



Rafinarea parametrilor orbitali ai sateliților care reintră în atmosferă (RESTORE)

Cod proiect: PN-IV-P7-7.1-PED-2024-0369

Contract Nr. 99PED din 05/05/2025

Web: <https://cv.utcluj.ro/restore/>

Etapa 1: Definirea cerințelor și a metodologiei de validare, implementare componente
Perioada: 05/05/2025 - 31/12/2025

Consortiu:

CO - Universitatea Tehnică din Cluj-Napoca:

Radu Dănescu, Răzvan Itu, Attila Fuzes, Mircea Mureșan

P - Control Data Systems SRL:

Ovidiu Rațiu, Ana Rednic, Vlad Rațiu, Cristian Olariu, Zoltan Padrah

Cuprins

Introducere - rezumatul etapei.....	2
A1.1. Definirea arhitecturii hardware pentru sistemul dual de achiziție și măsură.....	2
A1.2. Definirea arhitecturii software pentru sistemul dual de achiziție și măsură.....	5
A1.3. Definirea scenariilor și a metodologiei de validare.....	7
A1.4. Achiziția datelor preliminare (efectuare experimente pentru achiziția de date).....	8
A1.5. Implementarea sistemului hardware pentru observarea în timp real a obiectelor în proces de reintrare - partea 1.....	9
A1.6. Implementarea sistemului software pentru observarea în timp real a obiectelor în proces de reintrare - partea 1.....	16
A1.7. Proiectarea și implementarea componentelor software pentru rafinarea parametrilor orbitali - partea 1.....	18

Introducere - rezumatul etapei

În prima etapă a proiectului RESTORE au fost definite cerințele sistemului, arhitecturile hardware și software, precum și metodologia de validare necesară dezvoltării unei platforme capabile să observe în timp real sateliți aflați în proces de reintrare și să rafineze parametrii lor orbitali. Sistemul propus se bazează pe două stații de observație independente, conectate prin internet, care pot detecta obiectele, procesa local imaginile și genera seturi TLE rafinate, comunicate în timp real celeilalte stații.

La nivel hardware, analiza a vizat configurarea optimă a echipamentelor optice, fiind comparate mai multe combinații cameră–obiectiv. În urma testelor privind câmpul vizual și rezoluția unghiulară, s-a stabilit o configurație finală bazată pe un aparat Canon EOS R8 cu obiectiv de 50 mm, capabil să ofere un compromis bun între sensibilitate și acoperirea cerului. O componentă esențială a fost dezvoltarea unui sistem de timestamping de mare precizie, folosind un microcontroller ESP32, un modul GPS cu semnal PPS și un adaptor Hot Shoe pentru detectarea exactă a momentului expunerii, permițând corelarea riguroasă a imaginilor cu timpul global. De asemenea, au fost evaluate mai multe module GNSS și a fost realizat un banc de testare complex pentru măsurarea întârzierilor și validarea duratelor reale de expunere și transfer.

Arhitectura software a unei stații include un modul de procesare a datelor și un modul de control al achiziției, gestionat de un microcontroler. Pachetul software integrează componente pentru preluarea și interpretarea TLE-urilor, detecția sateliților, calibrare astrometrică și urmărire. În implementarea software, a fost analizată variabilitatea aspectului țintelor, de la dăre luminoase la obiecte de contrast redus sau mișcare aparentă mică. Pentru a crește robustețea detecției, au fost combinate două metode: segmentarea diferențială între cadre și detecția cu rețele neuronale, acestea fiind integrate într-un sistem unificat capabil să gestioneze toate scenariile relevante pentru reintrarea sateliților.

Au fost definite scenariile de validare, care stabilesc condițiile necesare pentru ca sistemul să fie considerat funcțional. Au fost realizate achiziții preliminare de date cu echipamentele UTCN și cu Observatorul Astronomic, incluzând observații reale ale unui satelit aflat în reintrare (Cluster II Rumba).

Obiectivele etapei au fost îndeplinite cu succes și chiar depășite.

A1.1. Definirea arhitecturii hardware pentru sistemul dual de achiziție și măsură

Arhitectura generală a sistemului dual

În timpul acestei faze inițiale, au fost analizate numeroase variante și s-au luat decizii majore cu privire la arhitectura conceptuală a sistemului. Arhitectura rezultată este prezentată în Figura 1. Sistemul este compus din două stații de observare care comunică între ele printr-o conexiune internet. Conexiunea nu trebuie să fie de mare viteză, deoarece datele comunicate vor fi

constituite din parametri orbitali în format TLE, inițiali sau rafinați. Aceste date au o dimensiune mică (sub 200 octeți), și sunt independente de poziția observatorului sau de intervalul de timp. Setul de parametri rafinați poate fi utilizat pentru observații mai precise din orice locație de pe Pământ.

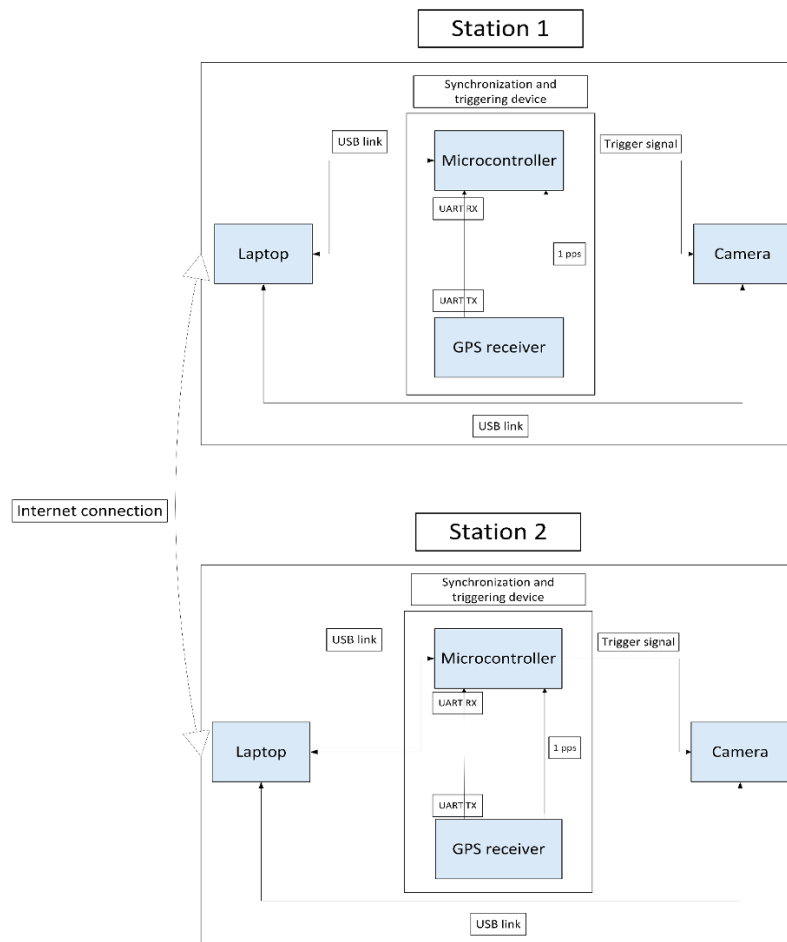


Figura 1. Arhitectura hardware a sistemului

Alegerea camerei foto și a obiectivului

Alegerea unei combinații potrivite de cameră și obiectiv este crucială pentru succesul proiectului, așa că a fost efectuată o analiză detaliată a posibilelor compromisuri. Indicatorii cheie care stau la baza acestei decizii sunt câmpul vizual și rezoluția unghiulară. Testele au fost efectuate folosind o cameră Nikon D7200 23,5x15,6 mm și două obiective: Nikon 70-300 mm f/4-5,6G, Nikon 18-55 mm f/3,5-5,6G.

