Subjektivno ispitivanje kvalitete oblaka točaka i usporedba s objektivnim mjerama

Lukman, Patrik

Master's thesis / Diplomski rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University North / Sveučilište Sjever**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:122:947335

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-28**

Repository / Repozitorij:

University North Digital Repository

Sveučilište Sjever

Zadatak diplomskog rada br. 004/MMD/2020

Subjektivno ispitivanje kvalitete oblaka točaka i usporedba s objektivnim mjerama

Patrik Lukman, 0894/336D

Varaždin, rujan 2020. godine



Sveučilište
Sjever

Odjel za Multimediju, oblikovanje i primjenu

Zadatak diplomskog rada br. 004/MMD/2020

Subjektivno ispitivanje kvalitete oblaka točaka i usporedba s objektivnim mjerama

Student

Patrik Lukman, 0894/336D

Mentor

izv. prof. dr. sc. Emil Dumić

Varaždin, rujan 2020. godine

Sveučilište Sjever
Sveučilišni centar Varaždin
104. brigade 3, HR–42000 Varaždin

UNIVERSITY NORTH

Prijava diplomskog rada

Definiranje teme diplomskog rada i povjerenstva

ODJEL: Odjel za multimediju

STUDIJ: diplomski sveučilišni studij Multimedija

PRISTUPNIK: Patrik Lukman | MATIČNI BROJ: 0894/336D

DATUM: 16.06.2020. | KOLEGIJ: Računalni vid

NASLOV RADA: Subjektivno ispitivanje kvalitete oblaka točaka i usporedba s objektivnim mjerama

NASLOV RADA NA ENGL. JEZIKU: Subjective quality assessment of point clouds and comparison with objective measures

MENTOR: Emil Dumić | ZVANJE: doc.dr.sc.

ČLANOVI POVJERENSTVA:

1. doc.art.dr.sc. Mario Periša - predsjednik
2. doc.art. Robert Geček - član
3. doc.dr.sc. Emil Dumić - mentor
4. doc.dr.sc. Andrija Bernik - zamjenski član
5.

Zadatak diplomskog rada

BROJ: 004/MMD/2020

OPIS:

U ovom radu će biti provedeno subjektivno ispitivanje statičkih oblaka točaka prema ITU-R P.910 standardu, uz neke specifične metode, vezane za same oblake točaka, te usporedbu dobivenih subjektivnih rezultata s postojećim objektivnim mjerama kvalitete.
Opisat će se općeniti način za subjektivno ispitivanje multimedijskog sadržaja, s naglaskom na videozapis. Potom će se ukratko opisati JPEG Pleno aktivnosti (svjetlosna polja, holografija i oblaci točaka). Nakon toga će se dati pregled dosadašnjih načina za prikaz, vizualizaciju i kompresiju oblaka točaka. Općeniti dosadašnji načini za prikaz su bili direktnim prikazom oblaka točaka, ili nakon rekonstrukcije površine nekom metodom. Metode za vizualizaciju su bile 2D, 3D ili pomoću virtualne stvarnosti. Potom će biti opisani neki od postojećih načina kompresije, poput octree bazirane metode (npr. unutar PCL), kompresije pomoću projekcija (npr. unutar 3DTK), te novijih MPEG kompresija (G-PCC i V-PCC). Nakon toga će biti dan kraći prikaz postojećih objektivnih mjera kvalitete oblaka točaka, koje se općenito dijele na "point-to-point", "point-to-plane" te na objektivne mjere kvalitete projekcija oblaka točaka.
Praktični dio rada bit će prikazan kroz subjektivna ispitivanja kvalitete i korelaciju s nekim od objektivnih mjera kvalitete statičkih oblaka točaka. Oblaci točaka će biti komprimirani G-PCC i V-PCC alatima za kompresiju. Subjektivno ispitivanje će se provesti prema preporuci ITU-R P.910. Objektivne mjere će biti izračunate pomoću npr. CloudCompare alata. Provedeno istraživanje je vezano za JPEG Pleno Point Clouds grupu, povezanu za standardizaciju statičkih oblaka točaka.

ZADATAK URUČEN: 18.6.2020. | POTPIS MENTORA: Emil Dumić

SVEUČILIŠTE SJEVER

Predgovor

Tema ovog završnog rada rezultat je osobne zainteresiranosti za metode skeniranja i konvertiranja stvarnih objekata u digitalni format. Namjera je bila sticanje novih znanja o programima te načinima dobivanja, spremanja i manipuliranja oblaka točaka uz pomoć različitih metoda kompresije kako bi postigli što realističniji izgled nakon provedbe postupka kompresije.

Inspiracija za ovaj diplomski rad bila su izuzetno zanimljiva i poučna predavanja profesora Dumića, kao i nedovoljno iskorišteni potencijal oblaka točaka te njihova svestrana primjena u danima koji dolaze.

Ovim putem zahvaljujem mentoru izv. prof. Emilu Dumić na pruženoj pomoći, uvodu u svijet oblaka točaka te nesebičnim savjetima u procesu realizacije ovoga rada. Također šaljem zahvale svim članovima moje obitelji i prijateljima na svakodnevnom strpljenju, razumijevanju, savjetovanju, poticanju i neizmjernoj podršci tijekom svih godina mojeg studija.

Sažetak

Mogućnost obuhvaćanja objekata kao i osoba iz stvarnog svijeta u oblake točaka doživjela je izuzetno veliki napredak u zadnjem desetljeću. Jedan od najpoznatijih i najkorištenijih oblika zabilježavanja objekata i osoba je uz pomoć oblaka točaka. Oblaci točaka su skupovi točaka u trodimenzionalnom virtualnom prostoru dobiveni kao rezultat različitih vrsta skeniranja, kao na primjer: lasersko skeniranje, skeniranje pomoću fotoaparata, skeniranje pomoću kamera i drugo. Oni nam pružaju realističan i kreativan prikaz ne samo skeniranih objekata, nego i njihovih atributa, poput udaljenosti, dubine i boje. Međutim, takvi modeli zahtjevaju mnogo virtualnog prostora pa su stoga efektivni načini kompresije postali glavni prioritet. Razvoj načina kompresije oblaka točaka rezultirao je velikim brojem efektivnih i jednostavnih metoda kompresije. koje omogućuju njihovo lagano spremanje i manipulaciju. Usporedbe prikazane u ovom radu dobivene su uz pomoć programa dobivenog od mentora Emila Dumić, kao i programa MatLAB, Meshlab i CloudCompare.

***Ključne riječi**: 3D, oblaci točaka, kompresija, MatLAB*

Abstract

The capturing of real-life object and people has seen massive improvements in the last decade. One of the most noticeable and used ways of capturing such objects is by the means of point clouds. Point clouds represent a cluster of points in a virtual, three-dimensional environment as a result of multiple scans such as: laser scanning, camera scanning, photogrammetry and others. They offer a realistic and creative display of objects as well as their attributes, like depth, distance and colour. However, point clouds can easily take up copious amounts of digital storage, thus effective methods of compression have become extremely important. The development of point cloud compression has resulted in effective and simple-to-use methods which allow the manipulation and storage of such models. The comparisons shown in this thesis were obtained with the aid of a program shared by professor Emil Dumić as well as the use of MatLAB, Meshlab and CloudCompare.

***Keywords:** 3D, point clouds, compression, MatLAB*

Sadržaj

1. Uvod ... 1
2. Oblaci točaka ... 2
2.1. Virtualna vidljivost i važnost oblaka točaka u imerzivnim aplikacijama ... 2
2.2. Priprema za procesiranje ... 3
3. JPEG Pleno ... 4
4. Kompresija oblaka točaka ... 7
4.1. MPEG G-PCC ... 9
4.1.1. Octree i TriSoup kodiranje ... 9
4.1.2. Kodiranje atributa ... 10
4.1.2.1. RAHT ... 10
4.1.2.2. Predviđajuća transformacija i Lifting transformacija ... 11
4.2. MPEG V-PCC ... 14
4.2.1. 3D zakrpe i njihovo generiranje ... 14
4.2.2. Pakiranje zakrpa ... 16
4.2.3. Geometrijske mape i mape zauzetosti ... 16
4.2.4. Postavljanje slike, re-apliciranje boje i video kompresija ... 17
4.2.5. Toniranje i redefiniranje geometrije i atributa ... 17
4.3. Kompresija pomoću panorama ... 18
4.3.1. Korištenje uobičajene kompresije slike ... 18
4.3.2. Kodiranje udaljenosti i kompresija oblaka točaka ... 19
4.4. Kompresija dinamičkih oblaka točaka pomoću projekcija ... 20
5. Objektivne mjere za ocjenjivanje kvalitete oblaka točaka ... 21
6. Subjektivne metode evaluacije oblaka točaka ... 23
7. Evaluacija ... 25
7.1. Detalji o istraživanju ... 25
8. Rezultati ... 34
8.1. Usporedba istraživanja iz drugih laboratorija ... 36
9. Zaključak ... 51
10. Literatura ... 53
11. Popis slika ... 56

1. Uvod

Konstantan napredak tehnologije zaslužne za učitavanje i obuhvaćanje trodimenzionalnih objekata u digitalan format, omogućio je inovaciju novijih područja istraživanja kao što su virtualna, proširena i miješana stvarnost (*engl.* „virtual, augmented and mixed reality“).

Načini prikupljanja podataka postaju sve jednostavniji, a količine obrađenih podataka sve veće. Laserko snimanje objekata, kao jeftina, ali kvalitetna metoda, omogućava pristup ne samo prostornim karakteristikama, nego i fizičkom izgledu i atributima obrađenog objekta i to u obliku takozvanih „***oblaka točaka***“. Privreme informacije o objektu dodane su obliku zasebnih instanca u kojem su uhvaćene ili poziciji objekta u određenom vremenu, slično slikama (*engl.* „frames“) u 2D videu.

Zbog brzog napretka u kreiranju i korištenju slika u različitim područjima multimedije i aplikacijama, grupa pod imenom Joint Photographic Experts Group (JPEG) izdaje standarde koji olakšavaju savladanje novijih izazova pri konstantno napredovanim aplikacijama.

2. Oblaci točaka

Oblaci točaka su skupovi točaka u trodimenzionalnom prostoru čija je svrha kreiranje fotorealističnog 3D objekta. Svaka točka u tom oblaku reprezentira točku na površini objekta te je opisana x, y i z koordinatama. Takvi trodimenzionalni objekti kreirani su pomoću 3D skenera, te ovisno o tipu 3D skenera, oblaci točaka mogu sadržavati različite informacije. To su na primjer: boja, intenzitet, transparentnost i normale, koji zajedno pridodaju sveukupne informacije o snimljenom objektu.

Korištenje metode 3D skeniranja i kreiranja trodimenzionalnih objekata koji se sastoje od oblaka točaka, zaslužilo je popularnost i upotrebu u zadnjih nekoliko godina. U usporedbi sa trodimenzionalnim modelima napravljenima ručno, oblaci točaka pružaju nam jednostavniji, gušći i puno realističniji prikaz.

Oblaci točaka mogu biti ***statički*** i ***dinamički***. Statički oblaci prikazuju samo jedan oblak točaka, dok dinamički oblaci prikazuju sekvencu oblaka točaka.

2.1. Virtualna vidljivost i važnost oblaka točaka u imerzivnim aplikacijama

U virtualnom okruženju kreiranom u programu Unity, HTC Vive headset korišten je kao centar vidljivosti u eksperimentu čiji je cilj bio prikupiti podatke o glavnom fokusu viljdivosti. Rezolucija virtualnog svijeta bila je 2880x1600 piksela sa kutem vidljivosti 110° i brojem prikazanih oblaka točaka u sekundi (*engl.* „frame rate per second“, FPS) od 90 Hz. Kako bi se sakupljale informacije, korišten je Pupil Labs hardware kao sustav za praćenje očiju (*engl.* „eye tracker“). Sudionicima je prikazano sveukupno 12 modela (6 ljudskih modela i 6 objekata) nakon čega su sve te modele trebali poredati prema vlastitom mišljenju.

Kao rezultat ovog eksperimenta, dokazano je kako su najčešće lokacije sa malim brojem detalja i lokacije sa velikim brojem detalja uvijek u centru pozornosti, nakon čega slijede bilo kakve lokacije sa tekstom ili nepredvidivim objektima/znakovima [**1**]. Sudionici su također često promatrali veće i kompleksnije objekte dulje od ostalih. Svi sudionici su već bili upoznati sa VR okruženjem te su pozitivno ocjenili svoje iskustvo kao i kvalitetu samih modela.

Nakon prikupljanja podataka, dobivena je precizna mapa usmjerenosti pogleda (*engl.* „fixation density map“) pomoću koje je predloženo daljnje istraživanje i kreiranje sheme glavnih područja interesa za oblake točaka [**2**].

2.2. Priprema za procesiranje

Oblaci točaka mogu, ali ne moraju biti spremljeni u njihovim trodimenzionalnim volumetrijskim reprezentacijama te je zbog toga uveden opcionalni korak u pripremi za procesiranje koji konvertira neobrađeni (*engl.* „raw“) format oblaka u trodimenzionalni prikaz uz pomoć (i, j ,k) baziranog kartezijevog koordinatnog sustava. Ovaj proces nazvan je vokselizacija [**3**].

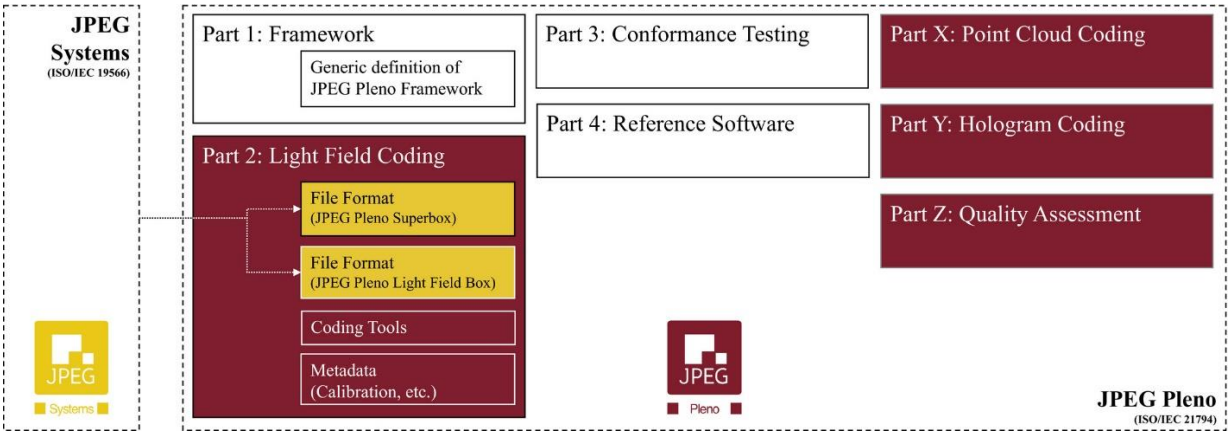
Promjena rezolucije slike u kompresiji slike ili videozapisa često je korištena metoda i njome se postiže očuvanje kvalitete same slike/videozapisa uz najčešće manju brzinu prijenosa (*engl.* „bit rate“). Metoda smanjenja broja točaka mogla bi se koristiti za smanjenje potrebnog broja bitova za pohranu. Skaliranje bi također pomoglo pri smanjenju udaljenosti između točaka kao i povećanju gustoće samog oblaka, što je korisno kod oblaka točaka sa cjelobrojnim koordinatama. Međutim, za decimalni zapis koordinata ova metoda nema prevelikog utjecaja, jer se oblak točaka možete promatrati s manje ili veće udaljenosti.

3. JPEG Pleno

U listopadu 2014. godine, grupa JPEG kreirala je temelje za takozvanu JPEG inicijativu kao odaziv za izuzetan uspjeh u napretku tehnologija za snimanje, obuhvaćanje, reprezentaciju i prikaz objekata i slika. Spomenuti napredak zahtjevao je nove standarde kako bi se isti uspješno adaptirao i prihvatio. Kako bi se postignuo cilj kreiranja novih standarda, JPEG Pleno zahtijeva specifikaciju sustavnih alata, alata za kodiranje kao i prikladne metapodatke.

Ime „**Pleno**“ referira na takozvanu „plenoptičku“ matematičku reprezentaciju modela koji uzima u obzir osvjetljenje i informaciju o boji za svaku točku na sceni te dodaje informaciju o smjeru, ovisno na kojim bi dijelovima ta kombinacija osvjetljenja i boje bila veća/manja kada se objekt promatra iz druge perspektive. Ova reprezentacijska struktura nagovijestila je prelazak sa dvodimenzionalne metode zabilježavanja informacija na trodimenzionalnu [**4**].

Tri najveća i najkorištenija plenoptička modaliteta su: svjetlosna polja, oblaci točaka i holografija (slika 3.1). Pomoću tih modaliteta, JPEG namjerava pružati standardne temelje za korištenje skeniranja i reprezentiranja objekata u digitalnom formatu u različitim industrijama.



Slika 3.1: Struktura JPEG Pleno-a, izvor [4]

Kako bi uspješno postignuo zadane ciljeve, JPEG Pleno podijeljen je na nekoliko dijelova, svaki sa nekom zasebnom svrhom:

- 1. dio: okvir/temelji (*engl*. „Framework“)
- 2. dio: kodiranje svjetlosnih polja (*engl.* „Light Field Coding“)
- 3. dio: ispitivanje sukladnosti (*engl.* „Conformance Testing“)
- 4. dio: referentni programi (*engl.* „Reference Software“)

1. dio: okvir/temelji (*engl.* „Framework“) definiraju svrhu JPEG Pleno-a te pomaže adaptirati na mnoge popularne i korištene standarde kao što su ISO/IEC JTC 1/SC 29 i ISO/IEC DIS 21794-1 WD2.0.

2. dio: kodiranje svjetlosnih polja (*engl.* „Light Field Coding“) najnapredniji je dio JPEG Pleno-a i on specificira tehnologije korištene u kodiranju spomenutih svjetlosnih polja. U ovaj dio također spada takozvani JPEG Pleno Superbox, koji sadrži bilo koju vrstu JPEG Pleno podataka i metapodataka.

3. dio: ispitivanje sukladnosti (*engl.* „Conformance Testing“) specificira metode i procese provjeravanja sukladnosti novounešenih dodataka u ostalim djelovima JPEG Pleno-a.

4. dio: referentni programi (*engl.* „Reference Software“) pruža implementaciju specifičnih programa sa ciljem stvaranja funkcionalnog rješenja.

Djelovi X, Y i Z služe kao dodatna podrška glavnim djelovima JPEG Pleno-a. Kodiranje oblaka točaka (*engl.* „Point Cloud Coding“) specificira programe, vrste datoteka i metadatoteka potrebne za kreiranje oblaka točaka, dok kodiranje hologrma (*engl.* „Hologram Coding“) specificira holografske tehnologije. Procjena kvalitete (*engl.* „Quality Assessment“) procjenjuje kvalitetu protokola koji su potrebni slikovnim modalitetima.

Uz standarde za kompresiju slike, JPEG Pleno također namjerava pružiti standarde za kompresiju holografije. Međutim, spremanje digitalne holografije još je uvijek slabo definirano, što stvara probleme pri samom pokušaju kompresije. Za spremanje i reprezentaciju svjetlosnih valova (nazvanih objektnim valovima), zasad se koriste samo dvije vrste kodiranih formata: realni zapis (*engl.* „real-value“) i kompleksni zapis (*engl.* „complex-value“).

Pošto trenutne vrste prikaza hologrma bazirane na prostornim svjetlosnim modulatorima mogu operirati funkcije samo sa stvarnim i pozitivnim vrijednostima, holografi se često kodiraju koristeći format realnog zapisa. Dvije vrste hologrma mogu se dobiti koristeći real-value format: amplitudni holografi i fazni holografi. Amplitudni holografi dobiveni su optički pomoću CCD (charged-coupled device) spojenog na računalo ili matematičkom simulacijom. Fazni holografi bazirani su na fazi referentnog vala, a pošto današnji uređaji ne mogu snimati faze svjetlosnih valova, oni se moraju kreirati pomoću kompleksnih i dugotrajnih metoda poput **Gerhberg-Saxtonovog algoritma** [**4**].

