

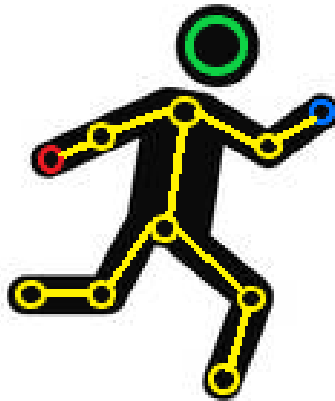
Prva osnovna škola Ogulin

Bolnička ulica 11

47 300 Ogulin

Tehnička dokumentacija

PROGRAM ZA DETEKCIJU TIJELA



Natjecanje iz informatike 2022.

Autor: Gabriel Teo Kaurić, 5.a razred

Mentorica: Valentina Blašković, učiteljica Informatike

Ogulin, ožujak 2022.

Sadržaj

Uvod	2
O meni	3
Ideja	4
Karakteristike i funkcionalnost programa	6
Što je program za detekciju tijela?.....	6
Kome je namijenjen program?.....	8
Prednosti programa.....	9
Tehnički preduvjeti programa	10
Program Python 3.9.....	10
Program PyCharm Community.....	10
MediaPipe Face Detection.....	11
NumPy	11
OpenCV-Python.....	11
Planovi za budućnost	12
Literatura	13
Popis slika	14

Uvod

Python je interpretirani, objektno orijentirani, programski jezik visoke razine s dinamičkom semantikom. Njegova ugrađena strukture podataka, u kombinaciji s dinamičkim tipkanjem i dinamičkim uvezivanjem, čine ga vrlo atraktivnim za brzi razvoj aplikacija, kao i za korištenje kao jezik za povezivanje postojećih komponenti zajedno. Pythonova jednostavna sintaksa koja se lako uči naglašava čitljivost i stoga smanjuje troškove održavanja programa.¹ Pythonova opsežna standardna biblioteka dostupna je u izvornom ili binarnom obliku bez naknade za sve glavne platforme i mogu se slobodno distribuirati. S obzirom na to odlučio sam upravo Python uključiti u izradu svog programa. Moć Python biblioteka razvijenih za analizu i vizualizaciju podataka je nevjerojatna i time je Python zadovoljio još jedan uvjet i potvrdio da je upravo to program u kojem bih trebao raditi.

Zdrav život vrlo je bitan u svakodnevnicu današnjeg života. Kretati se i baviti se tjelovježbom svakako je jedan od ključnih ideja samog pojma zdravlja. Zdravlje i briga oko zdravlja postale su svojevrsna misija suvremenog čovjeka. Briga o sebi u današnje vrijeme stalnih promjena i ubrzanog načina života više je nego opravdana. Većina ljudi koje poznajem, moji prijatelji, rođaci, susjedi bave se različitim vrstama sporta, neki rekreativno, neki profesionalno. Bez obzira na to svima je cilj prihvaćanje zdrave životne navike.

Uzimajući u obzir prethodna dva saznanja spojio sam ih u jedan projekt – Program za detekciju tijela. Spoj je to ohrabrenja za bavljenje tjelovježbom u svakodnevnom životu, sa dozom zabave, a sve kroz programski jezik Python. S obzirom na sve veći porast korištenja i širenja najnovijih tehnologija internet stvari (IoT) i umjetne inteligencije koji olakšavaju svakodnevni život ali i ubrzavaju mnoge procese i odrađuju ih čak i bolje od samih ljudi, neizbježno je bilo uključiti ih u ovaj projekt.

¹ Kalafatić, Z. i suradnici, „Python za znatiželjne, sasvim drukčiji pogled na programiranje“, Element, 2018.

O meni

Ja sam Gabriel Teo Kaurić i učenik sam 5. razreda Prve osnovne škole Ogulin. Robotikom sam se počeo baviti već u prvom razredu osnovne škole. Da, tako je. Tada sam za rođendan dobio robota koji hoda po ravnom, hoda po žici i pleše. Naravno, dobio sam ga u dijelovima, cijeloj kutiji punoj sitnih komadića koji su meni činili pravi izazov. Tako je počela ljubav, strast prema tom hobiju. Niz se nastavlja. Uslijedio je još jedan poklon, a nakon toga nisam više čekao da mi roditelji kupuju, počeo sam štedjeti za nešto što me zaista zanima. Izdvojit ću najzahtjevniju izradu robota. Riječ je o 3D printeru koji posjedujem u svojoj privatnoj kolekciji. Njemu, tj. njegovoj izradi, sam se najviše posvetio i doslovce sam uživao. Sve je bilo posvuda, vijci, matice ovdje i ondje. Trud se isplatio. Rijetkost je imati ga, a ja posjedujem vlastoručno izrađeni unikat.

