

# Mozgó objektumok szegmentálása LiDAR pontfelhőkben minimális számú mérés használatával \*

Madaras Ákos<sup>1</sup>, Rózsa Zoltán<sup>1,2</sup>, Szirányi Tamás<sup>1,2</sup>

<sup>1</sup> HUN-REN Számítástechnikai és Automatizálási Kutatóintézet (HUN-REN SZTAKI), Gépi Érzékelés Kutatólaboratórium, Kende u. 13-17., 1111 Budapest

<sup>2</sup> Budapesti Műszaki és Gazdaságtudományi Egyetem, Közlekedésmérnöki és Járműmérnöki Kar, Anyagmozgatási és Logisztikai Rendszerek Tanszék, Műegyetem rkp. 3., 1111 Budapest  
madaras.akos@sztaki.hun-ren.hu

**Kivonat** A LiDAR pontfelhők gazdag információforrást jelentenek az autonóm járművek és ADAS rendszerek számára. A mozgó objektumok szegmentálása azonban kihívást jelent ezekből az adatokból. A hagyományos módszerek a környezet (globális vagy lokális) térképére támaszkodnak, amelynek rekonstrukciója és frissítése kihívást jelentő feladat valós körülmények között főleg a mozgó objektumok jelenlétében. Ez a cikk egy újszerű megközelítést javasol ami a lehető legkevesebb mérés használatával végzi el a mozgó objektum szegmentációt (Moving Object Segmentation - MOS) a LiDAR pontfelhőkben, így lehetővé téve a számítási terhelés csökkentését és a térkép nélküli feldolgozást. Megközelítésünk multimodális tanulási modellel alapul, egy modálitást hasznosító predikcióval. A használt modell a LiDAR pontfelhők és a kapcsolódó kameraképek adatkészletén tanítottuk, így a modell megtanulja a két modalitás jellemzőinek társítását, lehetővé téve a dinamikus objektumok előrejelzését térkép és kamera modalitás hiányában is. Ezenkívül, javasoljuk a szemantikai információ használatát a több mérést felhasználó szegmentációhoz a teljesítménymutatók javítása érdekében. A SemanticKITTI és az Apollo valós autonóm vezetési adatkészleteken értékeltük ki a megközelítésünket. Eredményeink azt mutatják, hogy a módszer a legkorszerűbb teljesítményt képes elérni a mozgó objektumok szegmentálásakor, és teszi ezt csak néhány (akár mindösszesen egy) LiDAR mérést felhasználva. Az implementáció példákkal és előre betanított hálózatokkal elérhető a következő linken:

<https://github.com/madak88/2DPASS-MOS>

## 1. Bevezetés

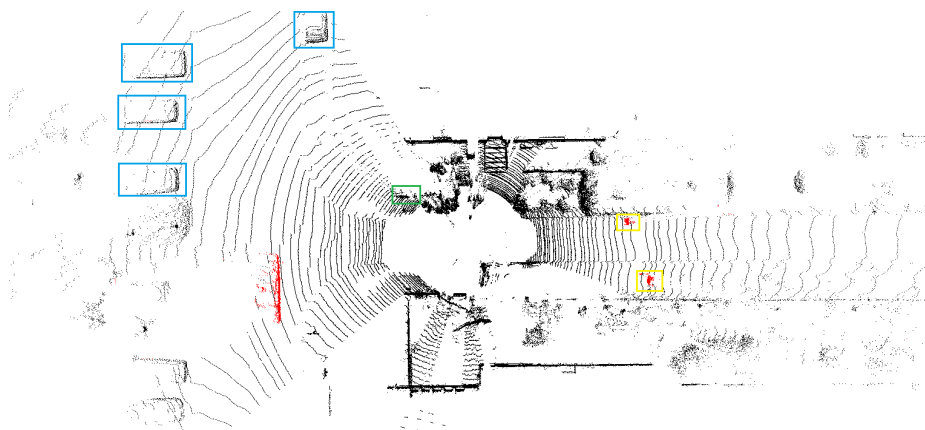
A mozgó objektumok szegmentálása számos alkalmazásban kritikus fontosságú, beleértve az autonóm vezetési, a robotikai és a felügyeleti rendszereket. [1] [2] Az autonóm vezetésben általában a dinamikus objektumok szegmentálásán, főként gyalogosok, kerékpárosok, autók és más járművek pontszintű címkézését értik. A kinyert információt az ütközések elkerülésére és a biztonságos távolságok betartására használják fel.

\* A cikk angol nyelvű változata az Open Journal of Signal Processing folyóiratban került elfogadásra "Efficient Moving Object Segmentation in LiDAR Point Clouds Using Minimal Number of Sweeps" címen.

A korábbi megközelítések egyik hátránya a számítási hatékonyság hiánya, ami a több pontfelhő (amiből egy kb. 0,5-1 millió pont) egyidejű feldolgozásából adódik. Másik hátrányuk, hogy előre meghatározott osztályokra támaszkodnak. A mozgó objektumok (pl. állatok) korábban nem látott osztályainak figyelmen kívül hagyása komoly veszélyt jelenthet. A mi javaslatunk a szükséges mérések számát (és így a számítási igényt) tekintve hatékony, és a szemantikai információt a kategóriák korlátozása nélkül is hasznosítani tudja, mivel a példányszintű szegmentálást csak finomítási lépésként alkalmazza.

A MOS megvalósítható különböző szenzorokkal, például kamerákkal [3], LIDAR-okkal [4], vagy szenzorfüziónal [5]. Az egyes modalitások teljesítményét maximalizálni szükséges az együttes teljesítmény maximalizálása és a redundancia érdekében. Munkánkban a predikció a LIDAR-modalitással történik, de a tanítás során egy másik szenzor (kamera) is felhasználásra kerül a teljesítmény növelése érdekében.

A MOS alapvető fontosságú téma. Egyrészt mivel jelentősen javíthatja a biztonságot azáltal, hogy felhívja a figyelmet a veszélyes tárgyakra, amelyeket figyelembe kell venni a pályatervezés során, illetve a körülöttünk lévő környezet pontosabb megismerésével a helyzetfelismerés is javítható.



(a) Becsült mozgó objektumpontok (pirossal jelölve) a LIDAR pontfelhőben



(b) A kamera (korábbi) képkockája, a LIDAR felülnézet bal oldali utcaképnék il-  
lusztrálásához (c) A kamera (aktuális) képkockája, a LIDAR felülnézet jobb oldali utcaképnék il-  
lusztrálására.

1. ábra. Példa az egyetlen szkennelésen alapuló mozgó objektum szegmentálásra. (A referenciaobjektumok kékkel - parkoló autók, zölddel - parkoló kerékpár, sárgával - gyalogosok - keretezve.)

Másrészt azért is rendkívül fontos, mivel az intelligens közlekedési rendszerek más összetevőinek teljesítményét növeli ha a mozgó objektumokat előfeldolgozásként meghatározzuk. A szimultán lokalizáció és térképezést (Simultaneous Localization and Mapping - SLAM) általában jelentősen megnehezíti a mozgó objektumok jelenléte. [6] Ez azzal magyarázható, hogy az egymást követő képkockákból történő lokális térképépítésben (regisztrációs problémában) zajként (outlierként) jelentkeznek a mozgó pontok [7].

