Duboki modeli za registraciju oblaka točaka

Matošević, Manuel

Master's thesis / Diplomski rad

2025

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Electrical Engineering and Computing / Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:168:446480

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2025-03-23**

Repository / Repozitorij:

FER Repository - University of Zagreb Faculty of Electrical Engineering and Computing repozitory

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 734

DUBOKI MODELI ZA REGISTRACIJU OBLAKA TOČAKA

Manuel Matošević

Zagreb, veljača 2025.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 734

DUBOKI MODELI ZA REGISTRACIJU OBLAKA TOČAKA

Manuel Matošević

Zagreb, veljača 2025.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 30. rujna 2024.

DIPLOMSKI ZADATAK br. 734

Pristupnik: **Manuel Matošević (0036511659)**

Studij: Računarstvo

Profil: Računarska znanost

Mentor: izv. prof. dr. sc. Tomislav Petković

Zadatak: **Duboki modeli za registraciju oblaka točaka**

Opis zadatka:

Registracija oblaka točaka je problem nalaženja rotacije i translacije koja opisuje preklapanje dva oblaka točaka čime se omogućava spajanje oblaka točaka u neprekinutu cjelinu. U diplomskom radu je potrebno dati pregled osnovnih koraka u klasičnom postupku 3D registracije oblaka točaka te ukratko opisati ICP algoritam. Zatim je potrebno pretražiti literaturu te identificirati koji se sve duboki modeli koriste za 3D registraciju. Pronađene duboke modele je potrebno razvrstati u grupe prema njihovim najvažnijim karakteristikama. Iz barem dvije od tih grupa je zatim potrebno odabrati barem jedan reprezentativni model te je zatim potrebno provesti eksperimentalnu verifikaciju odabranih modela na uobičajenim skupovima 3D podataka te ih usporediti međusobno i s klasičnim ICP algoritmom za registraciju.

Rok za predaju rada: 14. veljače 2025.

Ovim putem želim izraziti zahvalnost svojoj obitelji na njihovoj podršci i strpljenju tijekom cijelog mog obrazovanja, a posebno tijekom studija. Hvala vam što ste uvijek bili uz mene i što ste me poticali da težim najboljem.

Sadržaj

1. Uvod **3**

2. Osnove registracije oblaka točaka **5**

2.1. Uvod u registraciju 5

2.2. Osnovni koraci registracije 5

2.2.1. Preprocesiranje podataka 6

2.2.2. Ekstrakcija značajki 6

2.2.3. Uspostavljanje korespondencija 6

2.2.4. Procjena transformacije 6

3. Metode registracije oblaka točaka **8**

3.1. Iterativna najbliža točka (ICP):

Klasični pristup registraciji 8

3.1.1. Osnovni principi algoritma 8

3.1.2. Koraci algoritma ICP 8

3.1.3. Prednosti i ograničenja 9

3.1.4. Proširenja algoritma 10

3.2. Deep Closest Point (DCP):

Kombinacija klasičnih metoda i dubokog učenja 10

3.2.1. Osnovni principi modela 11

3.2.2. Ključni koraci algoritma 11

3.2.3. Arhitektura modela 12

3.2.4. Prednosti i nedostaci 12

3.2.5. Primjena u eksperimentu 13

3.3. PointNetLK:
Globalne značajke i iterativna optimizacija 13
3.3.1. Osnovni principi modela 13
3.3.2. Ključni koraci algoritma 13
3.3.3. Arhitektura modela 14
3.3.4. Prednosti i nedostaci 15
3.3.5. Primjena u eksperimentu 15

4. Eksperimentalna postavka i rezultati 16
4.1. Računalno okruženje 16
4.2. Skup podataka: ModelNet40 17
4.2.1. Specifikacije skupa podataka 17
4.2.2. Razlozi za odabir ModelNet40 skupa 18
4.2.3. Generiranje oblaka točaka 18
4.3. Rezultati metoda 19
4.3.1. Iterativna najbliža točka (ICP) 20
4.3.2. Deep Closest Point (DCP) 23
4.3.3. PointNetLK 27
4.3.4. Usporedba rezultata 30

5. Zaključak 33

Literatura 35

Sažetak 37

Abstract 38

1. Uvod

Registracija oblaka točaka važan je problem u obradi i analizi 3D podataka, s primjenom u područjima poput robotike, autonomne vožnje i računalnog vida. Cilj registracije je pronaći rotaciju i translaciju koje omogućuju usklađivanje dva skupa točaka u jedinstvenu i koherentnu cjelinu. U ovom radu istražujemo klasične i suvremene metode registracije oblaka točaka, s posebnim naglaskom na eksperimentalnu evaluaciju. Jedan od najpoznatijih klasičnih algoritama je Iterativna najbliža točka (engl. *Iterative Closest Point*) [1], koji će se u nastavku rada referencirati skraćeno kao ICP. S razvojem dubokog učenja pojavili su se modeli poput (engl. *Deep Closest Point*) [2], skraćeno DCP i PointNetLK [3], koji koriste neuronske mreže za izdvajanje značajki, omogućujući precizniju registraciju u složenim scenarijima [4]. Eksperimentalni dio rada temelji se na usporedbi ICP-a i odabranih dubokih modela koristeći standardni skup podataka ModelNet40. Cilj je usporediti točnost i računalnu složenost ovih metoda, pružajući uvid u njihove prednosti i ograničenja. Rad je strukturiran na sljedeći način:

- U **Poglavlju 2.** daje se pregled osnovnih koraka registracije oblaka točaka. Obrađuje se preprocesiranje podataka, metode usklađivanja točaka i procjena transformacija. Ovo poglavnje pruža teoretsku podlogu potrebnu za razumijevanje daljnjih poglavlja.

- **Poglavlje 3.** detaljno opisuje algoritam Iterativna najbliža točka (ICP) , njegov matematički model, glavne korake te prednosti i nedostatke. Istražuje duboke modele za registraciju oblaka točaka, s fokusom na Deep Closest Point (DCP) [2] i PointNetLK [3]. Obrađene su arhitekture ovih modela, njihove specifične metode učenja te prednosti u usporedbi s klasičnim pristupima.

- **Poglavlje 4.** opisuje eksperiment, uključujući pripremu skupa podataka Model-

Net40 [5], implementaciju algoritama i modela te metode evaluacije. Rezultati usporedbe ICP-a, DCP-a i PointNetLK-a predstavljeni su numerički i vizualno.

- **Poglavlje 5.** donosi zaključke te naglašava ključne doprinose rada.

2. Osnove registracije oblaka točaka

Oblak točaka je skup točaka u trodimenzionalnom prostoru koji predstavlja površinu objekta ili scene. Rigidna transformacija $[R, t] \in SE(3)$ sastoji se od rotacije $R \in SO(3)$ i translacije $t \in \mathbb{R}^3$, pri čemu održava udaljenost između točaka. Ova transformacija ne mijenja oblik ni dimenzije objekta.

2.1. Uvod u registraciju

Registracija oblaka točaka proces je u obradi 3D podataka koji omogućuje poravnanje i spajanje više oblaka točaka u jednu cjelovitu i konzistentnu reprezentaciju [4]. Ovaj proces je nužan u situacijama kada podaci dolaze iz različitih izvora ili su prikupljeni u različitim vremenskim razmacima, što često rezultira fragmentiranim i nepovezanim prikazima. Cilj registracije je usklađivanje tih podataka tako da se maksimalno očuva geometrijska konzistentnost, omogućujući preciznu analizu i daljnju upotrebu.

Registracija oblaka točaka rješava probleme poput nepotpunosti podataka, šuma i varijacija u orijentaciji. Proces registracije obuhvaća nekoliko ključnih koraka, od pripreme podataka do procjene optimalne transformacije između oblaka. Iako su klasični pristupi poput ICP-a [1] dugo dominirali ovim područjem, sve veća dostupnost podataka i računalne snage potaknula je razvoj modela temeljenih na dubokom učenju, koji nude značajna poboljšanja u točnosti.

2.2. Osnovni koraci registracije

Proces registracije može se podijeliti u četiri ključna koraka koje zajedno omogućuju precizno poravnanje oblaka točaka. Preprocesiranje podataka, ekstrakcija značajki, uspostavljanje korespondencija, te procjena transformacije.

2.2.1. Preprocesiranje podataka

Prvi korak uključuje pripremu podataka za registraciju. Oblaci točaka često sadrže šumove, redundantne točke ili su nepotpuni. Preprocesiranje osigurava kvalitetu podataka kroz uklananje šuma, smanjenje broja točaka (engl. *downsampling*) i normalizaciju. Cilj je smanjiti složenost podataka dok se zadržava dovoljno informacija za uspješnu registraciju. Primjerice, normalizacija podrazumijeva centriranje oblaka točaka u ishodište koordinatnog sustava i skaliranje na jediničnu sferu kako bi se uklonio utjecaj apsolutne veličine i pozicije.

2.2.2. Ekstrakcija značajki

Nakon preprocesiranja, iz podataka se izdvajaju značajke koje opisuju geometrijsku strukturu oblaka. Ove značajke mogu biti lokalne, kao što su detalji o relacijama između najbližih točaka, ili globalne, koje opisuju cjelokupni oblik oblaka. Klasične metode za izdvajanje značajki često koriste ručno dizajnirane deskriptore, dok moderni modeli dubokog učenja, poput PointNet [3] arhitekture, omogućuju automatsko učenje značajki iz sirovih podataka.

2.2.3. Uspostavljanje korespondencija

Na temelju izdvojenih značajki, određuju se parovi točaka između dva oblaka koji najbolje odgovaraju. Ovaj korak ključan je za povezivanje dvaju oblaka, a često koristi algoritme za pretraživanje najbližih točaka (npr. KD-stablo) radi ubrzanja. Kvaliteta korespondencija izravno utječe na točnost konačne registracije. Korespondencije kod dubokih algoritama često su puno bolje postavljene, zbog čega ti algoritmi obično daju izravno bolje rezultate.

2.2.4. Procjena transformacije

Posljednji korak je izračun optimalne transformacije koja minimizira razliku između korespodentnih točaka. Transformacija uključuje rotaciju i translaciju koje najbolje usklađuju oba oblaka. Klasične metode, poput SVD [4] algoritma, pružaju robusno rješenje za ovaj optimizacijski problem. U modernim pristupima, neuronske mreže integriraju procjenu transformacije kao dio svoje arhitekture, čime se dodatno poboljšava otpornost

na šumove i nepravilnosti u podacima.

U narednim poglavljima analiziraju se prednosti i ograničenja klasičnih i dubokih pristupa registraciji, s naglaskom na njihovu primjenu u različitim eksperimentalnim scenarijima.

3. Metode registracije oblaka točaka

3.1. Iterativna najbliža točka (ICP): Klasični pristup registraciji

Iterativna najbliža točka [1] jedan je od najpoznatijih algoritama za registraciju oblaka točaka. Ovaj klasični pristup iterativno procjenjuje rigidnu transformaciju, koja uključuje rotaciju i translaciju, kako bi minimizirao udaljenost između korespondirajućih točaka dvaju oblaka.

