

# 3D CNN alapú MLS pontfelhő szegmentáció

Nagy Balázs<sup>1,2</sup> és Benedek Csaba<sup>1,2</sup>

<sup>1</sup> Gépi Érzékelés Kutatólaboratórium Magyar Tudományos Akadémia, Számítástechnikai és Automatizálási Kutatóintézet

Pázmány Péter Katolikus Egyetem, Információs Technológiai és Bionikai Kar

<sup>2</sup> {vezetéknév.keresztnév}@sztaki.mta.hu

**Absztrakt.** Napjainkban egyre több városról elérhetőek olyan nagy felbontású, részletes (HD) térképek, melyeket a várostervezés és az egyéb okos város koncepciók mellett az önjáró járművek is felhasználhatnak többek között navigációs célokra. Cikkünkben egy olyan 3D konvolúciós neurális hálózat (CNN) alapú pontfelhő szegmentáló módszert mutatunk be, amely egy mobil térképező rendszer (MLS) által készített sűrű pontfelhő pontjait 9 különböző szemantikai osztályba sorolja be, ami később részletes HD térképek alapjául szolgálhat. Az ajánlott módszer figyelembe veszi a városi környezet nagymértékű diverzitását, továbbá megoldást kínál a változó pontsűrűség kezelésére és a nagyszámú dinamikus objektum mozgásából keletkező zaj kiszűrésére. Továbbá egy valós idejű lokalizációs problémán keresztül bemutatjuk a 3D CNN módszer által előállított sűrű, szegmentált pontfelhő egy lehetséges alkalmazását. A módszer kiértékelését egy saját, manuálisan annotált MLS adatbázison végeztük el, továbbá három, a szakirodalomban fellelhető referencia módszerrel hasonlítottuk össze.

## 1. Bevezetés

1

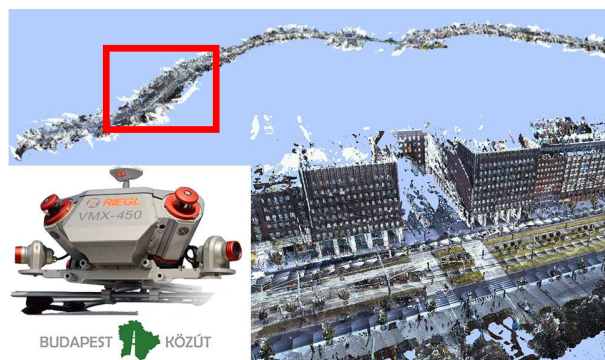
A valós idejű kinematikán alapuló GPS (RTK GPS) helymeghatározó rendszerek, nagy pontossággal képesek meghatározni egy objektum aktuális pozícióját és orientációját, azonban ezek az eszközök nagyon drágák, így csak néhány speciális területen van létjogosultságuk. Az általános GPS alapú helymeghatározó rendszerek, azonban nem elég pontosak az önjáró járművek precíz lokalizációjához és navigálásához, főleg városi környezetben, ahol sokszor a nem megfelelő erősségű GPS jel miatt a pozícionálási hiba 10 méternél is nagyobb lehet. A jármű lokalizációjának pontosítására egy lehetséges megoldás, ha a kezdeti GPS pozíció becslés után az önjáró járművek távolság mérő (Lidar, sztereó kamera) szenzor adatait egy globális geo-referált, nagy felbontású térképhez regisztrálni.

Az idő szinkronizált Lidar és navigációs szenzorokkal felszerelt mobil térképező rendszerek (MLS) rövid idő alatt képesek nagy területek részletes digitalizálására, vagyis

---

<sup>1</sup> A módszer az International Joint Conference on Neural Networks (IJCNN, 2017) angol nyelvű nemzetközi konferencián bemutatott munkán alapszik, a kibővített eredmények pedig bírálat alatt vannak az IEEE Sensors folyóiratban. A cikk a Kuba Attila Díjra pályázik.

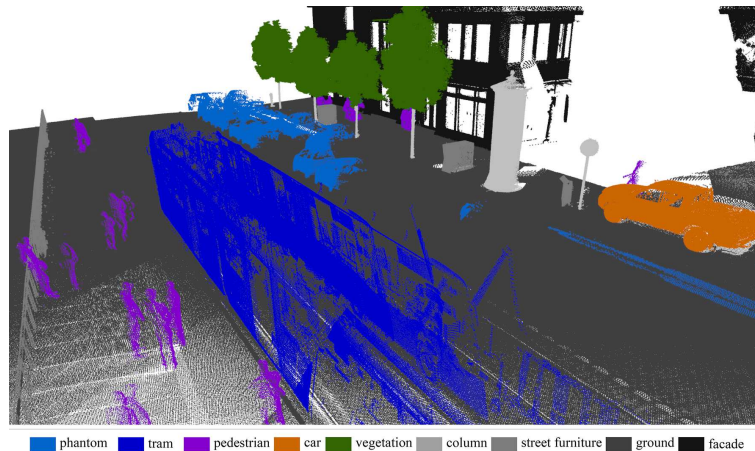
sűrű, jellemző gazdag sokszor színinformációt is tartalmazó pontfelhők előállítására (1 ábra), továbbá az egyes 3D térbeli mérések egy globális koordináta-rendszerbe való transzformálására [1–3]. Az így készült sűrű pontfelhőket felhasználhatjuk részletes 3D HD térképek előállításához, melyeket különböző útfelügyeleti és várostervezési feladatok mellett az önjáró járművek lokalizációs és navigációs célokra is felhasználhatnak. Azonban, ehhez szemantikusan címkézett pontfelhők (2 ábra) szükségesek, ami nagy kihívást jelent az automatikus címkéző algoritmusok számára.



**1. ábra:** MLS szenzor és az előállított nyers pontfelhő

Az egyik legnagyobb kihívást jelentő probléma az MLS pontfelhők automatikus feldolgozásával kapcsolatban a számos objektum mozgása által okozott *fantom* jelenség (3 ábra). Mivel a térképező rendszer a 3D pontokat egy globális koordináta-rendszerbe transzformálja, így a platformmal egyidejűleg mozgó objektumok (mozgó járművek és gyalogosok) egy *fantom* szerű elnyúlt, torzult struktúráként fognak megjelenni a globális pontfelhőben [4]. A *fantom* objektumok mellett az olyan dinamikus objektumokat is fontos megjelölni, amelyek a szkennelés pillanatában nem mozogtak (álló gyalogosok, parkoló járművek) [2], hiszen ezek sem részei a statikus háttér modelnek, továbbá jelenlétükkel gyalogos átkelőhelyekre, járdákra és parkoló helyekre utalhatnak. Az oszlopszerű objektumok (jelző táblák, fatörzsek, villanyoszlopok) [1] jó navigációs pontoknak tekinthetők, ezért fontos részét képezik a statikus háttér térképnek. A növényzet (bokrok, lombkorona) detektálása szintén szükséges [3], mivel az év során dinamikusan változik a megjelenésük és ezt a változásdetekció algoritmusoknak figyelembe kell vennie.

