

Rekonstrukcija objekta in poravnava oblakov točk

Martin Knap¹, Janez Lapajne¹

Univerza v Ljubljani, Fakulteta za elektrotehniko, Tržaška 25, 1000 Ljubljana, Slovenija
E-pošta: janez.lapajne@gmail.com, knap.martin@gmail.com

Povzetek. V nalogi je obravnavan postopek 3D rekonstrukcije objektov na podlagi večih zajetih slik. Za rekonstrukcijo je uporabljena FBP (angl. *Filtered back projection*) metoda. Temelji na rekonstrukciji 3D oblika iz slik, ki so zajete pri znanih kotih zasuka objekta. Velik poudarek je namenjen izgradnji in zasnovi sistema, saj je od njegove postavitve odvisna kakovost zajetih slik. Osredotočamo se predvsem na elemente, ki nam pomagajo pri čimboljši rekonstrukciji objekta v obliki oblaka točk. Prikazana je kalibracija sistema in preslikava slik v metrični koordinatni sistem. Predstavljena je uporabnost rekonstrukcije objektov in izvedba vzratnega inženirstva skeniranega objekta.

Ključne besede: rekonstrukcija 3d objekta, kalibracija kamere, poravnava oblakov točk, vzratno inženirstvo

1 UVOD

V robotskem vidu in računalniški grafiki je 3d rekonstrukcija proces zajemanja oblike in videza realnega objekta. Proces je lahko dosežen z aktivnimi ali pasivnimi metodami. V nalogi je uporabljena pasivna FBP metoda. Uporaba rekonstrukcije lahko določi profil 3d objekta oziroma koordinate katerikoli točke na tem profilu. Digitalna slika je v našem primeru vir informacij. Za dobro rekonstrukcijo je potrebno upoštevati vizualno dispariteto, osvetlitev, performance kamere in značilnosti scene. Uporabljeno kamero je potrebno ustrezno skalibrirati in merjene točke pretvoriti v metrični koordinatni sistem. Bolje kot je zastavljen zajem slik, ustrežnejši so končni rezultati. Dobro zajete slike olajšajo obdelavo. Rekonstrukcija objektov je uporabljena v različnih področjih raziskovanja. Med drugim v medicini, robotiki, računalniških igrar in v vzratnem inženirstvu.

2 TEORETIČNO OZADJE

2.1 Kalibracija

Pred začetkom izvedbe rekonstrukcije je potrebna geometrijska kalibracija sistema. Namen kalibracije je pridobitev parametrov toge preslikave, ki jih bomo potrebovali kasneje v algoritmu 3D rekonstrukcije s filtrirano povratno projekcijo. Cilj je projekcija iz slikovnega v metrični koordinatni sistem. Referenčne točke kalibracijskega objekta (3D) najprej togo premaknemo na mesto rotirajoče mize. Nato sledi DLT (angl. *Direct Linear Transformation*) kalibracija med 3D točkami referenčnega kalibra $T_{ref,i} = (X_i, Y_i, Z_i)$ ter 2D naborem točk kalibra iz slikovnega koordinatnega sistema

$t_{slik,i} = (x_i, y_i)$. Pri tem je število točk kalibracijskega objekta $i = 8$. Pri DLT kalibraciji z razcepom na singularne vrednosti rešujemo sledeči sistem:

$$a = \begin{bmatrix} X_i & Y_i & Z_i & 0 & 0 & 0 & -x_i X_i & -x_i Y_i & -x_i Z_i & 1 & 0 & -x_i \\ 0 & 0 & 0 & X_i & Y_i & Z_i & -y_i X_i & -y_i Y_i & -y_i Z_i & 0 & 1 & -y_i \end{bmatrix} \quad (1)$$

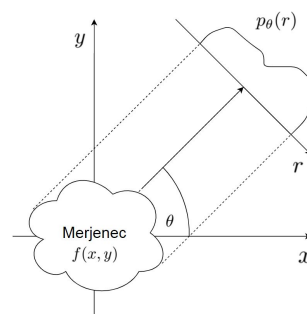
$$p = [r_{11} \ r_{12} \ r_{13} \ r_{21} \ r_{22} \ r_{23} \ r_{31} \ r_{32} \ r_{33} \ t_x \ t_y \ t_z] \quad (2)$$

$$ap^T = 0 \quad (3)$$

Matrika a ima dimenzije 16×12 . Rešitev sistema predstavlja 12 parametrov toge preslikave, ki predstavljajo rotacijo r_{ij} ter translacijo t_k .