Más LiDAR MOS módszerek lokalizációt és lokális (körülbelül 10 egymást követő mérést használó) [8] vagy globális térképeket [9] igényelnek a magas teljesítmény érdekében. Ezért, ha ezeket előfeldolgozó lépésként alkalmaznánk egy SLAM algoritmushoz, az nem lenne hatékony.

Ezen kívül a lehető legkevesebb információ (néhány pontfelhő) felhasználása és a döntési információk minél gyorsabb biztosítása [10] [11] fontos célkitűzés a számítási hatékonyság és a valós idejű működés szempontjából. Bizonyítani fogjuk, hogy mindössze néhány LiDAR pontfelhő (akár kettő vagy akár egy, térképépítés vagy akár helyzetek használata nélkül) is elegendő lehet a nagy pontosságú mozgó objektum szegmentálás elvégzésére. A példát az 1. ábra szemlélteti. Egy mérés alapú MOS (közvetlen mozgási jellemzők nélkül) is tanítható a környezet értelmezésére (hasonlóan az emberi környezetészleléshez), ennek magyarázata például a következőkben keresendő:

- A legtöbb esetben a mozgó vagy álló gyalogosok végtagjai különböző helyzetben vannak (1a és 1c ábra);
- A biciklik, ha kerékpárosok vannak rajtuk, akkor mozgásban vannak, nélkülük (1a ábra és 1b ábra) pedig állnak;
- Az úton lévő járművek általában mozgó objektumok, míg a parkoló területen lévők statikusak (1a ábra és 1b ábra).

Emellett a pontfelhő statikus és dinamikus részének torzítása is megfelelhet valamilyen (ego és ego-hoz viszonyított) sebességinformációnak.

### 1.1. Hozzájárulások

A cikk az alábbiakhoz járul hozzá:

- Bebizonyítottuk, hogy csak néhány szkennelés (akár egy is) elegendő a hatékony mozgó objektum szegmentáláshoz LIDAR pontfelhőkön.
- Ez az első olyan munka, amely sikeresen egyesíti a kamera- és LIDAR-adatokat a tanítás során, ezzel javítva a LIDAR-alapú következtetést a LIDAR mozgó objektum szegmentációban (MOS).
- Javítjuk a state-of-the-art eredményeket egy multimodális tanulási séma alkalmazásával, amelyet több pontfelhős példány-szegmentálással bővítünk.
- Egy hatékony MOS folyamatot javasolunk, amely robosztus, ha a bemeneti pontfelhők számát csökkentjük. Kevesebb pontfelhő használatával csökkenthető a késleltetés és a következtetési idő.
- A legkorszerűbb teljesítményt érjük el mind a SemanticKITTI, mind az Apollo leggyakrabban hivatkozott LIDAR-MOS adatbázisokon.

### 1.2.. A cikk felépítése

A cikk az alábbiak szerint van felépítve: A 2. fejezet áttekinti az irodalmat a kapcsolódó témákról. A 3. fejezet részletesen ismerteti a javasolt módszert és koncepciót. A 4. szakasz bemutatja a tesztheink eredményeit, és értékeli azokat, míg a 5. szakasz további elemzést végez. Végül a 6. fejezet levonja a következtetéseket és előrevetíti a jövőbeni munkákat.

## 2.. Kapcsolódó munkák

A LIDAR-alapú mozgó objektum szegmentálás egy viszonylag új kihívás a pontfelhőfeldolgozás területén. Ennek oka, hogy a LIDAR-adatok pontonkénti annotálása rendkívül időigényes munka; az egyik első adatbázis, amely ilyen jellegű adatokat biztosított (és még mindig az egyik leggyakrabban használt), a SemanticKITTI [12] [13], amely az egyik legismertebb önvezető autós adathalmaz, a KITTI vision benchmark suite-en alapul [14]. A LiDAR MOS területén a második leggyakrabban használt adathalmaz az Apollo adathalmazon alapul [15], amelyet általában különböző algoritmusok általánosítási képességeinek demonstrálására használnak. Mi is ezeket az adatbázisokat használjuk az összehasonlítások során. A LIDAR MOS kutatások két kategóriába sorolhatók: offline és online módszerek.

### 2.1.. Offline MOS Módszerek

Az offline mozgó objektum szegmentálási módszerek, amelyek egy nagyobb, pontosan regisztrált pontfelhőkészletet (globális térképeket) igényelnek, főként két célt szolgálnak. Vagy statikus térképek szűrésére használják őket [16] [17] [18] [19] [20], vagy címkéket generálnak az online módszerek számára [9] [21]. Különböző megközelítések léteznek ezek között a keretrendszerek között, de legtöbbjük foglaltsági rácsokra építi a megoldását, kiterjesztett talajszegmentálással és látótér alapú számításokkal. Nevükből adódóan ezek a módszerek nem alkalmasak az autonóm vezetés során történő döntéshozatalra.

### 2.2.. Online MOS Módszerek

A SemanticKITTI adatbázis eredménylistája szerint a legjobban teljesítő publikált LIDAR mozgó objektum szegmentálási módszerek a következők: [8], [22], [23], [24], [25] és [26]. A [8] szerzői voltak az elsők, akik különböző szemantikai szegmentációs hálózatokkal residual képeket (residual images) alkalmaztak a range képeken (range images) a mozgó objektum szegmentálási probléma megoldására. Ez a megközelítés sikeresnek bizonyult, és mások is átvették, például [23]. A [24] szerzői szintén range képeket (range images) használtak hálózatuk bemeneteként, és egy szemantikai és egy mozgási háló kombinációjával érték el a legjobb teljesítményt. A szemantikát előzetes információként alkalmazták, korlátozva az észlelhető mozgó objektum osztályokat, ami jelentős hátrányt jelent. A [22] - a korábban használt range képek (range images) helyett - a LIDAR szkenneléseket voxelizált ritka 4D pontfelhőkké alakította, és 4D konvolúciókat alkalmazott modelljében a mozgó objektumok becslésére. A [25] munkája

hasonlít a miénkhez abban az értelemben, hogy ők is kihasználják a példányinformációkat. Azonban a mi objektumpéldány-becslésünk eltérő módon működik (részletesen kifejtve a 3.4. szakaszban), ami nagyobb pontosságot eredményez a MOS feladatban. Jelenleg a legjobban teljesítő modellt a publikált MOS kutatások között a Semantic-KITTI adatbázison a [26] által javasolt modell, amely residual térképeket (residual maps) alkalmaz a mozgási jellemzők ábrázolására. Ennek az ábrázolásnak a létrehozása rendkívül számításigényesnek bizonyult. Az általunk javasolt módszer ezt teljesítményromlás nélkül felülmúlja ilyen számítási teher nélkül.

A fenti módszerek közös jellemzője, hogy mindegyikük legalább 6 egymást követő pontfelhőt és/vagy pozíciókat igényel a magas teljesítményű MOS eléréséhez.

