

ELEKTRO IN RAČUNALNIŠKA ŠOLA VELENJE
TRG MLADOSTI 3, 3320 VELENJE
MLADI RAZISKOVALCI ZA RAZVOJ SAŠA REGIJE

RAZISKOVALNA NALOGA

**VZPOSTAVITEV INTELIGENTNE POVEZAVE MED
RAČUNALNIŠKO VIZIJO IN GIBANJEM ROBOTSKÉ ROKE**

Tematsko področje: elektrotehnika, elektronika in robotika

AVTORJA:

Blaž Kristan, 2. TRA

Tim Rednjak, 2. TRA

MENTORJA:

Uroš Remenih, inž.

Samo Železnik, inž.

VELENJE, 2024

Raziskovalna naloga je bila opravljena na Elektro in računalniški šoli Velenje

Mentorja: Uroš Remenih, inž. inf.,

Samo Železnik, inž. inf.

Datum predstavitve: marec 2024

KLJUČNA DOKUMENTACIJA INFORMACIJA

ŠD SCV elektro in računalniška šola, šolsko leto 2023/2024

KG Robotika/Programiranje/Natančnost

AV Kristan Blaž

SA Rednjak Tim

KZ 3320 Velenje, SLO, Trg mladosti 3

ZA ŠCV elektro in računalniška šola

LI 2024

IN Vzpostavitev inteligentne povezave med računalniško vizijo in gibanjem robotske roke

TD Raziskovalna naloga

OP

IJ SL

JI Sl / En

AI Raziskovalna naloga "Inteligentna povezava računalniške vizije in gibanja robotske roke" raziskuje uporabo naprednih algoritmov za avtomatizacijo gibanja robotske roke. Namen raziskave je izboljšati natančnost in učinkovitost kirurških posegov s pomočjo robotske tehnologije. Rezultati kažejo na potencialno uporabo robotskega sistema kot kirurškega asistenta, ki bi lahko pomagal kirurgom pri izvajanju natančnih in kompleksnih operativnih posegov. Raziskava prinaša nove možnosti za izboljšanje varnosti in natančnosti kirurških postopkov ter napredek v razvoju medicinskih robotov. Pomembne ključne besede: računalniška vizija, gibanje robotske roke, kirurški asistent, natančnost operativnih posegov, medicinski roboti.

KEYWORD INFORMATION

ND SCV Electro and Computer School, academic year 2023/2024

CX Robotics/Programming/Precision

AU Kristan Blaž

AA Rednjak Tim

PP 3320 Velenje, SLO, Trg mladosti 3

PB SCV Electro and Computer School

PY 2024

TI Establishment of an intelligent connection between computer vision and robotic arm movement

DTResearch Project

NO

LA SL

AL SI / En

AB The research project "Intelligent Connection of Computer Vision and Robotic Arm Movement" explores the use of advanced algorithms for automating the movement of a robotic arm. The purpose of the research is to improve the accuracy and efficiency of surgical procedures through robotic technology. The results indicate the potential use of a robotic system as a surgical assistant, which could assist surgeons in performing precise and complex operative procedures. The research brings new opportunities to improve the safety and accuracy of surgical procedures and advances in the development of medical robots. Important keywords: computer vision, robotic arm movement, surgical assistant, accuracy of operative procedures, medical robots.

KAZALO VSEBINE

1. UVOD	1
1.1 PROBLEM	1
1.2 HIPOTEZE	1
1.2.1 Hipoteza 1	1
2. PREGLED OBJAV	2
2.1 PROGRAMSKA OPREMA	2
2.1.1 PYTHON	2
2.1.2 C++	2
2.2 STROJNA OPREMA	3
2.2.1 ARDUINO	3
2.3. SPLETNI ČLANKI IN NALOGE	4
2.3.1 THE MAKING OF A 3D-PRINTED, CABLE-DRIVEN, SINGLE-MODEL, LIGHTWEIGHT HUMANOID ROBOTIC HAND	4
3. SERIJSKA KOMUNIKACIJA	4
3.1 SERIAL COMUNICATION	4
3.2 MATERIAL IN METODE DELA	5
3.3 POTEK DELA	7
3.4 PROGRAMSKI DEL	9
3.4.2 POTEK PRIPRAVE VZPOZTAVITVE KOMUNKACIJE MED PYTHONOM TER ARDUINOTOM	9
4. REZULTATI	15
4.1 ODZIV ROKE	15

