

Tehnički fakultet Rijeka

Izborni projekt – Mobilna robotika

Autonomna navigacija bespilotnom
letjelicom



Deni Klen, Mateo Srića

Mentor: izv. prof. dr. sc. Kristijan Lenac

Rijeka, veljača 2021

Sadržaj

Uvod.....	3
Upute za korištenje	3
Inicijalno pokretanje	3
Pokretanje.....	3
Python	6
Aplikacija	6
ArUco.....	7
DJI Tello	8
DJI Phantom.....	8
Zaključak	9

Uvod

Tema ovog projekta bila je autonomna navigacija bespilotnom letjelicom i izrada programske podrške koja omogućuje odabir i crtanje putanje same letjelice. Projekt je podijeljen u dvije cjeline – programske podrške napisane u programskom jeziku Python te komunikacija fizičke opreme. Cilj projekta je kreiranje potpuno nezavisne kretnje bespilotne letjelice te mogućnosti dodavanja dodatnih funkcija na samu letjelicu uz pomoć snimke ili slike terena te putanje koju korisnik samostalno može unositi.

Upute za korištenje

Inicijalno pokretanje

Prije prvog pokretanja aplikacije potrebno je kalibrirati kameru koja će biti korištena za daljnji rad. To radimo tako da isprintamo šahovnicu „*checkerboard.pdf*“ priloženu u materijalima na veličinu A4 papira. Zatim je potrebno uzeti minimalno 15 slika iste te šahovnice iz različitih kutova, označiti ih brojevima od 1 do najvećeg broja i postaviti u zadanu datoteku. Nakon toga prilikom prvog pokretanja programa, dobiti ćete opciju odabira kalibracije. Ostalo će program odraditi za Vas.

Pokretanje

Prije otvaranja programa, potrebno je uključiti oba uređaja: dva drona ili kameru i drona te namjestiti uređaj s kojeg će se izvršavati video prijenos na željenu poziciju. Sljedeći korak je postavljanje ArUco markera na drugu letjelicu te njezino pokretanje. Nakon uspješnog pokretanja svih fizičkih uređaja, potrebno je pokrenuti program. Prilikom pokretanja, neposredno nakon pokrenuti će se i prikazati video prijenos s kamere ili drona te na samom središtu video prijensa crveni krug. Potrebno je centrirati akcijsku letjelicu u sam centar toga kruga kako bi uspješno odredili visinsku razliku i mjerilo video prijensa i stvarnog prostora. Nakon uspješnog centriranja, krug će se maknuti i omogućiti prikaz staze.



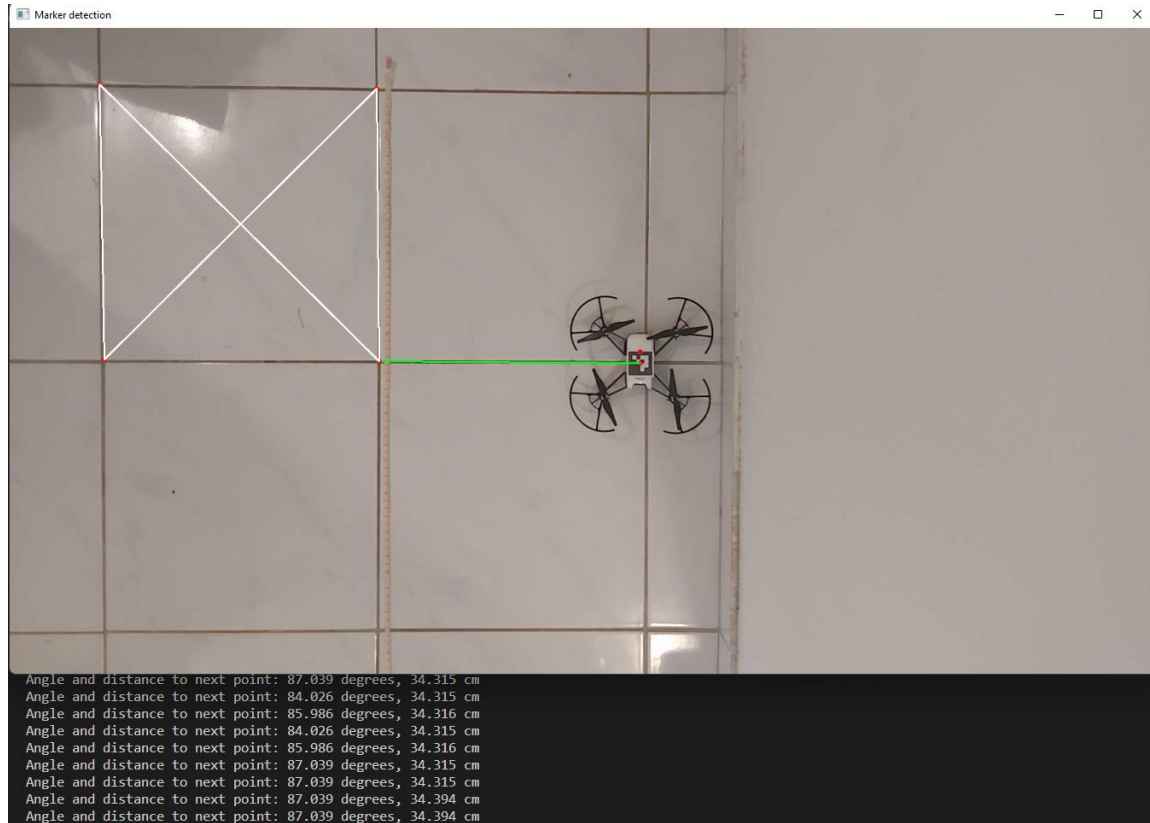
Slika 1. Video prijenos kamere prije centriranja ArUco markera