3.1.1. Osnovni principi algoritma

Algoritam ICP temelji se na minimizaciji kvadratne udaljenosti između korespondentnih točaka iz dvaju skupova točaka. Polazi od početne aproksimacije transformacije između dolaznog oblaka Q i polaznog oblaka P, a zatim iterativno pronalazi optimalnu transformaciju T, definiranu rotacijom R i translacijom t, koja usklađuje dva oblaka [1].

Matematički, funkcija troška koju algoritam minimizira je:

$$E(T) = \sum_{i=1}^{N} \|T(q_i) - p_i\|^2, \tag{3.1}$$

gdje je $T(q_i) = Rq_i + t$ transformirana točka iz oblaka Q, dok je p_i odgovarajuća točka u oblaku P. Cilj algoritma je pronaći transformaciju T koja minimizira trošak $E(T)$ [6].

3.1.2. Koraci algoritma ICP

Algoritam ICP sastoji se od nekoliko ključnih koraka:

1. **Inicijalizacija transformacije**: Postavlja se početna aproksimacija transforma-

cije T_0, obično matrica identiteta ako nije poznata bolja aproksimacija. Kvalitetna inicijalizacija ključna je za izbjegavanje lokalnih minimuma i za bržu konvergenciju .

2. **Pronalaženje korespondencija**: Za svaku točku $q_i \in Q$, algoritam pronalazi najbližu točku $p_i \in P$ prema euklidskoj udaljenosti:

$$p_i = \arg\min_{p \in P} \|q_i - p\|. \tag{3.2}$$

   Ovaj korak ubrzava se učinkovitom upotrebom struktura podataka poput KD-stabla, što omogućuje znatno brže pretraživanje u velikim oblacima točaka.

3. **Procjena transformacije**: Izračunava se optimalna transformacija T_k koja minimizira udaljenost između parova korespondentnih točaka (q_i, p_i)

4. **Ažuriranje transformacije**: Nova transformacija T_{k+1} kombinira prethodnu transformaciju T_k s trenutno procijenjenom transformacijom T_{opt}, čime se oblak Q iterativno približava oblaku P.

5. **Kriterij zaustavljanja**: Algoritam se ponavlja dok promjena funkcije troška između uzastopnih iteracija ne postane manja od unaprijed definirane tolerancije:

$$|E(T_k) - E(T_{k-1})| < \epsilon. \tag{3.3}$$

   Ako se konvergencija ne postigne unutar maksimalnog broja iteracija, algoritam se zaustavlja i vraća zadnje razmatrano rješenje.

3.1.3. Prednosti i ograničenja

ICP je široko korišten zbog svoje jednostavnosti i robusnosti u kontroliranim uvjetima. Međutim, algoritam ima nekoliko ograničenja:

- Osjetljivost na početnu aproksimaciju - loša inicijalizacija može dovesti do konvergencije u lokalni minimum.
- Ograničena otpornost na šum i stršeće točke (engl. *outliers*).

- Računalna složenost - pronalaženje najbližih točaka zahtijeva značajnu računalnu snagu za velike oblake.

Unatoč tim ograničenjima, ICP ostaje temeljni algoritam u registraciji oblaka točaka i često se koristi kao referenca pri razvoju i evaluaciji novih metoda.

3.1.4. Proširenja algoritma

Razvijene su brojne varijante ICP-a kako bi se prevladali njegovi nedostaci:

- **Point-to-plane ICP** [1]: Umjesto euklidske udaljenosti između točaka, koristi udaljenost točke do ravnine, što povećava preciznost u slučajevima s gustom strukturom oblaka.

- **Robust ICP**: Primjenjuje robusne funkcije troška za smanjenje utjecaja odstupajućih točaka [1].

- **Parallel ICP**: Koristi paralelne računalne resurse kako bi ubrzao procese pronalaženja korespondencija i procjene transformacije [7].

Ova proširenja omogućuju veću otpornost na šum i nepravilnosti u podacima, kao i poboljšanu računalnu učinkovitost, čime ICP ostaje relevantan u modernim aplikacijama. Iako postoje poboljšanja, u ovom radu fokusirati ćemo se na rezultate i ograničenja temeljnog ICP algoritma.

3.2. Deep Closest Point (DCP): Kombinacija klasičnih metoda i dubokog učenja

Deep Closest Point (DCP) [2] model temeljen na dubokom učenju predstavlja suvremeni pristup registraciji oblaka točaka. Za razliku od klasičnih metoda poput iterativne najbliže točke, DCP koristi neuronske mreže za učenje značajki i primjenjuje mehanizam pažnje kako bi precizno uspostavio korespondencije između točaka dvaju oblaka. Na temelju tih korespondencija procjenjuje se rigidna transformacija, koja uključuje rotaciju i translaciju potrebne za njihovo poravnanje.

3.2.1. Osnovni principi modela

DCP uzima dva oblaka točaka kao ulaz: polazni oblak $P = \{p_1, p_2, \dots, p_N\}$ i dolazni oblak $Q = \{q_1, q_2, \dots, q_N\}$, gdje su $p_i, q_i \in \mathbb{R}^3$. Model zatim procjenjuje transformaciju $T = [R|t]$, gdje $R \in \mathbb{R}^{3\times3}$ predstavlja rotaciju, a $t \in \mathbb{R}^3$ translaciju. Ova transformacija poravnava oblak Q s oblakom P, što omogućuje registraciju dvaju oblaka [2].

3.2.2. Ključni koraci algoritma

DCP proces uključuje tri glavna koraka: ekstrakciju značajki, uspostavljanje korespondencija i procjenu transformacije.

Ekstrakcija značajki

Model koristi neuronsku mrežu f_θ za mapiranje ulaznih točaka u značajke koje sažimaju geometrijske i semantičke informacije oblaka:

$$F_P = f_\theta(P), \quad F_Q = f_\theta(Q), \tag{3.4}$$

gdje F_P i F_Q predstavljaju skupove značajki za oblake P i Q. Ekstrakcija značajki omogućuje smanjenje složenosti problema i pruža robusniju reprezentaciju podataka. DCP koristi arhitekture temeljene na PointNetu koje su neosjetljive na rotacije, translacije i permutacije točaka.

Uspostavljanje korespondencija

Za razliku od klasičnih metoda koje se oslanjaju na euklidsku udaljenost, DCP koristi mehanizam pažnje kako bi generirao dinamične korespondencije između značajki dvaju oblaka. Matrica pažnje definira se kao:

$$A_{ij} = \text{softmax}(F_P F_Q^T), \tag{3.5}$$

gdje A_{ij} označava vjerojatnost da točka p_i odgovara q_j. Ovaj pristup omogućuje modelu da identificira ključne korespondencije i ignorira nebitne ili neispravne veze, čime se značajno povećava točnost i otpornost na šum u podacima.

Procjena transformacije

Procjena transformacije temelji se na minimizaciji udaljenosti između uspostavljenih korespondencija:

$$R, t = \arg\min_{R,t} \sum_{i=1}^{N} ||Rq_i + t - p_i||^2. \tag{3.6}$$

3.2.3. Arhitektura modela

Za ekstrakciju značajki iz oblaka točaka koristi se **PointNet** [3]. Ove mreže konstruiraju reprezentacije točaka koje se kasnije koriste za računanje transformacija. textbfPointNet koristi niz 1D konvolucijskih slojeva kako bi izdvojio značajke iz točaka:

$$x' = \text{ReLU}(\text{BN}(\text{Conv1D}(x))) \tag{3.7}$$

Nakon ekstrakcije značajki, koristi se transformatorski model kako bi se uparile značajke između izvora i cilja. Ključni mehanizam je višeglavi mehanizam pažnje (engl. *Multi-Head Attention*).

Nakon uparivanja značajki, procjenjuje se optimalna transformacija između dva skupa točaka. Ako je omogućena ciklična provjera, model dodatno izračunava transformaciju u suprotnom smjeru kako bi osigurao konzistentnost.

3.2.4. Prednosti i nedostaci

Prednosti:

- Robustnost na šum i stršeće točke (engl. *outliers*).
- Automatsko učenje značajki invarijantnih na rotaciju i permutaciju.
- Poboljšana točnost u odnosu na klasične metode.

Nedostaci:

- Veća računalna složenost i vrijeme učenja.
- Zahtijeva kvalitetne skupove podataka za učinkovito učenje.

3.2.5. Primjena u eksperimentu

Model DCP [2] treniran je na skupu podataka ModelNet40 [5], koristeći unaprijed određene arhitekture i hiperparametre. Evaluacija je provedena u scenarijima s različitim razinama šuma i djelomičnim podudaranjem, pri čemu je DCP pokazao značajne prednosti u usporedbi s klasičnim pristupima poput ICP-a.

3.3. PointNetLK: Globalne značajke i iterativna optimizacija

PointNetLK model predstavlja spoj klasične Lucas-Kanadeove (LK) optimizacije i moderne arhitekture PointNet za registraciju oblaka točaka. Ovaj model koristi duboko učenje za ekstrakciju globalnih značajki iz oblaka i iterativnu optimizaciju za procjenu rigidne transformacije.

3.3.1. Osnovni principi modela

PointNetLK uzima dva ulazna oblaka točaka: polazni oblak $P = \{p_1, p_2, \dots, p_N\}$ i dolazni oblak $Q = \{q_1, q_2, \dots, q_N\}$. Cilj modela je pronaći rigidnu transformaciju $T = [R|t]$, gdje $R \in \mathbb{R}^{3\times3}$ predstavlja rotaciju, a $t \in \mathbb{R}^3$ translaciju. Transformacija T minimizira razliku između globalnih značajki oblaka P i Q.

3.3.2. Ključni koraci algoritma

PointNetLK slijedi iterativni postupak koji uključuje ekstrakciju značajki, optimizaciju i ažuriranje transformacije.

Ekstrakcija značajki

Oblaci P i Q prolaze kroz PointNet arhitekturu f_θ, koja mapira točke u globalne značajke:

$$F_P = f_\theta(P), \quad F_Q = f_\theta(Q). \tag{3.8}$$

Globalne značajke F_P i F_Q sažimaju geometrijske informacije cijelog oblaka, pružajući reprezentaciju koja je invariantna na permutacije i translacije dolaznih točaka. Ovaj ko-

rak omogućuje modelu da se usredotoči na globalni kontekst, umjesto na lokalne značajke, što je posebno važno za nepravilno strukturirane podatke.

Lucas-Kanade optimizacija

Nakon ekstrakcije značajki, razlika između F_P i F_Q koristi se kao kriterij za procjenu transformacije T:

$$T = \arg\min_T \|F_P - F_Q(T)\|^2. \tag{3.9}$$

Ova optimizacija iterativno ažurira T kako bi minimizirala razliku između značajki. Lucasov i Kanadeov pristup koristi gradijentne informacije kako bi ubrzao konvergenciju prema optimalnoj transformaciji.

Iterativno poravnanje

Transformacija T ažurira se u svakoj iteraciji dok razlika između značajki F_P i F_Q ne postane manja od unaprijed definirane tolerancije:

$$\|F_P - F_Q(T)\|^2 < \epsilon. \tag{3.10}$$

Iterativni postupak omogućuje modelu prilagodbu složenim geometrijskim odnosima između oblaka, osiguravajući točno poravnanje čak i u prisutnosti manjih odstupanja.