5. RAZPRAVA	16
6. POVZETEK	18
7. ZAKLJUČEK	18
8. ZAHVALA	18
9. PRILOGE	19
10. VIRI IN LITERATURA	21

KAZALO SLIK

SLIKA 1 PRIKAZ LOGOTIPA OGRODJA PYTHON.....	2
SLIKA 2 PRIKAZ LOGOTIPA PLATFORME ARDUINO.....	3
SLIKA 3 PRIKAZUJE STOJALO ZA SERVO MOTORJE(VIR: LASTEN).....	7
SLIKA 4 PRIKAZUJE NAMEŠČENE MOTORČKE V NOTRANJI DEL ROKE. (VIR: LASTEN).....	8
SLIKA 5 SERVO MOTORJI POVEZANI NA SVOJE MESTO.....	8
SLIKA 6 PRIKAZUJE ZAZNAVANJE ROKE Z KODO (VIR: LASTEN).....	11
SLIKA 7 PRIKAZ IZPISA ODPRTIH ALI ZAPRTIH PRSTOV.....	11
SLIKA 8 PRIKAZUJE POVEZAVO ARDUINA Z SESTAVLJENO ROKO.....	15

1. UVOD

Računalniška vizija in robotika sta se v zadnjih desetletjih izjemno razvili, kar je omogočilo napredke na številnih področjih, vključno s kirurgijo. Vendar pa je ključna izzivna prepreka pri razvoju robotov v medicini ohranjanje natančnosti in prilagodljivosti pri kirurških posegih. Raziskovalna naloga se osredotoča na vzpostavitev inteligentne povezave med računalniško vizijo in gibanjem robotske roke z namenom izboljšanja kirurških postopkov.

1.1 PROBLEM

Eden od ključnih problemov, ki bi jih lahko reševala in izboljševala robotska roka, je delovanje kirurgov. S sposobnostjo izvajanja natančnih in stabilnih gibov bi lahko robotska roka podpirala kirurge pri operacijah, kjer je potrebna visoka stopnja natančnosti in nadzora.

Poleg tega bi lahko robotska roka koristila tudi pri delu na daljavo, kjer operater lahko upravlja roko iz oddaljene lokacije. To bi omogočilo izvajanje nalog in opravil v težko dostopnih ali nevarnih okoljih, kjer je prisotnost človeka omejena ali tvegana.

Druga potencialna uporaba je pomoč ljudem brez rok ali s fizičnimi ovirami pri vsakodnevnih opravilih. S prilagodljivostjo in raznolikimi funkcijami bi lahko robotska roka izboljšala kakovost življenja in neodvisnost posameznikov, ki se soočajo s takšnimi izzivi.

Navedeni primeri predstavljajo le nekatere možnosti, kako bi lahko robotska roka prispevala k reševanju različnih problemov in izboljšanju kakovosti življenja ljudi. Z nadaljnjim razvojem in prilagoditvami bi lahko postala ključni pripomoček v različnih panogah in situacijah.

1.2 HIPOTEZE

1.2.1 Hipoteza 1

Razvoj in implementacija inteligentne povezave med računalniško vizijo in gibanjem robotske roke bo omogočila zmanjšanje nenatančnosti pri kirurških posegih ter izboljšanje rezultatov in varnosti operativnih postopkov.

1.2.2 Hipoteza 2

Ali upravljanje robotske roke na daljavo omogoča učinkovito izvajanje nalog v težko dostopnih ali nevarnih okoljih ter prispeva k večji varnosti in učinkovitosti dela na daljavo?

2. PREGLED OBJAV

2.1 PROGRAMSKA OPREMA

2.1.1 PYTHON

Python je programski jezik, ki je interpretiran in objektno usmerjen. S svojimi vgrajenimi podatkovnimi strukturami na visoki ravni, dinamičnim tipiziranjem in dinamičnim vezanjem predstavlja učinkovit jezik za hitri razvoj aplikacij ter za uporabo kot skriptni ali povezovalni jezik pri integraciji obstoječih komponent.



Slika 1 Prikaz logotipa ogrodja Python

2.1.2 C++

C++ je programski jezik, ki je objektno usmerjen in se uporablja za razvoj različnih programskih aplikacij, vključno z vgrajenimi sistemi, operacijskimi sistemi, grafičnimi uporabniškimi vmesniki in igrami. Zaradi svoje učinkovitosti, zmogljivosti in razširjene uporabe se C++ pogosto uporablja pri razvoju aplikacij, kjer je pomembna hitrost izvajanja in učinkovita uporaba sistemskih virov. Kot večnamenski jezik omogoča tako visokonivojsko kot nizkonivojsko programiranje ter integracijo s sistemskimi knjižnicami in zunanjimi orodji. C++ je priljubljena izbira med programerji, ki potrebujejo zmogljiv jezik za razvoj kompleksnih in zahtevnih aplikacij.

2.2 STROJNA OPREMA

2.2.1 ARDUINO

Arduino je odprtokodna platforma za razvoj elektronskih naprav, ki jo sestavljajo strojna oprema (mikrokrmilnik) in programska oprema (IDE). Z uporabo preprostega in intuitivnega programskega jezika, ki je podoben C/C++ omogoča enostavno programiranje mikrokrmilnika za različne aplikacije. Arduino je priljubljena izbira med ustvarjalci, hobi navdušenci in profesionalci zaradi svoje preprostosti, prilagodljivosti in razširjene skupnosti uporabnikov.