Câmpul vizual (Field of View) este extinderea unghiulară a mediului văzut la un moment dat. Această metrică poate fi calculată pe baza distanței focale și a lățimii senzorului folosind următoarele ecuații:

$$\text{Horizontal FOV} = 2 \cdot \arctan((\text{sensor width}) / (2 \cdot \text{focal length})) \cdot 180 / \pi \quad (\text{degrees})$$

$$\text{Vertical FOV} = 2 * \arctan((\text{sensor height}) / (2 * \text{focal length})) * 180 / \pi \text{ (degrees)}$$

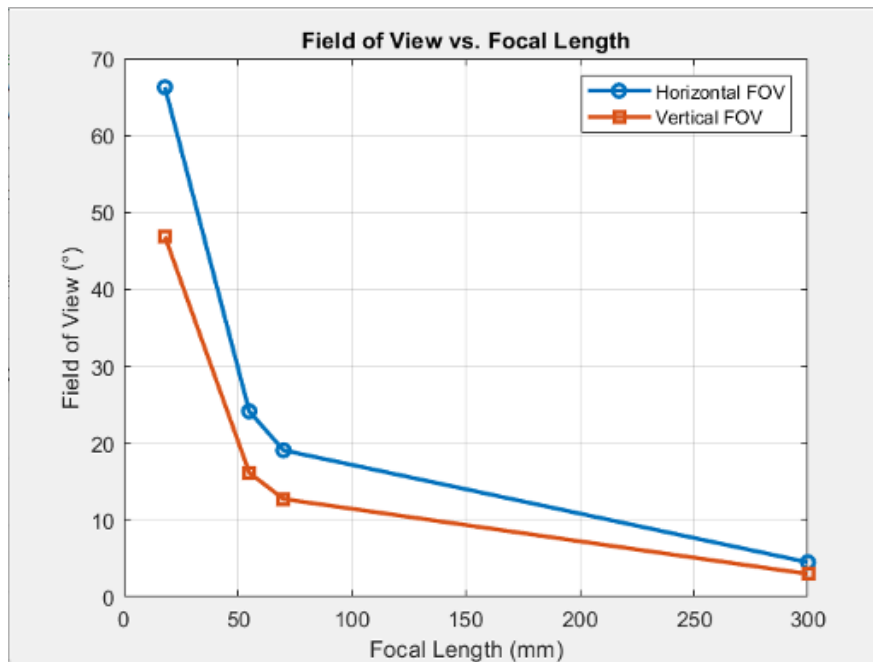


Figura 2. Metrica câmpului vizual (FOV) vs. Metrica distanței focale.

Rezoluția angulară: deși câmpul vizual (FOV) este o metrică cheie, deoarece arată porțiunea de cer capturată, este important să nu uităm de rezoluția angulară a imaginilor, care reprezintă exact câtă informație este capturată într-un singur pixel. Această metrică poate fi calculată folosind următoarele ecuații:

$$\text{Horizontal Angular Resolution} = (\text{Horizontal FOV}) / (\text{Horizontal Pixels}) * 3600 \text{ (arcseconds/pixel)}$$

$$\text{Vertical Angular Resolution} = (\text{Vertical FOV}) / (\text{Vertical Pixels}) * 3600 \text{ (arcseconds/pixel)}$$

Configurația preliminară a generat următoarele rezultate:

Angular Resolution (arcsec/pixel)					
6000x4000 px		4496x3000 px		2992x2000 px	
Min focal length	Max focal length	Min focal length	Max focal length	Min focal length	Max focal length
11.43 x 11.44	2.69 x 2.68	15.26 x 15.26	3.59 x 3.57	22.93 x 22.89	5.40 x 5.36

Cu un câmp vizual mai larg, este captată o porțiune mai mare a cerului, dar rezultă o luminosităe mai mică pentru pixelii individuali, ceea ce face mai dificilă detecția obiectelor slabe. Cu un câmp vizual mai mic, este captată o porțiune mai mică a cerului, dar cu mai multe

detalii, rezultând o detectare mai ușoară a sateliților slabi. Trebuie găsit un compromis bun pentru a echilibra acoperirea cerului și sensibilitatea detecțiilor.

Configurația finală: având în vedere toți factorii descriși mai sus, următoarea configurație a fost considerată cea mai potrivită pentru acest telescop: un aparat foto Canon EOS R8 cu un senzor optic de 36 mm x 24 mm, echipat cu un obiectiv RF de 50 mm, atingând un câmp vizual de 39,59 x 26,99, ceea ce înseamnă o rezoluție unghiulară de 23,75 x 24,29 secunde de arc/pixel pe imagini 4k, asigurând o acoperire suficient de largă cu sensibilitatea necesară pentru detectarea sateliților pe orbita joasă a Pământului.

A1.2. Definirea arhitecturii software pentru sistemul dual de achiziție și măsură

Arhitectura software a unei stații de observație

O stație (un instrument optic, o cameră foto, un echipament de calcul și o montură) va funcționa independent și va avea două obiective:

- Procesarea datelor: deoarece trimiterea de imagini de înaltă rezoluție către un server pentru procesare ar fi inefficientă, stațiile vor gestiona sarcini dificile din punct de vedere computațional, cum ar fi detectarea obiectelor, vederea stereo și rafinarea parametrilor. Pentru a aborda această problemă, fiecare stație va fi echipată cu puterea de calcul necesară pentru a executa algoritmul de procesare a imaginilor implementat în Python cu PyTorch.
- Controlul procesului de achiziție: stația trebuie să controleze montura camerei cu o precizie ridicată și să declanșeze procesul de achiziție a imaginilor la momentul corect, aceste sarcini urmând să fie rezolvate de un microcontroler conectat la montură, la cameră și la PC-ul portabil. Arhitectura software a unei stații este prezentată în Figura 3.

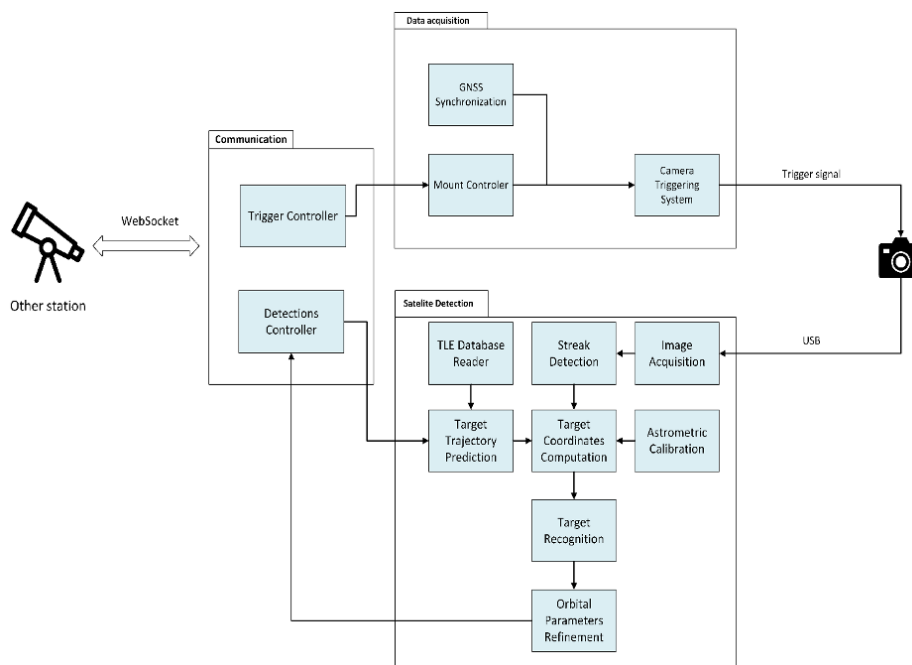


Figura 3. Arhitectura software a unei stații de observație.

Pachetul software de detecție și recunoaștere a sateliților

Scopul principal al acestui pachet software este de a procesa imagini secvențiale, de a detecta și recunoaște sateliții pe cadre. Acesta este alcătuit din 4 clase principale, așa cum este prezentat în Figura 4.