Holografi s kompleksnim zapisom mogu biti dobiveni matematičkom simulacijom širenja svjetla usmjerenog po sceni ili vizualno pomoću fazno-mijenjajuće (*engl.* „phase-

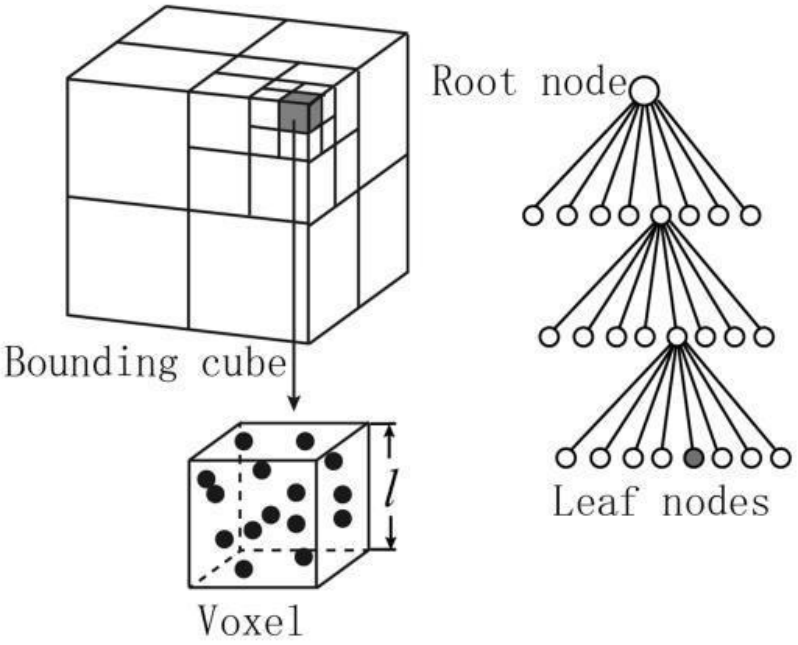
shifting“) holografije. Oni mogu biti kodirani koristeći amplitudno-faznu metodu ili realno-imaginarnu metodu.

Neki od ranijih testiranja kompresije hologramа, koji su izvršeni prateći standarde kodiranja (JPEG i JPEG 2000), koristili su video standarde H.264/AVC i H.265/HEVC na računalno generiranim hologramima virtualnih objekata. Rezultati su pokazali kako je video standard H.265/HEVC imao najbolji učinak gledajući izobličenja uz jednaku brzinu prijenosa.

4. Kompresija oblaka točaka

Kompresija trodimenzionalnih oblaka točaka fokusira se na pronalaženju i iskorištavanju višaka kreiranih tijekom snimanja (prostorni/geometrijski višak kod statičkih, a prostorni i vremenski višak kod dinamičkih oblaka točaka).

Spremanje podataka dobivenih pomoću oblaka točaka dosta je težak i zahtjevan proces, stoga se koristi struktura podataka pod imenom ***„octree“*** koja je naprednija verzija „quadtree“ tehnologije korištene u 2D. U ovakvoj strukturi, cijeli oblak točaka podijeljen je na 8 kocaka (slika 4.1), a taj proces je ponovljen toliko mnogo puta dok još postoje blokovi koji nisu prazni (blokovi sadrže oblake točaka). Individualne točke prvo su skenirane za dubinu, pomoću čega se identificira razina detalja na različitim razinama „octree“-a, te se stoga cijeli oblak točaka može kodirati unutar spomenutih blokova. Ovaj proces opisan je od strane znanstvenika ***Ruwena Schnabela*** [5].



Slika 4.1: Prikaz podjele octree-a, izvor: google slike

Još jedan pristup ovakvoj kompresiji osmislio je znanstvenik ***Yan Huang*** prema kojem se „octree“ isprva konstruira, a zatim se smanjuje kako bi zauzeo prostor potreban da bi se model komprimirao. Kodiranje boje dobiva se dodavanjem PCA i Lloyd-ovog algoritma kako bi se adaptivno zbrojio atribut boje.

Kompresija oblaka točaka može se podjeliti na ***intra frame i inter frame kodere***. Intra frame koder provjerava samo jedan izolirani oblak točaka i komprimira ga iskorištavanjem prostornog/geometrijskog viška. Dok inter frame predviđa trenutni oblak točaka baziran na prošlom oblaku točaka iskorištavanjem vremenskih viškova. Sličan postupak koristi se i pri videokodiranju [**6**].

Uz komprimiranje geometrije, atributi zadanih oblaka podataka također moraju biti komprimirani te su stoga predložene drugačije vrste kodiranja: kodiranje s gubicima (*engl.* „lossy“) i kodiranje bez gubitaka (*engl.* „lossless“).

Oblaci točaka veoma su korišteni u današnje vrijeme te nam pružaju izuzetno velik broj informacija o snimljenom objektu pa su stoga smišljene specializirane „knjižnice“ pod imenom ***Point Cloud Libraries (PCL)***. One sadrže mnoge algoritme uključujući algoritme za filtriranje, za predviđanje potrebne veličine oblaka točaka, za rekonstrukciju površine, za registraciju te za segmentiranje [**7**].

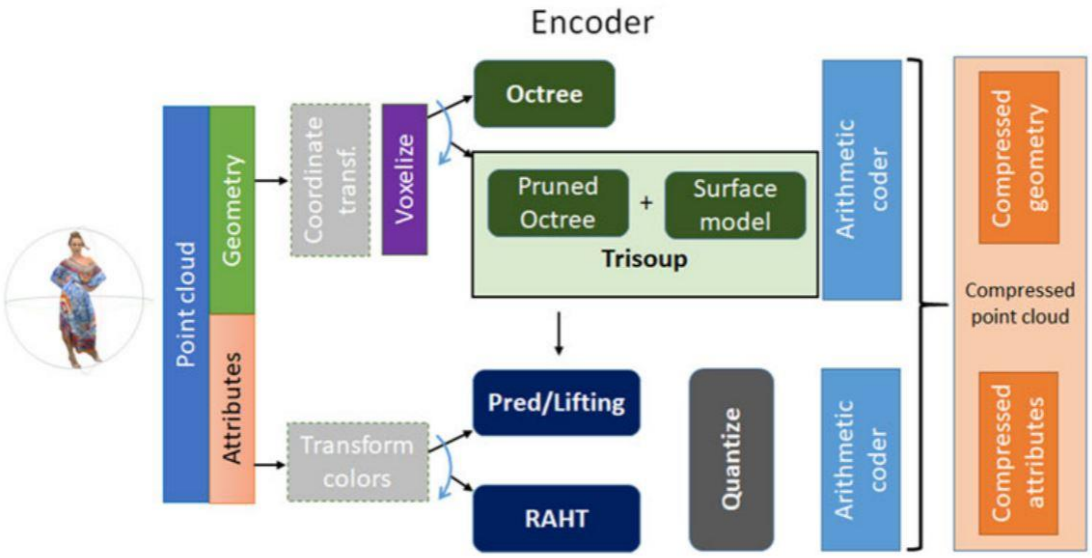
PCL je besplatan open source program koji je uspješno implementiran na Linux, MacOS, Windows i Android/iOS platforme. Sa ciljem olakšavanja rada, PCL (slika 4.2) se dijeli na skup manjih „knjižnica“ koje zasebno mogu raditi na drugim projektima istovremeno.



Slika 4.2: PCL logo, izvor: google slike

4.1. MPEG G-PCC

MPEG G-PCC sastoji se od dva modula. Već spomenuti „Octree“, koji je bazirani na dubini te „**Triangle Soup**“ (TriSoup), koji je baziran na površinskoj rekonstrukciji koristeći primitive kreirane od tri verteksa nakon što je model prošao kroz octree kompresiju. Octree je zaslužan za kompresiju i definiranje dubine koristeći parametar dubine, dok je TriSoup zaslužan za definiranje veličine bloka u kojem je model kodiran. TriSoup je predviđen za kompresiju modela veće gustoće (slika 4.3).



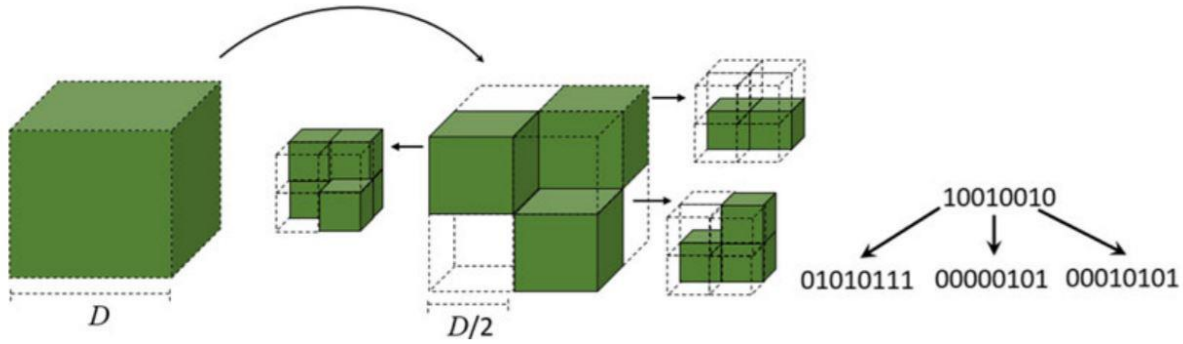
Slika 4.3: G-PCC dijagram za referentni koder, izvor: [9]

4.1.1. Octree i TriSoup kodiranje

Vokselizirani oblak točaka predstavljen je koristeći octree strukturu bez gubitaka informacija. Kao već spomenuto, svi zauzeti blokovi podjeljeni su na 8 manjih kocaka horizontalno i vertikalno, nakon čega su svi blokovi označeni sa 1 ili 0 ovisno o njihovoj zauzetosti (slika 4.5). Zauzeti blokovi tada su kodirani pomoću entropijskog kodera uzimajući u obzir korelaciju sa susjednim blokovima (slika 4.4).

Pošto kodiranje izoliranih blokova nije moguće pomoću entropijskog kodiranja uzimajući u obzir susjedne blokove, potrebno je korištenje DCM-a (Direct Coding Mode) kao alternative. Koordinate samih točaka u DCM-u kodirane su bez potrebe kompresije. DCM je izveden iz susjednih blokova kako bi se izbjegnulo njegovo korištenje za sve blokove [**8**].

Sama geometrija može biti reprezentirana pomoću octree-a, sastavljeno od „korijena“ do „lišća“ koji predstavljaju zakupljene blokove veće od voksela. Površina objekta pokrivena je trokutima bez povezanosti te se stoga i zove „Triangle Soup“ (TriSoup).



Slika 4.4: Prva dva koraka octree kompresije, izvor: [9]

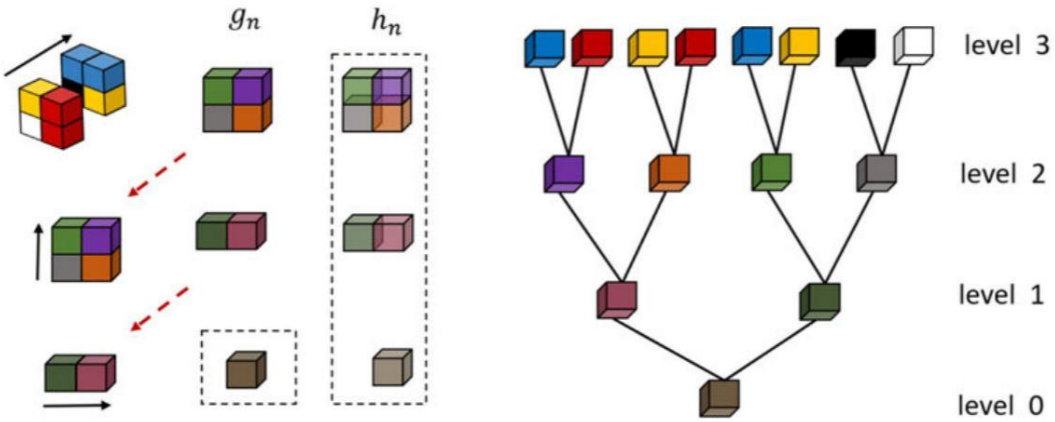
TriSoup kao alat za kodiranje nije obavezan, ali pomaže pri poboljšavanju subjektivne kvalitete kod manje brzine prijenosa. Kada koristimo TriSoup, geometrijski zapis bitova (*engl.* „bitstream“) postaje kombinacija octree-a, indikatora razmaka/segmentacije i informacije o poziciji verteksa.

4.1.2. Kodiranje atributa

Informacije o boji modela ponovno su dodane na model nakon što je model prošao kroz procese kodiranja i dekodiranja koristeći izlazne točke. Takve informacije zatim mogu biti kodirane pomoću tri različita pristupa: RAHT, Prediction transform i Lifting transformation (kraće: Lifting).

4.1.2.1. RAHT

Prvi je pristup baziran na 3D Haar transformaciji te koristi reprezentaciju octree-a kako bi iskoristio iznose atributa sa nižih razina octree-a za predviđanje iznosa u sljedećoj razini.



Slika 4.5: Transformacija bloka veličine 2x2x2 za octree kodiranje, izvor: [9]

Prvobitni način rada RAHT tranformacije je da kreće od „lišća“ (najviše razine) pa sve do „korijena“ (najniže razine) te je transformacija dodana na svaki blok i izvedena u tri koraka: po jedan korak za x, y i z smjer.

U svakom koraku generira se g_n sa niže razine i h_n sa više razine od strane signala v sa N elemenata. N-ti koeficijenti za g i h izračunati su koristeći jednadžbu 4.1:

$$\begin{bmatrix} g_n \\ h_n \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} v^2{}_n \\ v^2{}_{n+1} \end{bmatrix}. \tag{4.1}$$

RAHT transformacija može se ponovno izračunati koristeći trenutni **g** kao novi ulazni signal **v** te je za svako ponavljanje broj za niže razine **n** podijeljen sa 2. Međutim, jednadžba mora biti izmjenjena kada želimo koristiti Haar transformaciju kako bi u obzir uzeli same oblake podataka i njihovu veličinu.

$$\begin{bmatrix} g_n^l \\ h_l^l \end{bmatrix} = T \begin{bmatrix} g_{2n}^{l+1} \\ h_{2n+1}^{l+1} \end{bmatrix} \tag{4.2}$$

$$T = \frac{1}{\sqrt{w_1 + w_2}} \begin{bmatrix} \sqrt{w_1} & \sqrt{w_2} \\ -\sqrt{w_2} & \sqrt{w_1} \end{bmatrix} \tag{4.3}$$

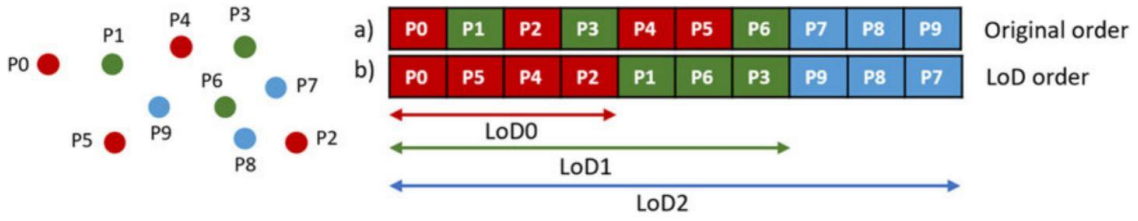
$$w_n^l = w_1 + w_2 \tag{4.4}$$

$$w_1 = w_{2n}^{l+1}\,,\; w_2 = w_{2n+1}^{l+1} \tag{4.5}$$

U ovim jednadžbama, l predstavlja dekompozicijsku razinu, w_1 i w_2 predstavljaju težinu asociranu sa g_{2n}^{l+1} i g_{2n+1}^{l+1} koeficijentima nižih razina $l+1$, dok w_n^l predstavlja težinu asociranu sa koeficijentima nižih razina l (formule 4.2-4.5). Uz pomoć tih jednadžba, RAHT adaptira i izjednačava raspodjelu većih težina po gušćim područjima.

4.1.2.2. Predviđajuća transformacija i Lifting transformacija

Predviđajuća transformacija (*engl.* „Predicting transformation“) bazirana je na predviđanju pomoću udaljenosti kako bi se kodirali atributi, a ona ovisi o **LoD** (***„Level of***



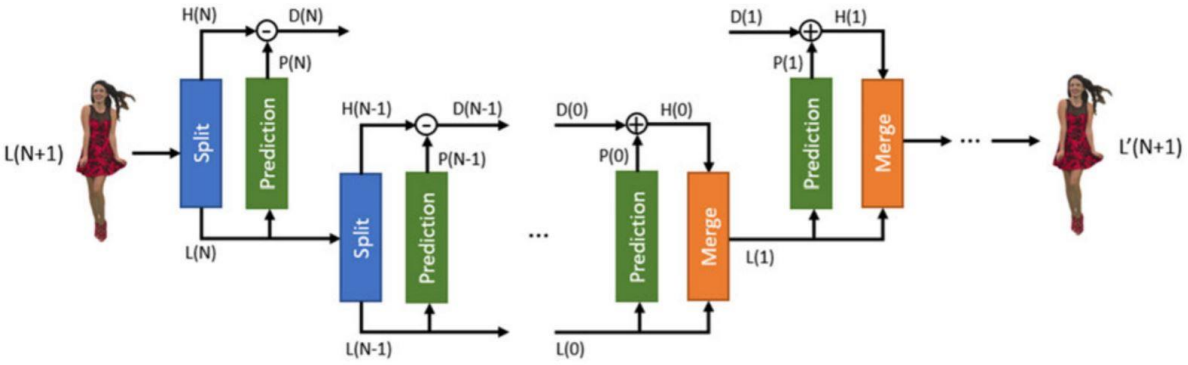
Slika 4.6: Proces generiranja razine detalja, izvor: [9]

Detail"). LoD je reprezentacija koja raspodjeljuje ulazne točke u više skupina zvanih „*Refinements Levels*" uz pomoć Euklidovog kriterija o udaljenosti (slika 4.6).

Atributi svake točke mogu biti kodirani koristeći predviđanje bazirano na LoD redoslijedu, kod kojeg bi na primjer mogli predvidjeti P_2 uz pomoć rekonstrukcija susjednih točaka P_4, P_5 i P_0. Kao alternativa, predviđanje P_2 točke također se može izvesti iz težinskog prosjeka susjednih točaka.

Limitiranje broja predvidivih točaka te susjednih točaka za predviđanje je također jedna od opcija kada koristimo Predicting transform, no kada je udaljenost između dvaju atributa prevelika, program će koristiti optimizacijsku proceduru kako bi odabrao najbolju točku za predviđanje. Vrijednosti atributa u jednoj razini („refinement level") uvijek su predviđene koristeći vrijednosti atributa najbližih susjednih točaka u prethodnoj razini.

Predviđajuća transformacija koristi dva operatora baziranih na LoD strukturi, koji su podijeljeni na operatore podjele (*engl.* „split") i spajanja (*engl.* „merge")(slika 4.7). Ako bi *L(j)* i *H(j)* predstavljali skupine atributa asociranih sa *LoD(j)* i *R(j)*, operator podjele bi uzeo *L(j+1)* kao ulaz i vratio primjerke niže rezolucije *L(j)* kao i primjerke više rezolucije *H(j)*. Operator spajanja bi međutim uzeo *L(j)* i *H(j)* kao ulaz i vratio *L(j+1)*.

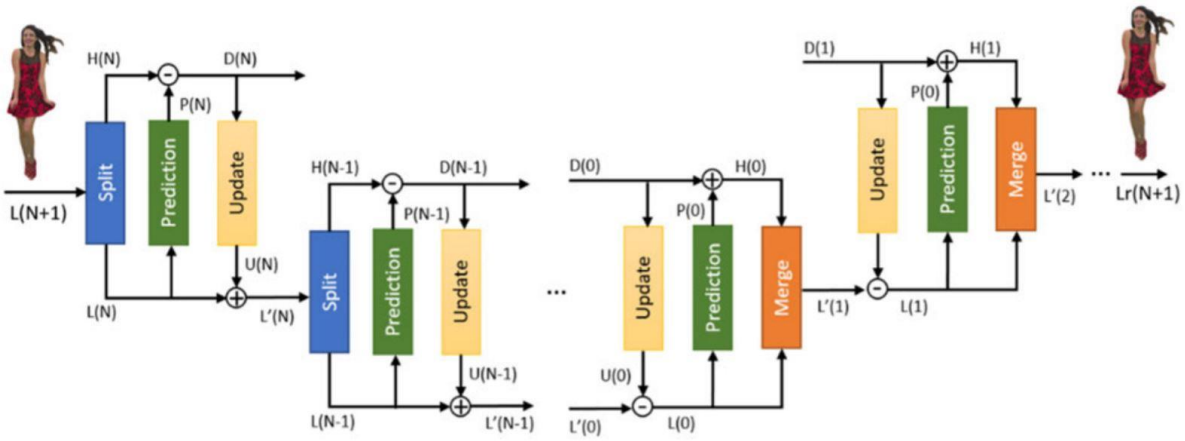


Slika 4.7: Predviđajuća transformacija, izvor: [9]

Kako je prikazano na slici, atributni signal L(N+1) koji predstavlja cijeli oblak točaka, podijeljen je na H(N) i L(N). L(N) korišten je za predviđanje H(N), a ostaci D(N) su tada izračunati, nakon čega se cijeli proces ponavlja.