2.2 Filtrirana povratna projekcija

Jedro naloge predstavlja postopek FBP (angl. *Filtered Backprojection*), ki je eden od temeljnih načinov rekonstrukcije pri slikanju z računalniško tomografijo na področju zdravstva. Idejo koncepta filtrirane povratne projekcije je najenostavneje predstaviti na 2D primeru, kjer 2D merjencu $f(x, y)$ za več kotov θ zajamemo 1D prečne projekcije $p_\theta(r)$.



Slika 1:: Filtrirana povratna projekcija.

Zajete projekcije so nato pretvorjene v frekvenčno domeno s pomočjo Fourier-ove transformacije $P_\theta(\omega)$ za vsak kot. Tu nato sledi filtriranje projekcij s frekvenčnim odzivom $|\omega|$ tako, da dobimo $g_\theta(r) = \int P_\theta(\omega) |\omega| e^{i2\pi\omega r} d\omega$. S filtriranjem se želimo znebiti visokih frekvenc v projekcijah in posledično pridobiti kvalitetnejšo rekonstrukcijo. Po filtriranju nato sledi še rekonstrukcija v $f'(x, y)$ in sicer tako, da filtrirane inverze Fourier-ove transformacije seštejemo.

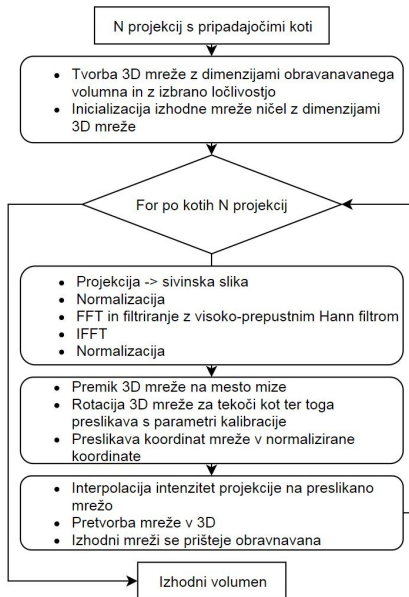
$$f'(x, y) = \int g_\theta(r) (x \cos \theta + y \sin \theta) d\theta \quad (4)$$

Ali v diskretni obliki, kjer je $\Delta\theta$ korak diskretizacije:

$$f'(x, y) = \frac{1}{2\pi} \sum_{i=0}^{N-1} \Delta\theta_i g_{\theta_i} (x \cos \theta_i + y \sin \theta_i) \quad (5)$$

Za potrebe naloge je postopek razširjen v tretjo dimenzijo, implementirani algoritem pa je predstavljen v spodnjem podglavju.

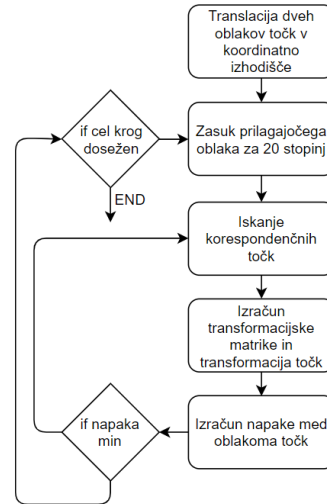
2.2.1 FBP algoritem: Na spodnji sliki je prikazan diagram poteka uporabljenega FBP algoritma.



Slika 2:: Diagram poteka FBP algoritma.