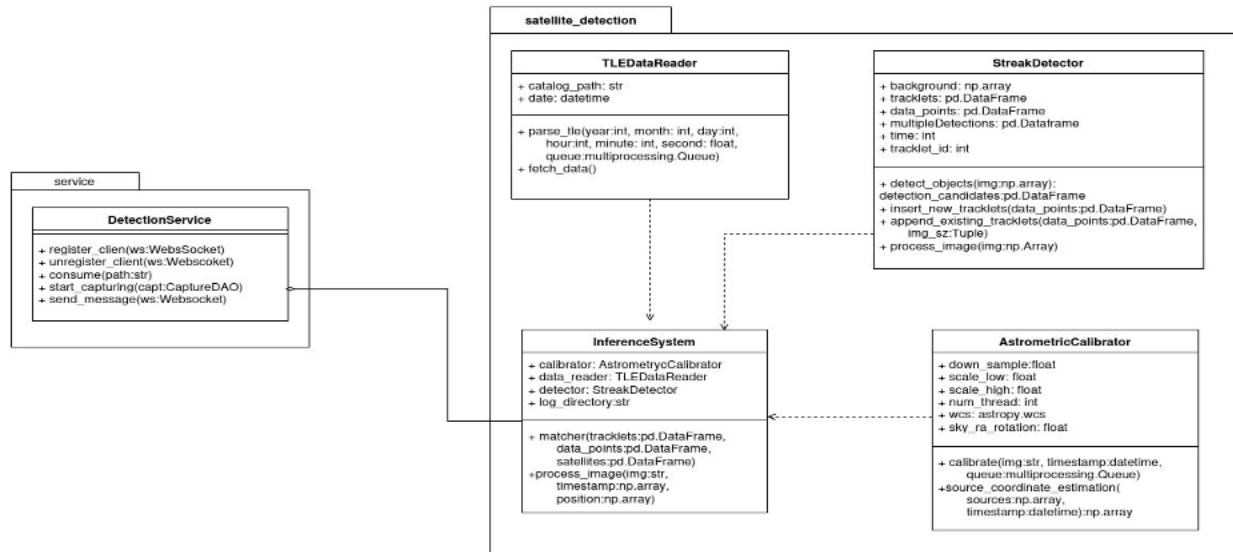


Figura 4. Diagrama UML a pachetului de detecție a sateliților.

Clasa *TLEDataReader* preia cataloage TLE de pe space-track.org folosind metoda `fetch_data` și calculează poziția tuturor sateliților într-un anumit interval de timp folosind metoda `parse_tle`.

Clasa *StreakDetector* se ocupă de sarcina de detecție a obiectelor și ține evidența tuturor sateliților urmăriți.

Clasa *AstrometricCalibrator* efectuează maparea dintre spațiul imaginii și coordonatele din lumea reală folosind funcționalitățile bibliotecii astrometry.net.

Modelarea comportamentală a pachetului

Dintr-o perspectivă comportamentală, pachetul are două faze:

- Faza de inițializare (prezentată în Figura 5) este pusă în mișcare odată cu sosirea primei imagini dintr-o secvență de cadre. În timpul acestui proces, sunt gestionate următoarele sarcini:
 - o Calibrarea inițială
 - o Procesarea inițială a imaginii
 - o Preluarea catalogului TLE

Faza de inferență (prezentată în Figura 6), declanșată odată cu sosirea oricăror cadre următoare, gestionează în paralel următoarele sarcini:

- o Procesarea imaginilor, detecția obiectelor și urmărirea lor
- o Calcularea pozițiilor sateliților folosind catalogul TLE preluat
- o Un proces de recalibrare periodică efectuat în intervale de timp bine definite

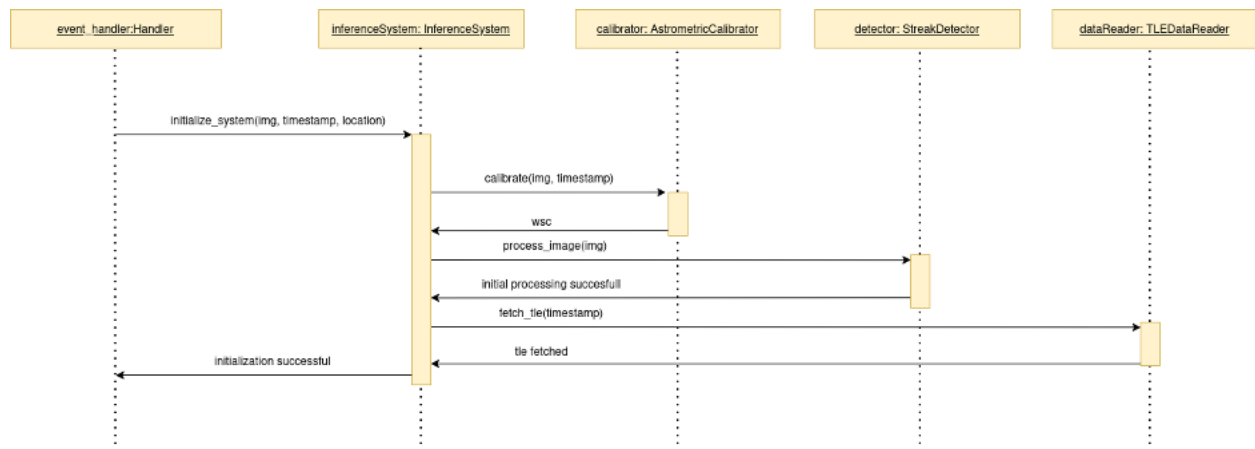


Figura 5. Diagrama de secvență a fazei de inițializare

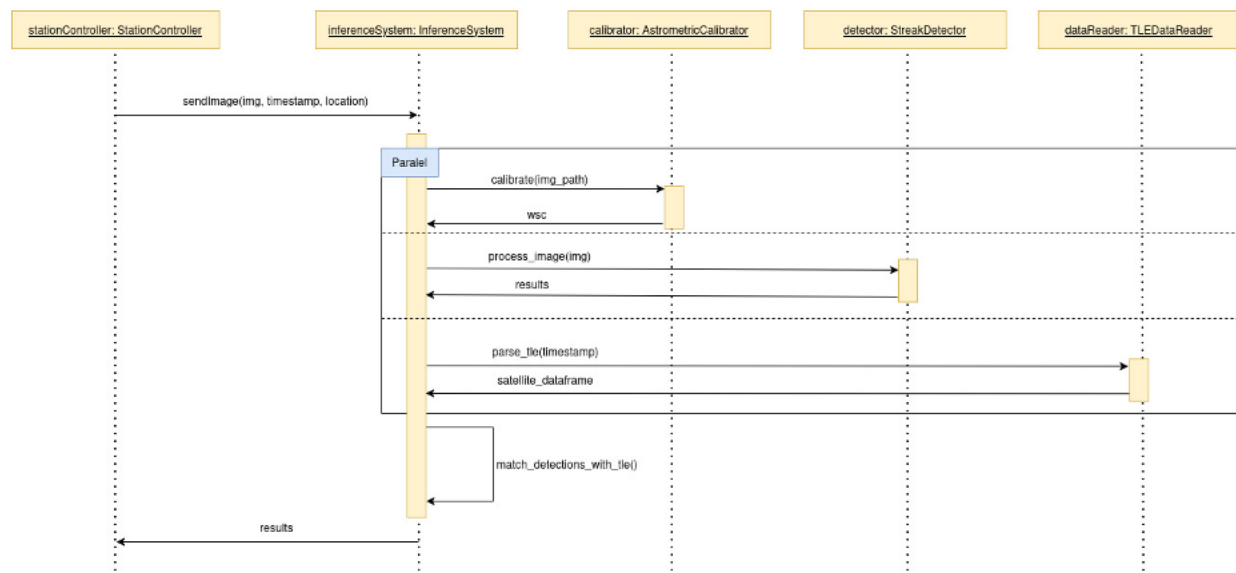


Figura 6. Diagrama de secvență a fazei de inferență

A1.3. Definirea scenariilor și a metodologiei de validare

În urma analizei capacităților actuale și potențiale ale viitorului sistem, și a analizei comportamentului sateliților care vor reintra în atmosferă, a fost stabilit următorul scenariu de funcționare și validare:

1. Primirea informațiilor despre obiectul în proces de reintrare, incluzând elementele orbitale în format TLE (Two Line Element)
2. Predicția poziției obiectului în viitorul apropiat, determinarea posibilității ca obiectul să fie observabil din zona Cluj-Napoca
3. Pregătirea instrumentelor și începerea procesului de achiziție de imagini, cu cel puțin 30 minute înainte de trecerea prezisă de elementele orbitale (dacă sunt mai multe predicții, care diferă mult între ele, se va lua cea care prezice trecerea cea mai devreme).
4. Prima stație care va detecta obiectul va rafina parametrii orbitali în mod automat, și va comunica către cealaltă stație noii parametri sub forma unui fișier TLE actualizat.
5. Ambele stații, și posibil orice altă stație disponibilă, vor utiliza noii parametri pentru a observa obiectul în continuare.
6. Parametrii orbitali vor fi rafinați cu fiecare nouă observație.