Az itt ismertetett online módszerekhez képest a cikkben bemutatott módszer a következő előnyöket kínálja:

- Az általunk javasolt módszer valós időben működik;
- kamera és LIDAR fúzióját alkalmazza a tanulás során, hogy maximalizálja a LIDAR pontfelhők következtetési teljesítményét;
- nem igényel előre meghatározott objektumkategóriákat;
- csupán egy vagy két pontfelhő elegendő a magas teljesítményű becsléshez;
- az egy pontfelhőn alapuló becslésünk nem igényel pozícióinformációt;
- és ami a legfontosabb, a legjobb eredményeket éri el a KITTI és az Apollo adatbázisokon.

### 3.. A javasolt módszer

Ebben a részben részletesen bemutatjuk a javasolt megoldás négy lépését.

1. Előfeltételek
2. Kamera és LIDAR egyesítése a tanítási folyamat során
3. Következtetés a LIDAR adatokon
4. A becslések javítása szemantikai alapú példány-szegmentálással

Ezeket a lépéseket a következő alfejezetek ismertetik.

#### 3.1.. Előfeltételek

Mielőtt a keretrendszer futtatása megkezdődik, előzetesen végre kell hajtani a kamera belső kalibrálását [27] és a LIDAR-kamera kalibrációt [28]. A LIDAR koordináta-rendszerből a kamera koordináta-rendszerébe történő transzformációs mátrixot  $T_{L,C}$ -ként, a belső mátrixot pedig  $K$ -ként jelöljük a továbbiakban. Így a 3D pontfelhőt az alábbi egyenlettel vetíthetjük a képsíkra.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K} \cdot \mathbf{T}_{L,C} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (1)$$

ahol  $[u \quad v]$  a képsíkban lévő koordináták, és  $[X \quad Y \quad Z]$  a 3D pont koordináták.

### 3.2.. Keresztmodalitású tudásátadás (Cross-modal Knowledge Transfer)

Keretrendszerünk adaptálja a keresztmodalitású tudásátadást (Cross-modal Knowledge Transfer) a 2DPASS hálózat architektúrájával [29]. A hálózatot eredetileg szemantikai szegmentálásra tervezték és alkalmazták. Ez a módszer jelenleg az első helyen áll a SemanticKITTI ranglistán ebben a feladatban. A különböző modalitások közötti tudásdesztilláció (knowledge distillation) történik ; ez gondoskodik a modalitás-specifikus tudásról. A séma először egyesíti a képek és pontfelhők jellemzőit, majd egyirányú illesztést végez az egyesített pontfelhő és az egyesített jellemzők között. A cél a modalitás-specifikus információ megőrzése és az egyesítésből származó információ teljes körű megtartása. A megvalósítás részletei a [29] forrásban találhatók.

**Alapmodell:** MOS problémánkhoz három kategóriát definiáltunk: statikus, dinamikus és 'nem érdekes' ('do not care') osztályok. (Ez utóbbi szükséges, mivel a SemanticKITTI adathalmazban vannak 'címkézetlen' ('unlabeled') és 'kiugró' ('outlier') pontok.) A teljes hibafüggvény (loss)  $\zeta_{all}$  a Lovasz ( $\zeta_{iou}$ ) [30] és a Kereszt-entropia ( $\zeta_{acc}$ ) összege:

$$\zeta_{all} = \zeta_{iou} + \zeta_{acc} \quad (2)$$

Az előbbi lehetővé teszi a IoU (Intersection over Union) (vagy Jaccard index, lásd Eq. 5) közvetlen optimalizálását, míg az utóbbi:

$$\zeta_{acc} = - \sum_{c=1}^C w_c \log(p_c) y_c \quad (3)$$

ahol  $p$  az adott osztály előrejelzett valószínűsége,  $y$  a célérték, és  $w$  a  $c^{th}$  osztály súlya a  $C$  osztályok száma között.

Az tanítások 64 epoch-on keresztül futottak, és SGD (Stochastic Gradient Descent) optimalizálót használtunk. Az egy pontfelhős predikció esetben a batch-méret 8 volt, míg a két pontfelhős predikció esetén a batch-méret 4 volt.

Az itt leírt tanulási módszer (a tudástransfer séma adaptálása a szemantikai szegmentálási problémából a mozgó objektum szegmentálásra) biztosítja az kiindulási állapot (Alapként jelölve a 4 táblázatunkban). Ez a megközelítés, közvetlen mozgásinformáció nélkül is, már felülmúl néhány jelenlegi MOS módszert. Ennek magyarázata a megfigyelések (felsorolva az 1. szekcióban), amelyeket képekből tanultak meg.

**Két egymást követő mérés:** Az adaptálás mellett a tudásátadási sémát kiterjesztettük egy két egymást követő mérésen alapuló modellre is. Azért használunk csak két egymást követő mérést, mivel ez a legalacsonyabb számú, ami már közvetlenül figyelembe tud venni mozgásjellemzőket is.

Számítási szempontból az egy képkocka alapú előrejelzés a leghatékonyabb, mivel nem igényel LiDAR-képkockák regisztrálását, és a legkisebb számú ponttal dolgozik (ami a futási időt jelentősen befolyásolja).

A mindössze két egymást követő képkocka alkalmazása azonban még mindig hatékonynak tekinthető, mivel nincs szükség optimalizálásra, és a modell bemeneti pontjainak száma is alacsony.

Az implementációkban két egymást követő LiDAR pontfelhőt fűzünk össze, hogy egy közös koordináta-rendszerben hozzáférjünk a mozgásinformációhoz. A pontfelhők

összekapcsolása ( $P = [P_{t-N+1,t-N+1} \ P_{t,t-N+1}]$ ) a második pontfelhőnek az első pontfelhő koordináta-rendszerébe történő koordinátatranszformációja után történik:

$$P_{t,t-N+1} = \mathbf{T}_{t,t-N+1} \cdot P_{t,t} \quad (4)$$

ahol a  $P_{t,t-N+1}$  pontfelhő, az első index  $t$  pedig a felvétel időpontját jelzi. A második index  $t-N+1$  a mérés azon időpillanatára utal, amelyre a koordináta-rendszert a jármű ego-mozgásából számított  $\mathbf{T}_{t,t-N+1}$  homogén transzformációs mátrix transzformálja. Ha az ego-mozgást nem mérjük közvetlenül, akkor azt a LiDAR-adatokból regisztrációs algoritmusokkal, például a KISS-ICP [31] számolhatjuk ki. Az  $N$  értéket úgy vezettük be, hogy általános leírást kapjunk, de kísérleteinkben maximum két egymást követő képkockát használtunk,  $N = 2$  (az alapmodellben  $N = 1$ ).

### 3.3.. Predikció

Az intermodális tudásátadás a tanítási fázisban történik. A következtetéshez csak LiDAR pontfelhőkre van szükség, ezzel maximalizálva az egy-modális becslés hatékonyságát. A predikció során egyetlen mérés esetén nem alkalmaztunk előfeldolgozást, két mérés esetén pedig csak koordináta-transzformációt (a 3.2. alfejezetben leírtak szerint).

