



FE

UNIVERZA V LJUBLJANI
Fakulteta za elektrotehniko

Program

Digitalna transformacija robotiziranih tovarnih prihodnosti

DIGITOP

Rezultat D3.1:

Prototip senzoričnega sistema ToF, integriranega z robotskim
krmilnim okoljem

Projekt/RRP	RRp 3, Industrijske raziskave: Napredna sodelujoča robotika za digitalni prehod
Vrsta dokumenta	prototip, Prototip realnega sistema na sodelujočem robotu
Verzija dokumenta	1
Sodelujoči partnerji	ULFE-LR
Avtorji	Jakob Gimpelj, Marko Munih
Datum priprave	28. 10. 2024

Kazalo

1	POVZETEK	3
2	AKTIVNOST, NAMEN	3
3	ZASNOVA SISTEMA	3
4	PROTOTIP REALNEGA SISTEMA NA SODELUJOČEM ROBOTU	6
5	DIGITALNI DVOJČEK	7
6	KALIBRACIJA REALNEGA SISTEMA IN DIGITALNEGA DVOJČKA	8
7	OSNOVNI MERILNI REZULTATI	9
8	ZAKLJUČEK	13
	BIBLIOGRAFIJA	14

1 Povzetek

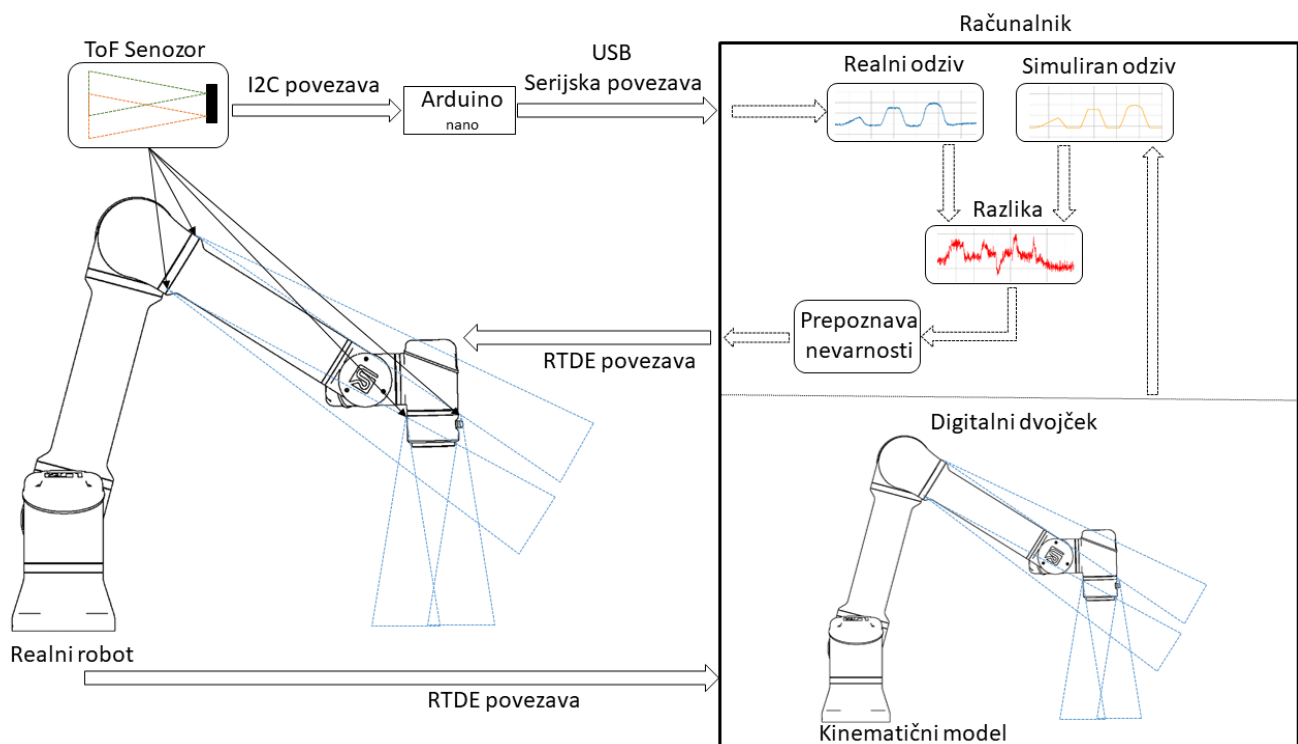
V tem poročilu predstavljamo razvoj varnostnega sistema za robotske aplikacije, ki temelji na uporabi Time of Flight (ToF) senzorjev in simulaciji z uporabo digitalnega dvojčka robota UR5e. Cilj je optimizirati zaznavanje nevarnih situacij in izboljšati odzivnost robotskega sistema v realnem času. Digitalni dvojček je bil implementiran v simulacijskem okolju Gazebo, kjer je natančno posnemal gibe realnega robota. Simulirani ToF senzorji so umeščeni na model robota na enak način kot v resnični konfiguraciji, kar omogoča realistično primerjavo med simuliranimi in realnimi podatki. Simulacija je bila osredotočena na optimizacijo senzorskih parametrov, da bi ti čim bolje posnemali realne senzorje. Preizkušali smo različne pristope obdelave simuliranih meritev, kot so izračun minimalne razdalje, povprečne vrednosti ali utežene meritve znotraj senzorjevega vidnega polja. S tem smo dosegli bistveno zmanjšanje napake med realnimi in simuliranimi meritvami, kar predstavlja dobro izhodišče za nadaljevanje dela.