Sistemul realizat va fi considerat validat dacă va îndeplini simultan următoarele condiții:

1. Va fi posibilă punerea în funcțiune a sistemului în timp de mai puțin de 60 de minute de la primirea notificării despre obiectul în proces de reintrare
2. Va putea funcționa fără întreruperi cel puțin 30 de minute înainte de momentul prezis al trecerii, și cel puțin 30 de minute după acest moment.
3. Va detecta obiectul, dacă condițiile meteorologice o permit
4. Va rafina parametrii orbitali și va comunica noii parametri în mai puțin de 1 minut
5. Parametrii rafinați vor permite o eroare mai mică de 1 grad între noua predicție și poziția reală a țintei, pentru a putea fi observată cu un telescop de câmp vizual îngust.

A1.4. Achiziția datelor preliminare (efectuare experimente pentru achiziția de date)

Folosind echipamentele și sistemul funcțional al UTCN au fost preluate mai multe secvențe de imagini, în datele de 21.07.2025, 24.07.2025, 25.07.2025, 1.09.2025, 4.09.2025, 5.09.2025, 21.09.2025. Deși aceste imagini nu conțin imagini cu sateliți care reintră în atmosferă, au fost utile pentru dezvoltarea continuă și testarea algoritmilor și a capabilităților sistemului de funcționare în timp real.

Folosind instrumentele Observatorului Astronomic Cluj-Napoca, a fost posibilă observarea unui satelit în proces de reintrare, obiectul Cluster II Rumba:

- O secvență de observații în data de 27.09.2025
- O secvență de observații în data de 8.10.2025
- O secvență de observații în data de 19.10.2025
- Trei secvențe de observații, procesate și transmise imediat ce au fost preluate, în data de 20.10.2025, ultima dată când satelitul a fost vizibil din Cluj-Napoca înainte de reintrare.

Imaginile preluate de Observatorul Astronomic, în calitate de subcontractor în cadrul proiectului, au fost capturate în modul precise object tracking, mod în care stelele sunt observate ca dâre și obiectul urmărit este punctiform, precum se poate observa în figura 7.

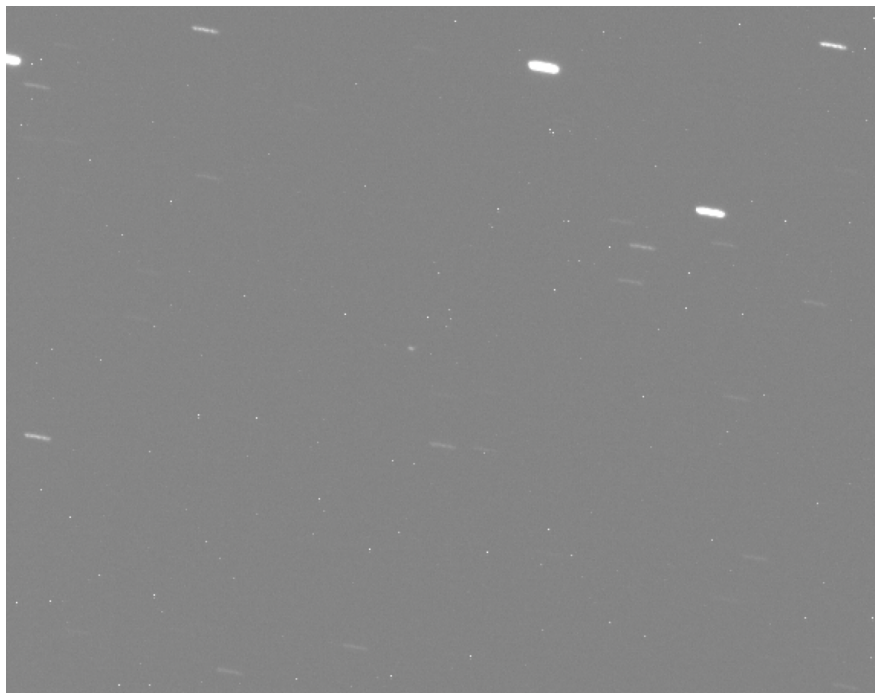


Figura 7. Observare obiect în modul precise object tracking.

A1.5. Implementarea sistemului hardware pentru observarea în timp real a obiectelor în proces de reintrare - partea 1

Achiziția sincronizată a imaginilor

Sincronizarea coordonatelor detectate din imagini cu timpul global este esențială pentru a putea avea un sistem eficient de corecție a parametrilor orbitali ai sateliților. Din acest motiv, am încercat să îmbunătățim procesul de estimare a timpului exact al imaginilor. În varianta de la care am pornit, am utilizat un sistem bazat pe GPS pentru a declanșa la un timp exact procesul de captură. Din păcate, experimentele au demonstrat că aparatul foto are un timp variabil între momentul în care primește semnalul de declanșare și momentul în care se pornește expunerea propriu-zisă a imaginilor. Din acest motiv, am ales să explorăm o soluție alternativă, în care citim semnalul furnizat de cameră la momentul începerii expunerii, și alocăm acestuia o marcă de timp, în loc să declanșăm camera la un timp anume dat.

Am realizat un dispozitiv bazat pe un modul ESP32, un receptor GPS cu semnal PPS, pe care l-am atașat camerei DSLR, având ca scop obținerea unei declanșări fotografice precise și a unei înregistrări riguroase a timpului de captură. Prin intermediul comunicației Bluetooth, utilizatorul poate solicita poziția geografică actuală, poate seta timpul de expunere și poate crea un nume de fișier ce include data, ora și miliseconda exactă a declanșării corespunzătoare imaginii achiziționate.

Schema de conectare a sistemului este prezentată în Figura 8.

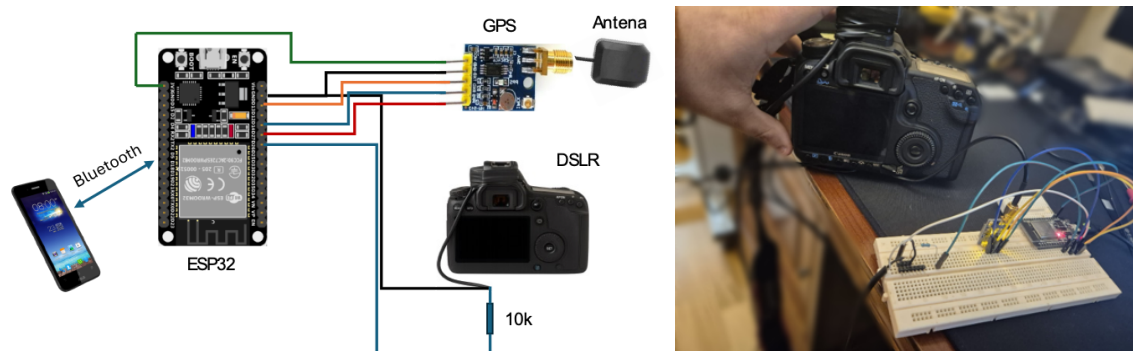


Figura 8. Sistemul pentru declanșare și temporizare

Sistemul este construit în jurul unui microcontroller ESP32. Acesta primește date de poziționare prin interfața UART de la un modul GPS, recepționează un impuls PPS pentru sincronizarea timpului la nivel de microsecunda și controlează declanșarea DSLR-ului printr-un semnal electric dedicat. PC-ul care va stoca și procesa imaginile comunică cu ESP32 prin Bluetooth, trimițând comenzi simple precum „L”, „T” sau „X”. În funcție de acestea, ESP32 fie transmite date de localizare, fie configurează timpul de expunere, fie inițiază declanșarea camerei foto.