Zatim, nakon centriranja moramo označiti točke koje bespilotna letjelica mora obići. Za dodavanje točke potreban je dupli lijevi klik s pokazivačem na željenom mjestu. Važno je napomenuti kako je unutar samog programa ograničenje koje nam bespilotna letjelica daje a to je minimalni pomak od 20 cm.



Slika 2. Prepoznavanje ArUco markera nakon kalibracije

Ukoliko je točka na udaljenosti manjoj ili jednako 20 cm od početne točke, točka se neće označiti jer nije valjana. Za ukidanje točke potreban je dvostruki desni klik.



Slika 3. Prikaz crtanja putanje

Međutim stavljeno je ograničenje da se može maknuti samo zadnja dodana točka. Također važno je napomenuti da je prva dodana točka ujedno i zadnja točka leta bespilotne letjelice. Tijekom ocrtavanja staze, prati se udaljenost do najbližeg točke te se ista na slici prikazuje zelenom linijom. U stvarnosti, pritiskom na tipku „m“ bi započeli kretanje letjelice, ali nemogućnost ispravnog letenja DJI Tello-m je tu funkcionalnost onemogućilo. Trenutno je aktivna funkcionalnost praćenja najbliže ili sljedeće točke te prikazane vrijednosti kuteva i udaljenosti koje bi se slale Tellu serijski (rotacija pa translacija). Sa slike 3 je vidljiva mjerna nesigurnost koja je testiranjem utvrđena da je 0.5 centimetara na udaljenosti markera od kamere 1.5 metara te veličini markera 3x3 centimetara.

Python

Za pisanje ovog rada odlučili smo se za programski jezik Python. Python nam omogućuje korištenje knjižnica poput OpenCV, DJITelloPy, Math itd. koje su uvelike olakšale rad i konstruiranje toka aplikacije. Pomoću OpenCV knjižnice možemo prikazati video prijenos uživo s letjelice, omogućiti crtanje parametara za centriranje, točaka za kretanje i putanju kretanja. Također pomoću OpenCV-a možemo ostvariti prepoznavanje i čitanje podataka s ArUco markera te sa saznanjem tih podataka upravljati koordinatama i orijentacijom objekata na kojem se markeri nalaze. Knjižnica DJITelloPy omogućuje nam izmjenu informacija prema i sa bespilotne letjelice Tello. Njom možemo upravljati letjelicom uz nekakva ograničenja zadana od strane letjelice.

Aplikacija

Prije prvog pokretanja aplikacije, potrebno je kalibrirati sve uređaje te pripremiti ih za uspješan rad aplikacije. Za kalibraciju kamere korištena je već unaprijed napisan program koji uzima 19 slika s kamere po kojima radi matricu „neispravnosti“ te ju primjenjuje na svaki sljedeći put izvršavanja.

Zamišljen tok rada aplikacije je inicijalizacija obje letjelice, te podizanje DJI Phantom letjelice na željenu poziciju. Nakon pokretanja kamere izvršava se sljedeći kod gdje na centar video prijenosa postavljamo krug za izračune udaljenosti.

```
def getWindowCenterCircleCoordinates(frame):
    circleCenterX = (int)((frame.shape[1]-1) / 2)
    circleCenterY = (int)((frame.shape[0]-1) / 2)

    return circleCenterX, circleCenterY, constants.CENTER_CIRCLE_RADIUS

def checkMeasuringMarkerPosition(frame, center):
    circleCenterX, circleCenterY, radius = getWindowCenterCircleCoordinates(frame)
    if (circleCenterX < center[0] + radius and circleCenterX > center[0] - radius) and (circleCenterY < center[1] + radius and circleCenterY > center[1] - radius):
        return True
    return False
```

Slika 4. Dobivanje koordinati središta i crtanje kruga za kalibraciju

Uspješno izvršavanje prethodnog koda pokreće novi algoritam za prepoznavanje letjelice te slanje komandi letjelici.