---

#### 1. Algorithm Objektum szintű döntés

---

**Require:** Összefűzött pontfelhő  $P$ , cél kategória  $y$ , mozgás küszöbérték  $moving\_threshold$

**Ensure:**  $P$  a mozgó objektumokat jelző frissített pontcímkékkel

- 1:  $points \leftarrow SelectPoints(P, y)$
  - 2:  $clusters \leftarrow DetectObjectInstances(points, min\_points = 200, max\_distance = 0.5)$
  - 3: **for all**  $cluster \in clusters$  **do**
  - 4:      $moving\_points \leftarrow FilterPoints(cluster, 'moving')$
  - 5:      $moving\_ratio \leftarrow Count(moving\_points)/Count(cluster)$
  - 6:     **if**  $moving\_ratio \geq moving\_threshold$  **then**
  - 7:          $CategorizeAllPointsAsMoving(cluster)$
  - 8:     **end if**
  - 9: **end for**
- 

A [32] szemantikus szegmentálási problémára egy szavazási sémát javasolt. Ezt a teszt alatti augmentációt (később 'szavazás' vagy Test-time agumentation - 'TTA' néven hivatkozunk rá) a mi MOS-problémánkhoz is alkalmaztuk. A következtetés során a TTA a bemeneti jelenetet különböző szögekben forgatja a Z tengely körül, és átlagolja a predikciós pontszámokat. Ez szerepel az 4 táblázatban szereplő 'alap' modellben. A predikció pontossága növelhető ezzel a szavazási sémával. Alkalmazása azonban kompromisszumot jelent a teljesítmény és a futási idő között, amit alaposan meg kell fontolni. Javasoljuk, hogy az egy mérés alapú következtetésünkhöz alkalmazzuk; mivel ezzel a konfigurációval még mindig jobb futási időteljesítményt érhetünk el, mint a versenytársak. A séma hatását kétkockás következtetés esetén is megvizsgáltuk; az ezzel kapcsolatos kísérleti eredményeinket a 5. szekcióban mutatjuk be.

### 3.4.. Szemantikus információ hasznosítása

Javasoljuk, hogy az elérhető szemantikai információkat is használjuk fel. Munkánkat megelőzően az LM-net [8] és az MF-MOS [26] szemantikus információt alkalmazott a MOS-problémára. Az ő megközelítésük különbözik a miénktől, mivel ők azt ellenőrizték, hogy a megjósolt mozgó objektumok mozgathatóak-e vagy sem, és konszenzusra volt szükség ahhoz, hogy egy pont megkapja a végső mozgó címkét.

A mi keretrendszerünkben a szemantikus szegmentációs eredményekből hozunk létre objektum példányokat, és csak ezek alapján lehetséges a pontok hozzáadása a mozgó kategóriához. Ez a megközelítés összhangban van azzal a felvetésünkkel, hogy a szemantikai kategóriákat nem szabad előzetes tudásként alkalmazni, mivel ez kizárná a lehetséges mozgó objektumokat (pl. állatok).

Az InsMOS [25] szintén a MOS-problémát segítő példányok létrehozását tűzte ki célul. Az ő megközelítésük abban különbözik a mi modellünktől, hogy nem használnak közvetlen szemantikai információt, hanem egy példányfelismerő fej a modelljük része.

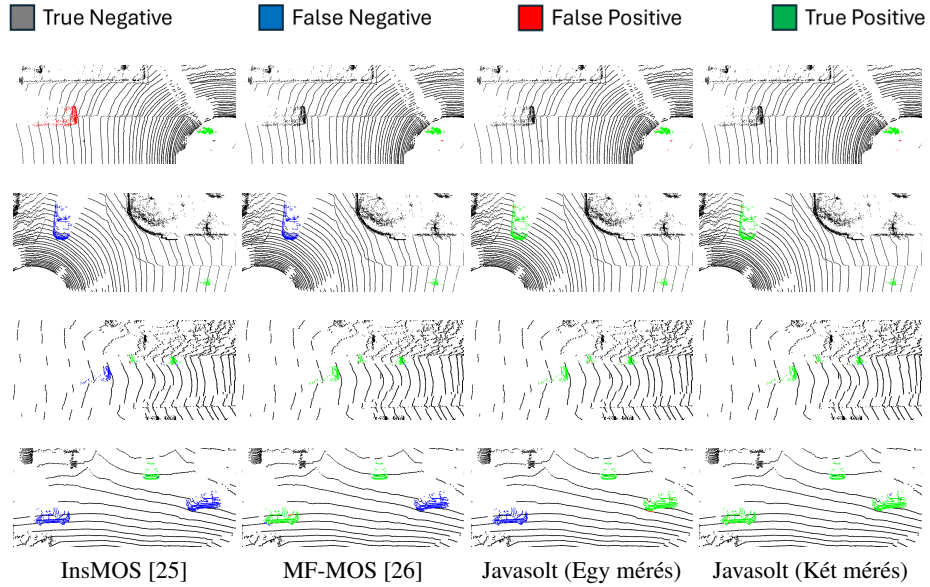
A mi megközelítésünk egyesíti e megközelítések előnyeit, és így felülmúlja őket (ahogy az a 4.1. szekcióban látható).

Javaslatunkban a szemantikus címkéket a következő lépésekben használjuk fel (Alg. 1 pszeudo kód):

1. Egy adott pontfelhőhöz (több szkennelésből egy közös koordináta-rendszerben összefűzött,  $P = [P_{t-K+1,t-K+1} \quad P_{t-K+2,t-K+1} \quad \dots \quad P_{t,t-K+1}]$ ) az adott kategóriába tartozó pontokat az előzetesen megjósolt szemantikai címkék alapján választjuk ki. Megjegyzés:  $K$  (az egyedszegmentálás során használt képkockák száma) eltérhet az  $N$  értéktől. Az eredeti 2DPASS [29] szemantikus szegmentációs hálózatot alkalmaztuk tesztjeinkben e címkék generálására.
2. A fennmaradó pontokon a tárgypéldányok szegmentálása dbscan algoritmus [33] kiterjesztésével történik (részletek Alg. 1). Megjegyzés: Az időbeli (és feltételes) dbscan kiterjesztésünk hatékony (lásd a 5 táblázatot) még növekvő  $K$  esetén is. Ennek oka, hogy az algoritmus mindösszesen a pontfelhő kis számú, az 1. lépésben kiválasztott pontjain fut).
3. Az adott képkocka minden egyes, az előző lépésekkel létrehozott objektumpéldányát megvizsgáljuk, és az objektum szintjén bináris döntést hozunk. Ha az objektum egy adott százaléknál több 'mozgó' címkejelölésű pontot tartalmaz, akkor az objektum összes pontját 'mozgó'-nak kell minősíteni. (Kísérleteinkben a küszöbérték 40 % volt.)

## 4.. Teszt eredmények

Ebben a szakaszban a teszteredményeinket mutatjuk be. A LIDAR-alapú mozgóobjektum-szegmentálásban a leggyakrabban használt benchmark a SemanticKITTI [12] [13], így ezen az adathalmazon értékeltük ki a javasolt módszert mi is. A SemanticKITTI a KITTI Odometry adathalmazból készült a LiDAR adatok pont szintű annotációjával. Az eredményeket a validálási adathalmazra (a tanítás során nem látott) közöljük. Ez 4071 LIDAR-szkennelést (kb. 500 millió pont) tartalmaz. A benchmarkban szokásos teljesítmény mérőszámot alkalmaztuk:



2. ábra. Kvalitatív példák a mozgó tárgyak előrejelzésére.

$$IoU_{MOS} = \frac{TP}{TP + FP + FN} \quad (5)$$

ahol  $FN$ ,  $TP$  és  $FP$  a mozgó pontok hamis negatív, igaz és hamis pozitív eredményeinek száma.