Lifting Transformacija izgrađena je na predviđajućoj transformaciji te uvodi novi operator obnavljanja (*engl.* „Update") kao i adaptivnu kvantizaciju. Ovaj novi operator određuje U(j) bazirano na ostacima D(j) te nakon toga obnavlja vrijednost L(j) koristeći U(j).

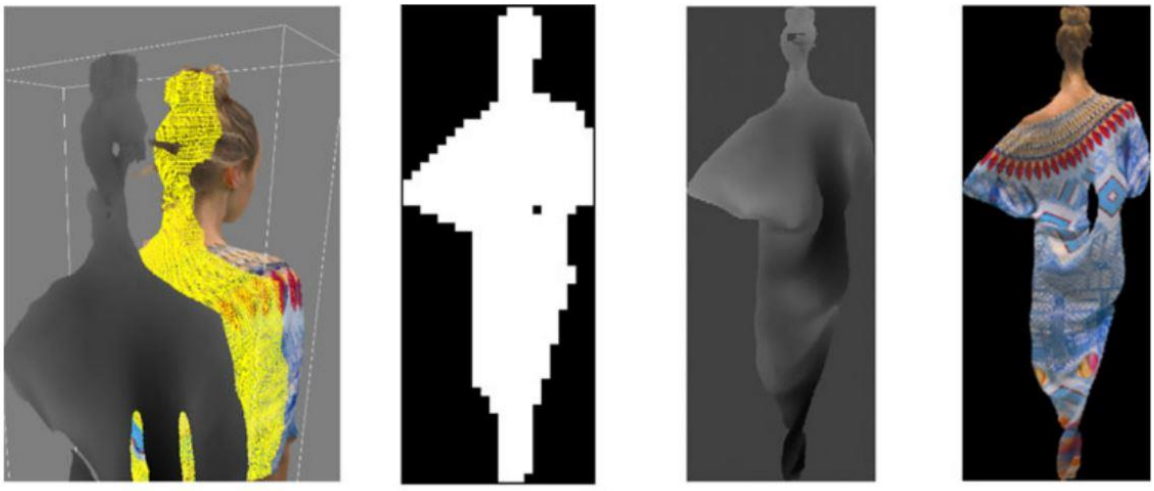
U procesu kvantizacije, transformirani koeficijenti asocirani sa svakom točkom multiplicirani su sa korijenom njihovih težina kao što je prikazano na slici 4.8.



Slika 4.8: Lifting transformacija kreirana na već postojećoj predviđajućoj transformaciji, izvor: [9]

4.2. MPEG V-PCC

Kako bi pojasnili način na koji radi V-PCC, prvo treba spomenuti glavne elemente/alate u samom radu kodera: generiranje zakrpa (*engl.* „patch“), pakiranje zakrpa, mapa zauzetosti (*engl.* „occupancy map“), geometrija, atributi, generiranje atlas slike, postavljanje slike i kompresija videozapisa. MPEG V-PCC kodira oblake točaka koristeći pristup projiciranja svakog oblaka točaka na skup ravnina (*engl.* „plane“) (najčešće šest) nakon čega slijedi njihovo 2D kodiranje. Uporaba samih alata nema specifičan način ili redoslijed izvođenja, što osobi koja vrši kodiranje omogućuje raznolikost i prilagodljvost. Tijekom standardizacije V-PCC-a, razvijen je novi referentni program: TMC_2 (Test Model Category 2) koji je zaslužan za generiranje glavnih alata V-PCC-a [**9**].



Slika 4.9: 3D zakrpa i njegove komponente s lijeva na desno:
3D zakrpa, mapa zauzetosti, geometrijska slika i slika tekstura
Izvor: [9]

4.2.1. 3D zakrpe i njihovo generiranje

3D zakrpe služe za kreiranje dvodimenzionalne slike sa što manjim brojem piksela, koja je zatim kodirana pomoću dvodimenzionalnog video kodiranja. Kako bi se generirale 3D zakrpe, procjena normala za svaku točku mora biti uspostavljena koristeći ortografske smjerove projekcije (±x, ±y, ±z). Svaka točka je zatim grupirana sa drugim točkama koje imaju isti smjer projekcije uz pomoć algoritma povezanih komponenata i te skupine su nazvane 3D zakrpe. Dobivene točke projicirane su ortogonalno na jednu od šest ravnina ovisno o osi na kojoj su bili pozicionirani.

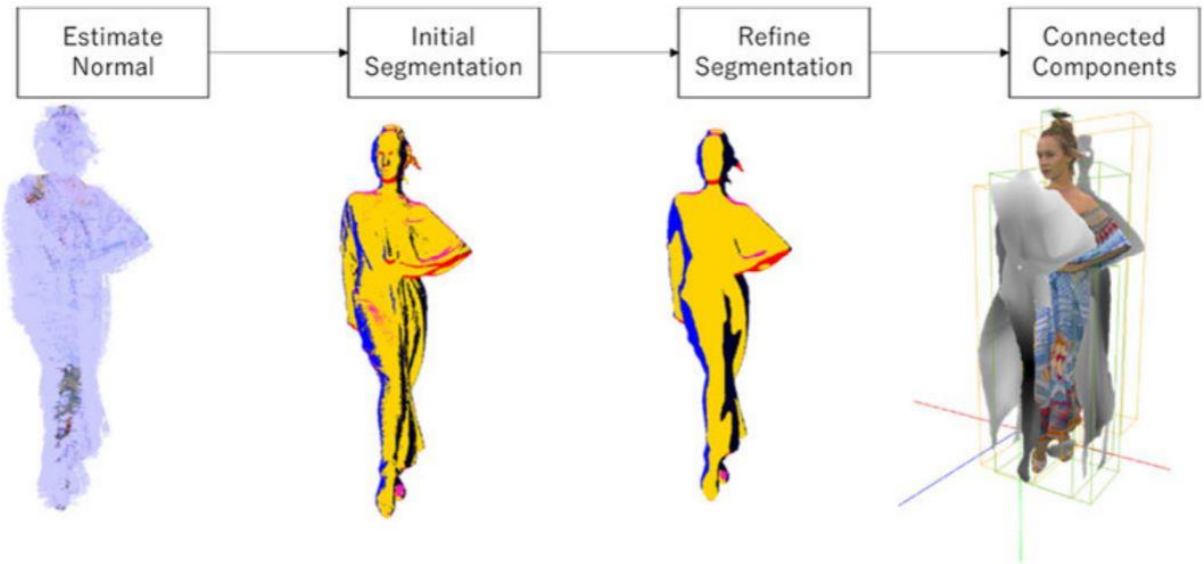
Pošto 3D zakrpa može imati više točaka projiciranih u jednu pikselsku lokaciju, TMC_2 koristi nekoliko takozvanih „mapa“ koje služe za definiranje dubine u slici (slika 4.9). Kao primjer, recimo da imamo skup točaka $H(x,y)$ projiciranih na istu (x,y) lokaciju te ih moramo

razdvojiti i definirati dubinu. TMC2 to radi na način da koristi dvije mape: blisku mapu (*engl.* „near map“) i daleku mapu (*engl.* „far map“). Bliska mapa će sadržati informacije točke *H(x,y)* sa najmanjom vrijednosti dubine *Do*. Daleka mapa sadrži točku sa najvećom vrijednosti dubine unutar intervala zadanog od strane korisnika (*Do*, *Do* + *D*) gdje D reprezentira širinu površine sa svrhom lakšeg kodiranja geometrije.

Pošto daleka mapa može potencijalno sadržavati informaciju o više elemenata na jednoj lokaciji, TMC2 (kada korišten u modu bez gubitaka) može preinačiti sliku daleke mape tako da kodira više točaka istovremeno umjesto samo jedne točke pomoću koda „enhanced delta-depth“ (EDD). EDD može indicirati kada je neka lokacija između najmanje vrijednosti dubine *Do* i najveće vrijednosti dubine D zauzeta te tu vrijednosti dodati na far map što omogućuje reprezentiranje više točaka koristeći samo dvije mape.

Nadalje, TMC2 je implementirao mnogo opcija filtracije i limitiranja samog kodiranja, poput eliminacije svih točaka sa vrijednostima dubine viših od zadanih iznosa, korištenjem informacije boje za odvajanje ravnina (kao i naknadni odabir najbolje ravnine za projiciranje dubine) te limitiranjem minimalnog broja točaka u 3D zakrpi (slika 4.10).

Također vrijedi spomenuti kako limitiranje minimalnog broja točaka u 3D zakrpi može rezultirati nekorištenim točkama koje bi bile ignorirane od strane samog TMC2 programa kada bi koristio kodiranje sa gubicima. Međutim, kodiranje bez gubitaka spremiti će informacije tih točaka u novu zakrpu koja tada može biti spremljena kao zasebna slika u obliku poboljšane slike (*engl.* „enhancement image“) ili nadodana na geometrijsku sliku *Do* za 2D kodiranje [**9**].



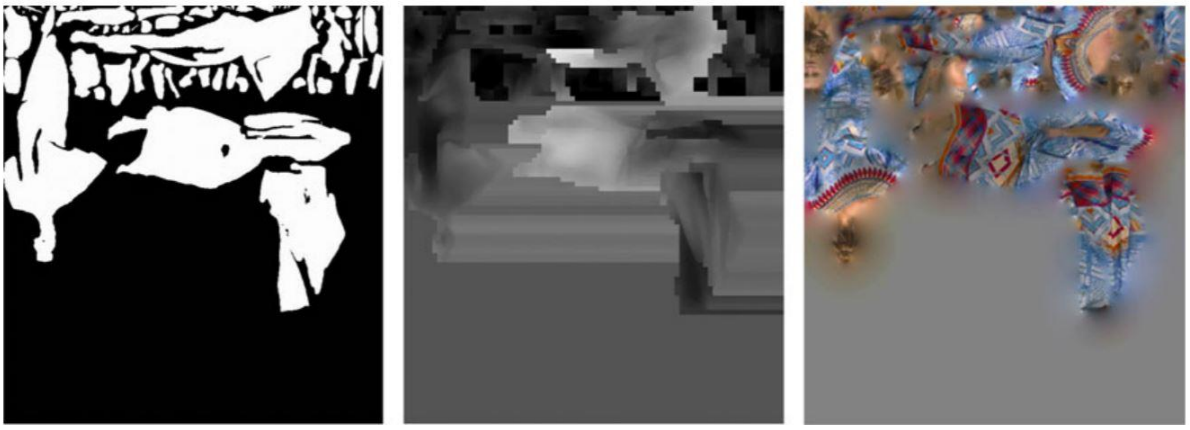
Slika 4.10: Generiranje 3D zakrpa, izvor: [9]

4.2.2. Pakiranje zakrpa

Takozvano „pakiranje“ zakrpa odnosi se na raspoređivanje projiciranih 2D zakrpa na 2D sliku veličine *W x H*. Zakrpe su prvo podijeljene i posložene ovisno o veličini, pa su nakon toga poredane koristeći rastersko skeniranje pomoću kojega se selektiraju slike čiji se blokovi ne preklapaju sa drugom zakrpom (slika 4.11).

U svrhu poboljšanja kompresije, zakrpe sa sličnim sadržajem trebale bi biti pozicionirane u slične pozicije kroz vremenski period. TMC_2 stoga pronalazi sličnosti u zakrpama te ih pokušava svrstati u slične lokacije koristeći IOU (Intersection Over Uunion) jednadžbu u kojoj preRect[i] reprezentira 2D graničnu kutiju same zakrpe [i] sa prethodnog oblaka točaka projiciranog u 2D prostoru, a Rect[j] predstavlja granični okvir zakrpe [j] sa trenutnog projiciranog oblaka točaka u 2D prostoru (formula 4.6).

$$Qi,j = \frac{\cap preRect[i],Rect[j]}{\cup preRect[i],Rect[j]} \tag{4.6}$$



Slika 4.11: Primjer spremanja zakrpa, izvor: [9]

4.2.3. Geometrijske mape i mape zauzetosti

Geometrijske mape u V-PCC-u, za raliku od standardnih geometrijskih mapa, spremaju udaljenost između nedostajućih koordinata pozicije točke u 3D prostoru i rubova graničnog okvira. Kako bi razlikovali piksele korištene za 3D rekonstrukciju i nekorištene piksele, TMC_2 stvara takozvane mape zauzetosti (*engl.* „occupancy maps“) za svaki pojedini oblak točaka [**9**].

Mape zauzetosti binarne su slike koje identificiraju i definiraju koji pikseli u 2D slikama sa teksturom i dubinom (*engl.* „texture and depth“) sadrže važne informacije. Kodirane su koristeći kodianje bez gubitaka video koder te pružaju fleksibilnu preciznost zadanu od strane kodera. Zauzetost samog bloka *B x B* označava se vrijednostima 1 i 0.

Kod kodiranja videozapisa, binarna slika dimenzija (W/B, H/B) spremljena je u kanal svjetline (*engl.* „luminance channel“) te je korišteno kodiranje bez gubitaka. Međutim, kodiranje s gubicima je također moguće, ali nije preporučeno s obzirom da može dodati neželjene nove točke zbog skaliranja mape zauzetosti.

4.2.4. Postavljanje slike, re-apliciranje boje i video kompresija

Ovi postupci služe za definiranje graničnih piksela i samog oblika oblaka točaka. Vrši se provjera nad svakim blokom *T x T* sa svrhom pronalaženja praznih blokova ili blokova čije vrijednosti nisu upotpunjene. Kod praznih blokova, informacije sa susjednih blokova kopirane su i korištene kako bi se oni ispunili, a nepotpune vrijednosti popunjene su sa prosječnom vrijednosti susjednih blokova.

Radi olakšane kompresije, bliska mapa i daleka mapa nisu promijenjene pa je zbog toga korištena grupna dilatacija u kojoj je izračunata srednja vrijednost praznih blokova, te je ta vrijednost korištena za obje mape [**9**].

Radi izobličenja geometrije, TMC_2 uzima originalnu boju i dodaje ju na novi oblak uzimajući u obzir sve najbliže točke i njihove originalne boje te ih mapira sa 3D u 2D.

4.2.5. Toniranje i redefiniranje geometrije i atributa

U slučaju kada TMC_2 koristi dvije mape kako bi definirao dubinu (bliska mapa i daleka mapa) moguće je generiranje identičnih kopija točaka što može utjecati na samu kvalitetu slike. Stoga je proces rekonstrukcije modificiran na način da stvara samo jednu točku po svakoj koordinati (*u*,*v*).

Za vrijeme generiranja mapa i zakrpa, mnogo novih slojeva sa jedinstvenim atributima može biti dodano na oblak točaka koji lako mogu stvoriti nesklad sa drugim atributima. Zbog toga, TMC_2 sadrži opciju redefiniranja informacija i atributa te odstranjivanja dvostrukih atributa. Ovaj je proces proveden u 3D prostoru, kako bi se u obzir uzele informacije susjednih točaka.

4.3. Kompresija pomoću panorama

LiDAR je metoda daljinskog očitavanja koja koristi svjetlost u obliku pulsirajućeg lasera za mjerenje udaljenosti od Zemlje. Koristeći ovu metodu, zajedno sa ostalim kombiniranim informacijama snimljenima iz zraka, moguće je generirati precizne oblake točaka o obliku i karakteristikama terena kojeg snimamo.

Informacije i podatci dobiveni pomoću LiDAR-a većinom su spremljeni u zasebnim binarnim podacima te ASCII reprezentaciji za dijeljenje između drugih korisnika. Međutim, kapacitet ovakvih podataka može zauzimati izuzetno mnogo prostora pa je stoga kreirana nova metoda kompresije oblaka točaka koristeći panorame. Panoramske slike sadrže mnogo različitih atributa poput udaljenosti, refleksije i vrijednosti boje. Generalna zamisao ovakve kompresije je da skenirana okolina može biti projicirana u sferu sa skenerom pozicioniranim u samom centru [**10**].

Mapiranje sferičnih koordinata na dvodimenzionalne koordinate već je poznata metoda i korištena je u kartografiji sa mnogo različitih metoda kreiranih kroz godine. Najjednostavnija vrsta projekcije je kvadratična projekcija (*engl.* „equirectangular“) u kojoj se dužina i širina mapiraju u horizontalne i vertikalne koordinate što se vrlo jednostavno može koristiti u 360° horizontalnom te 180° vertikalnom pogledu. **Merkatorova i Pannini projekcija** kompleksnije su metode koje teže očuvanju kuteva i matematičkom preciznošću.

Kod kompresije pomoću panorama, idealna metoda kompresija biti će kompresija bez gubitaka.

4.3.1. Korištenje uobičajene kompresije slike

JPEG standardi sadrže mnogo metoda za kompresiju slike. Najkorištenija metoda, kompresija s gubicima, bazirana na transformaciji kosinusa (DCT) koja grupira izvorna slike u 8x8 blokove čije su vrijednosti centrirane prema 0. Tada se računa DCT nad blokovima, te se potom vrši kvantizacija. Zbog kvantizacije, ova metoda rezultira s većim gubicima [**10**].

Međutim, JPEG standarni također sadrže i manje poznatu metodu kompresije bez gubitaka nazvanu JPEG 2000 koja je bazirana na diskretnoj wavelet transformaciji (engl. „Discrete Wavelet Transform“, DWT), skalarnoj kvantizaciji, kontekstualnom modeliranju i aritmetičnom kodiranju. JPEG 2000 pruža oboje: kompresiju sa i bez gubitaka.

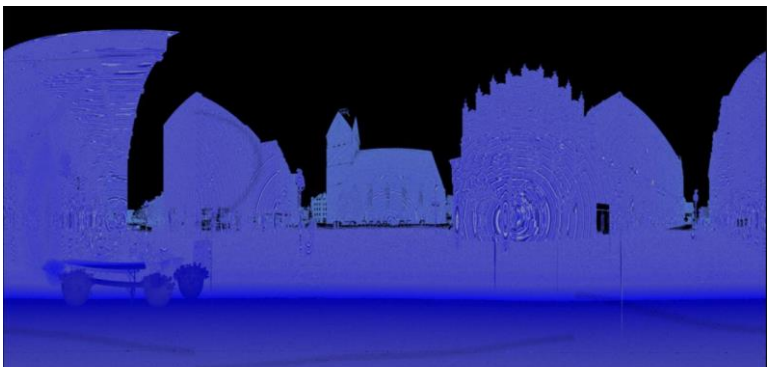
Portable Network Graphics (format za pohranu slika, PNG) je besplatan te je preporučen od strane W3C-a za kodiranje slika. Pruža jednostavno spremanje bez gubitaka. Ova vrsta kompresija izvršava se u dva koraka: predviđanje sheme i entropijsko kodiranje. Predviđanje koristi petero jedinica za predviđanje te se vrši na najbližem susjedu. Entropijsko kodiranje koristi metodu kompresije bez gubitaka nazvanu **Deflate algoritam**. PNG je metoda kompresije bez gubitaka te podržava kanale svjetline, boje i transparentnosti.

Tagged Image File Format (TIFF) se, kao i PNG, često koristi kod spremanja slika. Međutim, TIFF može sadržavati neobrađene podatke slike. TIFF također podržava mnoge metode kompresije. Najpoznatije metode koje TIFF podržava su: algoritam nazvan **Lampel-Ziv-Welch** (LZW), te metoda po imenu **PackBit**.

4.3.2. Kodiranje udaljenosti i kompresija oblaka točaka

Prije vršenja bilo kakve kompresije, atributi udaljenosti, boje i refleksije moraju biti razdvojeni u zasebne panoramske slike. Informacije o refleksiji obično su indicirane sa 8 bitnom sivom vrijednošću, a informacije o boji sa 8 bitnom crvenom, zelenom i plavom vrijednošću. Informacije o udaljenosti nemaju zadanu vrijednost nego pokazuju udaljenost između snimljenog prostora i kamere/skenera. Jedna informacija o udaljenosti u decimalnom zapisu (float) može imati do 32 bita pa se stoga float mora konvertirati u cijeli broj (integer) kako bi se generirala vrijednost udaljenosti sa 0.1 mm preciznošću [**11**].