3.3.3. Arhitektura modela

Za ekstrakciju značajki koristi se **PointNet**, koji konstruira reprezentaciju oblaka točaka pomoću višeslojnih perceptrona [3]. Lucas-Kanadeove metoda koristi iterativnu optimizaciju kako bi pronašla najbolju rigidnu transformaciju između dva skupa točaka. Korekcija transformacije se računa kao:

$$\Delta g = -J^{\dagger} r. \tag{3.11}$$

gdje je J Jakobijan matrica značajki, r rezidual između dolaznog i polaznog skupa, a $J^{\dagger}$ pseudoinverz Jakobijana [8]. Transformacija između dva skupa točaka se određuje

pomoću eksponencijalne mape iz SE(3):

$$g = \exp(\xi). \tag{3.12}$$

gdje je ξ vektor pomaka.

3.3.4. Prednosti i nedostaci

Prednosti:

- Globalni pristup - omogućuje točnost i robusnost pri radu s nepravilno strukturiranim podacima.
- Jednostavna implementacija zahvaljujući kombinaciji PointNet-a i klasične optimizacije.
- Prikladan za velike skupove podataka zbog sažimanja u globalne značajke.

Nedostaci:

- Osjetljivost na šum i stršeće točke (engl. *outliers*), osobito bez dodatnih prilagodbi.
- Potreba za preciznim podešavanjem hiperparametara kako bi se osigurala optimalna izvedba.

3.3.5. Primjena u eksperimentu

PointNetLK evaluiran je na standardnom skupu podataka ModelNet40, pri čemu su analizirani scenariji s različitim razinama šuma i djelomičnim podudaranjem oblaka. Model je pokazao visoku učinkovitost u složenim slučajevima, uz zadržavanje relativno niske računalne složenosti zahvaljujući globalnim značajkama.

4. Eksperimentalna postavka i rezultati

Cilj eksperimentalnog dijela ovog rada je evaluirati odabrane metode za registraciju oblaka točaka. Ova sekcija opisuje računalno okruženje korišteno za treniranje modela, definirani skup podataka te obrazlaže izbor tog skupa podataka za provođenje eksperimenata.

4.1. Računalno okruženje

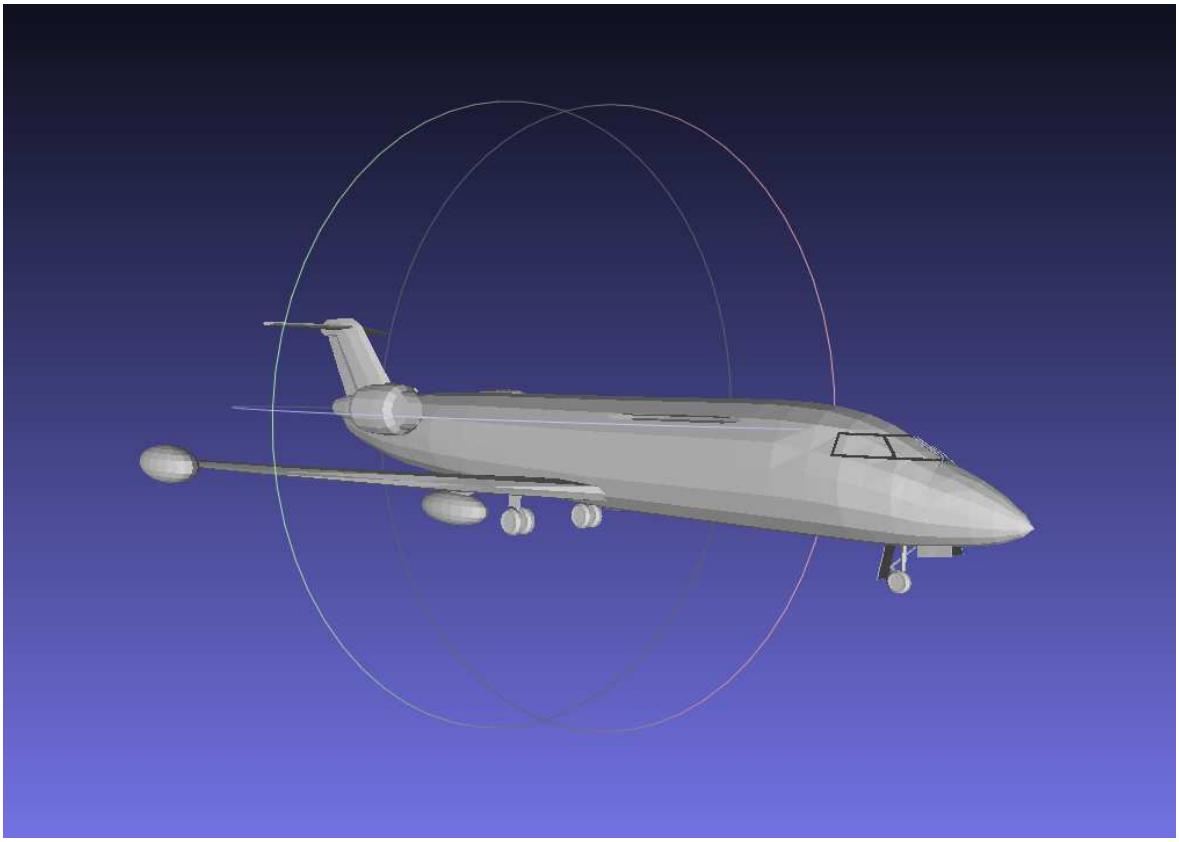
Eksperimenti su provedeni na računalu s konfiguracijom koja omogućuje učinkovito treniranje modela temeljenih na dubokom učenju. Detalji konfiguracije su sljedeći:

- **Grafička procesorska jedinica (GPU)**: NVIDIA RTX 3060 s 12 GB memorije.
- **Procesor (CPU)**: Intel i7-13700K, 16 jezgri, 24 dretvi.
- **Radna memorija (RAM)**: 64 GB DDR4.
- **Operacijski sustav (OS)**: Windows 11 Pro.
- **Razvojni alati i biblioteke**:
  - `Python 3.10.10` s paketima `PyTorch 2.5.1`, `NumPy 1.26.3` i `Matplotlib 3.9.2`.
  - `CUDA 11.7` za akceleraciju na GPU-u.
  - `Trimesh` za generiranje i obradu oblaka točaka iz 3D oblika.

Ova konfiguracija omogućila je stabilno treniranje modela uz paralelnu obradu velikih skupova podataka. Svi eksperimenti su reproducibilni.

4.2. Skup podataka: ModelNet40

Za evaluaciju odabranih metoda korišten je skup podataka **ModelNet40** [5], koji je standardni referentni skup u domeni registracije oblaka točaka. ModelNet40 sastoji se od 3D modela objekata iz 40 različitih kategorija, uključujući stolice, stolove, automobile, zrakoplove i zgrade. Ovi modeli predstavljeni su u obliku trodimenzionalnih modela (engl. *meshes*), što omogućuje njihovu transformaciju u oblake točaka za potrebe analize i registracije [9].



Slika 4.1. Prikaz jednog od modela iz kategorije zrakoplova u alatu MeshLab (`https://www.meshlab.net/`)

4.2.1. Specifikacije skupa podataka

Skup ModelNet40 [5] sadrži:

- **Ukupan broj modela**: 12,311 trodimenzionalnih modela, gdje svaki model predstavlja površinsku geometriju objekta putem skupa povezanih poligona.
- **Skup za učenje**: 9,843 modela korištenih za učenje dubokog modela.

- **Testni skup**: 2,468 modela korištenih za procjenu performansi dubokih modela na nepoznatim podacima.

Ovi trodimenzionalni modeli omogućuju generiranje oblaka točaka uzorkovanjem površinskih točaka s objekata, čime se dobiva ravnomjerno distribuirani oblak točaka koji opisuje geometriju modela. Takvi oblaci točaka služe kao ulazni podaci za algoritme registracije, omogućujući sustavnu analizu i evaluaciju njihovih performansi. Svaki model uzorkovan je u oblak točaka s 1024 točke kako bi se smanjila složenost i omogućilo korištenje u algoritmima za registraciju. Referentni radovi [1] [2] sadrže eksperimente s 512, 1024 i 2048 točaka.

4.2.2. Razlozi za odabir ModelNet40 skupa

Odabir skupa ModelNet40 temelji se na nekoliko ključnih faktora:

- **Standardizacija**: ModelNet40 često se koristi u literaturi kao referentni skup podataka, što omogućuje usporedbu s postojećim radovima.
- **Raznolikost**: Skup podataka uključuje širok raspon objekata različitih oblika i veličina, što omogućuje procjenu performansi metoda na raznovrsnim geometrijskim strukturama.
- **Kvaliteta podataka**: Podaci su konzistentni i sadrže trodimenzionalne modele visoke kvalitete, što smanjuje potrebu za dodatnim preprocesiranjem.
- **Praktičnost**: Skup je slobodno dostupan (besplatan) i pripremljen za uporabu u različitim eksperimentima, uključujući registraciju oblaka točaka.

4.2.3. Generiranje oblaka točaka

Za sve modele iz ModelNet40 skupa, oblak točaka generiran je korištenjem uniformnog uzorkovanja s površine 3D mreže. Ovaj postupak osigurava ravnomjernu distribuciju točaka, čime se minimizira utjecaj gustoće točaka na performanse registracijskih metoda. Primjer koda korištenog za generiranje oblaka točaka prikazan je u nastavku:

```
import trimesh
```

```
mesh = trimesh.load_mesh('model.obj')
points = mesh.sample(1024)
```

Normalizacija oblaka točaka provodi se kako bi svi modeli bili centrirani u ishodištu i skalirani unutar jedinične sfere, što osigurava konzistentnost ulaznih podataka za sve metode:

```
points -= points.mean(axis=0)
points /= np.linalg.norm(points, axis=1).max()
```

Rezultati su zatim spremljeni u .npz datoteke, što omogućava pohranu više numpy nizova u jednu datoteku uz kompresiju koja smanjuje veličinu podataka:

```
data = np.array(all_data, dtype='float32')
labels = np.array(all_label, dtype='int64')
np.savez_compressed(file, data=data, labels=labels)
```

Ovime je generiranje oblaka točaka završeno.

4.3. Rezultati metoda

U ovom dijelu prikazani su rezultati odabranih metoda za registraciju oblaka točaka: Iterativna najbliža točka (ICP) [1], Deep Closest Point (DCP) [2] i PointNetLK [3].

Rezultati su evaluirani pomoću metrika relativne pogreške rotacije (engl. *Relative Rotation Error, RRE*) i relativne pogreške translacije (engl. *Relative Translation Error, RTE*), koje su standardne metrike u literaturi za registraciju oblaka točaka. Ove metrike omogućuju kvantitativnu procjenu točnosti registracije, pri čemu niže vrijednosti označavaju bolje performanse.