Početak skupljanja robota počeo sam i programirati. Od početnih programa s blok naredbama u Scatch-u, MakeCode-u, mBlock-u brzo sam prešao na Python. Oduševljen sam svime što taj program nudi. Python je programski jezik koji neki smatraju najboljim načinom za svladavanje osnova programiranja, ali i univerzalno rješenje za većinu problema koje je uopće moguće riješiti programski.



Slika 1. Mindstorms, jedan od prvih ostvaraja kućne radinosti



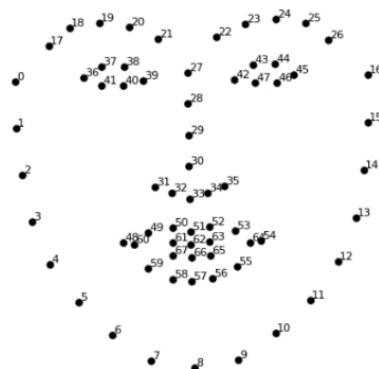
Slika 2. Bubica koja se miče u velikim krugovima



Slika 2. 3D printer samostalno složen

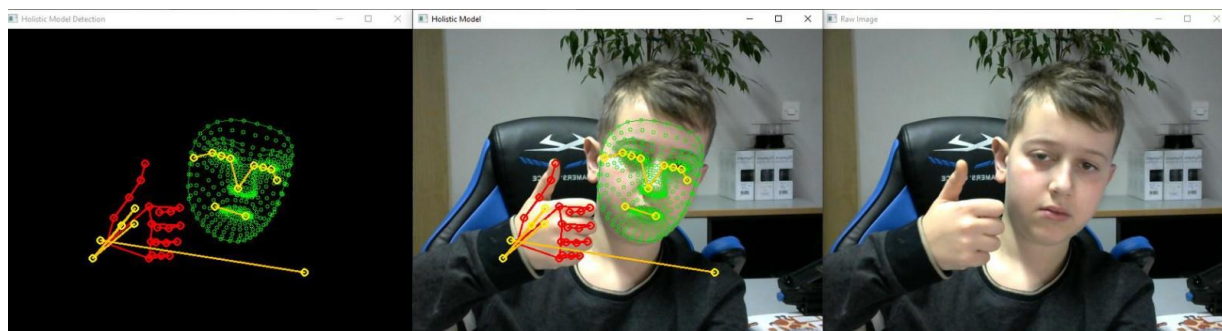
Ideja

Svakodnevnim radom u Pythonu i konstantnim istraživanjem programa uočio sam mogućnost korištenja mrežne kamera računala. Jednostavno uključivanje i isključivanje. Kasnije sam proučavanjem istog uspio dobiti otvaranje tri različita okvira. Svaki okvir imao je drugačiji prikaz osobe ispred kamere: Holistic Model Detection, Holistic Model i Raw Image. Snimanju kamere umetnuo sam mogućnost prepoznavanja lica. Za prepoznavanje lica potrebno je izabrati jedan machine learning algoritam, dati mu podatke i dobiti rezultate. Kod prepoznavanja važno je izabrati i isprobati pogodnu biblioteku za prepoznavanje lica koji odgovara potrebama projekta. Python nudi razne biblioteke i svaki ima posebnu značajke i namjenu.