Slika 2 Prikaz logotipa platforme Arduino

2.2.2 KAMERA

Kamera je naprava za zajemanje slik ali video posnetkov. Z uporabo optičnih in digitalnih komponent omogoča pretvorbo svetlobnih signalov v digitalne podatke, ki jih lahko shranimo, obdelamo ali predvajamo na različnih napravah. Kamere se uporabljajo v številnih aplikacijah, vključno s fotografijo, videonadzorom, medicinsko diagnostiko, avtomobilsko industrijo, znanstvenimi raziskavami in zabavno elektroniko. Obstajajo različne vrste kamer, vključno s digitalnimi fotoaparati, videokamerami, termalnimi kamerami, nadzornimi kamerami, mikroskopskimi kamerami itd. Kamere so postale nepogrešljiv del sodobne tehnologije in omogočajo zajemanje in deljenje vizualnih informacij na širokem spektru področij.

2.2.3 SERVO MOTOR SG90

Servo motor SG90 je majhen elektromehanski motor, ki ga odlikujejo visoka natančnost in hitrost ter enostavna integracija. Namenjen je za natančno krmiljenje položaja vrtenja na osi v določenem območju. Glavne značilnosti servo motorja SG90 vključujejo majhno velikost, lahko konstrukcijo, visoko zanesljivost in nizko ceno. Primeren je za številne aplikacije, kot so modelarstvo, robotika, avtomatizacija, radijsko vodenje in druge podobne aplikacije, kjer je potrebno natančno in stabilno krmiljenje gibanja. Servo motor SG90 je pogosta izbira med ustvarjalci in hobisti zaradi svoje preproste uporabe in učinkovitosti.

2.3. SPLETNI ČLANKI IN NALOGE

2.3.1 THE MAKING OF A 3D-PRINTED, CABLE-DRIVEN, SINGLE-MODEL, LIGHTWEIGHT HUMANOID ROBOTIC HAND

V članku na spletni strani <https://www.frontiersin.org/articles/10.3389/frobt.2017.00065/full>

Kjer so avtorji Li Tian, Nadia Magnenat Thalman, Daniel Thalmann, Jianmin Zheng. V svojem delu so avtorji proučevali, kako deluje človeška roka, ter nato poskušali reproducirati pomembne lastnosti človeške roke na svoji robotski roki. Roka je bila oblikovana s preprostim in praktičnim dizajnom, ki je uspešno simuliral večino pomembnih gest človeške roke v njihovem eksperimentu z oprijemanjem.

Ključne značilnosti njihovega dela vključujejo:

Uporaba mehanizma krogličnega sklepa za povečanje prožnosti in spretnosti, zlasti v delu palca. Roka je sestavljena iz sedmih 3D-natisljivih delov in uporablja mehanizme kablov za akcijo, kar omogoča poenostavitev montažnega procesa. Primerjava različnih gibljivih kotov sklepov s človeškimi in drugimi robotskimi rokami. Akcija in nadzor prstov roke poteka s pomočjo mehanizmov, ki jih poganjajo kabli, pri čemer ima vsak prst tri gibljive sklepe. Rezultati njihovega dela so pokazali, da lahko njihova robotska roka obvladuje večino pomembnih oblik oprijema. Ta raziskava lahko prispeva k izboljšanju celotne zmogljivosti humanoidnih robotov.

3. SERIJSKA KOMUNIKACIJA

3.1 SERIAL COMMUNICATION

Primerjava med serijsko komunikacijo in drugimi tehnologijami, kot sta RS485 ali RS422 ter brezžične tehnologije, kot sta Bluetooth in WiFi, razkriva različne značilnosti, ki vplivajo na izbiro glede na specifične zahteve aplikacij.

Serijska komunikacija omogoča prenos podatkov med napravami preko fizičnega serijskega vmesnika. Njena prednost je v enostavni implementaciji in zanesljivosti, zlasti pri povezovanju naprav na kratkih do srednjih razdaljah. Ta tehnologija je primerna za številne aplikacije, kot so industrijska avtomatizacija, pametne naprave in avtomobilski sistemi. Ključna prednost serijske komunikacije je njena robustnost in stabilnost na različnih področjih uporabe.

V primerjavi s podobnimi žičnimi tehnologijami, kot sta RS485 ali RS422, serijska komunikacija ponuja enostavnejšo implementacijo in ni odvisna od kompleksnih omrežnih konfiguracij. Medtem ko RS485 in RS422 ponujata večjo zmogljivost prenosa podatkov in boljšo odpornost proti motnjam, je njihova uporaba pogosto povezana z večjimi stroški in zahtevnejšo konfiguracijo.

Brezžične tehnologije, kot sta Bluetooth in WiFi, omogočajo tudi prenos podatkov med napravami, vendar imajo svoje posebnosti. Bluetooth je bolj primeren za povezovanje naprav na kratkih razdaljah, medtem ko WiFi omogoča prenos podatkov na večje razdalje, vendar z večjo porabo energije. Te tehnologije ponujajo večjo fleksibilnost in mobilnost, vendar imajo lahko omejitve v zanesljivosti in varnosti ter zahtevajo ustrezno konfiguracijo omrežja.