A fent említett komplex, többosztályos szemantikus címkézés feladat megoldására egy 3D konvolúciós neurális hálózat alapú módszert javasoltunk, amely a pontfelhő pontjait 9 szemantikai osztályba sorolja. A bemutatott módszert a SZTAKI CityMLS adatbázison teszteltük.



**2. ábra:** A bemutatott 3D CNN alapú pontfelhő szegmentációs eredmény (az adatokat a Budapest Közút Zrt. biztosította).

## 2. Szakirodalomban fellelhető módszerek

A szakirodalomban számos pontfelhő klasszifikáló és szegmentáló módszert publikáltak, azonban a legtöbb nem fektet hangsúlyt a nyers MLS adatból történő HD térkép generálás során fellépő gyakorlati problémákra, mint pl. a fantom objektumok eltávolítására. A fantomok felismerése jelentősen egyszerűsödik, ha minden egyes ponthoz hozzá tudjuk rendelni a szkennerek aktuális pozícióját és egy relatív időbélyeget (háromlábú állvány alapú szkennelés [5]), azonban a mi esetünkben az MLS rendszer egy globális koordináta-rendszerbe transzformálja a rögzített pontokat, így nem áll ilyen információ a rendelkezésünkre.

A pont alapú felismerő módszerek [6], [7] különféle statisztikai jellemzőket nyernek ki egy lokális pont szomszédságból, azonban ezek a módszerek nem veszik figyelembe a magasabb szintű strukturális jellemzőket [8], ami nagyban korlátozza a fantomok felismerését.

Számos módszer geometriai jellemzők alapján lehetséges objektum jelölteket nyer ki a pontfelhőből, majd az objektumokat alak leírók vagy mélytanuló hálózatok segítségével osztályozza [9]. A módszer előnye, hogy gyors, viszont az osztályozás minősége nagyban függ az objektumok pontos detektálásától, ami nagy kihívást jelent főleg zajos pontfelhők esetén.

A szakirodalomban több voxel alapú szegmentációs módszer [3, 10, 11] fellelhető, ahol először egy szabályos 3D voxel rácsot illesztnek a pontfelhőre, majd az egyes voxeleket, különböző jellemzők kinyerése után szemantikus kategóriákba sorolják (út, jármű, oszlop szerű objektum, növényzet, stb.). [12] előre meghatározott jellemzőket nyer ki, majd egy SVM alapú módszerrel osztályozza a pontfelhő pontjait. [13] egy random forest alapú osztályozót mutat be, ami kovariancia, pont sűrűség és strukturális információk alapján képes változó sűrűségű MLS adatokat szegmentálni. Ezen módszerek



(a) Fantom objektumokat tartalmazó nyers MLS pontfelhő (b) Fantom eltávolítás eredménye

**3. ábra:** Fantom jelenség és annak eltávolítása a bemutatott 3D CNN módszer felhasználásával.

egyik hátránya, hogy ha a felismerni kívánt osztályok száma és komplexitása növekszik, úgy egyre nagyobb kihívás a megfelelő jellemzők manuális meghatározása.

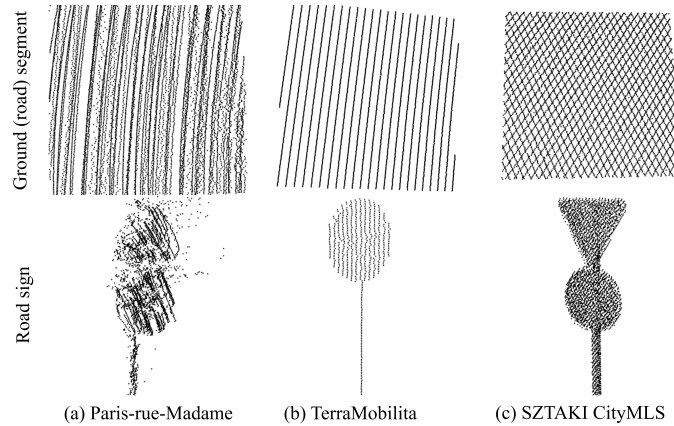
Az utóbbi években széleskörben alkalmaznak mélytanuló eljárásokat pontfelhők osztályozására és szegmentálására, ahol két fő irányt különböztethetünk meg. Az első a teljes szintért használja, így globális és magasabb szintű információkat is ki tud nyerni az adatokból, azonban a tanítás során fellépő nagy memória és GPU igény új adattárolási megoldásokat követel [14]. A másik módszer egy 3D mozgó ablak segítségével lokális részekre bontja a teljes pontfelhőt, ezzel csökkentve a számítási kapacitást a tanítás során, azonban így csak az adott lokális információkra támaszkodhat a modell. A *Vote3Deep* [10] módszer egy fix méretű 3D voxel tömbként kezel minden felismerendő osztályt, ami hatékonytalanná válhat, ha a felsimerendő osztályok száma és komplexitása nagy. [11] egy CNN alapú módszer, ami lokális jellemzők alapján osztályozza az egyes 3D voxel adatokat, azonban nem veszi számításba a pontsűrűséget, azonban a fantom objektumok felismerése pontos sűrűség modellt igényel [7, 8].

[15] több nézetből (12) leprojektálja a pontfelhőt egy 2D képsíkra, majd egy 2D CNN modellt használ a klasszifikáláshoz. Végül a klasszifikáció eredményét vissza projektálja a 3D pontfelhőbe. A többnézetes projektálást használó CNN modellek hatékonyan működnek szintetikus és olyan adatbázisokon, ahol a pontfelhő 3D objektum modellei teljesek és kevés zajt tartalmaznak, azonban a valós applikációknál használt MLS adatbázisok sokszor csak részlegesen szkennelt objektumokat, továbbá sok zajt és fantom objektumot tartalmaznak.

A PointNet++ [16] egy hierarchikus neurális hálózat, amit pont alapú pontfelhő klasszifikálására fejlesztettek ki. A megfigyelt pont egy adott sugarú környezetén belül véletlenszerűen mintavételezi a pontokat (tehát nem vesz figyelembe sűrűség információt), majd a hálózat ezeken a lokálisan kinyert pont halmazokon tanul. A modellt sűrű és pontos szintetikus, belső környezeteket tartalmazó adatbázisokon tesztelték, RGB színinformáció felhasználásával.