#### 4.1.. Számszerű eredmények

A 1 táblázat a SemanticKITTI validációs adathalmazra vonatkozó leghatékonyabb, egomozgás nélküli, egy mérésen alapú rendszerünkkel kapcsolatos eredményeket mutatja be. Itt, egy kivételével, az összes alternatív módszer szintén csak egy mérést használ a MOS-probléma megoldására. Ezek többsége a MOS feladatra áttanított szemantikus szegmentációs hálózatok. Az egyetlen kivétel a 4DMOS [22] módszer, amely napjaink egyik legjobban teljesítő MOS-megoldása. A 4DMOS ezen speciális esetét (amelyet [22] publikált) azért vettük be az összehasonlításunkba, mert ez sem használ egopozíciókat. A tanítás és a következtetés során 5 egymást követő képkockát használnak fel (különböző helyi koordinátarendszereket használva).

Látható, hogy az általunk javasolt megoldás jelentősen felülmúlja az összes többi, egy mérés alapú előrejelző hálózatot, és több, több mérés alapú hálózatot is a 2 táblázatból.

A kétlépcsős megoldásunk értékeléséhez a 2 táblázatban közöljük a versenytársak jelentett eredményeit. A 2 táblázat első részében olyan alternatívákat soroltunk fel, ahol csak két egymást követő képkockát használunk a modell bemeneteként. A tesztek során felhasznált pózok az adatbázisból származó ground truth adatok voltak. A 4DMOS rendszer teljesítménye [22] állt a legközelebb a miénkhez. A 1 táblázatban

1. táblázat. Az ego-pozíciókat nem használó módszerek teljesítményének összehasonlítása a SemanticKITTI validációs adathalmazon.

| Módszer                                     | A modell által használt input mérések száma ( $N$ ) | IoU         |
|---|---|-------------|
| MInet [34]                                  | 1   | 36.9        |
| Rangenet++ [35]                             | 1   | 39.5        |
| 4DMOS [22]                                  | 5   | 39.9        |
| LM-net [8]                                  | 1   | 51.9        |
| SalsaNext [36]                              | 1   | 53.4        |
| Proposed ( $TTA = 2, K = 1$ )               | 1   | <b>64.1</b> |
| Proposed <sup>+</sup> ( $TTA = 12, K = 1$ ) | 1   | <b>66.0</b> |

azonban látható, hogy a 4DMOS [22] nagyon rosszul teljesít a póz nélküli esetekben. Összeségében a mi keretrendszerünk teljesített a legjobban ezek közül a megoldások közül.

Felsoroltunk (a 2 táblázat második részében) más, a LIDAR MOS területén alkalmazott legkorszerűbb megoldásokat is, amelyek több mérést használnak (de még mindig nem olyan hosszú szekvenciákat, mint az offline módszerek). Hátrányuk a számítási hatékonyság hiánya, mivel több képkocka feldolgozására van szükség.

Az általunk javasolt módszer valós idejű futtatással korszerű megoldást kínál az olyan alternatívákhoz képest, amelyek nem használnak további adathalmazokat a tanításhoz. Emellett a TTA alkalmazásával a legjobb összteljesítményt érjük el további adatok felhasználása nélkül.

Emellett az alternatív módszerek nagyon érzékenyek az  $N$  értékének csökkenésére. Például az InsMOS [25] (a jelenleg közzétett második legjobb teljesítményű megoldás a SemanticKITTI tesztadatkészleten a MOS-problémában a ranglista szerint) adatokat szolgáltatott az  $N$  befolyásoló hatására vonatkozóan. A módszer a legjobb teljesítményt 10 képkocka használatával éri el. Ha azonban a képkockák számát 5-re csökkentjük a tanítás során, a [25] szerzői azt találták, hogy az IoU-értékek jelentősen visszaesnek olyan szintre, amelyet még az egy képkockás előrejelzésünk is felülmúl. A két képkockás, valós idejű becslésünk pedig jelentős különbséggel (több mint 10 %-kal) teszi ezt. A 5. szakaszban 3 ábra mutatja a versenytársak teljesítményének összehasonlítását különböző szkennelészámok esetén.

#### 4.2.. Kvalitatív eredmények

A következőkben kvalitatív példák illusztrálják a 1 és 2 táblázatban értékelt tesztjeinket. A 2 ábrán néhány tipikus példa látható, ahol a javaslat előnye összehasonlítható az alternatívákkal. Az ábra minden sora egy olyan példát mutat, a SemanticKITTI-adatkészletből a mi eredményeinket a jelenleg legjobban teljesítő másik két módszerrel

2. táblázat. Az ego-pozíciókat felhasználó módszerek teljesítményének összehasonlítása a SemanticKITTI validációs adathalmazon.

\* azt jelzi, hogy az adott megoldás a KITTI-road plusz adatszettet is felhasználta a tanításhoz.

+ azt jelzi, hogy az adott megoldás nem tekinthető valós idejűnek, mivel a tesztkonfigurációval nem érte el a kb. 5 Hz-es futási sebességet.

| Módszer  | A modell által használt input mérések száma ( $N$ ) | IoU         |
|--|---|-------------|
| LM-net [8]   | 2   | 56.0        |
| LM-net + residuals [8]                             | 2   | 59.9        |
| 4DMOS [22]   | 2   | 69.0        |
| Proposed ( $T\bar{T}A = 1, K = 2$ )                | 2   | <b>71.8</b> |
| InsMOS [25]  | 5   | 60.8        |
| LM-net + semantics [8]                             | 9   | 67.1        |
| RVMOS [24]   | 6   | 71.2        |
| Motionseg3D [23]                                   | 8   | 71.4        |
| 4DMOS [22]   | 10  | 71.9        |
| InsMOS* <sup>+</sup> [25]                          | 10  | 73.2        |
| Proposed <sup>+</sup> ( $T\bar{T}A = 12, K = 2$ )  | 2   | 74.9        |
| MF-MOS* <sup>+</sup> [26]                          | 8   | 76.1        |
| Proposed <sup>+</sup> ( $T\bar{T}A = 12, K = 10$ ) | 2   | <b>77.8</b> |

összehasonlítva. Az első sorban látható, hogy az úton parkoló autót [25] tévesen mozgónak érzékeli, míg a mi mindkét előrejelzésünk helyes. A másik három példa a többi módszer leggyakrabban előforduló hibáját mutatja: a mozgó autó pontokat statikusnak kategorizálják. Ennek oka az lehet, hogy más módszerek sokkal több mérést használnak az előrejelzésekhez, mint mi; ezért az ő modelljeik jelentős mozgással számolnak, és a lassan mozgó objektumokat statikusnak feltételezik.