Izbira med serijsko komunikacijo, RS485 ali RS422 ter brezžičnimi tehnologijami, kot sta Bluetooth in WiFi, je odvisna od specifičnih zahtev aplikacije, kot so razdalja prenosa podatkov, odpornost proti motnjam, poraba energije in prilagodljivost omrežja. Vsaka tehnologija ima svoje prednosti in omejitve, zato je ključno upoštevati specifične potrebe in zahteve uporabnika pri izbiri najboljše rešitve.

3.2 MATERIAL IN METODE DELA

3.2.1 STROJNA OPREMA

Za izdelavo modela robotske roke smo uporabili predlogo inmoov.fr, enega izmed najbolj priljubljenih odprtih virov za DIY (naredi sam) projekte na področju robotike. Ta vir smo izbrali zaradi njegove priljubljenosti med ustvarjalci ter razpoložljivosti podrobnih navodil in modelov za izdelavo. Pri nadaljnjem razvoju in prilagajanju smo se osredotočili na specifične potrebe našega projekta, ob upoštevanju možnosti prilagoditve, ki jih ponuja ta platforma

3.2.1.1 IZBIRA MATERIALA

Ko sva razmišljala o materialu za najino raziskovalno nalogo, sva se odločila za uporabo filameta. Videla sva, da imajo ključno vlogo v sodobnih tehnoloških aplikacijah, predvsem v področju 3D-tiskanja. Filamenti so tisti gradniki, ki omogočajo, da lahko iz digitalnih modelov ustvarimo resnične tridimenzionalne objekte. Fascinirala naju je tudi njihova raznolikost - lahko najdemo filamente iz različnih vrst plastike, kovinskih in kompozitnih materialov. To odpira res širok spekter možnosti za raziskave in razvoj novih izdelkov ter inovativnih rešitev. Zato sva se odločila natančno proučiti lastnosti filamentov ter razumeti, kako vplivajo na končni izdelek. Najin cilj je prispevati k razvoju naprednih tehnoloških aplikacij, ki bodo lahko izboljšale naše vsakdanje življenje.

3.2.1.2 IZBIRA MOTORČKOV

Servo motor SG90 je majhen in cenovno ugoden servo motor, ki se pogosto uporablja v različnih elektronskih projektih in napravah. Gre za vrsto motorja, ki se lahko natančno kontrolira glede na določene vhodne signale. SG90 je znan po svoji zanesljivosti, majhnosti in nizki ceni, kar ga naredi priljubljeno izbiro za hobiste, študente in ustvarjalce. Uporablja se lahko za premikanje majhnih mehanizmov, kot so ročaji,

krmilni elementi ali celo manjše pomične kamere. Ključne značilnosti SG90 vključujejo lahko težo, visoko natančnost pozicioniranja in enostavno upravljanje preko digitalnih signalov, kot so PWM signali.

3.2.1.3 IZBIRA VRVI

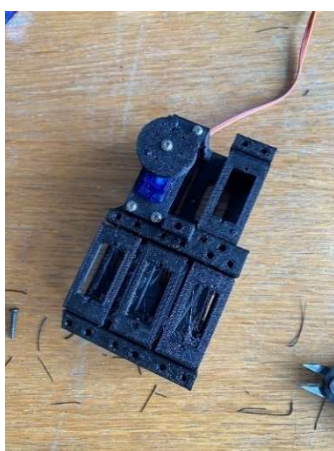
Uporaba laka kot vrvi ima številne prednosti zaradi njegove moči, lahкости in vzdržljivosti. Laks je prilagodljiv material, ki se enostavno zaveže in oblikuje glede na potrebe, kar ga naredi primerne za različne namene, kot so vzpenjanje, vrvne vaje, vleka in priprava bivališč. Poleg tega je laks vodoodporen in odporen na različne vremenske pogoje, kar ga naredi primerne za uporabo tako na prostem kot v notranjih prostorih. Zaradi svoje razpoložljivosti in relativne nizke cene je laks pogosto izbrana možnost za številne aplikacije, kjer je potrebna zanesljiva in trajna vrv.

3.3 POTEK DELA

Iz internetne strani inmoov.fr sva vzela že narisane modele robotske roke, ki sva jih potem naložila na USB ključek in jih shranila po imenu. Potem sva datoteke prekopirala na s SD kartico, ki sva jih uporabila pri 3D printerju.

Modele sva pregledala, če ustrezajo meram in jih po potrebi tudi spremenila.

Ker nisva imela pravilno velikih in močnih motorčkov, sva svoj model prilagodila z novo skico. Novo skico sva narisala večkrat, kajti prišla sva do problema, da se ni ujemala s staro skico.