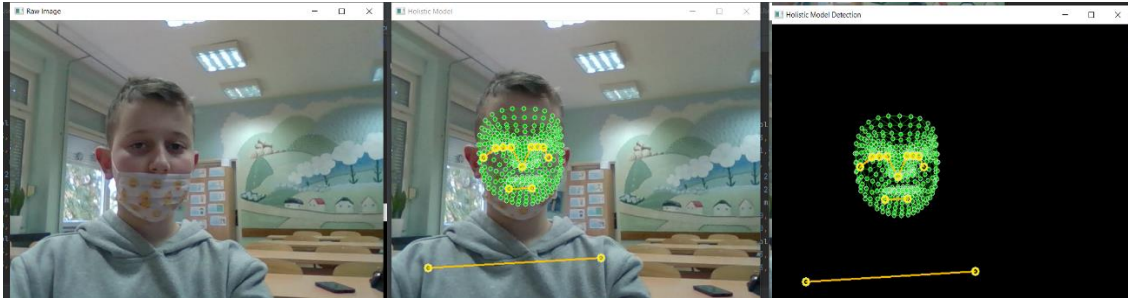
Slika 3. Shematski prikaz lica (prepoznavanje)

Odlučio sam dodatno urediti program postavljanjem ključnih točaka, detekcijskih točaka, na dijelove tijela – ruke, noge, trup. Tada je program postao zanimljiviji. Sam prikaz kroz tri različita okvira nakon detekcije lica program u Pythonu postavljao je i detekcijske točke. Zanimljivost je u tome da detekcijske točke prate pokrete osobe ispred kamere, što znači da pomicanjem osobe pomiču se i detekcijske točke sa minimalnim (gotovo neprimjetnim) kašnjenjem za pokretom.



Slika 4. Početak programa - otvaranje tri okvira i postavljanje detekcijskih točaka

Kôd programa bio je vrlo jednostavan i samim pregledavanjem i isprobavanjem stvarale su se razne ideje. Kombinirao sam koje sve mogućnosti bi se mogle napraviti od ove početne točke. Jedna ideja je bila postaviti kameru na ulazna vrata škole da detektira ima li osoba koja ulazi u školu masku ili ne. Program je tu također pokazao se efikasnim i uspio je detektirati lice unatoč maski.



Slika 5. Detekiranje lica i s maskom

No plan je bio izraditi nešto još zanimljivije, a za to sam mora doraditi program u kojem ću raditi i krenuo sam s istraživanjem.

```

1 import mediapipe as mp
2 import cv2
3 import numpy as mp
4
5 mp_drawing = mp.solutions.drawing_utils
6 mp_holistic = mp.solutions.holistic
7
8 cap = cv2.VideoCapture(0)
9
10 success, img = cap.read()
11 w, h, c = img.shape
12
13 with mp_holistic.Holistic(min_detection_confidence=0.72, min_tracking_confidence=0.6) as holistic:
14     while True:
15         hol = mp.zeros((h, h, 3), dtype="uint8")
16
17         success, img = cap.read()
18         img = cv2.cvtColor(img, 1)
19
20         image = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
21         results = holistic.process(image)
22
23         image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
24         mp_drawing.draw_landmarks(image, results.face_landmarks, mp_holistic.FACE_CONNECTIONS,
25                                 mp_drawing.DrawingSpec(color=(0, 255, 0), thickness=1, circle_radius=3),
26                                 mp_drawing.DrawingSpec(color=(0, 255, 0), thickness=1, circle_radius=2))
27         mp_drawing.draw_landmarks(image, results.left_hand_landmarks, mp_holistic.HAND_CONNECTIONS,
28                                 mp_drawing.DrawingSpec(color=(255, 0, 0), thickness=2, circle_radius=5),
29                                 mp_drawing.DrawingSpec(color=(255, 0, 0), thickness=2, circle_radius=2))
30         mp_drawing.draw_landmarks(image, results.right_hand_landmarks, mp_holistic.HAND_CONNECTIONS,
31                                 mp_drawing.DrawingSpec(color=(255, 0, 0), thickness=2, circle_radius=5),
32                                 mp_drawing.DrawingSpec(color=(255, 0, 0), thickness=2, circle_radius=2))
33         mp_drawing.draw_landmarks(image, results.pose_landmarks, mp_holistic.POSE_CONNECTIONS,
34                                 mp_drawing.DrawingSpec(color=(0, 255, 255), thickness=2, circle_radius=4),
35                                 mp_drawing.DrawingSpec(color=(0, 255, 255), thickness=2, circle_radius=2))
36
37         mp_drawing.draw_landmarks(hol, results.face_landmarks, mp_holistic.FACE_CONNECTIONS,
38                                 mp_drawing.DrawingSpec(color=(0, 255, 0), thickness=1, circle_radius=3),
39                                 mp_drawing.DrawingSpec(color=(0, 255, 0), thickness=1, circle_radius=2))
40         mp_drawing.draw_landmarks(hol, results.left_hand_landmarks, mp_holistic.HAND_CONNECTIONS,
41                                 mp_drawing.DrawingSpec(color=(255, 0, 0), thickness=2, circle_radius=5),
42                                 mp_drawing.DrawingSpec(color=(255, 0, 0), thickness=2, circle_radius=2))
43         mp_drawing.draw_landmarks(hol, results.right_hand_landmarks, mp_holistic.HAND_CONNECTIONS,
44                                 mp_drawing.DrawingSpec(color=(255, 0, 0), thickness=2, circle_radius=5),
45                                 mp_drawing.DrawingSpec(color=(255, 0, 0), thickness=2, circle_radius=2))
46         mp_drawing.draw_landmarks(hol, results.pose_landmarks, mp_holistic.POSE_CONNECTIONS,
47                                 mp_drawing.DrawingSpec(color=(0, 255, 255), thickness=2, circle_radius=4),
48                                 mp_drawing.DrawingSpec(color=(0, 255, 255), thickness=2, circle_radius=2))
49
50         if cv2.waitKey(1) && not == ord('q'):
51             break
52
53         cv2.imshow("Raw Image", img)
54         cv2.imshow("Holistic Face", image)
55         cv2.imshow("Holistic Model Detection", hol)
56         cv2.waitKey(1)
57
58 cap.release()
59 cv2.destroyAllWindows()