Relativna pogreška rotacije (RRE) mjeri kutnu razliku između matrica rotacije dobivene modelom ($\hat{R}$) i točne rotacije (R^*). Formula za RRE dana je izrazom [10]:

$$\mathrm{RRE} = \arccos\left(\frac{\mathrm{trace}(\hat{R}^T R^*) - 1}{2}\right) \cdot \frac{180}{\pi} \tag{4.1}$$

Ta je formula implementirana u Pythonu i izgleda ovako:

```
trace = np.trace(np.dot(pred.T, gt))
rre_angle = np.arccos(np.clip((trace - 1) / 2, -1.0, 1.0))
rre_angle_deg = rre_angle * (180 / np.pi)
```

Formula (4.1) daje rezultat u stupnjevima, a manja vrijednost RRE ukazuje na točniju procjenu rotacije. Relativna pogreška translacije (RTE) mjeri udaljenost između vektora translacije dobivenog modelom ($\hat{t}$) i točne translacije (t^*). Formula za RTE je [10]:

$$\text{RTE} = \|t^* - \hat{t}\| \tag{4.2}$$

te ekvivalent u kodu:

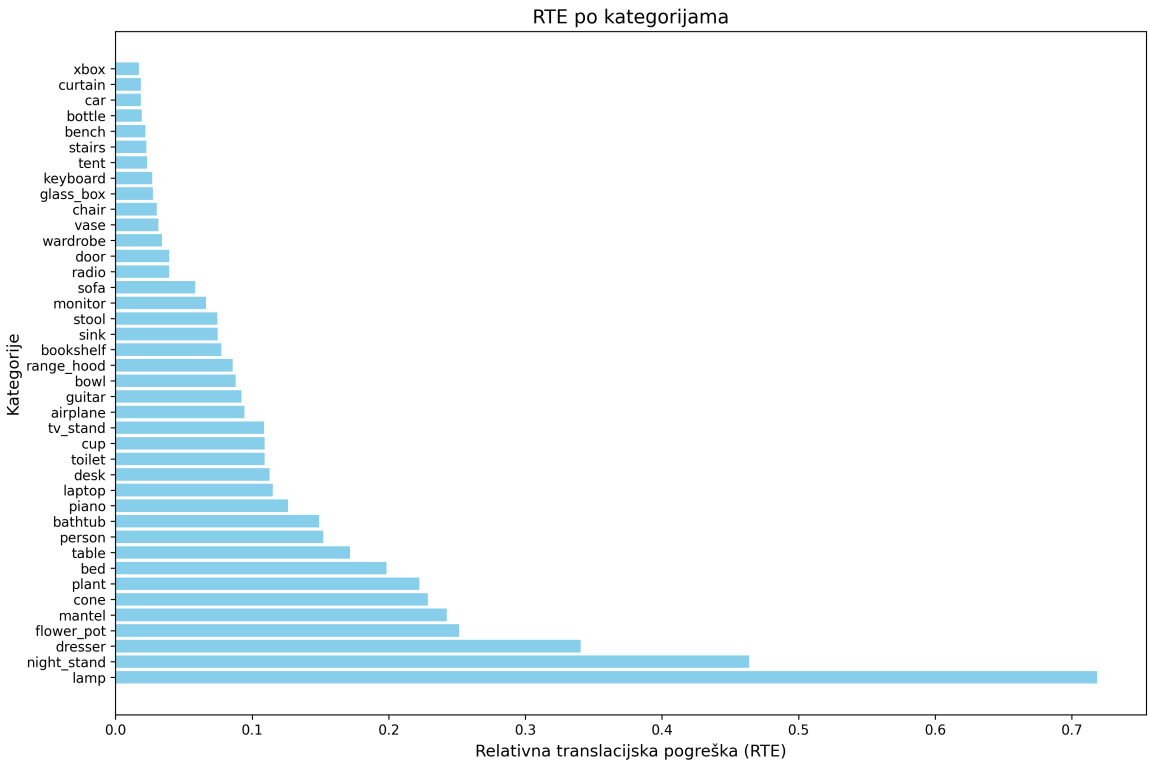
```
translational_diff = torch.abs(translation_ab_pred - translation_ab)
translational_error = torch.mean(translational_diff,
                                 dim=1).detach().cpu().numpy()
```

Metrika (4.2) izražava razliku u translaciji u metričkim jedinicama i ključna je za procjenu preciznosti pozicioniranja oblaka točaka.

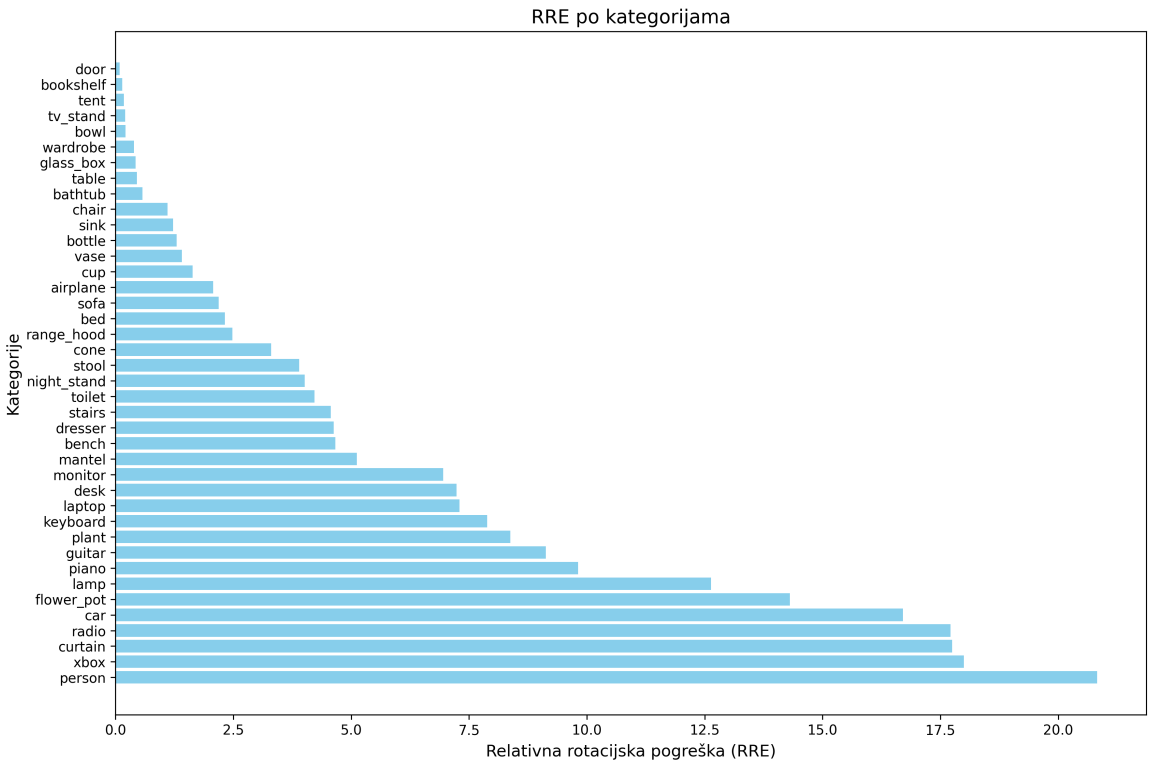
4.3.1. Iterativna najbliža točka (ICP)

Metoda ICP pokazala je očekivano ponašanje u eksperimentima, s rezultatima koji snažno ovise o kvaliteti početne aproksimacije transformacije. ICP je pokazao značajne prednosti u scenarijima s dobro poravnatim oblacima točaka, dok je u prisutnosti djelomičnog preklapanja ili kompleksnih geometrijskih struktura postigao ograničene rezultate.

Rezultati po kategorijama omogućuju detaljan uvid u performanse ICP-a.



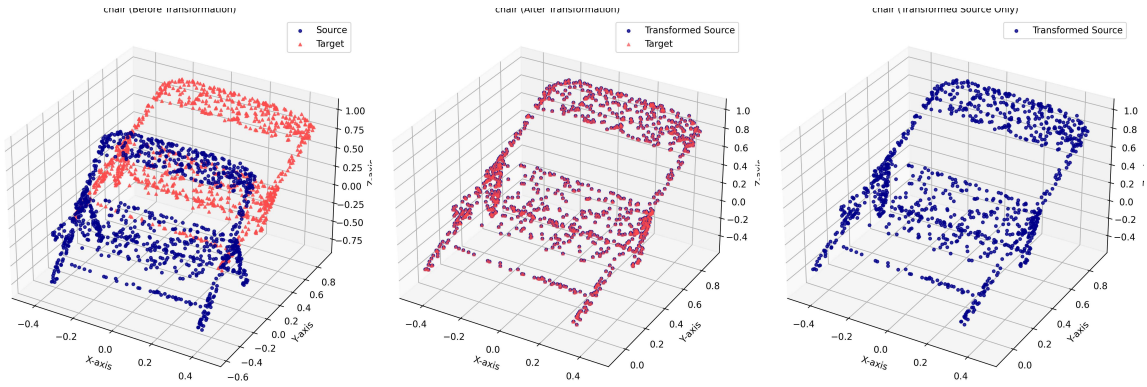
Slika 4.2. RTE po kategoriji za metodu ICP.



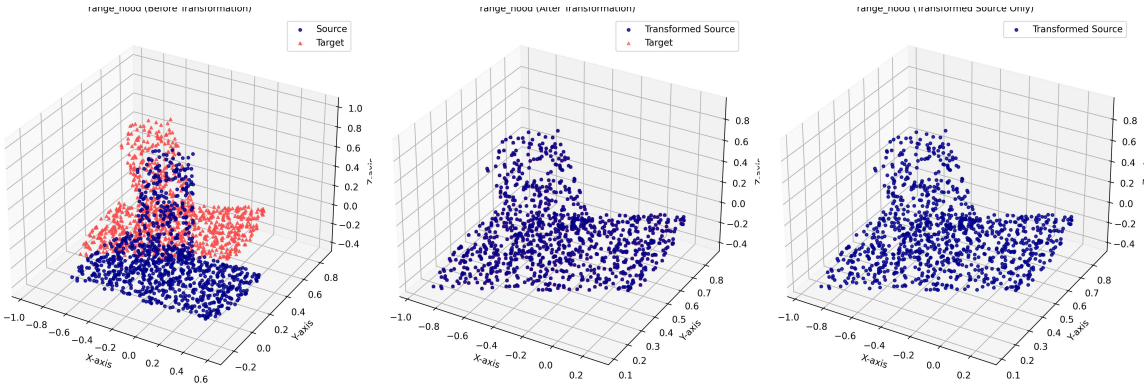
Slika 4.3. RRE po kategoriji za metodu ICP.

U kategorijama s jednostavnim i pravilnim geometrijskim strukturama, ICP je posti-

gao minimalne pogreške. Na primjer, kategorija *Chair* ima objekte s pravilnim rubovima i simetričnim oblicima, što olakšava prepoznavanje korespondencija. Slično, u kategoriji *Range Hood*, jasne konture i dobro definirane značajke omogućuju algoritmu precizno poravnanje.