### 3. MLS, TLS adatbázisok elemzése

Az elmúlt években több városi környezetben készült, címkézett 3D MLS adatbázist tett közzé: Oakland adatbázis [17] (1.6M pont), Paris-rue-Madame (20M pont)



**4. ábra:** A Paris-rue-Madame, a TerraMobilita és a SZTAKI CityMLS adatbázis karakterisztikájának összehasonlítása.

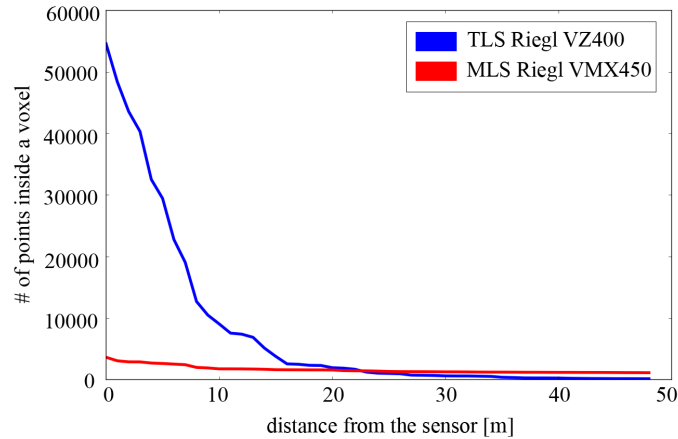
[18], IQmulus & TerraMobilita (12M pont) [19]. Azonban ezek az adatbázisok viszonylag kevés címkézett pontot tartalmaznak, ami korlátozza az adatigényes, túltanulásra (overfit) hajlamos mélytanuló eljárások alkalmazhatóságát.

A Semantic3D.net [20] adatbázis ugyan megfelelő mennyiségű címkézett pontot tartalmaz akár nagyobb modellek tanításához és teszteléséhez, azonban az adatokat egy staikus földi lézerszkennel (TLS) rögzítették, ami pontosabb és egy adott területen belül sokkal sűrűbb pontfelhőt állít elő, mint az MLS szenzorok. Azonban, ahogy az 5 ábrán látszik, egy mérési pozícióból, TLS szenzor által rögzített pontfelhő sűrűsége gyorsan csökken a szenzortól való távolság függvényében, ellenben ugyanazon a területen rögzített MLS pontfelhő sűrűsége egyenletes. Továbbá, egy TLS szenzorral több mérési pozícióból rögzített pontfelhő sűrűsége sokszor nagyon változó, mert a szenzort kezelő operátorok gyakran egyenletlen időközök mellett, tetszőleges szkennelési útvonalat járnak be, ezért gyakran egyazon TLS szenzorral rögzített adatbázisok karakterisztikája is jelentős eltéréseket mutat. Következésképpen, a mélytanuló és egyéb jellemző kinyerő alapú eljárásoknak számos gyakorlati nehézséggel kell szembenéznük TLS adatbázisok esetén.

Napjainkban az MLS pontfelhők címkézése és feldolgozása releváns kutatási terület, amit nagy ipari érdeklődés övez. Az MLS platformot egy mozgó jármű tetejére rögzítik, majd 30-50 kmh-ás (városi környezetben) átlagsebesség mellett, pontos, közel egyenletes sűrűségű pontfelhőt állít elő. Azonban a TLS adatfeldolgozással ellentétben itt nincs egyértelmű lehetőség a mozgásból származó fantom objektumok eltávolítására.

Cikkünkben bemutatunk egy manuális címkézéssel létrehozott adatbázist, melyet egy Riegl VMX-450 mobil térképező rendszer által készített nyers pontfelhő adatbázisból hoztunk létre, amit a Budapest Közút Zrt biztosított. Az új SZTAKI CityMLS adatbázis körülbelül 327 millió manuálisan címkézett pontot tartalmaz különböző városi helyszínekről, úgy mint fő utak kevés és nagy forgalommal, útkereszteződések, parkok és

járdák. Az adatbázisban számos különböző típusú objektum (autó, busz, villamos, gyalogos, kerékpár, városi növényzet és utcai objektum (jelzőlámpa, oszlop, buszmegálló, kuka, stb.)) található.



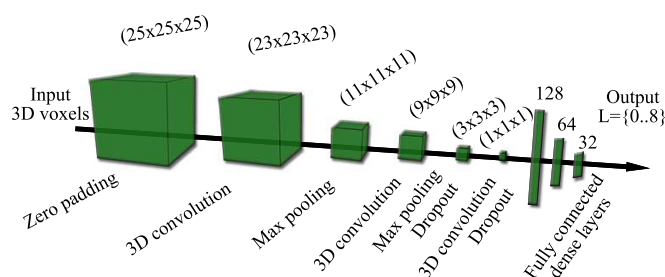
**5. ábra:** Pont sűrűség változása a szenzortól való távolság függvényében egy TLS referencia ponthoz képest. (Riegl VMX-450 MLS és Riegl VZ-400 TLS szenzor)

Ahogy az 4. ábra mutatja, a SZTAKI-CityMLS adatbázis karakterisztikája nagyban különbözik a TerraMobilita és Paris-rue-Madam adatbázisokétól. Habár a Paris-rue-Madame adatbázis tartalmazza a legdenser pontfelhőket, azonban a Velodyne típusú forgó sugaras lidarok regisztrációs nehézségei miatt a pontfelhők nagyon zajosak. A TerraMobilita adatbázis egy 2D sík lézer szkennelével készült, ami a szkennel platform sebességének a függvényében sok esetben túl ritkás pontfelhőket eredményez, így a kisebb tereptárgyak gyakran csak néhány vonalként jelennek meg a pontfelhőben. A Riegl VMX-450 térképező rendszer viszont kifejezetten városi környezet digitalizálására kifejlesztett eszköz, ami ipari minőségű, sűrű, nagy pontosságú és homogén pontfelhőt képes előállítani. Mindezek az új SZTAKI-CityMLS-t egy releváns, ipari minőségű adatbázissá teszik.

#### 4. 3D CNN alapú pontfelhő címkézés

Cikkünkben egy új 3D CNN alapú szegmentációs módszert ajánlunk, ami nagy mennyiségű fantom objektum okozta zaj mellett hatékonyan képes forgalmas városi környezetben készült sűrű MLS pontfelhők címkézésére. A bemutatott módszer egy korábbi [4], kifejezetten fantomok eltávolítására fókuszáló modellünkön alapul, amit jelen munkában egy kilenc osztályos szemantikus címkéző eljárás bővítettünk: *fantom, villamos/busz, gyalogos, autó, növényzet, oszlop, utcai objektum, talaj és épület*.