Slika 3 Prikazuje stojalo za servo motorje

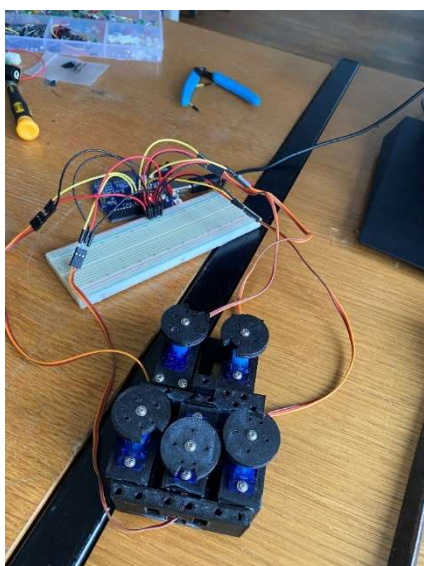
Po rešitvi tega problema sva nadaljevala s tiskanjem modelov. Sproti sva mu dele tudi prilagajala in lepila skupaj. Ko so bili vsi modeli narisanih, sva začela z prilagajanjem motorčkov v notranjost roke.

Motorčke sva povezala na arduino, ker sva naletela na nov problem in to je bilo napajanje motorčkov. Problem sva rešila, tako da napajanje ni prihajalo iz računalnika, ampak je prihajalo iz vtičnice. Nato sva motorček e uspešno postavila na svoje mesto.



Slika 4 Prikazuje nameščene motorčke v notranji del roke

Ko so bili motorčki uspešno postavljeni na svoja mesta sva jih povezala še z napajanjem.



Slika 5 Servo motorjji povezani na svoje mesto

Po uspešni povezavi motorčku in arduino. Ta sva to vse povezala še s kamero, ki zaznava premike človeške roke in potem informacije pošlje naprej na arduino, kjer motorčki odreagirajo in odprejo oziroma zaprejo prste.

3.4 PROGRAMSKI DEL

3.4.1 IZBIRA PROGRAMSKEGA OKOLJA

3.4.1.1 PYCHARM

PyCharm je integrirano razvojno okolje (IDE) za programski jezik Python, ki ga razvija podjetje JetBrains. Ponuja širok nabor funkcij in orodij, namenjenih olajšanju razvoja Python aplikacij. Značilnosti PyCharma vključujejo napredno urejevalno okolje z možnostjo samodejnega dokončanja kode, integrirano odpravljanje napak, podporo za upravljanje projektov, integracijo s sistemom za nadzor različic, možnost pisanja in izvajanja enotnih testov ter uporabo različnih vtičnikov za razširjanje funkcionalnosti. Zaradi svoje uporabnosti in učinkovitosti je PyCharm priljubljena izbira med programerji Pythona pri razvoju aplikacij v tem programskem jeziku.

3.4.1.2 ARDUINO

Arduino IDE (Integrated Development Environment) je programska oprema, ki jo je razvilo podjetje Arduino in je namenjena programiranju mikrokontrolerov Arduino. To je posebno okolje za razvoj, ki omogoča enostavno pisanje, urejanje, nalaganje in izvajanje programske kode za Arduino mikrokontrolerje. Arduino IDE vključuje vse potrebne komponente in orodja, ki jih potrebujete za začetek programiranja, vključno s sintaksnim poudarjanjem, avtomatskim zaključevanjem kode, upravljanjem knjižnic, vmesnikom za nalaganje kode na mikrokontroler in orodji za spremljanje serijskega monitorja. S tem omogoča programerjem, da enostavno razvijajo različne projekte in aplikacije za Arduino platformo.

3.4.2 POTEK PRIPRAVE VZPOSTAVITVE KOMUNKACIJE MED PYTHONOM TER ARDUINOTOM

3.4.2.1 PRIPRAVA ZAZNAVANJE ROKE

Začela sma delati v Pycharm edu v pythonu

Ka prikazovanje kamere sma uporabila

```
import cv2  
  
cap = cv2.VideoCapture(0)
```

Za zaznavanje roke sem uporabil


```
from cvzone.HandTrackingModule import HandDetector

import cv2

detector = HandDetector(maxHands=1, detectionCon=0.7)
```

Potem sva naredila zanko kjer odpre kamero in zaznava roko

Ter da zaznava prste ali so dvignjeni ali ne

Nato jih sprinta

```
while True:
    success, img = cap.read()

    # Detect hands
    hands, img = detector.findHands(img)

    if hands:
        for hand in hands:
            fingers = detector.fingersUp(hand)
            print(fingers)
```

nato sva pripravila serial port

dodala sva

```
import serial

ser = serial.Serial("COM3", 9600, timeout=0.5)
```