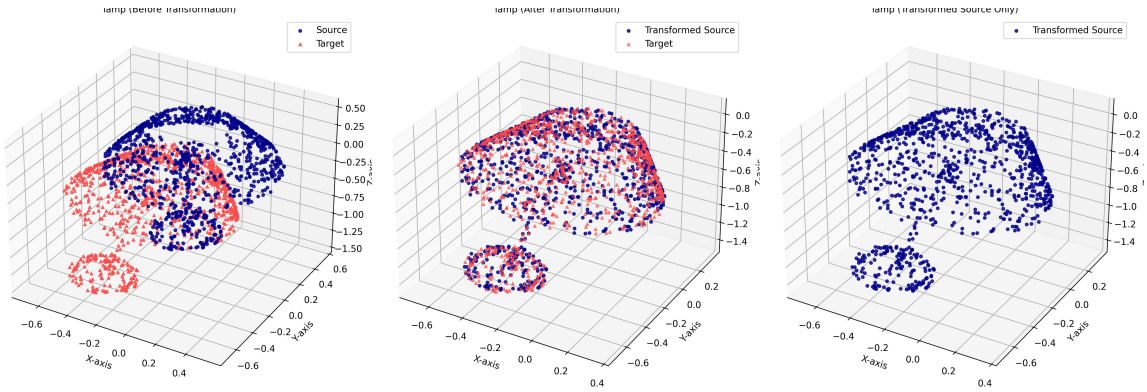


Slika 4.4. Vizualizacija registracije oblaka točaka za kategoriju *Chair* koristeći metodu ICP.

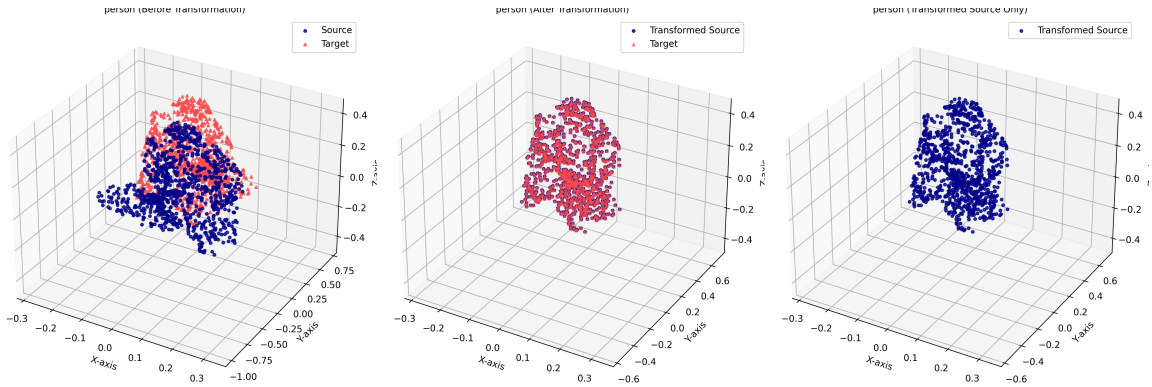


Slika 4.5. Vizualizacija registracije oblaka točaka za kategoriju *Range Hood* koristeći metodu ICP.

Najlošiji rezultati postignuti su u kategorijama s kompleksnim i višeznačnim strukturama. Na primjer, kategorija *Lamp* uključuje objekte s tankim komponentama i većinom asimetričnim oblicima, što otežava preciznu procjenu transformacije. U kategoriji *Person*, složene geometrijske karakteristike i više ravnina često uzrokuju pogrešno usklađivanje.



Slika 4.6. Vizualizacija registracije oblaka točaka za kategoriju *Lamp* za metodu ICP.



Slika 4.7. Vizualizacija registracije oblaka točaka za kategoriju *Person* za metodu ICP.

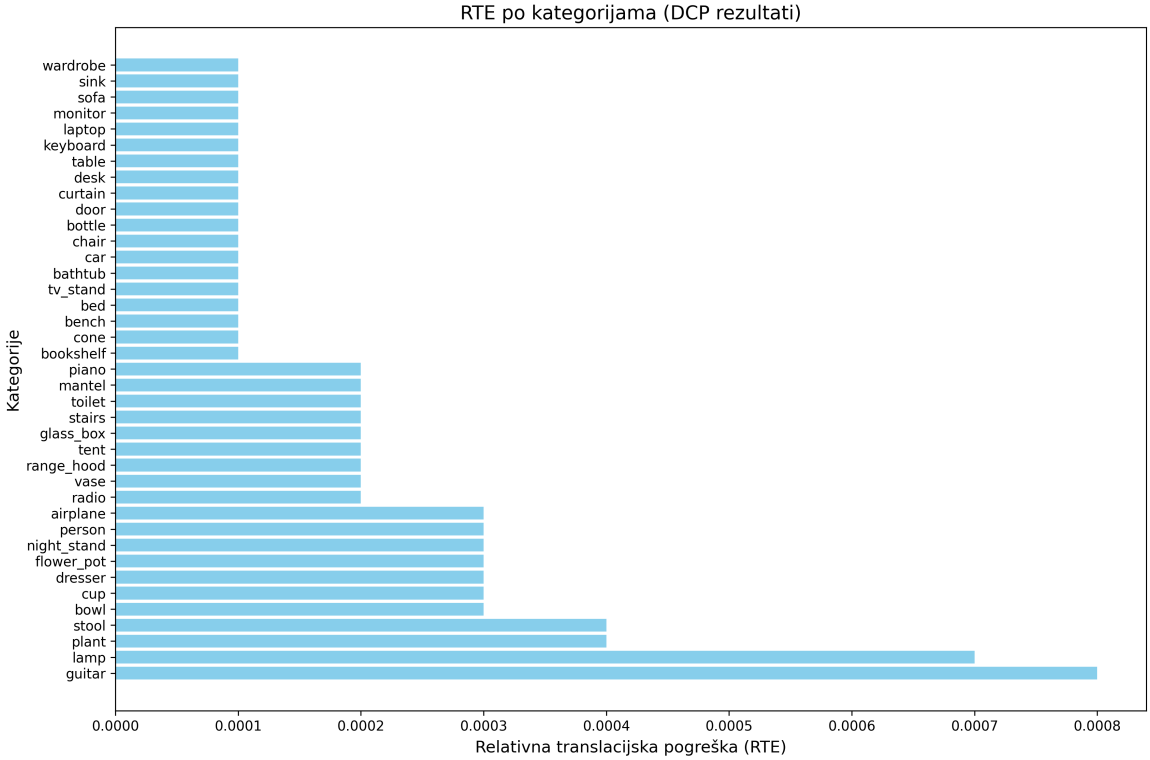
ICP pokazuje visoku učinkovitost u kategorijama s jednostavnim i pravilnim geometrijskim strukturama. Međutim, u prisutnosti kompleksnih oblika i geometrijske nepovezanosti, točnost registracije značajno opada. Vizualna analiza rezultata otkriva da, iako numeričke metrike poput RTE i RRE ukazuju na slabije performanse, u mnogim slučajevima registracija je dovoljno dobra za praktične primjene gdje nam preciznost nije najvažniji faktor.

4.3.2. Deep Closest Point (DCP)

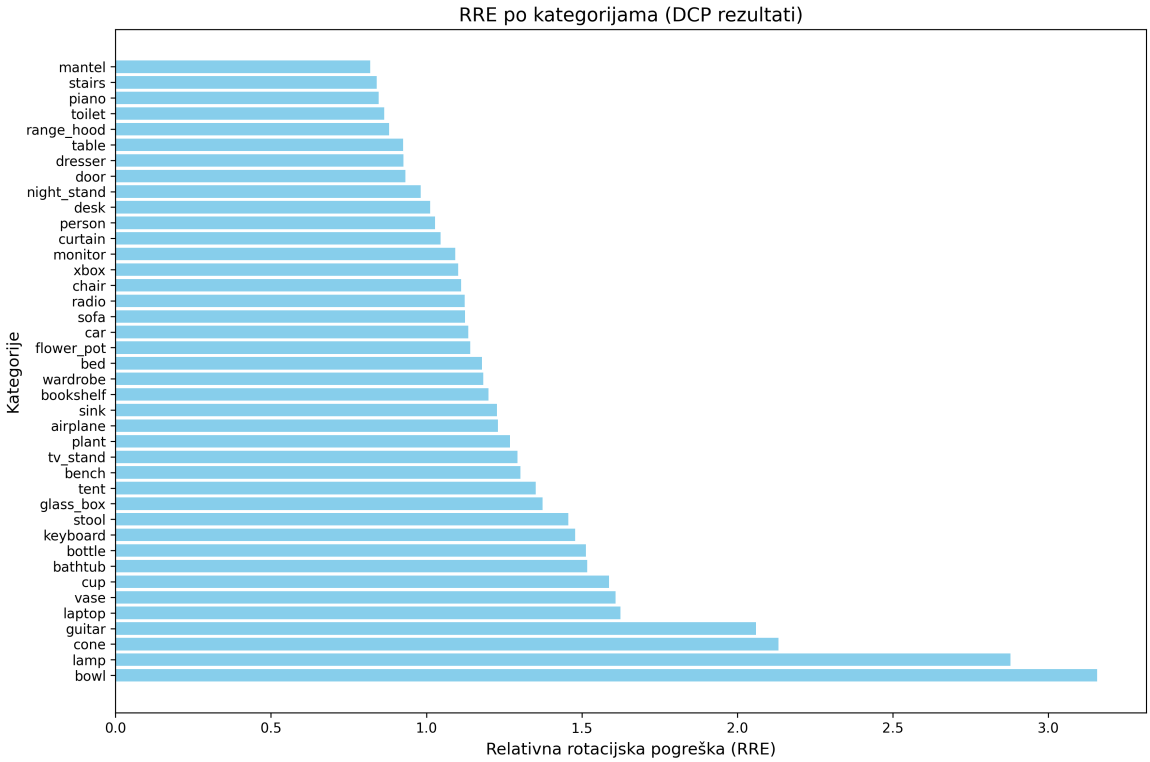
Metoda DCP pokazala je visoku otpornost na šum i preciznu registraciju čak i u scenarijima s djelomičnim preklapanjem. Snaga DCP-a leži u njegovoj sposobnosti automatskog učenja značajki i robusnom uspostavljanju korespondencija.

Rezultati po kategorijama omogućuju uvid u varijabilnost performansi DCP-a ovisno o specifičnim geometrijskim značajkama oblaka točaka. Analizom RTE i RRE identifi-

cirane su dvije naizgled bolje i dvije naizgled lošije kategorije.

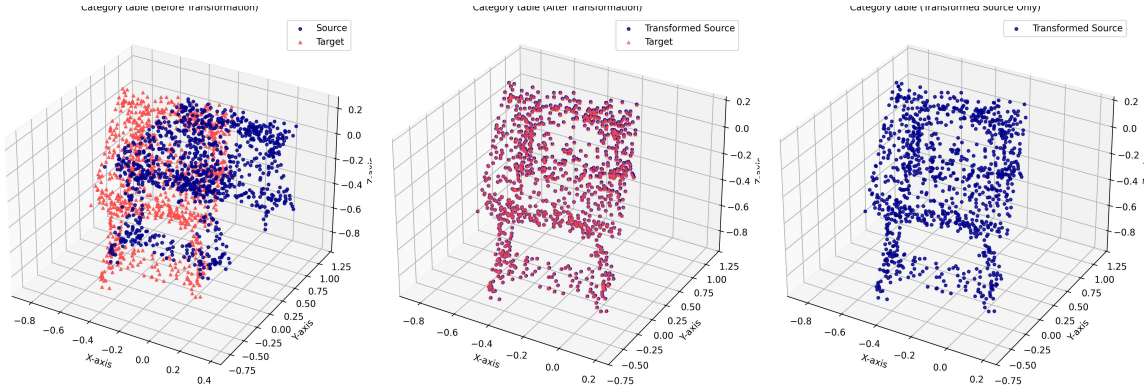


Slika 4.8. RTE po kategoriji za metodu DCP.