Az egyes tanító mintákat egy 3D voxel tömbként reprezentáljuk, de amíg a korábbi [4] módszernél minden egyes voxel az adott voxelbe eső 3D pontok számát tartalmazta, addig a jelenlegi munkában kiegészítettük ezt az információt egy második csatornával, ahol eltároljuk a lokális tanító minták globális magasságát a teljes pontfelhőhöz viszonyítva. A számítási kapacitás és memória igény minél alacsonyabban tartása érdekében egy ritkás voxel struktúra segítségével kezeljük a pontfelhőt, elkerülve ezzel a felesleges üres voxelek létrehozását.



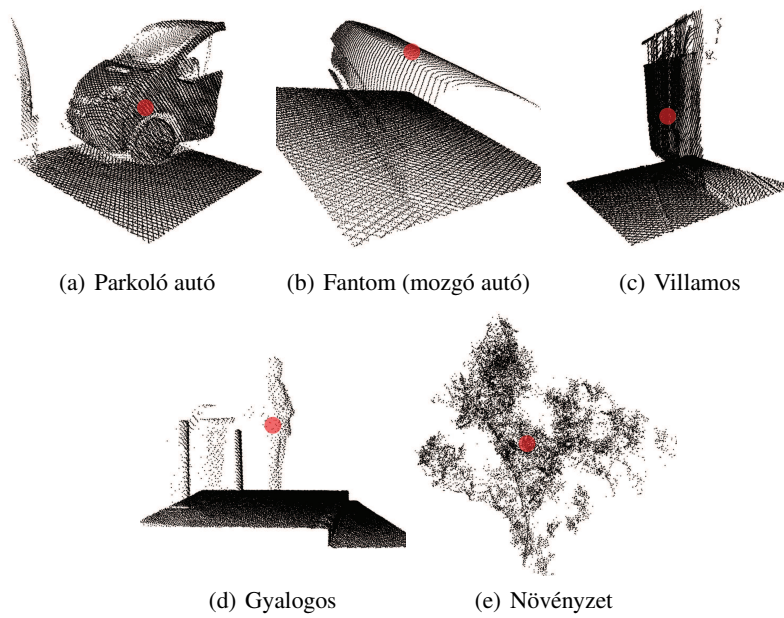
**6. ábra:** A bemutatott 3D CNN a következő elemekből épül fel: 3D konvolúciós réteg, max-pooling és dropout réteg. A hálózat bemenete  $K \times K \times K$  méretű voxel ( $K = 23$ ) tanítóminták, egy pontsűrűség és egy globális magasság csatornával. A hálózat kimenete egy egész szám a  $L = 0..8$  halmazból.

#### 4.1. Adat model

A tanítóminták kinyeréséhez első lépésben felépítünk egy ritkás  $\lambda = 0.1m$  felbontású voxel struktúrát a pontfelhő alapján, majd minden egyes voxelhez egy címként rendelünk az  $L$  halmazból, az adott voxelbe eső pontok osztályát figyelembe véve. A következő lépésben két jellemzőt rendelünk minden voxelhez: pont *sűrűség*, azaz az adott voxelbe eső pontok száma és a voxelbe eső pontok *átlag magassága*.

A tanításhoz használt bemenet egy  $K \times K \times K$  méretű voxel szomszédság ( $K = 23$ ), ahol minden voxelhez egy magasság és egy sűrűség jellemző van rendelve (két csatorna). A tanító minta osztályát a  $K \times K \times K$  méretű minta középső voxeléhez rendelt osztály határozza meg. A bemutatott 3D CNN módszer minden egyes  $K \times K \times K$  méretű mintát egyesével osztályoz, azaz mindig csak egy lokális részét látja a teljes pontfelhőnek. Azonban a mintákhoz rendelt globális magasság információ alapján a model információt kap az egyes minták globális vertikális pozíciójáról, ami segít elkülöníteni az egyes osztályokat.

Az 7 ábra demonstrálja a pontfelhőből kinyert különböző típusú tanító mintákat, ahol a vörös jelölő a középső voxel szemlélteti, ami alapján a minta címkézve lett. Mivel a tanító minta mérete  $K \cdot \lambda$  elég nagy ( $2.3m$ ) ezért sokszor többfajta objektum típust tartalmaz: pl. az 7(b) ábrán látható, hogy a minta fantom és talaj pontfelhő szegmenst is tartalmaz. Ez lehetővé teszi, hogy a hálózat a tanítás során kontextuális



**7. ábra:** A pontfelhőből kinyert különböző tanító minták. Minden minta  $K \times K \times K$  voxelt ( $K = 23$ ) tartalmaz, ahol a középső voxel (vörös szín) definiálja a minta osztályát.



információt is kinyerhessen, továbbá a tanító minták megjelenésében is nagy variációt eredményez.

## 4.2. 3D CNN architektúra

A bemutatott 3D CNN modell egy end-to-end módszer, ami két fő részből áll: jellemzők kinyerése (konvolúciós, max-pooling és dropout rétegek összessége), majd az adott jellemzők alapján a modell tanulása (teljesen összekötött rétegek). Mivel a tanító minták mérete  $(23 \times 23 \times 23)$  és a lehetséges osztályok száma (9) viszonylag kicsi, ezért a jól ismert LeNet-5 [21] struktúrához hasonlóan hoztuk létre a hálózatot: a 2D rétegeket kicseréltük a megfelelő 3D rétegekre, egy plusz konvolúciós réteget adtunk a hálózatához, továbbá regularizációs célból dropout módszert alkalmaztunk 6 ábra. A konvolúciós rétegeket  $3 \times 3 \times 3$  méretű konvolúciós kernellel és Rectified Linear Unit (ReLU) aktivációs függvényrel definiáltuk továbbá az egymást követő három konvolúciós réteg sorba 8, 16 és 32 filtert tartalmaz. A hálózat kimenetét Softmax függvényrel aktiváltuk és hogy elkerüljük a túltanulást a dropout regularizációs technikával minden egyes iterációnál véletlenszerűen deaktiváljuk a súlyok 30%-át. A hálózat optimalizálásához a Stochastic Gradient Descent (SGD) algoritmust használtuk és a validációs pontosság alapján változtatjuk a tanulási rátát.

Mivel az egyes minták szomszédsága nagy átfedést mutat, ezért a predikció eredménye egy simított 3D címkézet térkép lesz, ami lehetővé teszi objektumok kinyerését a szemantikailag összefüggő voxelek alapján (ld. 2 és 9 ábra).