Modulul GPS furnizează ESP32 atât informațiile de poziționare geografică, cât și timpul absolut necesar pentru generarea timestamp-ului. Transmisia se realizează prin interfața UART folosind propoziții NMEA, iar sincronizarea fină la nivel de microsecundă este realizată prin semnalul PPS (Pulse Per Second), generat cu precizie atomică. ESP32 va parsa șirurile de caractere NMEA, analizând două tipuri de mesaje: \$GNGGA (sau \$GPGLL) pentru poziție și altitudine, respectiv \$GNRMC (sau \$GPRMC) pentru timp și dată.

Semnalul PPS (Pulse Per Second) este elementul care conferă sistemului precizie sub-milisecundă. La fiecare secundă, modulul GPS emite un impuls digital perfect sincronizat cu ceasul atomic, indicând începutul exact al unei noi secunde UTC. La recepția acestui impuls, ESP32 execută o rutină de întrerupere care salvează instantaneu valorile millis() și micros(). Acest marcaj temporal reprezintă poziția zero a secunde curente, oferind un reper hardware extrem de stabil și precis.

La finalul expunerii camerei DSLR, sistemul determină câte milisecunde și microsecunde au trecut de la impulsul PPS până la momentul încheierii expunerii. Acest offset intra-secundă este apoi combinat cu timpul absolut (HH:MM:SS) extras din propoziția RMC, rezultând timestampul complet al fotografiei: timp GPS (din RMC) + offset în interiorul secunde (din PPS). Prin această combinație, secunda UTC este furnizată de mesajele RMC, iar poziția exactă în interiorul secunde este măsurată direct de hardware cu precizie de ordinul microsecundelor.

Momentul expunerii reale a camerei este citit prin intermediul unui adaptor de tip Hot Shoe, destinat în mod normal montării unui flash extern. Acest tip de adaptor este compatibil și cu camerele DSLR clasice, și cu camerele noi de tip mirrorless. În timpul expunerii, semnalul citit prin intermediul adaptorului va indica un contact între masă și electrodul central, iar în afara

timpului de expunere contactul va fi întrerupt. Preluarea acestui semnal, prin intermediul unui pin digital echipat cu o rezistență pull-up, va duce la citirea unui semnal zero logic pe timpul expunerii, și 1 logic în rest. Sistemul măsoară momentul de început, și durata acestui semnal. Tranziția de la 1 la 0 marchează începutul expunerii și este detectată printr-o întrerupere hardware configurată pe frontul descendent. În această ISR, ESP32 capturează valoarea curentă a funcției micros(). Finalul expunerii este detectat ca momentul revenirii semnalului la valoarea 1. Diferența dintre acest moment și timpul memorat în ISR reprezintă durata exactă a expunerii în microsecunde, ulterior convertită în milisecunde.

Generarea timestampului se face la finalul unei expunerii. La finalul expunerii, sistemul utilizează timpul absolut din mesajul RMC (hh:mm:ss) împreună cu offsetul în microsecunde calculat față de ultimul impuls PPS, construind numele complet al fișierului capturat în forma: IMG_YYYY_MM_DD_HH_MM_SS_mmm_uuu_xEXPOSURE.jpg.

Analiza preciziei spațiale și temporale a receptoarelor GNSS

Este crucial ca stațiile să fie echipate cu module precise de marcare temporală și de determinare precisă a locației de observare, având în vedere nevoia de a corela precis observațiile cu elementele orbitale ale sateliților. Echipa a comparat două module GNSS: modulul L80 și NEO-M9N. Pentru precizia localizării, rezultatele obținute sunt ilustrate în Figura 9.



Figura 9. Testarea modulului GNSS

Pentru a testa precizia marcării temporale a imaginilor, a fost creat un banc de testare format din:

- Dispozitivul de marcare temporală.
- Camera, un Canon EOS R8.
- Un laptop, care trimite timpul de expunere dorit către dispozitivul de marcare temporală și primește imaginile de la cameră.
- Un analizor logic - DigiView DV1-100, conectat la semnalele RX și de declanșare ale dispozitivului de marcare temporală
- Un osciloscop - Tektronix TDS 2014B, conectat la cablul de sincronizare cu PC-ul camerei

Bancul de testare înregistrează și evaluează patru semnale cheie:

1. Timpul de procesare a comenzii dispozitivului de marcare temporală
2. Durata de declanșare a ieșirii dispozitivului de marcare temporală
3. Timpul de expunere al camerei
4. Durata transferului de imagine între cameră și laptop

Un script Python care rulează pe laptop trimite, pentru mai multe „capturi” predefinite, o comandă care conține timpul de expunere dorit către dispozitivul de marcare temporală. De asemenea, monitorizează funcționarea camerei, detectând atât momentul finalizării expunerii, cât și punctul în care se termină transferul corespunzător al imaginii către laptop.

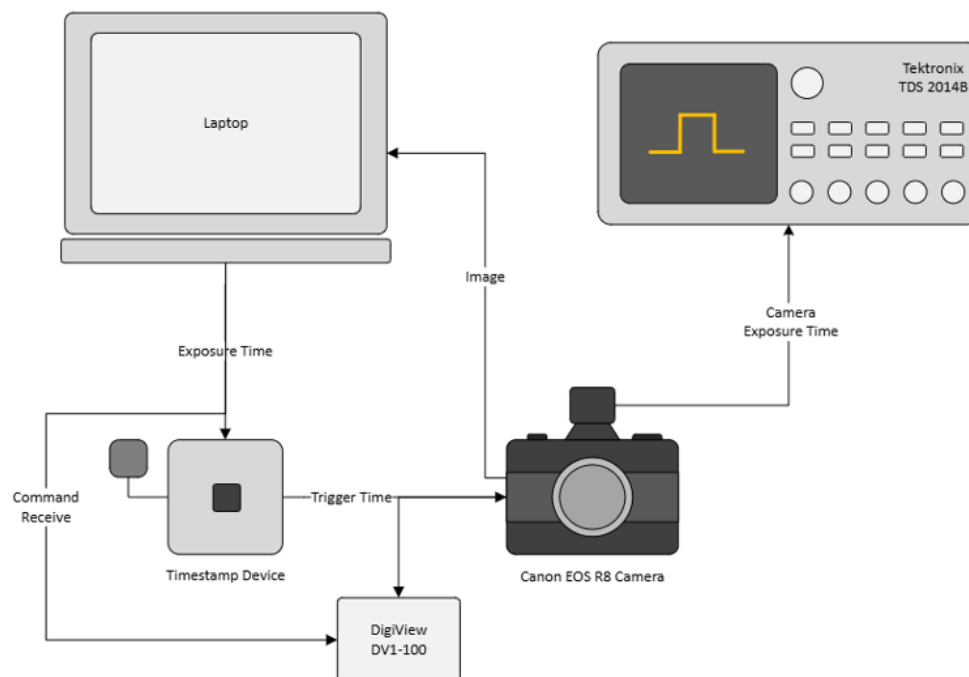


Figura 10. Diagrama configurației de test

Laptopul transmite un pachet de lungime fixă de 5 octeți către dispozitivul de marcare temporală, constând dintr-un marker de început de 1 octet (0xFF) urmat de 4 octeți

reprezentând timpul de expunere în milisecunde, codificat în format little-endian. În medie, întârzierea măsurată între recepția pachetului și momentul în care dispozitivul acționează pinul de declanșare la nivelul LOW este de 21,4488 μ s.