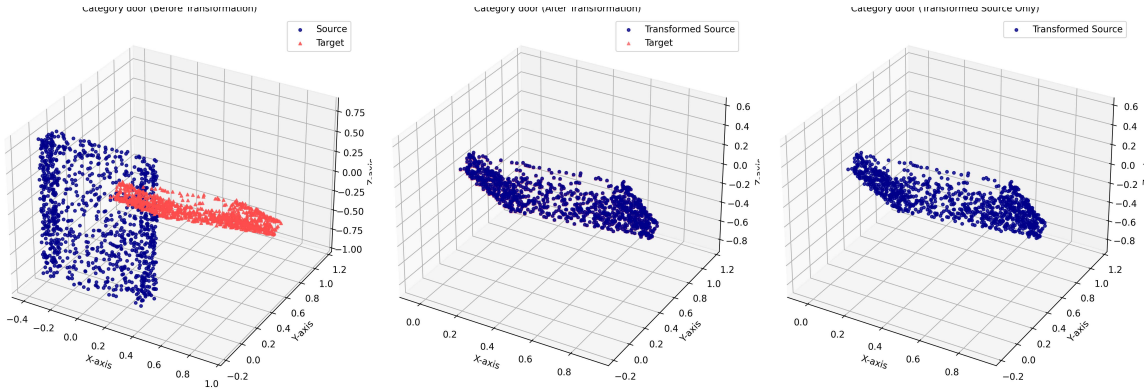


Slika 4.9. RRE po kategoriji za metodu DCP.

Bolje kategorije imaju dobro definirane značajke, što omogućuje modelu učinkovito učenje korespondencija. Na primjer, oblaci točaka u kategoriji *Table* i *Door* karakterizirani su pravilnim oblicima s visokim stupnjem simetrije i jasnim značajkama.

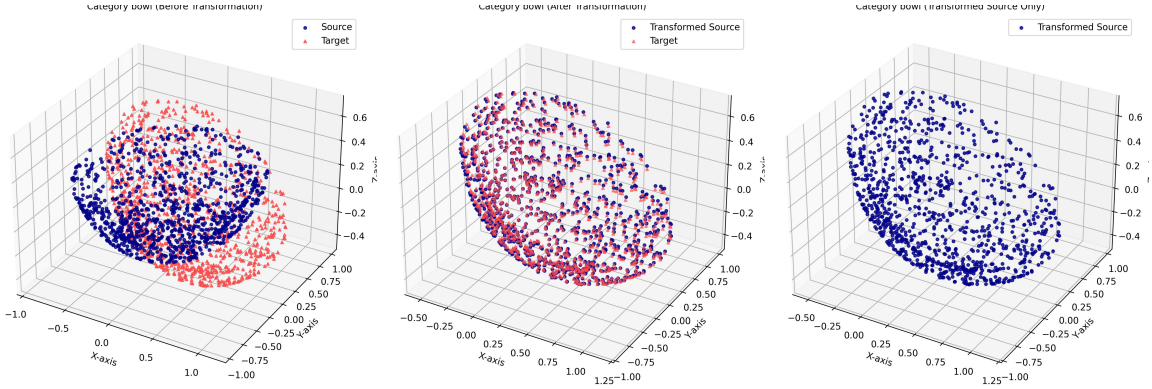


Slika 4.10. Vizualizacija registracije oblaka točaka za kategoriju *Table* za metodu DCP.

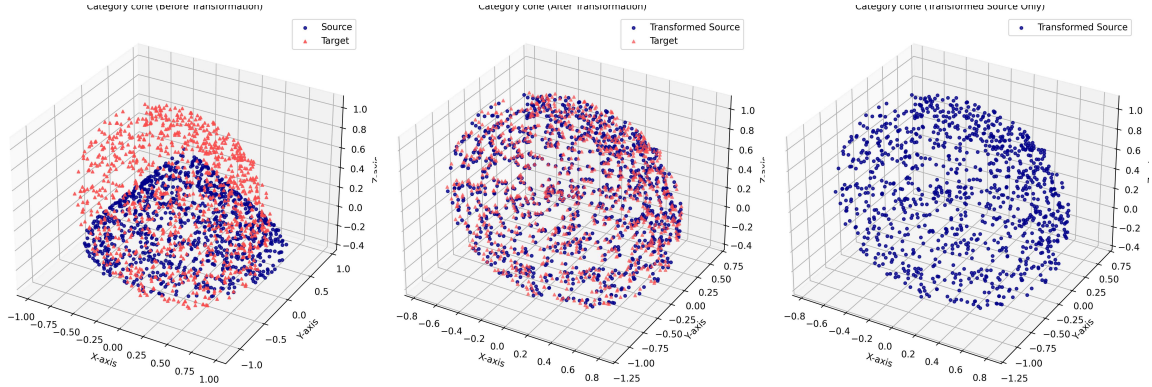


Slika 4.11. Vizualizacija registracije oblaka točaka za kategoriju *Door* za metodu DCP.

Lošije kategorije uključuju objekte s kružnim ili cilindričnim geometrijskim značajkama, koje često imaju višeznačne ili slične značajke, što otežava precizno uspostavljanje korespondencija. Na primjer, oblaci točaka za kategoriju *Bowl* često sadrže šuplje površine koje mogu uzrokovati zbunjujuće podudaranje, dok kategorija *Cone* ima sličan problem zbog svojeg cilindričnog oblika i sličnih rubnih značajki.



Slika 4.12. Vizualizacija registracije oblaka točaka za kategoriju *Bowl* za metodu DCP.



Slika 4.13. Vizualizacija registracije oblaka točaka za kategoriju *Cone* za metodu DCP.

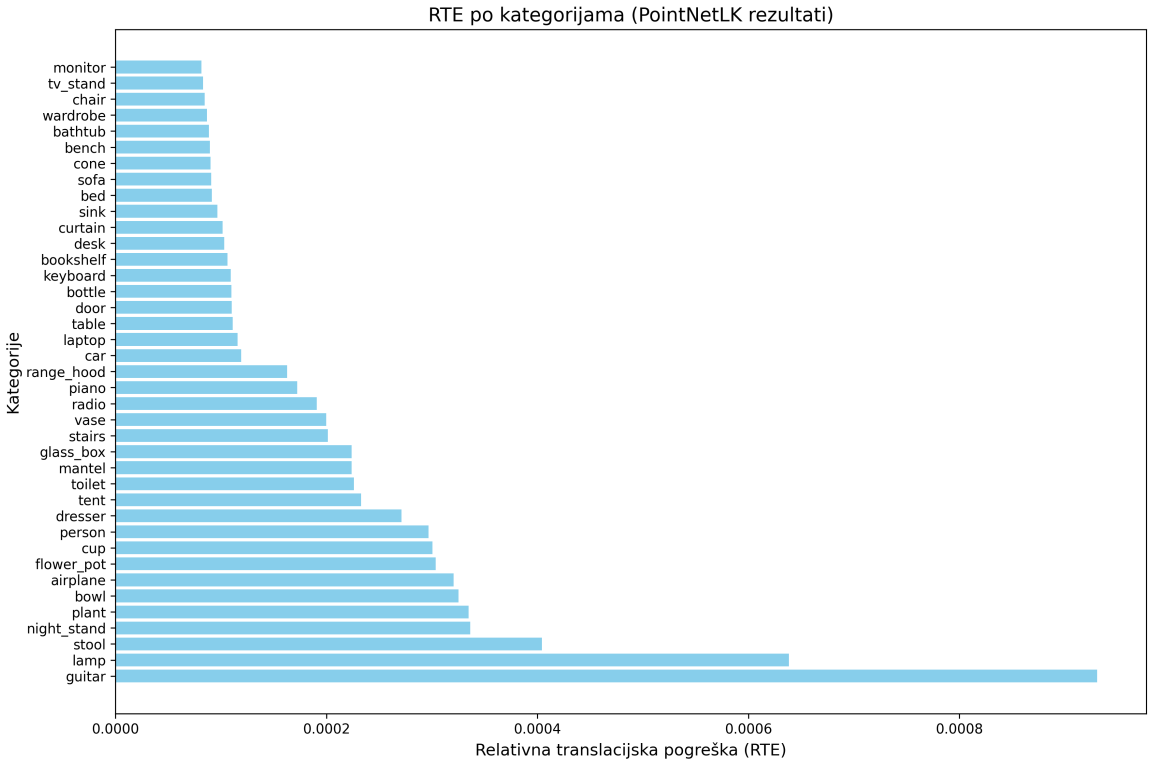
Analiza performansi metode DCP po kategorijama pružila je uvid u njezine prednosti i ograničenja. Kategorije poput *Table* i *Door* pokazuju visoku točnost jer prikazuju konzistentne oblike s jasnim konturama što omogućuje modelu da učinkovito uči značajke i uspostavi precizne korespondencije., dok se slabiji rezultati bilježe u složenijim kategorijama kao što su *Bowl* i *Cone*, takvi rezultati ukazuju na izazove s višeznačnošću i šupljim strukturama. Ovi objekti često uključuju složene unutarnje površine koje otežavaju preciznu procjenu korespondencija. Dodatno, geometrijska nepovezanost i nejasne granice između značajki mogu uzrokovati nepouzdane transformacije. Također, ovi rezultati naglašavaju važnost karakteristika oblika i geometrijske strukture za uspješnu registraciju.