## 5. Kiértékelés

### 5.1. Manuális pontfelhő címkézés és tanítás

3D pontfelhő alapú mélytanuló eljárások tanításához alapvető lépés nagy méretű MLS pontfelhő adatbázisok címkézése. Ezért létrehoztunk egy felhasználó barát pontfelhő címkéző alkalmazást, ami lehetővé teszi a felhasználónak, hogy gyorsan nagy pontosság mellett tetszőleges formájú 3D pontfelhőket címkézhessen.

A felhasználó egy téglalap alakú területet tud megjelölni a 2D képernyő síkon, ami az aktuális kamera nézőponttal egy gúlát definiál a 3D koordináta-rendszerben. A pontfelhő azon pontjait amelyek a gúla belsejébe esnek hozzáadjuk az aktuális kijelöléshez vagy éppen töröljük, attól függően, hogy milyen műveletet hajtunk végre. Számos gúla kombinációjából tetszőleges formájú objektumot lehet definiálni, majd a kijelölt pontokhoz egy címkét rendelhetünk. Az alkalmazással közel 327M pontot címkéztünk fel az említett kilenc osztály felhasználásával, ami nagyjából  $30.000 \text{ m}^2$  területnek felel meg több mint 50m szintemelkedéssel. A címkézés eredményeként egy új MLS adatbázist (SZTAKI CityMLS) hoztunk létre.

A következő lépésben tanítás, validáció és tesztelés céljából három, egymással nem átfedő részre osztottuk a címkézett adatbázist. A tanításhoz mind a kilenc osztályból véletlenszerűen kiválasztottunk 100.000 voxel és a bemutatott módon kinyertük azok  $K \times K \times K$  méretű szomszédságát a megfelelő sűrűség és magasság csatornákkal. A tanításhoz így összesen 900.000 mintát használtunk fel. A tanítás folyamata során egy

180.000 méretű (osztályonként 20000) validációs halmazon optimalizáltuk a modell paramétereit, ahol minden validációs mintát egy a tanító halmaztól független területről választottunk.

A modell kvantitatív kiértékelését egy a tanító és a validációs halmaztól teljesen független, két millió teszt mintát tartalmazó halmazon végeztük.

## 5.2. Tesztelés és összehasonlítás

A bemutatott módszert három referencia algoritmussal hasonlítottuk össze. Az első egy 3D CNN alapú módszer (OG-CNN) [11], ami a bemutatott módszerhez hasonlóan lokális voxel szomszédságot használ a tanításhoz, azonban nem használ sűrűség információt, csak egy egyszerű foglaltsági voxel rácsot, továbbá a globális magasságot sem veszi számításba. [15] több nézőpontból egy 2D képsíkra projektálja a 3D pontokat, majd egy 2D CNN modellt használ a tanításhoz. A harmadik módszer egy sokat hivatkozott pontfelhő klasszifikáló módszer: PointNet++ [16].

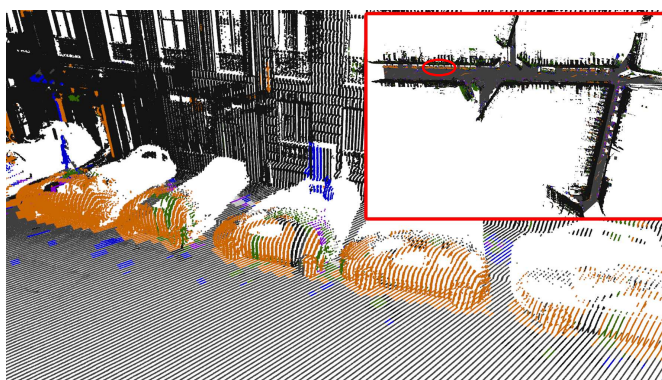
A 9 ábrán egy kvalitatív összehasonlítás látható a manuálisan címkézett, az OG-CNN, a többnézetes 2D CNN, és a PointNet++ referencia módszerek, továbbá a bemutatott C<sup>2</sup>CNN modell között. Az 1 táblázat egy kvantitatív összehasonlítást tartalmaz a referencia és a bemutatott módszerek között, ahol minden egyes osztály esetén voxel szintű precision (Pr), recall (Rc) és F-érték (F-r) mérőszámokkal mértük a modellek teljesítményét. Az 1 táblázat nem tartalmazza az *épület* és *talaj* mintákhoz tartozó mérőszámokat, mert ezen osztályokat a bemutatott C<sup>2</sup>CNN modell nagyon hatékonyan képes megtanulni (98% pontossággal) és ez a modell kiértékelésénél nagyban befolyásolná a végső eredményt.

A teszt eredmények azt mutatják, hogy a bemutatott C<sup>2</sup>CNN módszer minden osztályt nagyobb mint 83%-os F-értékel képes felsimerni. A precision és a recall értékek minden osztály esetén elég hasonlóak, tehát a hamis negatív és a hamis pozitív találatok közel kiegyenlítettek. A hálózat a nagy síkfelületekkel rendelkező *villamos/busz* kategóriát és a *növényzetet*, ami egy rendezetlen pont halmazként, tipikus magasság értékekkel (a bokrok az utca szinten, míg a lombkorona magasabb régiókban) jelenik meg detektálja a legnagyobb pontossággal. Mindazonáltal, a nagyobb strukturális variációt tartalmazó osztályokat (fantomok, gyalogosok, és autók) is közel 85-87% F-érték mellett detektálja a modell.

Mind a három referencia módszer teljesítménye alulmarad a bemutatott módszerrel szemben: az OG-CNN 14.8%-al, a 2D CNN 17.4%-al, míg a PointNet++ 3.8%-al. A referencia módszerek a fantomok felismerésénél teljesítenek a legrosszabbul, mivel a mérő platform és az objektumok változó mozgása miatt a fantom objektumok nagy strukturális diverzitással és változó pontsűrűséggel rendelkeznek. Azonban a szakirodalomban fellelhető módszerek úgy mint a referencia módszerek is főleg szintetikus adatbázisokra lettek kifejlesztve, így nem vesznek figyelembe olyan tényezőket mint a mérésből adódó változó pontsűrűség. A parkoló autók esetén a felismerési ráta (recall) minden módszer esetén elég magas, azonban a referencia módszerek gyakran összekeverik őket a fantomokkal ezért a precision érték alacsonyabb. Mivel az OG-CNN nem veszi figyelembe a globális magasság értékeket és csak lokális értékekre támaszkodik, ezért gyakran a lombkorona szinten is detektál gyalogosokat, míg a bemu-

tatott  $C^2$ CNN model a globális magasság információt felhasználva kiküszöböli ezeket az eseteket.

