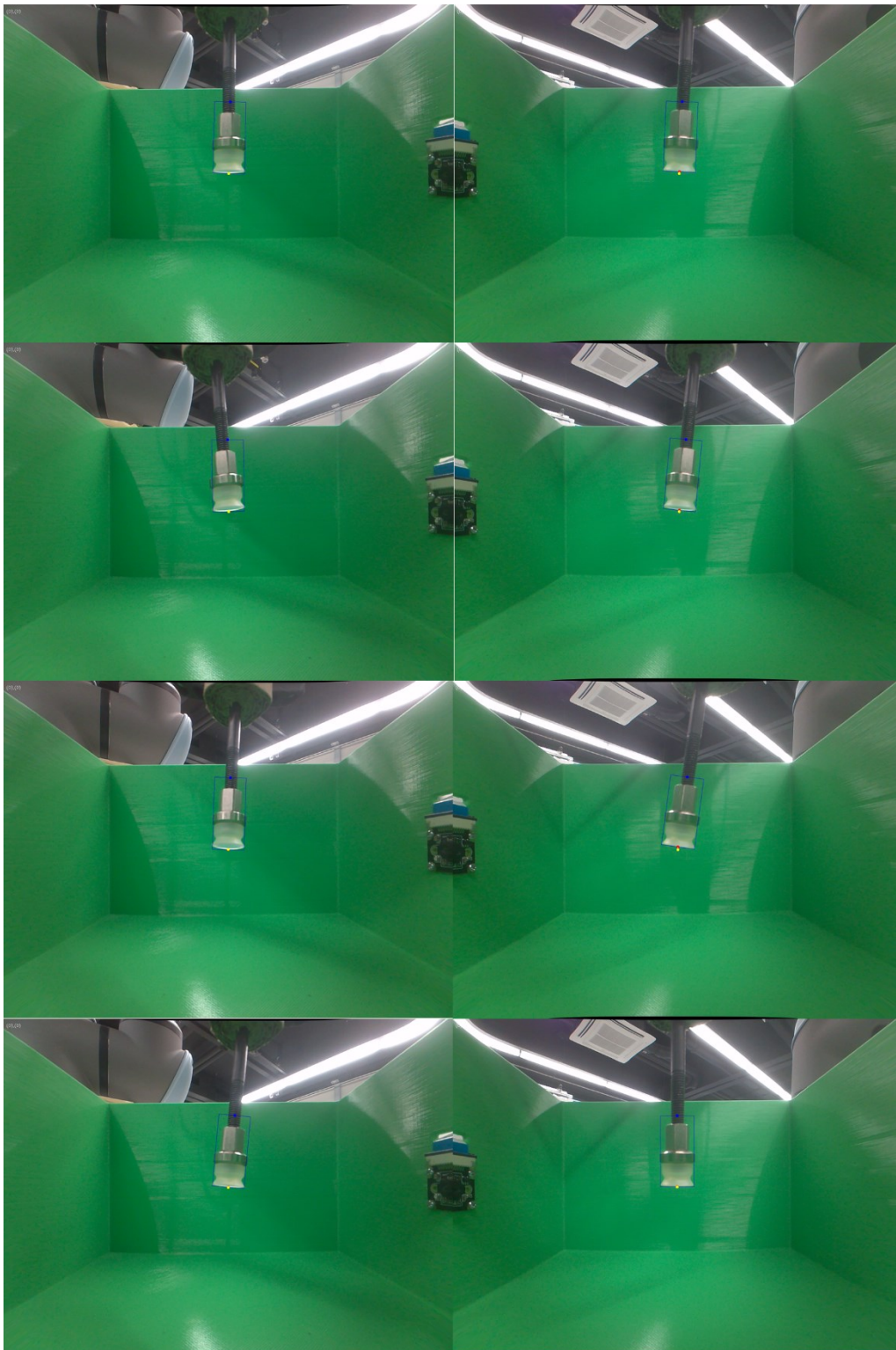
Slika 69. Treći alat - vakuumska hvataljka

Slika 70 prikazuje slike preko kojih se računa zakret robota kako bi se os trećeg alata poravnala sa z osi baze robota.