Kod samog prikazivanja udaljenosti na panoramskoj slici, koriste se RGB boje ovisno o vrijednosti/udaljenosti. Vrijednost plave boje najniža je vrijednost pa je zato najbliža 0 te se povećava svakih 6.5 metara otprilike. Nakon plave, dolazi zelena boja, a nakon zelene crvena. RGB vrijednosti tada su konvertirane u HSL (Hue-Saturation-Lightness) model boja i sortirani bazirano na boji, zasićenosti i intenzitetu (slika 4.12). Za prirodniji izgled oku, RGB može također biti konvertiran u YCbCr što također smanjuje vidljivost bilo kakvih pogrešaka u slici.



Slika 4.12: Pimjer prikaza udaljenosti kod kompresije slike, izvor: google slike

4.4. Kompresija dinamičkih oblaka točaka pomoću projekcija

Dinamički oblaci točaka i njihova kompresija, za razliku od statičkih oblaka točaka, zahtjevaju jaču kompresiju i više vremena. Međutim, nova metoda kompresije dinamičkih oblaka točaka pomoću projekcija nedavno je započela sa testiranjem. Za metodu konverzije oblaka točaka u projekciju i obrnuto, korišten je program 3D ToolKit (3DTK). Nakon konverzije, korišteni su FFmpeg i x265 koderi (H.265/HEVC) kako bi se sve projekcije spojile u jednu video sekvencu. Za kraj, normale za svaku skupinu točaka izračunate su koristeći 16 susjednih točaka te je također korišten rekonstrukcijski algoritam Poisson sa vrijednošću rekonstrukcijske dubine 12 kako bi se postinguo sveukupan bolji izgled komprimiranih oblaka točaka [**12**].

Kod korištenja ovakve metode kompresije, metoda bez gubitaka pružati će puno bolje rezultate pošto metoda s gubicima može stvoriti dodatan neželjeni šum.

5. Objektivne mjere za ocjenjivanje kvalitete oblaka točaka

Modeli i oblaci točaka korišteni u 3D printingu sadrže izuzetno mnogo podataka i informacija, stoga je kompresija obavezna kako bi se olakšalo rukovanje svim tim informacijama. Svaka vrsta kompresije dati će nam drugačije rezultate te su zbog toga uvedene neke objektivne mjere za ocjenjivanje same kvalitete oblaka točaka nakon kompresije. Objektivne mjere mogle bi se podijeliti na tri skupine: **Full Reference** (FR) mjere – mjere koje zahtjevaju originalne slike ili video sekvence koje nisu komprimirane, **Reduced Reference** (RR) mjere – zahtjevaju informacije dobivene od originalne slike/video sekvence, **No-Reference** (NR) – zahtjevaju samo procesiranu sliku/video sekvencu [**13**].

Jedna od prvih objektivnih mjera bila je metoda imenom „**point-to-point distances**“. Ova metoda pridružuje svakoj točci u oblaku po jednu točku s drugog oblaka (originalnog oblaka). Nakon toga, izračunata prosječna udaljenost između parova točaka korištena je kao mjera. Još jedno poznato ime ove metode je **cloud-to-cloud** (C2C).

Međutim, C2C rijetko uspjeva definirati neke od točaka koje reprezentiraju površinu samog modela/strukture. Zbog ovog razloga, proučena je nova metoda koja bi uspješno reprezentirala površinu, imenom **point-to-surface** ili **cloud-to-mesh** (C2M). Cloud-to-mesh prvo konstruira model iz referentnog oblaka točaka, zatim su udaljenosti između svih točaka izračunate i korištene kao baza za C2M mjere, što je također postalo glavna metoda računanja udaljenosti oblaka točaka u nekim od programa poput CloudCompare.

Kako bi ih mogli pregledati, oblaci točaka prvo moraju uvijek biti renderirani, pa je stoga razvijen novi potencijalni način evaluacije pomoću projekcije na ravnine. Projekcija na slikovnu ravninu prikazuje originalni oblak točaka prije kompresije što može služiti kao referenca te su pomoću toga izračunate normale. Ova metoda prvo pridružuje svaku točku u refrentnom oblaku sa točkom u degradiranom oblaku. Nakon toga, za svaku točku u referentnom oblaku, uzima se vektor normale zadane jedinice ako je moguće. Ako to nije moguće, vektor normale se procjenjuje koristeći najbližu susjednu točku računajući matricu kovarijance. Za kraj, izračunat i projiciran je point-to-nearest-point vektor kao i korijen magnituda svih projiciranih error vektora (point-to-plane distorzija). Komprimirani oblak točaka, tada je projiciran na istu slikovnu ravninu kako bi se generirala distorzija/iskrivljenje. Samo pronalaženje najbliže točke može često potrajati te se stoga koristi jedna od metoda optimizacije, najčešće **KD tree optimizacija**. Ova je metoda također nazvana **point-to-plane** ili **cloud-to-plane** (C2P).

U istraživanjima oblaka točaka, kao i kod ostalih istraživanja, potreba za minimalnom prosječnom kvadratnom razlikom između očekivanih i dobivenih vrijednosti uvijek je željena. Takvu razliku nazivamo **Mean Square Error** (MSE). Sa ciljem olakšanja MSE-ova dobivenih iz više oblaka točaka, te su vrijednosti pretvorene u **Peak Signal to Noise Ratio** (PSNR) vrijednosti sa dodanom jedinicom za oznaku najviše vrijednosti (peak value) p, čija je vrijednost određena (formula 5.1) [**14**].

$$PSNR_{A,B} = 10log_{10}\frac{p^2}{e_{A,B}} \tag{5.1}$$

Vrijednost broja p često koristi istu vrijednost dijagonalne udaljenosti okvira u kojoj se nalazi oblak točaka. Iako bi veći objekt stvorio veće PSNR vrijednosti, a manji objekt manje PSNR vrijednosti, njihova točnost trebala bi biti vrlo bliska pošto svaka točka koristi isto iskrivljenje/distorziju.

Hausdorff (Haus) mjera predstavlja maksimalnu udaljenost mjerenu kao najveću udaljenost između svih točaka u izobličenom oblaku i njihovim najbližim susjednim točkama u referentnom oblaku (formula 5.2).

$$d_{Haus}^{S} = \max(d_{Haus}^{A,B}, d_{Haus}^{B,A}) \tag{5.2}$$

Udaljenost od izobličenog prema originalnom i originalnog prema izobličenom oblaku točaka nikada nije jednaka te se stoga uzima veći MSE od zadanih, koji je tada nazvan simetrični MSE.

Kod kodiranja s gubicima atributa za boje, MSE svih triju boja mora biti izračunat zasebno. RGB prostor pretvoren je u YUV prostor koristeći standard ITU-R BT.709 (pošto je YUV prostor prirodniji ljudskog percepciji). Nakon toga se simetrično izračuna pogreška, slično kao i kod geometrijske pogreške. Ukoliko atributi boje imaju dubinu bitova jednakoj 8 bitova po točki, p (peak value) za računanje PSNR-a ispada 255.

6. Subjektivne metode evaluacije oblaka točaka

Multimedia user experience (MUE) kombinacija je psihofizičkih odaziva na boju, pokret, teksture, zvuk i kontekst te postiže najbolju preciznost putem opažanja koristeći skupinu sudionika. Subjektivna kvaliteta dobivena je pomoću **Mean Opinion Score-a** (MOS) koji je dobiven od strane sudionika.

Jedan od najpoznatijih i najkorištenijih subjektivnih protokola za ocjenjivanje kvalitete videozapisa naziva se **ITU-R BT.500-14** [**17**]. Taj protokol definira i pruža razne vrste metoda koje se najčešće djele na jednopodražajne (*engl.* „single stimulus“) ili dvopodražajne (*engl.* „double stimulus“), ovisno o načinu prikazivanja sadržaja sudionicima. Neki od primjera metoda mogu biti: **Single-Stimulus** (SS), **Double Stimulus Continuous Quality Scale** (DSCQS), **Stimulus Comparison** (SC), **Single Stimulus Continuous Quality Evaluation** (SSCQE), **Double Stimulus Impairment Scale** (DSIS) [**15**].

Double Stimulus Continuous Quality Scale (DSCQS) najkorištenija je metoda u kojoj sudionici ocjenjuju sveukupunu kvalitetu niza video sekvenci ili slika prezentiranih zasebno ili u parovima gdje jedna slika/video služi kao originalna dok je druga slika/video prošla kroz neku vrstu kompresije. Single Stimulus Continuous Quality Evaluation (SSCQE) koristi metodu u kojoj sudionici ocjenjuju sekvencu slika/videozapisa koji sadrže komprimirane datoteke te variraju kroz vrijeme.

Noviji protokol ITU-R BT.2021 baziran je na ITU-R BT.500-13 te je proširen kako bi uveo subjektivne procjene za 3DTV sisteme gdje su, osim očitog napretka u kvaliteti slike, dodane dvije nove ocjene: dubina kvalitete i vizualna udobnost. Prije bilo kakve evaluacije vršene pomoću 3DTV sustava, stereoskopska vidljivost kao i boja i vizualna oštrina sudionika moraju prvo biti testirani. Modeli također mogu biti redneririrani prije istraživanja za lakše pregledavanje vizualnih podataka [**16**].

Subjektivna evaluacija oblaka točaka koristeći neobrađene formate oblaka točaka testirana je već u mnogo različitih okolina sa dvije vrste degradacije: **Gaussov šum** i octree kompresija. U tim testiranjima, sudionici su mogli vršiti interakciju sa oboje originalnim i izobličenim oblacima točaka. Objektivne mjere dobro su predvidjele MOS za oblake točaka izobličenih Gaussovim šumom, a slabije su korelirale s MOS rezultatima za oblake točaka izobličene octree kompresijom.

Double stimulus impairment scale (DSIS) je metoda u kojoj je sudioniku prvo prikazana slika bez kompresije, zatim zajedno sa slikom koja je komprimirana [**17**]. Nakon toga, sudionik daje ocjenu (najčešće od 1 do 5 gdje 1 predstavlja loš i oku neugodan izgled slike, dok 5 predstavlja izgled koji je izvrstan). Ove sesije mogu trajati od 30 minuta do sat vremena u kojima su sudioniku prikazane sekvence slika u nasumičnom radoslijedu sa svim mogućim kombinacijama komprimiranih slika. Slika bez kompresije također je ubačena u te sekvence. Na kraju testiranja izračunati je MOS za svaku sekvencu što rezultira boljim rezultatima kada su u testiranju korištene slike sa slabijom kompresijom. Iako ova metoda često može limitirati jačinu kompresije koja će se prikazivati u slikama, preporučeno je korištenje punog spektra kompresija u slikama [**18**].

7. Evaluacija

Subjektivna istraživanja provedena su u četiri različita laboratorija: University of Beira Interior (UBI), Covilhã, Portugal; University of Coimbra (UC), Coimbra, Portugal; University North (UNIN), Varaždin, Hrvatska and University of Technology Sydney (UTS), Australija. Uvjeti za okruženje u kojem je istraživanje provedeno prilagođeno je prema preporuci ITU-R Recommendation BT.500-13 [**18**]. Za ovaj rad korišteni su podaci sa UNIN-a i UC-a.

Za sam prikaz zadanih oblaka točaka korištena je modificirana verzija MPV video playera u kojem su oblaci točaka prikazani kao kraći videozapis od 60 slika u sekundi kroz 12 sekundi, nakon čega su sudionici upisali svoje ocjene. Oblaci točaka korišteni u ovom istraživanju rotirani su po Y osi kako bi se omogućio pogled iz više kutova.

Za pripremu sadržaja kodiranog pomoću V-PCC-a korišteni je referentni program sa parametrima kodiranja definiranima u MPEG CTC dokumentu. Korišteni uvjet kodiranja bio je C2, Lossy Geometry – Lossy Atributes, te All Intra (AI) kao mod.

Za samu evaluaciju istraživanja, odabrano je šestero ocjena, petero uzetih sa MPEG CTC dokumenta (R01-R05) te još jedna ocjena niža od R01 kako bi se lakše usporedila sa najnižom ocjenom u G-PCC-u.

7.1. Detalji o istraživanju

Na samom početku evaluacije, sudioniku je prikazan testni oblak točaka koji nije dio same evaluacije kako bi se upoznali sa procesom evaluacije. Istraživanje je provedeno sa tri kodera: V-PCC, G-PCC (Octree), G-PCC (TriSoup). Broj bitova po točci (*engl.* „bits per point“, bpp) za svaki primjer postavljeni je na oko 0.1-4.5 bpp. Sveukupno 96 videa prikazano je sudionicima [**19**].

Modeli korišteni u istraživanju su skup oblaka točaka (x, y, z) položeni u trodimenzionalnoj mreži na specifičnim volumetrijskim elementima (*engl.* voxel) te su stoga nazvani **„voxelized point clouds“**.

Detaljniji podaci o vidljivosti, osvjetljenju, monitorima, udaljenosti te broju i dobi sudionika opisani su u tablici 1.

Tablica 1: Detalji o istraživanju za UNIN i UC

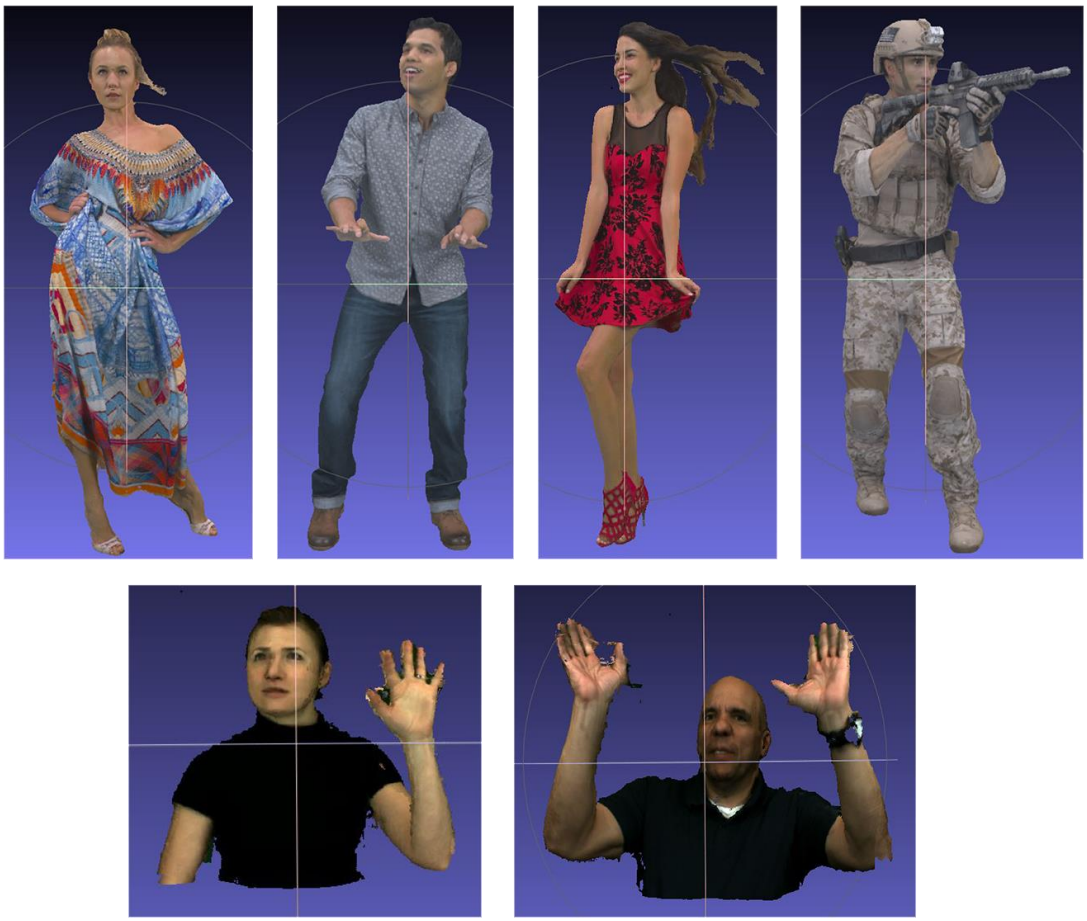
	Detalji o prikazu	**Udaljenost**	**Vidljivost**	**Osvjetljenje**	**Broj sudionika**	**Dob sudionika**
University North, Croatia	Monitor: Sony TV KD-55x8505C	1.5m ±15 cm	Provjerena vidljivost, boje	Konstantno osvjetljenje od 6500K	15 sudionika: 5 ženskih 10 muških	Prosječna dob: 29 g. Raspon: 19-59 g.
University of Coimbra, Portugal	Monitor: Sony KD-49X8005C	1.8m ±30cm	Provjerena vidljivost, boje	Kontrolirano osvjetljenje	15 sudionika: 8 ženskih 7 muških	Prosječna dob: 28 g. Raspon: 18-54 g.

Dinamične „voxalized point cloud“ sekvence zabilježene su imenom 8i Voxelized Full Bodies (8iVFB) [**20**]. U ovom skupu podataka postoji 6 test modela (longdress, loot, soldier, redandblack, ricardo i sarah), 5 razina izobličenja, 3 vrste kompresije te 6 originalnih modela (slika 14) što sveukupno ispada 96 modela (uključujući i originalne). Cijelo tijelo subjekta obuhvaćeno je pomoću 42 RGB kamere, posložene u 14 skupina gdje svaka skupina služi kao RGB-D kamera za mjerenje dubine slike. Snimanje objekta vršeno je sa 30 FPS (frames per second) preko perioda od 12 sekundi.

Svaka je sekvenca postavljena u zasebnu kocku veličine 1024x1024x1024 voksela tako da su samo vokseli u blizini objekta zauzeti. Sama kocka skalirana je ovisno o visini subjekta, a pošto je visina subjekta najdulja dimenzija, za subjekta visokog 1.8 metara potreban je voksel veličine ≈1.75 mm (formula 7.1).

$$\frac{1.8m}{1024voxels} \approx 1.75mm \tag{7.1}$$

Svaki model kodiran je kroz više bit rate-ova (slike 7.1-7.19) određenih od strane MPEG-a (r01, r02, r04, r05, r06 za octree i trisoup, a r01, r02, r04, r05, rb01 za V-PCC) [**21**].