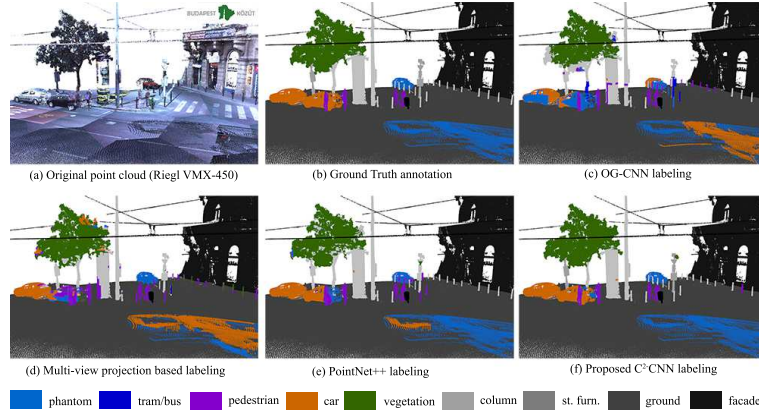
Az ajánlott  $C^2$ CNN módszert teszteltük a ritkább pontfelhőket tartalmazó *TerraMobilita* [19] adatbázison is, a szegmentálás kvalitatív eredménye a 8 ábrán látható. A tesztek megmutatták, hogy a model adaptálható a ritkásabb adatra, azonban maga az adatbázis túl kevés címkézett adatot tartalmaz mélytanuló modellek pontos tanításához. A *Paris-rue-Madame* [18] adatbázis esetén is hasonló eredményeket értünk el, azonban a *Semantic3D.net* [20] adatbázis esetén, ahol a statikus TLS adat rögzítési technikának köszönhetően drasztikus pontsűrűség különbségeket találunk, a bemutatott modell alul-teljesített.



**8. ábra:** Teszt eredmények a TerraMobilita adatbázison.

### 5.3. Jármű lokalizáció esettanulmány szemantikailag címkézett MLS adatok alapján

Ebben a fejezetben egy lehetséges felhasználási lehetőségét mutatjuk be a  $C^2$ CNN módszer által szegmentált pontfelhőknek egy automatív alkalmazáson keresztül. A feladat egy önjáró jármű valós idejű lokalizációjának és orientációjának meghatározása a szegmentált MLS pontfelhő és az önjáró autó Velodyne típusú [22] forgósugaras szenzora (RMB) által érzékelt pontfelhő felhasználásával. Tapasztalataink szerint városi környezetben, mint pl. Budapest belvárosa egy általános forgalomban elérhető GPS pontossága átlagosan 1 és 10 m között ingadozik. Azért, hogy kiküszöböljük a GPS pontatlanságából adódó pozicionálási hibát, egy objektum alapú lokalizációs módszert vezettünk be [23, 24], ami robotosan képes egy sűrű MLS pontfelhőt és egy ritkás RMB Lidar által rögzített pontfelhőt egy koordinátarendszerbe transzformálni. Első lépésben egy 2D rácsot illesztünk a vízszintes síkra és a pontfelhő 3D pontjait a megfelelő 2D cellákhoz rendeljük, majd egy adaptív összefüggő komponens kereső algoritmus alapján [22] objektumokat nyerünk ki mind az RMB mind az MLS pontfelhőből. A



**9. ábra:** A három referencia ((c) OG-CNN, (d) 2D CNN és (e) PointNet++) és a bemutatott (f)  $C^2$ CNN módszer szegmentációs eredményének a kvalitatív összehasonlítása, egy azon térrészen belül. A (b) ábra a validációhoz szükséges manuálisan címkézet adatot mutatja.

regisztrációs algoritmus fő lépése egy gyors objektum alapú transzformáció becslés [23] a két pontfelhőből kinyert objektum halmaz alapján. Az Iterative Closest Point (ICP) és a Normal Distributions Transform (NDT) [24] pontszintű regisztrációs algoritmusokkal ellentétben a bemutatott módszer [23] valós időben működik, továbbá kezdeti translációs és rotációs hibát képes kezelni a két pontfelhő között.

Azonban a bemutatott regisztrációs metódus azt feltételezi, hogy az MLS pontfelhőből kinyert objektumok nagy része statikus, állandó (jelzőlámpák, fatörzsek, buszmegálló, stb.) tájékozódási pont, míg a fantom és mozgó objektumok zajként jelennek meg a transzformáció becslés szempontjából. Nyilvánvalóan, minél több fantom objektum található a kinyert objektumok között, annál nagyobb mértékben befolyásolják a transzformáció pontosságát, hiszen ahelyett, hogy az algoritmus a megfelelő statikus MLS objektumhoz rendelné az RMB Lidar objektumait sokszor egy fantom objektumhoz rendel, ami nagyban befolyásolja a globális transzformáció eredményét. Ebből kifolyólag, a fantom és egyéb mozgó objektumok detektálása az MLS adatokon kritikus lépés a regisztrációs algoritmus szempontjából.

Az 10 ábra demonstrálja a regisztrációs algoritmus teljesítményét az eredeti fantom objektumokat is tartalmazó MLS pontfelhő és a  $C^2$ CNN módszerrel megtisztított pontfelhő felhasználásával. Az ábrán a RMB Lidar által rögzített pontfelhő *vörös*, míg a fantom objektumok *kék* illetve *lila* színnel jelennek meg. A maradék színek a  $C^2$ CNN módszerrel címkézett pontfelhőt jelenítik meg. A felső sor a kezdeti GPS transzformáció utáni hibát mutatja, ami jelen esetben 7 méter 8.5 fokos rotációs hibával kiegészülve. Az 10(a) ábrán a fantom objektumokat tartalmazó nyers MLS pontfelhőt használjuk referencia térképként, az 10(c) ábra pedig a regisztráció eredményét mutatja be. Azonban ha megtisztítjuk a pontfelhőt (10(b) ábra), akkor a regisztrációs sokkal pontosabb (10(d) ábra).

**1. táblázat:** A  $C^2$ CNN módszer kvantitatív kiértékelése a kihívást jelentő osztályokra fókuszálva, továbbá a referencia módszerekkel való kvantitatív kiértékelés [11], [15] and [16].