2.3 Poravnava oblakov točk

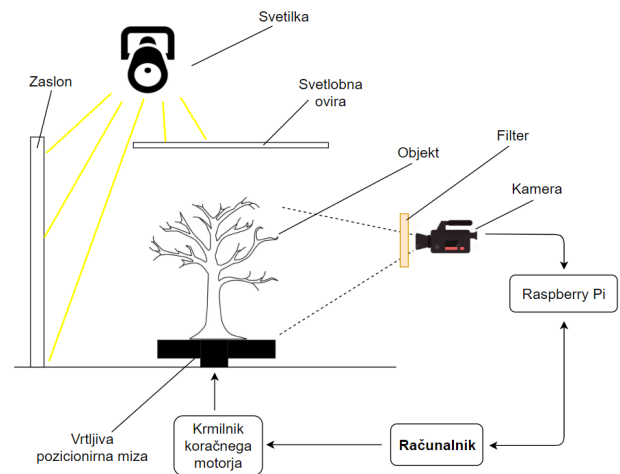
Potek algoritma poravnave točk je prikazan na sliki 3. Poravnavo izvajamo med dvema oblakoma točk tako, da enega prilagajamo na drugega. Na začetku oba oblaka točk postavimo v koordinatno izhodišče. Nato izvajamo optimizacijo prileganja na različnih rotacijah prilagajajočega oblaka. Slednje je izvedeno tako, da prilagajajoč oblak rotiramo po 20 stopinj in v vsaki legi izvajamo optimizacijo. To ponavljamo za 360 stopinj. Na koncu med vsemi izberemo poravnano lego, ki nam zagotavlja najmanjšo povprečno napako.



Slika 3:: Potek algoritma poravnave.

3 ZAJEM SLIK

Sistem s katerim smo izvajali rekonstrukcijo je blokovno prikazan na sliki 4. Objekt, ki ga želimo skonstruirati postavimo na vrtljivo pozicionirno mizo. Iz zgornje strani navzdol svetimo s homogeno, difuzno osvetlitvijo. Objekta direktno ne osvetljujemo, saj je namen osvetlitve predvsem osvetlitev ozadja z enakomerno intenziteto. Cilj je doseči čimvečjo razliko v intenziteti med opazovanim objektom in zaslonom. Za ta namen je med objekt in svetilko postavljena svetlobna ovira. Slike smo zajemali z Raspberry Pi kamero v osi na objekt in na osvetljeno ozadje. Pred kamero je dodan filter za namen filtriranja okoliške svetlobe.



Slika 4:: Semioperacijska shema sistema.

Celoten proces zajemanja krmilimo z računalnikom. Ta je žično povezan na krmilnik s katerim upravljamo rotirajočo pozicionirno mizo. Prav tako je računalnik brezžično povezan na Raspberry Pi. Ta zajete slike pošilja na računalnik. Objekt rekonstruiramo tako, da ga

postavimo na predvideno mesto. Nato lahko pričnemo z zajemom slik. Najprej inicializiramo kamero na začetne vrednosti. Nastavimo ISO na 200, svetlost na 45 in resolucijo na 900x1000. Objekt obračamo po začrtanih zasukih. Po vsakem izvedenem zasuku posnamemo eno sliko, ki se nato brezžično prenese iz Raspberry Pi-ja na računalnik. Postopek ponavljamo dokler objekta ne poslikamo v rangu celotnega kroga. Zajete slike ob znanih kotih zasuka se kasneje uporabijo za 3d rekonstrukcijo. V tabeli 1 so zapisane vse glavne komponente, ki so zajete v postavljenem sistemu. Manjše komponente kot so vijaki in nosilci v tabeli niso zajeti.

Tabela 1:: Komponente sistema.

Komponenta	Podrobnosti
Raspberry Pi	Raspberry Pi 3 Model B
Svetilka	Halogenska svetilka (12V)
Zaslon	Bela ravna plošča
Krmilnik	Fischertechnik stage control
Kamera	RP Cam V2-8 MP,1080p
Svetlobni filter	Ozkopasovni filter
Svetlobna ovira	Kartonasta zavesa
Pozicionirna miza	Fischertechnik rotary stage

Primer zajete slike obravnavanega objekta z opisanim sistemom je prikazan na sliki 5. S slike lahko vidimo dober kontrast objekta in precejšnjo razliko v barvi in svetlosti objekta proti ozadju.

