Nadgradnja ORB-SLAM 3 z detekcijo statičnih objektov z uporabo modela YOLO

Erik Sovdat [1], **Simon Tomažič** [2], **Sašo Blažič** [3]

[1] *Fakulteta za elektrotehniko, Univerza v Ljubljani, Tržaška cesta 25, 1000 Ljubljana*
E-pošta: erik.sovdat@fe.uni-lj.si

An Upgrade of ORB-SLAM3 with YOLOv8-seg

We present an extension of ORB-SLAM3 that integrates object segmentation from a YOLOv8-seg model to enrich the map with semantic information. Segmentation is run in a separate process and the masks are passed to the SLAM system, where they are combined with depth data. Large background classes such as walls and floors are visualized as colored meshes, while static semantic objects like chairs and tables are also included in the optimization. For these objects, we add extra constraints that link their 3-D positions with camera poses and nearby points, so they become stable landmarks in the map. The result is a SLAM system that keeps the accuracy of the original ORB-SLAM3 while providing a richer map with useful semantic objects. We also discuss current limits, such as occasional segmentation errors, and outline possible improvements with faster models and sensor fusion.

1 Uvod

SLAM (Simultaneous Localization and Mapping) je družina algoritmov, ki mobilnemu sistemu omogoča, da med gibanjem hkrati ocenjuje svojo lego in gradi zemljevid okolice. Posamezne metode se razlikujejo po uporabljenih senzorjih: od navadnih RGB ali stereo kamer, globinskih kamer tipa RGB-D, laserskih skenerjev LiDAR do inercialnih merilnih enot (IMU) ali kombinacij teh senzorjev. Vsak pristop prinaša svoje prednosti in omejitve glede natančnosti, dometa, stroškov ter računske zahtevnosti.

ORB-SLAM3 je ena najbolj razširjenih odprtokodnih rešitev na področju vizualnega SLAM-a. Temelji na uporabi značilk ORB in podpira mono, stereo ter RGB-D način zajema, možno pa je tudi združevanje z IMU podatki. Sistem dosega visoko natančnost pri lokalizaciji, vendar gradi predvsem geometrijski točkovni zemljevid brez semantičnih informacij o zaznanih elementih v prostoru. V številnih aplikacijah, kot so robotska manipulacija, razširjena resničnost ali avtonomna navigacija, pa je poleg geometrije pomembno tudi razumevanje vrste in položaja statičnih objektov.

V tem prispevku predstavljamo nadgradnjo sistema ORB-SLAM3 s semantično segmentacijo objektov na osnovi modela YOLOv8-seg. Segmentacija se izvaja v ločenem procesu, rezultati pa se skupaj z globinsko informacijo uporabijo za tvorbo 3-D površin, ki so neposredno poravnane z zemljevidom SLAM-a. Na ta način sistem poleg točkovnega oblaka gradi tudi enostavne mrežne prikaze (mesh), ki ponazarjajo razporeditev zaznanih površin v prostoru. Večje statične strukture, kot sta tla in stene, prikažemo kot obarvane mreže, ki vizualno dopolnijo geometrijski zemljevid in omogočajo lažje razumevanje scene. Izbrane semantične objekte, kot so mize in stoli, pa ne obravnavamo le vizualno: njihovo lego pretvorimo v 3-D predstavitev in vključimo v optimizacijski graf. S tem postanejo dodatna stabilna sidra, ki pomagajo ohranjati konsistenco in robustnost zemljevida. Predlagani pristop združuje prednosti klasičnega geometrijskega SLAM-a in sodobnih metod globokega učenja, kar omogoča bogatejši in bolj uporaben 3-D prikaz okolja. Pri tem natančnost lokalizacije ostane primerljiva z izvirnim ORB-SLAM3, vendar pa se zaradi dodatne obdelave poveča časovna zahtevnost sistema.