2 Aktivnost, namen

A3.1 Digitalni dvojček za dinamična varnostna območja robota (ULFE-LR, LTEK) Senzorji ToF bodo uporabljeni za laboratorijsko izvedbo, ki bo omogočila natančno zaznavanje okolice robota v realnem času. Uporabljen bo digitalni dvojček sodelujočega robota, skupaj z varnostnim algoritmom, ki nadzoruje dvojčka glede na morebitne zaznane nevarnosti v okolici. Predvidena sta razvoj in implementacija algoritmov za podporo odločanju, ki lahko prepoznajo in klasificirajo dinamično okolje robota, zaznajo in obravnavajo nepredvidene ovire ter te algoritme integrirajo v nadzorni sistem za omogočanje odzivov na nove situacije.

3 Zasnova sistema

Sistem za izboljšavo varnosti sodelujočega robota uporablja ToF senzorje, nameščene na robotu. Ti merijo razdalje do objektov v robotovi okolici in meritve primerjajo s simuliranimi meritvami, dobljenimi iz digitalnega dvojčka, ki se simulira sočasno z delovanjem robota. Prikaz celotnega sistema je viden na Sliki 1.



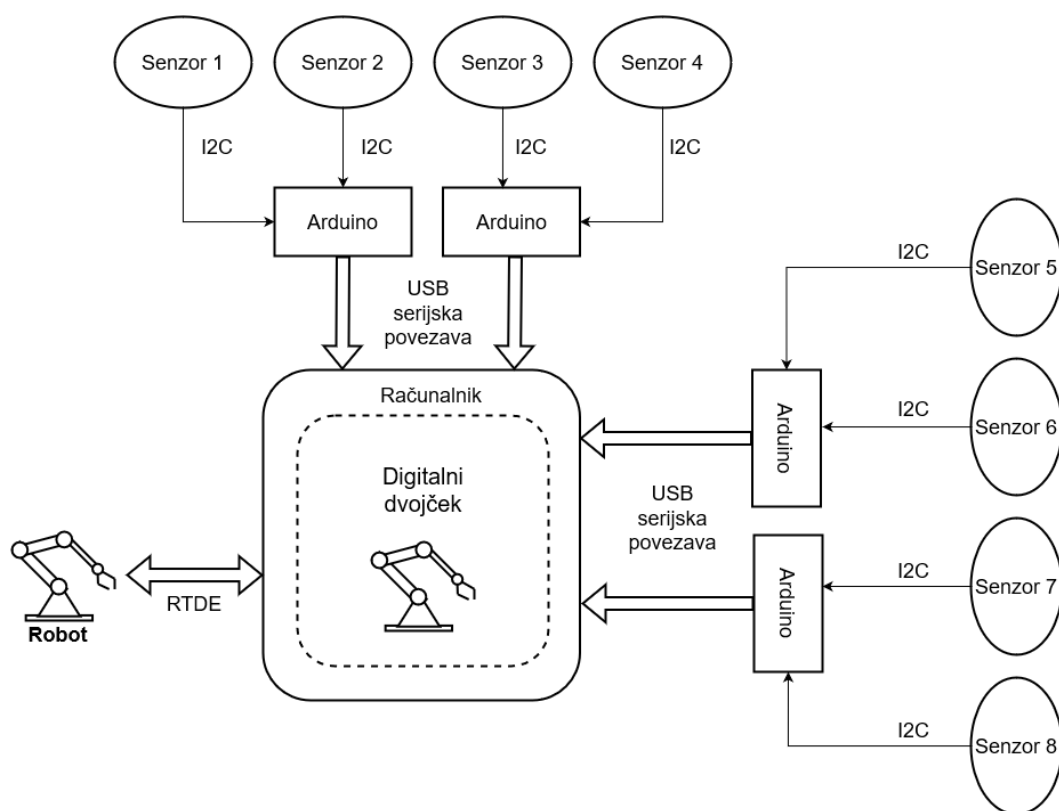
Slika 1: Prikaz toka podatkov v varnostnem sistemu. Leva stran prikazuje zajem senzorskih vrednosti, ki se preko I2C povezave posredujejo na Arduino mikroprocesor, nato pa prek USB serijske povezave na računalnik. Hkrati se sklepne vrednosti robota prenašajo na računalnik prek RTDE povezave. Desna stran prikazuje simulacijo digitalnega dvojčka robota, kjer so senzorji in robot simulirani v programu GAZEBO. Meritve simuliranih senzorjev se primerjajo z realnimi vrednostmi, pri čemer se izračuna razlika med njimi. Na podlagi te razlike se izvaja analiza za prepoznavo nevarnosti, rezultati pa se pošiljajo nazaj na robota za morebitno spremembo delovanja.

Varnostni sistem temelji na uporabi ToF (ang. *Time of Flight*) senzorjev. Vsak senzor ima infrardeč oddajnik, ki oddaja svetlobne impulze v določenem območju (zeleno območje na Sliki 1), in sprejemnika, ki je sestavljen iz več SPAD (angl. *Single Photon Avalanche Diode*) področij. Ta področja lahko zaznavajo odbite fotone, ki se vrnejo po interakciji z objektom znotraj prejemnikovega vidnega polja (oranžno območje na Sliki 1). Presek območja oddajnika in sprejemnika določa efektivno vidno polje senzorja. Glede na pretekel čas, ki ga je pulz potreboval za pot od oddajnika do objekta in nazaj do sprejemnika, število in jakost aktiviranih področij (odvisnih od moči signala) se določi razdaljo do tega predmeta. Ta princip omogoča natančne meritve tudi pri zelo kratkih razdaljah, hkrati pa deluje v različnih svetlobnih pogojih, saj uporablja valovno dolžino signala, ki je izven spektra vidne svetlobe.