Din rezultatele prezentate în tabelul următor, putem observa clar că, deși o comandă de 1 secundă este trimisă dispozitivului de marcare temporală și durata de ieșire a declanșatorului este corectă, camera expune imaginea cu aproximativ 200ms mai puțin decât este așteptat. Pentru a compensa acest comportament, a fost trimisă o comandă de timp de expunere cu un decalaj suplimentar de 200 ms pentru a asigura atingerea duratei de expunere dorite. O altă soluție este modificarea configurației temporizatorului intern al STM32.

Observăm o deviație progresivă a duratei de ieșire a declanșatorului pe măsură ce timpul de expunere crește. Acest comportament sugerează că configurația temporizatorului intern al STM32 este incorectă; această problemă va fi abordată și corectată. Cu toate acestea, în ciuda unor inexactități în sincronizarea declanșatorului, timpul de expunere al camerei rămâne precis.

Command Exposure Time (s)	Trigger-output duration (s) - average -	Camera exposure time (s) - average -
1	0.9982	0.8
1.2	1.1986	1.04
3.2	3.1877	3
5.2	5.1831	8
10.2	10.1770	10

Valorile prezentate în tabelul următor reprezintă rezultatele medii obținute pe parcursul mai multor teste efectuate cu timpi de expunere diferiți. Dimensiunea imaginii și durata de transfer asociată depind de nivelul de detaliu și diversitate prezent în imaginea capturată. Pentru măsurători, fotografiile au fost realizate cu capacul obiectivului pus. Pentru fiecare setare de rezoluție, au fost capturate și analizate un total de 200 de imagini.

Image resolution (px x px)	Image size (MB) - average -	Transfer time (ms) - average -
6000x4000	0.8418	410.3254
3984x2656	0.4109	358.7621
2976x1984	0.2773	376.2912

O arhitectură alternativă pentru achiziție și procesare

O variantă alternativă la utilizarea unui PC este dezvoltarea întregului sistem pe o placă de dezvoltare embedded. În acest sens au fost făcută o implementare experimentală folosind placa de dezvoltare NVIDIA Jetson Orin Nano, care conține un procesor hexacore, o placă grafică compatibilă CUDA și TensorRT, un NVMe SSD, și o placă Wi-Fi. Placa poate prelua imagini de

la o cameră foto conectată prin USB, declanșată și sincronizată folosind pinii GPIO ai plăcii. Folosind placa de dezvoltare, modulul GPS pentru timestamping global poate fi conectat direct, fără a mai necesita un microcontroller suplimentar.

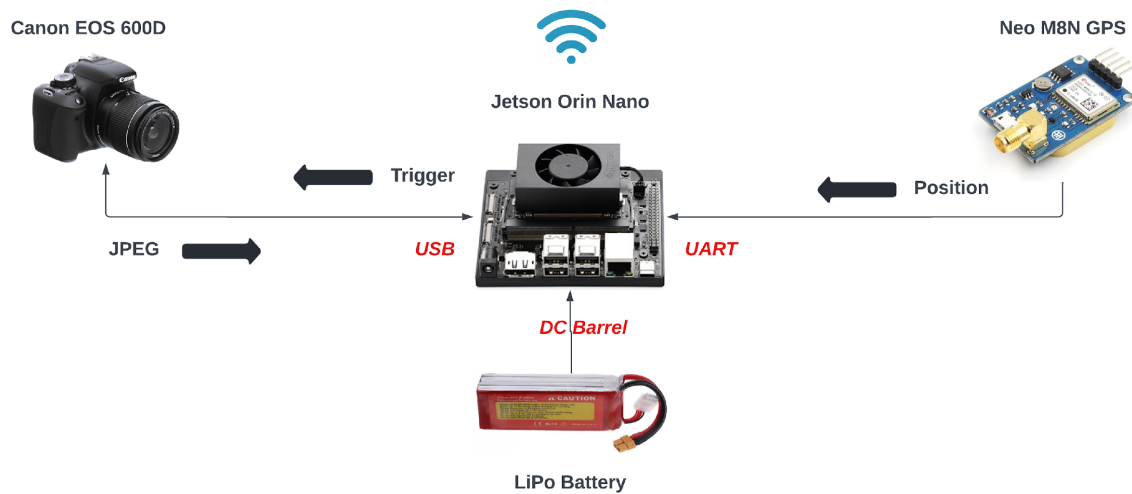


Figura 11. Arhitectura sistemului embedded

Placa de dezvoltare are un sistem de operare Linux Ubuntu 22.04 modificat, iar toolchain-ul pentru cross-compilarea aplicațiilor pentru arhitectura ARM64 + CUDA este disponibil public, împreună cu un container Docker compatibil.

Rețeaua YOLOv8 personalizată este antrenată pe o stație de lucru separată folosind framework-ul PyTorch, apoi este exportată în format ONNX (Open Neural Network Exchange) pentru compatibilitate cu TensorRT.

Aplicația executabilă principală utilizează cinci biblioteci externe, unu pentru comunicarea cu modulul GPS prin UART, unu pentru declanșarea camerei digitale prin portul USB și obținerea cadrului curent, o bibliotecă TensorRT cu backend pentru ONNX pentru încărcarea și rularea rețelei neuronale, OpenCV pentru operații generale de procesare a imaginilor, și Intel Thread Building Blocks (TBB) pentru suportul multi-threading.

Aplicația este dezvoltată pe o mașină desktop separată, cross-compilată pentru procesorul și GPU-ul ARM64 al plăcii, copiată într-un container Docker și ulterior deplasată pe placa de dezvoltare.

Flow-ul general al aplicației este prezentat în diagrama funcțională de mai jos:

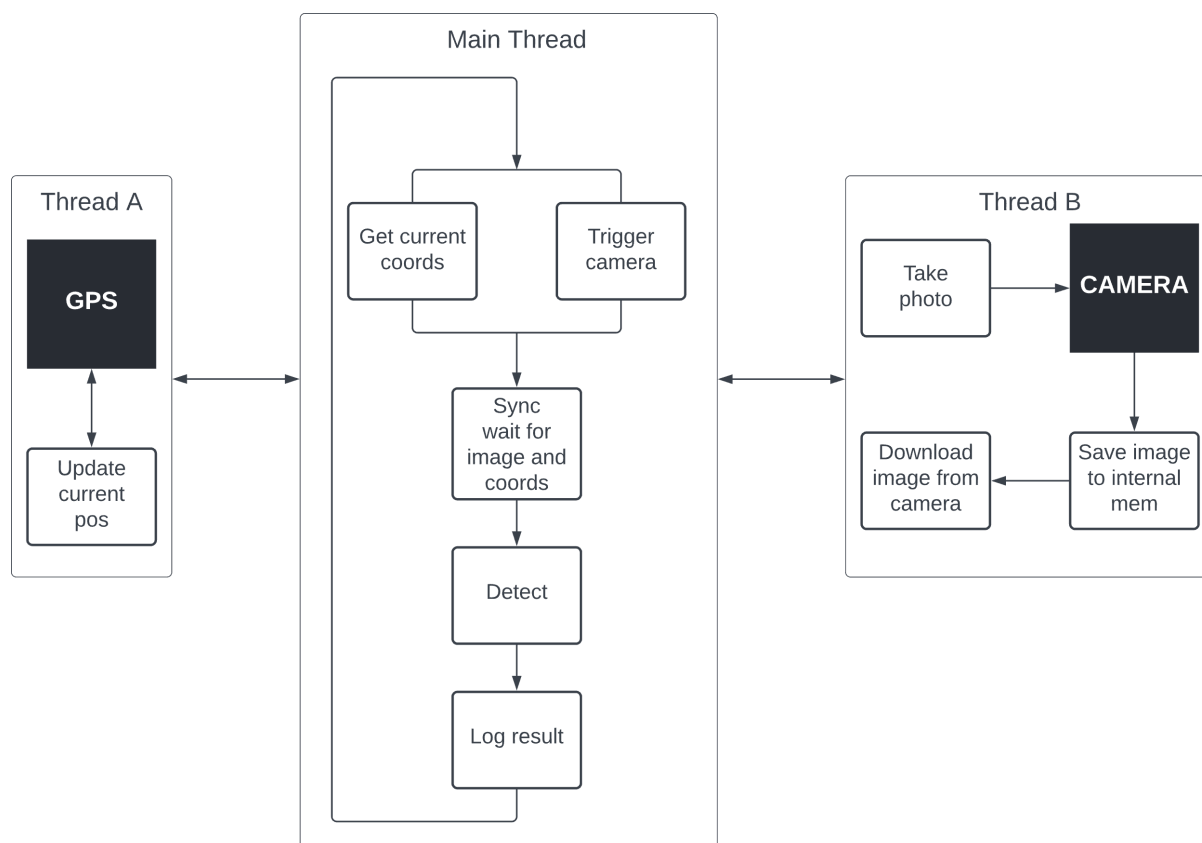


Figura 12. Fluxul de procesare pentru arhitectura alternativă.