```

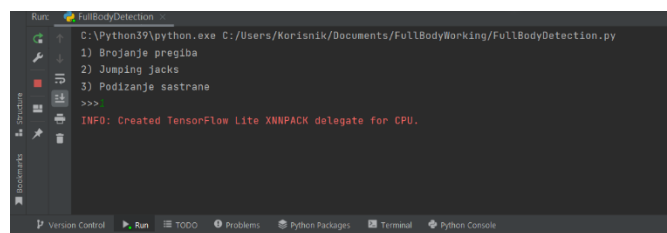
Slika 6. Prvi kôd programa u Pythonu

Karakteristike i funkcionalnost programa

Što je program za detekciju tijela?

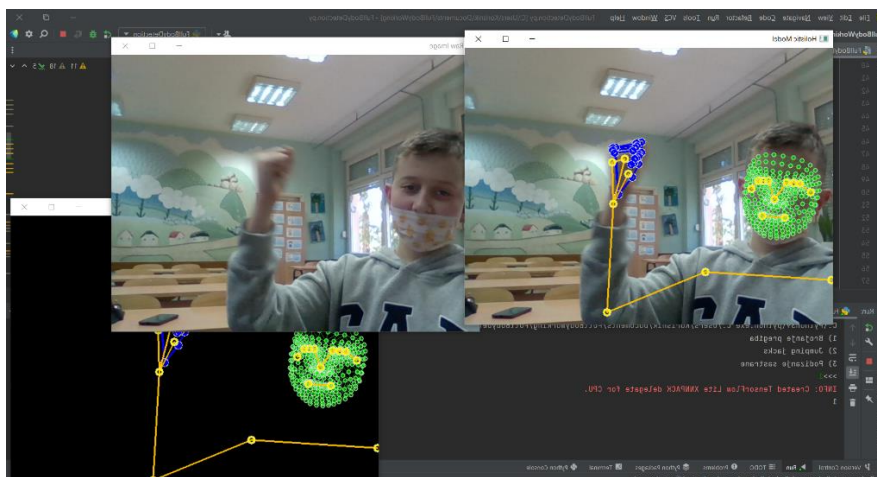
Program za detekciju tijela iako ima vrlo ozbiljnu namjenu možemo reći da je to zabavan program. Zabavan je iz razloga što postavljanjem različitih prikaza osobe ispred kamere program postavlja i detekcijske točke na tijelo osobe i prati njegove pokrete. Većina ljudi kada program detektira lice osobe i postavi detekcijske točke počne se kretati (mahati, pomicati glavu, „bježati“ od kamere i slično). Upravo takvo ponašanje navelo me je na promišljanje kako iskoristiti program i omogućiti osobi da izvodi pokrete koje će u konačnici dovesti do poboljšanja zdravlja, stvoriti dobru naviku kretanja i izvođenja tjelovježbe. Program u bazi za sada ima tri vježbe ali je u planu nadogradnja i povećanje broja tjelovježbi.