Unatoč ovim ograničenjima, važno je istaknuti da numeričke metrike poput RTE-a i RRE-a ne odražavaju uvijek cjelokupnu kvalitetu registracije. Vizualna inspekcija rezultata za kategorije s višim greškama, poput *Bowl* i *Cone*, pokazuje da su registrirani oblaci točaka često dovoljno dobro usklađeni za praktične aplikacije. Na temelju toga, može se

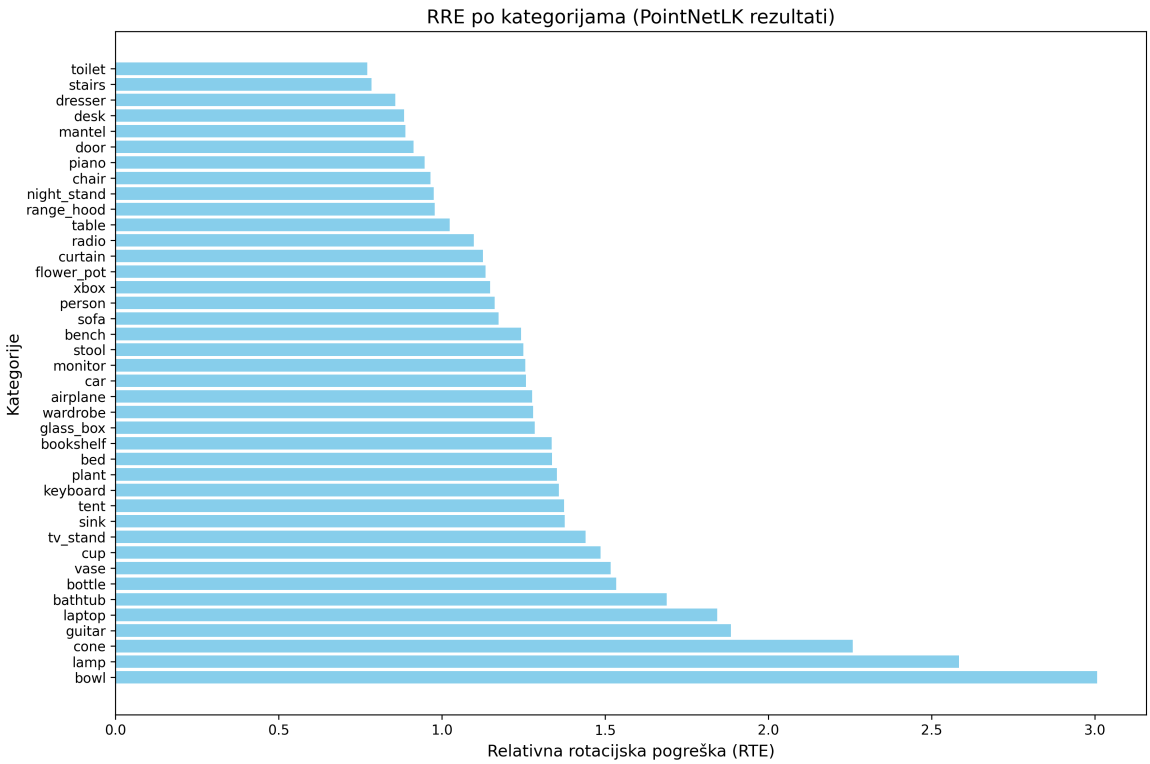
zaključiti da bi se ocjena performansi trebala temeljiti na kombinaciji numeričkih pokazatelja i vizualnih procjena.

4.3.3. PointNetLK

Metoda PointNetLK pokazala je rezultate koji su u velikoj mjeri usporedivi s metodom DCP, što ukazuje na to da obje metode dijele sličnu razinu robusnosti i preciznosti u zadacima registracije oblaka točaka. Sličnost u performansama proizlazi iz činjenice da obje metode koriste duboke neuronske mreže za učenje značajki.

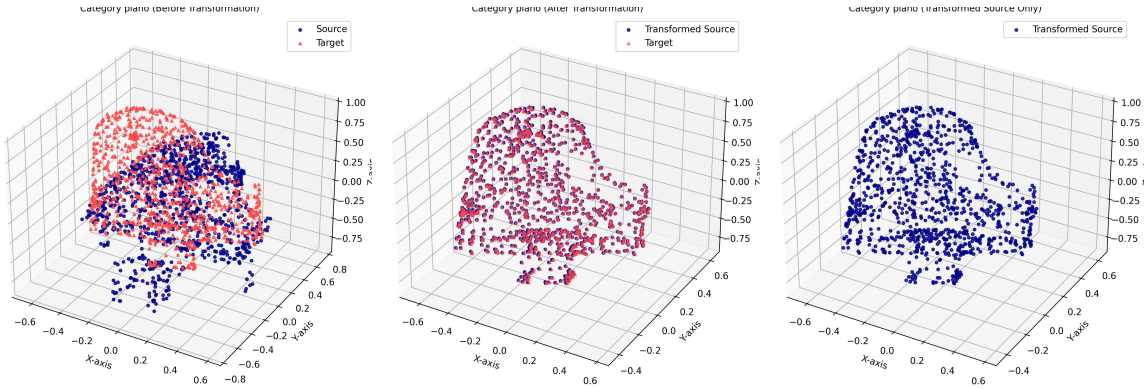


Slika 4.14. RTE po kategoriji koristeći metodu PointNetLK.

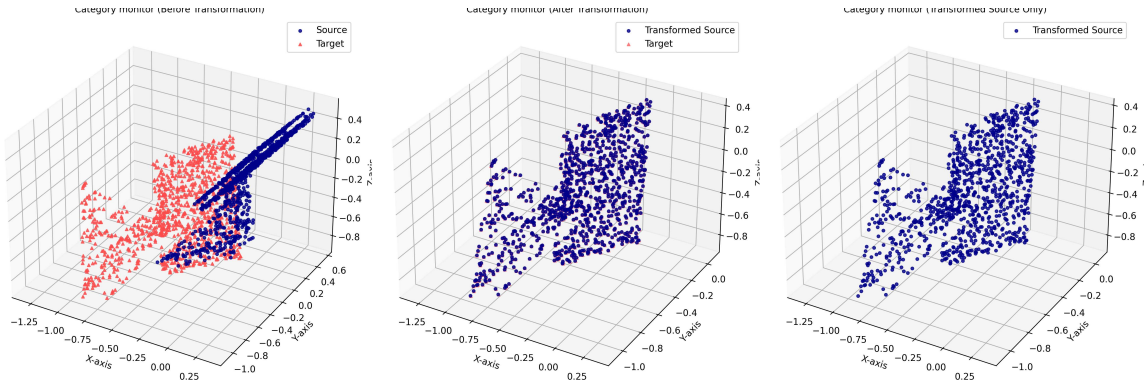


Slika 4.15. RRE po kategoriji koristeći metodu PointNetLK.

Bolje kategorije su one s dobro definiranim i konzistentnim geometrijskim značajkama. Na primjer, kategorije *Piano* i *Monitor* karakterizirane su pravilnim, kutnim oblicima i jasno definiranom strukturom, što omogućuje modelu učinkovito prepoznavanje i učenje značajki.

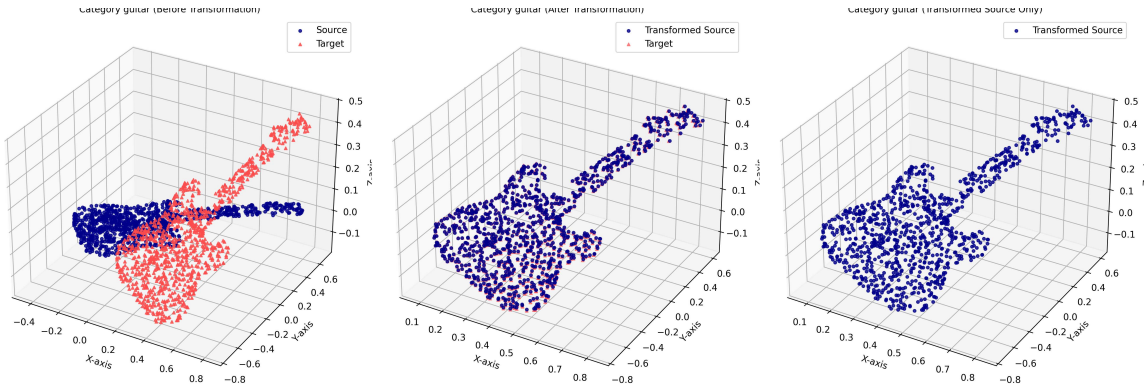


Slika 4.16. Vizualizacija registracije oblaka točaka za kategoriju *Piano* koristeći metodu PointNetLK.

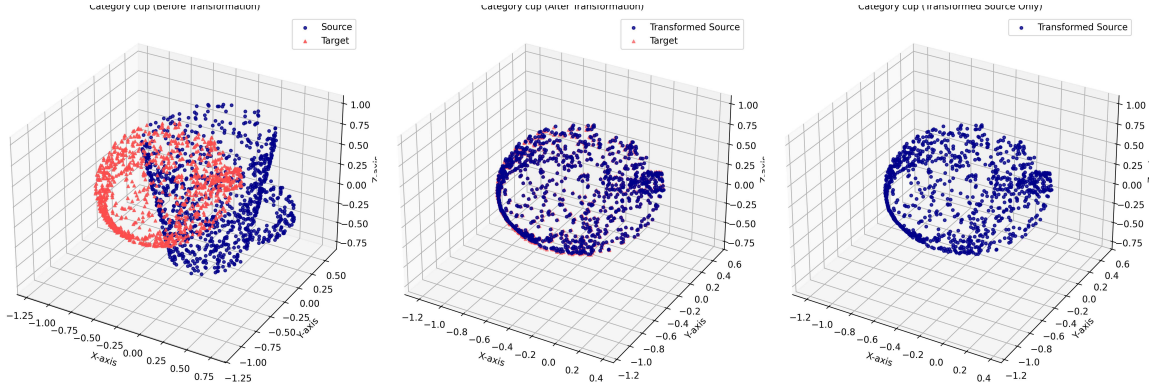


Slika 4.17. Vizualizacija registracije oblaka točaka za kategoriju *Monitor* koristeći metodu PointNetLK.

Nasuprot tome, lošije kategorije uključuju objekte s kompleksnim i višeznačnim geometrijskim značajkama. Kategorije *Guitar* i *Cup* karakterizirane su nepravilnim oblicima, tankim dijelovima (npr. vrat gitare) i zakrivljenim strukturama koje otežavaju precizno uspostavljanje korespondencija. Uz to, šuplje ili cilindrične značajke uzrokuju pogreške u registraciji.



Slika 4.18. Vizualizacija registracije oblaka točaka za kategoriju *Guitar* koristeći metodu PointNetLK.



Slika 4.19. Vizualizacija registracije oblaka točaka za kategoriju *Cup* koristeći metodu PointNetLK.

4.3.4. Usporedba rezultata

Rezultati su prikazani u tabličnom obliku, gdje su za svaku od triju metoda registracije 3D točaka prikazane vrijednosti relativne translacijske pogreške (RTE) i relativne rotacijske pogreške (RRE) za 40 kategorija. Kako bi istaknuo najpreciznije rezultate, unutar svake metode označena je najbolja kategorija s najmanjom vrijednošću RTE i RRE, pri čemu su te vrijednosti prikazane podebljanim tekstom.