Aplicația are două thread-uri de execuție suplimentare pe lângă thread-ul principal, unul este responsabil de actualizarea continuă a poziției curente, prin citirea și parsarea mesajelor NMEA de la modulul GPS prin UART. Celălalt, atunci când este declanșat, captează o imagine cu camera digital, o salvează pe memoria internă pe camera, apoi o descarcă de pe cameră pe sistemul de operare al plăcii de dezvoltare printr-un cablu USB.

După ce imaginea este descărcată cu succes, fișierul salvat pe memoria internă a camerei este șters, verificarea fiind efectuată prin CRC.

După ce sunt obținute coordonatele curente și cadrul curent, începe faza de detecție. Cum imaginile au o rezoluție de 6000×4000 pixeli (lățime $\times$ înălțime), acestea sunt — folosind din nou multi-threading — împărțite în ferestre de 1000×1000 pixeli, care sunt apoi procesate individual prin rețeaua YOLOv8 în format ONNX, pentru a efectua detecția obiectelor bazată pe bounding box utilizând TensorRT în precizie FP16.

Deoarece cadrul de detecție lucrează cu multipli de 32, ferestrele individuale sunt ulterior redimensionate la 992×992 pixeli.

După finalizarea detecției, rezultatele sunt înregistrate pe SSD în format text, împreună cu bounding box-urile obținute.

Deoarece Jetson Orin Nano are și placă Wi-Fi și acționează ca punct de acces, este posibilă conectarea la rețeaua sa și descărcarea coordonatelor bounding box-urilor împreună cu imaginea.

Tabelul de mai jos prezintă câteva rezultate obținute cu diferite variante ale framework-ului YOLOv8 pentru un cadru individual de 992×992 pixeli:

YOLOv8 variant	Time [ms]
Nano	28 – 30
Small	51 – 52
Medium	104 – 105

Acești timpi de execuție arată că realizarea unui sistem complet integrat pe o placă de dezvoltare Jetson Nano Orin este posibilă.

A1.6. Implementarea sistemului software pentru observarea în timp real a obiectelor în proces de reintrare - partea 1

În procesul de implementare, am identificat, în urma testărilor intermediare pe baza datelor achiziționate, că obiectele de interes pot apărea în mai multe moduri în imagine. Modul care pune cele mai puține probleme este cel în care obiectul are o mișcare aparentă mare, care duce la o dâră de lungime considerabilă, ușor de identificat. Dacă obiectul are și o luminozitate mare față de fundal (este un obiect de dimensiuni mari, sau are o suprafață puternic reflectorizantă), el va fi ușor de identificat în imagine. Dacă dâră este de dimensiuni mari, dar de contrast redus, detecția ei prin compararea cu luminozitatea fundalului, sau prin detecția diferențelor dintre cadre succesive nu mai este garantată. În această situație, o metodă bazată pe rețele neuronale, precum Yolo, dă rezultate mai bune. O a treia situație este atunci când obiectul are o mișcare aparentă redusă, datorită distanței mari față de observator. Dacă și contrastul este redus, detecția lui este imposibilă, dar dacă contrastul este acceptabil, el poate fi identificat prin analiza diferențelor dintre cadre. Cele trei situații sunt prezentate în figura 13.



Figura 13. Tipuri de ținte de observat: stânga - țintă vizibilă, cu mișcare aparentă mare, mijloc: țintă cu mișcare aparentă mare, dar contrast scăzut, dreapta: contrast mediu dar mișcare aparentă redusă.

Deoarece se dorește realizarea unui sistem capabil să detecteze cât mai multe obiecte, care mai ales în situația de reintrare pot să treacă prin toate scenariile de mai sus, soluția a fost

fuzionarea celor două tipuri de detecție: rezultatele date de rețeaua neuronală au fost combinate cu rezultatele segmentării pe baza diferențelor dintre două cadre consecutive. Combinarea celor două tipuri de rezultate se realizează la nivel de imagine de etichete:

- Se aplică metoda bazată pe rețele neuronale, și se identifică zone dreptunghiulare ce conțin un segment de dreaptă ce reprezintă un satelit în mișcare
- Se aplică o binarizare adaptivă în dreptunghiul indicat de rețeaua neuronală, și apoi se aplică transformata Hough pentru a identifica segmentul de dreaptă
- Fiecare segment de dreaptă are o etichetă unică în imaginea globală
- Pe baza diferențelor dintre cadre, se obține o imagine binară din care se extrage un nou set de obiecte etichetate
- Cele două seturi de etichete se combină pentru a genera rezultatul final.

În figura 14 este prezentat rezultatul detecției folosind doar rețelele neuronale. Secvența conține un obiect în proces de reintrare, care în acel moment era la o distanță de aproximativ 3000 km de Pământ, astfel încât mișcarea aparentă a acestuia este redusă. Obiectul este în zona centrală a imaginii, dar nu este detectat de rețeaua neuronală, datorită dimensiunii reduse a dărei.



Figura 14. Rezultatele detecției folosind doar rețele neuronale convoluționale.

În figura 15 se poate observa rezultatul combinării dintre cele două metode. În acest caz, obiectul de interes este detectat în mai multe cadre succesive, formând un tracklet în zona centrală a imaginii. În această situație combinarea celor două metode a fost esențială, deoarece în caz contrar obiectul de interes ar fi fost pierdut.

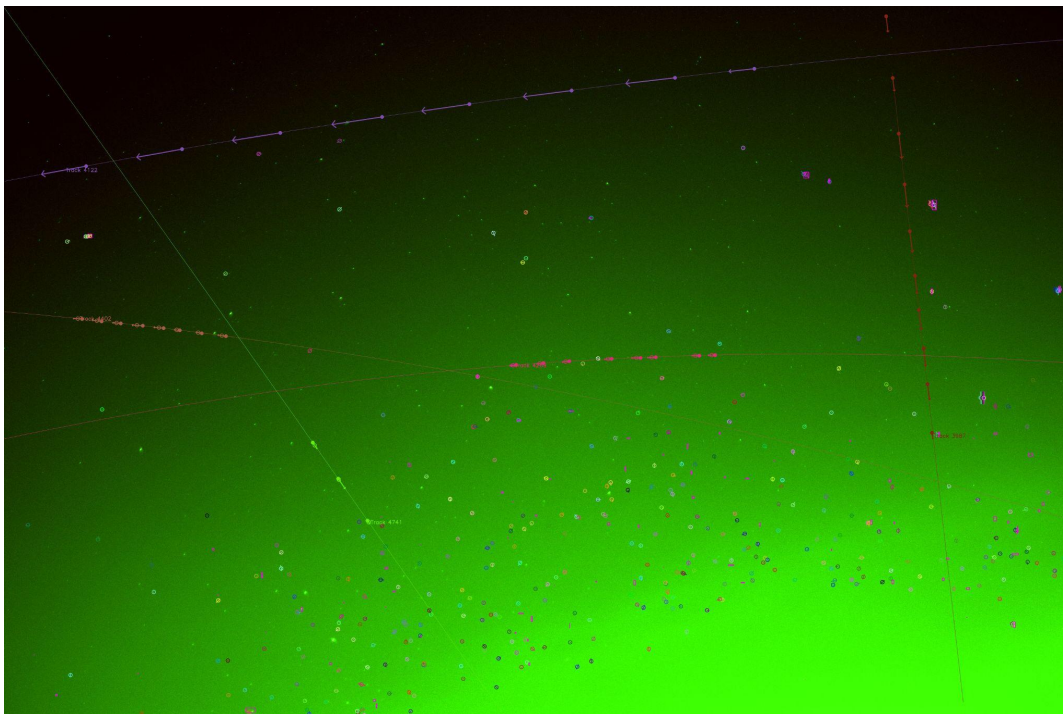


Figura 15. Rezultate folosind combinația dintre rețele neuronale convoluționale și detecția bazată pe mișcare și intensitate.