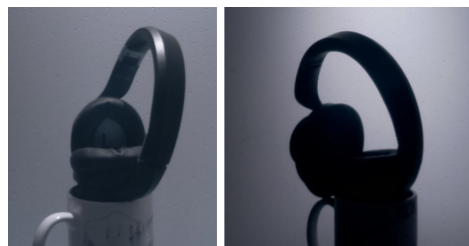


Slika 5:: Zajeta slika.

4 REZULTATI

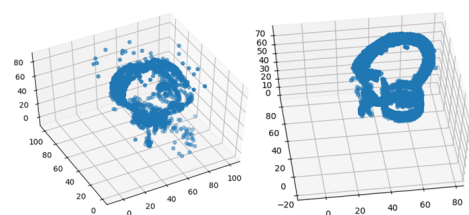
4.1 Vpliv kontrasta

Testirali smo kako svetlobna pregrada vpliva na rekonstrukcijo oblaka točk. Za ta namen smo vzeli enostaven objekt s stališča rekonstrukcije. Ta je prikazan na sliki 6. Na levi strani je prikazana slika objekta kjer svetlobne ovire nismo uporabili. Razvidno je, da se od objekta svetloba reflektivno odbiva v kamero. Kamera to zazna kot svetlo intenziteto. Deli objekta, ki so tako osvetljeni se po svetlosti veliko ne razlikujejo od ozadja, na nekaterih delih pa so celo zaznani, kot svetlejši. Na desni strani je prikazana slika objekta, kjer svetlobno oviro uporabimo. Objekt je v tem primeru bistveno temnejši od ozadja.



Slika 6:: Objekt: a) brez pregrade, b) s pregrado.

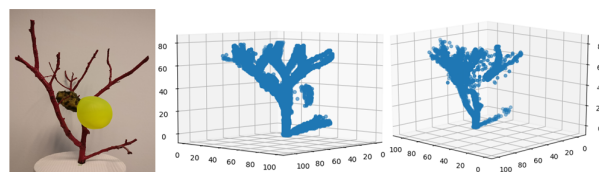
Na sliki 7 sta prikazana rezultata rekonstrukcij brez in z uporabo svetlobne ovire. Opazimo lahko, da je v primeru uporabljenе pregrade objekt bistveno lepše rekonstruiran. V primeru, ko zavese nismo uporabili nam odboji povzročajo napačno reprezentacijo objekta.



Slika 7:: Rekonstrukcija: a) brez pregrade, b) s pregrado.

4.2 Število zajetih slik

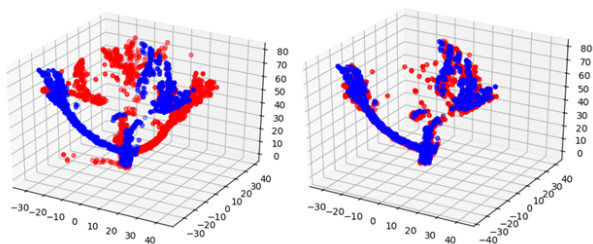
Dvakrat smo rekonstruirali isti predmet. Enkrat na podlagi 30 slik in drugič na podlagi 90 slik. Opazovali smo kako število zajetih slik vpliva na rekonstrukcijo. Na sliki 8 je čisto na levi strani prikazan skeniran objekt, na sredini objekt rekonstruiran na podlagi 90 slik in na desni objekt rekonstruiran na podlagi 30 slik. Vidimo, da je dobimo boljše obliko na podlagi večih slik. Za prikaz celotnega objekta v obeh primerih bi bilo potrebno prilagoditi meje upravljanja.



Slika 8:: a) objekt, b) zajetih 90 slik, c) zajetih 30 slik.