Slika 70. Slike sa obje kamere za poravnanje osi alata sa z osi baze robota - treći alat

Slika 71 prikazuje set kalibracijskih točka za treći alat.



Slika 71. Set kalibracijskih točaka s obje kamere - treći alat

Tablica 3 prikazuje usporedbu rezultata kalibracije trećeg alata korištenjem vizijskog sustava i klasične kalibracije alata.

Tablica 3. Rezultati kalibracije - treći alat

	Kalibracija – vizijski sustav	Kalibracija - robot	Razlika	
x[mm]	-3,52	-1,30	2,22	Translacija
y[mm]	-0,34	-5,11	4,77	
z[mm]	198,91	201,32	2,41	
Rx[rad]	0,0612	0,0685	0,0073	Rotacija
Ry[rad]	-0,0607	-0,0545	0,0062	
Rz[rad]	2,6359	2,6353	0,0006	

Iz tablice je vidljivo da se, korištenjem kompliciranijeg alata, razlike između klasične kalibracije i kalibracije vizijskim sustavom povećavaju. Najveća razlika iznosi skoro pet milimetara što je vrlo značajno ukoliko je zahtjev na preciznost velika.

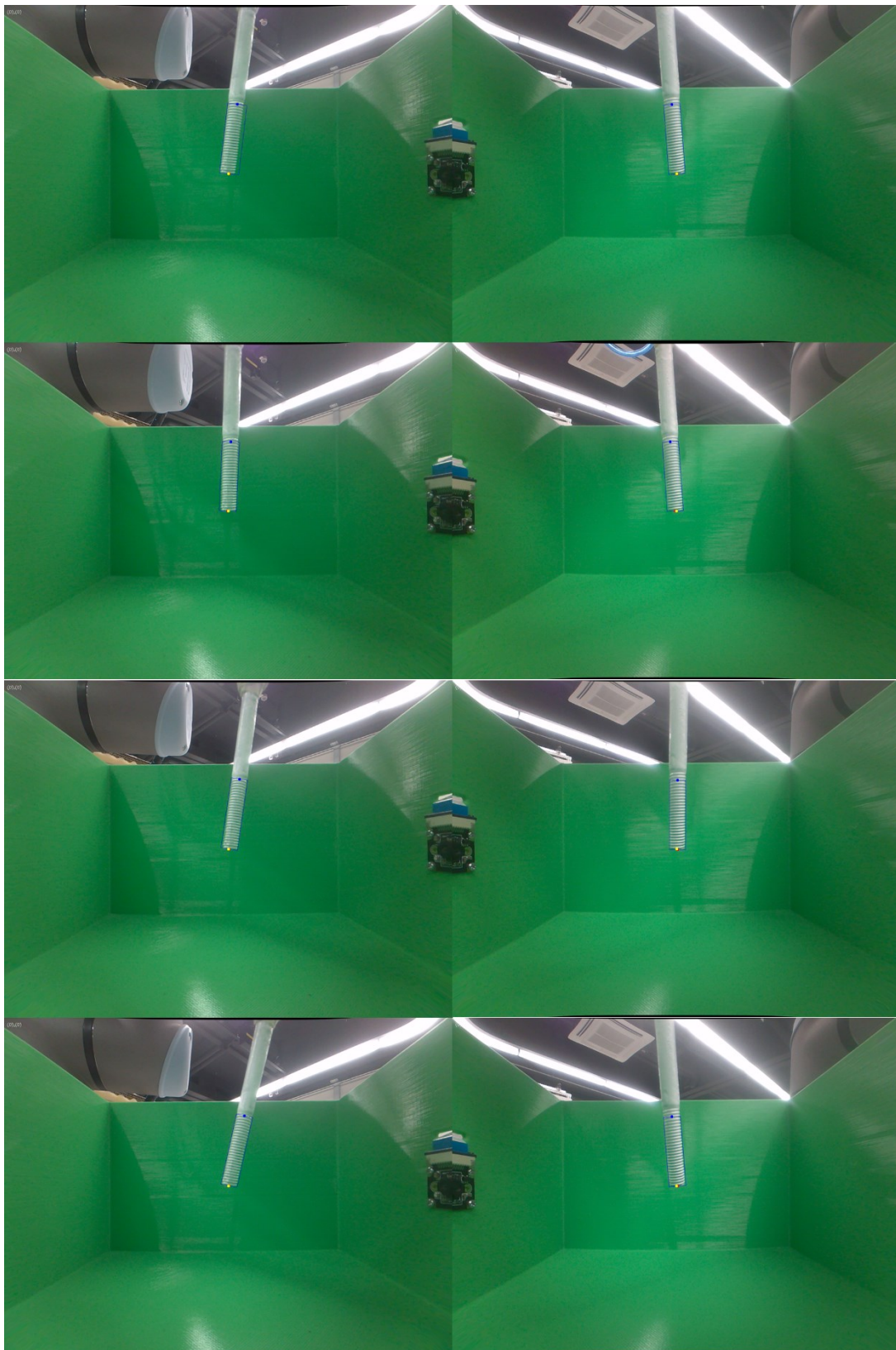
10.5. Četvrti alat – Alat pod kutem

Četvrti alat koji se koristi za validaciju karakterizira jednostavnost izvedbe i oblika vrha alata, dok sama os alata nije ista kao i os prirubnice. Najsličniji alat koji se koristi u industriji koji bi opisao ovaj alat jest robotski alat za zavarivanje. Slika 72 prikazuje alat pod kutem korišten za validaciju kalibracije.



Slika 72. Četvrti alat – improvizirani alat

Slika 73 prikazuje set kalibracijskih točka za četvrti alat.



Slika 73. Set kalibracijskih točaka s obje kamere - četvrti alat