in nadaljevala s pretvorbo v signal string

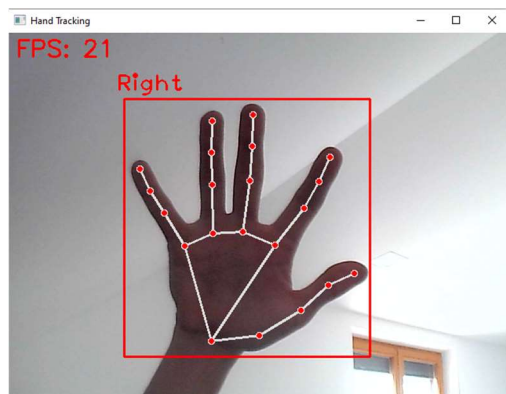
nato sva uporabila ser.write ta pošilja signal na arduinota

```
signal_str = ".join(map(str, fingers))
signal_str = f"${signal_str}"

# Send signal to Arduino
try:
```

```
ser.write(signal_str.encode())
except serial.SerialException as e:
    print(f"Error writing to serial: {e}")
```

zgledalo je tako



Slika 6 Prikazuje zaznavanje roke z kodo

```
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
[1, 1, 1, 1, 1]
```

Slika 7 Prikaz Izpisa odprtih ali zaprtih prstov

po želji sva dodala še fps count

(frames per second, slike na sekundo)

```
import time

curr_time = time.time()
fps = 1 / (curr_time - prev_time)
prev_time = curr_time
```

```
cv2.putText(img, f'FPS: {int(fps)}', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, text_color,
2)
```

ter za izhod iz programa

```
cv2.imshow("Hand Tracking", img)
if cv2.waitKey(1) & 0xFF == 27:
    break

ser.close()
cap.release()
cv2.destroyAllWindows()
```

3.4.2.2 PRIPRAVA ZA KODE NA ARDUINOTU

Najprej sem includal servo.h za delovanje pri pošiljanju signala

```
#include <Servo.h>
```

Definiral sem koliko znakov mora prejemati

```
#define numOfValsRec 5
```

```
#define digitsPerValRec 1
```

```
Servo servos[numOfValsRec];
```

```
int valsRec[numOfValsRec];
```

Tukaj se spremeni dolžina znakov saj se doda znak \$ v izpis

```
int stringLength = numOfValsRec * digitsPerValRec + 1;
```

```
int counter = 0;
```

```
bool counterStart = false;
```

```
String receivedString;
```

Tukaj se serial začne ter nastavi vrednosti na 5 servo motorjev

```
void setup() {  
  
  Serial.begin(9600);  
  
  for (int i = 0; i < numOfValsRec; i++) {  
  
    servos[i].attach(7 + i);  
  
  }  
  
}
```

Tukaj sprejema znake iz pycharma

```
void recieveData() {  
  
  char c;  
  
  while (Serial.available()) {  
  
    c = Serial.read();  
  
    if (c == '$') {  
  
      counterStart = true;  
  
    }  
  
    if (counterStart) {  
  
      if (counter < stringLength) {  
  
        receivedString += c;  
  
        counter++;  
  
      }  
  
      if (counter >= stringLength) {  
  
        for (int i = 0; i < numOfValsRec; i++) {  
  
          int num = (i * digitsPerValRec) + 1;
```

```
    if (num <= stringLength && num + digitsPerValRec <= stringLength) {  
        valsRec[i] = receivedString.substring(num, num + digitsPerValRec).toInt();  
    }  
}  
  
receivedString = "";  
  
counter = 0;  
  
counterStart = false;  
  
}  
  
}  
  
}  
  
}
```

Tukaj v zanki pa glede na to kakšen znak pridobi tako se odzove samo odprto ali zaprto

```
void loop() {  
    recieveData();  
  
    for (int i = 0; i < numOfValsRec; i++) {  
        if (valsRec[i] == 1) {  
            servos[i].write(180);  
        } else {  
            servos[i].write(0);  
        }  
    }  
}
```

4. REZULTATI

Povezana roka so odlično in hitro odzove na gibljivost človeške roke, ki jih zazna kamera. In potem te informacije pošlje naprej. Arduino to nato signale naprej pošlje v servo motorje. Kateri se premaknejo zdravstvo vrtijo glede na to, kakšen je signal arduina.

4.1 ODZIV ROKE

Roka se natančno in hitro odzove na dejanja človeške roke. Roka se lahko odpira ali zapira in se rokuje.



Slika 8 Prikazuje povezavo arduina z sestavljeno roko.

Razvoj in implementacija inteligentne povezave med računalniško vizijo in gibanjem robotske roke bo omogočila zmanjšanje nenatančnosti pri kirurških posegih ter izboljšanje rezultatov in varnosti operativnih postopkov.

5. RAZPRAVA

RAZPRAVA Hipoteza 1: Uporaba robotske roke v kirurgiji

Po temeljitem testiranju zmogljivosti sestavljene robotske roke smo prišli do ugotovitve, da roka ni dovolj močna in natančna, da bi lahko uspešno delovala v kirurških posegih. Težko je simulirati potrebno moč v pregibih ter doseči zadostno natančnost pri odpiranju in zapiranju. Najino hipotezo sva raziskala tudi po spletu in prišla do enake ugotovitve.