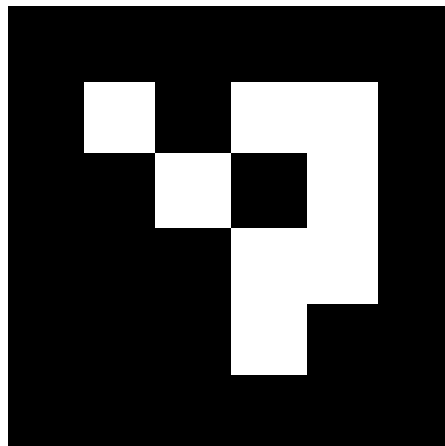
```
# finding markers in image
def findArucoMarkers(image, cameraMatrix, cameraDistortionCoefficients):
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    arucoDictionary = cv2.aruco.Dictionary_get(constants.ARUCO_TYPE)
    arucoParameters = cv2.aruco.DetectorParameters_create()
    markerCorners, ids, _ = cv2.aruco.detectMarkers(gray, arucoDictionary, parameters=arucoParameters, cameraMatrix=cameraMatrix, distCoeff=cameraDistortionCoefficients)
    rotationVectors, translationVectors, _ = cv2.aruco.estimatePoseSingleMarkers(markerCorners, constants.MARKER_SIDE_LENGTH, cameraMatrix, cameraDistortionCoefficients)
    return markerCorners, ids, rotationVectors, translationVectors
```

Slika 5. Kod za prepoznavanje ArUco markera unutar slike (jednog trenutka video prijenosa)

Sljedećim kodom uspješno prepoznavamo letjelicu te dobivamo podatke o njezinoj lokaciji i usmjerenju (slika podataka i slika koda) te nakon toga te podatke posebnim algoritmima pretvaramo u komande koje možemo proslijediti bespilotnoj letjelici Tello kako bi krenula s obilaženjem terena.

ArUco

ArUco marker sintetički je kvadratni marker sastavljen od širokog crnog ruba i unutarnje binarne matrice koja određuje njegov identifikator. Crni obrub olakšava njegovo brzo otkrivanje na slici, a binarna kodifikacija omogućuje njegovu identifikaciju i primjenu tehnika otkrivanja i ispravljanja pogrešaka. Veličina markera određuje veličinu unutarnje matrice. Za našu primjenu koristili smo jedinstveni ArUco marker kojemu smo dodali identifikator letjelice, te orijentacijsku logiku kako bi mogli odrediti smjer prema kojem je usmjerena letjelica.



Slika 6. Aruco marker

DJI Tello

Tello je mini bespilotna letjelica kategorije mini letjelica s mogućnošću leta 100 m od odašiljača. Tello nam daje mogućnost prijenosa uživo snimljene snimke dok u isto vrijeme može primiti komande potrebne za njegovo slobodno i neometano kretanje u prostoru. Prilikom izrade rada Tello je korišten kao primarna letjelica za autonomno kretanje radi njezinog SDK koju nudi u paketu. S time smo značajno olakšali slanje i primanje komandi i odgovora sa i prema letjelici.



Slika 7. Bespilotna letjelica DJI Tello

DJI Phantom

DJI Phantom V2. je druga bespilotna letjelica korištena u ovom radu. Phantom je bespilotna letjelica koja spada u kategoriju srednjih dronova, te ima mogućnost letenja preko 300 m od mjesta odašiljača. S 1-inčnim CMOS senzorom koji može snimati 4K/60fps videozapise i 20MP fotografije, Phantom 4 Pro V2.0 filmskim stvarateljima daje apsolutnu kreativnu slobodu. Sustav prijenosa OcuSync 2.0 HD osigurava stabilnu povezanost i pouzdanost, pet smjerova detektiranja prepreka osigurava dodatnu sigurnost, a namjenski daljinski upravljač s ugrađenim

zaslonom daje još veću preciznost i kontrolu. [1] Širok niz inteligentnih značajki čini letenje mnogo lakšim. Phantom 4 Pro V2.0 potpuno je rješenje za snimanje iz zraka što nam omogućuje kvalitetan video prijenos sa dovoljne visine kako bi imali kontrolu nad cijelim prostorom koji želimo razmatrati.



Slika 8. Беспилотна летјеліца DJI Phantom

Zaključak

Cilj ovog rada bio je napraviti program za stvaranje i obilazak ruta za bespilotno upravljanje letjelicama pomoću druge bespilotne letjelice. Korištenjem različitih tehnologija poput prepoznavanja objekata, ArUco markera ostvarili smo kreiranje i praćenje objekata. Glavni problemi na koje smo naišli prilikom izrade bili su loša podrška firme DJI i njihovih bespilotnih letjelica. U planu za daljnji rad su teme povećanja preciznosti mjerenja, izrade dronova koje bi zadovoljavale uvjete ispravnog rada (video prijenos, manja udaljenost pomicanja) te izvršavanje određenih funkcija na mjestima određenih točki.