Program pomaže osobi ispred kamere u izvođenju vježbi na način da osim što joj pokazuje kako njegovo izvođenje izgleda (kroz tri različita okvira) broji koliko je osoba odradila ponavljanja svake vježbe. Pokretanjem program prvo postavlja pitanje koju vježbu želite raditi te nakon toga spoji se na kameru i čim pronađe osobu ispred kamere (detektira lice) postavi sve detekcijske točke na tijelo osobe i krene računati kut između neke tri točke. Ovisi o vježbi koju ste odabrali, on će ili brojati pregibe ruke, *Jumping Jacks* ili podizanja ruku sa strane (odručenje).



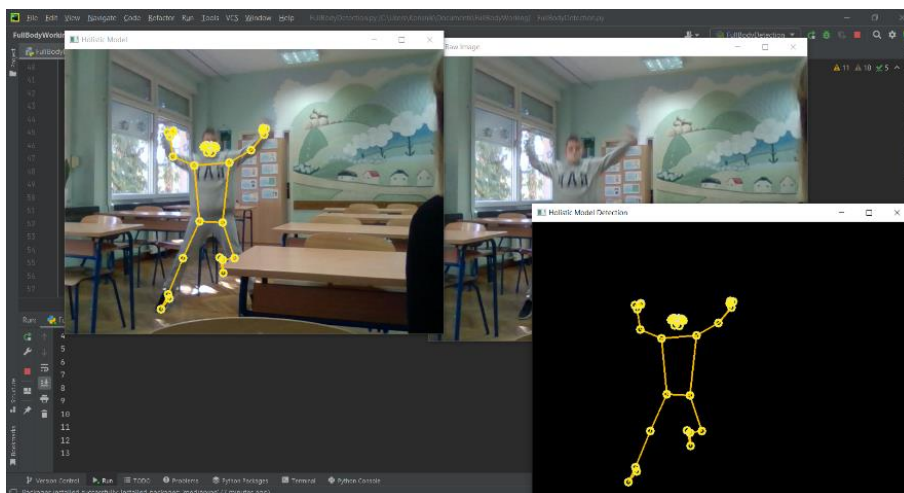
Slika 7. Odabir vježbe

U prvom primjeru možete vidjeti kako izgleda tijek programa ako kod upita *koju vježbu želite odabrati* odaberete broj 1 odnosno brojanje pregiba ruke. Program nakon utipkavanja brojke 1 uključuje kameru i otvara 3 okvira te počinje s mjerenjem kutova kod pomicanja ruke (približavanje šake ruke ramenu i odmicanje, udaljšavanje šake od ramena). Na slici ispod vide se tri ključne točke kojima program računa kut i na taj način broji koliko je ponavljanja te vježbe osoba ispred kamere učinila.



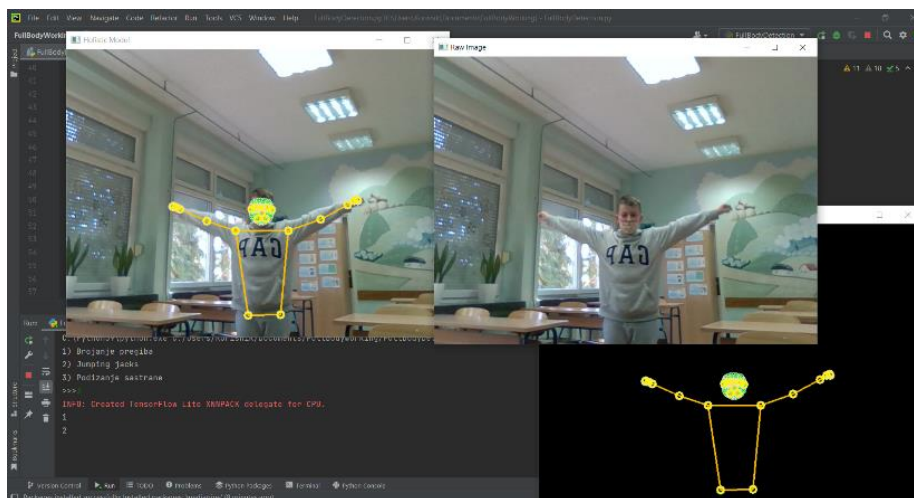
Slika 8. Demonstracije prve vježbe - pregib