A1.7. Proiectarea și implementarea componentelor software pentru rafinarea parametrilor orbitali - partea 1

Pentru observarea unui obiect în proces de reintrare, sunt puse la dispoziție mai multe seturi de parametri orbitali (în format TLE), descărcate la diferite momente de timp de pe site-ul space-track.org. Deoarece în ultimele zile ale satelitului înainte de momentul reintrării în atmosferă se produce, la fiecare trecere, o frecare semnificativă cu atmosfera, orbita obiectului suferă modificări semnificative, și nu este garantat că cele mai noi elemente orbitale publicate sunt și cele care oferă cea mai bună predicție pentru poziția obiectului. Din acest motiv, vom prezice în timp real poziția satelitului folosind toate seturile disponibile de parametri, ca și când ar fi sateliți diferiți. Procesul este ilustrat în figura 16, unde în cadrul imagine este vizibil tracklet-ul detectat, dar și două traiectorii prezise de două seturi de parametri. Se poate observa o diferență semnificativă nu doar între cele două predicții (mai mult de 30 de grade), dar și între predicția cea mai bună și poziția reală (mai mult de 3 grade).

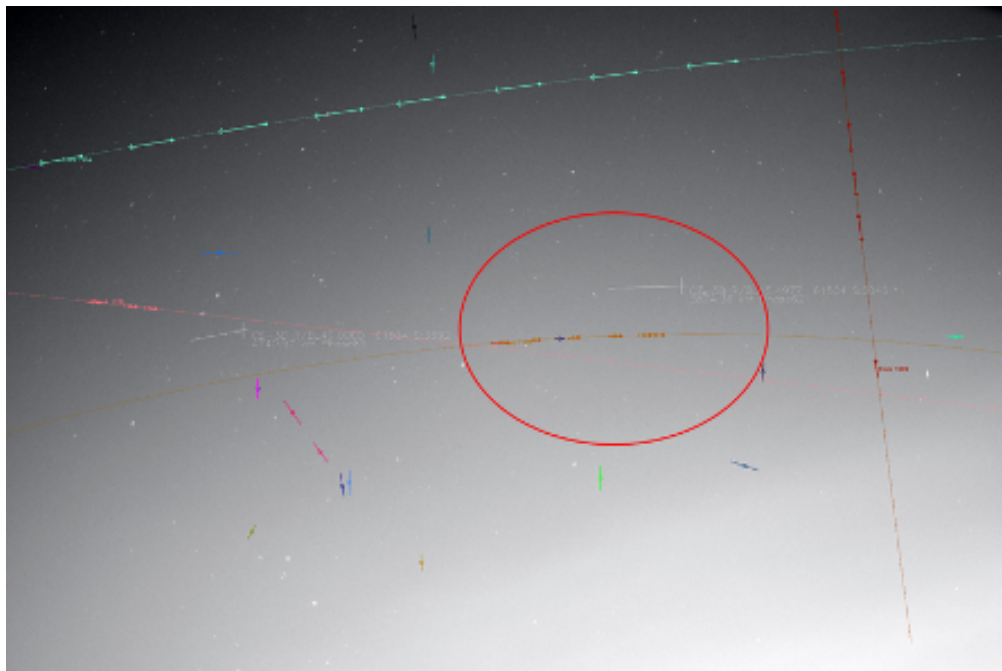


Figura 16. Obiectul detectat comparat cu două predicții date de două seturi de parametri orbitali diferiți.

În momentul în care vom obține o detecție, se va declanșa algoritmul de rafinare a parametrilor orbitali, pentru a putea să obținem o predicție mai bună. Acest prim algoritm, elaborat în această etapă, va rafina doar doi dintre parametri, Ascensia Dreaptă a Nodului Ascendent (care în imagine va afecta în principal distanța paralelă dintre traiectoria prezisă și traiectoria măsurată), și Argumentul Perigeului, care va afecta în principal poziția de-a lungul traiectoriei, cauzând o întârziere sau o accelerare față de momentul de timp la care ținta trebuie să treacă prin punctul prezis.

Algoritmul implementat are următorii pași:

- Generare de ipoteze de poziție folosind parametrii de catalog (TLE), adăugând variații aleatoare celor doi parametri ce urmează a fi rafinați.
- Pozițiile prezise sunt comparate cu măsurătorile.
- Cele mai bune N ipoteze sunt păstrate.
- Se repetă pașii de mai sus până și se reține o singură ipoteză câștigătoare.

Algoritmul este ilustrat în Figura 17.

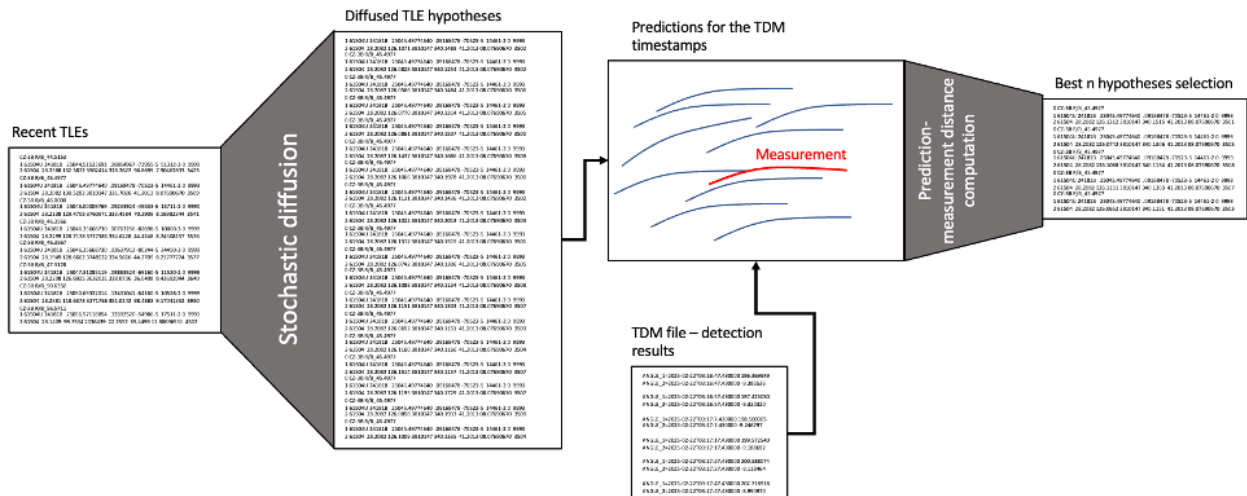


Figura 17. Ilustrarea algoritmului de rafinare a parametrilor orbitali.

Aplicând algoritmul descris mai sus, a fost obținută o rafinare a traiectoriei satelitelui urmărit astfel încât eroarea față de poziția prezisă, peste 3 minute față de observația inițială, să fie sub 0.5 grade. Această eroare, și timpul dintre prima măsurătoare și a doua, permite utilizarea în timp real a procesului de rafinare pentru a orienta un telescop de câmp vizual îngust pentru observarea precisă a obiectului.

Director proiect,
Prof. Dr. Ing. Radu Dănescu