Tablica 4.1. Tablična usporedba rezultata

Kategorija	ICP		DCP		PointNetLK	
	RTE	RRE	RTE	RRE	RTE	RRE
airplane	0.094500	10.3455	0.000300	1.2303	0.000305	1.2303
bathtub	0.148900	2.8643	**0.000100**	1.5169	0.000104	1.6737
bed	0.198400	11.6158	0.000100	1.1785	0.000109	1.2166
bench	0.021900	23.3364	0.000100	1.3026	0.000103	1.4888
bookshelf	0.077400	0.7318	0.000100	1.1997	0.000115	1.1562
bottle	0.019200	6.4953	0.000100	1.5124	0.000107	1.6848
bowl	0.087900	1.0673	0.000300	3.1573	0.000338	3.2093
car	0.018600	83.5336	0.000100	1.1349	0.000100	1.1101
chair	0.030300	5.5143	0.000100	1.1114	0.000110	1.1789
cone	0.228600	16.5302	0.000100	2.1319	0.000101	2.1965
cup	0.109100	8.1853	0.000300	1.5877	0.000350	1.5896
curtain	0.018600	88.7528	0.000100	1.0452	0.000108	1.1313
desk	0.112700	36.1746	0.000100	1.0118	0.000099	0.9636
door	0.039300	**0.4327**	0.000100	0.9327	0.000108	0.9522
dresser	0.340500	23.1427	0.000300	0.9259	0.000356	1.0500
flower pot	0.251600	71.5419	0.000300	1.1411	0.000296	1.2026
glass box	0.027300	2.1390	0.000200	1.3734	0.000216	1.3538
guitar	0.092200	45.6365	0.000800	2.0599	0.000909	2.3285
keyboard	0.027000	39.4209	0.000100	1.4779	0.000116	1.4490
lamp	0.718600	63.1864	0.000700	2.8783	0.000834	3.3076
laptop	0.115200	36.4892	0.000100	1.6236	0.000116	1.6051
mantel	0.242400	25.6100	0.000200	**0.8188**	0.000193	0.8152
monitor	0.066300	34.7767	0.000100	1.0928	**0.000098**	1.2301
night stand	0.463900	20.0770	0.000300	0.9811	0.000327	0.9465
person	0.152200	104.1412	0.000300	1.0281	0.000302	1.1072
piano	0.126400	49.0553	0.000200	0.8469	0.000193	0.8386
plant	0.222400	41.8944	0.000400	1.2684	0.000398	1.2542
radio	0.039400	88.5792	0.000200	1.1228	0.000232	1.2065
range hood	0.085700	12.4126	0.000200	0.8803	0.000240	0.8480
sink	0.074800	6.1209	0.000100	1.2265	0.000107	1.2451
sofa	0.058400	10.9423	0.000100	1.1241	0.000100	1.1438
stairs	0.022700	22.8232	0.000200	0.8406	0.000229	0.8942
stool	0.074600	19.4705	0.000400	1.4562	0.000444	1.5832
table	0.171700	2.2701	0.000100	0.9249	0.000106	1.0114
tent	0.023100	0.8803	0.000200	1.3517	0.000200	1.5392
toilet	0.109200	21.1184	0.000200	0.8647	0.000232	**0.8116**
tv_stand	0.108800	1.0211	0.000100	1.2931	0.000108	1.4614
vase	0.031500	7.0312	0.000200	1.6078	0.000225	1.7365
wardrobe	0.034000	1.9597	0.000100	1.1825	0.000111	1.1646
xbox	**0.017200**	89.9833	0.000208	1.1019	0.000227	1.1812

U kategoriji relativne translacijske pogreške (RTE), algoritam DCP postiže najbolje rezultate jer pokazuje najmanje vrijednosti u većini slučajeva, što znači da najpreciznije određuje translaciju između oblaka točaka. Također, u kategoriji relativne rotacijske pogreške (RRE), DCP se pokazuje kao najtočniji algoritam, jer u prosjeku ostvaruje niže vrijednosti u usporedbi s ICP-om i PointNetLK-om. ICP, iako tradicionalna metoda, ima veće pogreške u obje kategorije, dok PointNetLK u nekim slučajevima postiže niske vrijednosti RRE, ali nije dosljedno bolji od DCP-a.

5. Zaključak

U ovom diplomskom radu obrađena su tri algoritma za registraciju 3D oblaka točaka: klasični ICP [1], te dva duboka modela DCP [2] i PointNetLK [3].

Cilj istraživanja bio je analizirati performanse ovih algoritama te eksperimentalno utvrditi njihovu točnost na standardnim 3D skupovima podataka.

Rezultati eksperimenata pokazali su da DCP postiže najbolje rezultate u pogledu točnosti i otpornosti na različite transformacije. PointNetLK također je pokazao dobre performanse, no nešto slabije od DCP-a. S druge strane, ICP algoritam, iako široko korišten u praksi, ostvario je najslabije rezultate, posebice pri složenim transformacijama. Njegova glavna ograničenja uključuju sporiju konvergenciju i osjetljivost na početne uvjete.

S obzirom na brzi napredak dubokog učenja i računalne geometrije, može se očekivati daljnji razvoj još preciznijih i bržih metoda za registraciju 3D oblaka točaka. Novi modeli temelje se na poboljšanim arhitekturama neuronskih mreža koje bolje generaliziraju podatke i povećavaju robusnost algoritama na varijacije u ulaznim podatcima. Uz to, kombinacija klasičnih metoda poput ICP-a s dubokim modelima mogla bi rezultirati hibridnim rješenjima koja iskorištavaju prednosti oba pristupa.

Također, razvoj specijaliziranog hardvera, poput GPU-ova i TPU-ova, omogućava bržu obradu složenih neuronskih mreža, čime se dodatno povećava učinkovitost ovih algoritama.

Ovaj rad donosi nekoliko značajnih doprinosa u području registracije 3D oblaka točaka. Prvo, provedena je detaljna analiza i usporedba klasičnih i dubokih metoda za registraciju oblaka točaka, čime je omogućeno bolje razumijevanje prednosti i ograničenja svakog pristupa. Eksperimentalno je potvrđeno da modeli temeljeni na dubokom

učenju, poput DCP-a i PointNetLK-a, nadmašuju klasični ICP algoritam, osobito u kontekstu složenih transformacija i velikih podataka.

Ovi rezultati mogu poslužiti kao polazište za daljnja istraživanja i razvoj hibridnih metoda koje kombiniraju klasične i duboke pristupe s ciljem postizanja veće točnosti i učinkovitosti. Također, rad pruža smjernice za budući razvoj registracijskih metoda, uključujući mogućnosti poboljšanja kroz optimizaciju neuronskih arhitektura i integraciju naprednih modela strojnog učenja.

Zaključno, istraživanje registracije 3D oblaka točaka ostaje aktivno područje razvoja, a napredak u dubokom učenju i računalnoj obradi podataka ukazuje na potencijal za stvaranje novih, još učinkovitijih algoritama u budućnosti.

Literatura

[1] J. Zhang, Y. Yao, i B. Deng, “Fast and robust iterative closest point”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, str. 1–1, 2021. https://doi.org/10.1109/tpami.2021.3054619

[2] Y. Wang i J. M. Solomon, “Deep closest point: Learning representations for point cloud registration”, 2019. [Mrežno]. Adresa: https://arxiv.org/abs/1905.03304

[3] Y. Aoki, H. Goforth, R. A. Srivatsan, i S. Lucey, “Pointnetlk: Robust & efficient point cloud registration using pointnet”, 2019. [Mrežno]. Adresa: https://arxiv.org/abs/1903.05711

[4] X. Huang, G. Mei, J. Zhang, i R. Abbas, “A comprehensive survey on point cloud registration”, 2021. [Mrežno]. Adresa: https://arxiv.org/abs/2103.02690

[5] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, i J. Xiao, “Modelnet: A large-scale 3d cad model dataset”, https://modelnet.cs.princeton.edu/, 2015., accessed: 2025-02-10.

[6] J. Kosecka, “Introduction to the iterative closest point algorithm”, 2003., accessed: February 10, 2025. [Mrežno]. Adresa: https://cs.gmu.edu/~kosecka/cs685/cs685-icp.pdf

[7] T. D. Liu, F. Zhen Kong, M. He, X. M. Wu, i G. Fang Shao, “Parallel 3d icp based on conditionally constrained corresponding points and applications”, u *2021 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, sv. 1, 2021., str. 1–6. https://doi.org/10.1109/ICNSC52481.2021.9702130

[8] X. Li, J. K. Pontes, i S. Lucey, “Pointnetlk revisited”, 2021. [Mrežno]. Adresa: https://arxiv.org/abs/2008.09527

[9] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, i J. Xiao, “3d shapenets: A deep representation for volumetric shapes”, 2015. [Mrežno]. Adresa: https://arxiv.org/abs/1406.5670

[10] D. Bojanić, K. Bartol, J. Forest, T. Petković, i T. Pribanić, “Addressing the generalization of 3d registration methods with a featureless baseline and an unbiased benchmark”, *Machine Vision and Applications*, sv. 35, br. 41, March 2024. https://doi.org/10.1007/s00138-024-01510-w

Sažetak

Duboki modeli za registraciju oblaka točaka

Manuel Matošević

Registracija oblaka točaka jedan je od ključnih problema u obradi 3D podataka, a sastoji se od pronalaska rotacije i translacije koje omogućuju preklapanje dva oblaka točaka u jedinstvenu i povezanu cjelinu. Ovaj diplomski rad pruža pregled osnovnih koraka klasičnog postupka registracije 3D oblaka točaka, s posebnim naglaskom na Iterativna najbliža točka (Iterative Closest Point, ICP) algoritam, koji se često koristi za rješavanje ovog problema. Tijekom rada provedeno je istraživanje literature o primjeni dubokog učenja u registraciji 3D podataka. Identificirani su najvažniji modeli dubokog učenja te su podijeljeni u skupine prema njihovim glavnim karakteristikama. Iz svake skupine odabran je po jedan reprezentativni model, a za daljnje istraživanje odabrani su modeli DCP (Deep Closest Point) i PointNetLK. Ovi modeli uspoređeni su međusobno i s klasičnim ICP algoritmom koristeći skup podataka ModelNet40. Eksperimentalni dio rada obuhvaća evaluaciju učinkovitosti odabranih metoda prema kriterijima točnosti registracije. Rezultati istraživanja ističu prednosti i ograničenja klasičnih i dubokih pristupa registraciji oblaka točaka te pružaju smjernice za daljnji razvoj ovog područja. Zaključci rada doprinose boljem razumijevanju problema registracije oblaka točaka i ukazuju na potencijal primjene dubokog učenja za unaprjeđenje ovog procesa.

Ključne riječi: registracija oblaka točaka, iterativna najbliža točka (ICP), Deep Closest Point (DCP), PointNetLK

Abstract

Deep models for point cloud registration

Manuel Matošević

Point cloud registration is one of the key challenges in 3D data processing. It involves determining the rotation and translation that align two point clouds into a single, cohesive representation. This thesis provides an overview of the fundamental steps in the classical 3D point cloud registration process, with a particular focus on the Iterative Closest Point (ICP) algorithm, which is widely used to address this problem. A literature review was conducted to explore the application of deep learning in 3D point cloud registration. The most significant deep learning models were identified and categorized based on their main characteristics. From each category, one representative model was selected, with DCP (Deep Closest Point) and PointNetLK chosen for further investigation. These models were compared against each other and the classical ICP algorithm using the ModelNet40 dataset. The experimental part of the thesis includes an evaluation of the selected methods based on registration accuracy. The results highlight the strengths and limitations of both classical and deep learning-based approaches to point cloud registration, providing insights for further development in this area. The findings of this thesis contribute to a better understanding of the point cloud registration problem and emphasize the potential of deep learning to enhance this process.

Keywords: point cloud registration, Iterative Closest Point (ICP), Deep Closest Point (DCP), PointNetLK