Kot je avtor v tem članku omenil

<https://www.frontiersin.org/articles/10.3389/frobt.2017.00065/full>

»There is currently no good solution to simulate the force with precise motions of fingers.«

torej prevedeno »Trenutno ni dobre rešitve za simulacijo sile z natančnimi gibi prstov.«

Hipoteza: Zavrnjena.

RAZPRAVA Hipoteza 2: **Ali upravljanje robotske roke na daljavo omogoča učinkovito izvajanje nalog v težko dostopnih ali nevarnih okoljih ter prispeva k večji varnosti in učinkovitosti dela na daljavo?**

Upravljanje robotske roke na daljavo ima potencial, da omogoča učinkovito izvajanje nalog v težko dostopnih ali nevarnih okoljih ter prispeva k večji varnosti in učinkovitosti dela na daljavo. Naš model robotske roke, izdelan iz 3D printane plastike in opremljen z nizkoenergetskimi servo motorji, predstavlja potencialno rešitev za takšne izzive.

Med testiranjem smo ugotovili, da naša robotska roka trenutno ni sposobna premikanja v Z in X osi, kar omejuje njen operativni obseg. Kljub temu pa smo v testiranju potrdili, da je naša roka sposobna delati na daljavo in izvajati osnovne naloge, kot je pridobivanje in spuščanje predmetov, brez neposredne prisotnosti človeške osebe.

Zavedamo se omejitev našega modela, vendar smo hkrati prepričani v njegov potencial in možnosti nadaljnjega razvoja. Z natančnejšo analizo in razumevanjem teh omejitev se bomo v prihodnosti trudili poiskati ustrezne rešitve, ki bi omogočile širšo uporabo in funkcionalnost naše robotske roke.

Glede na dosežene rezultate lahko potrdimo, da upravljanje robotske roke na daljavo omogoča učinkovito izvajanje nalog v težko dostopnih ali nevarnih okoljih ter prispeva k večji varnosti in učinkovitosti dela na daljavo.

Hipoteza: Potrjena.

6. POVZETEK

Raziskovalna naloga je raziskovala povezavo med računalniško vizijo in gibanjem robotske roke za izboljšanje kirurških postopkov. Kljub temeljitemu testiranju sestavljene robotske roke rezultati niso pokazali zadostne natančnosti in moči za uporabo v kirurgiji. Kljub temu pa je raziskava prinesla pomembno spoznanje o izzivih in omejitvah uporabe robotske tehnologije v medicini, ki lahko koristi pri nadaljnjem razvoju na tem področju.

7. ZAKLJUČEK

V zaključku naše projektne naloge o izdelavi robotske roke lahko jasno opazimo, da smo dosegli večje razumevanje in spretnosti na področju inženirstva, tehnologije in inovacij. Izdelava robotske roke ni bila le tehnični podvig, ampak tudi izjemno poučen proces, ki je zahteval sodelovanje, ustvarjalnost in vztrajnost. Naša robotska roka, čeprav še ni v popolnosti izpopolnjena, kaže velik potencial za prihodnji razvoj in uporabo v različnih okoljih. Poleg tega smo med delom na projektu pridobili tudi spretnosti v timskem delu, problem reševanju, načrtovanju in izvedbi, ki nam bodo koristile v prihodnjih projektih in karieri. Skozi to projektno nalogo smo dokazali, da inovacija in tehnologija lahko prinesejo pozitivne spremembe in rešitve za izzive, s katerimi se sooča sodobna družba. Prepričani smo, da bo naša robotska roka le ena od mnogih inovativnih rešitev, ki bodo prispevale k napredku in izboljšanju kakovosti življenja v prihodnosti.

V zaključku sva prišla do ugotovitve da se takšna robotska roka ne more uporabljati na mestu kirurga zaradi tega ker ne moraš nadzirati sile prstov. Roka ni dovolj natančna da bi zamenjala kirurgovo delo saj za to delo potrebuješ veliko natančnost.

8. ZAHVALA

Vesela smo, da smo z raziskovalno nalogo dosegli, kot smo si zadali. Najprej bi se zahvalili mentorjema Urošu Remenihu in Samo Železniku za vso pomoč in podporo pri delu. Zahvaljujemo se bližnjim in družini za podporo pri najinem raziskovalnem delu.