## 5. Diszkusszió

Ebben a szakaszban a javasolt keretrendszerünk további vizsgálatát mutatjuk be. Először a javaslatunk általánosítási képességeit vizsgáljuk; ezután a különböző komponensek ablációs tanulmányait mutatjuk be. Végül a futási időket tárgyaljuk.

### 5.1. Általánosítási képesség elemzése

A második leggyakoribb MOS-adatbázison, az Apollo [15] adathalmazon végzett kísérletek segítségével bizonyítjuk, hogy javaslatunk jól általánosít a különböző jelenetekre. Követjük a [9] szabványos beállítását, és a teszteléshez (az általuk kiválasztott, nagy dinamikájú

képkockákat) a 2. és a 3. szekvenciát használjuk. A jelentett teljesítmény eléréséhez csak a SemanticKITTI tanító adathalmazon tanítottunk, és a keretrendszerünket az Apollo-adatkészleten értékeltük, anélkül, hogy bármilyen beállítást vagy finomhangolást módosítottunk volna.

3. táblázat. Összehasonlítás az Apollo adatszenzen.

| Módszer                       | A modell<br>által használt<br>input<br>mérések<br>száma ( $N$ ) | IoU         |
|-------------------------------|---|-------------|
| LM-net [8]                    | 9   | 16.9        |
| LM-net (fine-tuned) [8]       | 9   | 65.9        |
| MF-MOS [26]                   | 8   | 49.9        |
| MF-MOS (fine-tuned) [26]      | 8   | 70.7        |
| 4DMOS [22]                    | 10  | 73.1        |
| InsMOS [25]                   | 10  | 78.0        |
| Proposed ( $TTA = 1, K = 2$ ) | 2   | <b>80.6</b> |

A 3 táblázatban a javasolt módszer pontossága szerepel a legnagyobb teljesítményt elérő és még mindig valós időben működő paraméterkészlettel. Látható, hogy néhány korszerű alternatív módszer csak akkor nyújt elfogadható teljesítményt, ha az új adathalmazon finomhangoljuk őket. A mi rendszerünk bizonyult a leghatékonyabbnak finomhangolás nélkül, és lényegesen kevesebb szkennelést használva, mint mások.

## 5.2.. Ablációs vizsgálatok

A rendszerünk különböző összetevőinek hatásának vizsgálata érdekében beszámolunk a semanticKITTI validációs adathalmazon végzett ablációs vizsgálatról. A 4 táblázatban a pipa azt jelzi, hogy az adott komponenst (ezeket a 3. szakaszban mutatjuk be) használjuk, az x pedig azt jelenti, hogy nem.

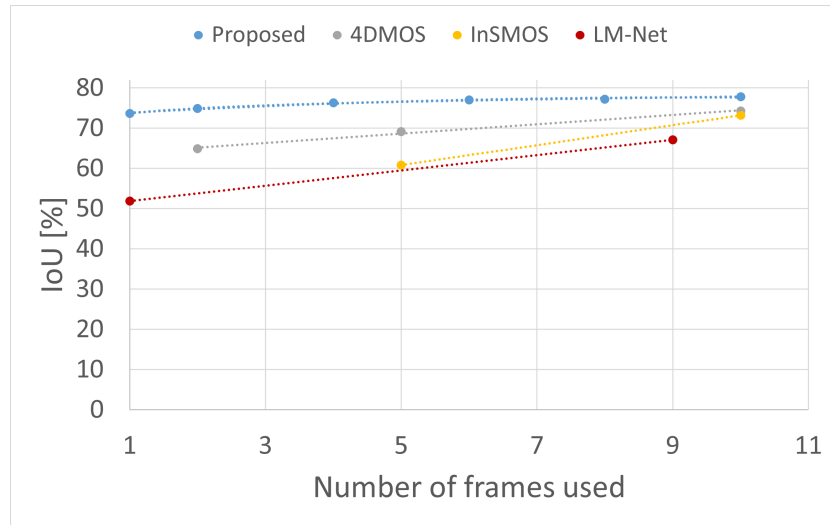
A 4 táblázatban az 'Alap' a MOS feladatra újratanított 2DPASS modellt jelenti,  $TTA = 12$  alkalmazásával. Látható, hogy minden hozzájárulásunk növelte a teljesítményt.

A 4 táblázatban  $K = 1$  és  $K = 2$  eseteket mutatjuk be. A javasolt rendszer példányszegmentációs részében használt képkockák számának ( $K$  paraméter) további vizsgálatához bemutatjuk az 3 ábrát (ahol az  $N = 2$  és  $TTA = 12$  paraméterek rögzítettek). Az ábrán az is megfigyelhető, hogy az alkalmazott mérések száma hogyan befolyásolja a többi módszert.

A többi módszer teljesítménye jelentősen csökken, ha a bemeneti képkockák száma csökken. A mi javaslatunk teljesítménye továbbra is magas marad. Az általunk mért legjobb teljesítmény ( $K = 10$  esetén) 77,8 a SemanticKITTI validációs készlet esetében.

4. táblázat. A különböző rendszerkomponensek hatása az IoU-ra a SemanticKITTI-adatkészletben.

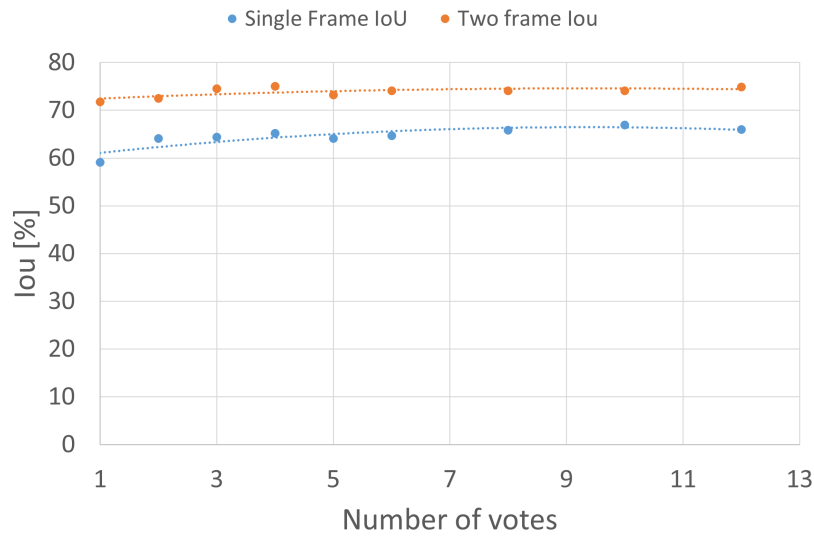
| Modell           |                                   | Szemantika                       |  | IoU  |
|------------------|-----------------------------------|----------------------------------|--|------|
| Alap ( $N = 1$ ) | Többképkockás ( $N = 2$ ) bővítés | Példány szegmentáció ( $K = 1$ ) | Többképkockás ( $K = 2$ ) példány szegmentáció |      |
| ✓                | x                                 | x                                | x  | 65.6 |
| ✓                | x                                 | ✓                                | x  | 66.0 |
| x                | ✓                                 | x                                | x  | 73.2 |
| x                | ✓                                 | ✓                                | x  | 73.6 |
| x                | ✓                                 | x                                | ✓  | 74.9 |



3. ábra. IoU a mérések különböző számának ( $K$ ) függvényében a mi példányszegmentálási eljárásunkban, összehasonlítva a IoU teljesítményének változásával a konkurens módszerek különböző számú mérések alkalmazása esetén.