2 Sorodna dela

Prvi semantični SLAM sistemi so razširili klasični SLAM tako, da so na posamezne slikovne okvirje nanesli izhode globokih konvolucijskih mrež in rezultate nato povezali s 3-D točkami zemljevida. Eden prvih takšnih pristopov, SemanticFusion [8], je združil ElasticFusion s semantičnim segmentatorjem in omogočil obarvan, semantično označen volumetrični zemljevid. Zaradi počasne klasifikacije pa sistem ni bil uporaben v realnem času. Kasneje so raziskovalci hitrost izboljšali z detektorji YOLO: Chen in sodelavci so v YOLO-SLAM povezali YOLOv3 z ORB-SLAM2 in dosegli okoli 15 FPS [9], a njihova rešitev temelji na 2-D omejitvenih okvirjih in ne uporablja globinskih podatkov. Cong in sodelavci so v YDD-SLAM uporabili YOLOv5 skupaj z globinskim filtrom za odstranjevanje gibajočih se točk, kar je povečalo robustnost v notranjih prostorih, vendar sistem še ni dodajal trajnih semantičnih vozlišč v optimizacijski graf [10].

2.1 Uporaba segmentacijskih mask v SLAM-u

Poleg detekcij na osnovi okvirjev so številni pristopi raziskali tudi uporabo segmentacijskih mask. DS-SLAM [11] uporablja semantično segmentacijo za identifikacijo in odstranjevanje premikajočih se objektov (npr. ljudi), s čimer izboljša natančnost lokalizacije v dinamičnih scenah. Podoben pristop uporablja tudi Detect-SLAM, kjer so segmentirane maske gibljivih objektov izločene, tako

da v optimizacijo vstopajo le stabilne točke okolja. Drugi sistemi, kot je MaskFusion [12], združujejo segmentacijo z metodo gostega sledenja in omogočajo gradnjo 3-D modelov posameznih objektov. Segmentacijske maske so bile uporabljene tudi za obogatitev zemljevidov z razredi površin, kot so tla, stene in pohištvo, kar omogoča boljšo navigacijo robotov in aplikacije v razširjeni resničnosti.

2.2 Povzetek sorodnih del

Pregled literature kaže, da segmentacijske maske omogočajo več različnih izboljšav: odstranjevanje dinamičnih elementov, gradnjo gostejših ali bolj semantično označenih zemljevidov ter vključevanje objektov kot posebnih vozlišč v optimizacijski graf. Kljub napredku pa ostaja izziv v učinkoviti integraciji semantike v optimizacijski proces SLAM sistema, ne da bi pri tem bistveno poslabšali časovno zahtevnost.

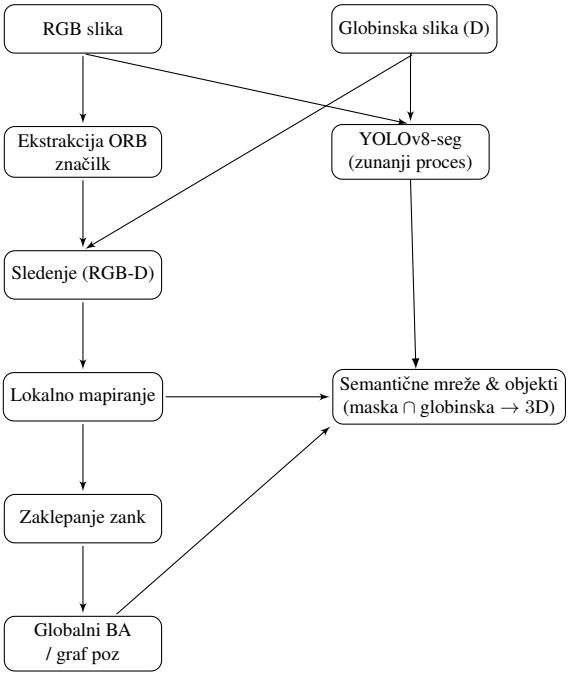
3 Tehnično ozadje ORB-SLAM3

ORB-SLAM3 [14] je eden izmed najbolj razširjenih odprtokodnih sistemov za vizualni SLAM. Temelji na nelinearnem grafu, kjer so posamezne pozicije kamer predstavljene kot vozlišča v prostoru $\mathrm{SE}(3)$. Lega kamere je minimalno parametrizirana z uporabo Liejeve algebre $\mathrm{SE}(3)$, kar omogoča učinkovito in numerično stabilno optimizacijo. Takšna zasnova omogoča, da se sistem enako dobro uporablja na različnih tipih senzorjev in v zahtevnih okoljih.