Pri izbiri smo se osredotočili na družino senzorjev VL53L proizvajalca STMicroelectronics, ki so cenovno ugodni in kompaktni. Pregledali smo modele VL53L0X, VL53L1, VL53L5CX in VL53L8CX. Ti omogočajo merjenje razdalje do najbližjega objekta znotraj določenega vidnega polja. VL53L0X ima vidno polje razpona 25°, VL53L1 pa ima do 27°. Odločili smo se za VL53L1, saj je zmogljivejši od VL53L0X. Modela VL53L5CX in VL53L8CX smo zavrnil, saj se posamezni senzorji med seboj opazno motijo. Senzor VL53L1 ima nastavljivo vidno polje, ki je na sprejemniku razdeljeno na 16x16 SPAD-ov, ki se da zmanjšati na 4x4 aktivnih SPAD-ov. Teoretični domet znaša 4 m, podatke

pa vzorči z največ 60 Hz. Na sodelujoči robot smo namestili več senzorjev, ki so na Sliki 1 prikazani z modrimi trikotniki, ki predstavljajo njihova vidna polja.

Senzorji so v parih preko I2C vodila povezani na mikroprocesor Arduino Nano, kakor prikazuje shema na Sliki 2. Arduino skrbi za proženje meritev senzorjev in zbiranju podatkov, ki so nato preko USB serijskega vodila posredovani na računalnik. Vse meritve senzorjev se hkrati pošiljajo na računalnik, kjer se shranjujejo in primerjajo s simuliranimi meritvami, dobljenih iz digitalnega dvojčka (desno na Sliki 1).



Slika 2: Shema povezav ToF senzorjev, Arduino Nano mikroprocesorjev, robota in računalnika, na katerem teče digitalni dvojček. Digitalni dvojček predstavlja drugi del varnostnega sistema. To je simulacijski model robotske celice, ki jo sestavljajo virtualni tridimenzionalni model sodelujočega UR5e robota in njegove okolice ter simulirani senzorji na robotu. Program digitalnega dvojčka bere sklepne podatke realnega robota (koti in hitrosti v sklepih), kar omogoča natančno sinhronizacijo gibov virtualnega modela robota z realnim robotom v realnem času. Digitalni dvojček je realiziran v okviru programske opreme ROS (Robotski Operacijski Sistem). Trenutne vrednosti kotov in hitrosti v sklepih robota v robotskem krmilniku UR5e so posredovane kot ROS teme (ang. Topic) v program GAZEBO. Ta vključuje tudi direktni kinematični model robota. Tako kot so v realnosti nameščeni ToF senzorji na vrhu robota, so tudi v digitalnega dvojčka kinematično pripeti in transformirani v ustrezne lege na model robota.

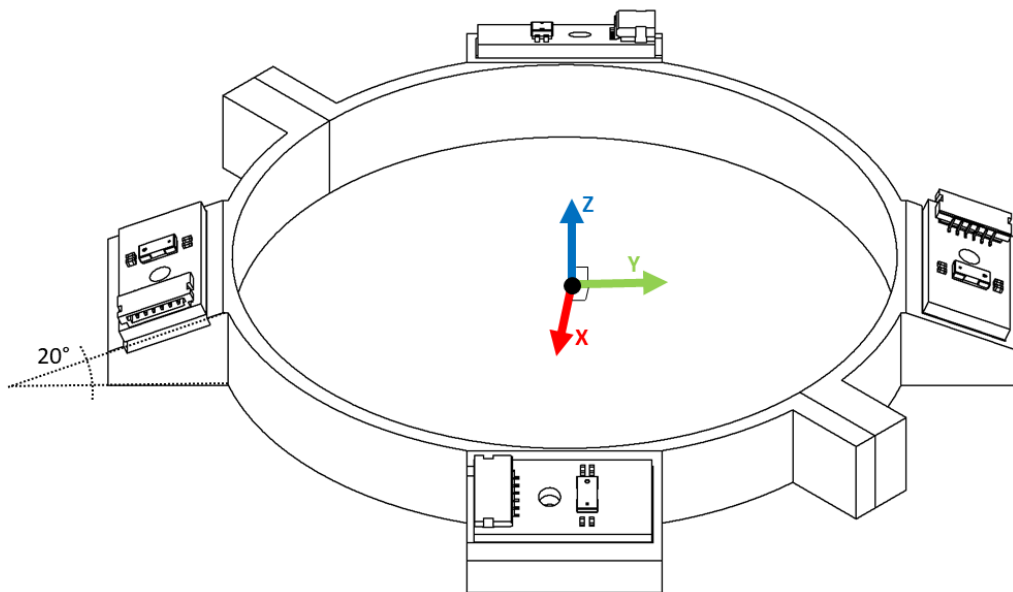
Naslednji ključni vidik v digitalnem dvojčku je pravilno modeliranje ToF senzorjev. Sevalni in detekcijski diagram resničnih senzorjev VL53L1 ni enak samo idealni središčni črti, ampak ima obliko 3D elipsoida. Odboj se torej dogaja tudi na nekoliko stranskih točkah izven idealne osi senzorja, skupna kombinacija odbojev pa prispeva k

izmerjeni vrednosti. Podobno želimo doseči v digitalnem dvojčku ToF senzorja. Najbolj osnoven digitalni dvojček senzorja je sestavljen iz več teoretično lepih linijskih žarkov. Vse izračunane razdalje do okolice za posamezne žarke znotraj enega senzorja v digitalnem dvojčku nato kombiniramo v skupno eno izračunano vrednost za ta senzor v digitalnem dvojčku.