Osztály	OG-CNN [11]			2D CNN[15]			PointNet++ [16]			$C^2$ CNN		
	Pr %	Rc %	F-r %	Pr %	Rc %	F-r %	Pr %	Rc %	F-r %	Pr %	Rc %	F-r %
Fantom	85.3	34.7	49.3	76.5	45.3	56.9	82.3	76.5	79.3	84.3	85.9	<b>85.1</b>
Gyalogos	61.2	82.4	70.2	57.2	66.8	61.6	86.1	81.2	83.6	85.2	85.3	<b>85.2</b>
Autó	56.4	89.5	69.2	60.2	73.3	66.1	80.6	92.7	86.2	86.4	88.7	<b>87.5</b>
Növényzet	72.4	83.4	77.5	71.7	78.4	74.9	91.4	89.7	90.5	98.2	95.5	<b>96.8</b>
Oszlop	88.6	74.3	80.8	83.4	76.8	80.0	83.4	93.6	<b>88.2</b>	86.5	89.2	87.8
Vilmos/Busz	91.4	81.6	86.2	85.7	83.2	84.4	83.1	89.7	86.3	89.5	96.9	<b>93.0</b>
Utcai objektum	72.1	82.4	76.9	57.2	89.3	69.7	84.8	82.9	<b>83.8</b>	88.8	78.8	83.5
Összesítés	76.9	74.2	75.5	72.5	73.4	72.9	85.6	87.5	86.5	90.4	90.2	<b>90.3</b>

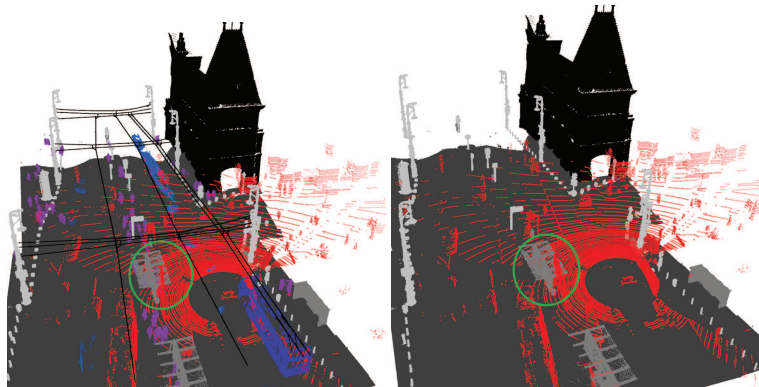
A voxel szintű precision (Pr), recall (Rc) és F-érték (F-r) %-os formában van feltüntetve.

Az RMB Lidar és az MLS pontfelhők között három különböző konfigurációval futtattuk a [23] regisztrációs algoritmust: (i) nyers MLS pontfelhő, (ii)  $C^2$ CNN módszerrel megtisztított pontfelhő és (iii) manuálisan címkézett MLS adat. A kvantitatív eredményeket a 2 táblázat tartalmazza, ahol öt különböző tesztet különítettünk el a regisztráció szempontjából: *főút*, *mellék utca*, *útkereszteződés*, továbbá *kevés* illetve *sok* fantomot tartalmazó régió. A táblázat a regisztráció utáni hibát mutatja az összetartozó objektumok átlagos távolsága és rotációs különbsége alapján. Látható, hogy a  $C^2$ CNN módszerrel szegmentált MLS térkép felhasználásával közel azt az eredményt érjük el, mintha a fantom mentes manuális adatot használtuk volna referencia térképként, azonban a nyers MLS pontfelhő esetén a regisztráció után is jelentős hiba marad.

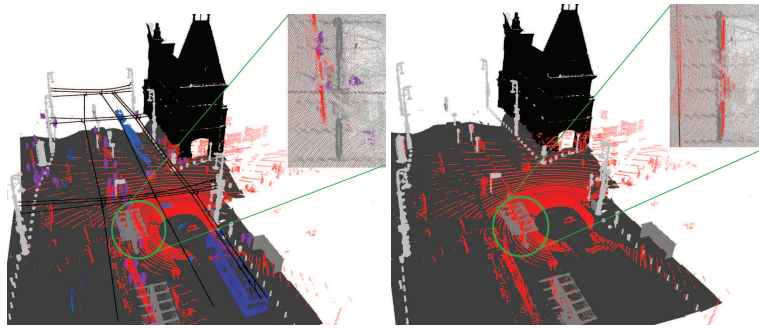
**2. táblázat:** A bemutatott pontfelhő regisztrációs módszer [23] kvantitatív kiértékelése a manuálisan címkézett MLS pontfelhő, a  $C^2$ CNN módszerrel címkézett pontfelhő, és az eredeti nyers szemantikus információ nélküli pontfelhő alapján.

Adat	Nyers MLS pontfelhő		$C^2$ CNN		Manuálisan címkézett adat	
	s [m]	rot [deg]	s [m]	rot [deg]	s [m]	rot [deg]
Fő út	1.74	3.92	0.37	1.19	0.26	0.97
Mellék utca	1.37	2.38	0.29	0.83	0.18	0.78
Útkereszteződés	2.42	4.02	0.45	1.33	0.29	0.89
Helyszín sok fantommal	0.93	1.60	0.26	0.87	0.21	0.77
Helyszín kevés fantommal	2.14	3.53	0.48	1.37	0.28	0.95
Összesítés	1.72	3.09	0.37	1.18	0.24	0.87

A regisztráció utáni eltolás hiba méterben, míg a rotációs hiba fokban adott.



(a) GPS alapú kezdeti összeillesztés di- (b) GPS alapú kezdeti összeillesztés di-  
namikus objektumokkal namikus objektumok nélkül



(c) Objektum alapú [23] regisztrációs (d) Objektum alapú [23] regisztrációs  
módszerer eredménye dinamikusan ob- módszerer eredménye dinamikusan ob-  
jektumokkal jektumok nélkül

**10. ábra:** A dinamikus objektumoknak nagy hatása van a regisztrációs folyamat eredményére, mert helytelenül erősítik a nem megfelelő transzformációk valószínűségét, tehát a dinamikus (fantom) objektumok nélküli térkép használata robusztusabbá és pontosabbá teszi a regisztráció eredményét. A vörös színű pontok az önjáró jármű által érzékelt pontfelhő, a kék és a lila színek a dinamikus objektumokat reprezentálják, míg a többi szín a  $C^2$ CNN által szegmentált térképet demonstrálja.

#### 5.4. Implementációs kérdések és futási idő

A  $C^2$ CNN modellt a Keras és a Tensorflow keretrendszerekkel implementáltuk. A  $C^2$ CNN modell tanítása (közel 36 óra) a *SZTAKI CityMLS* adatbázison történt egy Nvidia Geforce GTX 1080 GPU segítségével. Egy  $23 \times 23 \times 23$  méretű minta predikciójához kevesebb mint  $10^{-4}$  másodpercre van szükség, így egy körülbelül  $60m \times 110m$  méretű terület szegmentálása nagyjából 3 percbe kerül.