Za gradnjo zemljevida uporablja ORB značilke, ki združujejo hitro detekcijo z odpornostjo na spremembe osvetlitve, rotacije in perspektive. Na podlagi teh značilk se rekonstruira oblak 3-D točk, medtem ko bundle adjustment (BA) lokalno in globalno optimizira položaje kamer ter koordinate točk z minimizacijo reprojekcijske napake. Rezultat je geometrijsko natančen zemljevid in stabilna ocena poti kamere.

Sistem podpira tri osnovne konfiguracije kamer: mono, stereo in RGB-D. V našem delu se osredotočamo na RGB-D kamere, ki omogočajo neposreden dostop do globinske informacije. Ta lastnost je posebej pomembna pri povezovanju segmentacijskih mask z geometrijo, saj lahko semantične oznake (npr. objekti ali površine) neposredno preslikamo v 3-D prostor.

Ena ključnih prednosti ORB-SLAM3 je modularna arhitektura. Funkcionalnosti so razdeljene v jasno ločene sklope za sledenje, lokalno mapiranje, globalno optimizacijo in zaklepanje zank. Zaradi tega je sistem enostavno razširljiv in prilagodljiv novim senzorjem ali dodatnim algoritmom. V našem primeru smo modularnost izkoristili za integracijo semantične segmentacije YOLOv8-seg. Rezultati segmentacije se povežejo z globinskimi podatki ter se uporabijo za tvorbo dodatnih površin v zemljevidu in za vnos stabilnih semantičnih objektov v optimizacijski graf. Tako ohranimo natančnost osnovnega sistema, hkrati pa zemljevid obogatimo z informacijami, ki so uporabne za nadaljnje naloge, kot so navigacija ali interakcija z okoljem.



Slika 1: Potek podatkov v našem sistemu: RGB-D ORB-SLAM3 z zunanjo YOLOv8-seg vejo. Segmentacijske maske se združijo z globinskimi podatki za tvorbo 3-D mrež (tla, stene) ter za optimizacijo izbranih objektov (npr. mize, stoli).

3.1 Možnosti vključevanja semantičnih objektov v optimizacijo

V našem sistemu semantični objekti niso zgolj vizualno prikazani, temveč so vezani v optimizacijski graf, da ostanejo konsistentni z geometrijo in trajnostno prisotni v zemljevidu. Spodaj navajamo več možnih načinov oblikovanja optimizacijskih robov.

Reprojekcijski robovi Osnovni pristop temelji na projekciji središča objekta O_k v opazovani ključni okvir i:

$$\mathbf{e}_{ik}^{\mathrm{sem}} = \mathbf{u}_{ik}^{\mathrm{obj}} - \pi(\mathbf{T}_{iw}\,\mathbf{c}_k)\,. \tag{1}$$

kjer $\mathbf{u}_{ik}^{\mathrm{obj}}$ označuje središče maske v sliki, $\mathbf{T}_{iw}$ pa pozo kamere. Ta rob skrbi, da semantični objekti ostanejo poravnani z zaznanimi maskami.

Geometrijski rob $E_{o\text{-}p}$ Za vsako ORB-točko $\mathbf{p}_j$, katere projekcija pade v masko objekta k, dodamo vektorski ostanek

$$\mathbf{e}_{jk}^{\mathrm{geom}} = \mathbf{p}_j - \mathbf{o}_k. \tag{2}$$

kjer je $\mathbf{o}_k$ 3-D geometrijsko središče objekta. Utežimo ga z $\mathbf{\Psi}_k = w_k\,\mathbf{I}$, pri čemer je w_k obratno sorazmeren z razpršenostjo globin v maski.

Ploskovni robovi Za večje površine, kot so stene ali tla, lahko masko približamo kot ploskev $\pi : \mathbf{n}^\top\mathbf{x} + d = 0$. Rob meri skladnost točk znotraj maske s ploskvijo:

$$\mathbf{e}_j^{\mathrm{plane}} = \mathbf{n}^\top\mathbf{p}_j + d. \tag{3}$$

Robovi na osnovi mask (IoU) Namesto centroida lahko uporabimo celotno obliko maske. Definiramo napako kot pomanjkanje ujemanja med projekcijo objekta in masko YOLO:

$$\mathbf{e}_{ik}^{\text{mask}} = 1 - \text{IoU}\big(\mathcal{M}_{ik}^{\text{YOLO}},\, \pi\big(\mathbf{T}_{iw}, O_k\big)\big)\,. \tag{4}$$

Tak pristop je nelinearen, vendar bolje izkorišča obliko objekta.