Ko je znana izmerjena vrednost iz resničnega senzorja (moder graf, realni odziv na Sliki 1) ter tudi izračunana vrednost tega senzorja iz digitalnega dvojčka (rumen graf, simuliran odziv na Sliki 1) je s primerjavo teh dveh vrednosti (rdeč graf na Sliki 1) mogoče sklepati na morebitne ovire v delovnem prostoru, ki lahko predstavljajo tudi resno nevarnost za delovanje robota, ali za sodelujočega človeka. Taka primerjava podatkov simuliranih in realnih senzorjev omogoča zaznavo sprememb v okolici robota v realnem času, kar lahko vodi do ustreznih prilagoditev v delovanju robota.

4 Prototip realnega sistema na sodelujočem robotu

Za prvi prototip sistema smo izbrali opisane ToF senzorje tipa VL53L1. Te senzorje smo po štiri namestili na 3D natisnjen nosilni obroč, kot prikazan na Sliki 3.

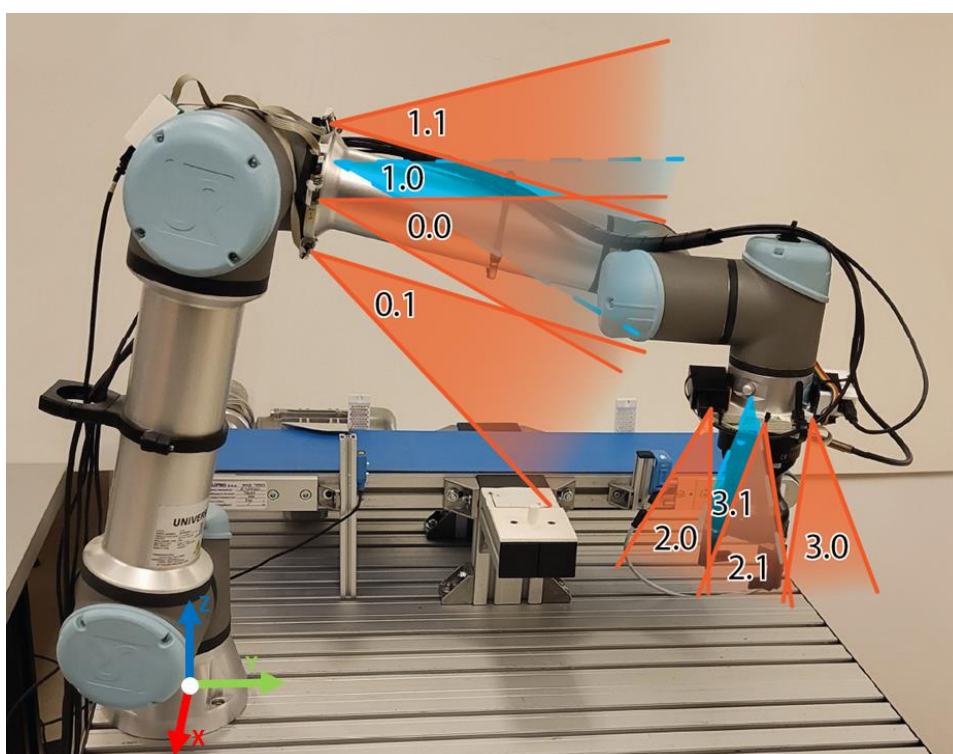


Slika 3: Slika nosilnega obroča senzorjev, ki so nameščeni nanj pod kotom 20° glede na horizontalno ravnino.

Nosilni obroč se oprime okoli segmenta robota. Senzorje smo na obroč pritrdili pod kotom 20° glede na horizontalno ravnino, vidno na Sliki 3. Tako imajo vsi senzorji vidna polja usmerjena rahlo navzven, stran od robota, da se izognemo situaciji, kjer se segment robota ves čas nahaja v vidnem polju vseh senzorjev. To bi vodilo do nezanesljivih meritev, saj bi ti zaznavali segment robota in ne drugih objektov, ki predstavljajo

morebitne ovire. Nosilni obroč je mogoče pritrditi na več različnih mestih na robotu, prav tako pa se lahko nanj namesti več obročev, s čimer zagotovimo popolno pokritost bližnje okolice robota.

Slika 4 prikazuje postavitev senzorjev po robotu in njihova vidna polja. Za komunikacijo in nastavljanje senzorjev smo pri tej izvedbi uporabili mikroprocesorje Arduino Nano, pri čemer je vsak Arduino upravljal dva senzorja. Senzorji na sliki so oštevilčeni po parih, glede na vezano na posamezen Arduino – tako sta senzor 0.0 in 0.1 vezana na en Arduino, senzorja 1.0 in 1.1 na drugega in tako naprej. Modra barva nakazuje, da je senzor zakrit na sliki, oziroma se nahaja na zadnji strani robota. Vse Arduino enote so bile preko USB vodil povezane z osebnim računalnikom, na katerem simuliramo tudi digitalnega dvojčka. Tu se nato meritve senzorjev lahko shranjujejo, prikazujejo in primerjajo s simuliranimi.