Slika 74 prikazuje slike preko kojih se računa zakret robota kako bi se os četvrtog alata poravnala sa z osi baze robota.



Slika 74. Slike sa obje kamere za poravnanje osi alata sa z osi baze robota - četvrti alat

U Tablica 4 prikazana je usporedba kalibracije četvrtog alata korištenjem vizijskog sustava i klasične kalibracije alata.

Tablica 4. Rezultati kalibracije - četvrti alat

	Kalibracija – vizijski sustav	Kalibracija - robot	Razlika	
x[mm]	16,67	14,04	2,63	Translacija
y[mm]	26,67	24,49	2,18	
z[mm]	232,97	232,18	0,79	
Rx[rad]	0,3230	0,3216	0,0014	Rotacija
Ry[rad]	0,6917	0,6975	0,0058	
Rz[rad]	3,0110	3,0047	0,0063	

Iz zadnje tablice vidljivo je da su razlike u kalibraciji ponovno oko dva milimetra. Pitanje je generira li se greška uvođenjem ljudskog faktora u proces kalibracije. Naravno, krivo je zaključiti da vizijski sustav ne generira nikakve greške prilikom kalibracije. Neke od grešaka kod korištenja vizijskog sustava generirane su zbog lošijeg pozicioniranja kamere u odnosu na CAD modele zbog tehnologija izrade, gdje sjecište optičkih osi nije točno. Također razne promjene okoline i uvjeti osvjetljenja mogu dovesti do loše segmentacije i detekcije vrha alata. Korištenje kompliciranih alata također otežava dobivanje zadovoljavajuće točnosti kalibracije. Međutim, sve te greške mogu biti mnogo manje negoli one dobivene uvođenjem ljudskog faktora u proces kalibracije ukoliko je on značajan. Utjecaj grešaka može se smanjiti većom količinom kalibracijskih točaka, kako bi se sa još više podataka generirao točniji rezultat.