Relacijski robovi med objekti Če sta v sceni prisotna dva objekta k in l, lahko uvedemo relacijo glede na tipične razdalje:

$$\mathbf{e}_{kl}^{\text{rel}} = \big\|\mathbf{o}_k - \mathbf{o}_l\big\| - \mu_{kl}. \tag{5}$$

kjer μ_{kl} predstavlja pričakovano razdaljo (npr. stol blizu mize).

Izbira robov je odvisna od vrste objektov in namena uporabe zemljevida. V našem delu uporabljamo standardno reprojekcijsko napako ORB-točk v sliko ter rob središčne razdalje za statične 3-D objekte, s katerim njihove položaje mehko sidramo v prostor.

Skupni strošek optimizacije Lokalni in globalni BA minimizirata

$$\begin{aligned} C = &\underbrace{\sum_{i,j} \rho\Big(\|\mathbf{u}_{ij} - \pi(\mathbf{T}_{iw}\mathbf{p}_j)\|^2_{\Sigma_{ij}} \Big)}_{\text{reprojekcija ORB-točk}} \\ &+ \lambda_g \underbrace{\sum_{k} \sum_{j \in \mathcal{J}(k)} \rho\Big(\|\mathbf{p}_j - \mathbf{o}_k\|^2_{\Psi_k} \Big)}_{\text{geometrijski rob } E_{o\text{-}p}} \end{aligned} \tag{6}$$

kjer je $\mathcal{J}(k)$ množica podpornih 3-D točk (njihove projekcije ležijo v maski objekta k), ρ robustna funkcija, Σ_{ij} kovariančna matrika meritve. (Tla/stene le vizualiziramo in ne prispevajo h kriterijski funkciji optimizacije.)

3.2 Življenjski cikel in združevanje objektov

Semantične statične objekte vodimo skozi čas, tla in stene so le vizualizacija. Objekt vključimo med vozlišča grafa in v optimizacijo, ko je v vsaj K_{conf} različnih sličicah zanesljivo zaznan ($s \geq \tau_{\text{conf}}$). Do takrat ostane začasna hipoteza. Težava nastopi, ko je isti fizični predmet večkrat z visoko verjetnostjo napačno razvrščen, kar lahko povzroči podvajanje ali napačno združevanje.

4 Eksperimentalni rezultati

4.1 Postavitev eksperimenta

Preizkuse smo izvedli na Intel i7, 32 GB RAM, RTX 4070 (Ubuntu 22.04) na sintetičnem naboru slik *TartanAir* (*hospital_sample P000*) [13], ki nudi RGB in globino.

4.2 Rezultati projekcije

Uporabimo presečišče RGB–D in semantike. Za semantično verjetnost $s(u, v)$ in globino $D(u, v)$ definiramo binarno masko

$$M(u,v) = \mathbf{1}\big(d_{\min} < D(u,v) < d_{\max}\big)\, \mathbf{1}\big(s(u,v) \geq \tau\big). \tag{7}$$

Točke z $M = 1$ povratno projiciramo v 3-D in v svet:

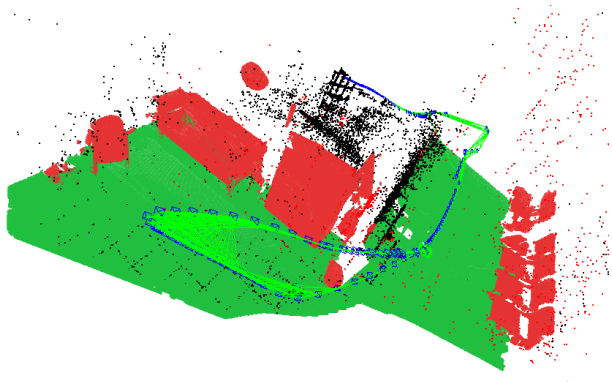
$$\mathbf{x}_c = D\,\mathbf{K}^{-1}[u, v, 1]^\top, \qquad \mathbf{x}_w = \mathbf{R}_{wc}\mathbf{x}_c + \mathbf{t}_{wc}. \tag{8}$$

Masko kratko očistimo (odpiranje/zapiranje) in podvzorčimo za tekoč prikaz v Pangolinu.