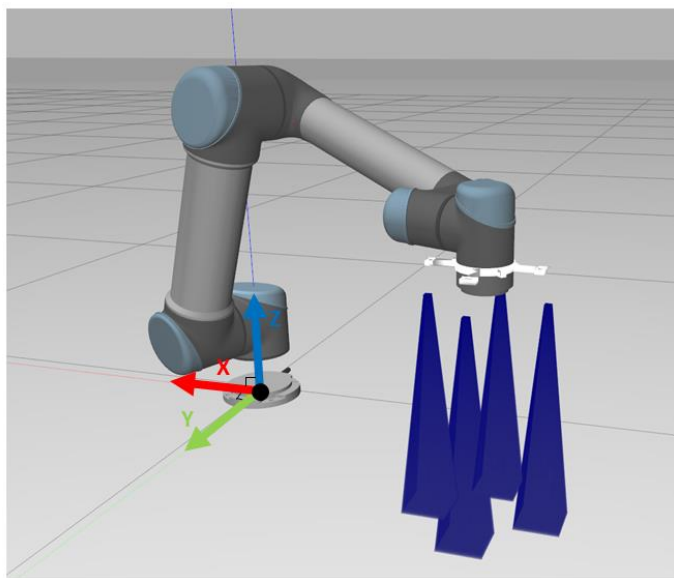


Slika 4: Postavitev senzorjev na robotu in njihova vidna polja.

5 Digitalni dvojček

Digitalni dvojček smo realizirali v odprtokodnem simulacijskem okolju GAZEBO, ki je del programskega okolja ROS. V simulacijo smo uvozili virtualni model sodelujočega robota UR5e, viden na Sliki 5. Ta je enak tistemu, ki ga uporabljamo za realna testiranja. Robot preko povezave RTDE (angl. *Real-Time Data Exchange*) pošilja podatke o sklepnih kotih in hitrostih iz krmilnika robota na računalnik. Ti podatki so zapakirani v ROS teme, ki jih simulacijsko okolje GAZEBO lahko bere. Prejete teme se nato uporabijo za pogon kinematičnega modela robota, kar omogoča, da virtualni model natančno sledi gibanju realnega robota v realnem času z minimalnim zamikom.

Na virtualnem modelu robota smo namestili tridimenzionalni model nosilnega obroča in simulirane VL53L1 senzorje, kakor je vidno na Sliki 5, pri čemer njihova pozicija in orientacija natančno posnemata realne senzorje. V simulaciji so senzorji definirani kot snopi 32 x 32 žarkov z razponom 27° (0,471 radiana), ki predstavlja senzorsko vidno polje (prikazano kot temno modre piramide na Sliki 5). Vsak žarek meri razdaljo do prvega objekta, ki prekine njegovo pot. Rezultat simulacije je kvadratna matrika velikosti 32 x 32. Vsako polje v tej matriki predstavlja razdaljo od izhodišča žarka do točke, kjer žarek zadane oviro. Senzor zaznava razdalje v območju od 0,05 m do 2 m. Meritve se kot ROS teme posredujejo za nadaljnjo obdelavo. Za omejitev računske zahtevnosti smo določili maksimalno dolžino žarka 2 metra. Simulacija ne upošteva morebitnih odbojev žarkov, ki bi sicer vplivali na natančnost meritev v realnem okolju. Prav tako ne upošteva refleksivnosti različnih materialov ali površin.



Slika 5: Virtualni model UR5e robota in vizualizacija snopa žarkov štirih senzorjev nameščenih na enem nosilnem obroču v GAZEBO.

6 Kalibracija realnega sistema in digitalnega dvojčka

Za zagotovitev ustrezne skladnosti med realnim in simuliranim okoljem je potrebna natančna kalibracija senzorjev. Realne senzorje kalibriramo z namestitvijo na znano referenčno višino nad nerefleksivno, mat površino bele ali svetlo sive barve. Ta kombinacija refleksivnosti in barve površine zagotavlja stabilen in močan odbiti signal, kar vodi do zanesljivega merjenja razdalje. Kalibracija mora biti izvedena v enakem okolju in enakimi svetlobnimi pogoji, v katerih bodo senzorji operirali med delovanjem. Vpliv sprememb svetlobnih pogojev na merilno natančnost namreč ni zanemarljiv. Na senzorjih s pomočjo mikroprocesorja Arduino nastavimo referenčno vrednost. Tako prilagodimo notranje parametre senzorjev, da pri določeni referenčni razdalji vračajo pravilno vrednost. Če so površine zelo temne, se intenziteta odbitega signala zmanjša, kar lahko privede do večjega šuma in manjše zanesljivosti meritev. Pri zelo refleksivnih površinah, kot so ogledala, se lahko

ustvarijo večkratni odboji, kar zmede senzor, ali pa se svetlobni signal popolnoma odbije stran od senzorja, kar vodi do napačnih meritev.

Pri digitalnem dvojčku kalibracija senzorjev ni potrebna, saj so ti idealni in vedno natančno izmerijo razdaljo do objekta. Ključnega pomena pa je natančna postavitev vseh objektov v virtualnem okolju. V digitalnem dvojčku je koordinatni sistem baze robota poravnan z globalnim koordinatnim sistemom simulacije, kar pomeni, da so vse pozicije ostalih objektov definirane glede na robota. To omogoča uporabo realnega robota kot merilno napravo za neposredno merjenje pozicij vseh objektov v njegovem delovnem prostoru.