11. ZAKLJUČAK

U ovom radu prikazan je razvoj automatske kalibracije alata robota korištenjem vizijskog sustava. Za ovu funkciju korišten je Jetson nano za programiranje te je za njega razvijeno kućište, a korištena je i biblioteka OpenCV za obradu slike. Na CSI priključke Jetson-a spojene su dvije kamere koje služe za dohvaćanje scene. Kamere su postavljene pod kutom od 90° unutar kalibracijske kutije koja je odvojena pošto u njoj robot odrađuje razna gibanja. Gibanje robota, dohvaćanje scene i izmjena podataka uvjetovani su komunikacijom. Komunikacija je napravljena putem socket-a. Socket-om je ostvarena dvosmjerna komunikacija između robota i vizijskog sustava. Tijek kalibracije alata također je definiran komunikacijom, a provodi se na način da se sa kamera dohvaća položaj alata u odnosu na sjecište optičkih osi kamera, kako bi se alat iz više orijentacija nalazio uvijek u istoj točki prostora. Krajnje se vizijski sustav koristi za poravnanje osi alata sa z osi baze robota radi kalibracije orijentacije alata. Rezultati dobiveni kalibracijom alata robota pomoću vizijskog sustava uspoređeni su sa rezultatima klasične kalibracije robotskog alata. Iz rezultata se može zaključiti da se korištenjem vizijskog sustava za kalibraciju alata robota dobiva na točnosti kalibracije zbog mogućeg uklanjanja faktora ljudske greške iz procesa kalibracije. Najveća prednost ovakvog sustava je automatizam zbog kojeg se proces dodatno pojednostavljuje dok se povećava sigurnost točnosti rezultata. Međutim, prilikom implementacije došlo je do pojave dodatnih problema korištenjem vizijskog sustava. Prilikom segmentacije alata moguće je da se površina alata lažno oboja zbog refleksije, što narušava točnu detekciju vrha alata, a time i točnost kalibracije. Vizijski sustav može biti osjetljiv na razne promjene uvjeta osvjetljenja, čime se isto može narušiti kvaliteta kalibracije. Svi ovi problemi zahtijevaju daljnje istraživanje kako bi se ovakav način kalibracije alata dodatno poboljšao, s obzirom na velike prednosti vizijskih sustava. Rezultati su pokazali nesukladnost klasične kalibracije i kalibracije alata robota vizijskim sustavom. Ova činjenica upućuje na potrebu za dodatnim istraživanjima u ovom području.

LITERATURA

- [1] <https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#intro> , 06.09.2024.
- [2] https://www.waveshare.com/wiki/IMX219-120_Camera#Overview, 06.09.2024.
- [3] <https://asset.conrad.com/media10/add/160267/c1/-/en/002619576DS00/datasheet-2619576-joy-it-rb-camera-jt-v2-120-joy-it-cmos-colour-camera-unit-compatible-with-development-kits-raspberry-pi.pdf>, 06.09.2024.
- [4] F. Šuligoj, 'Konfiguracije i kalibracije vizijskih sustava u robotici', Kolegij: Vizijski sustavi, Fakultet Strojарstva i brodogradnje, Sveučilište u Zagrebu, 2023 , 06.09.2024.
- [5] <https://developer.nvidia.com/embedded/downloads#?search=jetson%20nano>, 11.09.2024.
- [6] https://files.waveshare.com/upload/8/8e/IMX219-120_Camera_3D_Drawing.zip, 11.09.2024.
- [7] <https://www.w3schools.com/cpp/>, 17.09.2024.
- [8] <https://opencv.org/>, 17.09.2024.
- [9] https://medium.com/@onix_react/what-are-sockets-and-what-are-sockets-for-8eef56436b7b , 08.01.2025.
- [10] <https://markhedleyjones.com/projects/calibration-checkerboard-collection> , 08.01.2025.
- [11] <https://forum.universal-robots.com/t/base-coordinate-system-documentation/140> , 09.01.2025.
- [12] <https://s3-eu-west-1.amazonaws.com/ur-support-site/32554/scriptManual-3.5.4.pdf> , 09.01.2025.
- [13] J. Hallenberg: Robot Tool Center Point Calibration using Computer Vision, MSc Thesis, Linkoping University, Linkoping., Sweden, 2007. , <https://liu.diva-portal.org/smash/get/diva2:23964/FULLTEXT01.pdf>
- [14] Gordic, Z., and Ongaro, C. (2016). Calibration of Robot Tool centre point Using Camera-Based System, [10.2298/SJEE1601009G](https://doi.org/10.2298/SJEE1601009G)
- [15] Krakhmalev, O.N., Petreshin, D.I., Krakhmalev, G.N.: Methods of calibrating the orientation of the industrial robot tool: In: 2018 International Multi-Conference on Industrial Engineering and Modern Technologies, [10.1109/FarEastCon.2018.8602519](https://doi.org/10.1109/FarEastCon.2018.8602519)

PRILOZI

- I. Git-hub (https://github.com/pg220823/kalibracija_alata_robota_25)