4.3 Kvalitativni rezultati

Na slikah 2 in 3 so prikazani končni 3-D zemljevidi, rekonstruirani v okolju Pangolin, iz sintetičnega nabora *TartanAir* (zaporedje *hospital_sample P000*) [13]. Modra krivulja ponazarja ocenjeno pot kamere, temne točke predstavljajo rekonstruirane ORB-točke, rdeče strukture pa statične objekte, zaznane z YOLOv8-seg modelom in projicirane v prostor. Zelena podlaga pa predstavlja zaznana tla. V notranjosti vsakega 3-D objekta so vidne ORB-točke, na podlagi katerih je bila določena povprečna globina $\bar{d}$.

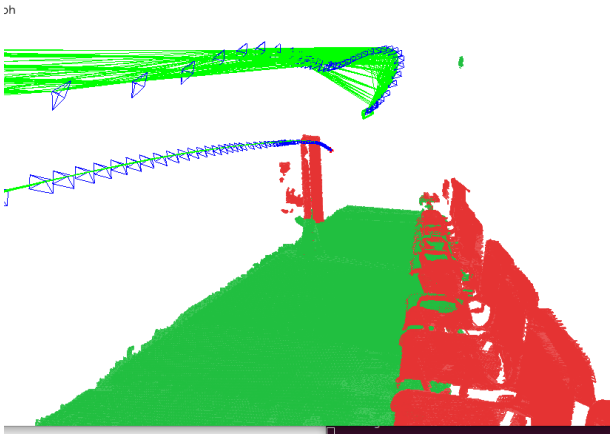
Sintetični nabor smo izbrali, ker omogoča enostavnejše in stabilnejše semantično segmentiranje (ostrejši robovi, manj šuma, konsistentne teksture in osvetlitev) ter posledično čistejše maske objektov. To omogoča zanesljivejšo projekcijo semantičnih struktur v 3-D prostor in zmanjšuje vpliv merilnega šuma na optimizacijo. Rezultati jasno kažejo, da so prepoznani objekti pravilno umeščeni med točke okolja, pri čemer vključitev semantičnih struktur ne poslabša geometrijske konsistence ocene poti kamere.



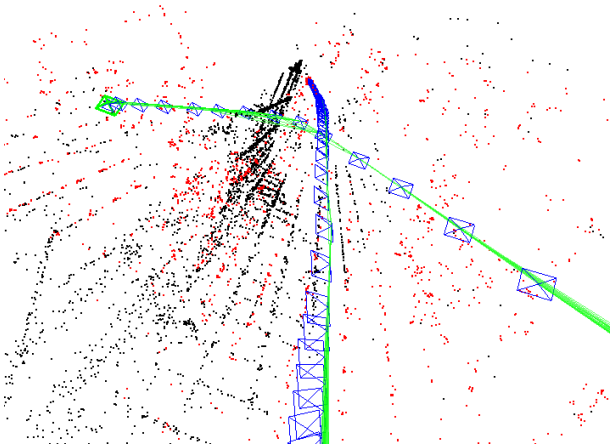
Slika 2: 3-D zemljevid z združenimi ORB-točkami (črna), zaznanimi tlemi (zelena), objekti (rdeče) in potjo kamere (modro).

5 Zaključek

Predstavili smo razširitev ORB-SLAM3, ki s presekom RGB-D in semantičnih mask zazna tla ter statične objekte in jih dosledno umešča v 3-D prostor. Statične objekte



Slika 3: Sekvenca *TartanAir_hospital_sample_P000* rekonstrukcija zaporedja z zaznanimi tlemi (zelena) in statičnimi objekti (rdeče).



Slika 4: Slam brez semantike: zemljevid samo z orb značilkami

vodimo kot vozlišča grafa in jih mehko sidramo z geometrijskim robom $E_{o\text{-}p}$, medtem ko tla in stene služijo predvsem vizualnemu in navigacijskemu kontekstu. Empirično opažamo, da takšna semantična plast ne degradira ocene poti in hkrati obogati karto z informacijami, ki so neposredno uporabne za načrtovanje poti (tla kot prehodne površine, objekti kot ovire) ter za semantično usmerjene naloge.

5.1 Praktične aplikacije

Dodana semantika omogoča tvorbo prehodnostnih zemljevidov in izogibanje oviram ter osnovo za interakcije z izbranimi objekti (npr. manipulacija). S tem presegamo zgolj geometrijsko "pust" zemljevid klasičnih SLAM algoritmov in vzpostavimo podlago za nadaljnje akcije.