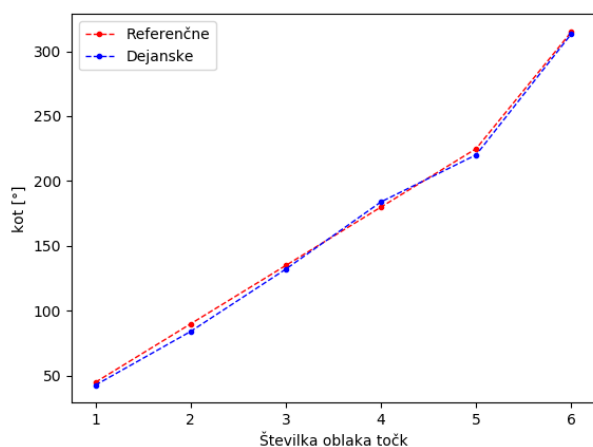
4.3 Poravnava oblakov točk

Za dodaten eksperiment smo izvedli medsebojno poravnavo oblakov točk. Referenčni oblak točk smo zajeli z 90 slikami in si zapolnili referenčno lego objekta na pozicionirni mizi. Nato smo izvedli še šest zajemov po 30 slik. Ob vsakem zajemu smo skeniran objekt zasukali za predpisan kot glede na referenčno lego okoli vertikalne osi. Tako smo zajeli oblake točk zasukane za 45, 90, 135, 18, 225 in 315 stopinj. Na sliki 9 sta zaradi preglednosti prikazana le referenčni oblak in oblak zasukan za 90 stopinj. Na levi strani sta oblaka prikazana pred poravnavo in na desni po poravnavi.



Slika 9:: a) neporavnana, b) poravnana.

Za vse, pod različnimi koti zajete oblake točk, smo izvedli poravnavo. Nato smo iz pridobljene transformacijske matrike izluščili rotacijo okoli vertikalne osi. Ker smo poznali kote zasuka objekta glede na referenčno lego, smo jih primerjali s temi pridobljenimi iz transformacijske matrike. Na sliki 10 so z rdečo barvo prikazani ročno določeni zasuki, z modro pa zasuki pridobljeni iz transformacijske matrike. Vidimo, da dobimo z izračunanimi koti na podlagi transformacijske matrike dokaj zanesljive rezultate. Prileganje testiranih zasukov glede na referenčne je v rangi petih stopinj. Napaka se verjetno pojavi zaradi napak v pozicionirnem sistemu, nenatančne meritve kota glede na referenčno lego in napake v numeričnem izračunu.



Slika 10:: Razlika med referenčnimi in dejanskimi koti.

4.4 Vzvratno inženirstvo

Vzvratno inženirstvo je proces, kjer na podlagi fizičnega objekta rekonstruiramo njegov model. Model dobimo tako, da oblak točk prilagodimo in pretvorimo v ustrezen format. S pomočjo programa Solidworks smo tako oblak točk pretvorili v stl format. Kot takega ga lahko uvozimo v program za 3d tisk in rekonstruiramo še v fizični obliki.

5 ZAKLJUČEK

Z našo tehniko rekonstruiranja nismo popisali dovolj dobrih delailov, čas rekonstrukcije se povečuje z velikostjo sepljanja, ugotovili smo, da je sistem zelo pomemben

+ osvetlitev. Pomembno tud število slik. Poravnava slik, nam je dobro uspela - lahko če ne bi poznali pod kakšnim kotom je poslikal in dobil koliko je zamaknjen glede na referenčno lego.

LITERATURA

- [1] Ž. Špiclin, B. Likar, M. Burmen. *Predavanje pri predmetu - Biomedicinske slikovne tehnologije: Rekonstrukcija slik*, Univerza v Ljubljani, 2018.
- [2] Ž. Špiclin. *Predavanje pri predmetu - Robotski vid: Prileganje modelov na slike*, Univerza v Ljubljani, 2019.
- [3] Moons, Theo, Luc Van Gool. *3D reconstruction from multiple images part 1: Principles*, Foundations and Trends in Computer Graphics and Vision 4.4: 287-404, 2010.
- [4] Vosselman, George, and Sander Dijkman. *3D building model reconstruction from point clouds and ground plans*, International archives of photogrammetry remote sensing and spatial information sciences 34.3/W4: 37-44, 2001.
- [5] Woodham, Robert J. *Photometric method for determining surface orientation from multiple images*, Optical Engineering. 19 (1): 138-141, 1980.
- [6] *Raspberry Pi 3, Model B V1.2 Technical Specifications*, RS Components, 20.9.2017.
- [7] *Fischertechnik Technical Specifications*, fischertechnik GmbH, 2016.
- [8] , ,
- [9] , ,