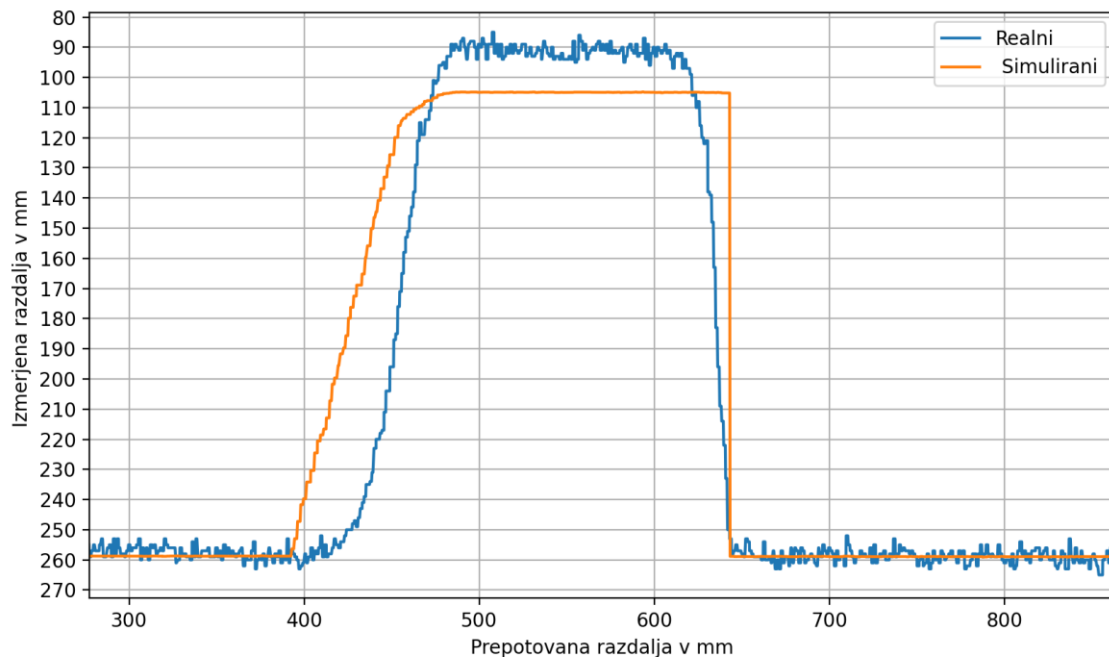
Za kalibracijo uporabimo kalibracijsko špico, ki jo namestimo na vrh robota. Z robotom nato definiramo koordinatne osi posameznih objektov, iz katerih izračunamo točno pozicijo in orientacijo objektov glede na znane dimenzije. Nato v simulacijo postavimo virtualne modele teh objektov, da natančno ustrezajo realnim položajem.

7 Osnovni merilni rezultati

Z zgoraj opisanim sistemom smo izvedli meritve, katerih cilj je bil identificirati simulacijske parametre, preko katerih bi lahko minimizirali razliko med simulacijskim in realnim izhodom senzorja. Na vrh robota smo namestili en nosilni obroč s štirimi senzorji. Z robotom smo nato opravili prelet lesene kocke z velikosti stranice 15 cm, pri čemer je bil vrh robota ves čas obrnjen navzdol, tako da so vsi štirje senzorji zaznavali kocko. Izmerjene vrednosti realnih senzorjev ter sklepne spremenljivke robota smo shranjevali v » .bag« formatu v okolju ROS, kar nam je kasneje omogočilo večkratno predvajanje gibanja robota in senzorskih vrednosti v simulaciji, brez potrebe po ponovnem gibanju realnega robota.

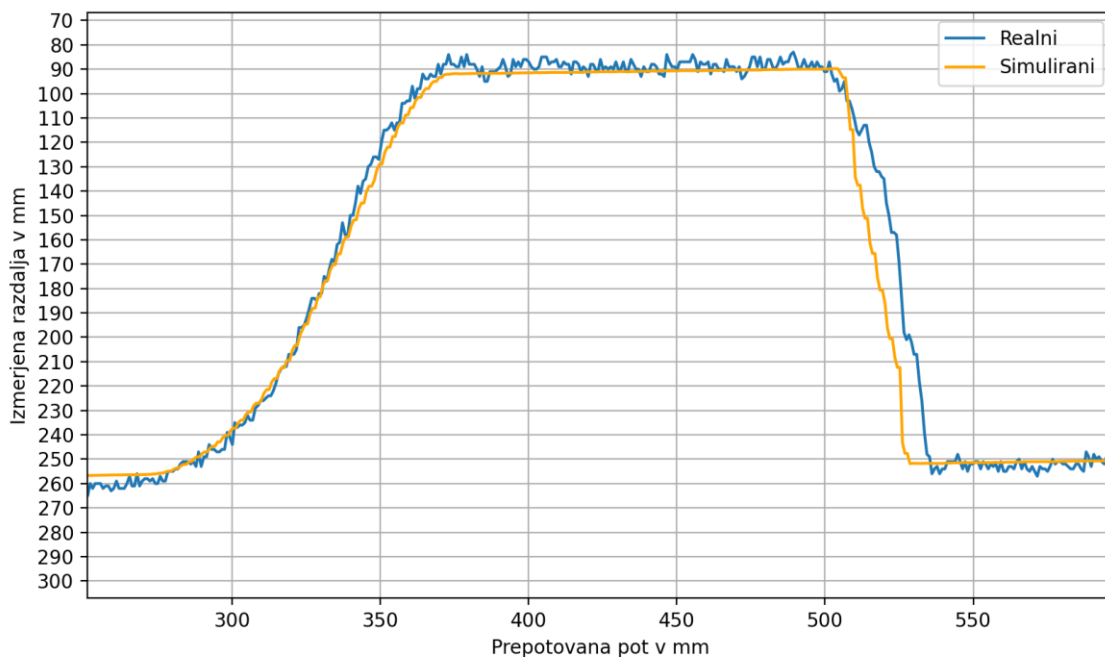
Med iteracijami simulacije smo prilagajali parametre simuliranih senzorjev, kot so velikost vidnega polja, gostota vzorčenja in kot namestitve senzorjev. Hkrati smo spreminjali tudi metode za obdelavo pridobljenih meritev (matrik velikosti 32 x 32). Testirali smo različne pristope, pri čemer smo analizirali rezultate glede na minimalno zaznano razdaljo, povprečna vrednost v vidnem polju in utežene vrednosti glede na lokacijo meritev.