Megjegyzés: Kompromisszumot kell kötni a pontosság (a bemeneti képkockák számának növelése révén) és a számítási teher között. A 3 ábrát és a 5 táblázatot megvizsgálva megfigyelhető, hogy a  $K$  érték növelése algoritmusunkban számítási szempontból hatékony. (Tesztjeinkben a  $K$  1-gyel való növelése legfeljebb 9 ms futási időnövekedést eredményezett az utófeldolgozásban).

A 4 ábrán a különböző számú tesztidő-növelés hatását is elemezzük. A TTA futási ideje egyenesen arányos a szavazatok számával. Ezért a valós időre a tesztünkhöz hasonló konfigurációnál  $N = 1$  esetén  $TTA = 2$ ,  $N = 2$  esetén pedig  $TTA = 1$  javasolt.



4. ábra. A szavazatok különböző számának (TTA) hatása a tesztidő növelésében az egy ( $N = 1, K = 1$ ) és két méréses ( $N = 2, K = 2$ ) előrejelzéseinkre.

### 5.3.. Futási idő elemzés

Itt összehasonlítjuk a futási időt az alternatívákkal.

5. táblázat. Futásidejű összehasonlítás *ms*-ban a SemanticKITTI adatbázison

| Módszer         | LM-net [8] | InsMOS [25] | MF-MOS [26] | Javasolt ( $N = 1, K = 1$ ) | Javasolt ( $N = 2, K = 2$ ) |
|-----------------|------------|-------------|-------------|-----------------------------|-----------------------------|
| Előfeldolgozás  | 72         | 0           | 514         | 0                           | 0                           |
| Pozíció becslés | 50         | 50          | 50          | 0                           | 50                          |
| Predikció       | 24         | 255         | 106         | 82                          | 136                         |
| Utófeldolgozás  | 11         | 16          | 207         | 30                          | 39                          |
| <b>Összesen</b> | <b>157</b> | <b>321</b>  | <b>877</b>  | <b>112</b>                  | <b>225</b>                  |

A 5 táblázatban az egy és két képkocka alapú előrejelzési esetünk mellett a futási idő értékeket a két jelenleg legjobban teljesítő versenytárs (InsMOS [25] és MF-MOS [26]), valamint az LM-net [8] módszerekre is közöljük, amely a SemanticKITTI MOS ranglistán jelenleg a leggyorsabb implementációt biztosítja. Tesztkonfigurációnk a következő volt: AMD Ryzen 7 6800H Radeon Graphics 3,20 GHz-es processzorral, 32 GB RAM, NVIDIA GeForce RTX 3070 GPU.

Előfeldolgozásra van szükség az LM-net és a MF-MOS esetében, ezek range és residual image-k előállítását jelentik. Az MF-MOS-nak lényegesen több ilyenre van szüksége, mint az LM-netnek, és ez a művelet számításigényes; valós idejű futtatás nem lehetséges ennél a módszernél a teszthardveren.

Pózbecslés minden módszer esetében szükséges, kivéve az egy képkocka alapú megoldásunknál, de a legtöbb versenytárs nem nyújt pózbecslési implementációt. Így a KISS-ICP [31] futási ideje lett feltüntetve a mi tesztkonfigurációnkra vonatkozóan.

Az utófeldolgozás a 4DMOS esetében a Bayes-szűrő alkalmazását jelenti finomítás az InsMOS és az MF-MOS esetében, szemantika alkalmazása az LM-net és a javaslatunk esetében. Az LM-net feltételezte, hogy a szemantikus szegmentálás párhuzamosan futhat a mozgó objektum szegmentálásával; a tisztességes összehasonlítás érdekében mi is ezt a feltételezést alkalmaztuk.

Összességében a 5 táblázatban látható, hogy az egykockás megoldásunk a leggyorsabb a bemutatott megoldások közül, és az alternatívák közül csak az LM-net gyorsabb, mint a kétkockás megoldásunk. IoU teljesítménye (lásd a 2 táblázatot) azonban nem hasonlítható össze a mi javaslatunkkal.

## 6.. Konklúzió

Újszerű keretrendszert javasoltunk LiDAR pontfelhőkben mozgó objektumok szegmentálására. A javasolt módszer a legkorszerűbb mozgóobjektum-szegmentálást végzi és a legkorszerűbb általánosítást végzi új adathalmazokra.

Bebizonyítottuk, hogy javaslatunkat az alternatíváknál kevésbé befolyásolja a bemeneti mérések száma; már egy pásztázás is elegendő a nagy teljesítményű mozgóobjektum-szegmentáláshoz. Ezt egy multimodális tanulási modell alkalmazásával, valamint LiDAR pontfelhőkön és kameraképeken történő tanítással (és csak LiDAR pontfelhőkből történő egymodális következtetéssel) értük el. A más módszerekhez képest kevesebb képkocka használata jobb számítási hatékonyságot eredményez. Megközelítésünk potenciális alkalmazási lehetőségei az autonóm járművekben és az ADAS rendszerekben jelentősek.

A jövőben a LiDAR MOS-t szeretnénk továbbfejleszteni más autonóm érzékelőkből származó tudás transzferrel.

## Köszönetnyilvánítás

A kutatómunka a Bolyai János Kutatási Ösztöndíj támogatásával készült. A munkát a Nemzeti Kutatási, Fejlesztési és Innovációs Hivatal (NKFIH) STARTING 149552, K139485 és TKP2021-NVA-01 számú pályázata támogatta. A publikációban szereplő kutatást a HUN-REN SZTAKI az Európai Unió támogatásával valósította meg, az Autonóm Rendszerek Nemzeti Laboratórium keretében. (RRF-2.3.1-21-2022-00002)

Köszönetet mondunk az HUN-REN Cloud ([37]; <https://science-cloud.hu/>) használatáért, ami hozzájárult a publikált eredmények eléréséhez.