Slika 7.1: Originalni oblaci točaka prije kompresije



Slika 7.2: Longdress V-PCC (r01, r02, r04, r05, rb01)



Slika 7.3: Longdress octree (r01, r02, r04, r05, r06)



Slika 7.4: Longdress TriSoup (r01, r02, r04, r05, r06)



Slika 7.5: Loot V-PCC (r01, r02, r04, r05, rb01)



Slika 7.6: Loot octree (r01, r02, r04, r05, r06)



Slika 7.7: Loot TriSoup (r01, r02, r04, r05, r06)



Slika 7.8: Redandblack V-PCC (r01, r02, r04, r05, rb01)



Slika 7.9: Redandblack octree (r01, r02, r04, r05, r06)



Slika 7.10: Redandblack TriSoup (r01, r02, r04, r05, r06)



Slika 7.11: Soldier V-PCC (r01, r02, r04, r05, rb01)



Slika 7.12: Soldier octree (r01, r02, r04, r05, r06)



Slika 7.13: Soldier TriSoup (r01, r02, r04, r05, r06)



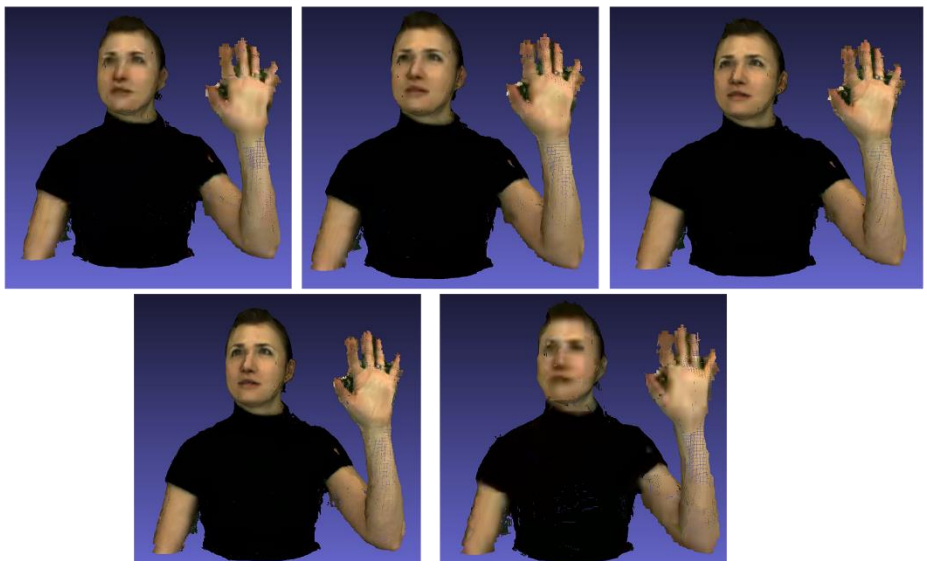
Slika 7.14: Ricardo V-PCC (r01, r02, r04, r05, rb01)



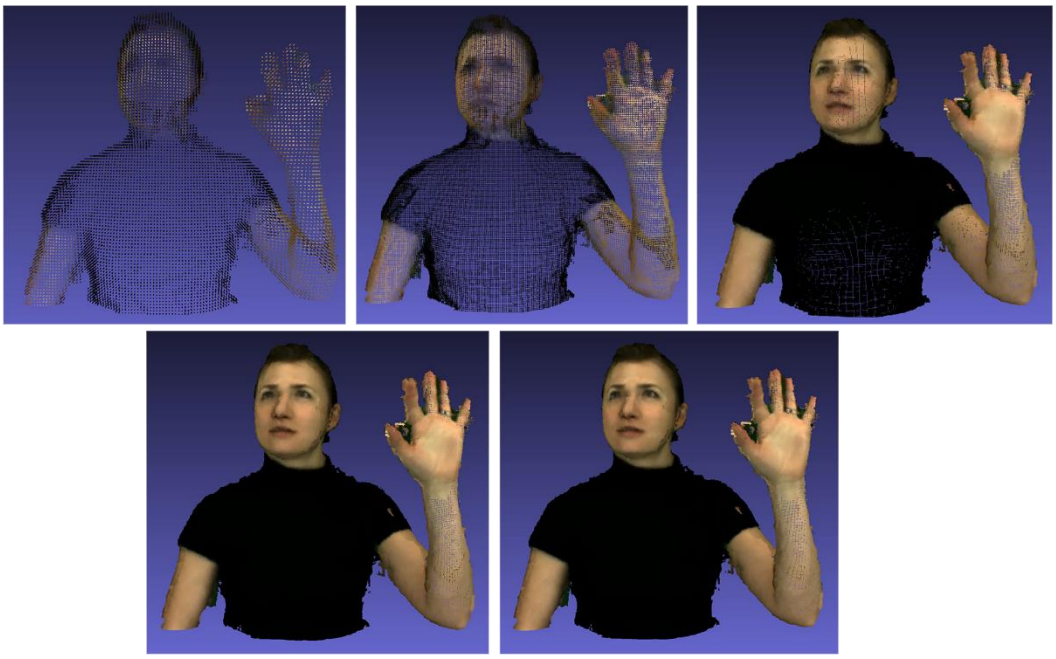
Slika 7.15: Ricardo octree (r01, r02, r04 ,r05, r06)



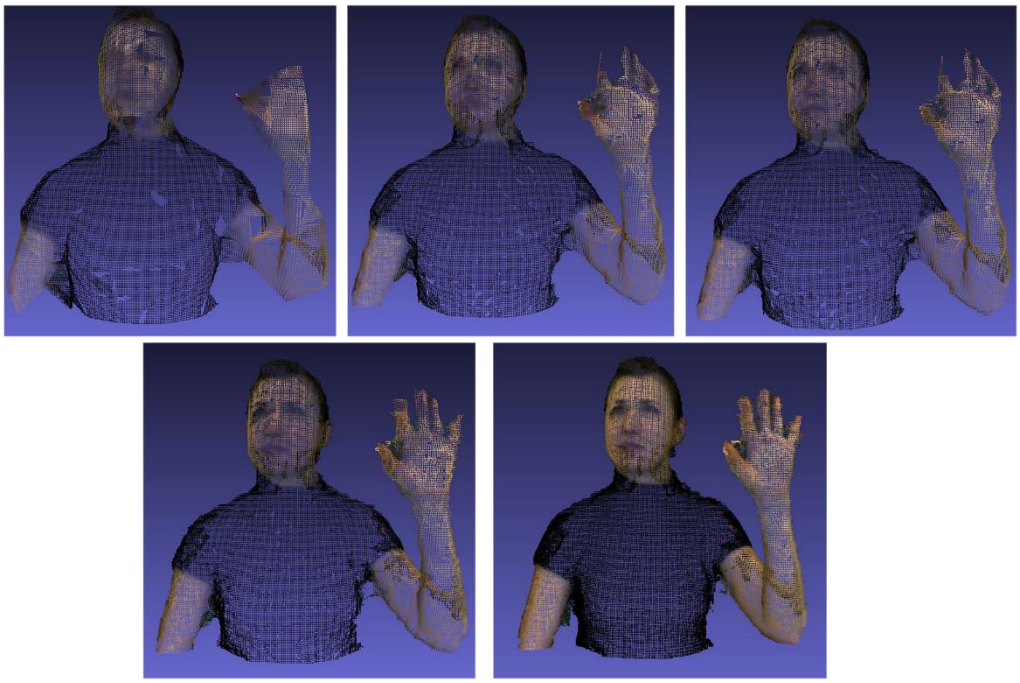
Slika 7.16: Ricardo TriSoup (r01, r02, r04, r05, r06)



Slika 7.17: Sarah V-PCC (r01, r02, r04, r05, rb01)



Slika 7.18: Sarah octree (r01, r02, r04, r05, r06)



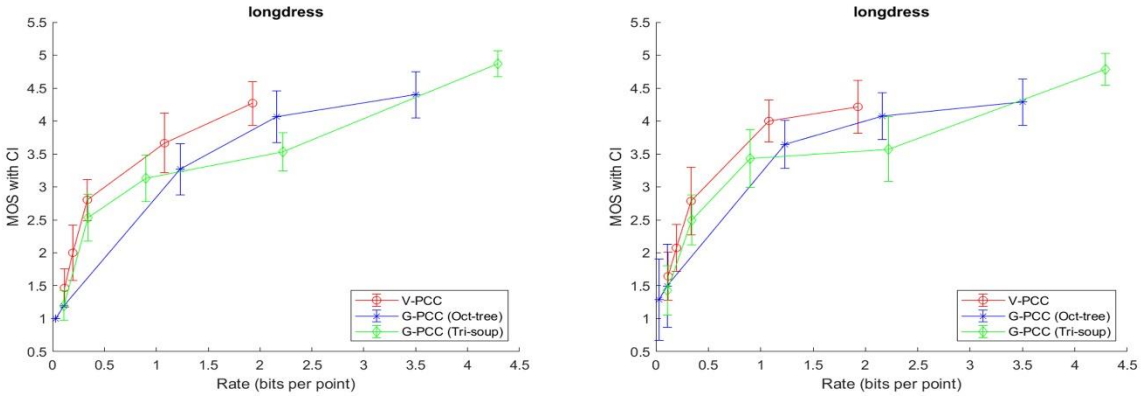
Slika 7.19: Sarah TriSoup (r01, r02, r04, r05, r06)

8. Rezultati

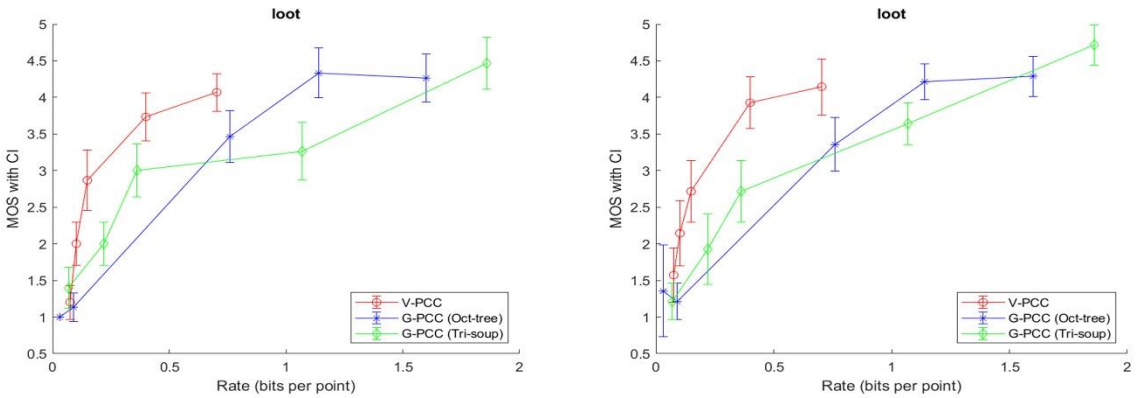
Razultati evaluacije definirani su grafom u kojem se prikazuje MOS po bpp (bits per point) za svaki oblak točaka nakon skupljanja svih dobivenih rezultata u svakon laboratoriju. Bitrate je mjeren u bpp (bits per point) kao sveukupan broj bitova potrebnih za određeni stimulus podijeljen sa brojem ulaznih točaka (input points) odgovarajućeg sadržaja.

Mean opinion scores (MOS) predstavlja vrijednost za mjerenje kvalitete/iskustva, to jest, predstavlja cjelokupnu kvalitetu podražaja ili sustava.

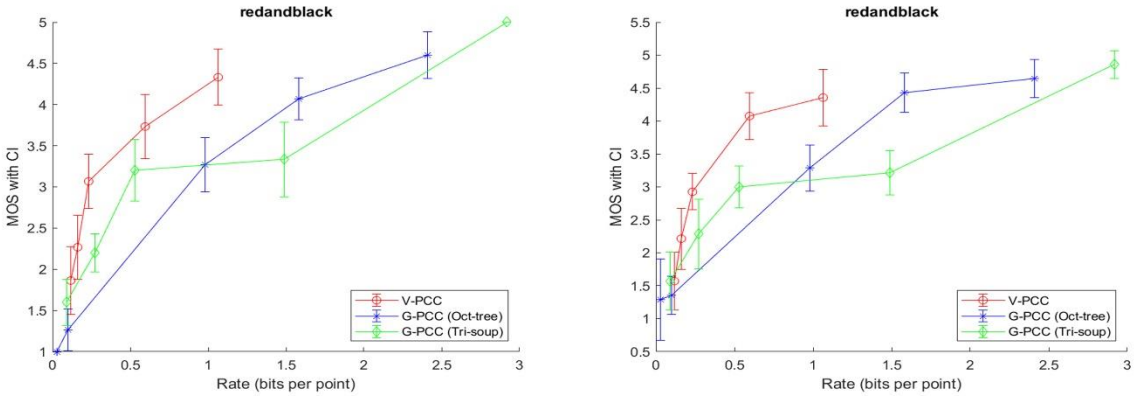
Rezultati subjektivnih ocjena po modelima (slike 8.1-8.6):



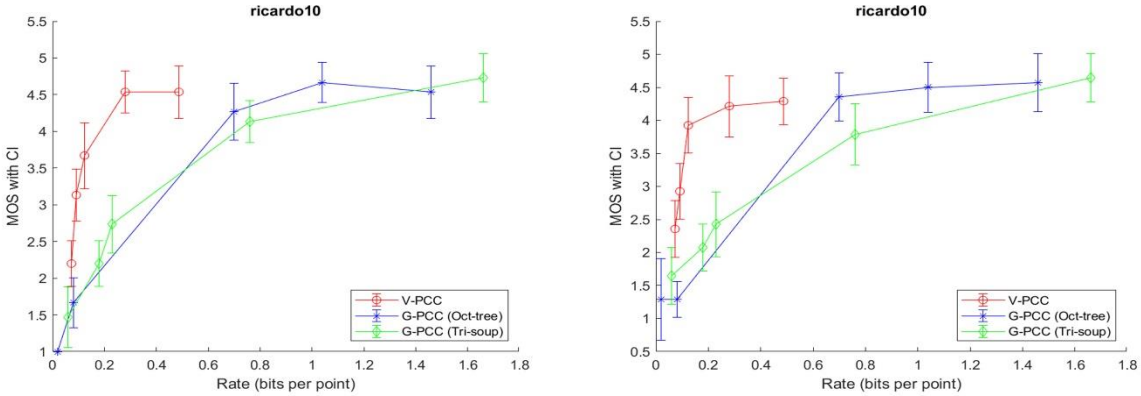
Slika 8.1: Rezultati za longdress (UNIN-lijevo, UC-desno)



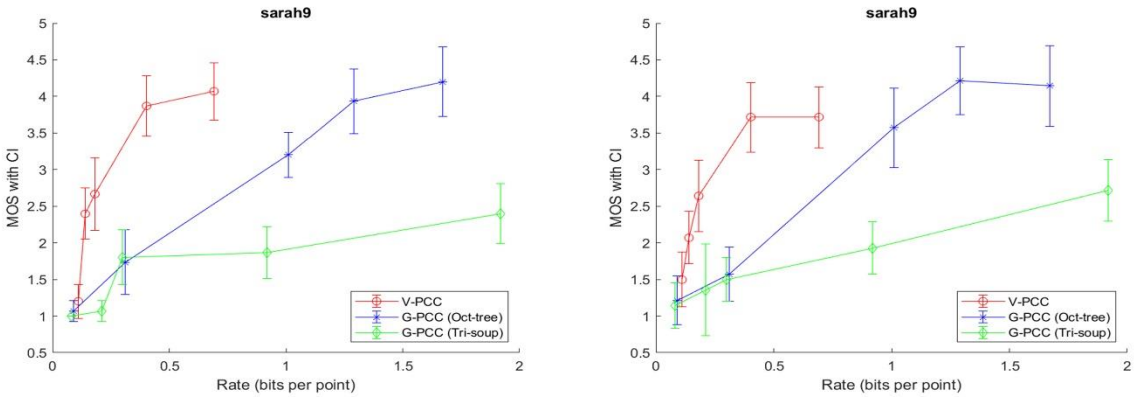
Slika 8.2: Rezultati za loot (UNIN-lijevo, UC-desno)



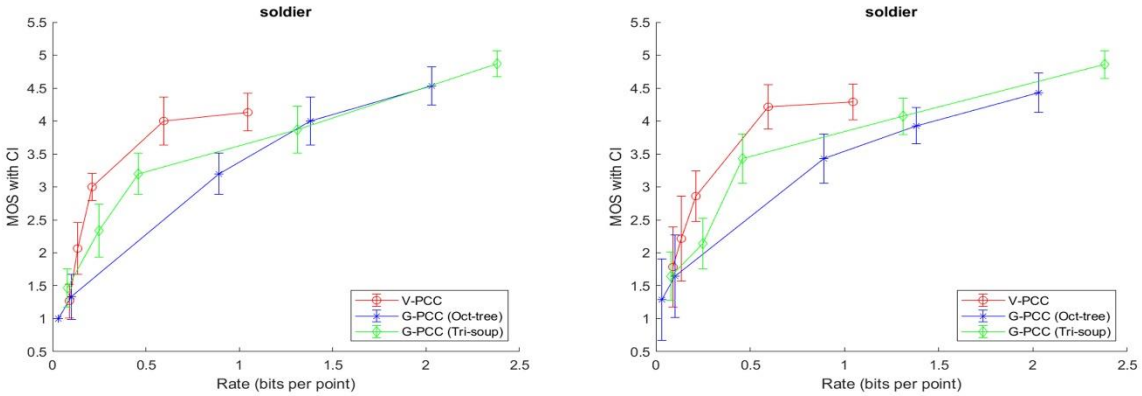
Slika 8.3: Rezultati za redandblack (UNIN-lijevo, UC-desno)



Slika 8.4: Rezultati za ricardo (UNIN-lijevo, UC-desno)



Slika 8.5: Rezultati za sarah (UNIN-lijevo, UC-desno)



Slika 8.6: Rezultati za soldier (UNIN-lijevo, UC-desno)

8.1. Usporedba istraživanja iz drugih laboratorija

Kako bi se dokazala valjanost, preciznost, konzistencija i linearnost dobivenih rezultata, provedene su sljedeće metode: Pearson Correlation Coefficient (PCC), Spearman Rank Order Correlation (SROCC), Kendall's Correlation (KROCC), Root Mean Square Error (RMSE) te Outlier Ratio (OR).

Pearson Correlation Coefficient (PCC) izračunat je kao normalizirana kovarijanca između dvije varijable, x i y:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{(n-1)*s_x*s_y} \tag{8.1}$$

U kojem su x_i i y_i vrijednosti rezultata objektivnih (x) i subjektivnih (y) ispitivanja, dok $\bar{x}$ i $\bar{y}$ predstavljaju srednje vrijednosti, a s_x i s_y standardne devijacije varijabli x i y.

Spearman's correlation (SROCC) procjenjuje kako dobro proizvoljna monotona funkcija može opisati vezu između dvije varijable bez pretpostavljanja raspodjele frekvencije samih varijabli. SROCC kalkuliran je slično kao i PCC, međutim, svaki uzorak obiju varijabli rangiran je ovisno o redoslijedu samih uzoraka (prosječan rang pridružen je uzorcima koji su imali identične rangove).

Kendall's correlcation (KROCC) također je baziran na rangiranju te nadgleda i spaja rangirane varijable u parove koje tada dijeli na 3 skupine: suglasne parove (gdje x i y imaju jednak smjer), proturječne parove (gdje x i y imaju suprotne smjerove) te moguće identične parove. Uz to, KROCC se također sastoji od triju tipova koeficijenta: T_a, T_b i T_c. Koeficijent T_a ne vrši nikakve prilagodbe za moguće identične parove, za razliku od T_b i T_c. Koeficijent T_b

korišteni je kada osnovni raspon obiju varijabli ima jednak broj mogućih vrijednosti prije rangiranja, a koeficijent T_c koristi se kada je broj mogućih vrijednosti različiti.

Nakon nonlinearne regresije, Kendallova korelacija (KROCC) izračunata je pomoću četiri (Q_1-Q_4) parametara logičkih funkcija izračunatih pomoću formula 8.2-8.5.

$$Q_1(z) = b_1 * \left(\frac{1}{2} - \frac{1}{1+e^{b_2*(z-b_3)}}\right) + b_4 * z + b_5 \tag{8.2}$$

$$Q_2(z) = \frac{b_1 - b_2}{1+e^{\frac{z-b_3}{b_4}}} \tag{8.3}$$

$$Q_3(z) = b_1 * z^3 + b_2 * z^2 + b_3 * z + b_4 \tag{8.4}$$

$$Q_4(z) = b_1 * z + b_2 \tag{8.5}$$

Root Mean Square Error (RMSE) izračunat je koristeći formulu 8.6:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(fit(MOS_1(i)) - MOS_2(i))^2} \tag{8.6}$$

Gdje ***N*** predstavlja broj testiranih video sekvenca, što u našem slučaju ispada 96. $\boldsymbol{Fit(MOS_1(i))}$ je prilagođen MOS rezultat ***i***-te video sekvence u prvom laboratoriju, dok je $\boldsymbol{MOS_2(i)}$ rezultat sa drugog laboratorija iste video sekvence ***i*** [**15**].

Outlier Ratio (OR) izračunat je kao broj video sekvenci koje ne pripadaju u 95% interval pouzdanosti (*engl.* „confidence interval“, CI) sa MOS laboratorijskih testova, gdje je CI računat pomoću formule 8.7:

$$CI = t(M-1)\frac{std(scores)}{\sqrt{M}} \tag{8.7}$$

U ovoj formuli, t(M) predstavlja kritičnu vrijednost studentove distribucije sa M-1 slobodnih parametara (M predstavlja koliko puta je ista video sekvenca bila ocjenjena) sa 95% vjerojatnosti, dok je std(scores) standardna devijacija svih ocjena iste video sekvence.

Dobiveni rezultat smatrao se netipičnom vrijednosti (*engl.* „outlier“) ako vrijedi formula 8.8:

$$|fit(MOS_1(i)) - MOS_2(i)| > \frac{1}{2}(CI_1(i) + CI_2(i)) \tag{8.8}$$

Nakon toga, OR je izračunat kao broj netipičnih vrijednosti podijeljen sa brojem testiranih video sekvenci (96 u našem slučaju).