Slika 6 prikazuje primerjavo meritev preleta kocke med realnim in simuliranim senzorjem z začetnimi vrednostmi simulacijskih parametrov, ki natančno posnemajo teoretične specifikacije realnega senzorja. Oranžna črta prikazuje meritve simuliranega senzorja, modra pa meritve realnega senzorja. Obe krivulji kažeta nesimetrijo, ki je posledica naklona senzorjev na nosilnem obroču. Ker slika prikazuje rezultate senzorja, usmerjenega neposredno proti kocki, je leva stran grafa v obeh primerih bolj položna. Pri simuliranem senzorju je nesimetrija še bolj izrazita, saj smo kot merilni rezultat uporabili minimalno vrednost znotraj vidnega polja. Zaradi tega je simulirani senzor zaznal kocko prej kot realni, saj jo zazna takoj, ko vstopi v vidno polje. To kaže, da realni senzor uporablja bolj kompleksen algoritem za obdelavo meritev, ne zgolj funkcijo minimuma.



Slika 6: Primerjava meritev realnih in simuliranih senzorjev pred začetkom optimizacije simulacijskih parametrov.

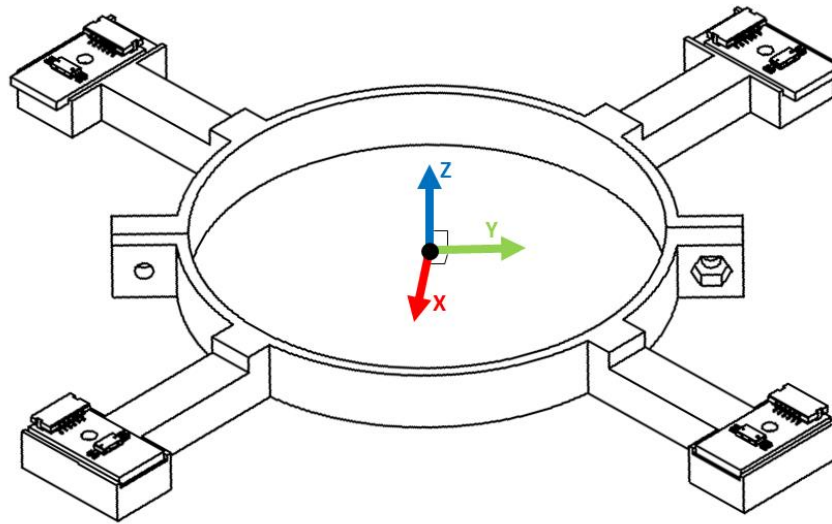
Ugotovili smo, da se realno vidno polje senzorja ne ujema v celoti s teoretičnim modelom. V nadaljnjih iteracijah smo s spreminjanjem širine vidnega polja simuliranega senzorja ter z uporabo drugih algoritmov za obdelavo merilnih podatkov, kot sta povprečenje in uteževanje, poskušali izboljšati ujemanje med krivuljami simuliranega in realnega senzorja. Z inkrementalnim ožanjem vidnega polja simuliranega senzorja smo želeli doseči boljše ujemanje s širino grafa realnega senzorja, pri čemer je sama oblika krivulje ostala nespremenjena. Za spremembo naklona oziroma same oblike krivulje smo uporabili algoritma povprečenja in uteževanja. Rezultati teh prilagoditev so prikazani na Sliki 7.



Slika 7: Primerjava meritev z uporabo algoritmov povprečenja in uteževanja, z zoženim vidnim poljem simuliranega senzorja.