Ako kod pokretanja programa i na upit *koju vježbi bismo htjeli izvoditi* odaberemo broj 2, program će nakon pokretanja kamere, detekcije lica i postavljanja detekcijskih točaka, otvoriti tri okvira i započeti s računanjem kutova triju ključnih točaka u izvođenju vježbe *Jumping Jacks*.



Slika 9. Demonstracija druge vježbe - Jumping Jacks

Brojanje zadnje vježbe, podizanje ruku sa strane, pokrećemo odabirom brojke 3 nakon pokretanja programa. Program ponovno detektira tri ključne točke i mjeri kutove između njih te broji broj ponavljanja.



Slika 10. Demonstracija treće vježbe - podizanje ruku sa strane

Kome je namijenjen program?

Ovaj program namijenjen je svim ljudima bez dobnih granica koji su voljni isprobati zabavan način tjelovježbe bilo da se pri tome žele samo dobro zabaviti ili im je namjera istegnuti se nakon dužeg sjedenja ispred računala. Može se koristiti u školama za vrijeme odmora ili u informatičkom kabinetu nakon dužeg rada za računalom. Isto tako vidim ga i kao dodatak uredima u nekim tvrtkama gdje također može poslužiti za istezanje i razgibavanje. Program može biti i dio zabave u vrtićima gdje će se vrtićka djeca vjerujem jako dobro zabaviti i nasmijati ali i razgibati što i je pravi cilj ovog programa. Program bi se mogao naći i u raznim dječjim igraonicama, rođendaonicama i zabavama. Odlično bi se uklopio na treninzima djece i odraslih u uvodnom dijelu satu za zagrijavanje ili razgibavanje. Mogao bi se iskoristiti i kao dio stalnog postava u parkovima tako bi svima i stalno bio dostupan. Vjerujem da vidite koliko je namjena i iskoristivost ovog programa široka, a istovremeno korisna.

Prednosti programa

Smatram da je osnovna prednost mog programa što za rad mu nije potrebna internet veza, već radi *offline*. Time se uvelike smanjuju eventualni troškovi, te mu upravo to omogućuje široku primjenu, brzo instalaciju tj. smještanje u prostore i početak korištenja.

Još jedna prednost ovog programa je to što osoba koja ga koristi ne treba biti programer niti mora dobro baratati računalom da bi pokrenula program. Dovoljno je samo utipkati broj vježbe koju želi izvoditi i program kreće s radom.

Tehnički preduvjeti također nisu zahtjevni jer većina ljudi uz računalo ima i koristi se mrežnom kamerom, a to je minimalna oprema koja je potrebna za funkcioniranje i korištenje programa.

Tehnički preduvjeti programa

Da bi Program za detekciju tijela funkcionirao potrebno je prije svega računalo i montirana, instalirana mrežna kamera. Preporučena konfiguracija sustava:

- 64-bitne verzije sustava Microsoft Windows 10, 8
- najmanje 2 GB slobodnog RAM-a, preporuča se 8 GB ukupne RAM-a sustava
- 2,5 GB prostora na tvrdom disku, preporučuje se SSD
- minimalna razlučivost zaslona 1024x768

Na računalo je potrebno instalirati:

1. Program Python 3.9
2. Program PyCharm Community
3. Biblioteku Mediapipe Face Detection
4. Biblioteku NumPy
5. Biblioteku OpenCV-Python

Program Python 3.9

Za izvršavanje prepoznavanja lica korišten je Python jer je lagan za naučiti, čak i početnici mogu razumijeti kôd na površnoj razini jer je lako razumljiv, a Python omogućuje i korištenje mnogih biblioteka koje olakšavaju programiranje. Jezik je zapravo korišten najviše za znanost podataka i strojno učenje (treniranje) što je i potrebno za ovaj program. Za prepoznavanje lica izabrao sam algoritam, dao mu podatke i dobio rezultat.