5.2 Smernice za nadaljnje delo

V nadaljevanju želimo izboljšati združevanje in ponovno prepoznavanje objektov (tudi ob napačnih razredih), robove utežiti glede na negotovost semantike in globine ter pristop preveriti na realnih zaporedjih z vključenim načrtovalnikom poti v zaprti zanki.

Zahvala

Članek je nastal s finančno podporo Javne agencije za znanstvenoraziskovalno in inovacijsko dejavnost Republike Slovenije (št. P2-0219).

Literatura

[1] C. Campos, R. Elvira, J. J. Gómez Rodríguez, J. M. Montiel in J. D. Tardós, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial and Multimap SLAM," *IEEE Transactions on Robotics*, 37(6):1874–1890, 2021. :contentReference[oaicite:0]index=0

[2] J. McCormac, A. Handa, A. J. Davison in S. Leutenegger, "SemanticFusion: Dense 3-D Semantic Mapping with Convolutional Neural Networks," v *Proc. IEEE ICRA*, 2017, pp. 4628–4635. :contentReference[oaicite:1]index=1

[3] H. Zhang, J. Liu, Y. Zhou *et al.*, "Dynamic Object-Aware SLAM: YOLOv8-seg Integration and Enhanced Multi-View Geometry for Reliable Mapping in Dynamic Environments," *Sensors*, 24(6):2346, 2024. :contentReference[oaicite:3]index=3

[4] Y. Anebarassane, D. Kumar, C. A. *et al.*, "Enhancing ORB-SLAM3 with YOLO-Based Semantic Segmentation in Robotic Navigation," v *Proc. IEEE World Conference on Applied Intelligence and Computing (AIC)*, 2023, doi:10.1109/AIC57670.2023.10263892. :contentReference[oaicite:4]index=4

[5] P. Cong, J. Liu, J. Li *et al.*, "YDD-SLAM: Indoor Dynamic Visual SLAM Fusing YOLOv5 with Depth Information," *Sensors*, 23(23):9592, 2023. :contentReference[oaicite:5]index=5

[6] J. Hong, R. Choi in J. J. Leonard, "Semantic Enhancement for Object SLAM Using Foundation Models (SEO-SLAM)," arXiv:2411.06752, 2024. :contentReference[oaicite:6]index=6

[7] Q. U. Islam, F. Khozaei, S. A. Barhoumi *et al.*, "Advancing Autonomous SLAM Systems: Integrating YOLO Object Detection and Enhanced Loop Closure Techniques for Robust Environment Mapping," *Robotics and Autonomous Systems*, (online ahead of print), 2024. :contentReference[oaicite:7]index=7

[8] J. McCormac, A. Handa, A. J. Davison in S. Leutenegger, "SemanticFusion: Dense 3D Semantic Mapping with Convolutional Neural Networks," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2017.

[9] H. Anebarassane, R. Chen in X. Zhou, "YOLO-SLAM: Combining Object Detection and ORB-SLAM2 for Real-Time Performance," *Sensors*, vol. 23, no. 8, p. 4147, 2023.

[10] Y. Cong, Y. Wang, X. Yang in Z. Zhang, "YDD-SLAM: A Real-Time Semantic SLAM System with YOLOv5 and Depth Filtering," *Remote Sensing*, vol. 15, no. 14, p. 3506, 2023.

[11] C. Yu, Z. Liu, X. J. Liu, F. Xie, Y. Yang, Q. Wei in Q. Fei, "DS-SLAM: A Semantic Visual SLAM Towards Dynamic Environments," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2018.

[12] M. Runz, M. Buffier in L. Agapito, "MaskFusion: Real-Time Recognition, Tracking and Reconstruction of Multiple Objects," in *Proc. IEEE Int. Symp. on Mixed and Augmented Reality (ISMAR)*, 2018.

[13] W. Wang, D. Zhu, X. Wang, Y. Hu, Y. Qiu, C. Wang, Y. Hu, A. Kapoor, and S. Scherer, "TartanAir: A Dataset to Push the Limits of Visual SLAM," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 2020, pp. 4909–4916.

[14] C. Campos, R. Elvira, J. J. Gómez Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multi-Map SLAM," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.