Usporedbe između objektivnih i subjektivnih mjera vršene su uz pomoć programa „Project1.exe“ dobivenog od strane profesora Emila Dumić koji isčitava i uspoređuje po dva modela, jedan originalan referentni model te jedan izobličeni model. Project1 je konzolna aplikacija, što znači da će se izvršavati preko Command Prompta. Jednom kada se je koristik pozicionirao u zadanu korisničku mapu, program je pokrenut pomoću posebnih komandi ovisno o korisnikovoj namjeri (slika 8.7).

```
Usage: Project1.exe [options]
Allowed options:
  -h [ --help ]                    produce help message
  -a [ --fileA ] arg               Input file 1
  -b [ --fileB ] arg               Input file 2
  -A [ --saveNorm ] arg            File name to output the normals of original
                                   point cloud
  -k [ --knn ] arg (=12)           Set KNN number of neighbor points for normal
                                   estimation. Default normal estimation method
  -t [ --rtimes ] arg (=0)         Guide to set radius for normal estimation,
                                   times to NN points distances
  -f [ --forceNormalEstimation ]   Force normal estimation even normals are
                                   provided
  -s [ --singlePass ]              Force running a single pass, where the loop
                                   is over the original point cloud
  -d [ --hausdorff ]               Send the Haursdorff metric as well
  -1 [ --point2point ]             Force to report point-to-point metric only
```

Slika 8.7: Moguće komande unutar programa Project1.exe

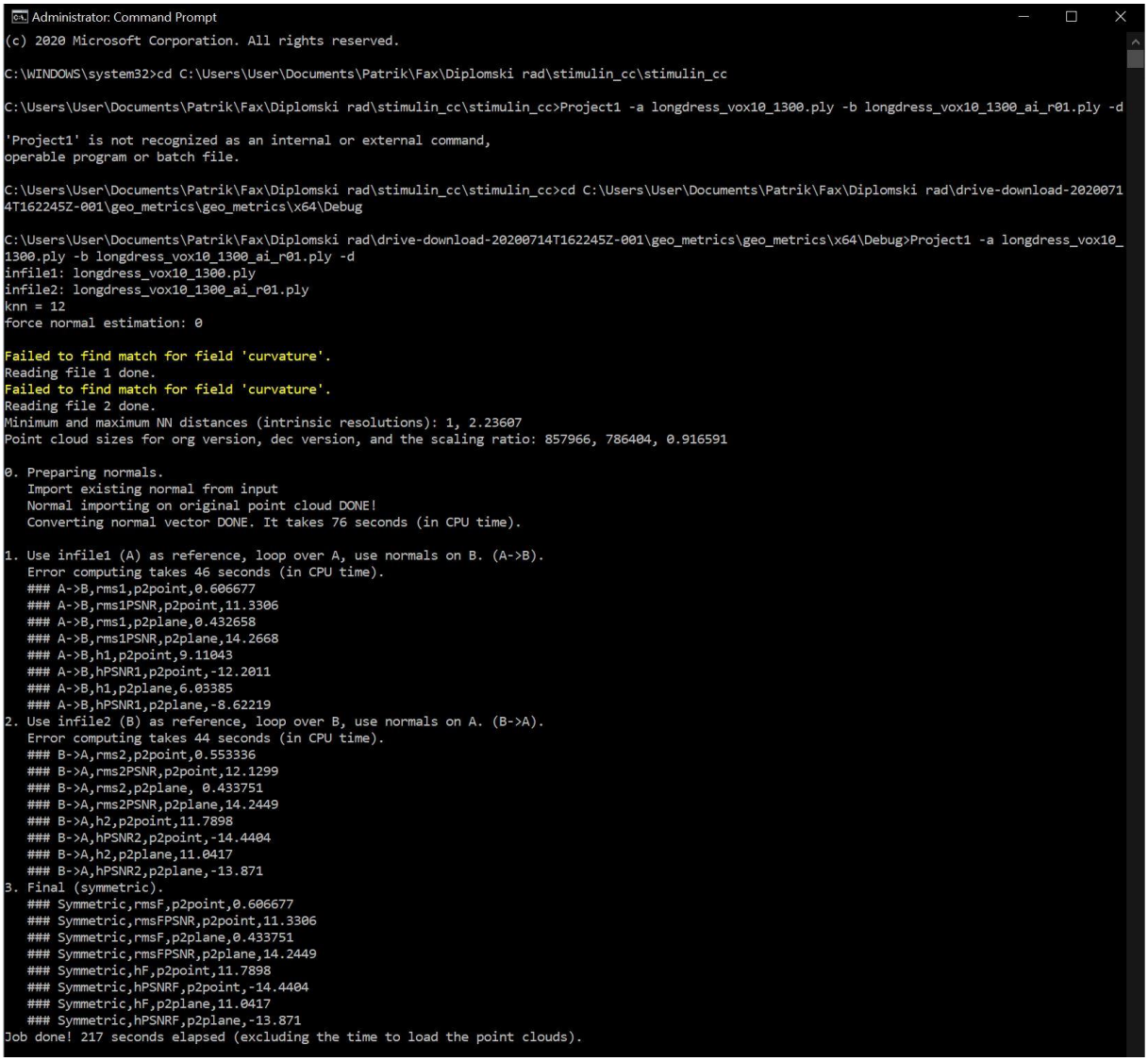
U ovom su radu korištene komande -a, -b i -d. Komanda -a predstavlja prvi model u usporedbi, koji je uvijek referentni i originalni, dok komanda -b predstavlja jednu od petnaest izobličenih inačica tog istog modela. Zadnja komanda, komanda -d, predstavlja izračunavanje Haussdorf udaljenosti. Kada spojimo sve komande, komandna linija koju moramo upisati izgleda kao u primjeru 8.9.

```
Project1 -a longdress_vox10_1300.ply -b longdress_vox10_1300_ai_r01.ply -d
```

(8.9)

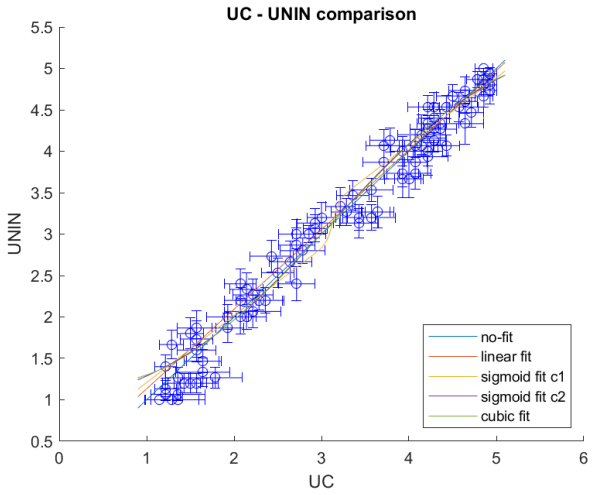
Nakon pokretanja, program počinje učitavati upisane oblake točaka i njihove normale kako bi ih pripremio za usporedbu. Normale zadanih oblaka točaka mogu također biti izračunate pomoću ovog programa, ali su u ovom primjeru korištene već izračunate normale. Koeficijenti koje ovaj program izračunava su: RMS p2point, RMS PSNR p2point, RMS p2plane, RMS PSNR p2plane, Hausdorff p2point, Hausdorff PSNR p2point, Hausdorff p2plane i Hausdorff PSNR p2plane.

Program se ponavlja tako dugo dok svi modeli nisu uspoređeni te također ispisuje rezultate (slika 8.8) u tablice kreirane pomoću skripte. Svi su rezultati tada izračunati kroz više različitih funkcija prilagodbe.



Slika 8.8: Rezultati prvog učitavanja i uspoređivanja koristeći program "Project1.exe"

Usporedbe subjektivnih rezultata između UNIN i UC:

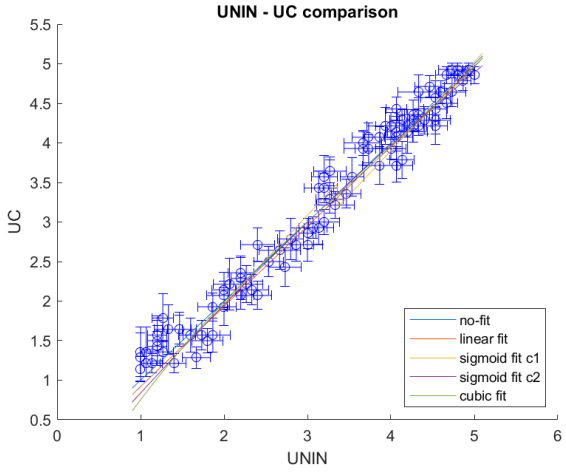


Tablica 2: Rezultati usporedbe subjektivnih rezultata UC-UNIN kroz više funkcija prilagodbe

Funkcija prilagodbe	PCC	RMSE	OR	SROCC	KROCC
C1	0,989	0,183	0,093	0,982	0,899
C2	0,988	0,188	0,041	0,982	0,899
C3	0,988	0,188	0,072	0,982	0,899
C4	0,986	0,205	0,062	0,982	0,899
Bez prilagodbe	0,986	0,218	0,072	0,982	0,899

Tablica 3: Funkcije prilagodbe i vrijednosti koeficijenata b

C1	C2
Generalni model: c1(x) = b1*(1/2-1/(1+exp(b2*(x-b3))))+b4*x+b5 **Koeficijenti (95% interval pouzdanosti):** ***b1*** = 0.4421 (0.2341, 0.65) ***b2*** = 28.75 (-29.52, 87.02) ***b3*** = 3.133 (3.058, 3.207) ***b4*** = 0.8097 (0.7332, 0.8863) ***b5*** = 0.6193 (0.3868, 0.8518)	**Generalni model:** c2(x) = (b1-b2)/(1+exp((x-b3)/b4))+b2 **Koeficijenti (95% interval pouzdanosti):** ***b1*** = 0.7021 (0.207, 1.197) ***b2*** = 5.574 (4.92, 6.227) ***b3*** = 3.119 (2.915, 3.324) ***b4*** = 1.065 (0.7353, 1.394)
C3	**C4**
Generalni model: c3(x) = b1*x^3+b2*x^2+b3*x+b4 **Koeficijenti (95% interval pouzdanosti):** ***b1*** = -0.05505 (-0.08488, -0.02523) ***b2*** = 0.516 (0.2512, 0.7809) ***b3*** = -0.4969 (-1.214, 0.2198) ***b4*** = 1.336 (0.7691, 1.903)	**Generalni model:** c4(x) = b1*x+b2 **Koeficijenti (95% interval pouzdanosti):** ***b1*** = 0.9608 (0.928, 0.9936) ***b2*** = 0.1721 (0.06485, 0.2793)



Tablica 4: Rezultati usporedbe subjektivnih rezultata UNIN-UC kroz više funkcija prilagodbe

Funkcija prilagodbe	PCC	RMSE	OR	SROCC	KROCC
C1	0,987	0,200	0,104	0,982	0,899
C2	0,986	0,207	0,083	0,982	0,899
C3	0,987	0,206	0,093	0,982	0,899
C4	0,986	0,211	0,083	0,982	0,899
Bez prilagodbe	0,986	0,218	0,072	0,982	0,899

Tablica 5: Funkcije prilagodbe i vrijednosti koeficijenata b

C1	C2
Generalni model: c1(x) = b1*(1/2-1/(1+exp(b2*(x-b3))))+b4*x+b5 **Koeficijenti (95% interval pouzdanosti):** ***b1*** = -0.3073 (-0.5069, -0.1076) ***b2*** = 512.8 (-8.937e+18, 8.937e+18) ***b3*** = 3.362 (-9.105e+13, 9.105e+13) ***b4*** = 1.123 (1.044, 1.201) ***b5*** = -0.4361 (-0.6824, -0.1897)	**Generalni model:** c2(x) = (b1-b2)/(1+exp((x-b3)/b4))+b2 **Koeficijenti (95% interval pouzdanosti):** ***b1*** = -107.1 (-1.628e+04, 1.607e+04) ***b2*** = 16.48 (-225.3, 258.2) ***b3*** = -21.94 (-2353, 2309) ***b4*** = 11.88 (-368.9, 392.7)c3
C3	**C4**
Generalni model: c3(x) = b1*x^3+b2*x^2+b3*x+b4 **Koeficijenti (95% interval pouzdanosti):** ***b1*** = 0.02387 (-0.0137, 0.06144) ***b2*** = -0.2514 (-0.5982, 0.09538) ***b3*** = 1.82 (0.8286, 2.812) ***b4*** = -0.8398 (-1.691, 0.01112)	**Generalni model:** c4(x) = b1*x+b2 **Koeficijenti (95% interval pouzdanosti):** ***b1*** = 1.013 (0.9781, 1.047) ***b2*** = -0.09303 (-0.2074, 0.02129)

Rezultati objektivnih mjera mogli bi se podijeliti u dvije skupine: point-to-point (p2point) i point-to-plane (p2plane). Prva je skupina bazirana na geometrijskoj udaljenosti između referentne i izobličene točke. Druga je skupina bazirana na pogrešci dobivenoj nakon projekcije referentne točke na vektor normale izobličene točke. Način računanja sveukupnih pogrešaka za obje mjere je identičan, što znači da za svaku točku izobličenog oblaka, potrebno je identificirati najbližu susjednu točku u referentnom oblaku. Svaka individualna greška tada je posebno izračunata ovisno o odabranoj mjeri.

Nakon toga, dodani su im ili Mean-Squared Error (MSE) ili Hausdorf udaljenost vrijednosti kako bi dobili ukupnu vrijednosti degradacije. S obzirom da rezultat ovisi o tome koji oblak točka uzmemo kao referentni, veća vrijednost odabrana je kao simetrični MSE. Vrijednosti PSNR također su izračunate na kraju.

MSE na YCbCr reprezentaciji korišteni je kako bi se izračunale degradacije u boji, nakon konverzije RGB-a u YCbCr prema standardu ITU-R Recommentation BT.709-3. Vektori normale izračunati su za svaki oblak točaka pomoću kvadratne funkcije prilagodbe na susjednim točkama odabrane lokacije, slično kao i način rada CloudCompare programa.

Za usporedbu objektivnih i subjektivnih rezultata, predloženi su standardi iz Recommendation ITU-T P.1401, što znači da su PCC, SROCC, RMSE i OR podijeljeni u parove sa MOS i predviđenim MOS-om kako bi se izračunala korelacija svake objektivne mjere sa subjektivnim MOS ocjenama. Predviđeni MOS dobiven je nakon korištenja 3 funkcije prilagodbe, navedenih kasnije, na objektivne rezultate.

1.) Unin

Tablica 6: Rezultati usporedbe objektivnih mjera sa subjektivnim mjerama za UNIN

Mjera	PCC1	PCC2	PCC3	SROCC	KROCC	RMSE	OR
rmsF_p2poit	0,880	0,876	0,844	0,821	0,620	0,568	0,142
rmsFPSNR_p2point	0,654	0,618	0,612	0,588	0,429	0,905	0,285
rmsF_p2plane	0,942	0,939	0,922	0,919	0,745	0,401	0,071
rmsFPSNR_p2plane	0,675	0,642	0,635	0,619	0,471	0,883	0,261
hF_p2point	0,572	0,550	0,416	0,418	0,305	0,982	0,333
hPSNRF_p2point	0,507	0,478	0,482	0,475	0,332	1,032	0,333
hF_p2plane	0,592	0,550	0,526	0,537	0,395	0,965	0,297
hPSNRF_p2plane	0,511	0,498	0,479	0,472	0,339	1,029	0,345

Jednadžba za PCC1_c1:

data(x) = b1*(1/2-1/(1+exp(b2*(x-b3))))+b4*x+b5

Jednadžba za PCC2_c2:

data(x) = (b1-b2)/(1+exp((x-b3)/b4))+b2

Jednadžba za PCC3_c3:

data(x) = b1*x^3+b2*x^2+b3*x+b4

Tablica 7: Vrijednosti koeficijenata b funkcije prilagodbe PCC1

Mjera	PCC1_c1
rmsF_p2point	**b1** = 2.345 (1.688, 3.003) **b2** = -13.33 (-22.36, -4.313) **b3** = 0.6697 (0.6207, 0.7187) **b4** = -0.1831 (-0.4141, 0.04785) **b5** = 2.877 (2.55, 3.204)
rmsFPSNR_p2point	**b1** = -1.107 (-1.864, -0.3498) **b2** = -62.52 (-8.4e+09, 8.4e+09) **b3** = 6.11 (-9.722e+07, 9.722e+07) **b4** = 0.04984 (0.01913, 0.08055) **b5** = 1.749 (1.375, 2.122)
rmsF_p2plane	**b1** = -2.817 (-3.46, -2.175) **b2** = 12.4 (7.268, 17.53) **b3** = 0.3997 (0.3621, 0.4373) **b4** = -0.2652 (-0.5464, 0.01594) **b5** = 3.023 (2.756, 3.29)
rmsFPSNR_p2plane	**b1** = 1.031 **b2** = 98.32 **b3** = 12.24 **b4** = 0.04728 (0.01673, 0.07783) **b5** = 1.677 (1.195, 2.159)
hF_p2point	**b1** = -1.965 (-7.944, 4.013) **b2** = 8.452 (-41.12, 58.02) **b3** = 1.106 (0.2579, 1.953) **b4** = -0.01682 (-0.03661, 0.002977) **b5** = 3.686 (0.7093, 6.662)
hPSNRF_p2point	**b1** = -0.9125 (-1.965, 0.1399) **b2** = 84.17 (-4.418e+08, 4.418e+08) **b3** = 6.325 (-1.078e+06, 1.078e+06) **b4** = 0.0749 (0.04, 0.1098) **b5** = 2.926 (2.604, 3.247)
hF_p2plane	**b1** = 1.571 (-1.322, 4.464) **b2** = -10.42 (-58.1, 37.25) **b3** = 1.106 (0.6247, 1.587) **b4** = -0.05239 (-0.0954, -0.009371) **b5** = 3.733 (2.294, 5.173)
hPSNRF_p2plane	**b1** = -0.8186 (-1.701, 0.06421) **b2** = -654.5 (-3.983e+19, 3.983e+19) **b3** = 0.9686 (-6.455e+13, 6.455e+13) **b4** = 0.02245 (-0.01607, 0.06097) **b5** = 2.975 (2.731, 3.219)

Tablica 8: Vrijednosti koeficijenata b funkcije prilagodbe PCC2

Mjera	PCC2_c2
rmsF_p2point	**b1** = 4.029 (3.664, 4.393) **b2** = 1.293 (1.037, 1.549) **b3** = 0.6862 (0.6383, 0.7342) **b4** = 0.09009 (0.04233, 0.1378)
rmsFPSNR_p2point	**b1** = 0.808 (-0.4131, 2.029) **b2** = 3.526 (3.059, 3.992) **b3** = 6.675 (1.587, 11.76) **b4** = 4.003 (-0.00694, 8.013)
rmsF_p2plane	**b1** = 4.477 (3.98, 4.974) **b2** = 1.25 (1.057, 1.442) **b3** = 0.4084 (0.3681, 0.4487) **b4** = 0.09376 (0.05902, 0.1285)
rmsFPSNR_p2plane	**b1** = 0.9619 (0.06089, 1.863) **b2** = 3.438 (3.065, 3.812) **b3** = 11.2 (7.694, 14.71) **b4** = 2.94 (0.2729, 5.607)
hF_p2point	**b1** = 4.602 (-0.8134, 10.02) **b2** = 2.455 (2.204, 2.706) **b3** = 1.14 (0.4849, 1.796) **b4** = 0.1198 (-0.6322, 0.8719)
hPSNRF_p2point	**b1** = -78.45 (-3.483e+04, 3.468e+04) **b2** = 4.987 (-16.56, 26.54) **b3** = -146 (-1.919e+04, 1.889e+04) **b4** = 38.47 (-782.1, 859)
hF_p2plane	**b1** = 4.446 (1.535, 7.356) **b2** = 2.455 (2.204, 2.706) **b3** = 1.159 (0.8352, 1.483) **b4** = 0.1029 (-0.4571, 0.663)
hPSNRF_p2plane	**b1** = 2.333 (2.042, 2.624) **b2** = 3.582 (3.188, 3.977) **b3** = 0.9687 (-2.578e+13, 2.578e+13) **b4** = 0.001769 (-4.461e+13, 4.461e+13)