Program PyCharm Community

PyCharm je višepatformski IDE koji pruža dosljedno iskustvo na više operativnih sustava. Dostupan je u tri verzije. U svom programu koristio sam verziju Community jer je to projekt otvorenog koda i besplatan je. Iako ima manje značajki za rad i funkcioniranje mom programu omogućuje sve što je potrebno.

MediaPipe Face Detection

To je ultrabrzo rješenje za prepoznavanje lica koje dolazi sa 6 orijentira. Temelji se na detektoru lica s dobrim performansama prilagođenom za zaključivanje mobilnog kretanja. Izvedba detektora u super stvarnom vremenu omogućuje primjenu na bilo koje iskustvo uživo koje zahtijeva preciznu detekciju (pronalazak) lica.

NumPy

NumPy je skraćenica za Numerical Python. To je Python biblioteka koja se koristi za rad s nizovima, a ima funkcije za rad u domeni linearne algebre, Furierove transformacije i matrica. To je projekt otvorenog koda i može se slobodno koristiti.

OpenCV-Python

OpenCV-Python je ogromna biblioteka otvorenog kôda za računalni vid, strojno učenje i obradu slika. Podržava široku paletu programskih jezika kao što su Python. Može obraditi slike i videozapise kako bi identificirao objekte, lica ili čak rukopis čovjeka. Kada se integrira s raznim bibliotekama, kao što je NumPy koja je visoko optimizirana knjižnica za numeričke operacije, tada se povećava broj alata, tj. sve operacije koje se mogu raditi u NumPy-u mogu se kombinirati s OpenCV-om.

Planovi za budućnost

Prije svega planiram proširiti bazu tjelovježbi, odnosno planiram dodati još vježbi za odabir. U ovoj verziji programa stavio sam vježbu za ruke i za cijelo tijelo, stoga bih mogao dodati neke vježbe za noge i vrat. Također, razmislit ću o tome koje bih još vježbe mogao dodati da program bude bogatiji.

Kako bih stekao bolji uvid u samo korištenje programa, testirat ću program uz pomoć mentorici s učenicima 1. razreda osnovne škole kako bih vidio njihovu reakciju na program. Doznat ću od njih što im je bilo zanimljivo, dobro, a što bi eventualno trebao još popraviti kako bi poboljšao program. Drugi dio testiranja provest ću s učiteljima moje škole. Tako ću dobiti povratne informacije iz druge perspektive i vjerujem da će mi to pomoći u daljnjem poboljšanju programa.

Ukoliko to bude moguće razgovarat ću s udrugama na području grada Ogulina (Društvo pedagoga tehničke kulture Ogulin, Društvo Naša djeca Ogulin, Društvo za sportsku rekreaciju Ogulin) o mogućnosti korištenja mog programa na njihovim satovima i radionicama, a sve u svrhu široke primjene i promocije zdravog života, kretanja i tjelovježbe.

Literatura

1. Budin L. i suradnici, „Računalno razmišljanje i programiranje u Pythonu“, Element, 2017.
2. Deljac S., Dimovski Z., „Python - Moj portal 3.0, priručnik za programiranje od 5. do 8. razreda osnovne škole“, Školska knjiga, 2016.
3. Kalafatić, Z. i suradnici, „Python za znatiželjne, sasvim drukčiji pogled na programiranje“, Element, 2018.
4. <https://numpy.org/>
5. <https://www.python.org/>
6. <https://www.w3schools.com/python/>

Popis slika

Slika 1. Mindstorms, jedan od prvih ostvaraja kućne radinosti	3
Slika 2. 3D printer samostalno složen	
Slika 3. Bubica koja se miče u velikim krugovima	3
Slika 4. Shematski prikaz lica (prepoznavanje).....	4
Slika 5. Početak programa - otvaranje tri okvira i postavljanje detekcijskih točaka	4
Slika 6. Detektiranje lica i s maskom.....	5
Slika 7. Prvi kôd programa u Pythonu.....	5
Slika 8. Odabir vježbe.....	6
Slika 9. Demonstracije prve vježbe - pregib.....	7
Slika 10. Demonstracija druge vježbe - <i>Jumping Jacks</i>	7
Slika 11. Demonstracija treće vježbe - podizanje ruku sa strane	8