9. PRILOGE

PYTHON PROGRAM ZA ZAZNAVO ROKE TER POŠILJANJU SIGNALA V ARDUINO

```
import cv2
import time
import serial
from cvzone.HandTrackingModule import HandDetector

# Initialize serial communication with Arduino
ser = serial.Serial("COM3", 9600, timeout=0.5)

cap = cv2.VideoCapture(0)
detector = HandDetector(maxHands=1, detectionCon=0.7)

prev_time = 0
curr_time = 0

boundingbox_color = (0, 0, 255)
text_color = (0, 0, 255)
velikost_txt = 2

while True:
    success, img = cap.read()

    # Detect hands
    hands, img = detector.findHands(img)

    if hands:
        for hand in hands:
            fingers = detector.fingersUp(hand)
            print(fingers)

            # Construct the signal string
            signal_str = "".join(map(str, fingers))
            signal_str = f"${signal_str}"

            # Send signal to Arduino
            try:
                ser.write(signal_str.encode())
            except serial.SerialException as e:
                print(f"Error writing to serial: {e}")

            # Draw bounding box and text on the image
            bbox = hand["bbox"]
            cv2.rectangle(img, (bbox[0] - 20, bbox[1] - 20), (bbox[0] + bbox[2] + 20, bbox[1] + bbox[3] + 20),
                          boundingbox_color, 2)
            cv2.putText(img, hand["type"], (bbox[0] - 30, bbox[1] - 30), cv2.FONT_HERSHEY_PLAIN, velikost_txt,
                       text_color, 2)

        # Calculate FPS
        curr_time = time.time()
        fps = 1 / (curr_time - prev_time)
        prev_time = curr_time
        cv2.putText(img, f'FPS: {int(fps)}', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, text_color, 2)

    cv2.imshow("Hand Tracking", img)
    if cv2.waitKey(1) & 0xFF == 27: # Press 'Esc' to exit the loop
        break

# Close the serial connection and release the camera
ser.close()
cap.release()
cv2.destroyAllWindows()
```

C++ KODA V ARDUINOTU

```
#include <Servo.h>
```

```
#define numOfValsRec 5

#define digitsPerValRec 1

Servo servos[numOfValsRec]; // Array to store all servos

int valsRec[numOfValsRec];

int stringLength = numOfValsRec * digitsPerValRec + 1;

int counter = 0;

bool counterStart = false;

String receivedString;

void setup() {

  Serial.begin(9600);

  // Attach each servo to a pin

  for (int i = 0; i < numOfValsRec; i++) {

    servos[i].attach(7 + i); // Assuming servo pins are consecutive (e.g., 7, 8, 9, 10, 11)

  }

}

void recieveData() {

  char c;

  while (Serial.available()) {

    c = Serial.read();

    Serial.print(c);

    if (c == '$') {

      counterStart = true;

    }

    if (counterStart) {

      if (counter < stringLength) {

        receivedString += c;

        counter++;

      }

      if (counter >= stringLength) {

        for (int i = 0; i < numOfValsRec; i++) {

          int num = (i * digitsPerValRec) + 1; // Adjust the indexing
```

```

    if (num <= stringLength && num + digitsPerValRec <= stringLength) {
        valsRec[i] = receivedString.substring(num, num + digitsPerValRec).toInt();
    }
}

receivedString = "";

counter = 0;

counterStart = false;

}

}

}

}

void loop() {
    recieveData();

    // Control each servo based on the corresponding value in valsRec

    for (int i = 0; i < numOfValsRec; i++) {
        if (valsRec[i] == 1) {
            servos[i].write(180); // Rotate to 180 degrees if valsRec[i] is 1
        } else {
            servos[i].write(0); // Rotate to 0 degrees if valsRec[i] is 0
        }
    }

    for (int i = 0; i < numOfValsRec; i++) {
        Serial.print("valsRec[" + String(i) + "]: " + String(valsRec[i]) + " ");
    }

    Serial.println(); // Print a newline to separate values

}

```

10. VIRI IN LITERATURA

Za izdelavo modela robotske roke smo uporabili predlogo inmoov.fr, ki predstavlja enega izmed najbolj priljubljenih odprtih virov za DIY (naredi sam) projekte na področju robotike. Ta vir smo izbrali zaradi njegove priljubljenosti med ustvarjalci ter razpoložljivosti podrobnih navodil

in modelov za izdelavo. Pri nadaljnjem razvoju in prilagajanju smo se osredotočili na specifične potrebe našega projekta, ob upoštevanju možnosti prilagoditve, ki jih ponuja ta platforma

Slika 1: Prikaz logotipa ogrodja Python

Vir: <https://upload.wikimedia.org/wikipedia/commons/thumb/c/c3/Python-logo-notext.svg/115px-Python-logo-notext.svg.png>

Slika 2: Prikaz logotipa platforme Arduino

Vir: <https://brandslogos.com/wp-content/uploads/images/large/arduino-logo-1.png>

Slika 3: Prikazuje stojalo za servo motorje.

Vir: Lasten

Slika 4: Prikazuje nameščene motorčke v notranji del roke.

Vir: Lasten

Slika 5: Servo motorji povezani na svoje mesto.

Vir: Lasten

Slika 6: Prikazuje zaznavanje roke z kodo.

Vir: Lasten

Slika 7: Prikaz Izpisa odprtih ali zaprtih prstov.

Vir: Lasten

Slika 8: Prikazuje povezavo Arduina z sestavljeno roko.

Vir: Lasten

RAZPRAVA IN PREGLED OBJAV:

<https://www.frontiersin.org/articles/10.3389/frobt.2017.00065/full>

MATERIAL: <https://inmoov.fr//hand-and-forarm/>