

Inferența mișcării micro-vehiculelor aeriene din secvențe video

Conducător științific: Ș.L. dr. ing. Cristian Constantin DAMIAN

Obiectivele proiectului: S-a realizat implementarea unei rețele de învățare profundă utilizând o bibliotecă specifică pytorch, antrenarea și testarea rețelei pe un set de date cu secvențe de imagini adnotate cu poziția și unghiul camerei video. Ținând cont de importanța dezvoltării unei astfel de rețele neuronale convoluționale, care să se poată folosi în sarcina de odometrie vizuală, obiectivele principale sunt următoarele:

- Încărcarea datelor provenite de la baze de date și prelucrarea acestora, pentru a putea deveni compatibile cu rețeaua neuronală;
- Proiectarea și implementarea unei rețele neuronale convoluționale care să realizeze estimarea mișcării de rotație și translație a unui micro-vehicul aerian;
- Testarea și evaluarea celor mai bune tehnici de antrenare, relative la bazele de date disponibile și evidențierea punctelor puternice și slabe ale acestei alternative de odometrie vizuală;

Realizarea proiectului și rezultate obținute: Pentru realizarea acestei lucrări, a fost nevoie de realizarea unor multiple etape de implementare și adaptări. Pentru început, contribuția cea mai mare a constat în realizarea antrenării rețelei convoluționale. În cadrul acestei etape esențiale, am testat efectul antrenării pe mai multe baze de date, cu scopul de a evidenția performanțele unei rețele neuronale convoluționale în contextul estimării parametrilor de mișcare. Am antrenat 3 modele diferite, unul bazat doar pe baza de date Kitti, cel de al doilea folosind doar baza de date NTU, iar în final s-a încercat extragerea performanțelor unui model antrenat cu ambele baze de date disponibile. Modelele de care avem nevoie pentru a estima parametrii de mișcare al unui micro-vehicul aerian sunt ultimele 2 amintite. Acestea au avut la dispoziție un număr total de 500 de epoci de antrenare. Anterior acestei etape, a fost nevoie de a realiza un script care să prelucreze pozițiile absolute pe care ni le ofereau baze de date, în poziții relative la momentul anterior. Pozițiile relative au fost apoi scrise într-un document de tip Excel alături de perechile de imagini și au fost folosite pentru a încărca datele reale în rețeaua neuronală. Această transformare este explicată în capitolul 3.1.1. O altă etapă în care am contribuit este găsirea și adaptarea unei arhitecturi de rețea neuronală convoluțională. Astfel, modelele create au la baza arhitectura ResNet50, dar a fost nevoie să realizăm adaptarea primului strat din rețea pentru a accepta o structură de date cu 6 canale de culori, care reprezintă perechea de imagini suprapuse pe canalul de culoare. Această arhitectură este în general folosită pentru sarcinile de clasificare, așa că a fost nevoie de realizarea unui bloc adițional de straturi în care să realizăm partea de regresie. Această arhitectură este prezentată în amănunt în capitolul 3.2. Alte contribuții care merită menționate sunt realizarea structurilor de date numite dataloader, care au fost folosite pentru a realiza încărcarea datelor din bazele de date în rețeaua neuronală. Au fost de asemenea, realizate diferite funcții sau scripturi de prelucrare a datelor pentru a putea configura și duce la capăt realizarea acestei sarcini de odometrie vizuală.

Model și mediu testare	Eroare X	Eroare Y	Eroare Z
Model 1 Mediu cunoscut	8.51E+09	5.18E+09	8.17E+09
Model 2 Mediu cunoscut	7.95E+09	8.37E+09	8.81E+09
Model 1 Mediu necunoscut	3.20E+11	1.69E+11	1.58E+11
Model 2 Mediu necunoscut	2.31E+11	1.12E+11	1.90E+11

Tabel 1: Erori pentru parametrii de translație

De asemenea, pentru a putea face comparații rapide și pentru a ne referi ușor la fiecare model în parte, vom atribui numere modelelor. Astfel, vom considera modelul antrenat doar cu baza de date NTU ca fiind "Modelul 1", iar modelul antrenat cu ambele baze de date ca fiind "Modelul 2". Această abordare ne va permite să identificăm și să discutăm despre fiecare model într-un mod în care să nu existe confuzii.