Tablica 9: Vrijednosti koeficijenata b funkcije prilagodbe PCC3

Mjera	PCC3_c3
rmsF_p2point	**b1** = -0.3172 (-0.5402, -0.09417) **b2** = 2.423 (1.103, 3.743) **b3** = -5.851 (-7.863, -3.839) **b4** = 5.762 (5.01, 6.515)
rmsFPSNR_p2point	**b1** = 4.538e-05 (-0.0001563, 0.000247) **b2** = -0.0029 (-0.01041, 0.004611) **b3** = 0.1237 (0.05246, 0.195) **b4** = 1.623 (1.15, 2.096)
rmsF_p2plane	**b1** = -1.249 (-1.888, -0.6112) **b2** = 6.189 (3.906, 8.473) **b3** = -9.68 (-11.75, -7.613) **b4** = 5.879 (5.407, 6.351)
rmsFPSNR_p2plane	**b1** = 5.972e-05 (-0.0001191, 0.0002386) **b2** = -0.004117 (-0.01319, 0.004956) **b3** = 0.1576 (0.03052, 0.2846) **b4** = 1.029 (0.3859, 1.671)
hF_p2point	**b1** = -3.53e-05 (-8.766e-05, 1.707e-05) **b2** = 0.003573 (-0.001204, 0.00835) **b3** = -0.1191 (-0.2225, -0.0157) **b4** = 3.541 (3.028, 4.054)

hPSNRF_p2point	**b1** = 4.173e-05 (-6.935e-05, 0.0001528) **b2** = -0.0004165 (-0.00196, 0.001127) **b3** = 0.03498 (-0.007171, 0.07713) **b4** = 3.101 (2.741, 3.462)
hF_p2plane	**b1** = -0.0001893 (-0.0005906, 0.0002119) **b2** = 0.01186 (-0.0062, 0.02993) **b3** = -0.2636 (-0.4693, -0.05786) **b4** = 3.967 (3.38, 4.553)
hPSNRF_p2plane	**b1** = 1.369e-05 (-0.0001197, 0.0001471) **b2** = 2.395e-05 (-0.001676, 0.001723) **b3** = 0.04834 (0.003136, 0.09353) **b4** = 2.956 (2.632, 3.28)

2.) UC

Tablica 10: Rezultati usporedbe objektivnih mjera sa subjektivnim mjerama za UC

Mjera	PCC1	PCC2	PCC3	SROCC	KROCC	RMSE	OR
rmsF_p2point	0,870	0,869	0,840	0,820	0,624	0,568	0,023
rmsFPSNR_p2point	0,604	0,572	0,559	0,552	0,393	0,919	0,214
rmsF_p2plane	0,942	0,941	0,921	0,917	0,737	0,385	0
rmsFPSNR_p2plane	0,630	0,601	0,583	0,575	0,428	0,895	0,214
hF_p2point	0,603	0,572	0,460	0,453	0,326	0,920	0,202
hPSNRF_p2point	0,514	0,484	0,491	0,449	0,321	0,989	0,178
hF_p2plane	0,621	0,572	0,557	0,539	0,389	0,904	0,190
hPSNRF_p2plane	0,490	0,469	0,471	0,431	0,315	1,006	0,214

Jednadžba za PCC1_c1:

$$data(x) = b1*(1/2-1/(1+\exp(b2*(x-b3))))+b4*x+b5$$

Jednadžba za PCC2_c2:

$$data(x) = (b1-b2)/(1+\exp((x-b3)/b4))+b2$$

Jednadžba za PCC3_c3:

$$data(x) = b1*x^3+b2*x^2+b3*x+b4$$

Tablica 11: Vrijednosti koeficijenata b funkcije prilagodbe PCC1

Mjera	PCC1_c1
rmsF_p2point	**b1** = 2.447 (1.749, 3.146) **b2** = -11.62 (-19.05, -4.187) **b3** = 0.6732 (0.619, 0.7274) **b4** = -0.09331 (-0.3296, 0.1429) **b5** = 2.844 (2.504, 3.184)
rmsFPSNR_p2point	**b1** = -1.049 (-1.795, -0.3026) **b2** = -70.54 (-4.169e+24, 4.169e+24) **b3** = 6.198 (-1.704e+22, 1.704e+22) **b4** = 0.04181 (0.01077, 0.07286) **b5** = 1.925 (1.547, 2.302)

rmsF_p2plane	**b1** = -2.863 (-3.482, -2.244) **b2** = 12.63 (7.767, 17.49) **b3** = 0.3899 (0.3545, 0.4253) **b4** = -0.1461 (-0.4137, 0.1214) **b5** = 3.042 (2.785, 3.299)
rmsFPSNR_p2plane	**b1** = 1.012 (0.334, 1.69) **b2** = 105.8 (-1.304e+19, 1.304e+19) **b3** = 12.48 (-4.734e+15, 4.734e+15) **b4** = 0.03865 (0.007341, 0.06996) **b5** = 1.884 (1.39, 2.379)
hF_p2point	**b1** = -1.728 (-5.518, 2.062) **b2** = 9.606 (-52.07, 71.29) **b3** = 1.144 (0.6339, 1.654) **b4** = -0.02 (-0.03854, -0.001459) **b5** = 3.659 (1.773, 5.546)
hPSNRF_p2point	**b1** = -1.033 **b2** = 40.96 **b3** = 10.37 **b4** = 0.06722 (0.04001, 0.09443) **b5** = 2.796 (2.391, 3.2)
hF_p2plane	**b1** = 1.475 (-1.022, 3.973) **b2** = -10.56 (-64.98, 43.87) **b3** = 1.128 (0.7082, 1.548) **b4** = -0.0557 (-0.09603, -0.01538) **b5** = 3.762 (2.518, 5.006)
hPSNRF_p2plane	**b1** = 0.768 **b2** = -366.4 **b3** = 12.48 **b4** = 0.06275 (0.03529, 0.0902) **b5** = 2.738 (2.313, 3.163)

Tablica 12: Vrijednosti koeficijenata b funkcije prilagodbe PCC2

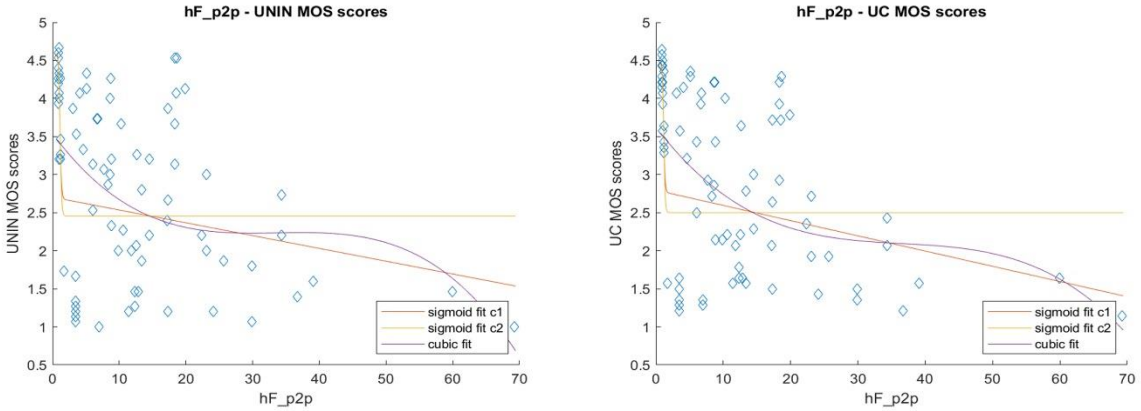
Mjera	PCC2_c2
rmsF_p2point	**b1** = 4.029 (3.664, 4.393) **b2** = 1.293 (1.037, 1.549) **b3** = 0.6862 (0.6383, 0.7342) **b4** = 0.09009 (0.04233, 0.1378)
rmsFPSNR_p2point	**b1** = 1.203 (0.2269, 2.18) **b2** = 3.453 (3.032, 3.873) **b3** = 7.233 (2.753, 11.71) **b4** = 3.3 (-0.257, 6.858)
rmsF_p2plane	**b1** = 4.502 (4.049, 4.954) **b2** = 1.411 (1.235, 1.588) **b3** = 0.3947 (0.3581, 0.4312) **b4** = 0.08667 (0.05665, 0.1167)
rmsFPSNR_p2plane	**b1** = 1.33 (0.5702, 2.09) **b2** = 3.396 (3.049, 3.744) **b3** = 11.8 (8.602, 15) **b4** = 2.343 (-0.1519, 4.837)
hF_p2point	**b1** = 4.465 (1.028, 7.902) **b2** = 2.5 (2.262, 2.738) **b3** = 1.178 (0.8672, 1.488) **b4** = 0.1028 (-0.6248, 0.8305)
hPSNRF_p2point	**b1** = -125.1 (-8.378e+04, 8.353e+04) **b2** = 4.867 (-12.95, 22.68) **b3** = -159.4 (-2.68e+04, 2.648e+04) **b4** = 36.73 (-674.7, 748.2)

hF_p2plane	**b1** = 4.412 (2.065, 6.759) **b2** = 2.5 (2.262, 2.738) **b3** = 1.182 (0.9513, 1.413) **b4** = 0.09603 (-0.5003, 0.6924)
hPSNRF_p2plane	**b1** = 2.426 (2.14, 2.712) **b2** = 3.562 (3.18, 3.944) **b3** = 0.9701 (-2.94e+14, 2.94e+14) **b4** = 0.001811 (-1.272e+15, 1.272e+15)

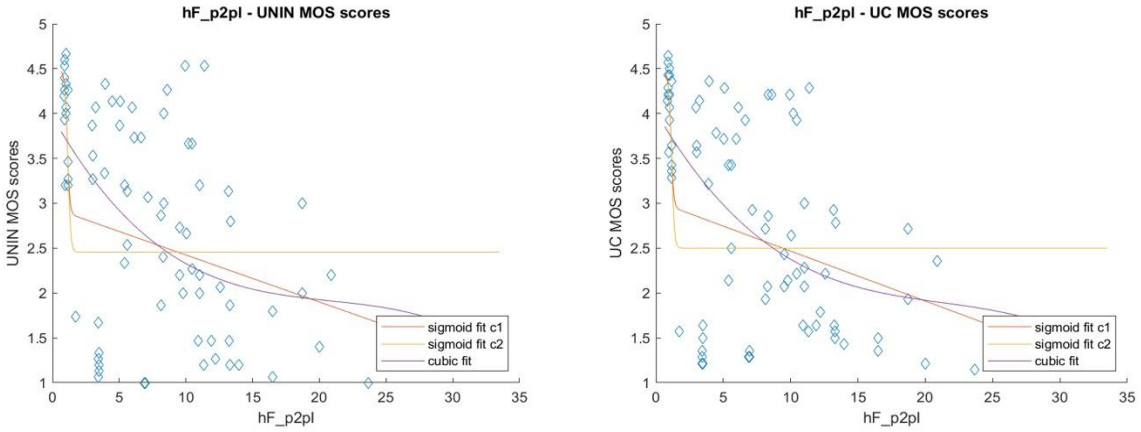
Tablica 13: Vrijednosti koeficijenata b funkcije prilagodbe PCC3

Mjera	PCC3_c3
rmsF_p2point	**b1** = -0.3171 (-0.535, -0.09915) **b2** = 2.44 (1.151, 3.73) **b3** = -5.855 (-7.82, -3.889) **b4** = 5.783 (5.048, 6.517)
rmsFPSNR_p2point	**b1** = 9.09e-06 (-0.0001948, 0.0002129) **b2** = -0.001518 (-0.009111, 0.006075) **b3** = 0.1046 (0.03259, 0.1767) **b4** = 1.774 (1.295, 2.252)
rmsF_p2plane	**b1** = -1.331 (-1.949, -0.7127) **b2** = 6.518 (4.307, 8.73) **b3** = -9.901 (-11.9, -7.899) **b4** = 5.936 (5.479, 6.393)
rmsFPSNR_p2plane	**b1** = 3.136e-05 (-0.0001501, 0.0002128) **b2** = -0.002617 (-0.01182, 0.006585) **b3** = 0.1283 (-0.0005773, 0.2572) **b4** = 1.285 (0.6329, 1.937)
hF_p2point	**b1** = -2.776e-05 (-7.705e-05, 2.152e-05) **b2** = 0.003046 (-0.001449, 0.007542) **b3** = -0.1162 (-0.2135, -0.01884) **b4** = 3.625 (3.142, 4.108)
hPSNRF_p2point	**b1** = 5.074e-05 (-5.573e-05, 0.0001572) **b2** = -0.0003805 (-0.00186, 0.001099) **b3** = 0.03085 (-0.009558, 0.07125) **b4** = 3.132 (2.786, 3.478)
hF_p2plane	**b1** = -0.0001745 (-0.0005522, 0.0002032) **b2** = 0.01116 (-0.005847, 0.02816) **b3** = -0.2583 (-0.452, -0.06468) **b4** = 4.019 (3.467, 4.571)
hPSNRF_p2plane	**b1** = 3.871e-05 (-9.054e-05, 0.000168) **b2** = -3.702e-05 (-0.001683, 0.001609) **b3** = 0.03783 (-0.005948, 0.08162) **b4** = 2.991 (2.678, 3.305)

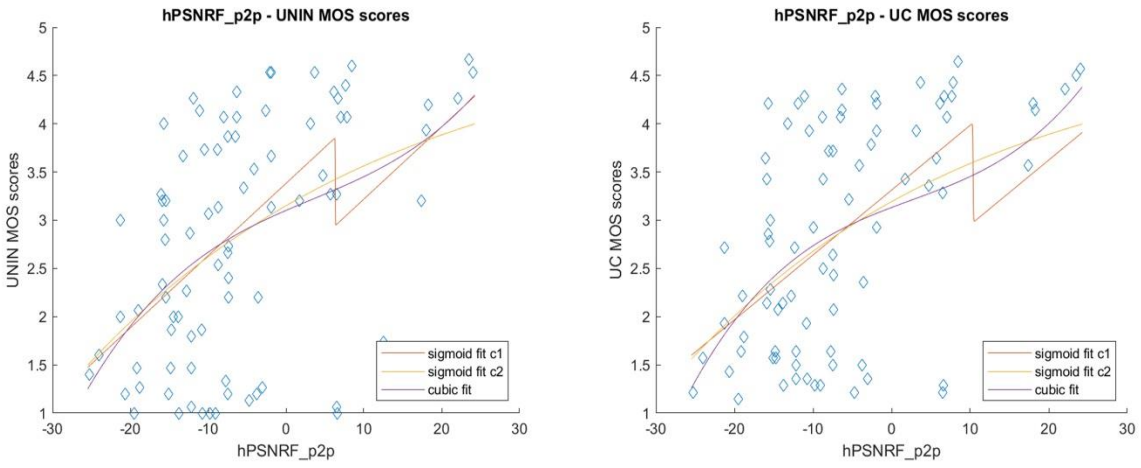
Rezultati objektivnih mjera po kategorijama (UNIN i UC):



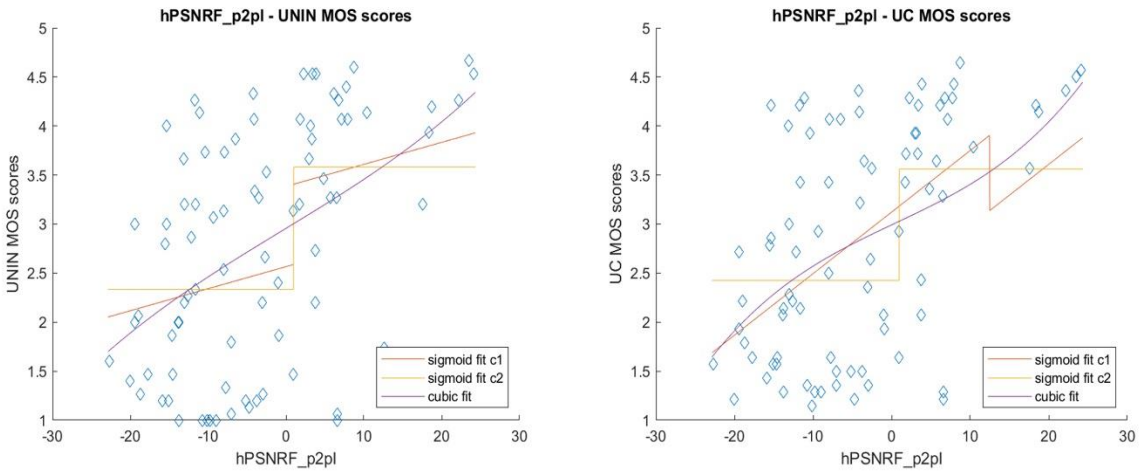
Slika 8.9: Hausdorff p2point rezultati (UNIN-lijevo, UC-desno)



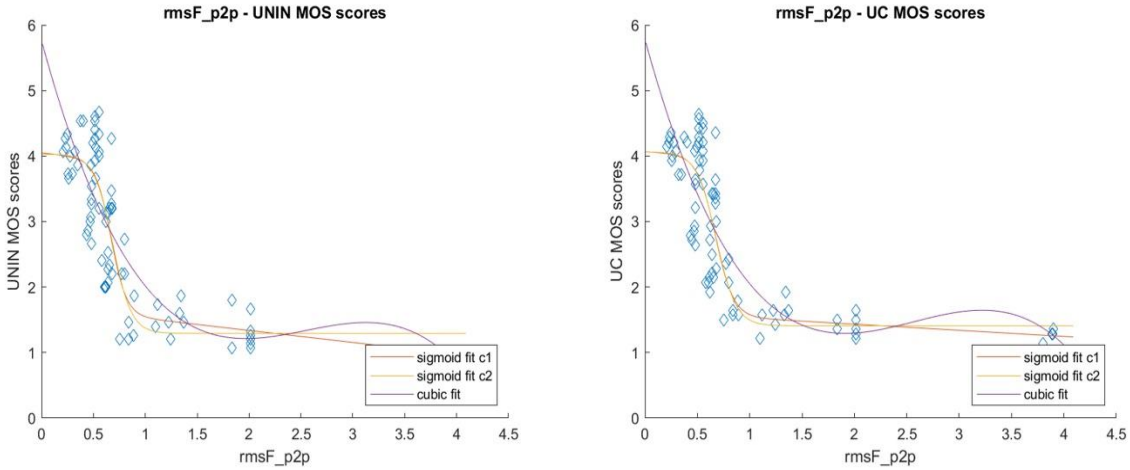
Slika 8.10: Hausdorff p2plane rezultati (UNIN-lijevo, UC-desno)



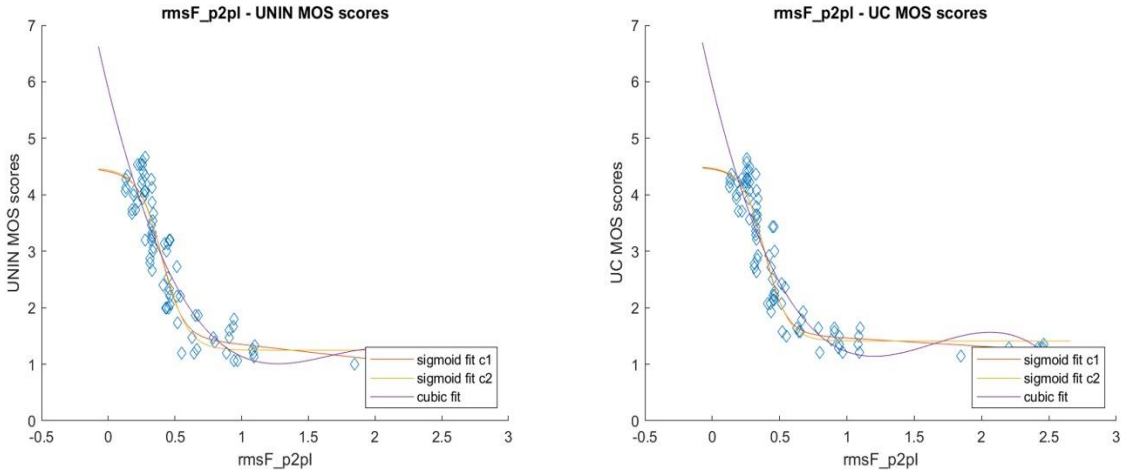
Slika 8.11: Hausdorff PSNR p2point rezultati (UNIN-lijevo, UC-desno)