## 6. Konklúzió

Cikkünkben bemutattunk egy két csatornás 3D CNN alapú módszert nagyméretű MLS pontfelhők kilenc osztályba történő szegmentálására, ami később alapja lehet pontos, nagy felbontású 3D HD térképek létrehozásához. Változatos, valós város környezetben teszteltük a modellt és három referencia modellel hasonlítottuk össze, továbbá bemutattunk egy lehetséges alkalmazását a szegmentált térképnek.

## 7. Köszönetnyilvánítás

A szerzők köszönetet mondanak a Budapest Közút Zrt.-nek a Riegl VMX-450 MLS tesztadatok biztosításáért. A projektet a Nemzeti Kutatási, Fejlesztési és Innovációs Alap (NKFIH KH-125681) és az Emberi Erőforrás Fejlesztési Operatív Program (EFOP-3.6.2-16-2017-00013) támogatta. Nagy Balázs részéről a közreműködés részben az Emberi Erőforrások Minisztériuma ÚNKP-18-3 kódszámú Új Nemzeti Kiválóság Programjának támogatásával történt.

## Irodalom

1. Zheng, H., Wang, R., Xu, S.: Recognizing street lighting poles from mobile LiDAR data. *IEEE Trans. Geosci. and Rem. Sens.* **55** (2017) 407–420
2. Yu, Y., Li, J., Guan, H., Wang, C.: Automated detection of three-dimensional cars in mobile laser scanning point clouds using DBM-Hough-Forests. *IEEE Trans. Geosci. and Rem. Sens.* **54** (2016) 4130–4142
3. Wu, B., Yu, B., Yue, W., Shu, S., Tan, W., Hu, C., Huang, Y., Wu, J., Liu, H.: A voxel-based method for automated identification and morphological parameters estimation of individual street trees from mobile laser scanning data. *Rem. Sens.* **5** (2013) 584
4. Nagy, B., Benedek, C.: 3D CNN based phantom object removing from mobile laser scanning data. In: *Int'l Joint Conference on Neural Networks (IJCNN)*, Anchorage, Alaska, USA (2017) 4429–4435
5. Kanzok, T., Süß, F., Linsen, L., Rosenthal, R.: Efficient removal of inconsistencies in large multi-scan point clouds. In: *Int'l Conf. in Central Europe on Computer Graphics, Visualization and Computer Vision*, Pilsen, Czech Rep. (2013)
6. Papadimitriou, S., Kitagawa, H., Gibbons, P.B., Faloutsos, C.: LOCI: fast outlier detection using the local correlation integral. In: *Int'l Conf. on Data Engineering*, Los Alamitos, CA, USA (2003) 315–326
7. Sotoodeh, S.: Outlier detection in laser scanner point clouds. In: *ISPRS Arch. Photogramm. Remote Sens. and Spatial Inf. Sci.* Volume XXXVI-5. (2006) 297–302

8. Köhler, J., Nöll, T., Reis, G., Stricker, D.: Robust outlier removal from point clouds acquired with structured light. In: Eurographics (Short Papers), Cagliari, Italy (2012) 21–24
9. Börcs, A., Nagy, B., Benedek, C.: Instant object detection in lidar point clouds. In: IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 7. (2017) 992–996
10. Engelcke, M., Rao, D., Wang, D.Z., Tong, C.H., Posner, I.: Vote3Deep: Fast object detection in 3D point clouds using efficient convolutional neural networks. In: IEEE International Conference on Robotics and Automation (ICRA), Singapore (2017) 1355–1361
11. Huang, J., You, S.: Point cloud labeling using 3D convolutional neural network. In: International Conference on Pattern Recognition (ICPR), Cancun, Mexico (2016) 2670–2675
12. Koppula, H.S., Anand, A., Joachims, T., Saxena, A.: Semantic labeling of 3D point clouds for indoor scenes. In: Int'l Conf. Neural Inf. Processing Systems (NIPS), Granada, Spain (2011) 244–252
13. Hackel, T., Wegner, J.D., Schindler, K.: Fast semantic segmentation of 3D point clouds with strongly varying density. ISPRS Ann. Photogramm. Remote Sens. and Spatial Inf. Sci. **III-3** (2016)
14. Riegler, G., Ulusoy, A.O., Geiger, A.: OctNet: Learning Deep 3D Representations at High Resolutions. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2017) 6620–6629
15. Pang, G., Neumann, U.: 3D point cloud object detection with multi-view convolutional neural network. In: International Conference on Pattern Recognition (ICPR), Cancun, Mexico (2016) 585–590
16. Qi, C., Yi, L., Su, H., Guibas, L.: PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In: Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA (2017) 5105–5114
17. Munoz, D., Bagnell, J., Vandapel, N., Hebert, M.: Contextual classification with functional max-margin markov networks. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). (2009) 975–982
18. Serna, A., Marcotegui, B., Goulette, F., Deschaud, J.: Paris-rue-Madame database: a 3D mobile laser scanner dataset for benchmarking urban detection, segmentation and classification methods. In: Int'l Conf. on Pattern Rec. App. and Methods (ICPRAM), Angers, France (2014)
19. Vallet, B., Brédif, M., Serna, A., Marcotegui, B., Paparoditis, N.: TerraMobilita/iQmulus urban point cloud analysis benchmark. Computers and Graphics **49** (2015) 126–133
20. Hackel, T., Savinov, N., Ladicky, L., Wegner, J., Schindler, K., Pollefeys, M.: Semantic3D.net: A new large-scale point cloud classification benchmark. In: ISPRS Ann. Photogramm. Remote Sens. and Spatial Inf. Sci. Volume IV-1-W1. (2017) 91–98
21. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86** (1998) 2278–2324
22. Börcs, A., Nagy, B., Benedek, C.: Fast 3-D urban object detection on streaming point clouds. In: Workshop on Computer Vision for Road Scene Understanding and Autonomous Driving at ECCV'14. Volume 8926 of LNCS. Springer, Zürich, Switzerland (2015) 628–639
23. Nagy, B., Benedek, C.: Real-time point cloud alignment for vehicle localization in a high resolution 3d map. In: Workshop on Computer Vision for Road Scene Understanding and Autonomous Driving (ECCV), Munchen, Germany (2018)
24. B. Gálai, B.N., Benedek, C.: Crossmodal point cloud registration in the hough space for mobile laser scanning data. In: International Conference on Pattern Recognition (ICPR), Cancun, Mexico (2016) 3363–3368