În privința realizării parametrilor de translație, modelul numărul 1 s-a descurcat mult mai bine în mediul cunoscut, având o eroare mai mică pentru translația pe axa y și z, iar pe axa x fiind foarte apropiat de modelul 2. În ceea ce privește mediul necunoscut, modelul numărul 1 a avut valori ale erorii foarte apropiate de modelul numărul 2, ceea ce îl face cea mai bună variantă pentru a determina mișcarea de translație dintre 2 imagini, mai ales dacă modelul a fost testat cu secvențe din același mediu în care a fost și antrenat.

Model și mediu testare	Eroare Yaw	Eroare Pitch	Eroare Roll
Model 1 Mediu cunoscut	4.32E+08	2.32E+09	5.33E+08
Model 2 Mediu cunoscut	3.35E+08	1.88E+09	1.21E+08
Model 1 Mediu necunoscut	1.53E+08	1.31E+09	1.02E+09
Model 2 Mediu necunoscut	4.57E+08	2.67E+09	8.08E+08

Tabel 2: Erori pentru parametrii de rotație

Unghiurile de rotație au evidențiat faptul că modelul nu mai este atât de sensibil la mediu, deoarece pe ambele secvențe de imagini, modelele au avut performanțe apropiate. Acest comportament este unul dorit și indică o evoluție bună a modelelor. Folosind secvența din mediul cunoscut, modelul numărul 2 a reușit să aibă predicții mai corecte și apropiate de cele reale comparativ cu modelul 1, dar diferențele nu sunt atât de mari cât să putem spune că modelul 2 este câștigătorul, mai ales dacă privim ce s-a întâmplat cu cele 2 modele când au primit secvențele din mediul necunoscut. În acest caz, modelul numărul 1, încă odată reușește să aibă performanțe foarte bune pentru 2 din 3 unghiuri, iar al 3 lea nu este foarte departe de valorile celui alt model.

Un alt lucru pe care trebuie să îl luăm în calcul pentru a alege cea mai bună variantă de a realiza sarcina propusă în această lucrare, este timpul de antrenare necesar. Timpii de rulare au fost menționați în capitolul 4, dar nu a fost realizată o comparație directă între aceștia. Pentru antrenarea modelului numărul 1, au fost necesare 26 de ore de rulare, în care s-au realizat 500 de epoci, iar funcțiile de pierdere au fost reduse cel mai mult de acest model. În cazul modelului numărul 2, timpii de rulare au constat în aproximativ 25 de ore pentru antrenarea a 250 de epoci cu baza de date Kitti, iar apoi alte 13 ore pentru antrenarea cu baza de date NTU. În total avem 38 de ore, deci o diferență de aproximativ 12 ore între timpii de antrenare.

În concluzie, este mult mai optim să antrenăm o astfel de rețea neuronală cu date provenite direct dintr-o baza de date special creată pentru micro-vehicule aeriene, deoarece timpul de antrenare este mai redus, erorile cumulate au fost mai mici, iar acest model a reușit să reducă cel mai mult funcțiile de pierdere. Pentru a maximiza potențialul său și a aduce îmbunătățiri ulterioare, putem continua antrenarea acestui model pe parcursul a mai multor epoci și cu o gama mai variată de imagini. Pentru a adresa problema acumulării de erori, o variantă care s-ar putea arata foarte eficient în acest domeniu, este schimbarea arhitecturii CNN cu una recursivă de tipul RNN sau CRNN, în așa fel încât modelul să își ajusteze parametrii, nu doar în funcție de fiecare imagine, ci și de cele anterioare. Aplicând aceste îmbunătățiri am putea obține o rețea capabilă să fie integrată într-un astfel de micro-vehicul aerian sau orice altă sarcină din acest domeniu.