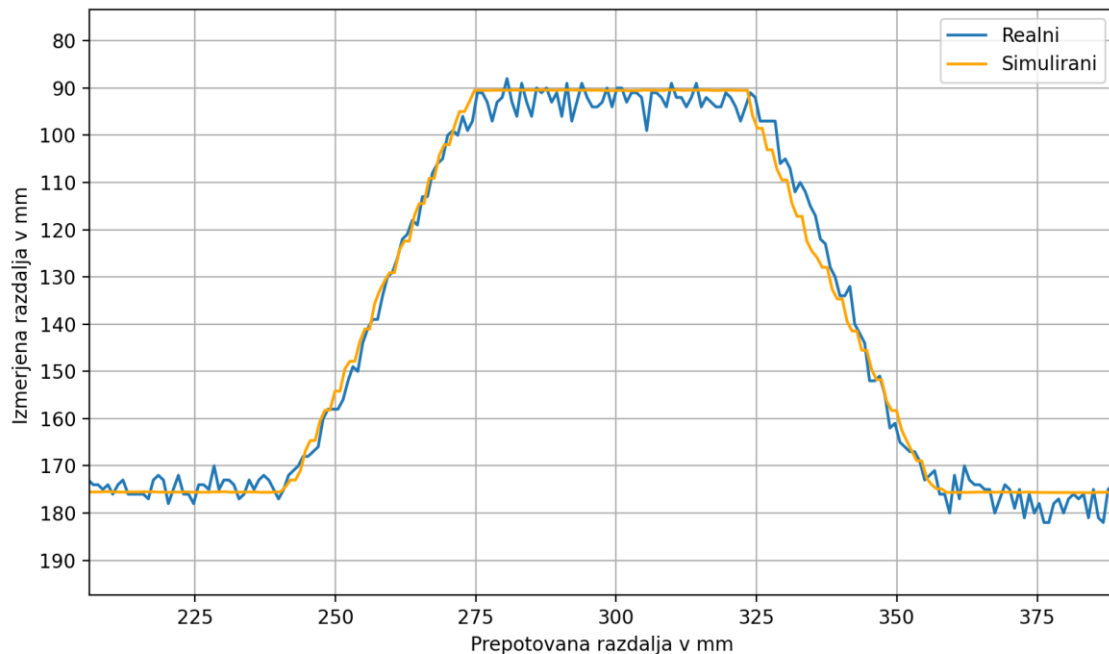
V naslednji izboljšavi smo spremenili efektivno vidno polje realnega senzorja tako, da smo omejili število aktivnih sprejemnih polj na senzorju s 16x16 na 4x4. Naklon krivulje na obeh straneh se je rahlo povečal, kar je prineslo le manjšo izboljšavo ujemanja. Sklepamo, da je razlog za slabše ujemanje krivulj na desnem naklonu oster vpadni kot oddanega signala na navpično stranico kocke, kar povzroča različne odboje svetlobnega signala, ki jih simulacija ne upošteva.

Zato smo se odločili razviti nov nosilni obroč, pri katerem so senzorji nameščeni navpično navzdol namesto pod kotom. Da bi zagotovili, da se robot ne nahaja v vidnem polju senzorjev, smo namestitvene točke senzorjev odmaknili dlje od središča obroča, kot je vidno na Sliki 8.



Slika 8: Slika posodobljenega nosilnega obroča senzorjev, nameščenih pravokotno na horizontalno os in z večjim odmikom od središčne osi segmenta robota.

Meritve, pridobljene s to različico nosilnega obroča, so pokazale najboljše ujemanje med simuliranimi in realnimi vrednostmi ter najmanjšo napako. V tej konfiguraciji je odziv senzorja na kocko simetričen, vidno na Slika 9. To je bistveno poenostavilo iskanje optimalnih simulacijskih parametrov. Vidno je, da se sedaj realne in simulirane vrednosti dobro ujemajo, kar je najbolj izrazito pri naklonih.



Slika 9: Primerjava merilnih rezultatov z uporabo novega nosilnega obroča, kjer so senzorji obrnjeni navpično navzdol. Velikost kocke, ki jo pri meritvi preletimo ima tokrat stranico dolgo 8 cm.

8 Zaključek

Varnostni sistem uporablja ToF senzorje nameščene na sodelujočem robotu UR5e in digitalni dvojček v simulacijskem okolju Gazebo. Sistem primerja meritve realnih in simuliranih ToF senzorjev v realnem času. Ključna sta natančna kalibracija realnih senzorjev in ustrezna postavitve objektov v virtualnem okolju. S spreminjanjem parametrov simulacije, kot so širina vidnega polja in načini obdelave simuliranih vrednosti, ter prilagoditvijo namestitve realnih senzorjev so se zmanjšala odstopanja med simulacijskimi in realnimi meritvami. Trenutna izvedba predstavlja dobro izhodišče za nadaljnjo dograjevanje sistema.

Bibliografija

- [1] P. Fürsattel, S. Placht, M. Balda, C. Schaller, H. Hofmann, A. Maier in C. Riess, „A Comparative Error Analysis of Current Time-of-Flight Sensors,“ *IEEE Transactions on Computational Imaging* , Izv. 2, št. 1, pp. 27 - 41, 2016.
- [2] S. Pasinetti, C. Nuzzi, M. Lancini, G. Sansoni, F. Docchio in A. Fornaser, „Development and Characterization of a Safety System for Robotic Cells Based on Multiple Time of Flight (TOF) Cameras and Point Cloud Analysis,“ v *2018 Workshop on Metrology for Industry 4.0 and IoT*, Brescia, 2018.
- [3] S. Tsuji in T. Kohama, „Proximity Skin Sensor Using Time-of-Flight Sensor for Human Collaborative Robot,“ *IEEE Sensors Journal*, Izv. 19, št. 14, pp. 5859-5864, 2019.
- [4] Y. Ding, F. Wilhelm, L. Faulhammer in U. Thomas, „With Proximity Servoing towards Safe Human-Robot-Interaction,“ v *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, 2019.
- [5] O. A. Adamides, A. S. Modur, S. Kumar in F. Sahin, „A Time-of-Flight On-Robot Proximity Sensing System to Achieve Human Detection for Collaborative Robots,“ v *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, Vancouver, 2019.
- [6] U. B. Himmelsbach, T. M. Wendt, N. Hangst in P. Gawron, „Single Pixel Time-of-Flight Sensors for Object Detection and Self-Detection in Three-Sectional Single-Arm Robot Manipulators,“ v *2019 Third IEEE International Conference on Robotic Computing (IRC)*, Naples, 2019.
- [7] F. Giovinazzo, . F. Grella, . M. Sartore, . M. Adami, . R. Galletti in G. Cannata, „From CySkin to ProxySKIN: Design, Implementation and Testing of a Multi-Modal Robotic Skin for Human–Robot Interaction,“ *Sensors*, Izv. 24, št. 4, p. 1334, 2024.