## Hivatkozások

1. H. Lee, J. Yoon, Y. Jeong, and K. Yi, „Moving object detection and tracking based on interaction of static obstacle map and geometric model-free approach for urban autonomous driving,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3275–3284, 2021.
2. C. Jiang, D. P. Paudel, D. Fofi, Y. Fougerolle, and C. Demonceaux, „Moving object detection by 3D flow field analysis,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 1950–1963, 2021.
3. M.-N. Chapel and T. Bouwmans, „Moving objects detection with a moving camera: A comprehensive review,” *Computer Science Review*, vol. 38, p. 100310, 2020.
4. J. An, B. Choi, H. Kim, and E. Kim, „A new contour-based approach to moving object detection and tracking using a low-end 3 dimensional laser scanner,” *IEEE Transactions on Vehicular Technology*, vol. PP, pp. 1–1, 06 2019.
5. Y. Cai, B. Li, J. Zhou, H. Zhang, and Y. Cao, „Removing moving objects without registration from 3D LiDAR data using range flow coupled with imu measurements,” *Remote Sensing*, vol. 15, no. 13, 2023.
6. Z. Rozsa, M. Golarits, and T. Sziranyi, „Localization of map changes by exploiting SLAM residuals,” in *Advanced Concepts for Intelligent Vision Systems*, J. Blanc-Talon, P. Delmas, W. Philips, D. Popescu, and P. Scheunders, Eds. Cham: Springer International Publishing, 2020, pp. 312–324.
7. A. EFRAIM and J. M. Francos, „On minimizing the probability of large errors in robust point cloud registration,” *IEEE Open Journal of Signal Processing*, vol. 5, pp. 39–47, 2024.
8. X. Chen, S. Li, B. Mersch, L. Wiesmann, J. Gall, J. Behley, and C. Stachniss, „Moving Object Segmentation in 3D LiDAR Data: A Learning-based Approach Exploiting Sequential Data,” *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, pp. 6529–6536, 2021.
9. X. Chen, B. Mersch, L. Nunes, R. Marcuzzi, I. Vizzo, J. Behley, and C. Stachniss, „Automatic Labeling to Generate Training Data for Online LiDAR-Based Moving Object Segmentation,” *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 3, pp. 6107–6114, 2022.
10. Z. Rozsa and T. Sziranyi, „Object detection from a few LIDAR scanning planes,” *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 4, pp. 548–560, 2019.
11. Z. Rozsa, M. Golarits, and T. Sziranyi, „Immediate vehicle movement estimation and 3d reconstruction for mono cameras by utilizing epipolar geometry and direction prior,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 23 548–23 558, 2022.
12. J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, „SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences,” in *Proc. of the IEEE/CVF International Conf. on Computer Vision (ICCV)*, 2019.
13. J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, J. Gall, and C. Stachniss, „Towards 3D LiDAR-based semantic scene understanding of 3D point cloud sequences: The SemanticKITTI Dataset,” *The International Journal on Robotics Research*, vol. 40, no. 8-9, pp. 959–967, 2021.
14. A. Geiger, P. Lenz, and R. Urtasun, „Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite,” in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 3354–3361.
15. W. Lu, Y. Zhou, G. Wan, S. Hou, and S. Song, „L3-net: Towards learning based lidar localization for autonomous driving,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 6382–6391.
16. J. Schauer and A. Nüchter, „The peopleremover—removing dynamic objects from 3-D point cloud data by traversing a voxel occupancy grid,” *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1679–1686, 2018.

17. G. Kim and A. Kim, „Remove, then revert: Static point cloud map construction using multi-resolution range images,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 10 758–10 765.
18. M. Arora, L. Wiesmann, X. Chen, and C. Stachniss, „Mapping the static parts of dynamic scenes from 3D LiDAR point clouds exploiting ground segmentation,” in *2021 European Conference on Mobile Robots (ECMR)*, 2021, pp. 1–6.
19. H. Lim, S. Hwang, and H. Myung, „Eraser: Egocentric ratio of pseudo occupancy-based dynamic object removal for static 3D point cloud map building,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2272–2279, 2021.
20. B. Mersch, T. Guadagnino, X. Chen, I. Vizzo, J. Behley, and C. Stachniss, „Building Volumetric Beliefs for Dynamic Environments Exploiting Map-Based Moving Object Segmentation,” *IEEE Robotics and Automation Letters (RA-L)*, vol. 8, no. 8, pp. 5180–5187, 2023.
21. P. Pfreundschuh, H. F. C. Hendriks, V. Reijgwart, R. Dubé, R. Y. Siegwart, and A. Crumariuc, „Dynamic object aware LiDAR SLAM based on automatic generation of training data,” *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11 641–11 647, 2021.
22. B. Mersch, X. Chen, I. Vizzo, L. Nunes, J. Behley, and C. Stachniss, „Receding Moving Object Segmentation in 3D LiDAR Data Using Sparse 4D Convolutions,” *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 3, pp. 7503–7510, 2022.
23. J. Sun, Y. Dai, X. Zhang, J. Xu, R. Ai, W. Gu, and X. Chen, „Efficient spatial-temporal information fusion for lidar-based 3D moving object segmentation,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022.
24. J. Kim, J. Woo, and S. Im, „RVMOS: Range-view moving object segmentation leveraged by semantic and motion features,” *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8044–8051, 2022.
25. N. Wang, C. Shi, R. Guo, H. Lu, Z. Zheng, and X. Chen, „InsMOS: Instance-Aware Moving Object Segmentation in LiDAR Data,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 7598–7605.
26. J. Cheng, K. Zeng, Z. Huang, X. Tang, J. Wu, C. Zhang, X. Chen, and R. Fan, „MF-MOS: A Motion-Focused Model for Moving Object Segmentation,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.
27. Z. Zhang, „A flexible new technique for camera calibration,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.
28. Y. Hao, X. Jin, and D. Du, „Multi-dimensional geometric feature-based calibration method for lidar and camera fusion,” in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 8406–8410.
29. X. Yan, J. Gao, C. Zheng, C. Zheng, R. Zhang, S. Cui, and Z. Li, „2DPASS: 2D priors assisted semantic segmentation on lidar point clouds,” in *European Conference on Computer Vision*. Springer, 2022, pp. 677–695.
30. M. Berman, A. Triki, and M. B. Blaschko, „The Lovasz-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, jun 2018, pp. 4413–4421.
31. I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss, „KISS-ICP: In Defense of Point-to-Point ICP – Simple, Accurate, and Robust Registration If Done the Right Way,” *IEEE Robotics and Automation Letters (RA-L)*, vol. 8, no. 2, pp. 1029–1036, 2023.
32. X. Zhu, H. Zhou, T. Wang, F. Hong, Y. Ma, W. Li, H. Li, and D. Lin, „Cylindrical and asymmetrical 3d convolution networks for lidar segmentation,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9939–9948.

33. M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, „A density-based algorithm for discovering clusters in large spatial databases with noise,” in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, ser. KDD'96. AAAI Press, 1996, p. 226–231.
34. S. Li, X. Chen, Y. Liu, D. Dai, C. Stachniss, and J. Gall, „Multi-scale interaction for real-time lidar data segmentation on an embedded platform,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 738–745, 2022.
35. A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, „Rangenet ++: Fast and accurate LiDAR semantic segmentation,” *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4213–4220, 2019.
36. T. Cortinhal, G. Tzelepis, and E. Erdal Aksoy, „Salsanext: Fast, uncertainty-aware semantic segmentation of lidar point clouds,” in *Advances in Visual Computing: 15th International Symposium, ISVC 2020, San Diego, CA, USA, October 5–7, 2020, Proceedings, Part II*. Berlin, Heidelberg: Springer-Verlag, 2020, p. 207–222.
37. M. Heder, E. Rigo, D. Medgyesi, R. Lovas, S. Tenczer, F. Torok, A. Farkas, M. Emodi, J. Kadlecsek, G. Mezo, A. Pinter, and P. Kacsuk, „The past, present and future of the ELKH cloud,” *Informacios Tarsadalom*, vol. 22, no. 2, p. 128, aug 2022.