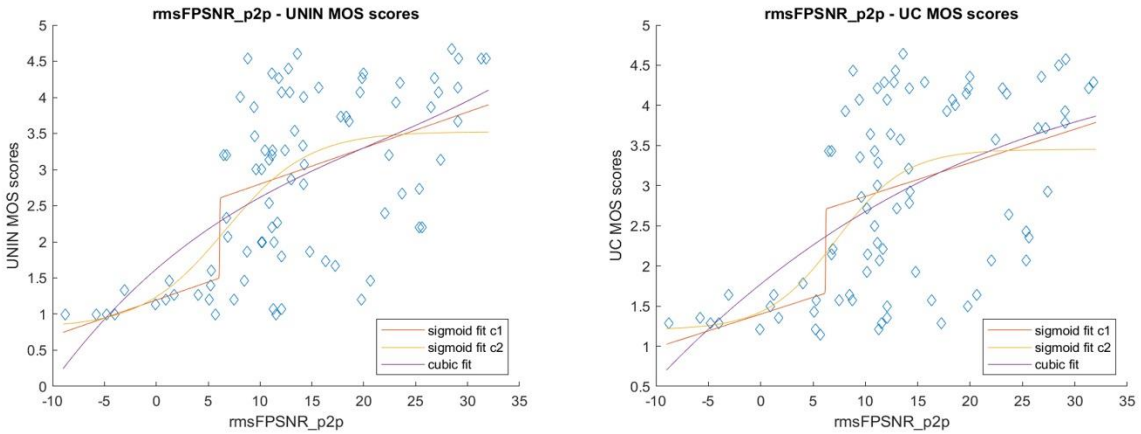
Slika 8.12: Hausdorff PSNR p2plane rezultati (UNIN-lijevo, UC-desno)



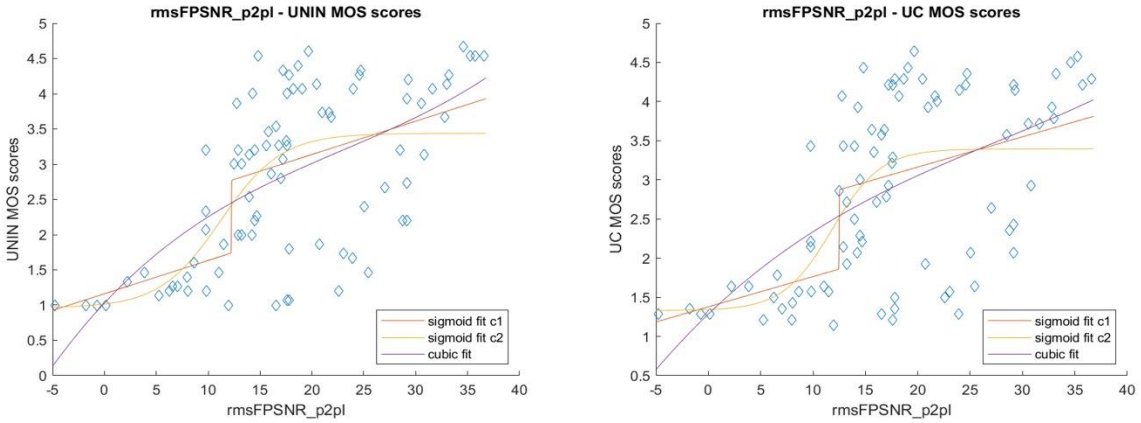
Slika 8.13: RMSE p2point rezultati (UNIN-lijevo, UC-desno)



Slika 8.14: RMSE p2plane rezultati (UNIN-lijevo, UC-desno)



Slika 8.15: RMSE PSNR p2point rezultati (UNIN-lijevo, UC-desno)



Slika 8.16: RMSE PSNR p2plane rezultati (UNIN-lijevo, UC-desno)

Kao što vidimo, najbolji rezultati dobiveni su uz pomoć rmsF_p2plane i rmsF_p2point. Gledajući ove rezultate, možemo zaključiti kako je kompresija MPEG V-PCC-a izuzetno efikasna kada se koristi kod kompresije statičkih oblaka točaka. Sve metode korištene u ovim eksperimentima uspjele su konzistentno producirati solidne rezultate sa svakim ponavljanjem kroz dva laboratorija.

9. Zaključak

U uvodnom djelu rada objašnjeni su oblaci točaka i njihova raznolika upotreba. Ovo područje je nažalost još uvijek slabije poznato i iskorišteno, iako se vidi veliki pomak u razdoblju od proteklih tridesetak godina. U ovom, ali i drugim područjima koji spadaju u sferu računalne grafike, prioritet je oduvijek bio pronalaženje efikasnih metoda kompresije.

JPEG Pleno organizacija postavila je važne temelje za izradu standarda i metoda vezanih za uspješnu kompresiju različitih kompleksnih vrsta podataka. Jedan od takvih vrsta bili su oblaci točaka. Kompresija oblaka ovisi o više faktora kod zadanih oblaka točaka, kao što su npr. statički ili dinamični oblaci, manji ili veći bpp, sama veličina oblaka točaka, itd.

MPEG grupa izdala je već mnogo različitih vrsta kompresije, od kojih su najpoznatiji i najkorišteniji MPEG G-PCC i MPEG V-PCC. MPEG G-PCC korišten je za kompresiju oblaka točaka baziranu na geometrijskim izračunima dubine i površinskoj rekonstrukciji, dok je MPEG V-PCC korišten kod dinamičnih oblaka točaka generirajući više slojeva za atribute u svakoj slici. Kompresija pomoću panorama koristi informacije dobivene najčešće putem korištenja LiDAR tehnologije te se bazira na projiciranju takvih informacija unutar sfere. Korištenje projekcija kao metoda kompresije dinamičnih oblaka točaka je sama po sebi novija metoda i ista zahtjeva još dodatna testiranja.

Kako bi se uspješnost kompresija mogla ocjeniti, koristimo takozvane objektivne i subjektivne metode za kvalitetu oblaka točaka. Objektive mjere pomažu nam procjeniti preciznost i točnost izobličenog oblaka točaka uspoređujući ga sa referentnim oblakom koristeći mjere poput: p2point, p2plane, MSE, PSNR i Hausdorff koeficijent. Subjektivne metode s druge strane opisuju vrste metoda korištene za testiranja između više oblaka točaka poput:SS, DSCQS, SC, SSCQE i DSIS.

Za usporedbe dobivenih rezultata korištene su metode za provjeravanje preciznosti kompresije: PCC, RMSE, OR, SROCC, KROCC kao i usporedbe MSE sa YCbCr bojama.

Dobiveni rezultati pokazali su kako je većina metoda izuzetno efikasna i uspješna u kompresiji oblaka točaka, no neprestano proučavanje postupaka i pronalaženje novih metoda, kao i stalni napredak tehnologije uvijek će biti glavni pokretači kreiranja još boljih metoda.

U Varaždinu, _____ 2020. godine _____

potpis studenta

Sveučilište
Sjever

SVEUČILIŠTE
SJEVER

IZJAVA O AUTORSTVU
I
SUGLASNOST ZA JAVNU OBJAVU

Završni/diplomski rad isključivo je autorsko djelo studenta koji je isti izradio te student odgovara za istinitost, izvornost i ispravnost teksta rada. U radu se ne smiju koristiti dijelovi tuđih radova (knjiga, članaka, doktorskih disertacija, magistarskih radova, izvora s interneta, i drugih izvora) bez navođenja izvora i autora navedenih radova. Svi dijelovi tuđih radova moraju biti pravilno navedeni i citirani. Dijelovi tuđih radova koji nisu pravilno citirani, smatraju se plagijatom, odnosno nezakonitim prisvajanjem tuđeg znanstvenog ili stručnoga rada. Sukladno navedenom studenti su dužni potpisati izjavu o autorstvu rada.

Ja, PATRIK LUKMAN *(ime i prezime)* pod punom moralnom, materijalnom i kaznenom odgovornošću, izjavljujem da sam isključivi autor/~~ica~~ ~~završnog~~/diplomskog *(obrisati nepotrebno)* rada pod naslovom SUBJEKTIVNO ISPITIVANJE KVALITETE OBLAKA TOČAKA I USPOREDBA S OBJEKTIVNIH MJERAMA *(upisati naslov)* te da u navedenom radu nisu na nedozvoljeni način (bez pravilnog citiranja) korišteni dijelovi tuđih radova.

Student/ica:
(upisati ime i prezime)
Patrik Lukman
(vlastoručni potpis)

Sukladno Zakonu o znanstvenoj djelatnost i visokom obrazovanju završne/diplomske radove sveučilišta su dužna trajno objaviti na javnoj internetskoj bazi sveučilišne knjižnice u sastavu sveučilišta te kopirati u javnu internetsku bazu završnih/diplomskih radova Nacionalne i sveučilišne knjižnice. Završni radovi istovrsnih umjetničkih studija koji se realiziraju kroz umjetnička ostvarenja objavljuju se na odgovarajući način.

Ja, PATRIK LUKMAN *(ime i prezime)* neopozivo izjavljujem da sam suglasan/~~na~~ s javnom objavom ~~završnog~~/diplomskog *(obrisati nepotrebno)* rada pod naslovom SUBJEKTIVNO ISPITIVANJE KVALITETE OBLAKA TOČAKA I USPOREDBA S OBJEKTIVNIH MJERAMA *(upisati naslov)* čiji sam autor/~~ica~~.

Student/ica:
(upisati ime i prezime)
Patrik Lukman
(vlastoručni potpis)

10. Literatura

*[**1**] E. Alexiou; T. Ebrahimi , "Exploiting user interactivity in quality assessment of point cloud imaging", 2019. Eleventh International Conference on Quality of Multimedia Experience (QoMEX), Berlin, Njemačka, 5-7 Lipanj 2019., str. 1-6, DOI:*

https://doi.org/10.1109/QoMEX.2019.8743277

*[**2**] E. Alexiou; P. Xu; T. Ebrahimi, "Towards Modelling of Visual Saliency in Point Clouds for Immersive Applications", 2019. IEEE International Conference on Image Processing (ICIP), Taipei, Tajvan, Rujan 22-25, 2019., str. 4325-4329, DOI:*

https://doi.org/10.1109/ICIP.2019.8803479

*[**3**] J. Wang, H. Zhu, Zhan Ma, Tong Chen, Haojie Liu, Qiu Shen, "Leanred Point Cloud Geometry Compression", Arxiv, str. 1-13, dostupno 3. rujna 2020, URL:*

https://arxiv.org/pdf/1909.12037.pdf

*[**4**] P. Schelkens, T. Ebrahimi, A. Gilles, P. Giola, K.-J. Oh, F. Pereira, C. Perra, A. M. G. Pinheiro, "JPEG Pleno: Providing representation interoperability for holographic applications and devices", ETRI journal, vol. 41, br. 1, veljača 2019., str. 93-108, DOI:*

https://doi.org/10.4218/etrij.2018-0509

*[**5**] S. Perry, H. P. Cong, L. A. da Silva Cruz, J. Prazeres, M. Pereira, A. Pinheiro, E. Dumic, E. Alexiou, T. Ebrahimi, "Quality evaluation of static point clouds encoded using MPEG codecs", 27th IEEE International Conference on Image Processing (ICIP 2020), listopad 2020., prihvaćen*

*[**6**] E. Alexiou, I. Viola, Tomas M. Borges, Tiago A. Fonseca, Ricardo L. De Queiroz, T. Ebrahimi, "A compreshensive study of the rate-distortion performance in MPEG point cloud compression", APSIPA Transactions on Signal and Information Processing, vol. 8, E27, studeni 2019., str. 1-27, DOI:*

https://doi.org/10.1017/ATSIP.2019.20

*[**7**] E. Alexiou, M. V. Bernardo, L. A. Da Silva Cruz, L. Gotal Dmitrović, C. Duarte, E. Dumić, T. Ebrahimi, D. Matković, "Point cloud subjective evaluation methodology based on 2D rendering", 2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX), Cagliari, Italija, 29 svibanj – 1 lipanj 2018., str. 1-14, DOI:*

https://doi.org/10.1109/QoMEX.2018.8463406

*[**8**] A. Javaheri, C. Brites, F. Pereira, J. Ascenso, "Subjective and objective quality evaluation of compressed point clouds", 2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP), Luton, UK, 16-18 listopad 2017., str. 1-6, DOI:*

https://doi.org/10.1109/MMSP.2017.8122239

*[**9**] D. Graziosi, O. Nakagami, S. Kuma, A. Zaghetto, T. Suzuki, A. Tabatabai, "An overview of ongoing point cloud compression standardization activities: Video-based (V-PCC) and geometry-based (G-PCC)", APSIPA Transactions on Signal and Information Processing, vol. 9, E13, travanj 2020., DOI:*

https://doi.org/10.1017/ATSIP.2020.12

*[**10**] H. Houshiar, A. Nüchter, "3D point cloud compression using conventional image compression for efficient data transmission", 2015 XXV International Conference on Information, Communication and Automation Technologies (ICAT), Sarajevo, Bosna i Hercegovina, 29-31 listopad 2015., str. 1-8, DOI:*

https://doi.org/10.1109/ICAT.2015.7340499

*[**11**] ScienceDirect, Point Cloud, dostupno rujna 3. 2020, URL:*

https://www.sciencedirect.com/topics/engineering/point-cloud

*[**12**] E. Dumić, A. Bjelopera, A. Nüchter, "Projection based dynamic point cloud compression using 3DTK toolkit and H.265/HEVC", Proceedings of Second International Colloquium on Smart Grid Metrology 2019 (SMAGRIMET 2019), Split, Hrvatska, 9-12 travanj 2019., str. 1-4, DOI:*

https://doi.org/10.23919/SMAGRIMET.2019.8720392

[13] D. Tian, H. Ochimizu, C. Feng, R. Cohen and A. Vetro, "Geometric distortion metrics for point cloud compression," 2017 IEEE International Conference on Image Processing (ICIP), Peking, Kina, 17-20 rujan 2017., str. 3460-3464, DOI:

https://doi.org/10.1109/ICIP.2017.8296925

*[**14**] E. Alexiou; T. Ebrahimi, "Point Cloud Quality Assessment Metric Based on Angular Similarity", 2018. IEEE International Conference on Multimedia and Expo (ICME), San Diego, Kalifornija, SAD, 23-27 srpanj 2018., str. 1-6, DOI:*

https://doi.org/10.1109/ICME.2018.8486512

*[**15**] E. Dumic, K. Sakic, L.A. da Silva Cruz, "Crowdsourced subjective 3D video quality assessment", Multimedia Systems, vol. 25, br. 6, prosinac 2019, str. 673–694 DOI:*

https://doi.org/10.1007/s00530-019-00619-7

*[**16**] E. Dumić, C. R. Duarte, L. A. da Silva Cruz, "Subjective Evaluation and Objective Measures for Point Clouds – State of the Art", Proceedings of First International Colloquium on Smart Grid Metrology 2018 (SMAGRIMET 2018), Split, Hrvatska, 24-27 travanj 2018., str. 1-4, DOI:*

https://doi.org/10.23919/SMAGRIMET.2018.8369848

*[**17**] ITU-R BT.500-14: Methodologies for the subjective assessment of the quality of television images, listopad 2019.*

*[**18**] E. Alexiou, A. M. G. Pinheiro, C. Duarte, D. Matković, E. Dumić, L. A. Da Silva Cruz, L. Gotal Dmitrović, M. V. Bernardo, M. Pereira, T. Ebrahimi: "Point cloud subjective methodology based on reconstructed surfaces", SPIE Optical Engineering + Applications, San Diego, Kalifornija, SAD, 19-23 kolovoz 2018., str. 1-14, DOI:*

https://doi.org/10.1117/12.2321518

*[**19**] E. Alexiou and T. Ebrahimi, "Towards a Point Cloud Structural Similarity Metric", 2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), London, United Kingdom, 6-10 srpanj 2020., str. 1-6, DOI:*

https://doi.org/10.1109/ICMEW46912.2020.9106005

*[**20**] d'Eon, E.; Harrison, B.; Myers, T.; Chou, P.A., "8i Voxelized Full Bodies – A Voxelized Point Cloud Dataset", Technical report, ISO/IEC JTC1/SC29/WG1 input document M74006 and ISO/IEC JTC1/SC29/WG11 input document m40059, Geneva, Switzerland, 2017.*

*[**21**] Emerging image modalities representation and compression, dostupno 3. rujna 2020., URL:*

http://emergimg.di.ubi.pt/icip2020PC.html

11. Popis slika

Slika 3.1: Struktura JPEG Pleno-a, izvor [4] 4
Slika 4.1: Prikaz podjele octree-a, izvor: google slike 7
Slika 4.2: PCL logo, izvor: google slike 8
Slika 4.3: G-PCC dijagram za referentni koder, izvor: [9] 9
Slika 4.4: Prva dva koraka octree kompresije, izvor: [9] 10
Slika 4.5: Transformacija bloka veličine 2x2x2 za octree kodiranje, izvor: [9] 10
Slika 4.6: Proces generiranja razine detalja, izvor: [9] 11
Slika 4.7: Predviđajuća transformacija, izvor: [9] 12
Slika 4.8: Lifting transformacija kreirana na već postojećoj predviđajućoj transformaciji, izvor: [9] ... 13
Slika 4.9: 3D zakrpa i njegove komponente s lijeva na desno: 3D zakrpa, mapa zauzetosti, geometrijska slika i slika tekstura Izvor: [9] 14
Slika 4.10: Generiranje 3D zakrpa, izvor: [9] 15
Slika 4.11: Primjer spremanja zakrpa, izvor: [9] 16
Slika 4.12: Pimjer prikaza udaljenosti kod kompresije slike, izvor: google slike 19
Slika 7.1: Originalni oblaci točaka prije kompresije 27
Slika 7.2: Longdress V-PCC (r01, r02, r04, r05, rb01) 27
Slika 7.3: Longdress octree (r01, r02, r04, r05, r06) 28
Slika 7.4: Longdress TriSoup (r01, r02, r04, r05, r06) 28
Slika 7.5: Loot V-PCC (r01, r02, r04, r05, rb01) 28
Slika 7.6: Loot octree (r01, r02, r04, r05, r06) 29
Slika 7.7: Loot TriSoup (r01, r02, r04, r05, r06) 29
Slika 7.8: Redandblack V-PCC (r01, r02, r04, r05, rb01) 29
Slika 7.9: Redandblack octree (r01, r02, r04, r05, r06) 30
Slika 7.10: Redandblack TriSoup (r01, r02, r04, r05, r06) 30
Slika 7.11: Soldier V-PCC (r01, r02, r04, r05, rb01) 30
Slika 7.12: Soldier octree (r01, r02, r04, r05, r06) 31
Slika 7.13: Soldier TriSoup (r01, r02, r04, r05, r06) 31
Slika 7.14: Ricardo V-PCC (r01, r02, r04, r05, rb01) 31
Slika 7.15: Ricardo octree (r01, r02, r04 ,r05, r06) 32
Slika 7.16: Ricardo TriSoup (r01, r02, r04, r05, r06) 32
Slika 7.17: Sarah V-PCC (r01, r02, r04, r05, rb01) 32
Slika 7.18: Sarah octree (r01, r02, r04, r05, r06) 33
Slika 7.19: Sarah TriSoup (r01, r02, r04, r05, r06) 33
Slika 8.1: Rezultati za longdress (UNIN-lijevo, UC-desno) 34

Slika 8.2: Rezultati za loot (UNIN-lijevo, UC-desno) 34
Slika 8.3: Rezultati za redandblack (UNIN-lijevo, UC-desno) 35
Slika 8.4: Rezultati za ricardo (UNIN-lijevo, UC-desno) 35
Slika 8.5: Rezultati za sarah (UNIN-lijevo, UC-desno) 35
Slika 8.6: Rezultati za soldier (UNIN-lijevo, UC-desno) 36
Slika 8.7: Moguće komande unutar programa Project1.exe 38
Slika 8.8: Rezultati prvog učitavanja i uspoređivanja koristeći program "Project1.exe" 39
Slika 8.9: Hausdorff p2point rezultati (UNIN-lijevo, UC-desno) 48
Slika 8.10: Hausdorff p2plane rezultati (UNIN-lijevo, UC-desno) 48
Slika 8.11: Hausdorff PSNR p2point rezultati (UNIN-lijevo, UC-desno) 48
Slika 8.12: Hausdorff PSNR p2plane rezultati (UNIN-lijevo, UC-desno) 49
Slika 8.13: RMSE p2point rezultati (UNIN-lijevo, UC-desno) 49
Slika 8.14: RMSE p2plane rezultati (UNIN-lijevo, UC-desno) 49
Slika 8.15: RMSE PSNR p2point rezultati (UNIN-lijevo, UC-desno) 50
Slika 8.16: RMSE PSNR p2plane rezultati (UNIN-lijevo, UC-desno) 50